TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

HEIKKI EUROPAEUS

# DEVELOPING A DIMENSIONING MODEL FOR NETACT PERFORMANCE TESTING

Master of Science thesis

Examiner: prof. Jarmo Harju
Examiner and topic approved by the Faculty Council of the Faculty of Communication Engineering
on 14th January 2015

## ABSTRACT

**HEIKKI EUROPAEUS**: Developing a dimensioning model for NetAct performance testing
Tampere University of technology
Master of Science Thesis, 49 pages
February 2015
Master's Degree Programme in Information Technology
Major: Communication networks and protocols
Examiner: Professor Jarmo Harju

Keywords: OSS, NetAct, dimensioning, performance testing, testing, LTE, 3G, 2G

In recent years, mobile technology has taken huge steps in equipment as well as network-side development. This has made possible to develop new type of mobile services and applications. Therefore the requirements for data rates have been increasing rapidly and this has created a need for even more efficient network monitoring and management tools. Nokia's solution for network management is called NetAct operations management system. The research problem in this master's thesis is the quality of dimensioning models used in NetAct performance testing. The currently used dimensioning model in testing is based on expert estimates of operator networks and there is no information of the reliability of the dimensioning model.

Solving the dimensioning problem started by collecting actual operator data used by the customer support team. The operator data included information about the number of network elements and their status. When enough material was collected from the operator networks, the material was put into an Excel file. This time-consuming operation gave a motivation for writing automation program for data gathering. The automation program was written with Java. From the compiled Excel file, ratios for the element types and the minimum, average, median and maximum values for the number of network elements could be calculated. The dimensioning model is based on these network element ratios and calculated minimum, average, median and maximum values.

The automation significantly speeded up the gathering of data from the operator files to the Excel file; this also eased the collection work. The collected operator data revealed that the currently used dimensioning model represents operator networks fairly accurately, but it also revealed that the testers test the NetAct with oversized networks compared to the network size that operators really have. The developed dimensioning model represents the operator networks more accurately, and therefore it enables more accurate performance testing of NetAct.

# TIIVISTELMÄ

**HEIKKI EUROPAEUS**: Developing a dimensioning model for NetAct performance testing
Tampereen teknillinen yliopisto
Diplomityö, 49 sivua
Helmikuu 2015
Tietotekniikan diplomi-insinöörin tutkinto-ohjelma
Pääaine: Tietoliikenneverkot ja protokollat
Tarkastaja: professori Jarmo Harju

Avainsanat: OSS, NetAct, matkapuhelinverkon mitoitus, suorituskyvyn testaus, testaus, LTE, 3G, 2G

Viime vuosina matkapuhelinteknologia on saavuttanut suuria edistysaskeleita niin laitteiden kuin verkonkin puolella. Tämä on mahdollistanut myös uusien mobiilipalveluiden ja –sovellusten kehityksen. Tästä johtuen verkon datamäärät ovat kasvaneet huomattavasti ja on tullut tarve hallita ja monitoroida matkapuhelinverkkoa entistä tehokkaammin. Nokian tarjoama ratkaisu verkonhallintaan on NetAct-järjestelmä. Tutkimusongelma tässä diplomityössä on NetActin suorituskyvyn testauksessa käytetty matkapuhelinverkon mitoitusmalli. Ongelman taustalla on se, että aiemmin testauksessa käytetty mitoitusmalli on perustunut asiantuntijoiden luomaan arvioon operaattoriverkoista ja mallin luotettavuudesta ei ole ollut tarkkaa tietoa.

Verkon mitoitusongelmaa lähdettiin ratkaisemaan keräämällä asiakastukitiimin käyttämää operaattoriverkon dataa, joka pitää sisällään verkon elementtien lukumäärän ja tilan. Kun dataa oli tarpeeksi kerätty operaattoriverkoista, lähdettiin Excel-tiedostoa koostamaan käsin datasta. Käsin koostamisen hitauden takia, tuli motivaatio automatisoida Excel-tiedoston koostaminen. Automatisointi tapahtui käyttäen Java-ohjelmointikieltä. Koostetusta Excel-tiedostosta, pystyttiin laskemaan verkkoelementtien suhteet ja lukumäärien minimit, keskiarvot, mediaanit ja maksimit. Verkon mitoitusmalli pohjautuu verkkoelementtien suhteisiin, referenssidataan ja laskettuihin minimiarvoihin, keskiarvoon, mediaaniin ja maksimiarvoihin.

Tiedon koostamisen automatisoinnilla Excel-tiedoston luonti nopeutui huomattavasti samalla helpottaen keräämistyötä. Kerätty operaattoridata osoittaa, että suorituskyvyn testauksessa käytetty mitoitusmalli on hyvin lähellä sitä, millaisia verkkoja operaattoreilla on käytössään. Toisaalta operaattoridata osoittaa myös, että testauksessa testataan verkkoja, joiden koko on moninkertainen verrattuna todellisuuteen. Työssä kehitetty mitoitusmalli on lähempänä operaattoreiden käyttämiä verkkoja, kuin nykyinen käytössä oleva malli, ja näin ollen se mahdollistaa tarkemman NetActin suorituskyvyn testauksen.

# PREFACE

This thesis has been created for Nokia by the request of NetAct performance testing team. Most of the material used in this thesis is gathered from the Nokia intranet and by interviewing performance testing team.

I would like to thank the team for the help and give special thanks for the following persons: Melinda Simon, Maria Lahti, Timo Satola, Timo Kuusela, Jarmo Harju, Jukka Mannila and Jyri Heinonen.

Tampere, 17.2.2015

Heikki Europaeus

# CONTENTS

## LIST OF SYMBOLS AND ABBREVIATIONS

| | |
|---|---|
| 2G | Second generation of mobile telecommunications technology |
| 3G | Third generation of mobile telecommunications technology |
| 4G | Fourth generation of mobile telecommunications technology |
| 5G | Fifth generation of mobile telecommunications technology |
| AAA | Authentication, Authorization and Accounting |
| API | Application Programming Language. |
| AuC | Authentication Center |
| BG | Border Gateway |
| BSC | Base Station Controller |
| BTS | Base Transceiver Station |
| CGN | Charging Gateway Node |
| CLI | Command Line Interface |
| CM | Configuration Management |
| CN | Core Network |
| CPU | Central Processing Unit |
| CS | Circuit Switched |
| CS-MGW | Circuit Switched Media Gateway |
| DB | Database |
| EDGE | Enhanced Data rates for GSM Evolution |
| EIR | Equipment Identification Register |
| eNodeB | Name of BTS in 4G |
| FM | Fault Management |
| GERAN | GSM EDGE Radio Access Network |
| GGSN | Gateway GPRS Support Node |
| GMSC | Gateway MSC |
| GPRS | General Packet Radio Service |
| GSM | Global System for Mobile Communications |
| HAT | High Availability Testing |
| HLR | Home Location Register |
| I/O | Input / Output |
| KPI | Key Performance Indicator |
| LNBTS | Same as eNodeB |
| LNCEL | Cell in eNodeB |
| LTE | Long Term Evolution |
| M2M | Machine-to-Machine |
| MIMO | Multiple Input Multiple Output |
| MME | Mobility Management Entity |
| MSC | Mobile Switching Center |
| NodeB | Name of BTS in 3G |
| OSS | Operations Support System |
| PDN GW | Packet Data Networks Gateway |
| PET | Performance Testing |
| PI | Performance Indicator |
| PM | Performance Management |
| PS | Packet Switched |
| PSTN | Public Switched Telephone Network |
| RAN | Radio Access Network |

| | |
|---|---|
| RNC | Radio Network Controller |
| SGSN | Serving GPRS Support Node |
| S-GW | Serving Gateway |
| SIM | Subscriber Identity Module |
| SM | Security Management |
| SON | Self-Organizing Network |
| UE | User Equipment |
| UMTS | Universal Mobile Telecommunications System |
| UTRAN | UMTS Terrestrial Radio Access Network |
| VM | Virtual Machine |
| WBTS | Same as NodeB |
| WCDMA | Wideband Code Division Multiple Access |
| WCEL | Cell in NodeB |

# 1. INTRODUCTION

The purpose of this study is to develop a dimensioning model based on operator data for NetAct performance testing. NetAct is an operations support system (OSS) developed by Nokia for managing radio networks.

In performance testing testers are verifying the capability of NetAct to manage operator networks of different sizes. At the moment, the dimensioning models used in performance testing are based on expert estimates of mobile operators' network sizes. Incorrect estimates can cause a situation where NetAct is tested with over- or underestimated dimensioning values and eventually lead to biased test results. By developing a dimensioning model based on actual operator data for the testers, the testers can overcome the problem and achieve more accurate test results.

The theoretical part of the thesis covers the first five sections of the thesis. The NetAct section of the thesis gives a brief introduction to NetAct and what kinds of modules are important for the NetAct system. In mobile operators section, the different radio network technologies are discussed. This section introduces adjacencies of network elements as one topic. The other topics covered by the mobile operators section are the core network, 2G, 3G and LTE. The fourth section discusses different factors that effect on NetActs system performance. The factors discussed in this section include network elements, used hardware, user actions and databases. In performance testing section the basic topics of testing and tested metrics are introduced. This section also gives an introduction to the methods of testing and test automation while giving a preface for the practical part of the thesis by introducing the dimensioning concept in radio network planning and NetAct point of view.

The practical part of the thesis begins from the data analysis section. The data analysis section introduces the basis for the dimensioning model by discussing raw support data and how it has been analyzed for the thesis. The dimensioning model section introduces the data compiler software for gathering the operator data and the new dimensioning model for NetAct performance testing. The future challenges section introduces the possibilities and challenges brought by 5G technology. The challenges are viewed from the NetAct point of view and how they are going to effect on NetAct performance testing.

# 2. NETACT

According to the Nokia NetAct™ operating documentation, NetAct is a network management system for multi-vendor and multi-technology networks; in other words, it is an operations support system (OSS). In practice NetActs combines the management of overall network operations, individual network elements and services [1].

NetAct is a modular system, where customer can activate features based on his or her needs. NetAct modules can be divided to four major categories: configuration, fault, performance and security management. NetAct is used with a graphical user interface and its start page can be seen in Figure 1.

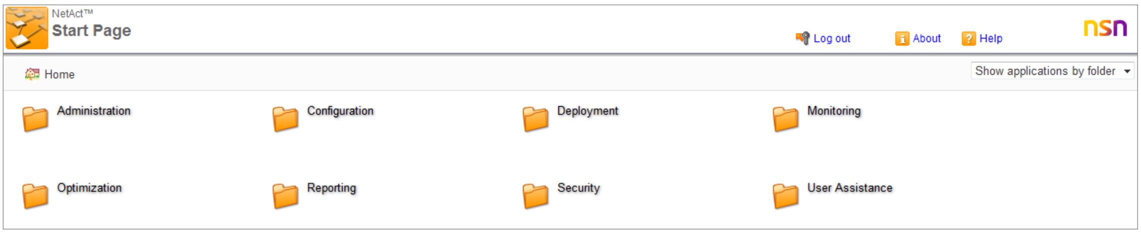


***Figure 1.*** *NetAct Start Page*

Start page provides a central access point to NetAct applications and from there the user can navigate further to configure and manage the network and its elements [1].

## 2.1 CM – Configuration management

The basic function of configuration management (CM) is to configure and manage the radio and core networks. Configuration management enables network-wide operations, which include single parameter changes in network elements, managing hardware data, or launching new technologies and services.

In configuration management there are five major applications and their aim is to provide automated tools for managing network configuration data, provisioning changes, and correcting inconsistencies. These five applications are [2]:

- **CM Analyzer** for checking the consistency of the different networks, network elements and parameter configurations handled by the system.
- **CM Editor** for managing network elements and configurations.
- **CM Operations Manager** for scheduling different operations supported by the system. CM Operations Manager also provides real-time feedback on the progress of the operations and history information on the executed operations.

- **CM Reference** for managing reference configurations that represent the planned network configurations.
- **SON Scheduler** for viewing and configuring operations in the LTE Flexi Multi-radio base transceiver station (BTS) elements. These operations include auto-connection and auto-configuration features both for LTE and WCDMA.

An example of one of the applications can be seen in Figure 2. It is the main window of CM Editor, where the user is editing LTE related parameters in multiradio base station (MRBTS).

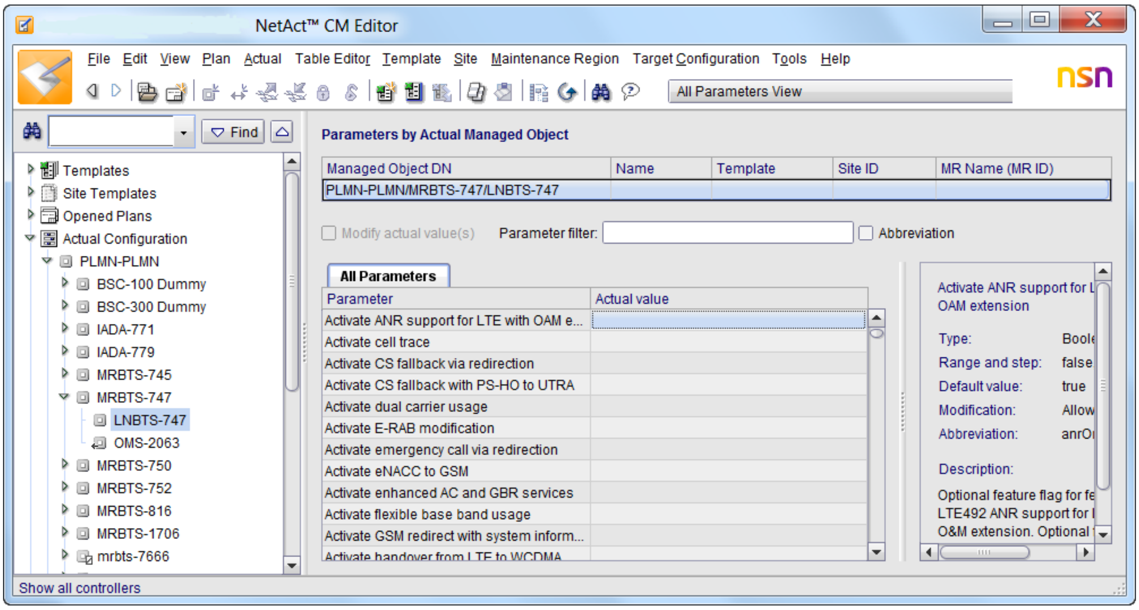


***Figure 2.*** *NetAct CM Editor*

## 2.2 PM – Performance management

Performance Manager (PM) is a set of applications for processing, analyzing and visualizing performance data coming from different sources. More specifically, PM is multi-vendor-capable and collects data from the entire network that consists of network elements. Multivendor-capable means that PM is capable to handle seamlessly network elements manufactured by different manufacturers. The aim of any PM activity is to collect data to support the following activities [3]:

- Verifying the physical and logical configuration of the telecommunications network
- Monitoring continuously how the network functions
- Localizing potential problems as early as possible
- Monitoring subscriber behavior
- Providing optimum services to mobile subscribers.

PM applications can be divided to two categories depending how the applications use the collected data. These categories are called performance monitoring and performance reporting. Example of performance reporting can be seen in Figure 3. It illustrates NetAct Report Creator Wizard in action.

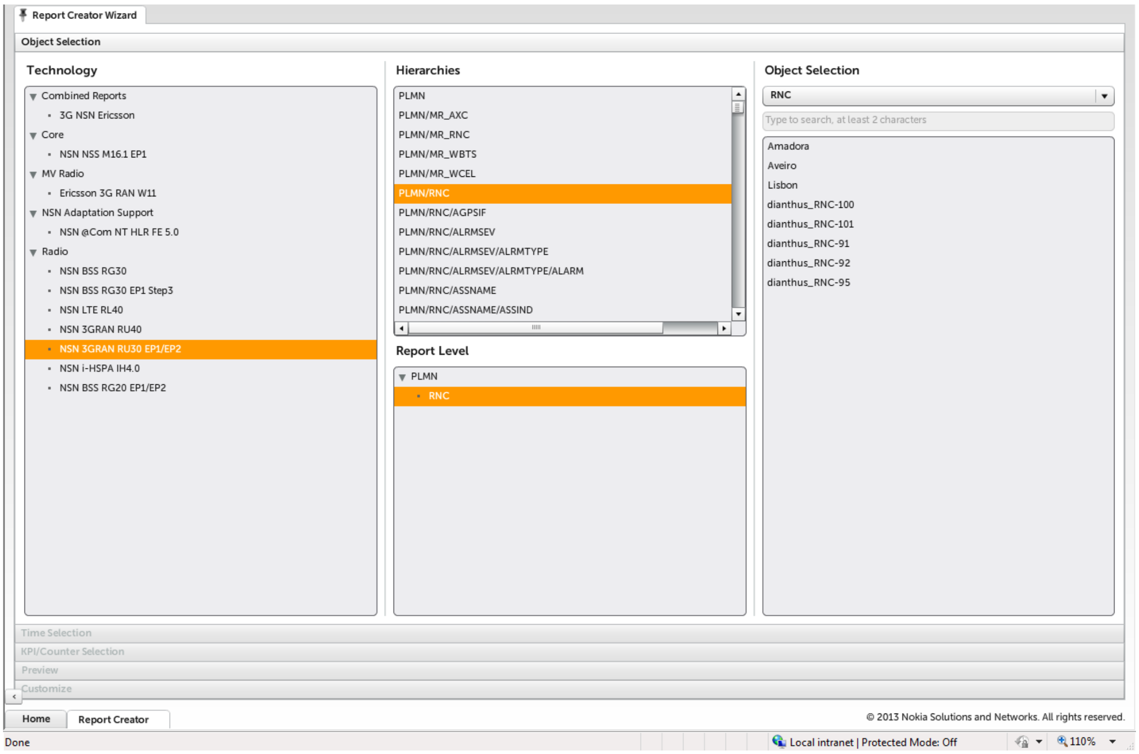


***Figure 3.*** *NetAct Performance Manager*

Performance monitoring applications are online-oriented, meaning that they provide real-time information on the network. The purpose of these applications is to provide additional information with short measurement output interval for problem cases where no alarm information is available. The measurement output interval determines how often and when measurement results are transferred from the network element to the NetAct database

Performance reporting applications are offline-oriented and they provide information on what happened in the network over a certain period of time, so that the user can afterward check what caused the issue. These applications use counters, Key Performance Indicators (KPI) and produce reports. Performance reports can be used in troubleshooting, network planning or optimizing [3].

## 2.3 FM – Fault management

The purpose of fault management (FM) is to detect, isolate and correct malfunctions in a network as soon as they are detected by the system. Fault management consists of two applications, Monitor and Alarm Reports Dashboard.

The Monitor application enables users to:

- Collect, process, store, and display alarm information from the network in real-time
- Visualize the network topology
- Detect and analyze faults in network elements.

The target of Alarm Reports Dashboards is to detect, collect, and store the failures in the network in a way that the users can retrospectively analyze the occurred faults and generate reports from the faults. Together with these applications the operator can troubleshoot what has caused the alarm in the network and what could be done for fixing the situation [4].

## 2.4 SM – Security management

The security management (SM) target is to manage and enforce security-related information and policies in NetAct. Generally these security management related functions can be divided into four main areas:

- **System hardening**, meaning that the unauthorized internal and external use of the system is prevented by removing unnecessary services and usernames.
- **User security**, consisting of authentication, authorization, and user event logging.
- **Network security** means the protection of the traffic in a network where NetAct is used. Network security includes traffic access control, monitoring and protection, including encryption.
- **Security supervision** is performed through logging and tracing.

These four areas also follow the international guidelines of confidentiality, integrity and availability [5].

# 3. MOBILE OPERATOR NETWORKS

The mobile networks have developed dramatically during the last 20 years and the change is ongoing. Figure 4 shows development in global connections by technology during six years and prediction for near future. The figure doesn't take account to 5G or machine-to-machine (M2M) communications, but shows the trend where older technologies are losing markets for newer and faster ones are gaining more market share, for example the 2G connections in 2020 is predicted to decrease to 3.2 billion [6]. High number of connections is explained by multiple subscriber identity module (SIM) ownership.

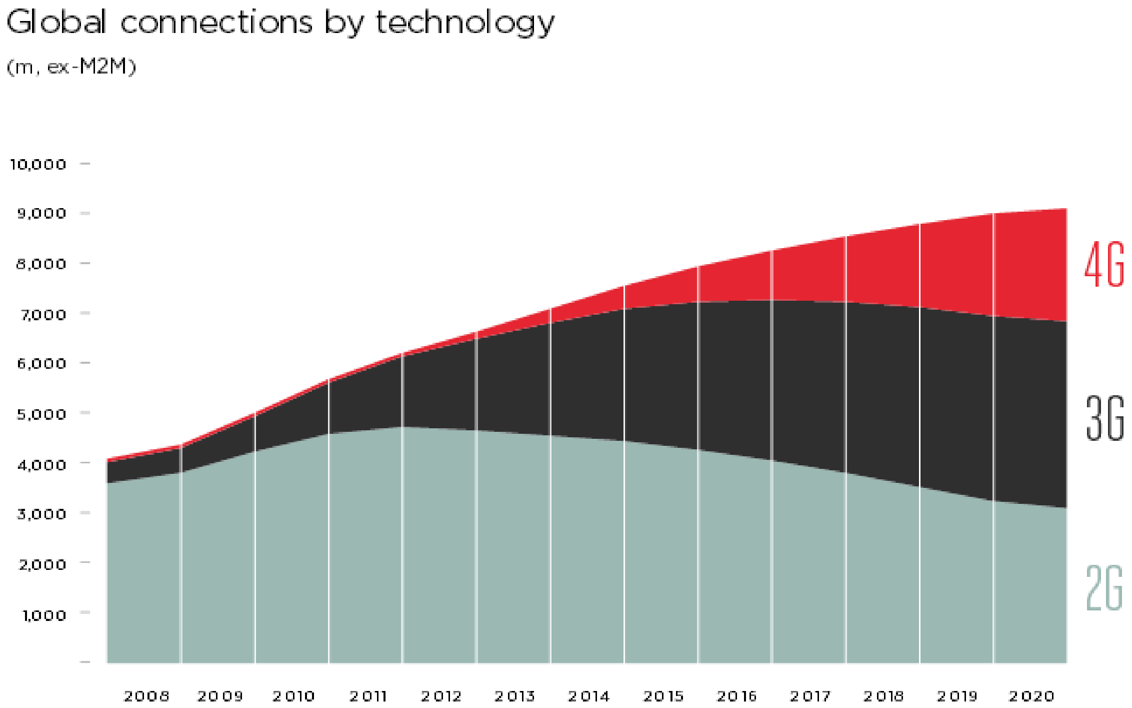


***Figure 4.*** *Global connections by technology [6]*

A mobile operator network consists of different elements for providing telecommunication service to its customers. Examples of telecommunication operators are often multinational companies, like Vodaphone or TeliaSonera. The operator network itself is formed from several different network elements and core networks. These will be discussed in more detail in the following sections.

It is useful for a NetAct tester or anyone who works with radio networks to understand the differences and functionalities between different radio access technologies. This way he or she can form the big picture about the radio networks and understand how different network elements affect each other and what the differences are between different radio access technologies.

## 3.1 Network elements

There are several network elements in the radio network and these elements can be categorized to radio access network (RAN) and core network. RAN differs a little in each technology generation, but mainly it could be said that it includes base stations and controller elements for the base stations. Shortly the core network includes gateway elements for accessing traditional public switched telephone network (PSTN) or the internet. Access to PSTN happens through circuit switched (CS) core network (CN) where the connection forms an end-to-end circuit between the callers. Access to the Internet and other IP-based (Internet Protocol) services is routed through packet switched (PS) core network, where all the information is transferred in IP-packets [7].

Network elements have seen huge development during the last twenty years, from large and dedicated devices to smaller and more universal devices. In future plans this means that most parts of the network are going to be virtualized to cloud servers. The goal of this development direction is to cut the network costs and improve scalability. Nokia's answer for this future is called liquid net; it consists of liquid broadband, liquid radio and liquid core. In practice this means the virtualization of core network and network elements [8][9].

### 3.1.1 2G - GERAN

2G is a term that means the second generation of cellular technologies and it covers the whole network architecture from radio access to core network. The acronym GERAN is more specific and stands for GSM/EDGE Radio Access Network [10]. In practice this means the technology between mobile terminal and the base station. The GERAN architecture is illustrated in Figure 5, where we see the Base Transceiver Station (BTS) and the Base Station Controller (BSC) elements. Core network elements are discussed later in the core network chapter. The following list describes GERAN elements [11]:

- **BTS** contains transmitter and receiver equipment, such as antennas and amplifiers.
- **BSC** contains protocol functions for radio channel allocation, channel setup and management of handovers. Typically one BSC handles hundreds of BTSs.

2G contains three major technology categories; GSM, GPRS and EDGE. The GSM technology is dedicated to speech calls and GPRS and its successor EDGE to packet data. While these technologies start to be outdated, GSM still is one of the best options for speech, mainly because of its wide cell coverage and simple implementation.

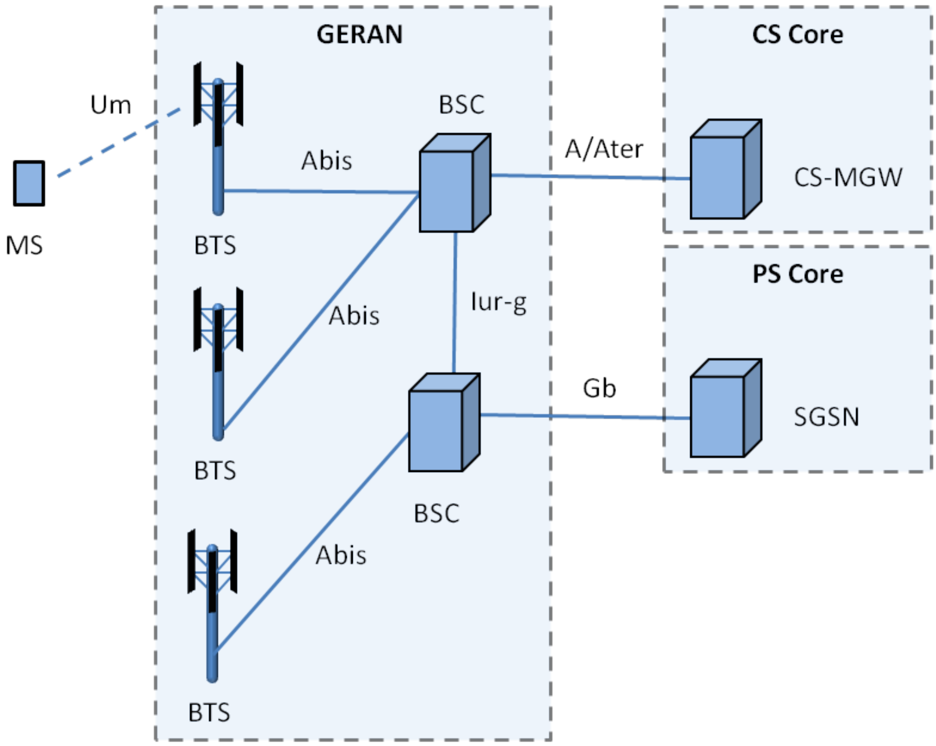


***Figure 5.*** *GERAN system architecture*

Communication relationships between the GSM network components are described by a number of standardized interfaces. These interfaces can be used for managing the data transfers or controlling the connections in mobility management. The interfaces of GERAN system are listed below [10]:

- **Um**, an interface between MS and BTS, target to provide access to the network.
- **Abis**, an interface between BTS and BSC. It is used for controlling the radio equipment and radio frequency allocation in the BTS. The Abis interface also carries synchronization information from the BSC to the BTS and MS.
- **Iur-g**, an interface between two BSCs.
- **Gb**, an interface between GERAN and SGSN. The Gb interface enables communication between SGSN and MS.
- **A/Ater**, A is an interface between BSC and CS-MGW. It is used for signaling and carrying traffic channels. Ater is an interface between BSC and transcoder, it carries the A interface information from the BSC leaving it untouched [12].

### 3.1.2 3G - UTRAN

3G means the third generation of mobile cellular technology developed by the 3GPP organization. The goal of 3G was to offer a greater spectral efficiency and bandwidth for growing telecommunication markets. 3G is also called the Universal Mobile Telecommunications Systems (UMTS) and its radio access goes under the acronym UT-

RAN, UMTS Radio Access Network [13]. Similarly as in GERAN, UTRAN is the radio technology used between mobile terminal and base stations.

3G offered a huge improvement in data rates when compared to 2G, but it also increased complexity on the network planning side. The reason for complexity is that in 3G, every user is generating interference for the other users, because all share the common frequency band. Another 3G-specific thing is cell breathing, this means that the cell size is increasing and decreasing depending on the number of users in the cell [14].

UTRAN network architecture and its elements are illustrated in Figure 6. The UTRAN network elements are described in the following list [15]:

- **NodeB** means the same as the base station. Its task is to convert the data flow between Iub and Uu interfaces and participate to radio resource management.
- **Radio Network Controller** (RNC**)** owns and controls the radio resources in its domain. It also works as a connection point for core network services.

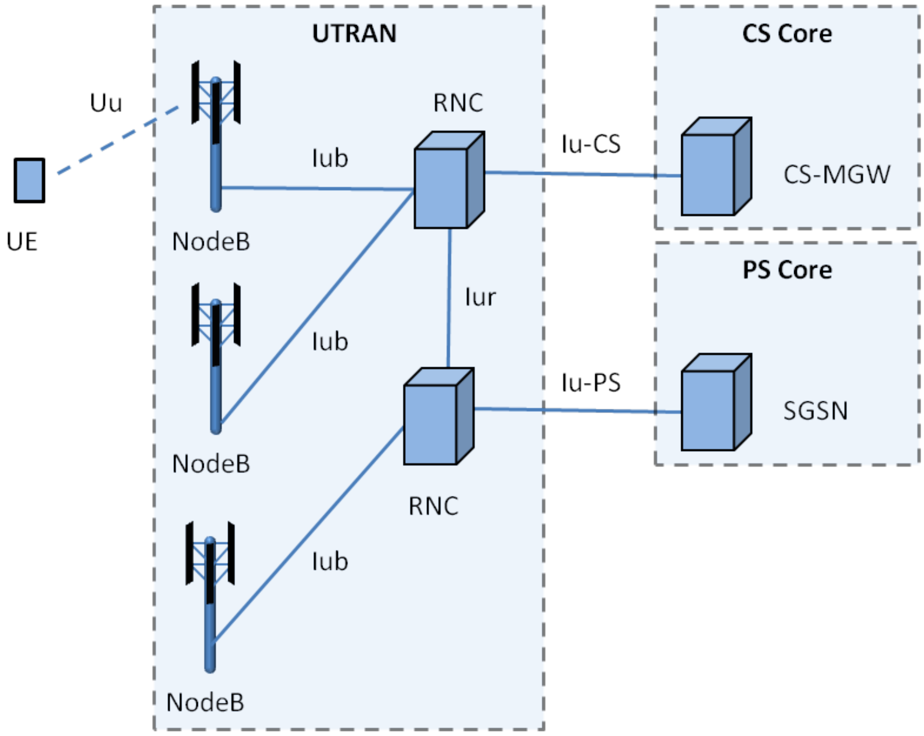


***Figure 6.*** *UTRAN system architecture*

The control plane in UTRAN includes the application protocols and the signaling bearers, which transport the control information [16]:

- **Uu**, is the WCDMA radio interface, through Uu UE can access the network.
- **Iub**, is a logical interface between a NodeB and a RNC.
- **Iur**, is a interface that enables soft handovers between RNCs from different manufacturers.

- **Iu-CS**, connects UTRAN to CS core network and enables circuit switched mobile calls.
- **Iu-PS**, connects UTRAN to PS core network and enables access to Internet.

### 3.1.3 4G - LTE

Long Term Evolution or more commonly known with its abbreviation LTE is the fourth generation mobile cellular technology. The goal of LTE has been to provide even higher data rates than 3G and improve the spectrum efficiency while lowering the network operation costs. As a difference to older technologies, in LTE all the data is IP-based packet data, which also means that LTE does not have Circuit Switched (CS) Core. [17] At the moment this also creates a drawback for LTE. When a user establishes or receives a call in an LTE network, the mobile device has to do a CS-Fallback procedure and switch back to 2G or 3G network to be able to do so. In future this is changing with the Voice over LTE (VoLTE) technology which enables IP-based calls in the LTE network [17].

LTE introduced Evolved Packet Core (EPC) which should handle the data traffic efficiently in terms of performance and costs while having a simplified flat architecture. One of the major improvements in EPC is the separation of the user and the control planes, which makes the network scaling more independent and easier for the operators [7].

The LTE network and EPC can be seen in Figure 7, where there are following elements [18]:

- **Evolved NodeB** (eNodeB) is the LTE element equivalent to BTS and NodeB, but differs in the sense that eNodeB has radio resource control, radio mobility management and full layer 2 protocol support features.
- **Mobile Management Entity** (MME) is used for the control plane functions related to subscriber and session management.
- **Serving Gateway** (S-GW) is used as a connection point of the packet data towards Evolved UMTS Terrestrial Access Network (E-UTRAN). In practice this means that the S-GW enables mobility between E-UTRAN and other 3GPP technologies.
- **Packet Data Network Gateway** (PDN GW) is functioning similarly as S-GW and works as a termination point of the packet data interface towards the packet data network.

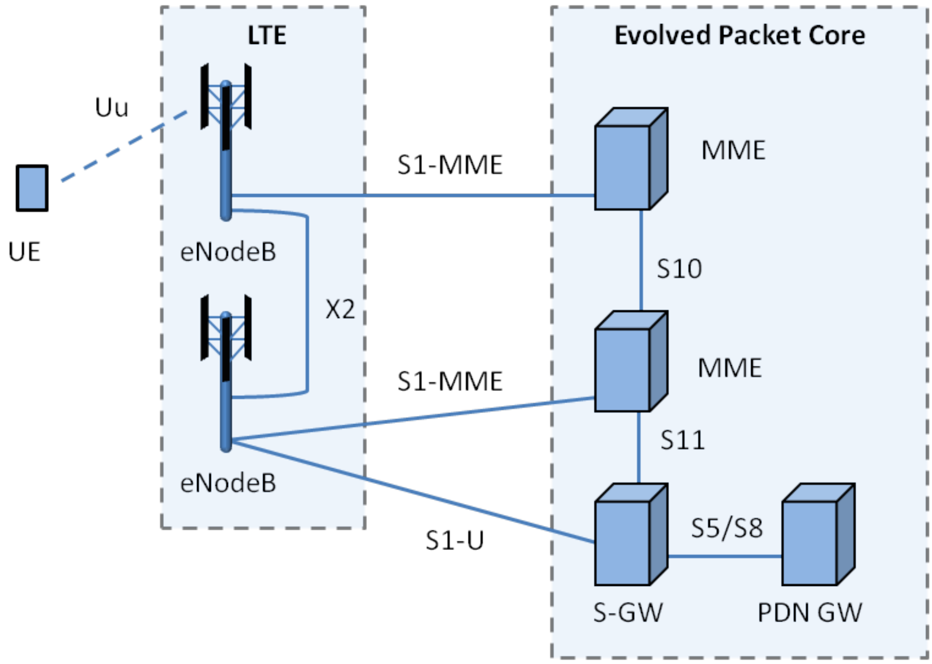


***Figure 7.*** *LTE system architecture*

Understanding the control plane interfaces in LTE architecture is as important as understanding the function of the elements in the network. The control plane interfaces describe functionality between two network elements and ease the understanding of relations in the network. The LTE control plane interfaces are illustrated in Figure 7 [19]:

- **S10** is an interface between MME and another MME; it is used between MMEs for MME relocation and MME to MME information transfer.
- **S11** is an interface between MME and SGW, its target is to function as a reference point between MME and Serving GW.
- **S5 or S8** are interfaces between SGW and PGW. S5 provides user plane tunneling and tunnel management between SGW and PGW. S8 is an inter-PLMN reference point providing the user and control planes between the SGW in the visitor PLMN and the PGW in the home PLMN.
- **S1-U** is an interface between eNodeB and SGW, and it is used between E-UTRAN and Serving GW for the per bearer user plane tunneling and inter eNodeB path switching during handover.
- **S1-MME** is an interface between eNodeB and MME, it is a reference point for the control plane protocol between E-UTRAN and MME.
- **X2** is an interface between two eNodeBs, it can be used for signaling and handling radio resources between the base stations.
- **Uu** is an interface between eNodeB and UE, its target is to provide data transfer between eNodeB and UE.

### 3.1.4 Core network

The core network (CN) means the backbone of the telecommunication networks. It has also evolved during the technological advancement from 2G to LTE and the direction at the moment is towards the liquid core, meaning virtualization and cloud services. The key features of core networks are aggregation, authentication, call control or switching, charging, service invocation and gateways to other services [20].

As discussed earlier in The LTE chapter, LTE has introduced an evolved packet core and by doing so, it has simplified the core network architecture. When comparing 3G and 2G technologies, the GSM core architecture relied on circuit switching until the GPRS was introduced; it added packet switching to circuit switching, which enabled the transportation of packets without establishing dedicated circuits. When UMTS was released, it evolved some network elements but mainly it kept this dual-domain concept in the core network. Circuit and packet domains are illustrated in Figure 8 [21].

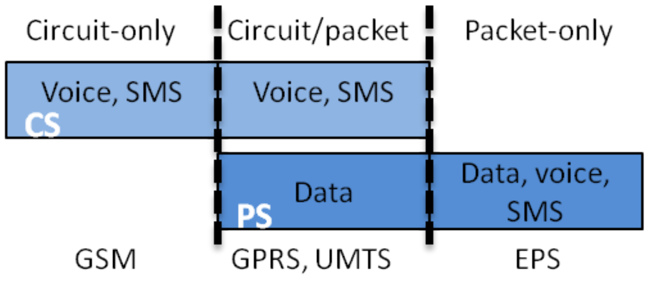


***Figure 8.*** *Circuit and packet domains [21]*

In Figure 9 we can see the following registers and PS/CS core network elements [14][20]:

- **Circuit Switch Media Gateway** (CS-MGW) is a node which handles CS connection capacity and handles all physical connection matters.
- **Mobile Switching Center** (MSC Server) is a switch that serves the UE in its current location for circuit-switched services. MSC also contains visitor location register (VLR) database.
- **Gateway MSC** (GMSC Server) is the switch at the point where public land mobile network (PLMN) is connected to external circuit switched networks. All incoming and outgoing circuit switched connections go through GMSC.
- **Authorization, Authentication and Accounting** (AAA) is a register that authenticates and authorizes the subscriber's network access. AAA is also responsible for subscriber billing.
- **Equipment Identity Register** (EIR) contains information about terminal equipment and can be used for blocking specific terminals from network.

- **Home Subscriber Server / Authentication Center** (HSS/AuC) presents the registers such as home location register (HLR). The function of HSS is to provide information about user's service priorities and data rates. AuC part of the HSS is used for generating security information from user identity keys, which then is used for network-terminal authentication.
- **Serving GPRS Support Node** (SGSN) has a similar functionality as MSC but is used for packet switched services.
- **Charging Gateway Node** (CGN) collects charging data from PS domain elements and relays them to the billing centre to be post-processed.
- **Gateway GPRS Support Node** (GGSN) PS counterpart for GMSC.
- **Border Gateway** (BG) is a gateway which enables roaming between two separate PS domains belonging to separate network.
- **Public Switched Telephone Network** (PSTN) means the traditional wired telephone network.

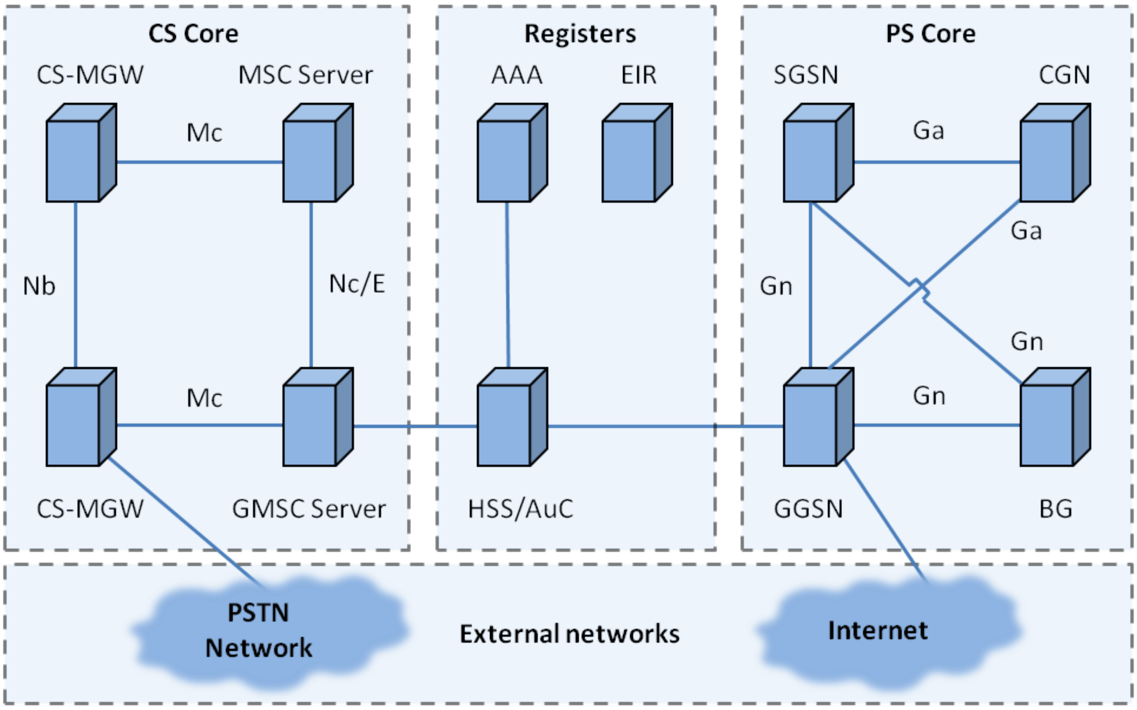


***Figure 9.** Registers, CS and PS Core architecture*

Interfaces in the core network describe the connection or function between two different elements in a similar way as with radio networks. Generally interfaces in core networks are used for control purposes [22][23]. Interfaces are illustrated in Figure 9 and explained in the following list:

- **Nb**, an interface between two media gateways. It is used for bearer control and transport.

- **Mc**, an interface between an MSC or TSC and its controlled MGWs. It is used for the separation of call control entities from bearer control entities and vice versa.
- The **Nc/E** interfaces are between two MSCs or TSCs. An Nc interface is used for network-to-network based call control. The E interface is used for handover purposes when mobile user is changing or moving from one MSC area to another during the call.
- The **Ga** interface is between GSN and CGN. It is used for offline charging purposes.
- **Gn**, an interface used to carry signaling and data traffic between GSNs.

## 3.2 Network adjacencies

Network adjacencies mean in practice that a network element is next to another network element and they have some kind of relation. Adjacency can be illustrated with a graph like in Figure 10, in which BTS A is adjacent to BTSes B and C. In the network, an adjacency can be used for controlling purposes, i.e. handovers between two technologies or between used frequencies.

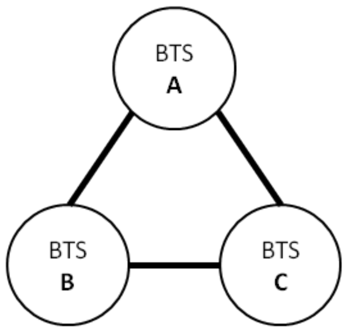


***Figure 10.*** *Undirected adjacency graph example*

Both 3G and LTE have several different adjacencies, and in both technologies there are different terms for describing the adjacencies. In this thesis the emphasis is on the following adjacencies [24][25]:

- **ADJS**, Intra-frequency adjacency, used for controlling intra-frequency handovers. Intra-frequency handovers in WCDMA are soft handovers.
- **ADJI**, Inter-frequency adjacency, used for controlling inter-frequency handovers. Inter-frequency handovers in WCDMA are hard handovers.
- **ADJD**, Additional adjacent cell.
- **ADJG**, Inter-system adjacency, used for controlling handovers from WCDMA to GSM.
- **LNADJ**, neighbouring LTE BTS.
- **LNADJL**, neighbouring LTE BTS cell.

- **LNADJW**, neighbouring WCDMA BTS cell.
- **LNADJG**, neighbouring GERAN BTS cell.
- **LNADJX**, neighbouring CDMA 1xRTT BTS cell.
- **LNADJT**, neighbouring TD-CDMA BTS cell.
- **LNREL**, LTE neighbour relation.
- **LNRELG**, LTE neighbouring GERAN relation.
- **LNRELW**, LTE neighbouring WCDMA relation.
- **LNRELX**, neighbouring CDMA 1xRTT BTS cell.
- **LNRELT**, neighbouring TD-CDMA BTS relation.

These adjacencies are important for NetAct performance testing, because adjacencies determine the dimensioning of the simulated network.

## 3.3 Network traffic in telecommunications and in NetAct

Traffic in telecommunications can be seen in many ways, but typically it's measured with a dimensionless unit, Erlang. Erlang represents the continuous use of one voice path and it is the total traffic volume during one hour [26].

From mobile operator's point of view, network traffic can be defined as user actions (calls, SMS, web browsing, etc.) and signaling between the UE and the network (user and control plane actions).

In radio network planning, network traffic can be seen as part of capacity planning, where the planner takes into account the available spectrum, subscriber growth and traffic density information. The available spectrum effects on bandwidth that the cell has in its use and thus how many subscribers it can handle efficiently. Traffic density describes how much traffic is generated in one cell or sector and it includes the call blocking probability. The traffic density is measured in Erlangs and it comes from the derivation of arriving and departing calls per hour [14].

NetAct has a different point of view on network traffic. NetAct sees the network traffic as a PM, FM and CM data, which it receives through southbound interfaces. Southbound interface can be seen in Figure 14. These interfaces are used for integrating lower-level systems and network elements into NetAct. In other words this means that the network traffic in NetAct is seen as events like network failure, configuration change or network performance change [27].

### 3.3.1 PM data

Performance Management data or shortly PM data consists of measurements collected by counters in different network elements. Measured information can consist of HO, success rate, events, resets, resource usage, signaling, etc. The PM data is essential for evaluating the behavior of any PLMN system [28].

The measurement information can be directly uploaded to the NetAct database, or it can be preprocessed in the network element or post-processed. Preprocessing depends on the element type and the configuration of the element. If the data is preprocessed, it either goes through a filtering process or some calculation before it is uploaded to NetAct through the element-specific interface. In post-processing the data undergoes further processing, so that, for example, it can be used for reporting busy hour information. Figure 11 illustrates the described measurement data flow in the radio access network [28].

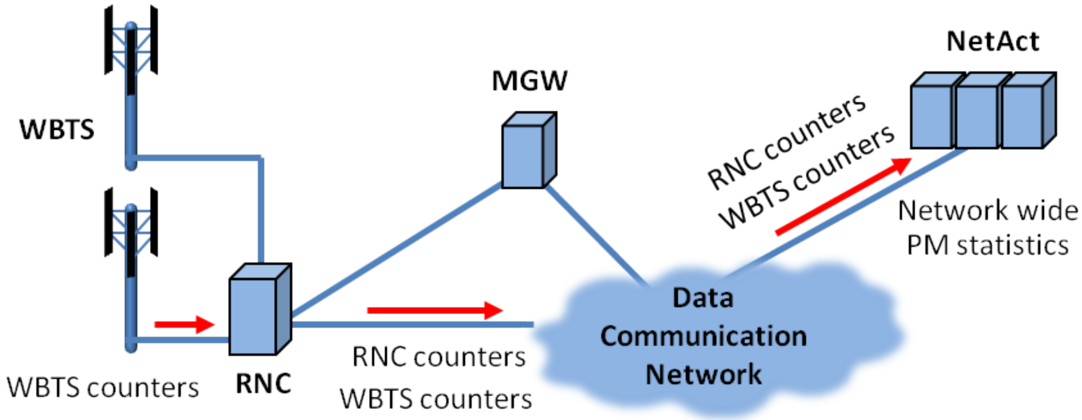


***Figure 11.*** *RAN measurement data flow*

Collected measurement data comes to NetAct in different formats, for example RNC uses XML and BSC uses raw binary format. The data format depends on network element and its interfaces [29].

The PM data can be also used for monitoring subscriber behavior by charting out the usage of different services that are available to the end-users. Such information is useful for operators in a way that they can make management decisions based on service use, identify current or future problems and opportunities [28].

### 3.3.2 Alarms – FM data

NetAct has several types of FM events, for example, cancel, acknowledge, unacknowledge and alarms. Emphasis on this chapter is on alarms, because through them the FM data is easiest to understand. All of these FM events are handled by the alarm collection and processing engine of NetAct. In capacity point of view in NetAct, all FM events are valued equally.

Alarms are indicators triggered by an error or a problem in a network element. In NetAct the alarms are handled with FM. When a network element triggers an alarm, the alarm details are transferred from network element to NetAct database [30].

NetAct has two types of alarms, threshold and exceptional alarms. Threshold alarm is raised when the measurement values exceed the defined alarm rule, for example, the desired severity of the alarm. Exceptional alarms are raised when threshold alarms cannot be generated [30].

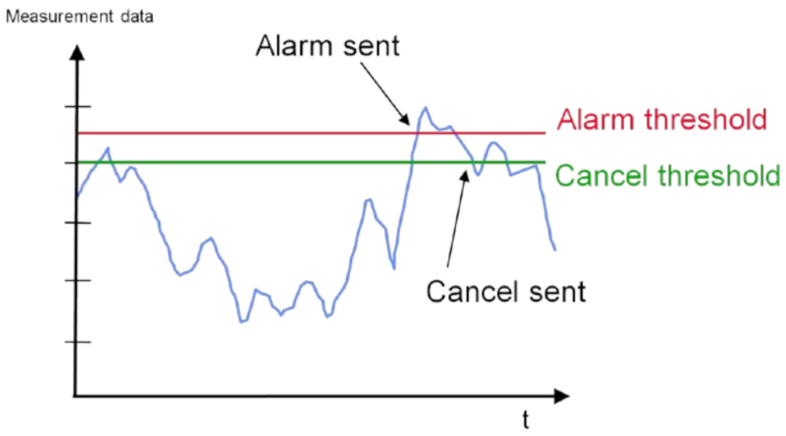


***Figure 12.*** *Example of an alarm and cancelation of an alarm*

Thresholds in NetAct are managed with Threshold for Measurements application. It allows the user to define threshold definitions for performance management. Threshold for Measurements helps to detect exceptional situations in the network such as faults and overloading. The alarm rules are formed by combining performance measurement counters with arithmetical and logical operators and it is defining the threshold condition. Figure 12 illustrates an example when the threshold condition is fulfilled; the application generates an alarm which is time-stamped at the time the problem occurred. When the threshold condition is no longer fulfilled, the alarm is automatically cancelled by a cancel rule. The automatic cancel rule is in a way the opposite of the alarm rule [30].

# 4. FACTORS THAT AFFECT SYSTEM PERFORMANCE

By definition system performance can be defined, as a system's capability to handle effectively the tasks that it has been assigned to do. The characteristics that have an effect on system performance are called performance parameters. In NetAct there are dozens of parameters that affect system performance and capacity. Because of the large number of parameters, it is impossible to estimate the precise effect of every single parameter. Furthermore, all parameters do not have an equal effect on the performance, and the parameters that have nothing in common at a first glance can have combined effects that are different to the sum of individual parameters. Thus only the most important parameters are examined during the NetAct performance process. Other parameters are fixed at their default values. Those fixed values are then presented with the capacity estimation figures. A parameter whose variation has a significant effect on system performance is called a factor.

The performance parameters of NetAct can be divided into two categories: hardware dimensioning parameters and workload parameters. Hardware dimensioning parameters affect the amount of resources offered by the NetAct hardware and workload parameters affect the system workload, i.e., how much the resources are used and needed. Hardware dimensioning parameters are the following [31]:

- Offered CPU time
    - CPU type
    - Number of CPU's
    - Number of servers in cluster
- Amount of memory
- Storage system
- Data communication network.

From an end user's point of view, performance means how well the system can handle the given tasks. And therefore it affects user experience and usability. Slow system performance can, in worst case, cause the user to replace the system altogether.

From a software development viewpoint, performance can be a complex topic and contain several different factors. The following list is an example of some of these factors affecting performance [32]:

- Application design and complexity

- Application testing
- Network
- Virtualization and the cloud
- Peak usage.

These factors affect also NetAct system performance. Therefore hardware, databases and system load are discussed in the next sections.

## 4.1 Hardware

NetAct has a virtualized architecture so that it can operate independently from the underlying hardware. This means that the servers' physical hardware can be shared and used as a resource pool for the virtualized infrastructure. As seen in Figure 13, the virtualization software is used for dividing the underlying hardware resources to smaller elements called virtual machines (VM), in a way that each VM has a designated amount of hardware resources in use. Each virtual machine can run either directly over a hypervisor or within a conventional operating system environment. Within virtualization each virtual machine has its own allocated hardware recourses [33].

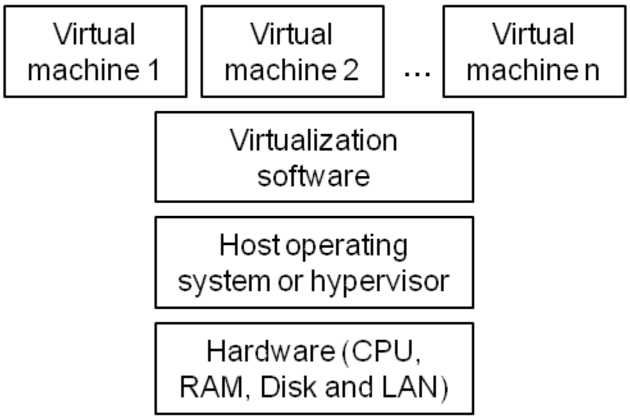


***Figure 13.*** *Server virtualization*

Hardware defines the servers' capability to handle different amounts of I/O operations. In NetAct this means the system's capability to handle different sizes of networks and numbers of simultaneous users. NetAct is delivered to the customers with three configuration options: small, mainstream and large [33]. The purpose of these configurations is to meet the requirements of operator networks of different sizes. For example, we can define that a small network contains 50000 GSM/EDGE, 30000 WCDMA and 15000 LTE network elements. Based on this statement we can say that the defined small server configuration in Table 1 should be able to handle this network size. The CPU configuration 1p8c means that the server has 1 processor with 8 cores. NetAct configurations are updated in each release to ensure that the hardware is not outdated; this makes the configurations differ from release to release.

***Table 1.*** *Example configurations*

| Server configuration | CPU configuration | Memory (GB) |
|---|---|---|
| Small | 1p8c | 48 |
| Mainstream | 2p6c | 64 |
| Large | 2p8c | 128 |

In practice, operators tend to have mixed server setups and it is not uncommon that the operators are using older hardware in servers. Therefore the configurations may not follow the guidelines. In practice this means that the operator can have part of the network which can be categorized for the older G6 medium and rest for the newer G8 medium configuration setup [34].

NetAct performance has to consider the legacy support and therefore performance test even older configuration setups. This ensures the scalability of the hardware.

## 4.2 Databases

Database is one of the most important elements that affect NetAct performance. The reason for this is that NetAct stores the data it gathers from network elements and from NetAct itself into an Oracle relational database [35].

This creates high requirements for the underlying hardware to be able to handle all of the events generated by the network. The number of events usually goes up to several thousands in a second depending on the operators' network size.

The database also affects the system's capability to handle disaster situations. In Oracle database disaster recovery is handled with a feature called redo log. In practice this means that the redo log contains all the events that have occurred in the database [36]. Despite the practicality of this feature, there is tradeoff in system performance due to the doubling of the number of root I/O events.

Usage of redo log in NetAct can be described with an example of backup process during the database crash. It is common to make backup from the database during the night (for example 3:00), when the system load is not high. If the database then crashes at 16:00 o'clock we can recover the database from the backup and the changes what happened during the 3:00 and 16:00 from the redo log. This way we can get to the point before the database crash.

## 4.3 System load

The system load is defined as the amount of resource utilization of the system. Typically, from a hardware perspective, the system load is presented as an average system load.

Its definition is an average number of processes in the CPU waiting queue per unit time. System load average helps to narrow down problems in performance testing. For example, if the average load is the same at a bad situation as in the normal situation, then the performance issues must be looked somewhere else [37].

From NetAct perspective the system load is seen as events caused by users in management systems through northbound interface, network elements through southbound interface and system itself with different management applications. These causes are seen in Figure 14, which illustrates the NetAct infrastructure.

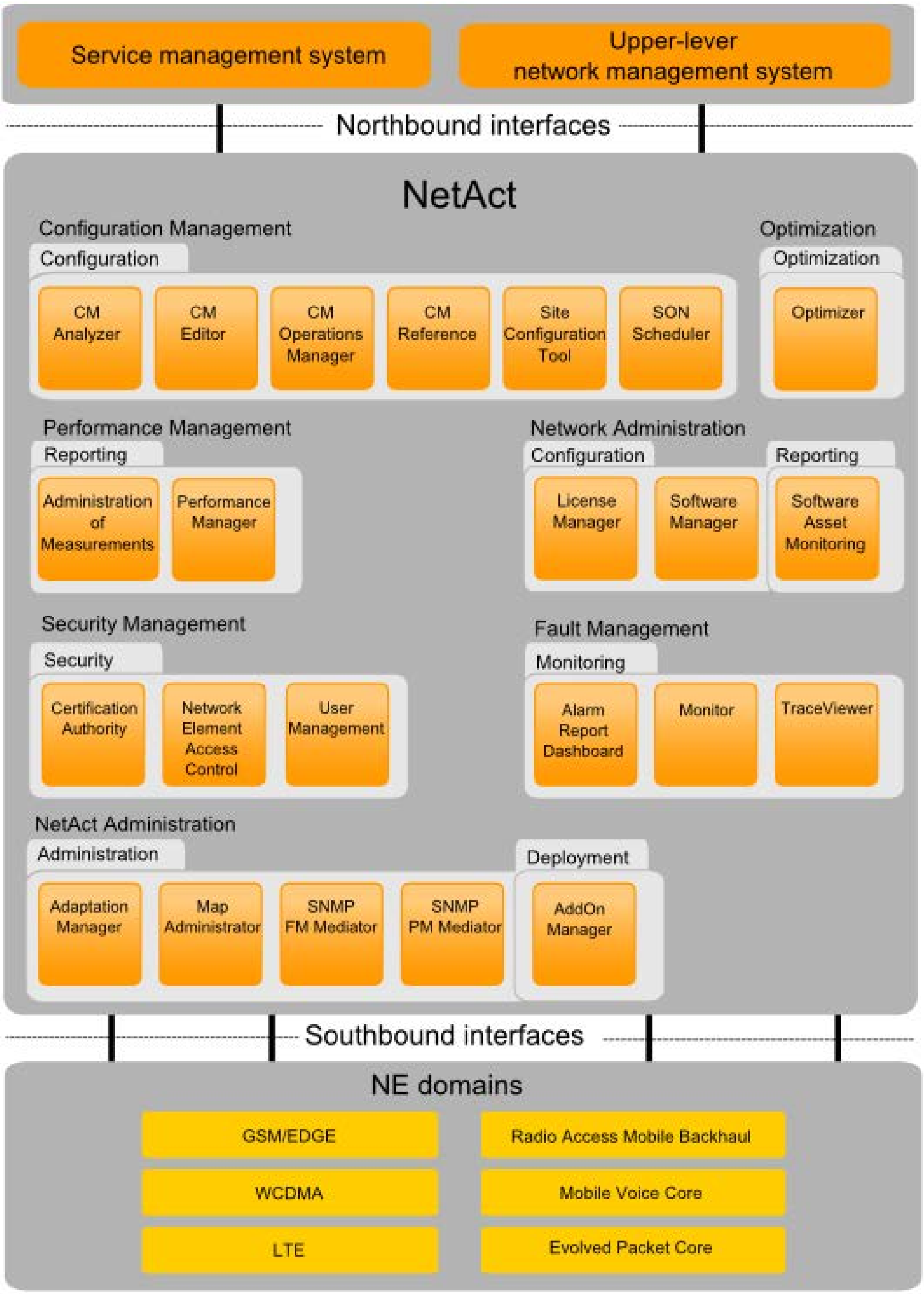


***Figure 14.*** *NetAct infrastructure [38]*

Internal load in NetAct comes mainly from database I/O operations and user actions. Overall causes for increased system load may vary from a high number of simultaneous user events to a bottleneck situation caused by hardware failure.

# 5. PERFORMANCE TESTING

Testing cannot prove that there are no defects in the software, and it does not improve the quality of the software but measures it. It is also not about making sure that the software works as it is supposed to work, because the person designing the test cases sees only what he or she wants to see. Therefore a successful test run is one that causes a failure. A failure in testing is an event that is externally observed and is caused by a fault, even if all faults do not lead to a failure. Testing is an important part of software development process alongside coding. Traditionally testing involves certain phases [39]:

- Test design
- Creation of test cases
- Executing the test cases
- Evaluating and reporting the results of the test runs.

Overall testing can be divided to several different categories and one of them is performance testing. Performance testing is part of system testing and its objective is to uncover bottlenecks in performance, determine or validate the speed, scalability, and stability characteristics of the product under test via technical investigation [40]. Therefore in order to comprehensively determine the run-time performance, performance tests are often coupled with stress testing and usually require both hardware and software instrumentation. This is because often it is necessary to measure hardware resource utilization in demanding fashion [41].

In the NetAct performance testing, the simulators play an important role because with them the testers create scenarios which simulate operator networks with certain number of network elements and network functionalities. Simulators functionality is monitored with software called Introscope and it is discussed more in section 5.3.1. The general target in NetAct performance testing is to ensure that the resource usage and system speed of the newer release is in balance. Therefore one key target of NetAct performance testing is to ensure that the new system release is capable to handle the same amounts of load and users as the previous release. In practice this means that the system is tested with the same load and simulator versions as in the previous release.

## 5.1 Key types of performance testing

Performance testing can be divided into several sub-categories by test type. In order to benefit from performance testing, it is important to understand the differences between these performance test types. By understanding different test types, testers are able to decide when to apply an appropriate test over the course of a given performance-testing project; this reduces risks and minimizes costs of the project [40].

In literature, performance testing is often categorized from three to seven major categories. Usually some additional concepts are also used due to the different test environments and needs [40]. The following list introduce the three major categories of performance testing and some of the additional concepts [40][42].

- **Load test** focuses on the system's capability to handle increasing levels of anticipated realistic loads resulting from the transaction requests generated by numbers of concurrent users or processes.
- **Stress test** focuses on testing the ability of a system or a component to handle peak loads at or beyond the limits of its anticipated or specified workloads, or with reduced availability of resources such as accessible computer capacity and available bandwidth.
- **Scalability / Capacity test** focuses on the system's ability to meet future efficiency requirements and capability to handle defined user transactions while maintaining defined performance goals.

There are some additional concepts commonly used in performance and other types of testing:

- **Component test** is any performance test that targets an architectural component of the application or system.
- **Smoke test** is the initial run of a performance test to see if application can perform its operations under a normal load.
- **Unit test** is any test that targets on verification of module of code, with focus on performance characteristics. Unit testing also includes **regression testing**.
- **Validation test** focuses on testing the product if it meets the defined criteria and works as intended.

In NetAct performance testing, the emphasis is generally on *stability or also called soak testing* while testing also includes dimensioning, load, stress and overload tests. Stability testing can be categorized under stress testing and its target is to determine how long the system can operate under a defined workload and what kind of errors occur during the workload.

Practically this means that in the NetAct stability testing, the system is run with moderate loads while having some management functions and network elements. Stability tests include combined network element mixes while testing all interfaces simultaneously. The target of stability testing is to test, the set performance requirements to identify problems that may appear only after an extended period of time and check if the system can continuously function well during that time. From stability testing, load and dimensioning testing are derived. The target of dimensioning testing is to gather data and metrics to create a performance model for the dimensioning tool, while load testing requires loads with all management functionalities and network elements. Load testing is divided to two parts, stress and overload testing. The target of stress tests is to test the system with increased load in management functionalities and network elements so that the upper limits are found. With stress tests, we can get data for evaluating requirements for upcoming releases. The target of overload testing is to exceed the upper limits of the system while observing the behavior of the system and its components [43].

## 5.2 Metrics – what is measured in testing

The definition of a metric is that the metric is a measure of some property of a piece of software or its specifications. In practice this means that the metric is a measure of some characteristic of the software [44].

NetAct is tested for FM, PM and CM components. Tested metrics for the mentioned components vary from a number of alarms or events in a specific time span. Alarm in FM can rise from several combined events or a single severe problem in the network. It produces an integer value for the occurred event. In some occasions processed metric values are used as well [45].

## 5.3 Methods of testing

Most common testing method used in software testing is the box approach method, which contains black box, white box and their combination gray box testing.

Black box testing means a testing approach where the tester has no knowledge of the code in the application. In practice the tester who uses black box methods, interacts with the system’s user interfaces by providing input and examining outputs without knowing how the software handled the input. The benefit of black box testing is that it can be performed with moderately skilled testers and without code access. As a disadvantage, test case design is more difficult and limited.

White box testing means the detailed investigation of internal logic and structure of the code. White box testing in practice requires that the tester looks inside the source code and checks if it is functioning correctly. This requires coding or code-reading skills from the tester, but at the same time, it helps optimize the code.

Gray box testing combines parts from black and white box testing, so that the tester has a limited knowledge of the internal functionality of the software. This means that the tester does not have access to the source code but has access to the design documents and the database. The main advantage of gray box testing comes from the combination of black and white box testing. The testing is done from the point of view of the end-user and not the designer. This way usability issues can be noticed more easily. Gray box testing is not without disadvantages. One of the major disadvantages in this kind of testing is that it is impossible to cover all the input streams, because it would be time consuming and the tester's ability to go over the code and test the coverage is limited [46].

NetAct's system performance testing is primarily categorized to grey box testing methods. In practice these methods are used to tests southbound interface connections, meaning the communication between network elements and NetAct, maturing the database and testing the system with SQL-queries.

### 5.3.1 Monitoring

Monitoring can be defined as tracking of performance by utilization of server or software functionality. Monitoring can be categorized to different types based on what is monitored. The generic monitoring typically concentrates on load and stress of the server and includes metrics of the CPU and memory utilization. Monitoring the web and application server tier typically tracks the performance of a particular application server such as Oracle or IBM WebSphere. Database server tier monitoring monitors databases such as MySQL or Oracle. Mainframe tier monitoring concentrates on covering end-to-end monitoring. It focuses on a small set of metrics based around memory and CPU utilization per job and logical partition [47].

In performance testing, server monitoring is essential for stability testing due to its nature of revealing slowly developing problems such as memory leaks [47]. It is also common to use KPI values. In NetAct the monitoring of WCDMA network utilization is realized by monitoring KPIs [48].

In NetAct performance testing, monitoring is used for analyzing the impact of actions, such as added system load. With monitoring tools such as Introscope, the testers can see a graphical display of the utilization of disks, CPUs and memory or the status of monitored services. This gives a wider viewpoint for analyzing the system under testing. Occasionally offline monitoring is used. It collects data to predefined files for later analysis.

### 5.3.2 Introscope

Introscope is a commercial web-based monitoring system for performance testing, developed by CA Wily Technology. It can be used for continuous application monitoring, incident detection and notification, root cause diagnosis and trend analysis [49]. In NetAct performance testing Introscope is used for monitoring and profiling purposes due to its ability to monitor Java applications. It collects information from hardware, databases and NetAct itself. With Introscope, the testers can verify if the end-to-end transactions in NetAct are working as intended. Operator view of Introscope can be seen in Figure 15.

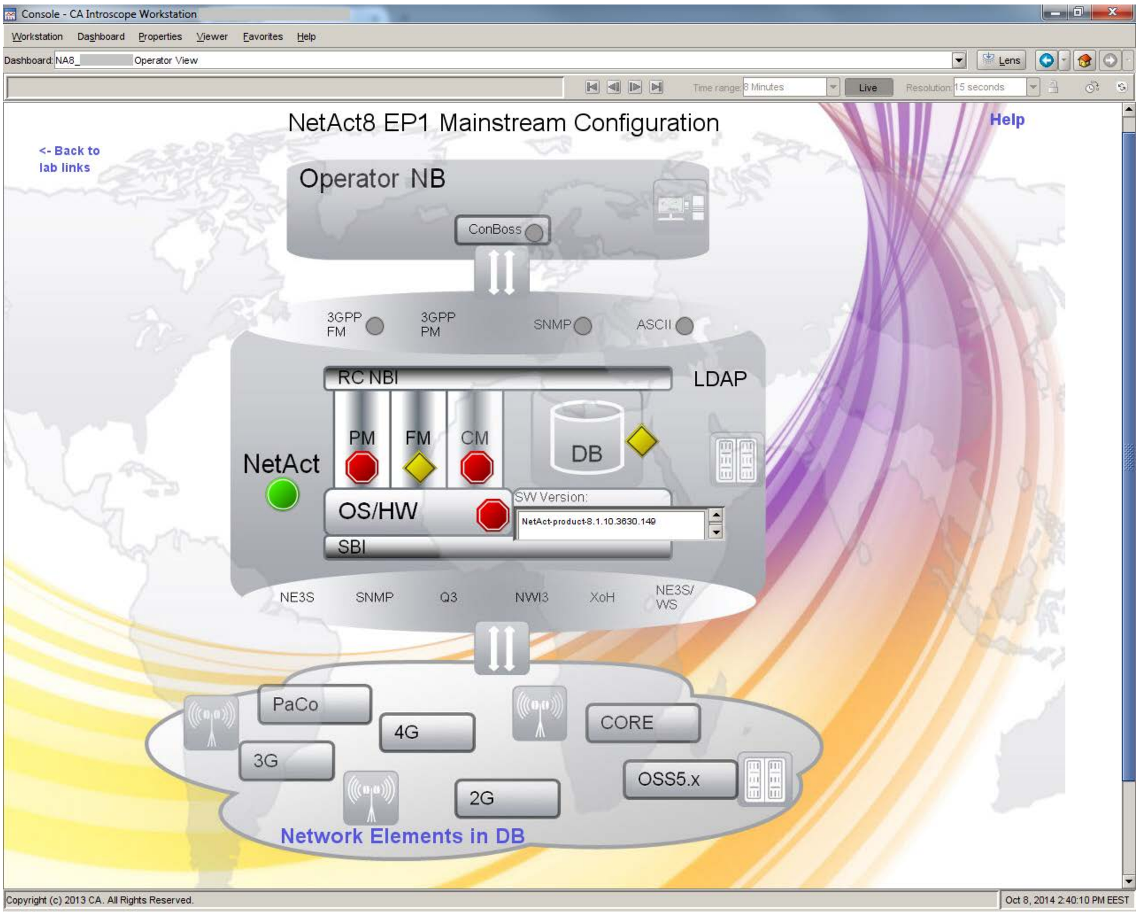


***Figure 15.*** *Introscope operator view*

From the overall operator view the tester can see which network element simulators are connected to NetAct and check the status of each NetAct component and simulator. Introscope also provides testers the tools for analyzing and reporting faults with predefined alarm sets and dashboards. Events can be viewed in real-time, so that during the performance tests, the testers see what is happening in the system under testing.

### 5.3.3 Test automation

Test automation by definition is a testing method where test software executes predefined test cases on the system or software under test, compares the results with expected outcomes and reports interpreted results [51]. The goal of test automation is to save time and money and offer regression for test cases. Test automation also improves testing accuracy because human tester can make mistakes during repetitive monotonous manual testing [50].

In NetAct performance testing (PET) the goal of test automation is to improve test case coverage and work efficiency by automating test scenario execution, data collection, analysis and reporting [50]. Yet in practice, the coverage depends on the tester who designs and creates the test case. The test automation in NetAct provides regression for testing the stability, load and high availability (HA) of management and network elements, in a way that the same test cases are run for each NetAct release so that the new errors are found quicker than with the manual test rounds.

## 5.4 Dimensioning

In NetAct performance dimensioning, the testers select a suitable number of NetAct users and network elements for the tests. The number of simultaneous users and network elements and their combinations varies in test cases from test to test and from tester to tester; this is mainly because the operator networks are also varying a lot. Therefore this kind of versatile testing is necessary for ensuring good performance for different NetAct installations. Even though the NetAct is not exactly a tool for radio network planning and it only takes parts of traditional network planning into account, it can be still used to help the planning process by providing data about faults and bottlenecks in network. Because of this it is good idea to introduce some of the basics of radio system planning process.

In radio system planning, dimensioning is a first phase of the overall planning process. Table 2 illustrates the highlights of the radio system planning process. The purpose of dimensioning in radio system planning is to initially draft the radio network configuration and deployment strategy for the long term [52]. In practice the dimensioning phase consists of estimating roughly the network layout and needed elements, such as BTSes, for covering a certain area and serving a certain capacity.

***Table 2.*** *Highlights of the radio system planning process*

| Subject | Findings |
|---|---|
| Dimensioning | Global parameters: traffic, coverage, threshold and base station antenna height |
| Detailed radio planning | Practical issues |
| Configuration management | Optimised features for different radio network evolution phases |
| Coverage planning | Accurate coverage predictions |
| Capacity and frequency planning | Accurate coverage predictions are a **must** |
| Parameter planning | Radio network functions |
| Optimisation | Adjustment of the radio network |
| Radio system planning process | Importance of documentation and site folder |

The radio network planning process differs slightly from one mobile technology generation to another; for example, in the WCDMA network planning process the base station height is defined in the dimensioning phase due to its effects on cell coverage and radio propagation channel. Compared to WCDMA network planning, GSM planning emphasizes frequency bands and frequency reuse.

# 6. DATA ANALYSIS

In this chapter the methods for gathering the operator data are presented and the operator data is analyzed further. Most of the operators whose data was included in the dimensioning model are from two different market regions. One of them is a slowly developing market region, where network penetration does not cover the whole continent and where poverty limits growth [53], while the other is a more stable and mature market region in which operators have a good network penetration and a large customer base [6]. This affects the network data research. Market development of unique mobile subscribers can be seen in Figure 16. The number of users in this figure is presented in billions.

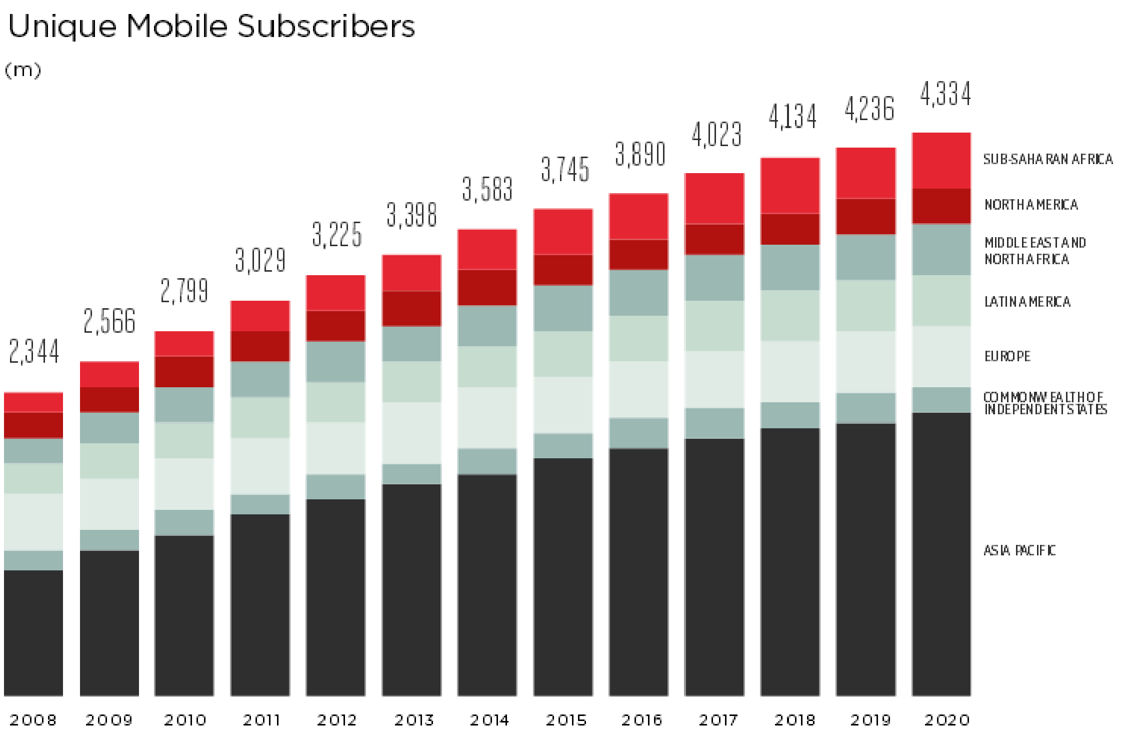


***Figure 16.*** *Global mobile market development [6]*

Based on these facts, the networks in these regions are growing at different rates, but what is common for both of the market regions is the requirement for faster connections. Therefore LTE growth will increase in both regions rapidly in the upcoming years.

## 6.1 Raw support data collection

The source for data was a data store generated for support teams' usage. It includes data from several operators and it is mainly used for generating monthly reports. The collected data comes in a zipped package which contains several files from the network alarms to different measurements and amounts of different radio and core network elements. In this thesis the focus was on the number of the network elements and NetAct health report files. This report file is generated by a tool in NetAct that performs a number of test cases checking system health. It also contains information about installed NetAct setup, for example, the numbers of data storage and server nodes and the hardware information on the nodes. From the hardware data the size of the NetAct installation can be determined. At this point the data collection for the developed dimensioning model happened manually from a specific dedicated server and from intranet.

## 6.2 Data analysis

The analyzed data included the number of different network elements varying from different radio access technologies to Core network. In Figure 17, we can see the structure of the analyzed file. The numbers are randomly generated to maintain the confidentiality of operator data. As an example, we can analyze the example file in Figure 17 and see that the example network has 4565 LNCEL and 1001 WBTS elements. LNCEL signifies LTE cells and WBTS signifies WCDMA base stations.

Total Object Count
------------------
789456

NOKIA OBJECTS:

| NAME | STATE | COUNT | % of Total |
|---|---|---|---|
| LNCEL | Operational | 4565 | 0 |
| LNREL | Operational | 54654 | 6 |
| LNHOIF | Operational | 4566 | 0 |
| PMRNL | Operational | 976 | 0 |
| FTM | Operational | 2429 | 0 |
| RET | Operational | 2146 | 1 |
| DECO | Operational | 1 | 0 |
| OMC | Operational | 1 | 0 |
| WS | Operational | 17 | 0 |
| STPG | Operational | 2459 | 0 |
| WBTS | Operational | 1001 | 0 |
| PWNE | Operational | 1483 | 0 |
| SVTT | Operational | 144 | 0 |
| SCFVER | Operational | 1253 | 0 |
| BSC | Operational | 9 | 0 |
| TWAMP | Operational | 10570 | 2 |
| LNADJL | Non-Operational | 64 | 0 |
| MODPR | Operational | 1 | 0 |
| A2NE | Operational | 1483 | 0 |
| RNHSPA | Operational | 9 | 0 |
| RNRLC | Operational | 9 | 0 |
| LCSE | Non-Operational | 13 | 0 |
| PUNIT | Non-Operational | 32 | 0 |
| MRBTS | Operational | 542 | 0 |

***Figure 17.*** *An example of operator data*

The analysis happened manually, by picking up predefined network objects and their amount. After gathering all the operator network data to the Excel sheet, the data could be analyzed by calculating minimum, mean, median and maximum values for each network element. After this the reference object models from NetAct Capacity statement document were taken into account and ratios for gathered network elements were calculated and compared to capacity statement reference values [54].

In the capacity statement documentation, ratio values define the size of the network and define the numbers of needed network elements. Ratios used in calculations and in capacity statement are shown in Table 3. From the ratios it can be analyzed, for example how many adjacencies or controlled objects, each element has. For example, the ratio LNCEL/LNBTS tells how many cells one eNodeB contain. This information is also interesting for radio network planning, because base stations commonly follow a certain cell structure. The cell structure usually consists of groups of 3, 6, 9 or 12 cells. From the analyzed data we can also verify that the WCEL/WBTS ratio follows this structure with only slight deviation. In the data the minimum WCEL/WBTS ratio was 2.5, average 5.1 and maximum 8.1. A possible explanation for this deviance from the cell structure can be explained by the fact that operators commonly have one or more non-working sectors in their base stations and a limited set of operator data which the analysis is based on.

***Table 3.*** *Network configuration ratios*

| 2G | 3G | LTE |
|---|---|---|
| BTS/BSC | WBTS/RNC | LNBTS/OMS |
| TRX/BTS | WCEL/WBTS | LNCEL/LNBTS |
| | ADJS/WCEL | LNADJ/LNBTS |
| | ADJI/WCEL | LNADJL/LNADJ |
| | ADJD/WCEL | LNADJG/LNBTS |
| | ADJG/WCEL | LNADJW/LNBTS |
| | | LNREL/LNCEL |
| | | LNRELG/LNCEL |
| | | LNRELW/LNCEL |

The purpose of comparing the reference models to actual operator network data is to find out how well the reference models reflect the actual operator network data and if the reference models require adjustment. Based on the comparison it can be seen that the used reference values in NetAct are either the same or very close to the average and maximum calculated values. There were also some ratio values which were over- and underestimated. By having overestimated reference values the system can handle the network growth in future, but having underestimated values, the system can crash due to sudden load, and therefore, in the worst case scenario, cause financial loss for the company. From the data we can also see that the most common NetAct hardware configuration types from selected operator networks are medium or also called mainstream and mixed medium type. Based on these server configurations, most of the operator networks can be categorized to medium sized networks. Another finding was that, based on the number of network elements in some of the operator networks, the hardware configuration should be different, but it might be that the collected data does not have information from the whole network and therefore the configuration size seems to be overestimated.

# 7. DIMENSIONING MODEL

The dimensioning model can be seen as a description of the capacity that the system should be able to handle without failures. The capacity of a system is a narrower notion than performance; it is limited to the ability to provide a particular amount of a particular service. This is probably the most important part of the performance and it should always be objectively assessed. The capacity is also a limiting factor for the system, in the sense that the maximum capacity is the total load that the system can still manage without causing conspicuous harm to the users and the rest of the system. This can be interpreted as demands that when the system is working at its maximum capacity level, the data that is coming into the system must be handled with steady buffering, (that is, buffers are not growing), user response times must be acceptable, and background processes must be able to perform their tasks within tolerable time limits. If any of these requirements is not fulfilled, the maximum capacity has been exceeded and the user experience starts to decrease [33].

In this chapter, the dimensioning model means the model which is used for NetAct performance testing. The model represents operator networks in three different sizes: small, mainstream and large. The model is actually the maximum capacity that NetAct can handle in each size of network. The dimensioning models are formed from collected operator data by calculating ratios for different network elements and by calculating minimum, maximum, median and mean values. After the data processing, the obtained values were compared to reference values in the capacity statement documentation to ensure that the generated models are not underestimated.

The operator data used in dimensioning comes in several separate files and therefore the data collection is a time-consuming operation. This gave a motivation for writing an automated program for data collection and calculating the required ratios and values.

## 7.1 Automation of data collection

The target of automation is typically to speed up processes and to reduce the required amount of manual labor. Easier and faster data collection process also creates an incentive for gathering statistical information in a more systematic way, so that it can be used for forecasting the mobile network development. As mentioned above in chapter 6.1 the operator data comes in several different types of files. In automation the interest was in managed object count files, which are log files containing lists of network objects. An example of this kind of file can be seen in Figure 17 in Section 6.2.

The goal of the written program was to speed up and automate the log file gathering process. It was written in Java language due to its universal nature and wide range of libraries. The program generates an Excel file from the collected operator data. This meant using libraries, able to handle XML and Excel files. The program automates the data collecting from the files, but still it requires some manual work in order to get the files. The manual part in the process is the extraction of operator data files from a packet file to a folder where the program can access it.

### 7.1.1 Data compiler

The data compiler program is compiled to a jar packet to ease usability. The jar packet is named as *automatic_mo_object_parser.jar* and it can be run from the command line with the command:

```
java -jar automatic_mo_object_parser.jar -x “C:\folder\template.xml”
-p "C:\folder\location_of_the_mo_objects"
```

The purpose of the command line arguments is to provide flexibility for the usage of the program. For the command line options the Apache Commons CLI [55] library was used. With this CLI (command line interface) library it was possible to create command line arguments for specifying locations for the needed files. The argument –x defines the path of the template.xml file, and the argument –p defines the location of the folder where the managed object count files are located. The template.xml is required for generating the excel file. It determines what objects are parsed from the operator data and what ratios are calculated. Its tree information is also used for creating the element name and ratio columns in the Excel file. For generating the Excel file. Apache POI API [56] library was used. It is a Java API (application programming language) which enables handling Microsoft documents. This API library was selected because it was fairly easy to use and its developers provided good examples of usage on their website.

The user of the compiler can get help about the command line options by typing

```
java –jar automatic_mo_object_parser.jar --help
```

The program structure can be illustrated with a Unified Modeling Language (UML) class diagram. Figure 18 shows the design of the developed data compiler program. In the diagram we can see the classes and their relations.

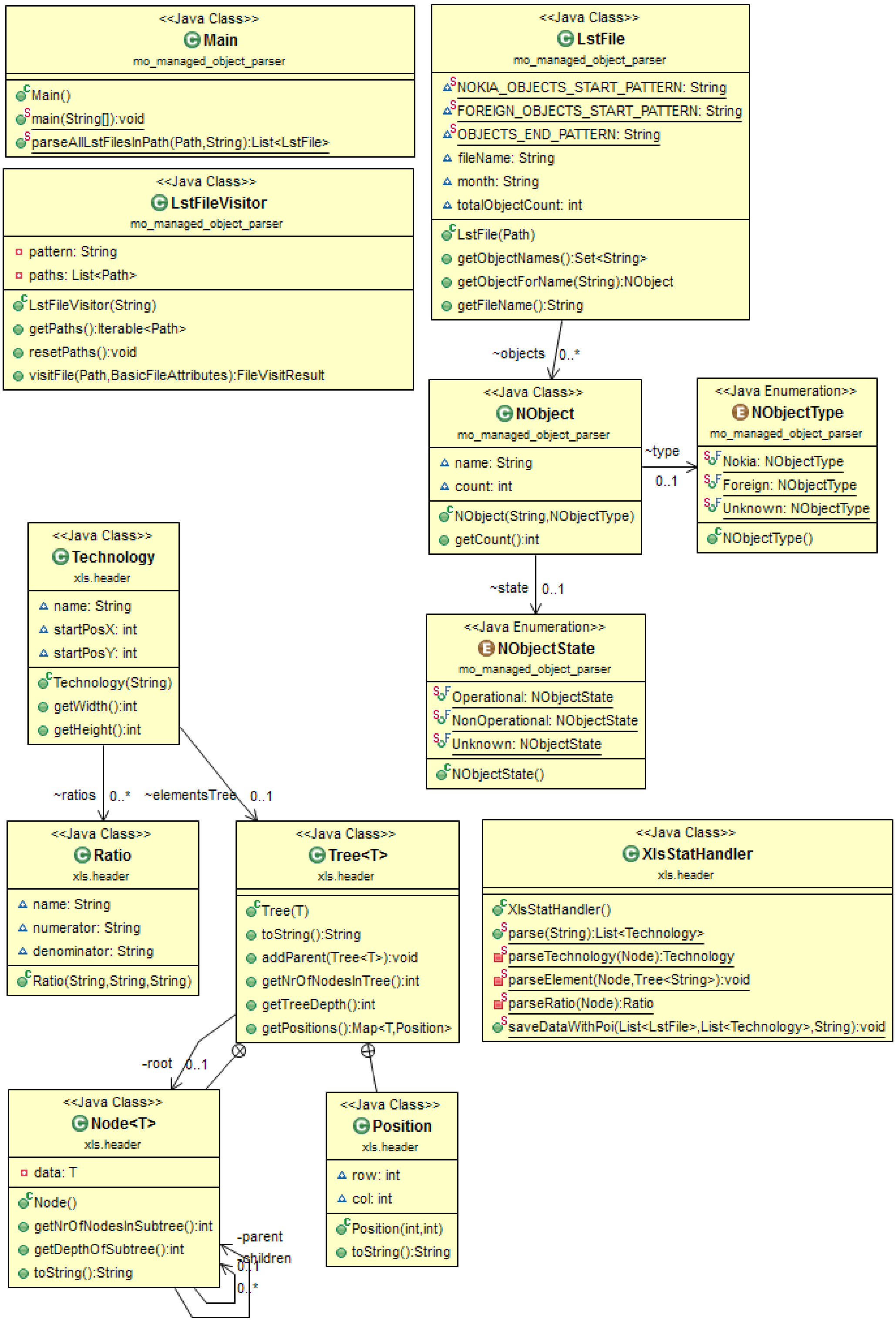


***Figure 18.*** *UML class diagram of the data compiler program*

When the program is running, it goes through the folder structure which the user has defined and parses selected Nokia objects from the files into a list. An example of the parsed file can be seen in Figure 17 in section 6.2. The selected objects are defined in the **template.xml** file. The XML file consists of the defined network elements and ratios. The following example of GERAN objects (Listing 1) shows how the XML template file has been constructed:

```
<?xml version="1.0" encoding="UTF-8"?>
<MANAGED_OBJECT_COUNT>
<Technology name = "GERAN 2G">
      <Elements>
            <Element name = "BSC">
                  <Element name = "BCF"/>
                  <Element name = "BTS">
                        <Element name = "TRX"/>
                  </Element>
            </Element>
      </Elements>
      <Ratios>
            <Ratio name = "BTS/BSC"/>
            <Ratio name = "TRX/BTS"/>
      </Ratios>
</Technology>
```

***Listing 1*** *Template example of GERAN objects*

After the parsing, the program generates an Excel file from the list and template.xml. From the template, the program parses ratios, for example, LNCEL/LNBTS, and generates formulas picking up the correct cells containing the values for calculating ratio. If the program cannot get one of the values for calculating a ratio, it leaves the cell empty. This eases the process if the user wants to calculate minimum, mean, median and maximum values at later on. An example of a parsed Excel output with randomly generated operator data can be seen in Figure 19. The random data is due to the sensitive nature of the operator data.

| | | | | | | c:\autom | c:\autom | c:\autom | c:\autom | c:\autom. |
|---|---|---|---|---|---|---|---|---|---|---|
| GERAN 2G | | | | | | | | | | |
| | BSC | | | | | 46,8 | 74,1 | 1,3 | 68,9 | 2,6 |
| | | BTS | | | | 11934 | 7289,1 | 4849 | 5510,7 | 7,8 |
| | | | TRX | | | 2644,2 | 47594 | 464,1 | 8004,1 | 37649 |
| | | BCF | | | | 3898,7 | 3277,3 | 22,1 | 828,1 | 1,3 |
| ratio | BTS/BSC | | | | | 331,5 | 127,88 | 4849 | 103,98 | 3,9 |
| ratio | TRX/BTS | | | | | 0,288 | 8,4884 | 0,1244 | 1,8882 | 6274,9 |
| UTRAN_3G | | | | | | | | | | |
| | RNC | | | | | 6,5 | 10,4 | 20,8 | 22,1 | 20,8 |
| | | WBTS | | | | 787,8 | 625,3 | 378,3 | 470,6 | 1,3 |
| | | | ADJG | | | 217495 | 135192 | 157946 | 3759,6 | 126612 |
| | | | ADJD | | | | | | 1,3 | |
| | | | ADJI | | | 460,2 | 104855 | 131001 | 4270,5 | 255256 |
| | | | ADJS | | | 427,7 | 187587 | 208195 | 3066,7 | 607,1 |
| | | | WCEL | | | 12214 | 11625 | 2159,3 | 13662 | 113,1 |
| | | MRBTS | | | | | | | | |
| | | OMS | | | | | | | | |
| | AXC | | | | | | 68,9 | | 11,7 | |
| ratio | WBTS/RNC | | | | | 157,56 | 78,163 | 23,644 | 27,682 | 0,0813 |
| ratio | WCEL/WBTS | | | | | 20,154 | 24,168 | 7,4203 | 37,74 | 113,1 |
| ratio | ADJS/WCEL | | | | | 0,0455 | 20,978 | 125,34 | 0,2918 | 6,9782 |
| ratio | ADJI/WCEL | | | | | 0,049 | 11,726 | 78,869 | 0,4064 | 2934 |
| ratio | ADJD/WCEL | | | | | 0 | 0 | 0 | 0,0001 | 0 |
| ratio | ADJG/WCEL | | | | | 23,15 | 15,119 | 95,091 | 0,3578 | 1455,3 |
| EUTRAN_LTE | | | | | | | | | | |
| | OMS | | | | | 2,6 | 7,8 | 7,8 | 10,4 | 7,8 |
| | | LNBTS | | | | | 49,4 | 1640,6 | 962 | 29,9 |
| | | | LNCEL | | | | 102,7 | 267,8 | 691,6 | 107,9 |
| | | | | LNRELT | | | | | | |
| | | | | LNRELX | | | | | | |
| | | | | LNRELW | | | | | 39056 | 22086 |
| | | | | LNRELG | | | | | 12305 | 1049,1 |
| | | | | LNREL | | | 2,6 | 111627 | 92730 | 94887 |
| | | | LNADJT | | | | | | | |
| | | | LNADJX | | | | | | | |
| | | | LNADJG | | | | | | 7087,6 | 605,8 |
| | | | LNADJW | | | | 7,8 | | 23802 | 12675 |
| | | | LNADJ | | | | 32776 | 32585 | 217,1 | 25935 |
| | | | | LNADJL | | | 114,4 | 97894 | 1,3 | 68840 |
| | | MRBTS | | | | 9,1 | 2752,1 | 3554,2 | 315,9 | 3355,3 |
| ratio | LNBTS/OMS | | | | | 0 | 8,2333 | 273,43 | 120,25 | 4,9833 |
| ratio | LNCEL/LNBTS | | | | | 0 | 2,7026 | 0,2122 | 0,9346 | 4,6913 |
| ratio | LNADJ/LNBTS | | | | | 0 | 862,52 | 25,82 | 0,2934 | 1127,6 |
| ratio | LNADJL/LNADJ | | | | | 0 | 0,0045 | 3,9056 | 0,0078 | 3,4506 |
| ratio | LNADJG/LNBTS | | | | | 0 | 0 | 0 | 9,5778 | 26,339 |
| ratio | LNADJW/LNBTS | | | | | 0 | 0,2053 | 0 | 32,164 | 551,09 |
| ratio | LNREL/LNCEL | | | | | 0 | 0,0329 | 541,88 | 174,31 | 1143,2 |
| ratio | LNRELG/LNCEL | | | | | 0 | 0 | 0 | 23,129 | 12,64 |
| ratio | LNRELW/LNCEL | | | | | 0 | 0 | 0 | 73,413 | 266,09 |
| CORE | | | | | | | | | | |
| | FLEXINS | | | | | | | | | |
| | FING | | | | | | | | | |
| | GGSN | | | | | | | | | 5,2 |
| | SGSN | | | | | | | | | |
| | | PAPU | | | | | | | | |
| | MSC | | | | | 1,3 | | | | |
| | MSS | | | | | | | | | |
| | OMGW | | | | | | | | | |
| | MGW | | | | | | | | | |

***Figure 19.*** *Example output of the data processer*

In Figure 19 we can see how the XML tree, containing network elements, element adjacencies and ratios, is parsed from template.xml into the first five columns in the example Excel output. From Figure 19 we can also see the object ratio values that the program has calculated from different network objects. These calculated ratio values form the baseline for the dimensioning model.

## 7.2 Small, mainstream and large dimensioning models

It is reasonable to divide the dimensioning model to different sizes so that the models would correspond to operator networks. As mentioned earlier, the NetAct installation comes in small, mainstream and large sizes based on the number of network elements and on hardware requirements. In the capacity statement for NetAct the models are categorized by NetAct installation size, but more specifically by technology, for example, capacity statement has GSM/EDGE + WCDMA RAN + LTE network combination instead of just one technology, even if the operator data shows that the operators can have networks with just one technology, e.g., GSM or LTE. In the capacity statement LTE comes with varying numbers of cells and carriers per eNodeB, because by increasing the number of carrier signals and cells we can have more capacity in the network.

The capacity of an operator network can be analyzed by calculating ratio values. The number of available carrier signals in LTE network can be estimated from the LNREL/LNCEL ratio, e.g., 32 or less indicates one carrier signal, and 64 indicates two carrier signals. The number of cells in an LTE network can be estimated from the LNCEL/LNBTS ratio. The only drawback in the operator data is that it does not tell us the number of simultaneous NetAct users.

Another important aspect in dimensioning models that should be taken into account is the market region; this is because the operators from Asia differ in regard to their network configuration from their American or European counterparts. Therefore the dimensioning model is also separated in terms of different market regions. Assigning the operator data to different regions is somewhat problematic because the data only consists of 44 operator networks. From this set it is impossible to create accurate models for all the regions. Therefore we only defined a precise model for two specific market regions and an overall model.

As a demonstration, the developed LTE dimensioning models from random data is presented in Table 4. The LTE dimensioning models were selected because it is the latest mobile technology and therefore the most interesting. Dimensioning models contain a suggestion for small, medium and large network configurations. The models are formed from the set of 44 operator networks and the minimum, average and maximum values were calculated from this set.

***Table 4.*** *Dimensioning models for LTE*

| LTE | | Minimum | Average | Maximum | Small | Medium | Large |
|---|---|---|---|---|---|---|---|
| Elements | OMS | 6 | 10 | 29 | 13 | 18 | 44 |
| | │└ MRBTS | 1045 | 2571 | 3679 | | | |
| | └ LNBTS | 82 | 711 | 1961 | 4000 | 2667 | 2222 |
| | │└ LNADJ | 426 | 22396 | 115973 | 80000 | 170667 | 284444 |
| | │ └ LNADJL | 1264 | 62594 | 316368 | 240000 | 1024000 | 2560000 |
| | │└ LNADJW | 1 | 4353 | 18309 | 128000 | 170667 | 284444 |
| | │└ LNADJG | 29 | 1982 | 5452 | 80000 | 106667 | 177778 |
| | └ LNCEL | 56 | 1728 | 5689 | ***12000*** | ***16000*** | ***20000*** |
| | │└ LNREL | 3170 | 78472 | 332688 | 384000 | 1024000 | 2560000 |
| | │└ LNRELG | 2 | 3425 | 9465 | 72000 | 144000 | 240000 |
| | └ LNRELW | 1 | 8587 | 30043 | 180000 | 320000 | 500000 |
| Ratios | LNBTS/OMS | 4 | 102 | 327 | 300 | 150 | 50 |
| | LNCEL/LNBTS | 3 | 3 | 3 | 3 | 6 | 9 |
| | LNADJ/LNBTS | 5 | 23 | 59 | 20 | 64 | 128 |
| | LNADJL/LNADJ | 3 | 3 | 3 | 3 | 6 | 9 |
| | LNADJG/LNBTS | 0 | 3 | 7 | 20 | 40 | 80 |
| | LNADJW/LNBTS | 0 | 8 | 25 | 32 | 64 | 128 |
| | LNREL/LNCEL | 13 | 31 | 61 | 32 | 64 | 128 |
| | LNRELG/LNCEL | 0 | 2 | 5 | 6 | 9 | 12 |
| | LNRELW/LNCEL | 0 | 6 | 15 | 15 | 20 | 25 |

The LTE dimensioning model is built around the number of LNCELs, meaning the LTE cells, and it is adjusted with the ratio values. It is evident that the small, average and maximum values are not the same as the suggested models for small, medium and large networks; this is because some of the operator networks are clearly giving too low values to be taken into account. Therefore it is reasonable to overestimate the models slightly. In practice this means that the average values give a good direction for small networks and the maximum values for the medium sized network. The large network model in the LTE case is clearly overestimated compared to actual operator data, but the point is that it takes into account the probable future growth of the network.

## 7.3 Recommendations for model usage

The model created in this study can be either used for adjusting the existing test sets or as a comparison for the used reference model. In NetAct stability testing the system is tested with the highest expected load and the tested load is based on the defined capacity statement. With the developed dimensioning model we can ensure the required load amount is comparable to what operators have in their networks.

Since NetAct is delivered in three different sizes, this developed dimensioning model can be also used to verify those setup sizes and improve their test sets.

# 8. FUTURE CHALLENGES IN TESTING

Lately mobile technologies have evolved rapidly in both the hardware and software fields. The development from 3G to LTE happened only in ten years and now 5G is a hot topic. This rapid development has also brought new requirements for software development and testing. Recently companies have started to develop and offer more and more scalable cloud services. With this trend, the stress, load, performance and compatibility testing will be more cost effective in future. This is because of the quick accessibility to infrastructure testing due the nature of the cloud system. Companies have also realized the benefits of test automation and started to shift more resources towards it [57].

## 8.1 5G and how it affects performance testing

5G means the upcoming fifth generation of mobile networks. Its goal is to meet new capacity requirements, improve network latency, decrease energy consumption and lower the overall costs in network deployment [58]. Figure 20 illustrates one possible roadmap for 5G. From the roadmap we can see that at the moment 5G is in research stage where the standards are still under discussion. Some of the earliest 5G product debuts are targeted to year 2018 by Huawei and Samsung. The debuts are targeted to happen during the FIFA World Cup in Russia [59] and the Winter Olympic Games in South Korea [60].

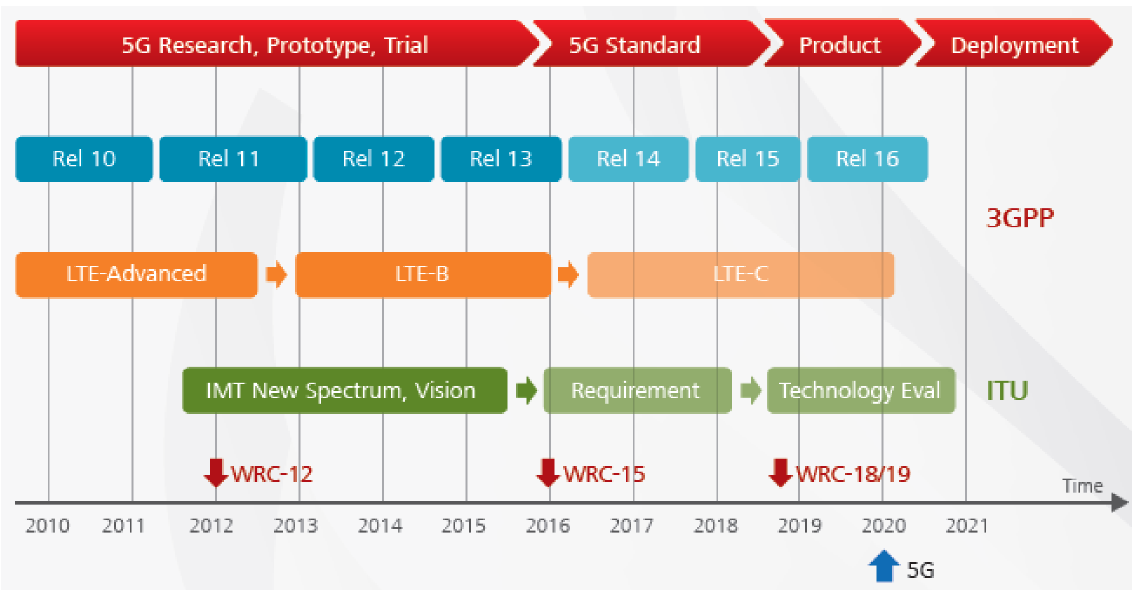


***Figure 20.*** *5G roadmap and timeline* [61]

The impact of 5G on NetAct performance testing can be only speculated at the moment, but a brief idea was given in April 2014, when the Brooklyn 5G Summit was held in New York. In the discussions small cells, massive MIMO, and software defined wireless network were suggested to be key enabler technologies for 5G [62]. Higher bandwidths and millimetre waves are also considered to be enablers for 5G and therefore they should also be taken into account in the 5G discussions.

These technologies can be seen to affect cell size and therefore the number of network elements, as well as the number of user devices connected to the network will dramatically increase in the future.

## 8.2 Continuation for this study

This study has introduced a dimensioning model and automation program for NetAct performance testing. The next logical step is to improve automation and make it more dynamic. The automation of operator data gathering can be improved in many ways, for example, the next step could be adding information on the used server configuration, so that it would be also included in the generated Excel file. Other improvements would be automated market region analysis and a simplified report where the network development could be seen in a longer time span. These improvements might be time-consuming, because e.g. an automated market analysis would require a database of operators and their market regions. However, these improvements would ease the dimensioning of performance testing in a way that the operator network trends could be calculated more efficiently and the testing could be scaled accordingly.

# 9. CONCLUSIONS

The goal of the thesis was to develop and ensure that the used dimensioning model in NetAct performance testing corresponds to operator networks. The thesis introduced NetAct, what NetAct performance testing means and what its goals are. Most of the information on NetAct in this thesis is based on NetAct documentation, and the information on NetAct performance testing is based on the interviews of the NetAct performance testing team. For the dimensioning model, used operator data came from the customer support team, and it was collected from September till October 2014 from an intranet page. The developed dimensioning model was based on this collected operator data. During the data collection process, a motivation for automation rose due to the amount of data and required manual work. The automation program was written in Java with Eclipse and it managed to speed up and ease the data gathering process. By creating the program it created a possibility for easier continuation for this thesis.

The overall conclusion for the dimensioning model problem presented in this thesis is that the developed dimensioning model was able to provide additional benefit to the NetAct performance testing by assuring that the used dimensioning model reflects real operator networks fairly accurately, even though the used test sets are oversized when compared to the actual amount of network elements that the operators have.

Even though the oversized test set is pointed out, it does not mean that it is bad for the system or testing point of view, because by oversizing, we can ensure the NetAct will maintain scalability in the future as well.

## REFERENCES

[1] NetAct™ Operating Documentation – System Overview, 2014. Available: Nokia Intranet

[2] NetAct™ Operating Documentation – Configuration Management, 2014. Available: Nokia Intranet

[3] NetAct™ Operating Documentation – Performance Management, 2014. Available: Nokia Intranet

[4] NetAct™ Operating Documentation – Fault Management, 2014. Available: Nokia Intranet

[5] NetAct™ Operating Documentation – Security Management, 2014. Available: Nokia Intranet

[6] GSMA – The Mobile Economy 2014, 2014. Available: http://www.gsmamobileeconomy.com/GSMA_ME_Report_2014_R2_WEB.pdf

[7] F. Firmin – The Evolved Packet Core, Available: http://www.3gpp.org/technologies/keywords-acronyms/100-the-evolved-packet-core

[8] Liquid Core, 2014. Available: http://networks.nokia.com/portfolio/liquidnet/liquidcore

[9] Nokia Liquid Radio White Paper.pdf, 2014. Available: Nokia Intranet

[10] Overview of technology by ETSI – GERAN, 2014. Available: http://www.etsi.org/technologies-clusters/technologies/mobile/geran

[11] J. Eberspächer, H.J Vögel, C. Bettstetter, C. Hartmann - GSM Architecture, Protocols and Services, 2009, ISBN: 978-0-470-03070-7

[12] Base station subsystem: BSS interfaces, 2014. Available: http://en.wikipedia.org/wiki/Base_station_subsystem#BSS_interfaces

[13] T. Saxena, J.S. Jadon - Review on 2G, 3G and 4G Radio Network Planning, 2013. Available: http://iasir.net/IJEBEApapers/IJEBEA13-354.pdf

[14] H. Holma, A. Toskala - WCDMA for UMTS HSPA Evolution and LTE, Wiley, 2010, ISBN: 978-0-470-68646-1

[15] Overview of technology by ETSI – UTRAN, 2014. Available: http://www.etsi.org/technologies-clusters/technologies/mobile/utran

[16] Overview of technology by ETSI – UMTS, 2014. Available: http://www.etsi.org/technologies-clusters/technologies/mobile/umts

[17] Overview of technology by ETSI - Long Term Evolution, 2014. Available: http://www.etsi.org/technologies-clusters/technologies/mobile/long-term-evolution

[18] LTE Network Infrastructure and Elements, 2011. Available: https://sites.google.com/site/lteencyclopedia/lte-network-infrastructure-and-elements

[19] B. Barton - LTE and beyond: Interfaces and their protocol stacks, January 2012. Available: http://www.lteandbeyond.com/2012/01/interfaces-and-their-protocol-stacks.html

[20] C. Janssen - What is a Core Network?, 2014. Available: http://www.techopedia.com/definition/6641/core-network

[21] H. Kaaranen, A. Ahtiainen, L. Laitinen, S. Naghian, V. Niemi – UMTS Networks Architecture, Mobility and Services, Wiley,2005, ISBN: 0-470-01103-3

[22] Wireshark – Global System for Mobile communication (GSM) protocol family, 2013. Available: http://wiki.wireshark.org/GsmProtocolFamily

[23] 3GPP - Overview of 3GPP Release 4 Summary of all Release 4 Features, December 2004. Available: ftp://www.3gpp.org/tsg_ran/TSG_RAN/TSGR_26/Docs/PDF/RP-040548.pdf

[24] LTE Basic and Optimisation, 2014. Available: Nokia Intranet - LTE Basic and Optimisation.pdf

[25] Managed Object Reference, August 2014. Available: Nokia Intranet - Managed_Object_Reference_5.5_Aug29th.pdf

[26] Westbay Engineers - What is an Erlang?, January 2014. Available: http://www.erlang.com/whatis.html

[27] NetAct™ Operating Documentation – Southbound interfaces, 2014. Available: Nokia Intranet

[28] Performance Management – Principles Theory and Exercises - 01_OS86021EN54GLA00_Performance_Management_Principles_Theory_and_Exercises.pptx, 2014. Available: Nokia Intranet

[29] Nokia OSS - Nokia NetAct OSS Database Introduction, April 2012. Available: http://nokiaoss.blogspot.fi/2012/04/nokia-netact-oss-database-introduction.html

[30] Performance Monitoring – Theory and Exercises - 07_OS86027EN54GLA00_Performance Monitoring_Theory_and_Exercises.pptx, 2014. Available: Nokia Intranet

[31] T. Kuusela - OSS4 performance essentials.doc, 2005. Available: Nokia Intranet

[32] APM Digest - 15 Top Factors that Impact Application Performance, April 2013. Available: http://apmdigest.com/15-top-factors-that-impact-application-performance

[33] NetAct™ Operating Documentation – NetAct hardware, 2014. Available: Nokia Intranet

[34] NetAct 8 EP1 Capacity statement – Reference Hardware Configurations, 2014. Available: Nokia Intranet - NetAct_8_EP1_Capacity_statement_-_Reference_Hardware_Configurations.pdf

[35] NetAct™ Operating Documentation – NetAct Oracle database management, 2014. Available: Nokia Intranet

[36] Oracle – Database Administrator's Guide, 2014. Available: https://docs.oracle.com/cd/B28359_01/server.111/b28310/onlineredo001.htm#ADMIN11305

[37] E. Nemeth, G. Snyder, T.R. Hein, B. Whaley - Unix and Linux System Administration Handbook, Pearson Education, 2011, ISBN: 978-0-13-148005-6

[38] NetAct™ Operating Documentation – Introduction to NetAct, 2014. Available: Nokia Intranet

[39] M. Katara, M. Vuori, A. Jääskeläinen - OHJ-3060 Ohjelmistojen testaus, 2012. Available: http://www.cs.tut.fi/~testaus/s2012/luennot/OHJ-3066_2012.pdf

[40] J.D. Meier, C. Farre, P. Bansode, S. Barber, D Rea - Performance Testing Guidance for Web Applications, 2007. Available: http://msdn.microsoft.com/en-us/library/bb924357.aspx

[41] R.S. Pressman - Software Engineering, A Practioner's Approach, McGraw-Hill, 2010, ISBN: 978-0-07-337597-7

[42] International Software Testing Qualifications Board - Advance Level Syllabus Technical Test Analyst, 2012. Available: http://www.istqb.org/downloads/finish/46/94.html

[43] J. Heinonen - System PET test phases, 2014. Available: Nokia Intranet - System PET test phases.pptx

[44] Software Metrics, Available: http://www.sqa.net/softwarequalitymetrics.html

[45] SPS Metrics, 2014. Available: Nokia Intranet

[46] Tutorialspoint - Software Testing Methods, 2014. Available: http://www.tutorialspoint.com/software_testing/testing_methods.htm

[47] I. Molyneaux - The Art of Application Performance Testing, O'Reilly Media, 2009, ISBN: 978-0-596-52066-3

[48] NetAct Operating Documentation – Key Performance Indicators monitoring, 2014. Available: Nokia Intranet

[49] CA Technologies - Product brief: CA Wily Introscope, 2007. Available: http://www.ca.com/us/~/media/files/productbriefs/introscope_product_brief.aspx

[50] T. Kuusela, J. Kokkila, Roadmap of e2e PET automation, 2014. Available: Nokia Intranet - Roadmap of e2e PET automation 1 1.pptx

[51] SmartBear Software - Why Automated Testing?. 2014. Available: http://support.smartbear.com/articles/testcomplete/manager-overview/

[52] J. Lempiäinen, M. Manninen - Radio Interface System Planning for GSM/GPRS/UMTS, Kluwer Academic Publishers, 2002, ISBN: 0-306-47319-4

[53] L. Bibolini, H. Lancaster - Latin America – Telecoms, Mobile and Broadband Forecasts, March 2014. Available: https://www.budde.com.au/Research/Latin-America-Telecoms-Mobile-and-Broadband-Forecasts.html?r=51

[54] Nokia NetAct 8 EP1 Capacity statement, 2014. Available: Nokia Intranet – NetAct_8_EP1_Capacity_statement_-_Managed_Network_Reference_Information.pdf

[55] Apache Commons CLI library, 2014. Available: http://commons.apache.org/proper/commons-cli/

[56] Apache POI API, 2014. Available: http://poi.apache.org/

[57] Galmont Team – The Future of Software Testing, June 2014. Available: http://galmont.com/2014/06/the-future-of-software-testing/

[58] V. Held – 5G – beyond the hype, January 2014. Available: http://blog.networks.nokia.com/mobile-networks/2014/01/27/nsn-and-5g-positioning-paper/

[59] M. Alleven – Huawei targets 2018 World Cup in Russia for 5G trial, November 2014. Available: http://www.fiercewireless.com/tech/story/huawei-targets-2018-world-cup-russia-5g-trial/2014-11-20

[60] D. Bushell-Embling – Samsung aims to debut 5G at 2018 Olympics, October 2014. Available: http://www.telecomasia.net/content/samsung-aims-debut-5g-2018-olympics

[61] Huawei 5G whitepaper, 2014. Available: http://www.huawei.com/5gwhitepaper/

[62] A. Goldsmith, 5G and Beyond: What lies ahead for cellular system design, Conference presentation, Apr 2014. Available: Nokia Intranet – 24 April - 1330-1400 - Andrea Goldsmith.pdf