GSM HLR Unstructured Supplementary Service Data (USSD) service

Mohammadreza Taghilu
Abstract

GSM HLR Unstructured Supplementary Service Data (USSD) service

Mohammadreza Taghilu

The Unstructured Supplementary Service Data (USSD) provides a service additional to GSM main service (teleservices and bearer services). Similar to its counterparts, Mobile Station (MS) user and entities of GSM network operator act as main roles in order to establish communication in which the message traffic is transparent for the mentioned actors. USSD service uses real-time, session-oriented communication of messages. There are two possibilities in order to establish the USSD dialog: one is MS initiated (MI) and the other one is Network Initiated (NI).

In this thesis, the target is to develop the Home Location Register (HLR) part of USSD service using Erlang. The project consisted of a large amount of team work, large scale programming and testing in Telecommunications. The development was required to be based on Erlang projects including Mobile Arts' USSD Gateway (USSDGW) unit and Uppsala university Project CS 2011-a prototype of GSM voice call service-. Hence, the USSD Handler of HLR was built and accommodated according to handlers in USSDGW and MSC/VLR. The system was defined to be run and tested over SS7/SIGTRAN stack as the Signaling Transport underlying network used for routing and end-to-end delivery of packets. Thus, it was also required to setup, configure, and make the SIGTRAN stack up and running in a server. A cluster of Virtual Machines (VM) of different network nodes interconnected over SIGTRAN were used in order to verify the message passing while system testing. The message passing was checked through Wireshark and also MS display interface either over MI or NI USSD single operation or multiple operations.

The thesis gives both the identification of developed USSD services in HLR according to 3GPP specifications and the necessary integration with USSD Gateway along with the demonstration of rightness of Erlang intrinsic features using processes' message passing as the core functionality over the network and also functional programming.

The thesis can be extended towards HLR specific Operation, Administration and Operation support, Charging support, Product documentation, and also Supplementary Service management(service activation, deactivation, registration, erasure, interrogation and invocation).
To My Dearest Parents, & Brother...
## Contents

1 Introduction
   1.1 Background .................................................. 10
   1.2 Motivation ..................................................... 12
   1.3 Methodology .................................................. 12
   1.4 Scope .......................................................... 12
   1.5 Related works ............................................... 13
   1.6 The extention ................................................. 13
   1.7 Organization of thesis ...................................... 14

2 Preliminaries
   2.1 An introduction to GSM ...................................... 16
      2.1.1 GSM services ............................................. 16
      2.1.2 GSM technical architecture ............................. 17
         2.1.2.1 Signaling ............................................. 17
      2.1.3 GSM network structure .................................. 22
   2.2 An introduction to SS7 ...................................... 25
      2.2.1 Layers .................................................... 28
         2.2.1.1 MTP 1 (OSI Layer 1) ................................. 28
         2.2.1.2 MTP 2 (OSI Layer 2) ................................. 28
         2.2.1.3 MTP 3 (OSI Layer 3) ................................. 30
         2.2.1.4 SCCP ................................................ 30
      2.2.3 An introduction to SIGTRAN ........................... 37
         2.3.1 Layers .................................................... 38
            2.3.1.1 M3UA layer ....................................... 39
            2.3.1.2 SCTP .............................................. 43
         2.3.2 An introduction to USSD ............................... 44
   2.4 An introduction to Erlang .................................. 46
      2.4.1 OTP ....................................................... 46
      2.4.2 Mnesia .................................................... 48
### 3 Requirements

#### 3.1 Functional Requirements

- 3.1.1 Mobile Initiated (MI) USSD
  - 3.1.1.1 MMI-Mode
  - 3.1.1.2 Application mode
  - 3.1.1.3 MI Operation details
- 3.1.2 Network Initiated (NI) USSD
  - 3.1.2.1 Application mode
  - 3.1.2.2 NI Operation details

#### 3.2 Non-functional Requirements

#### 3.3 Behavioural Requirements

### 4 Design

#### 4.1 Software Design

- 4.1.1 Application platform
- 4.1.2 BSC
- 4.1.3 Hardware component
- 4.1.4 HLR
- 4.1.5 MG
- 4.1.6 MSC
- 4.1.7 MUS
- 4.1.8 VLR

#### 4.2 Decomposition Description

- 4.2.1 USSD_HLR component
  - 4.2.1.1 Processes

#### 4.3 Communication Design

- 4.3.1 Communication with USSDGW and VLR
- 4.3.2 TCAP
- 4.3.3 MAP
  - 4.3.3.1 Communication between MAP and its users

### 5 Implementation

#### 5.1 Plan

#### 5.2 Process

#### 5.3 Testing environment

#### 5.4 Problems

#### 5.5 Solutions

### 6 Conclusions and future work

#### 6.1 Conclusions

#### 6.2 Future work
Chapter 1

Introduction

1.1 Background

One of the rapidly flourishing telecommunication applications has been cellular communications initiating from 1G as the first generation which was formed based on an analog FDMA system for voice-only purposes. Later, the Global System for Mobile communications (GSM) was developed by European Conference of Postal and Telecommunications Administrations (CEPT) which introduced the digital mobile telephony standard for cellular users across Europe due to the existing incompatibility between different analog communications systems in 80s. Within the CEPT organization, the joint West-European venture started to define specifications of the GSM system via the form of recommendations for implementing the GSM system in each country. The recommendations were not sufficient due to compatibility issues when MS is roaming between countries, thus the memorandum of understanding (MOU) over recommendations and conditions were evolved in 1987 and further signed by 12 countries. Hence, the specifications and the MOU, as an established forum of discussions on pure operational matters, were transferred to ETSI (European Telecommunication Standards Institute). GSM-MOU was mainly aimed for a framework of all required measures in order to be taken into consideration for GSM development by signatories as an assurance for the opening of its commercial services through the contributing countries. In addition, network operators started implementing networks by 1991, and by 1995 roaming all over Europe was accomplished. [23]
The benefits of GSM were the increase of capacity by 2 to 3 times, because of improved digital techniques and appliance of frequency, compared to the first generation which was analog. Also automatic update location and higher flexibility of coverage through multiple BTS and BSC configuration were added. There was also a progress towards ISDN as an extension of PSTN (so-called PLMN).
1.2 Motivation

This thesis was aimed to investigate the possibility of implementing a prototype of the HLR part of the USSD communication service for GSM using SS7 over IP as a signaling bearer towards existing USSDGW implementation and MSC/VLR implementation. The service is developed to give the capability of performing half-duplex short message communication among GSM terminal equipments within a dialog which provides a service resided in GSM HLR or in the accessible network used by HLR via a USSDGW (GateWay).

1.3 Methodology

The methodology was comprised of investigating operation, requirements and architecture, detailed design, implementation, and testing of the project. Specifications of 3GPP USSD 2 phases \cite{2} \cite{3} \cite{9} \cite{10} \cite{11}, \cite{8}, data coding \cite{1} \cite{5} \cite{6}, and MAP \cite{4} were used for the first phase of development. Thereafter, development details of the project were specified based on such specs and Erlang design concepts and patterns. Then, Software Requirement Specification (SRS) and Software Design Description (SDD) reports based on IEEE standards, and progress plan of the project along with another peer thesis worker on MSC/VLR project were done; some parts of both reports related to this thesis were re-used in this report. Related works such as Project CS of students of Uppsala 2011, were used to be integrated and developed on. Hence the thesis followed the V-model for software development methodology. More details about the development will be explained in the next chapters.

1.4 Scope

When the complete USSD service is supported, the system will be able to benefit from high-speed message communication services such as WAP browsing weather forecast, balance query and re-fill. The communication is in between the mobile station (MS) and the USSD application platform using real-time interactive session-oriented mechanism which is independent from calling service. Other benefits of this service support is being economical over operational procedures and also allowing service carriers to provide local user requirements with the least possible modification on original configuration as USSD service is neither based on
phone nor SIM. USSD service is compliant with all GSM phones and is developed by a GSM standard implementation with no special requirement other than USSD centre. Furthermore, new value-added services can be implemented quickly by using menu configuration on the application platform.

This project includes implementation of USSD phase 2; handling of USSD dialogue including both Mobile Initiated (MI) and Network Initiated (NI); USSD Operations including Process Unstructured Supplementary Service Request (PUSSR), Unstructured Supplementary Service Request (USSR) and Unstructured Supplementary Service Notify (USSN); configured MSC and HLR routing of USSD Dialogue creating both MI USSD message (PSSR Request) and NI USSD message (USSR/USSN request); Handling of USSD (Dialogue Operation) timers; and using SS7 over IP as signaling bearer towards the Terminal Equipment (TE), the GSM HLR and the USSD-GW. (the latter provided by Mobile Arts, the product owner).

However, both HLR and MSC/VLR specific operation, administration, charging support, product documentation, and supplementary service management are out of scope.

1.5 Related works

Related work includes project CS 2011 [24] provided by Uppsala university students which includes an existing functional GSM call service, and also already existing USSD gateway provided by Mobile Arts. There are two main components of project CS 2011-GSM voice call service architecture that shall be used in this project as well, e.g. hardware component and virtual machine MSC/VLR.

1.6 The extension

The thesis was developed to be an extension over the course project CS-2011 as a prototype of basic GSM call service. MAP/TCAP encoders and decoders along with USSD dialog developed as an interconnecting Erlang application with USSD HLR component. USSD workers in HLR were designed and implemented for either incoming or outgoing messages, as both for network or mobile initiated processes, and local application in HLR. USSD gateway was also integrated properly.
1.7 Organization of thesis

Chapter 1: Introduction
A brief general history and background of the thesis along with the motivation, the development methodology, scope, related work to the thesis on which it was developed and a brief description of actual implementation of thesis included in 'The extention' section are given in this chapter. Current section as the last section is 'Organization of thesis' which tells about general overview of thesis contents.

Chapter 2: Preliminaries
A handy introduction of concepts, abstractions, relations, positions, tasks and data flow of components in GSM, SS7, SIGTRAN networks are given in this chapter. Erlang language is also explained briefly thereafter.

Chapter 3: Requirements
Functional along with non-functional requirements of the thesis in details and also behavioural requirements as depictions of former are given in this chapter.

Chapter 4: Design
Software design of existing/extended components, decomposition description of implemented components, and communication design between developed components are given in this chapter.

Chapter 5: Implementation
The description of development plan of thesis, actual process of the development, testing environment set-up, problems and impediments, and practical solutions of the latter are given in this chapter.

Chapter 6: Conclusions and future work
A wrap-up conclusion of gained knowledge, considerations, lessons and experiences in the thesis and possible future work over such are given in this chapter.

Chapter 7: Bibliography
References and materials which are used in the thesis are given in this chapter.

Appendix A: Glossary
Acronyms and abbreviations of concepts used in the report and their extended forms are given within this appendix.
Chapter 2

Preliminaries

2.1 An introduction to GSM

The essential elements of a GSM system consist of Mobile users, Base Station System (BSS), Mobile Switching Center (MSC), and the public voice and data networks. Some other important elements of the system include the operations and management center, the billing center, and the various networks (SS7 and X.25) interconnecting these subsystems.[23]

2.1.1 GSM services

Telecommunication services are services which are offered by a PLMN operator or service provider for its subscribers in order to fulfill telecommunications requirement. Two first are considered basic services and the latter is an add-on: (1) Teleservices, (2) Bearer Services, (3) Supplementary Services.[23]

<table>
<thead>
<tr>
<th>TELECOMMUNICATION SERVICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bearer Service</td>
</tr>
<tr>
<td>Basic Bearer Service</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

Figure 2.1: Categorisation of Telecom services [2]

The first one as teleservices is a type of telecommunication service which gives complete capability including mobile equipment functions used in order to communicate between users based on standardised protocols and transmission capabilities
according to agreement among network operators. There are all seven OSI layers specified for teleservices.[23]

The second one is a bearer service for transmission of signals between user network interfaces such as data service and Short Message Service (SMS), cell broadcast, and local features. This service includes only 1-3 layers of OSI which gives the transmission capability of signals among access points.[23]

The third one is supplementary service as add-on which is added to teleservices and bearer services as value-added services on top of teleservices and bearer services and may not be offered standalone.[23]

![Figure 2.2: Telecom service][23]

### 2.1.2 GSM technical architecture

#### 2.1.2.1 Signaling

Signaling is the means that telecommunication components use in order to talk with one another. User uses signals which are to be converted in the order of a special format for machines to be sent to a remote entity. The signals or a calling/called party identity are not part of the communication, i.e. payload. The major responsibility of signaling is to establish and clear a connection among end parties. The applications which are driven by signaling includes voice call and Supplementary Services (SS), as mentioned before.[19]
In order to establish a connection, a mobile station (MS) including a TE as GSM handset and SIM are needed which contain the functionalities of ciphering up to HLR. MS is peer of BTS that is also in communication with MSC and VLR directly through mobility management (MM) and call control (CC) protocols. MS is also responsible for data and fax connections through a transparent interface (Terminal Adaptation Function, TAF).[19]

When calling happens manually, signals are mainly direct current impulses created by variety of microchips, giving the central office a distinctive index between called numbers. Signals are usually transmitted on a medium serially and receiver does the remaining incoming functionality. Through time, the transmission speed has been increased. Bit, as the unit signal which can be created by electric voltage, is measurable by receiver during a time period that can be either low as 0, or high as 1 or other way around based on mutual agreement of transceivers. Hence, coding and transmission of complex messages happens through a sequence of bits which is applied in order to handle a process or send information. This is called Pulse Code Modulation (PCM) as a global standard process for sending and receiving digital signals, i.e. signaling data and payload. It can be formalized as hierarchical based on transmission rate. Signaling system No. 7 (SS7) and Link Access Procedures, D channel (LAPD) are one of those signaling channels that distinguish between the traffic channel from the signaling or control channel. Mentioned channels are overall named as subset of outband signaling that needs reservation of traffic channel. This makes further efficiency of resources since less resource time is needed to be engaged for traffic channel while call setup and it can be handled by one control channel. This contrasts from inband signaling in which control information that comes along with traffic channel. Considering, for example, a 1 Mbps frequency bandwidth and having 16 separate channels, there can be 64Kps for each channel and one of them for signaling data, thus a call establishment might only take 1 to 2 Kbps.[19]

- Signaling modulation
  Bidirectional communication of GSM is the most frequent form which provides simultaneously or quasi-simultaneously transmission and receive as a framework of full-duplex systems. In order to provide this, there are two applied fundamental duplex procedures: Frequency Division Duplex (FDD) which uses different frequency bands in either direction and it was used in analog mobile radio systems. The other procedure is Time Division Duplex (TDD) which sporadically switches direction in transmission.[15]
1. **FDD**

In order to establish communication among an MS and an BS, the frequency band is divided into two partial bands, for simultaneous send and for simultaneous receive. One is for uplink (MS to BS) transmissions band and the other for downlink (BS to MS) one. In this system, in order to benefit from proper distinction of both directions, the partials pairs of a connection are required to have enough frequency distance apart. Normally, the same antenna is applied for transmission. Thus, a duplexing unit need to be used for directional separation.[15]

2. **TDD**

Time duplexing is used in digital systems with Time Division Multiple Access (TDMA). In the system, the transmitter and receiver function quasi-simultaneously at dissimilar points over time. This is done by directional distinction through time switch between transmissions, and duplexing unit is emitted. There is this sufficient frequent switching which the communication acts on in order to maintain a quasi-simultaneous full-duplex connection. This happens through the sporadic interval $T$ for the transmission of a TS. Merely a small part of the connection can be used, thus a time duplex system needs more than two times of the bit rate of a frequency duplex system.[15]

3. **Multiple access**

In GSM, communication medium is radio channel. Due to competition over frequency resource by subscribers, and in order to avoid collisions, each of them is allocated a separate channel which is done through multiple access procedures, through either FDMA (Frequency Division Multiple Access), or TDMA. GSM uses mixture of channels of two-dimensional frequency modulation of FDMA and TDMA on the Air interface to transmit radio signals and data. This is because, e.g., when there is a mere FDMA system, for each user on a call there should be a dedicated frequency which itself results in an overload if there

---

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simplex</td>
<td>One way only</td>
<td>FM radio, television</td>
</tr>
<tr>
<td>Half duplex</td>
<td>Two way, only one at a time</td>
<td>Police radio</td>
</tr>
<tr>
<td>Full duplex</td>
<td>Two way, both at the same time</td>
<td>Mobile systems</td>
</tr>
</tbody>
</table>
are high numbers of call establishment requests. GSM also benefits from data packets with defined time binding and frequencies. Multiple conversations can be happening within the same frequency by assigning different TSs. Also GSM systems may have different concurrent receive and send sides.[15]

GSM also benefits from higher level of user privacy, and well-defined Open Network Architecture (ONA). The carriers contain spacing of 200 kHz. The bandwidth is 25 MHz providing 125 carriers in which the first one is not used due to interference with other systems. For each user there should be a channel, hence there are 1000 speech or data channels full rate coder or 2000 of the same half rate. The frequency bandwidth for uplink from MS to BS is 890-915 MHz and downlink 935-960 MHz from BS to MS. The power class ratings as mentioned above has the maximum transmission power of an MS. Initially it was specified to be five power classes for GSM 900, of which the most powerful class allowed for was 20W output. Nowadays it is mostly 8W at the most powerful class.[19]
In dual systems, there should be 8 mapped time slots for each frequency, hence when there is a halfrate configuration, for each frequency there are 16 time slots. This means that via FDMA, signal are being sent permanently by each user, but instead in TDMA, impulses similar to signal are being sent only periodically. For duplex service, two of TSs are needed in order to provide simultaneous transmission and receive both for downlink and uplink. There is a TS for each impulse which is also named as a burst, thus eight of them comprises a TDMA frame.[19]
BTS uses physical channels which are available TSs of a BTS. There is a correspondent TS for each physical channel. There are two variants of distinguishable channels, halfrate and fullrate. Physical channels piggyback logical channels each of which are in charge of distinctive tasks. While a call is happening, signals of MS are being sent periodically in a TDMA frame with the same burst position and TS towards the BTS. This is also the same for BTS in the opposite way. The channel mapping happens to be the same TS number of sequential TDMA frames. Logical channels are composed of traffic channels (TCHs) and control channels (CCHs). TCHs are placed among the various bearer services. However, in order to differentiate between CCH channels, it is needed to distinguish between various signaling requirements in different situations, but all of them other than some exceptions are defined as downlink or uplink.[19]

2.1.3 GSM network structure
In order to route incoming calls to the correct exchange switch and towards the called user, it is needed to have a well-designed telephone network. There are
following area networks for GSM: (1) GSM PLMN (HLR) service area; (2) MSC service area; (3) Location area (LA); and (4) Cells.[23]

Figure 2.7: System hierarchy [15]

There are several types and numbers of PLMN service areas (distinguishable by the size) which altogether, can be considered as a country-level PLMN area. There are links among GSM-PLMN network and other PSTN, ISDN, or PLMN networks which are on the level of (inter)national transit exchange.[23]

The next subsection of GSM is PLMN and in GSM-PLMN, incoming call is handled by a transit exchange called Gateway MSC. Also all call connections between PLMNs or to fixed networks and mobile terminated calls must be routed to the same gateway. In short, the gateway MSC contains the interworking functions to make such connections as well as routing incoming calls to proper MSC inside the network.[23]

The next area is MSC/VLR service area as a subsection of GSM-PLMN which is the sole call controller of its domain, even though there can be multiple of them in one PLMN area and is used for routing a call to a mobile user. Such route is the path through the network links of the registered user in MSC area.[23]

LAs are next areas within a MSC/VLR configuration. MSC/VLR is made of multiple of LAs. LAs are the areas that MS can roam without location update required and it is the area that paging message can be broadcast for finding the
called user and it can be identified by Location Area Identity (LAI) which is used to check the status of user.[23]

Finally cells are division of LA which are BTS identities. MS uses Base Station Identification Code (BSIC) to make distinction between them. Cells are basically main idea of a cellular network and are aimed to divide the frequency spectrum, in order to allocate divisions to compliant BTSs. Also one fundamental purpose of such planning is to decrease interference between BTSs considering that attenuation of frequencies are inevitable and with low power emission, the coverage of maximum of 5 Km around a BTS is plausible.[23]

A GSM system has three main subsystems: Network, radio and operation support subsystems. Within the subsystems, there are three main interfaces: A-interface between BSC and MSC, A-bis between BSC and BTS, and Um between BTS and MS. More interfaces are named in the below figure.[23]

Structure of a GSM network

![GSM structure](image)

Figure 2.8: GSM structure [25]

In the network subsystem (NSS) which is the central component of each mobile network, there are equipments and functions which are for end-to-end calls, sub-
scriber management, mobility, and some interfaces with the fixed PSTN. Switching subsystem includes MSCs, Visitor Location Register (VLR), Home Location Register (HLR), Authentication Center (AuC), and Equipment Identity Register (EIR). MSC is responsible for call setup, routing, and handover between BSCs. The HLR is responsible for storing all users registered in the PLMN. The VLR is for all mobile users which are roaming in the MSC. The difference is that VLR preserves more precise data of the mobile location. The AuC which is connected to the HLR, provides authentication parameters and ciphering keys. The EIR is the storage for the International Mobile Equipment Identity (IMEI) numbers of whole registered mobile equipment. There is also an interface which is between MSC and other networks such as PSTN and ISDN which is called Interworking Function (IWF). The SS7 network is used to interconnect subsystems either directly or indirectly. There is slightly more flexibility of NSS structure than the hierarchy existing in the structure of the BSS. There could be multiple MSCs for a single shared VLR; it is optional to utilize EIR, and expected number of subscribers specify required number of HLRs. Interfaces of NSS are all virtual, i.e., reference points for signaling among the network elements.[19]

Radio subsystem is the second subsystem which has the equipment and functions for connection management on the radio path. It is comprised of a BSC, BTS, and the MS. For managing mobility, MS is responsible for holding dialogues with the NSS which has the capabilities of network termination and mobile termination. As there are radio cells for each BTS with several transceivers, multiple BTSs are controlled by one BSC which has multiple configurations joint together. The former and latter entities is together called BSS viewed by MSC via a single interface. Such interface is an entity for communication with MSs associated with the radio management, transmission functions and radio link control and quality assessment. Some are for high traffic, and some suited for moderate-to-low traffic areas. A BSC is a controller of handover and power control.[19]

2.2 An introduction to SS7

Signaling System Number 7 is both signaling and controlling network which gives the capability of OSI Layers from 1 to 3 as fundamental layers for the signaling traffic on all NSS interfaces, and also A-interface. Here "control" means the capability of controlling features and tasks in a remote telephone switch or centralized computer. [19]
The user of SS7 is normally another telephone switch or computer system. The SS7 network is composed of directly connected signaling points (SPs) which are connected through signaling transfer points (STPs) or a mixture of SPs and STPs. An SP is a network entity which has user parts such as SCCP and ISUP that provide the processing of messages addressed to that SP. The MSC, the BSC, and the exchange of the PSTN are also placed in this group. [19]
The functionality of the STP is identical to those of the SP, with the extra capability of relaying SS7 messages. STP gives network access and routing, screening and transfer of signaling messages. It is also possible to have a STP without SP functionality which is solely responsible of relaying messages. [19]

![SS7 configuration](image)

**Figure 2.11: SS7 configuration [15]**

SS7 messages are transmitted over 56 or 64 Kbps bidirectional channels as signaling links. Dedicated channels or networks signaling is performed out of band and for voice channels are carried in band. Out-of-band signaling gives better call setup times, and also more efficiency in use of voice circuits. Signaling links are established after that the interconnection of signaling terminals in the traffic switching is finalized through physical circuits. Signaling links consist of signaling points (SP), signaling data link, and the equipment interconnecting the SPs and signaling link.[14][19]
SS7 without user parts is equivalent to OSI layers 1 to 3 along with SCCP which altogether is called as Classical Telephony Signaling Protocols (CTSP). Those three layers are mainly represented through the message transfer part (MTP). The MTP of SS7 also has the responsibility of the following tasks in general:

- All the necessary functionalities of OSI layers from 1 to 3 which are needed to support a reliable signaling data transport to the multiple SS7 user parts are provided by MTP.

- In order to suppress the problems, the MTP takes the major steps in order to maintain and prevent loss of data.

2.2.1 Layers

The MTP may have three layers which are:

2.2.1.1 MTP 1 (OSI Layer 1)

This layer is in charge of the transferring single bits or defining and provisioning of needed physical, electrical, and functional features of the digital signaling link. Possible physical interfaces are E-1 (2048 Kbps; 32 64-Kbps channels), DS-1 (1544 Kbps; 24 64-Kbps channels), V.35 (64 Kbps), DS-0 (64 Kbps), and DS-0A (56 Kbps). This level is used to provide the requirements as physical circuit, e.g. Pulse Code Modulation (PCM) channel.

2.2.1.2 MTP 2 (OSI Layer 2)

This layer provides end-to-end exchange of messages across signaling links which themselves are in charge of distinguishing between messages, error detection, error correction, and supervision. This leads to the fundamental frame structure that SS7 uses for whole of different message varieties. The mentioned frame has beginning and ending flags, an acknowledgment field which has sequence numbers for transmission, a length indicator, an optional field, and the FCS (Frame Check
Sequence) in order to transport a checksum. Such types of messages are SS7 message or a signal unit (SU), and the most important one is MSU. [14]

MSU is used for transferring data among different network entities and is the mere SS7 message unit which is capable of carrying traffic data that is used by all user parts (ISUP, SCCP, OMAP). It carries call control, query and response, network management, and network maintenance data in the signaling information field (SIF). There is a signaling label inside SIF field which gives the originating SP the capability of sending information to a destination SP across the network. LI in MSU is a 6-bit field which indicates the number of octets in combination with the LI field up to cyclic redundancy check (CRC) field. Overall, this determines what kind of SU is. There are also subfields such as: backward sequence number (BSN) used for acknowledging the receipt of SUs by the remote SP, backward indicator bit (BIB) used for indication of a negative acknowledgment, Forward sequence number (FSN) as indicator of sequence number of SU, and forward indicator bit (FIB) for error recovery together with BIB. There is also a service information octet (SIO) which is merely in MSUs. It consists of two 4-bit subfields: the service indicator (SI) and the subservice field (SSF). The former identifies the MTP user, thus gives the information in the SIF to be decoded. The SIF is solely present in MSUs which carries the information to the MTP user part. It consists of 8 or 16 bits. The CRC is used to detect correct data transmission errors.[14]
2.2.1.3  MTP 3 (OSI Layer 3)

This layer is used for routing messages among SPs in the SS7 network. It provides re-routing traffic from failed links and SPs and it also controls traffic in case of congestion. MTP3 level protocols are ISUP, SCCP, TUP which normally apply SS7 MTP3 services. There are two signaling network functionalities as signaling message handling and signaling network management performed by MTP3: [14]

1. **Signaling Message Handling:** This functionality directs the user data part of the received MSU towards (A) the successor SP or (B) at a terminating SP, the right User Point (UP) such as ISUP or Application Provider (AP) such as TCAP through SCCP. In case that an MSU is pointed to the other SP in the network, then the routing is done towards the correspondent signaling link which is based on instructions given by the function of signaling network management. An MSU which is going to be transmitted from a SP is routed towards selected signaling link, and the selection is based on MSU’s Destination Point Code (DPC) which is the signaling network node that the message needs to be reached. On the other hand, Originating Point Code (OPC) is the signaling network node from which the message is sent. Those messages which their DPC is different from the receiving one are routed to the correct DPC, provided that the receiving entity has message transfer capabilities (like an STP). The selection of the outgoing link is based on information in the DPC and the SLS. Signaling link selection (SLS) directly binds the signaling network to the traffic circuit.[14]

2. **Signaling Network Management:** This function gives continuous supervision over the signaling network in order to detect errors and abnormal situations. [14]

3. **Routing Label:** There are four types of routing labels as (A)MTP management messages, (B)TUP, (C)ISUP (circuit-related) messages, and (D)SCCP messages.[14]

2.2.1.4  SCCP

Signaling Connection Control Part (SCCP) is solely utilized with TCAP, but also can be used with ISUP. SCCP is used for end-to-end routing, since MTP can only be used for point-to-point routing. Addressing for routing a message is provided by SCCP which is utilized at each SP through MTP level 3 in order to specify linkset.[19]
SCCP messages are encapsulated in the SIF field of MSU which begins with a type indicating octet field SCCP message in combination with data, and its coding which is based on message type having elements: AI, PC, GT, and SSN.[14]

1. **Address indicator (AI)** which is an indicator flag of the translation type necessary for SCCP.[14]

2. **Point code (PC)** which is DPC in an SCCP address and does not need any translation and considered as a measure if the message is either aimed at that SP or requires to be routed forward over the SS7 network via MTP. If messages are outgoing, the DPC is same as the MTP’s DPC.[14]

3. **Global title (GT)** which contains called digits or another form of address that might not be recognizable by SS7 network. The translation is needed if the associated message should be routed over the SS7 network. This translation makes a new DPC and maybe also a new SSN and GT.[14]

4. **Subsystem number (SSN)** which is an identifier of a local subsystem accessible by SCCP inside the node, such as VLR or MSC. If based on the DPC of the incoming message, the message is directed to the right node, then SSN needs to say which of SCCP users is related to the message.[14]

**Addressing and Routing of Messages:**

In an SS7 network, exchanging of MSUs does not happen solely among close neighbors (SP/STP). The MSC and BSC are neighbors, but the information exchange among the MSC and the HLR could involve multiple STPs. SS7 applies point codes (PC) in order to route and address MSUs. PCs are singleton identifiers in SS7 network. In case that DPC in a message gives the information of receiving SP,
the message is transmitted to the aimed user part (e.g., ISUP or SCCP) identified by the service indicator in the SIO. For each SP and STP, there is merely one point code, a signaling point code (SPC). Routing of messages in MTP layer 3 is done by the routing label in the SIF of MSUs. An MSU owns a routing label including the PCs of the sender of the MSU, the OPC and the addressee of the MSU or the DPC and SLS. Addressing as in national basis happens merely in between SPCs. However, for higher levels, international addressing is needed, especially SCCP or ISUP, in order to supply necessary features. There are also 4 extra bits of routing label which compose SLS field that is used in order to balance the load among multiple SS7 connections of a link grouped together with the DPC, for the selection of the outgoing link. This ensures that message sequencing will occur. This can be like even values assigned for one network entity at one side and odd ones for the other. SS7 uses MTP and SCCP routing (PC, GT, or SSN routing). One, two, or all of these elements might be inside the calling or called party addresses which is based on the AI field. [19]

Layers of MTP 1 to 3 are used by SCCP. Subsystems in NSS uses such services, specially through the BSS application part (BSSAP) on the A-interface and through the transaction capabilities application part (TCAP) along with the mobile application part (MAP) on multiple interfaces inside the NSS. ISUP is also able to use the SCCP, but the trend of applications is towards applying the combination. SCCP using its layer 3 functions gives end-to-end addressing even in between multiple network nodes and countries. Also, it provides a distinguishing criteria in between multiple applications inside a network node in which the SCCP addresses such applications as subsystems. There are two service classes defined as connection-oriented and connectionless to the users of SCCP for actual data transfer. [19]

Additionally, SCCP has management functions for administrative tasks which are independent from those known from the SS7 signaling network management. SCCP also has features of layer 4, such as error detection and optional segmentation of the data mechanisms for transmission. Two network entities may establish a virtual connection between the two subsystems for transaction 1, 2, or 3 in the connection-oriented mode. Reference numbers, the source local reference (SLR) and the destination local reference (DLR) are identifications of each connection. Whenever the connection is active, exchanging data among network entities and also addressing individual transactions may happen. SCCP’s responsibility is to receive data from the MTP and forward it to the referred subsystem in which the input data is related to the multiple active transactions. As an example for connection-oriented transactions, there is this location update procedure. In con-
nectionless services, since in this case SCCP should not provide referencing, the
receiver of a message needs to allocate it to an active process, and there are ex-
amples such as paging in BSSAP, SCCP management, and TCAP protocol. [19]

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
<th>bit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Reserved for national use</td>
</tr>
<tr>
<td></td>
<td>Routing indicator</td>
<td></td>
<td></td>
<td>Global title indicator</td>
<td></td>
<td></td>
<td></td>
<td>(value for MAP inter-PLMN = 0100)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SSN indicator</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Point code indicator</td>
</tr>
</tbody>
</table>

SPC (signaling point code)/bits 1–8 of 14

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>SPC (signaling point code)/bits 9–14 of 14</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>byte 2</td>
</tr>
</tbody>
</table>

(SSN) subsystem number (e.g., BSSAP, EIR, HLR, SCCP management)

How to translate the global title (translation type)

Numbering plan (e.g. ISDN; E.163/E.164) How is the global title coded (e.g., BCD, even/odd number of digits)

Even/odd number of digits Structure of the address information (international number, subscriber number, …)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th>byte 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>2nd digit</td>
<td>1st digit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>byte 8</td>
</tr>
<tr>
<td>4th digit</td>
<td>3rd digit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>byte 9</td>
</tr>
<tr>
<td>......</td>
<td>......</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

‘1111’ (fill digit ↔ in case of odd number of digits) last digit

byte N

Figure 2.15: Format of Calling-Party Address (CaPA) and the Called-Party Address (CdPA) [19]
The protocol class parameter is the distinction parameter for connection-oriented or connectionless services within the SCCP. In case of the comparison between connection oriented or connectionless services, we may suppose an analogy here in which MSUs act as envelopes and SCCP message as letter, and there would be a mailing system as a means of propagation process. Thus, in connection-oriented services, two parties tries to make a virtual connection which is not actually a permanent, dedicated physical path, and the attempt for connection is done with a unique reference. On the other hand, when there is a connectionless service, no response and no reference for an issue exist. If there is a connectionless SCCP transaction, the SCCP messages has the header of SCCP addressing information including called and calling party addresses and local references.\[19\]

SCCP’s address information is static unless the PC and SSN turn into unknown to the originator. If this happens, an STP should be providing translation which is the case when a number is called, that is, the message is in unrouteable state in

Figure 2.16: SCCP addressing based on GT \[17\]
the network. The address of SCCP has three sections: called/calling party, PC, and SSN. The routing usually is based on all of them but when it is happening by PC, the address is a mixture of the PC and SSN. The usage of SCCP calling party address is to give the origination of SCCP message. The address translation happening in intermediate node leads to the OPC of the MTP 3 routing label modified into the PC of the intermediate node. It is the role of SIF of MSU to host SCCP message, and also SIO (which is out of SIF) plays the role of the user identifier part of SCCP. CAlling-Party Address (CaPA) and the CAlied-Party Address (CdPA) have identical format and are the identifiers of the address of message and its type and SCCP’s end-to-end addressing. The SPC has 2 bytes, SSN has 1 byte, and GT has more than 3 bytes. MAP utilizes all possible combinations, but the BSSAP needs merely the SPC and the SSN (= BSSAP) for addressing.[19]

After dialling a number, it is not possible to route the call based on ordinary routing methods since the numbering plan applies the area code of a number in order to specify the route to destined area of country’s network. The prefix here normally identifies a to-be-routeable central office for the call to the subscriber. If the area codes were not included in the called number, the SS7 network needs to supply a routing number by which the end exchange can route the call that itself needs TCAP and SCCP services. As aforesaid, dialed digits are provided by called party address of SCCP, even if not all of the digits are needed and the only required ones would be area code and the prefix. There is a database which is queried for the number in order to provide the routing number for TCAP. The routing number is then given to the exchange via TCAP and SCCP, thus a connection might be set up for the call.[19]

While a TCAP message is being routed, the SP should be identifying the destination. There might be a case that there is no called digits associated with the transaction. Even in such a case, SCCP is still in charge of providing the addressing required by MTP for routing a TCAP message through the network.[19]
Figure 2.17: SCCP message format UDT with GT [17]

<table>
<thead>
<tr>
<th>NA (Nature of Address)</th>
<th>NP (Numbering Plan)</th>
<th>TT (Translation Type)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000 0011  National</td>
<td>0001 = E.164 (GT)</td>
<td>Usually coded as</td>
</tr>
<tr>
<td>0000 0100  International</td>
<td>0111 = E.214 (MGT)</td>
<td>0000 0000 for GSM</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ES (Encoding Scheme)</th>
<th>Address Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001 BCD odd number</td>
<td>000x 00x1 DPC Included</td>
</tr>
<tr>
<td>0010 BCD even number</td>
<td>000x 001x SSN Included</td>
</tr>
<tr>
<td></td>
<td>0001 00xx GT Included</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mandatory Fixed Part (Protocol Class)</th>
<th>Message Type (SCCP)</th>
<th>SSN (Subsystem Number)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic CL class</td>
<td>CR</td>
<td>0000 0110 HLR</td>
</tr>
<tr>
<td></td>
<td>CC</td>
<td>0000 0111 VLR</td>
</tr>
<tr>
<td>Sequenced (MTP) CL class</td>
<td>CREF</td>
<td>0000 1000 GMSC, MSC</td>
</tr>
<tr>
<td>Basic CO class</td>
<td>RLSR</td>
<td>0000 1001 EIR</td>
</tr>
<tr>
<td>Flow control CO class</td>
<td>RLC</td>
<td>0000 1010 AUC</td>
</tr>
<tr>
<td></td>
<td>DT1</td>
<td>0000 1100 SC</td>
</tr>
<tr>
<td></td>
<td>UDT</td>
<td>1111 1110 MSC-BSC</td>
</tr>
<tr>
<td></td>
<td>IT</td>
<td>BSSAP</td>
</tr>
</tbody>
</table>

Map:
For routing using an SCCP network node, specially for international connections, an identification is needed. This identification which is called global title (GT) is composed of regular directory number and the required information of how to interpret the number. One of the fundamental parts of GT is protocol class (PC) which is the indication of the service class of a message. There are four protocol classes (0, 1, 2, 3). Classes 0 and 1 are the representing the connectionless services, and on the other hand, 2 and 3 are for connection-oriented services. Classes 0 and 2 constitute the fundamental version and classes 1 and 3 constitute additional data security in the form of acknowledgments. GSM Map utilizes protocol classes 0 and 1. Bits 0 through 3 define the real protocol class. Bits of 4 to 7 are not for connection-oriented service classes 2 and 3, but at the same time, they are needed for the connectionless service classes 0 and 1, if a UDT message needs to be replied in case of an error. [19]

The SSN tells the user from which SCCP message comes from or needs to be addressed. The SSN may be a section of CaPA/CdPA or a section of an SCCP management message. In order to establish a SCCP connection, one side needs to request a SCCP connection by CR (Connection Request) messages, which has SLR as a reference number. The called party agrees by sending CC (Connection Confirm) and its DLR. Also, in order to release a SCCP connection, an RLSD message is propagated, and the receiving side confirms by RLC message if the RLSD message was received, and thus the SCCP connection was released, and resources were cleared.[19]

Dialed GT digits are the called party address. PC and SSN are needed for the message in order to reach the destination. In case that originated message is not aware of its PC or the SSN of the database (which provides a routing number for the requesting exchange), for routing purposes the GT digits needs to be utilized by MTP level 3. Whole of such process is called GT translation (GTT) and it is provided by the STP close to the destination database.[19]

2.3 An introduction to SIGTRAN

By introduction of Voice-over-IP (VoIP) technology, the interworking of PSTN and PLMN was developed through an SS7-based-over-IP (SS7oIP) as a signaling protocol aimed for interworking among a SS7 MTP-based PSTN or PLMN and SS7 IP-based networks (SIGTRAN=SIGnaling TRANsport). SIGTRAN is based on the Stream Control Transmission Protocol (SCTP) transport protocol as an asso-
ciation and stream-based protocol to meet the rigorous requirements of telephony signaling networks over IP network along with multiple User Adaptation Layers (UALs) such as MTP3 User Adaptation Layer (M3UA). This is a replacement of MTP or MTP+SCCP or even MTP+SCCP+TCAP for the Internetworking. The UALs are used to support ”primitives” (a fundamental interface or segment of code that may be used in order to construct further advanced and complex program elements or interfaces) which is also accompanied by providing multiple signaling protocols to use the SCTP layer.[14]

SIGTRAN as a standardization of internet telephony protocol stack is utilized for transporting Signaling System 7 (SS7) traffic (MAP, CAP, INAP, ISUP and BICC) and GCP over an IP bearer/Internet. SS7 signals is made of specific commands for handling a telephone call. This standard consists of the O&M actions, functions and tasks that are required for handling the IP data and addressing, the SCTP Layer and the M3UA. The flow is that after reception of SS7 signals by a signaling gateway(SG), SG turns the signals into SIGTRAN packets in order to transfer them over IP towards the next SG or in case that PSTN is not the aimed destination, packets will be propagated towards a softswitching mechanism.[14]

2.3.1 Layers

User Adaptation Layer (UAL) are a set of protocols which are in charge of encapsulating the upper layer telephony signaling protocols that are needed to be transported over SCTP/IP. The adaptation is implemented via making the being relayed upper signaling protocols uninformed of the lower layers which themselves are separate from the original lower telephony signaling layers.[14] In case that a MTP user is being propagated over the IP network, the UAL (which needs to be used in order to transport the MTP user) will have the identical upper interface as MTP owns. There are various UALs for an SS7 based service in order to provide interworking with the two signaling transport protocols (PSTN or IP):[14]

1. **TUA** (TCAP User Adaptation layer) which is not defined yet would be similar to M3UA but for TCAP user application parts, such as MAP or INAP;

2. **SUA** (SCCP User Adaptation layer) which is similar to M3UA but for SCCP user protocols such as TCAP which has the responsibility of routing over IP along with routing on PC and routing on GT;

3. **M3UA** (MTP 3 User Adaptation layer);

4. **M2UA** (MTP 2 User Adaptation layer) which is is similar to M3UA but only for the usage of MTP 3 protocol layer. It defines a protocol which is
aimed for transporting SS7 MTP2 user signaling over IP through the services of SCTP;

5. **SCTP** (Simple Control Transmission Protocol).

### 2.3.1.1 M3UA layer

There is a standard framework architecture for circuit switched network signaling transport over IP which utilizes several components, i.e. a common signaling transport protocol and a supporting adaptation module of the services. MTP 3 user adaptation layer (M3UA) is used for transporting signaling messages of any protocol layer over the IP network. M3UA applies the SCTP services as the underlying reliable common signaling transport protocol in order to use several SCTP benefits.[20]

There are following common terms regarding M3UA[20]:

- **Destination**: This is an entity or node which is identified with a SPC.[20]
- **Linkset**: A number of directly interconnecting SLs of two SPs, which makes a module overally. [20]
- **Signaling End Point (SEP)**: A SS7 network node associated with an originating or terminating local exchange (switch) or a gateway exchange.[20]
- **IP Signaling End Point (ISEP)**: An IP node which implements SCTP and UAL.[20]
- **Signaling protocol user (SPU)**: The application over the UAL.[20]
- **Association**: An SCTP association which is essentially a transport connection of delivery among two SCTP peers for M3UA data units and M3UA adaptation layer peer messages. There are 4 states of such an association as Down, Established, Inactive and Active.[20]
- **Stream (An SCTP stream)**: A one-way logical channel established among SCTP peer nodes in SCTP association in which all user datagrams are placed inside with sequential delivery.[20]
- **Application Server (AS)**: This is a logical entity which is in charge of a specific routing key and signaling for a scope. It consists of one or multiple unique ASPs of which one or multiple is usually actively processing traffic. An example could be a virtual database element, which is in charge of all HLR transactions for a specific SS7 SIO/DPC/OPC combination. There is a one-to-one relationship among an AS and a Routing key. [20]
• **Application Server Process (ASP):** This process is an instance of an AS which is in charge of handling an active or backup process of an AS, e.g. process instances of MGCs, IP SCPs, or IP HLRs. It consists of an SCTP endpoint and is capable of processing traffic for multiple ASs. It can also be distributed over several IP addresses. ASP can have the configuration of signaling traffic associated with multiple AS over a one SCTP association. The discrimination of signaling traffic which is about to start or stop, performed through the routing context parameter in ASP’s active or inactive management messages. The routing context parameter is the unique identifier of the range of signaling traffic related to each AS in which there is a configuration of which ASP to receive. ASPs needs to benefit from excessive or load-sharing configuration which are distributed over different hosts. All ASPs and the connected SG may share the same PC using a single SS7 signaling endpoint, however, it is also possible that all ASPs may be defined to share a PC which is not equal to that of the SG. This happens by defining ASPs to have a single SS7 SEP, and SG is defined as an STP.[20]

• **Signaling Gateway (SG):** This is a signaling agent which is in charge of transmitting circuit switched network native signaling over an IP Network. It is equivalent to SP in SS7. SG can be a group of routing nodes over SIGTRAN. It may consist of a set of several SG processes, of which one or multiple of them is usually actively processing traffic. SG may have more than one SGP, then SG is a logical entity, and such SGPs are coordinators of a single management SS7 network and also supported ASs. SG can be called an SS7 PC in MTP management. Then SG’s PC is utilized for addressing local MTP3-Users, e.g. a local SCCP layer.[20]

• **Signaling Gateway Process (SGP):** This is an instance of an SG process which is in charge of handling an active, backup, loadsharing process, or broadcast process of an SG.[20]

• **IP Server Process (IPSP):** This is an instance of an IP-based application process which is basically identical to an ASP, with the only exception of its point-to-point utilization of M3UA. Conceptually, an IPSP is not applying the services belonging to SG node.[20]

• **Host:** The processing platform of SGP, ASP, or IPSP.[20]

• **Signaling process:** A process instance that uses M3UA to communicate with other signaling processes. An ASP, an SGP, and an IPSP are all signaling processes.[20]
• **Layer management**: This is a functionality of a node which is in charge of handling the inputs and outputs among the M3UA layer and a local management node.\[20\]

• **Routing key**: A SS7 parameters group and values that are utilized for identifying a signaling traffic range which itself needs to be handled through a specific AS by filtering SS7 messages. Such key can be DPC, DPC+OPC, or DPC+OPC+SI which is obtained from an analogy of OPC, DPC, and SIO in the MTP3 routing label.\[20\]

• **Routing context**: A unique identifier value of a routing key, which are configured either applying a configuration management interface, or the routing key management procedures; the specific identifying information of such are application and network dependent. It is a 4-octet value (integer) which is associated with Routing key in a one to one relationship. It is an index of a sending node’s message distribution table which consists of the entries of routing key.\[20\]

• **Route**: A route is an identifier of outgoing association which is necessary for transmission of a message towards a particular destination node. Such a route may have these states: Inactive, Active-Unavailable, and Active-Available. The distribution of SS7 messages between the SGP and the ASs is determined by the Routing keys and their associated Routing Contexts.\[20\]

• **Network appearance**: This is a mechanism which is aimed for distinction of signaling traffic among an SG and an ASP, such as international SG. This mechanism is a shared M3UA local reference by SG and AS as an integer, which combined with SPC is the identifier of an SS7 node through particular superset SS7 network. This is also a distinguisher of signaling traffic belonged to a separate network among SG and ASP over a common SCTP association.\[20\]

• **Signaling Point Management Cluster (SPMC)**: Whole of ASs in the SS7 network are identified by one MTP entity (SP) in a single particular network appearance. Such a cluster is utilized to combine the availability, congestion, and user part status of an SP, distributed in the IP domain in order to provide MTP3 management procedures into SS7 network. The SG can be a member of the SPMC.\[20\]
M3UA Services

This layer in an ASP or IPSP gives identical group of primitives on the upper layer for the M3UA services, e.g. ISUP, SCCP, TUP, and TCAP in analogy to the MTP3-Users in an SS7 SEP. That is, mentioned protocol layers at an ASP or IPSP are not informed about whether MTP or M3UA/SCTP/IP being used. M3UA layer is in charge of providing MTP level routing for SS7 messages which is done by routing messages and their states along with redundant routes to destination nodes; holding the the M3UA associations and their states, the availability and connectivity of the local SCTP layer; mapping between IP addresses and SPCs; and configuration of signaling routes between local and peer nodes. There are two operational modes which M3UA supports, one in which it bridges the boundaries of the SS7 network and IP network through transporting the signaling messages from an SG located at the edge of the SS7 network to an IP-resident gateway, and in the other mode, it is also responsible for communication inside IP network for SS7 call signaling protocol such as SCCP, which makes a supervisory function communication towards other nodes. M3UA supports no single point of failure, that is, SGs are to be set up at least in pairs. In a remote IP-based application, M3UA gives an extension to the MTP3 layer services. The packet consists of a common message header and a mandatory parameter Protocol Data. The latter has the information pertaining to the MTP message, i.e. the SIO and the Routing label. The fields OPC, DPC, SI, NI, MP, and SLS are identical to the MTP ones. The User Protocol Data field holds the MTP-User information such as SCCP or ISUP message. This is identical to the User Data field in the MTP one.

Figure 2.18: Message headers of M3UA and lower layers [22]
2.3.1.2 SCTP

Simple Control Transmission Protocol (SCTP) as an IETF developed protocol is applied for a real-time transport-level datagram transfer protocol, and is made of multiple independent message streams with sequential delivery, operating on top of a datagram service such as IP, instead of TCP or UDP. It provides reliable IP connection-oriented, message-based services communication between SCTP peer nodes (IP SPs). SCCP supervises connections applying SCTP associations. SCTP is capable of association beginning by a request from SCTP user using cookie mechanism and tearing it down. It might also use segmentation of multiple user datagram within SCTP datagram if necessary in order to guarantee that SCTP packet are transmitted to MTU (Maximum Transmission Unit) which is the largest datagram size sendable over a network.[20][21]
It essentially provides several transport addresses in association site(s). SCTP is defined with user adaptation (UA) layers in order to support interworking features of the SS7 network and is used in association with M2UA and M3UA protocols. SCTP uses TCP-friendly congestion control e.g. slow-start, congestion avoidance, and fast retransmit. However, there is a difference between TCP and SCTP which is in their delivery mechanism. SCTP utilizes explicit packet-oriented delivery, but TCP applies stream-oriented one. Also SCTP has the capability of detection and prevention of network flooding. It also gives the capability of acknowledged, error-free and non-duplicated transfer of messages.\[20\]\[21\]

### 2.3.2 An introduction to USSD

Unstructured Supplementary Service Data (USSD) is a Supplementary Service (SS) which modifies or enhances a basic telecommunication service. SS must be offered along with such services. USSD is defined to enable conversational (session-based) text services of communication between telecom equipments of service providers of 2G (or more). ”Conversational” means that the terminal equip-
USSD (TE) may begin a session, get it back and reply with an available choice, reply to that response, etc. Also it is possible to use USSD services for prepaid users while roaming into another country (another HLR/PLMN area) which is not likely in SMS service. In addition, there might be no charges for such a roaming for the user in case of using USSD service. The bandwidth requirement is way lower than needed for WAP services, but much more than SMS services. It is similar to a maintained voice connection in the period of a session on the air interface using solely MAP messages without engaging a circuit. In addition, it is the USSD platform and not SMS system in the home network which is able to initiate a dialogue with its users. USSD is a flexible and unexpensive solution for small to medium sized operators that are aimed to present efficient services to the users instead of heavy slow portals. USSD platforms are usually can be equipped with application programs using in-house department resources. The interactive applications can be implemented by any appropriate language for Internet access. [14] [9]

USSD contains messages which can be transmitted up to 182 alphanumeric characters per message in length. Such messages establish a USSD session as a real-time connection of exchanging data. This session-oriented structure of connections is the main difference between SMS and USSD. Instead of store and forward strategy via SMSC gateway, USSD messages are transmitted directly towards an application platform which handles the USSD service. Such service can be in sending-side network or in visited network. The communication over the radio interface happens on the signaling channels through short dialogues with peak data throughput rate of up to approximately 600 bits/s outside of a call and 1 000 bits/s during a call. [14] [9]

The session is much helpful in building interactive Man Machine Interface (MMI) applications. Invocation of USSD service can be done either as (1) Push service: by Mobile user or Mobile-Initiated (MI) USSD service, or (2) Pull service: by Network entity (USSD gateway platform/HLR/VLR/MSC), or Network-Initiated (NI) USSD service. [14]

Also there are two modes of USSD, MMI-mode and Application mode:

1. **MMI-mode** USSD used in order to execute user given transparent transport of MMI strings to the network (MI) and transparent transport of text strings from the network displayable by the mobile for user information (NI);[9]

2. **Application mode** USSD used in order to execute user given transparent transport of data between the network and the UE, (MI, NI), aimed for applications in the network and their peer applications in the UE. [9]
USSD using directory number is enabled by short codes containing asterisk (*), and terminating hash (#) along with digits (0 to 9) and associated with a nature of address indicator and number plan indicator. Asterisks are usually used to separate the codes into multiple requests. The in-between digits named as parameters, may have variable length and are separated by the asterisks. [3]

When the public MMI is used for the control of supplementary services, * and # should not be included with SC (Service Code, made of 2 or 3 digits which is the unique identifier of Supplementary Service either as predefined or an additional service code) or SI (Supplementary Information, made of variable digits which contains a PIN code or Directory Number) fields. In case that a specific service request does not need any SI, ”*SI” is not needed to be entered, such as *SC#SEND. In case that more supplementary information is needed, entry should be *SI, such as *SC*SIA*SIB*SIC#SEND. It is possible that SIB is utilized to denote the tele or bearer service. This is to express a Basic Service Group to which such supplementary service request is applied, in addition that SIC can be utilized to identify the value of the ”No Reply Condition Timer”. [3]

1. When other MMI is being used in order to control supplementary services, there should be not any limitation in the SC and SI fields. [3]

2. If public MMI is being used, there should be specific limitation over application of one and two digit directory numbers. Applying other MMI may be an option in order to eliminate such limitations. [3]

2.4 An introduction to Erlang

For development of this project, Erlang as a soft real-time, functional, single assignment, dynamic typing, general-purpose, and garbage-collected language was used. Such features are also accompanied with almost no mutable data structures. This leads to no locks and also easiness over parallelization. Overall Erlang also supports concurrency, distributedness, and fault-tolerance, message passing and lightweight processes. Also context switching is less costly in two or three times compared to the same in C programs. In addition, Erlang benefits from VM which gives the capability to make compiled code runnable on any other platform. Erlang systems contains modules which are made of functions and attributes.[12]

2.4.1 OTP

Erlang uses the OTP (Open Telecom Platform) as formalization of process design patterns. They are implemented in library modules that are provided with
the standard Erlang distribution. This is a way to package a system component as library modules, e.g. supervision tree, servers, event handlers, or finite state machines to do all of the generic process work and error handling. Such design principles give the ability to the user to build Erlang applications. The task of such components is to monitor workers and other subsupervisors (a hierarchical tree of processes used to program fault-tolerant systems). The specific code, written by the programmer, is placed in a separate module and called through a set of predefined callback functions.[12]

OTP behaviors include worker processes which do the actual processing, and supervisors whose task is to monitor workers and other supervisors. Worker behaviors, often denoted in diagrams as circles, include gen_servers, gen_fsm, and gen_events. Supervisors, denoted in illustrations as squares, monitor their children which can be either workers or other supervisors. This creates a supervision tree.[12]

Gen_servers are generic behaviour modules aimed for resource management operation. Gen_servers implement the server side of a client-server pattern and interact with any number of clients in the way that many distinct clients are supposed to share a common resource. Gen_servers include standard interface functions and the functionality of tracing and error reporting. They are also capable of being placed into an OTP supervision tree. All particular parts of them are placed in callback module which export a pre-defined set of functions.[12]

Gen_fsms are generic behaviour modules aimed for implementation of Finite State Machines with the relation of State(S) x Event(E) → Actions(A), State(S'). It means after occurring the event E when the state is S, the action A needs to be performed which results in the transition to the state S'. Gen_fsms similar to gen_servers include standard interface functions and the functionality of tracing and error reporting. They are also capable of being placed into an OTP supervision tree. All particular parts of them are placed in callback module which export a pre-defined set of functions.[12]

OTP component may have 6 divisions: (1) Basic applications: Basic Erlang/OTP functionality; (2) Compiler; (3) Kernel to run Erlang apps; (4) SASL (System Architecture Support Libraries); (5) stdlib (STandard LIBrary); and (6) Operations And Maintenance (OAM).[12]

Main data structures used in Erlang are lists, enclosed within braces and tuples enclosed within brackets. There are also records in Erlang as data types that
contain a specific number of elements which can be accessed and manipulated by name. As stated before, Erlang also creates processes which are executed concurrently but in separate memory space. The means of communication between processes are messages which are transferred asynchronously, and after transmission are usually placed in process’s mailbox. The address of communication is Process’ Identifier (PID) or its alias which is bound to that PID. There can be millions of processes in a virtual machine. Termination of processes happens either if there is no more code to execute or there is a run-time error if either non-normal exit signals are not trapped by the receiving process or there is a ‘kill’ message received.[12]

2.4.2 Mnesia

Mnesia is a Distributed Database Management System (DBMS) mainly aimed for telecommunications applications and also some other Erlang applications that give capabilities of continuous operation and soft real-time features. Mnesia tries to address all of the data management issues required for typical telecommunications systems and it has a number of features that are not normally found in traditional databases. Mnesia is designed to benefit from dynamic reconfiguration at runtime. It is also highly fault tolerant for supporting nonstop systems, which is also run in the identical address space as the application is. Also the possible impedance mismatch among data format utilized by the DBMS and the one utilized by the programming language is resolved.[13]
Chapter 3

Requirements

3.1 Functional Requirements

3.1.1 Mobile Initiated (MI) USSD

3.1.1.1 MMI-Mode

USSD may be used in order to send data by MS towards the network in which data is sent to nodes in the VPLMN and in the HPLMN. In such a way, the serving network is responsible for delivering the received message to its destination node. In case that VPLMN would not be able to route the message to the destination, message will be forwarded to the HPLMN. Once an MMI string is entered by user according to [3] evaluated as USSD by MS, the MS shall send the string to the network applying a proper operation [5]. An MS is able to begin a USSD request to the network at will. Not any prior provision of the service is required (only those provisionings belonging to services making use of USSD). Thus, the USSD operation is started by sending ProcessUnstructuredSS-Request(PUSSR) USSD message through MS. In all of such messages and their responses, there is USSD string along with alphabet indicator and language indicator. The alphabet indicator shall indicate the alphabet used in the operation. The MS is able to use an USSD operation either during a call or out of call. After invoking such operation, the network shall examine the alphabet indicator and take suitable action in accordance with following rules, but just in case that it does not recognize the mentioned string, it should send it to the HLR. [10] [9]

1. Handling at HLR

   • MAP transaction shall be established by HLR, incoming from VLR
towards USSDGW after reception of USSD operation which has service code of HPLMN.

- Either propagating received USSD message from VLR to USSDGW or relaying it towards local application is determined by HLR, based on IMSI or MSISDN or USSD service code.
- In case of release of the transaction between either MS and HLR or HLR and GW, the HLR shall release the other transaction.
- Application in HLR would receive operation if there is no HPLMN service code contained.

2. Actions at Network

- (Case a), Reserved for HPLMN use:
  This case happens if a serving network receives a USSD message from either visiting user which should be forwarded directly to HPLMN, or from home user which is handled locally or forwarded it to HLR based on the network preference. The string in the message includes 1 up to 3 digits from the set (*, #) along with 1X(Y), in which X is arbitrary number 0-4, and Y is arbitrary number 0-9, and with possible addition of "* along with arbitrary number of arbitrary characters", finishing with # SEND. [9]

- (Case b), Reserved for VPLMN use:
  This case happens if a serving network receives a USSD message from visiting user which should be handled by VPLMN based on its preference. The string in the message includes 1 up to 3 digits from the set (*, #) along with 1X(Y) in which X is arbitrary number 5-9, and Y is arbitrary number 0-9, and with possible addition of "* along with arbitrary number of arbitrary characters", finishing with # SEND. [9]

- (Case c), Reserved for HPLMN use:
  This case happens if a serving network receives a USSD message either from visiting user which should be forwarded directly to HPLMN, or from home user which is handled locally or forwarded it to HLR based on the network preference. The string in the message includes 7(Y) finishing with SEND, where Y is arbitrary number 0-9.[9]

- (Case d), All other formats:
  It is up to the visited network to check the USSD message, and if it is comprehensible, the VPLMN acts upon it, otherwise it is forwarded to the HLR, but even if it is not comprehensible by HLR, MS should be informed then.[9]
The termination of MI operation should be done by network through answering the request from the mobile by error signal or a text string which gives the operation result, in which the string contains the characters at hand in the chosen alphabet as defined by [1] that is also used by USSD scheme. In case that no indication is needed, the answer should be empty. [9]

The answer to the MI USSD operation shall have alphabet and language indicators in which set of values for these indicators is the network operator choice. [9]

3. Allocation of service codes in control of Supplementary Services

There are international agreements which have standardized mentioned service codes and except those assigned for PLMN use, the rest shall not to be used by PLMNs, and the message format should be 1 up to 3 digits from the set (*, #), along with NN(N) known as the service code, in which N is equal to 0-9, with possible addition of ”* and arbitrary number of arbitrary characters”, and finishing with # SEND. Solely, codes defined at [3] and those defined in cases (a) and (b) shall be utilized. All other values are already taken for other purposes. Same also applies if the message format is X(Y) SEND, in which X is equal to 0-6 or 8-9 and Y is equal to 0-9 in which the codes X(Y) are standardized. In addition, only codes identified in [3] subclause 4.5.5 as in MMI can be used. All other values are already taken for other purposes.[9]

3.1.1.2 Application mode

Communication among an MS application and Network application is supported by USSD through transparent transfer of binary data among the network and the MS in which USSD can be used by application either while a call happening or out of call.[9]

Application in HLR would receive operation if there is no HPLMN service code contained. In case that application mode USSD is not supported by network, the MI operation shall be rejected and also the MS needs not to avoided over any attempt for automatic return to phase 1 USSD or to MMI mode USSD if such incompatibility occurs. In case that the network was not capable of identifying the destination node, HLR should receive the message forwarded by network.[9]
• Processing the USSD request

When a network node (e.g. HLR) shall process a USSD request locally, it should be done by a suitable application. And the specifications (location, nature and contents) of such an application are dependant on service provider and network operator. In addition, it might also contain establishing or releasing signaling and/or speech channels; forwarding request to the other network node (either modified request or not); forwarding a different USSD request to the other network nodes; and/or requesting more information from the MS (single/multiple times). After completion of request handling, the network node shall answer the request and release the transaction.[10]

3.1.1.3 MI Operation details

• Handling of USSD request at HLR

In this case, a USSD request is invoked by MS through sending a REGISTER message which has a ProcessUnstructuredSS-Request invoke component towards the network. The network node which receives the sent data in the request to the application, is in charge of USSD operations and shall wait for the application response. The application and consequently network may either end the dialogue or request multiple times more information in order to perform the requested operation. However, when the application requests more information to process the request, the network shall begin an USSD request which uses itself the on-going transaction. Then MS shall return the user’s response in FACILITY message which includes a return result component. The network shall pass the received data in the response to the application. In case that MS was not capable of processing the received request from the network, it shall return an error indication through sending a FACILITY message consisting a return error component. Whenever the application ends the dialogue, the network shall clear the transaction through sending a RELEASE COMPLETE message which includes a return result component. MS also shall clear the transaction through sending a RELEASE COMPLETE message in case that the user requests. In case that the network is not capable of processing the received request from the MS, it shall clear the transaction through sending a RELEASE COMPLETE message which includes a return error component. Error values can be found in [5]. An HLR shall always process a USSD request locally. If it does not support the alphabet used in a USSD request, it shall inform MS and release
MS is not allowed to begin an USSD operation in parallel to any transaction which is a call independent supplementary service. There is merely one transaction for USSD operations permitted for each user at a time. Nevertheless, the MS is permitted to begin USSD operations in parallel to transactions which are call related. Then MS shall clear the transaction through sending a RELEASE COMPLETE in case the user requests.

### 3.1.2 Network Initiated (NI) USSD

USSD may be used in order to send data by network (USSDGW, HLR, VLR, MSC) towards MS if MS is registered within the network. Hence, the network may either send UnstructuredSS-Request(USSR) message which queries the MS about information, or a notification which does not ask any information in the response from MS. Such a message has operator information pertained to the user. In all of such messages and their responses, there is USSD string along with alphabet indicator and language indicator. In case that MS is unreachable by network, an error is returned to the originated node of the operation. Not any prior provision of the service is needed (but only those responses belonging to services making use of USSD).

#### 3.1.2.1 Application mode

The USSD mechanism provides the MS user and a PLMN operator defined application to communicate transparently in accordance with MS and also intermediate network nodes. Such a mechanism also provides development of PLMN specific supplementary services. Wherever there is multiple applications in a network node, routing is done by the USSD handler towards suitable application. The MMI for USSD is defined in [3] and [9]. Also the alphabet indicator and the data coding scheme are in [1].

If an application residing in the HLR is going to send a USSD request or notification towards an MS, it shall establish a transaction towards VLR where the user is currently registered and send the operation back to VLR. Thus a timer will start for application and it waits. Here, HLR is in charge of controlling the transaction, and thus usually release the transaction after receiving a response from VLR. It is also possible that HLR release the transaction before a response is received in case that it is necessary. In case that it is necessary for an application in HLR to send more operations to the same MS as a process of the same application, it is
possible to use the same transaction before completion of all operations. In case that any other transaction shall be used for any next operation, HLR shall release the first transaction before subsequent one starts. However, if the releasing happens by VLR, HLR shall tell the application and end the USSD operation[1],[10]

**Invoking from HLR**

- HLR may contain an application which initiate USSD operation to set up MAP transaction to MSC/VLR in order to send USSD request or USSD notification towards registered MS.

- In order to release transaction if any response is coming from MSC/VLR or even if before such a response comes, HLR may be in charge of action.

- In case that application residing in HLR wants to send more operations, identical transaction can be used by HLR up to the time that operations needed to be completed.

- Release of the first transaction can be done by HLR in case that a new transaction shall be started.

**3.1.2.2 NI Operation details**

After invoking NI operation by network, if the alphabet indicated by the network is not supported by MS, MS shall inform the network. Also the network shall explicitly tell to MS that an answer needed from the user. Thus, the entered next string shall be used as an answer to this string which is not interpreted based on normal MMI procedures. An MMI command also shall be provided in order to permit the user for termination of the dialogue with a null answer. If no answer is required, normal MMI procedures in the MS are continued to be applied. The MS shall include alphabet and language indicators in answer to the network (if any). Also applications that use USSD operations might be in either the HPLMN side or already roaming into VPLMN. The network applications using USSD operations may either use several USSD operations (possibly a combination of mobile and network initiated) plus a part of a dialogue with the user. The associative relation among distinct operations as compartments of a dialogue are only implemented locally in the network application and does not result in special operation mode in the MS. The network initiated request for an answer from the user and the corresponding answer is a single operation; Also it is the same for the action over calls in progress or placing new call as a part of service which are provided by applications. In addition, connection release used for an unstructured dialogue is usually the responsibility of the network and must be done based on demand of
application applying the USSD operations. It is also possible that user initiate a
connection release by an MMI procedure.[9]

1. USSR
   In this case, a USSD request is invoked by the network through sending a
   REGISTER message which has an UnstructuredSS-Request invoke compo-
   nent towards the MS. The MS shall respond the request through a FACILITY
   message which has the mobile subscriber’s response in a return result com-
   ponent. The network then shall pass the received data in the response to the
   application which handles USSD operations and shall wait for the response
   by application which either decides to continue or end the dialogue. If the
   application continues the dialogue, it shall begin another USSD operation
   by sending a FACILITY message. The operation shall be either a USSD
   request or notification. When the application ends the dialogue, the net-
   work shall clear the transaction through sending a RELEASE COMPLETE
   message. Then the MS would be able to clear the transaction by sending
   a RELEASE COMPLETE message if the user requests. In case that MS is
   not able to process the request received from the network, it shall return an
   error indication through sending a FACILITY message including a return er-
   ror component. In case that MS receives a USSD operation in parallel to any
   transaction which is a supplementary service but call independent, it shall
   answer with a return error component in a RELEASE COMPLETE message
   which includes the ”USSD-Busy” error as defined in [5]. This is to express
   the failure in handling a parallel USSD operation. However, the network
   is able to begin USSD operations in parallel to transactions which are call
   related. The MS shall clear the transaction through sending a RELEASE
   COMPLETE if the user requests. Also the second USSD operation shall be
   a USSD notification, and the network should use the on-going transaction in
   order to send more USSD operations.[11]

2. USSN
   In this case, a USSD notification is invoked by the network through sending
   a REGISTER message which has an UnstructuredSS-Notify invoke compo-
   nent towards the MS. The MS shall respond the request through a FACIL-
   ITY message which has an empty result component to the network. The
   application shall either continue or end the dialogue. If the application con-
   tinues the dialogue, it shall begin another USSD operation through sending
   a FACILITY message. The operation shall be either an USSD request or
   notification. If the application ends the dialogue, the network shall clear the
   transaction through sending a RELEASE COMPLETE message. The MS
   shall also clear the transaction through a RELEASE COMPLETE message
whenever the user requests. In case that MS is unable to process the received request from the network, it shall return an error indication through sending a FACILITY message which includes a return error component. Error values are defined in [5]. In case that the MS receives an USSD operation in parallel to any transaction which is a supplementary service but call independent, it shall answer with a return error component in a RELEASE COMPLETE message which includes the “USSD-Busy” error as defined in [5]. This is to express the failure in handling a parallel USSD operation. However, the network is capable of beginning USSD operations in parallel to call related transactions. MS shall clear the transaction through sending a RELEASE COMPLETE if the user requests. Also the second USSD operation shall be an USSD request, and the network should use the on-going transaction in order to send more USSD operations.[11]

- **Forwarding USSD operations**

  The VLR/MSC shall be ready to receive a USSD operation from the HLR/VLR. In case that user is called, the VLR/HLR is in charge of establishing a transaction with MSC/MS and forward the operation without modifications. Any more information exchange among HLR/VLR and MSC/MS shall be transparent to the VLR/MSC. When one transaction is released, the VLR/MSC shall release the other.[10]

- **Information stored in the HLR**

  The HLR does not need to store any information regarding how to the use of USSD, but it can be for services pertained to USSD applications.[10]

### 3.2 Non-functional Requirements

The system needs to be designed and perform in the way that every component is modular and independent and also loosely coupled to one another. This feature comes from the main characteristics of distributed systems embedded in Erlang. It should be maintainable and also easy to be scaled up. By adding extra core hardware, it should be expected that the efficiency and performance of the system increases either in single or multiple functioning systems. It is also expected that multiple USSD dialogs to be operated in the same time. Also the fast recovery of system is desired, since Erlang design patterns are used in which trees are utilized for restarting the system through recoverable data from the moment of crash point.
3.3 Behavioural Requirements

Figure 3.1: Mobile Initiated Request-single operation
Figure 3.2: Mobile Initiated Request-multiple operations
Figure 3.3: Network Initiated Request-single operation
Figure 3.4: Network Initiated Request–multiple operations
Figure 3.5: Network Initiated Notification-single operation
Figure 3.6: Network Initiated Notification-multiple operations
Chapter 4

Design

4.1 Software Design

USSD architecture
4.1.1 Application platform

This platform is majorly designed for call related applications, but also has components which are in charge of building the virtual switches, storing session, and controlling media gateway hosts. Application platform of MSC which is connected to MG host, is the subcomponent which contains: Session Store as an internal storage of the MUS module for storing sessions, Transit Module which routes ISUP messages transmitted between MSCs and keeps track of the CIC and PC pair for every connection; RTP termination Agent which manages voice circuit information connected to MUS and the MGC; Connection Service which uses soft-switch mechanism and provides a media context for each call service, MGC (H.248 client) which handles signaling and controlling media; and last but not least the Operations and Maintenance components which adds the media resources and maintains the media resources available. The application platform can be re-used by other GSM services. For subscriber data management storage purposes, there is also VLR and HLR along with their workers which both interact with each other and also with MUS subcomponent explained in below.

4.1.2 BSC

Base Station Controller implementation acts as a subset of functionalities of MSC, HLR, VLR, Media Gateway and also Abis and A interface. Nevertheless, it was only used as BSC-only mode and media relay, a standalone component. The communication of OpenBSC with the MUS application is performed by GSM over IP on the A-interface.

4.1.3 Hardware component

The hardware component is composed of two subcomponents, one is an IP.access nanoBTS -a 2G picocell- which is used to provide radio connection between the BSC and the MS, and also to encapsulate the Abis protocol into IP packets and media into RTP. The other one is MS, which is located in the coverage area of BTS.

4.1.4 HLR

Home Location Register is in charge of handling the basic MAP signaling on the C and D interfaces for location and call procedures which has an operator defined subscriber white-listing procedure as a mapping between an IMSI with an MSISDN. The MSC extracts subscriber data from the HLR in case a subscriber attaches and eliminates the data when the subscriber goes out of the MSC area.
Next section will explain more about entities with data set built by Erlang records that are used for this project. One HLR is needed in each PLMN in order to maintain a permanent large user specific identifications database with the least possible access time. There are subscriber identifications residing in such a database, and each subscriber is registered permanently only in one HLR which is a deterministic reference in order to identify the present location of user. Thus, HLR is involved in tasks which are independent from subscriber’s location.

4.1.5 MG

Media Gateway is connected to MG host in the same component which is responsible for converting digital media streams and deciphering the Megaco commands from MGC. It acts based on the instructions forced by the operator providing resources to MGC. Both MG and MGC are users of Megaco/H.248 protocol which is in charge of modifying the terminations features. The registration of MG happens by sending a Service Change command in case it discovers a Megaco user. It provides multimedia communications over multiple transport protocols such as Asynchronous Transfer Mode (ATM) and IP. Here, MG acts as the transcoder of the media bearers. Also RTP used as bearer for both terminations.

4.1.6 MSC

Virtual machine Mobile-service Switching Centre handles call set-up and subscriber management, and is the main routing entity in CN. It is in charge of A and E interface and acts based on the requested service by creating circuits and controlling signaling and media. It is logically adjacent to nanoBTS in OpenBSC. It has the capabilities of SMS service. It is able to make use of codecs retrieved from the TCP server which was enhanced for encoding and decoding of messages related to call service, and also IPA and SCCP codecs based on Osmocom project.

4.1.7 MUS

Mobile User Service framework is a terminal of all messages coming from and going to BSC. It contains MUS Core and other workers e.g. MUS-Con (MUS Controller), call and gateway workers. The former as a gen_server is decoder of IP packets from TCP server carrying the level-3(L3) message payload to Erlang message format using the codecs library. The server module which is in between external A and E interfaces, makes the decision for incoming messages whether they may belong to already created session or, a new session is required to be created for them. If the latter happens, the server module is in charge of instantiating a new suitable FSM worker and also giving initial values to internal
records equipped with state information. Meanwhile, session store module keeps and manages session information, holding isolation property of ACID. Initiated transmitter workers communicate with the help of MUS-Con. The MUS-Con as a gen_fsm is in charge of creation of worker processes for each request and also routing the messages received from the MUS Core towards a correct worker. It also captures PID of sessions after initiation which is saved with SLR, DLR, IMSI and TMSI, in order to either build session key, accessible for associated session, or forward messages to previous MUS-Con worker by checking session key through MUS. Service logic is decided by MUS Controller. That is, it checks the message and decides to which service it is required to forward the message. Mus-Con is responsible for encoding Erlang messages being transmitted to the BSC into the format of a binary IP packets, and then sending the packets to the TCP/IP Server and also releasing of the relevant SCCP connection after completion of session. The most important feature of MUS controller is that it can easily be extended in case of requiring a new service, such as SMS, and USSD, which was done for the thesis. Other workers are Mobile Originated (MO) call worker, Mobile Terminating (MT) call worker and gateway worker. There is also a TCP server that is not included in MUS but connected to MUS Core and is in charge of maintaining a TCP/IP connection to the BSC on A interface.

4.1.8 VLR

Visitor Location Register is a dynamic local subscriber database tied to one MSC, (i.e. it is responsible for a limited geographical area) which is used for core services such as call setup and location procedures for each MSC in order to store data about subscribers for a particular MSC area. The VLR used in the project is implemented as a gen_server handling the B and D interfaces for each associated subscriber saved in a DETS database in the front-end in which IMSI and TMSI are key designation and information in connection with location update and call forms the rest of key. A VLR is usually in connection with a single MSC. However, there could be one VLR for multiple MSCs.

4.2 Decomposition Description

4.2.1 USSD_HLR component

USSD HLR Component
4.2.1.1 Processes

Application

**USSD_HLR_app**
This process is used in order to create, monitor and terminate USSD_HLR component as a unit.

Database

**USSD_HLR_db**
This module is used in order to create, monitor and terminate table of record of process IDs and database schema of such for USSD_HLR component using Mnesia. Mnesia as a truly distributed DBMS which provides a set of customizable atomic named tables of records to handle transactions over stored data. USSD_HLR_DB is defined to have a schema as configuration system table for the database with disc Copies property which provides a list of Erlang nodes in which tables are created, moved, handled, modified, removed, or reconfigured both in RAM and on disc based on ACID properties. Every transaction on tables, in addition to being written into the RAM table, are also appended to a LOG file. Access and reconfiguration of tables can be done simultaneously.

Supervisors

**USSD_HLR_sup**
This supervisor is static and implemented in order to handle creating, monitoring and terminating USSD_HLR process, MS_init_workers_sup and NW_init_workers_sup processes. The supervisor is also in charge of maintaining their possible crash or exit (restarting the process again) in face of errors and exceptions which follows one_for_one restart strategy and permanent restart mode. It is the responsibility of this supervisor to restart its singleton instance of child processes as USSD_HLR, MS_init_workers_sup, and NW_init_workers_sup in case of their termination. In case of termination of the main supervisor (USSD_HLR_sup), all of its children are also terminated accordingly in a top-down approach.

**MS_init_workers_sup**
This supervisor process is dynamic and implemented in order to handle
creating, monitoring and terminating processes in charge of MS initiated messages (PUSSR) incoming to HLR originated from MS and propagated through MSC/VLR (and response); same also goes with posterior chain of messages as USSD requests and responses in case of multiple operations executed by network nodes (USSDGW, HLR). The supervisor is also in charge of their possible crash or exit (restarting the process again) in face of errors and exceptions which follows simple_one_for_one restart strategy and temporary restart mode. Under this supervisor, all the children will be started dynamically and incrementally through function call. Also any child process after termination is not restarted.

**NW_init_workers_sup**

This supervisor process is dynamic and implemented in order to handle creating, monitoring and terminating processes in charge of NW initiated messages (USSR or USSN) originated from HLR application or USSDGW and propagated through MSC/VLR towards MS. The supervisor is also in charge of their possible crash or exit (restarting the process again) in face of errors and exceptions which follows simple_one_for_one restart strategy and temporary restart mode. Under this supervisor, all the children will be started dynamically and incrementally through function call. Also any child process after termination is not restarted.

**Workers**

**USSD_HLR**

This is the kernel process as a gen_server which is responsible for maintaining list of worker processes and their possible crash or exit (restarting the process again) in face of errors and exceptions; routing messages based on routing table to the appropriate handling process(es’ chain); maintaining local or remote application request and responses; and also dealing with tcap server connections.

**USSD_app_adaptor**

This process as a gen_server is used to handle local or remote application invocations’ requests and responses. It has similar properties to USSD_HLR, but is working as a subset and sub-handler of that kernel process and is called by that.

**MS_init_USSD_HLR**

This process as a gen_fsm is implemented for taking charge of MS initiated messages (PUSSR) incoming to HLR originating from MS as relayed or propagated through MSC/VLR towards USSDGW (and re-
sponse); same also goes with posterior chain of messages as USSD requests and responses in case of multiple operations given by network nodes (USSDGW, HLR). The supervisor is also in charge of maintaining their possible crash or exit (restarting the process again) in face of errors and exceptions which follows simple_one_for_one restart strategy and temporary restart mode. Under this supervisor, all the children will be started dynamically and incrementally through function call. Also any child process after termination is not restarted. Based on routing table defined in USSD_HLR, the process is capable of routing messages in-node or locally, or send it to USSDGW. [4]

**NW_init_USSD_HLR**

This process as a gen_fsm is implemented for taking charge of NW initiated messages (USSR or USSN) incoming to HLR originated from USSDGW, or created by HLR application and relayed or propagated through MSC/VLR towards MS (and response); same also goes with posterior chain of messages as USSD requests and notifications and responses in case of multiple operations given by network nodes (USSDGW, HLR). The supervisor is also in charge of maintaining their possible crash or exit (restarting the process again) in face of errors and exceptions which follows simple_one_for_one restart strategy and temporary restart mode. Under this supervisor, all the children will be started dynamically and incrementally through function call. Also any child process after termination is not restarted. Based on routing table defined in USSD_HLR, the process is capable of routing messages in-node or locally, or send it to USSDGW. [4]
4.3 Communication Design

4.3.1 Communication with USSDGW and VLR

MAP dialogue handlers

- **ein_ss7_app** shall be invoked along with USSD_HLR_app in order to create communication unit **ein_ss7_ussd** between SS7/SIGTRAN and HLR unit. Supervision tree along with kernel process as tcap and other codecs and api modules are also invoked in similar procedure described above for HLR component.
• **ussd_map_in and ussd_map_out** both as gen_fsm workers are MAP parameters codecs and MAP message dispatchers/handler used in order to communicate with ussd_dialog. MAP_in is used for incoming invocation from either side, MS Initiated or Network Initiated. MAP_out is used to transfer the message out after being processed by USSD_HLR component either towards another network node in which HLR is used as in between node or from the HLR itself (as application invocation). Following section will explain more.

• **ussd_dialog** as a gen_fsm worker called by either tcap server unit, or USSD_HLR unit is both codecs module and dispatcher/handler of tcap level ussd messages used for mapping of between TCAP messages and MAP ones and vice versa. It is also used in order to either distinguish or rebuild such messages based on their incoming or outgoing flow. Following section will explain more.

• **tcap(_server)** as an abstraction of tcap unit and also exe file (which is get bounded to tcap gen_fsm also stack gui in order to make the communication with the stack) is used for communication with SS7/SIGTRAN. Similar to USSD_HLR module described above, tcap as a gen_server is also written as kernel process for handling processes used for tcap transactions either for incoming or outgoing flow for ussd dialog.

### 4.3.2 TCAP

TCAP and MAP are the foundations of signaling of all NSS interfaces. TCAP contains layer 4 to 6 of OSI. MAP comes over TCAP and NSS subsystems as HLR, VLR, MSC and EIR come over MAP. TCAP which utilizes SCCP is the most significant protocol of GSM stack providing the core functionality for roaming services. This layer gives the capability of a session-oriented transactional service to upper protocol layers, which is required in order to perform the functional operations that network elements such as HLR and MSC might require. TCAP provides users accessing databases and switching exchanges through the SS7 network and invoking services and modifying parameters. It also provides pure data transfer while inter-MSC handover is happening. It also gives the feasibility of integrating translation functionalities into a message, e.g. phase 1 and 2 of GSM networks. Conversation happens when the receiving party finds the information in the dialog control information or the dialog portion. There is also the component portion which is responsible for carrying the actual user data which is the MAP traffic itself. TCAP has been developed upon Association Control Service Element (ACSE) protocol which gives a connection-oriented mechanism to associate a specific function with a remote similar function by the utilization of an application context i.e.
associate, release, etc.; and Remote Operations Service Element (ROSE) protocol which gives Invoke-Result operations. TCAP is in charge of SSN based routing in order to identify to which upper layer the messages should be directed based on particular SSN, e.g. SSN 6 as a direction address for an HLR must be routed to the MAP protocol. This layer gives the possibility of computing the identical node with multiple functions based on various application sections in HLR. The TCAP layer gives an abstraction level for application sections that fully mask the SS7 routing and addressing problems. In general, TCAP provides (1) addressing by point code/SSN and/or GT and (2) subsystem and PC status change indications. TCAP utilizes the connectionless type of SCCP (PCs 0 and 1). Thus, the SCCP-UDT messages are used for transport of TCAP. The sender party of TCAP addresses the destination through the SCCP. The SCCP is capable of routing the messages through STPs.[14] [19]

In addition, the flexibility of TCAP gives the possibility of processing of all types of parameter types and data formats. TCAP needs to have the capability of processing length indicators starting from one up to thousands of bytes which requires adequate size of the TCAP that needs to be reserved for length indicators, distinguishing and preprocessing the data between multiple parameter types such as strings, integer numbers, real numbers and especially ASN.1 (Abstract Syntax Notation one), in order to give service to Layer 7 (MAP). [19]

Regarding coding of parameters and length of TCAP messages, since UDT message can have the maximum size of 255 bytes, every message begins with a TAG as an identifier of classification of following content or data part (including data classes defining the data type, data formats defining the primitive and constructor as collection of parameters such as IMSI which is made of MCC, MNC, and MSIN), after a length indicator of the content field. This is accompanied by data content which is the actual information, which may also comprised of several TAGs, length, and content fields. TAG value provides the recipient, the parameter type which data field needs to carry. There are five standard TCAP messages and only four of them are utilized by GSM.[19]

**TCAP Architecture:**

There are two distinct sections of TCAP layer, the component and the transaction layers based on ROSE and ACSE protocols:[14] [19]

1. **Transaction layer**

The transaction layer which is in charge of establishment and maintenance
of end-to-end communication as an association among sender information (a local application side) and addressee of the message (a remote one). This value is given to SCCP in order to perform addressing. There would be an extra TCAP-internal identifier added to the process which is transaction ID, similar to SLR and DLR in case of connection oriented SCCP. It utilizes BEGIN, CONTINUE, END, and ABORT TCAP messages (ASSOCIATE and RELEASE ACSE messages). Among the BEGIN and END (or ABORT), the transaction layer is in charge of data exchange among the two application sides through CONTINUE. There might be multiple CONTINUE frames among BEGIN and END. Each of them is connected to a unique transaction ID by utilizing an Invoke ID and linked ID. Transaction layer forms Dialogs in MAP layer as explained below. The dialog control along with other related information makes up optional dialog portion of a TCAP message.[14]

- The Dialog Unit
  This optional section in a TCAP message is utilized in order to provide the synchronization means of processing of the data for both end points of a connection, which is located in the component section. There are three different dialog units as parts of the dialog portion including the dialog request as proposal of a protocol, the dialog response as confirmation of request, and the dialog abort as termination of a dialog which could be or could not be pertained to the request. There is precisely one dialog unit available for each TCAP message. TCAP utilizes the application context name for the dialogs which are made of an identifier for the standard and also the utilized protocol for a transaction. Received string specifies which MAP module version was sent by sender. TCAP and MAP use the ASN.1 when coding data that is also identical for the dialog unit, which uses the values of object descriptor and external.[19]  

2. Component layer

The component layer in the OSI Layers 5 and 6 is in charge of synchronizing and coordinating of a communication which also gives the possibility of a uniform data interface to its users as application protocol data unit (APDU). Such unit is also mentioned as components -transporting means of the payload- that MAP and the component layer may utilize in order to exchange. This layer formats Invoke, Result, Error, Reject, Return-Result, and Cancel messages, that are in charge of carrying data and machine state invocations among the two application sides into transactional frames (BEGIN or CONTINUE). The Invoke ID is responsible of a specific Invoke operation
which needs a remote application to perform processing and transfer back the result with a Result, Error, etc.. This relies on the positive result of the operation. While the processing of this operation is happening, the remote entity might also initiate an Invoke on the originator entity. The component portion has the actual user data if any which is a MAP signaling information. Following is messages within such layer.[19][14]

- **Invoke Component**
  An application specific operation is started with Invoke component on the receiver side which is defined by operation code as MAP operations. These values identify the arrangement of subsequent parameter field. Each component has an invoke ID assigned only for an operation or a dialog. It may also have a linked ID, in order to provide relations among various dialogs. The value of the invoke ID in subsequent dialog is also included in linked ID that is linked to the first one. The MAP user data goes in between the optional parameter field of an invoke component.[19]

- **Return Result Component**
  This component is the result of an opened dialog over Invoke component that is transported in a Return Result and it might be segmented due to size of UDT message. The Return Result component is utilized for all data segments. The last result data segment is transmitted in a Return Result component.[19]

- **Return Error Component**
  The answer to Invoke component is Return Error component in case that there was not a possibility for the operation to be completed which could be protocol error, or user not responding to the paging process.[19]

- **Reject Component**
  If it is infeasible for an application to process a component due to errors, the Reject component will be sent to the sender.[19]

### 4.3.3 MAP

MAP is an interface or adapter which is logically placed over TCAP and under the application.[14] [4]

MAP messages are structured in this format [4]:
4.3.3.1 Communication between MAP and its users

In below, primitives between MAP and TCAP is mentioned. Since the application such as HLR is not considered as subset of the OSI, MAP services as primitives are needed for handling tasks and data propagation among various applications and MAP layer. The dialog among two TCAP users is of structured type. In this dialog, after delivery of data, a reaction, an acknowledgment, or a response is created, because a dialog among MAP applications needs to structured i.e., it should be initiated with a BEGIN message and if there would be no errors, it will be followed by CONtinue message sent by receiver towards sender which contains originating transaction identifier and a destination transaction identifier. Then it might be carried for multiple operations on the same transaction or terminate with an END message. It is also possible that in an special case abortion of a dialog happens that might be sent by TCAP or MAP.[14] [19]

MAP Services

MAP and a MAP application communicate through MAP services as primitives. There are two kinds of such services as common MAP services for communication control among MAP and application and also special MAP services as signaling bearer of signaling data. Both types will be more explained. There is a communication flow through MAP applicaiton, MAP, TCAP and the rest of layers downwards and then on the other peer, upwards and in the reverse order as mentioned. MAP both receives requests and responses from both MAP application sides and forwards it to the other side. Services and directions of such are needed to be distinct.[14] [19]

1. Common MAP services: There are six services that might be utilized in order to control a communication among MAP and its application. Based on type of the MAP service, whole four or merely a few of the primitives as Request, Indication, Response, and Confirmation are required. [19] [4]

   - MAP_DELIMETER
     This service is sent when application is to announce that a data packet is ready and it is available to be transferred to the peer element which
may have MAP-OPEN service as communication control, special MAP services, or both. There are Request and Indication types that can be defined as MAP-DELMITER service.[19] [4]

- **MAP_OPEN**
  This service is used when an application asks MAP to set up a dialog with another one. This is made of application context name (specification of the requested transaction) and also it specifies the sender and the receiver. None of parameters and data are placed in such primitive. And whole of primitives as Request, Indication, Response, and Confirmation are included for the MAP-OPEN service.[19] [4]

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Request</th>
<th>Indication</th>
<th>Response</th>
<th>Confirm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application context name</td>
<td>M</td>
<td>M(=)</td>
<td>U</td>
<td>C(=)</td>
</tr>
<tr>
<td>Destination address</td>
<td>M</td>
<td>M(=)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Destination reference</td>
<td>U</td>
<td>C(=)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Originating address</td>
<td>U</td>
<td>O</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Originating reference</td>
<td>U</td>
<td>C(=)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Specific information</td>
<td>U</td>
<td>C(=)</td>
<td>U</td>
<td>C(=)</td>
</tr>
<tr>
<td>Responding address</td>
<td>U</td>
<td>C(=)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Result</td>
<td></td>
<td>M</td>
<td>M(=)</td>
<td></td>
</tr>
<tr>
<td>Refuse-reason</td>
<td></td>
<td>C</td>
<td>C(=)</td>
<td></td>
</tr>
<tr>
<td>Provider error</td>
<td></td>
<td>O</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **MAP_CLOSE**
  This service is utilized in order to terminate current process which is transferred to the MAP and forwarded to TCAP when a MAP application is willing to terminate a dialog. There are Request and Indication types of this service.[19] [4]

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Request</th>
<th>Indication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release method</td>
<td>M</td>
<td></td>
</tr>
<tr>
<td>Specific information</td>
<td>U</td>
<td>C(=)</td>
</tr>
</tbody>
</table>

- **MAP_U_ABORT**
  This service as MAP User ABORT is used when an application is willing to interrupt a dialog as MAP user abort. There are Request and Indication types of it.[19] [4]
<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Request</th>
<th>Indication</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Reason</td>
<td>M</td>
<td>M(=)</td>
</tr>
<tr>
<td>Diagnostic Information</td>
<td>U</td>
<td>C(=)</td>
</tr>
<tr>
<td>Specific information</td>
<td>U</td>
<td>C(=)</td>
</tr>
</tbody>
</table>

- **MAP_P_ABORT**
  This service as MAP Provider ABORT used when TCAP is willing to interrupt or already interrupted the dialog as MAP service provider abort. There is Indication type for MAP-P-ABORT service.\[19\] \[4\]

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Indication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider Reason</td>
<td>M</td>
</tr>
<tr>
<td>Source</td>
<td>M</td>
</tr>
</tbody>
</table>

- **MAP_NOTICE**
  This service which has only Indication type is used when an application is willing to provide information regarding problems on the peer party. This service is expected for MAP application in case that it receives a TCAP message along with a reject component and particular problem codes. There could be various causes such as protocol errors or unexpected values and parameter variants.\[19\] \[4\]

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Indication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem diagnostic</td>
<td>M</td>
</tr>
</tbody>
</table>

2. **Special MAP services: USSD MAP services**

- Supplementary services\[4\]
  REGISTER
  FACILITY
  RELEASE COMPLETE
### MAP_PROCESS_UNSTRUCTURED_SS_REQUEST [4]

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Request</th>
<th>Indication</th>
<th>Response</th>
<th>Confirm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invoke Id</td>
<td>M</td>
<td>M(=)</td>
<td>M(=)</td>
<td>M(=)</td>
</tr>
<tr>
<td>USSD Data Coding Scheme</td>
<td>M</td>
<td>M(=)</td>
<td>C</td>
<td>C(=)</td>
</tr>
<tr>
<td>USSD String</td>
<td>M</td>
<td>M(=)</td>
<td>C</td>
<td>C(=)</td>
</tr>
<tr>
<td>MSISDN</td>
<td>C</td>
<td>C(=)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User error</td>
<td></td>
<td></td>
<td>C</td>
<td>C(=)</td>
</tr>
<tr>
<td>Provider error</td>
<td></td>
<td></td>
<td></td>
<td>O</td>
</tr>
</tbody>
</table>

### MAP_UNSTRUCTURED_SS_REQUEST [4]

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Request</th>
<th>Indication</th>
<th>Response</th>
<th>Confirm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invoke Id</td>
<td>M</td>
<td>M(=)</td>
<td>M(=)</td>
<td>M(=)</td>
</tr>
<tr>
<td>USSD Data Coding Scheme</td>
<td>M</td>
<td>M(=)</td>
<td>C</td>
<td>C(=)</td>
</tr>
<tr>
<td>USSD String</td>
<td>M</td>
<td>M(=)</td>
<td>C</td>
<td>C(=)</td>
</tr>
<tr>
<td>Alerting Pattern</td>
<td>C</td>
<td>C(=)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User error</td>
<td></td>
<td></td>
<td>C</td>
<td>C(=)</td>
</tr>
<tr>
<td>Provider error</td>
<td></td>
<td></td>
<td></td>
<td>O</td>
</tr>
</tbody>
</table>

### MAP_UNSTRUCTURED_SS_NOTIFY [4]

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Request</th>
<th>Indication</th>
<th>Response</th>
<th>Confirm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invoke Id</td>
<td>M</td>
<td>M(=)</td>
<td>M(=)</td>
<td>M(=)</td>
</tr>
<tr>
<td>USSD Data Coding Scheme</td>
<td>M</td>
<td>M(=)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>USSD String</td>
<td>M</td>
<td>M(=)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alerting Pattern</td>
<td>C</td>
<td>C(=)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User error</td>
<td></td>
<td></td>
<td>C</td>
<td>C(=)</td>
</tr>
<tr>
<td>Provider error</td>
<td></td>
<td></td>
<td></td>
<td>O</td>
</tr>
</tbody>
</table>
Chapter 5

Implementation

5.1 Plan

Since the thesis was supposed to be in cooperation with another student doing similar project, the whole of design and implementation needed to be coherent and compatible with one another, and in addition, previous code base projects along with new code base, and also new modules needed to be integrated as well. This mandated a bottom-up approach for implementation and also testing as one of the most appropriate methods to follow. It was decided to follow a few weeks for whole of the implementation which was divided by two weeks. Along with USSD modules residing inside HLR VM, the author needed to develop the MAP encode/decoder in and out for communication with SIGTRAN and also to test them in order to both make the propagation possible for incoming/outgoing USSD and related messages from/to VLR to/from USSDGW in which HLR acts in between node.

It was also required to first make virtual machines for each network node on the test-bed server up and running, and then doing the same for the SIGTRAN system. Also there was a need for tight resource sharing for the mere one nanoBTS. Both took a considerable amount of time in order to be used properly for testing purposes which added to timeline of thesis.

5.2 Process

Due to the nature of project and expected impediments, V-model was chosen since based on the thesis, evaluable as a small to medium sized project, the requirements were plainly defined and also ample technical resources at hand because of projects experience and also existing specifications. After each implementation phase,
the written modules were planned to be tested.

In the V-model, verification (and validation) model was followed in such a way that review and inspections of specs and proposed system design as sequential development life-cycle of execution of processes in the project along with software tests were used since very early stage after requirements were rectified, digitized and followed in parallel to development phases.

In the V-model, there are two crossing opposite-directed lines of development. The design development line is in the left which faces down and implementation and test development is on the right facing up. These two crossing lines of development also have interactive correspondence with each others’ peer for each phase of development from modules and submodules up to whole of the system similar to OSI layers in which there is protocol peering of one another in two different stacks and layers that only are serving adjacent upper layer. There are two phases: first one is a high level design which is about system architecture and design. The second one is called low-level design phase in which actual software component are designed for implementation. The actual logic for each of components are given along with diagram including all their methods and relations. The results of phases were SRS and SDD. This was also accompanied by component tests. Module design was mapped into code, and the same for each module starting from MS or Network Initiated along with required record definitions and databases and also higher levels as components i.e. supervisors and application were implemented.

This facilitated identification of possible errors and reduced potential next level defects later. Thus, there was a test plan developed in each stage using gen_servers with codecs up to integrated components and eventually manual test as both system and acceptance test using the testing environment illustrated below in which there was an installed network surveillance tool as Wireshark, and also the physical distinct equipment as User Equipment (UE, the attached Mobile phone with functional SIM card to the testing BTS) for either demonstrating or verifying the results along with sample input and expected output.
5.3 Testing environment

It was planned to perform unit testing for modules and manual testing for the entire system. Latter was done through calling MSC/VLR, HLR and USSDGW nodes contained in Virtual Machines and checking the well-functioning behaviour of the system as expected over SIGTRAN infrastructure. The former test was not done because of time limitations.
5.4 Problems

There were some impediments and problems regarding the plan which delayed the process as below:

- Although skeleton of most of modules were ready on time, but both 3GPP documents for implementing MAP decoder/encoder were not sufficient to follow and also Tieto SIGTRAN was not properly configured, thus testing and troubleshooting phases took longer than expected. Also there was a dependency over another ongoing project to be finished to test over whole of the integrated system for making the message communication possible. The minimum requirements was decided to follow.

- There was no proper guideline or documentation in order to follow HLR component, SS7/SIGTRAN and USSD gateway set-up and troubleshooting. Setting up and maintenance of the USSDGW VM for a proper and stable responsiveness was tedious task.

- Sharing resources as BTS and also MSC VM was extremely time consuming. There were some modifications happening without making the author aware of them, such as changing the SSNs and IPs of each node and also mal-configuration of BTS, even though the author was supposed to be sole admin of the server. This added a large amount of re-work. Also the dependency of the tests on BTS and the server was another difficulty since the only possible way to access them was to be on-site and to have all of them properly functioning and up and running.

5.5 Solutions

- It was decided to have a strict time-bound usage of resources.

- The author was obliged to re-configure both BTS and SS7/SIGTRAN for personal use a couple of times, and lock the admin access over MSC VM. Getting to know the problems and solutions of SS7/SIGTRAN took a while.
• Integrating USSD MSC/VLR took way longer than expected, thus the author had to focus on finishing the HLR USSD component with MS initiated in the first stage and then to continue to the network initiated one in the next phase.
Chapter 6

Conclusions and future work

6.1 Conclusions

Because of Erlang intrinsic features and also reliable test-bed environment, given results through MS display interface and Wireshark either by single or multiple operation(s) were repeatable and verifiable. By development of concurrent USSD MSC/VLR project, author was able to test the HLR modules integrated over them. Due to above features, scalability of system is also facilitated.

The Mobile Arts’ USSD Gateway was not capable of supporting network initiated ussd dialog in application level. Thus other scenarios were tested.

V-model with a lean towards incremental and iterative method was adapted (as modified water fall model), thus it would make it possible for both authors to develop and test the modules accordingly. This is since the latter method is modeled around a step-by-step continuous increase of addition of features concluding with the deployment of completed software at the end of each iterative development along with a possible cyclical upgrade.

Estimation of project life cycle was very challenging in the project, since there were both newly added codebases along with units and components that needed to be implemented from scratch and then tested and integrated to the former. Also hardware dependency over BTS and VMs installed in the server aimed for testing modules and components was another burden.

It was desired and planned that establishing the test-bed for SS7 network (SIGTRAN), USSD gateway, HLR, BCS applications and their configuration would be taken two to three weeks long at most, and also facilitated maintenance of them.
would be plausible but this was not followed, hence the development time took longer than expected. Thus, a backup plan needed to be considered beforehand. Also access time to resources was increased over time since they were also used by colleagues to meet their deadlines and often system was overloaded. Commuting to the testing place as MA resides in the other city than that the author lives, became time consuming problem after few months, which added time to the test part of project because of hardware dependency.

This thesis required a lot of detailed (re-)comprehension of several 3GPP documents, Erlang and GSM books and manuals, over implementation, configuration, re-configuration and testing from the beginning. It was a great experience to perform in a professional Erlang-based Telecom working environment as the MA AB follows.

6.2 Future work

Adding security features for TCAP/MAP layers (since in SCTP and M3UA layers, security issues are already supported by Tieto SIGTRAN stack) which was not listed in the requirements as non-functional requirements can be considered.

The project consisted of all possible fundamental functionalities of the USSD HLR, which can be extended to support whole of them.

Unit testing may be added to the system features, which was not included due to time limit.

Some parts were excluded from the beginning and were not included in the requirements specifications as below. Such parts can be integrated/developed later.

- HLR specific Operation, Administration and Operation support
- HLR specific Charging support
- HLR specific Product documentation
- Supplementary Service management (service activation, deactivation, registration, erasure, interrogation and invocation as defined by [7]).
Acknowledgement

I would like to thank Martin Kjellin -my thesis supervisor-, Mobile Arts AB - sponsor-, Dr. Ping Wu -reviewer-, Dr. Ivan Christoff -examiner-, and Dr. Olle Eriksson -thesis coordinator- for their thoughtful helps and considerations. Also I would like to express my sincerest gratitude towards my parents, Atakhan and Nahid, for their life long attentive support, wholehearted love and serene tolerance. In the end, I am much grateful of my friends, my teachers and all the human society who also provided me with the chances and necessities to accomplish my tasks throughout my living, especially this thesis which supposedly might be an insignificant contribution in return.
Chapter 7

Bibliography


Appendix A

Glossary

Abbreviations, and Acronyms

3GPP  The 3rd Generation Partnership Project
ACID  Atomicity, Consistency, Isolation, Durability
ACSE  Association Control Service Element
AI  Address Indicator
AP  Application Provider
API  Application Programming Interface
APDU  Application Protocol Data Unit
ARPU  Average Revenue per User
AS  Application Server
ASN.1  Abstract Syntax Notation One
ASP  Application Server Process
ATM  Asynchronous Transfer Mode
AUC  Authentication Center
BIB  Backward Indicator Bit
BS  Base Station
BS  Bearer Service
BSC  Base Station Controller
BSIC  Base Station Identification Code
BSN  Backward Sequence Number
BSS  Base Station Subsystem
BSSAP  Base Station System Application Part
BSSMAP  Base Station System Management Application sub-Part
BTS  Base Transceiver Station
CaPA  Calling-Party Address
CC  Call Control
CC  Connection confirm
CCH  Control CHannel
CdPA  Called-Party Address
CEPT  European Conference of Postal and Telecommunications Administrations
CN  Core Network
CR  Connection Request
CRC  Cyclic Redundancy Check
CS  Connection Service
DBMS  Database Management System
DETS  Disk-Based Erlang Term Storage
DLR  Destination Local Reference
DPC  Destination Point Code
DS-1  Digital Signal 1
DTAP  Direct Transfer Application Part
EDGE  Enhanced Data rates for GSM Evolution
EIR  Equipment Identity Register
ETSI  European Telecommunications Standard Institute
FCS  Frame CheckSum
FDD  Frequency Division Duplex
FDMA  Frequency Division Multiple Access
FIB  Forward Indicator Bit
FISU  Fill-In Signal Unit
FSN  Forward Sequence Number
G-MSC  Gateway Mobile-service Switching Centre
GMSC  Gateway Mobile-service Switching Centre
GSM  Global System for Mobile Communication
GPRS  General Packet Radio Service
GT  Global Title
GTT  Global Title Translation
HSCSD  High-Speed Circuit-Switched Data
HLR  Home Location Register
HPLMN  Home Public Land Mobile Network
IMEI  International Mobile Equipment Identity
IMSI  International Mobile Subscriber Identity
INAP  Intelligent Network Application Protocol
IP  Internet Protocol
IPA  IP.access
IPSP  IP Server Process
ISDN  Integrated Services Digital Network
**ISUP**  ISDN User Part

**IWF**  InterWorking Function

**LA**  Location Area

**LAI**  Location Area Identity

**LAPD**  Link Access Procedures, D channel

**L3**  Level-3

**LI**  Length Indication

**M2UA**  MTP 2 User Adaptation layer

**M3UA**  MTP 3 User Adaptation layer

**MA**  Mobile Arts

**MAP**  Mobile Application Part

**MCC**  Mobile Country Code

**ME**  Mobile Equipment

**MG**  Media Gateway

**MGC**  Media Gateway Controller

**MG Host, M-MGW**  Media Gateway Host, Mobile Media Gateway

**MI**  Mobile-Initiated

**MM**  Mobility Management

**MMI**  Man-Machine Interface

**MNC**  Mobile Country code

**MO**  Mobile Originated

**MOU**  Memorandum Of Understanding

**MP**  Message Priority

**MS**  Mobile Station
MSC Mobile-service Switching Centre
MSIN Mobile Subscription Identification Number
MSISDN Mobile Subscriber Integrated Services Digital Network
MSU Message Signal Unit
MT Mobile Terminating
MTP Message Transfer Part
MTU Maximum Transmission Unit
MUS Mobile User Service
MUS-Con Mobile User Service Controller
NI Network Indicator
NI Network-Initiated
NSS Network Switching Subsystem
OAM Operations And Maintenance
OMAP Open Multimedia Applications Platform
ONA Open Network Architecture
OPC Originating Point Code
OSI Open Systems Interconnection
OSMOCOM Open Source MOBILE COMMUNICATIONS
OTP Open Telecom Platform
PC Point Code
PC Protocol Class
PCM Pulse Code Modulation
PID Process Identifier
PLMN Public Land Mobile Network
**PSTN** Public Switched Telephone Network

**PUSSR** Process Unstructured Supplementary Service Request

**RAND** RANDomly generated Number

**RLC** ReLease Complete

**RLSD** ReLeaSeD

**ROSE** Remote Operations Service Element

**RTP** Real-time Transport Protocol

**SASL** System Architecture Support Libraries

**SC** Service Code

**SCCP** Signalling Connection Control Part

**SCH** Synchronization CHannel

**SCTP** Stream Control Transmission Protocol

**SEP** Signaling End Point

**SG** Signalling Gateway

**SGP** Signalling Gateway Process

**SI** Service Indicator

**SI** Supplementary Information

**SIF** Signaling Information Field

**SIGTRAN** SIGnaling TRANsport

**SIM** Subscriber Identity Module

**SIO** Service Information Octet

**SLR** Source Local Reference

**SLS** Signalling Link Selection

**SMS** Short Message Service
SMSC  Short Message Service Center
SMPP  Simple Messaging Peer-Peer
SP    Signalling Points
SPC   Signaling Point Code
SPMC  Signalling Point Management Cluster
SRES  Signed RESponse
SS    Supplementary Service
SS7   Signalling System 7
SSF   SubService Field
SSN   SubSystem number
SS7oIP SS7-based- over-IP
STDLIB STandarD LIBrary
STP   Signalling Transfer Points
SU    Signal Unit
SUA   SCCP User Adaptation Layer
TA    Timing Advance
TAF   Terminal Adaptation Function
TCAP  Transaction Capabilities Application Part
TCH   Traffic Channel
TDD   Time Division Duplex
TDMA  Time Division Multiple Access
TE    Terminal Equipment
TMSI  Temporay Mobile Subscriber Identity
T-MSC Terminating Mobile-service Switching Centre
**TMSC** Terminating Mobile-service Switching Centre

**TS** Time Slot

**TUA** TCAP User Adaptation layer

**TUP** Telephony User part

**UA** User Adaptation

**UE** User Equipment

**UDT** Unit DaTa

**UMTS** Universal Mobile Telecommunications System

**UP** User Point

**USSD** Unstructured Supplementary Service Data

**USSDGW** USSD GateWay

**USSD-GW** USSD GateWay

**USSN** Unstructured Supplementary Service Notify

**USSR** Unstructured Supplementary Service Request

**VAS** Value-Added services

**V-model** Verification and Validation model

**VMSC** Visiting Mobile-service Switching Centre

**V-MSC** Visiting Mobile-service Switching Centre

**VLR** Visitor Location Register

**VPLMN** Visiting Public Land Mobile Network

**WAP** Wireless Application Protocol