SIP-DSS1 Signalling Gateway

Boštjan Vlaovič, Simon Nedok, Peter Vicman, Zmago Brezočnik
University of Maribor
Faculty of Electrical Engineering and Computer Science
Smetanova ulica 17, Maribor, Slovenia

Abstract

New applications are driving the convergence of traditional circuit-switched telecommunication networks and packet based networks. As long as full transition to packet based operation will not be possible, interoperability between the two networks will be essential. In this paper we focus on functional and design specification of signalling gateway between Session Initiation Protocol (SIP) and V5.2 Interface using Digital subscriber Signalling System No.1 (DSS1). The signalling gateway is designed to be a part of Iskratel's SI2000 V5 product family. Its main objective will be to enable users of SIP User Agents – local or remote – to use the switch node's functionality in the same manner as the ISDN (Integrated Services Digital Network) users do.

1 Introduction

In the year 1974 the first packet based call was made on Arpanet between the Informational Sciences Institute (University of Southern California) and Lincoln's Laboratory (Massachusetts Institute of Technology). Since then, especially in the last eight years, many new architectures and protocols have been proposed. The most prominent and widely used is ITU's H.323 protocol suite. Its first version was published by ITU (International Telecommunications Union) in 1996. At the same time Internet Engineering Task Force (IETF) was developing Simple Conference Invitation Protocol. In 1999 IETF's working group *mmusic* (Multiparty Multimedia Session Control) published the first version of RFC 2543 — Session Initiation Protocol (SIP). Since then nine corrections took place and many products are available from commercial, academic, and GNU communities. Currently H.323 and SIP compete for the dominance at Voice over IP (VoIP) market and their role in the Next Generation Networks (NGNs).

New applications are driving the convergence of traditional circuit-switched telecommunication networks and packet based networks. As long as full transition to packet based operation will not be possible, interoperability between the two networks will be essential. In this paper we focus on functional and design specification of signalling gateway between SIP and Digital subscriber Signalling System No.1 (DSS1).

The signalling gateway is designed to be a part of Iskratel's SI2000 V5 product family and its main objective will be to enable users of SIP User Agents – local or remote – to use the switch node's functionality in the same manner as the ISDN (Integrated Services Digital Network) users do. We are not intending to prepare switching products to be included in SIP based network, but to extend its subscriber base with SIP users.

In Section 2 we take a quick look at digital switching system SI2000 V5. It gives us enough understanding about the basic concepts and building blocks of the hardware and software. Next, in Sections 3 and 4 an overview of protocols used by ISDN and SIP subscriber is given. It is followed by introduction to basic properties of Specification and Description Language (SDL). In Section 6 we propose an architecture for SIP-DSS1 signalling gateway and provide protocol mapping function example. In the conclusion we give some final remarks.

2 Digital switching system SI2000 V5

IskraTEL is the company for development, marketing, planning, manufacturing, installing, and servicing of telecommunication systems. It was formed in 1989 with the capital from Slovene investors and the German company Siemens and is the largest Slovenian telecommunications company. The current generation of its digital switching system is called SI2000 V5. It is a digital switching system with a few hundred to several thousands of ports. It is an advanced modular system that offers basic functionality as well as a rich range of services including PSTN, ISDN, signalling No.7, H.323, Centrex, IP Centrex, etc. The signalling gateways are located in Access Nodes (ANs) of the telecommunication network (Fig. 1). AN's task is to present IP terminals as basic rate ISDN subscribers to the Switch Node (SN). Supplementary service provision and switching is performed by SN. The MLC (Line Mod-

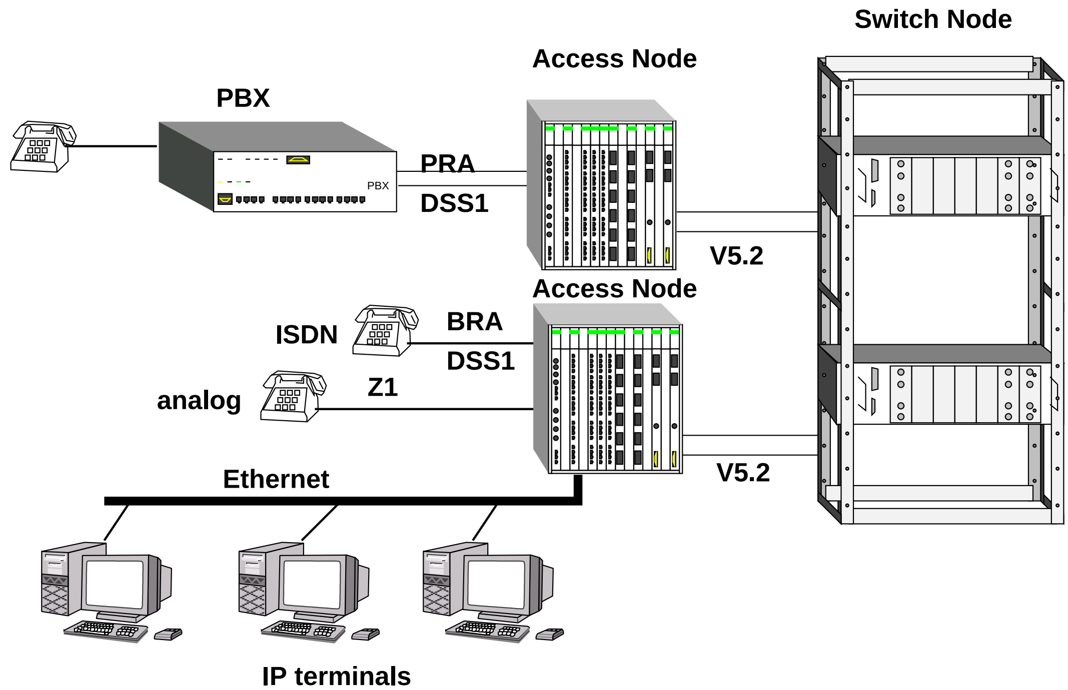

Figure 1: Network architecture.

ule Version C) is the core of the AN. The task of MLC is to connect analog subscribers, ISDN terminals, H.323 terminals, and network transmission paths. Sub-rack configuration of MLC consists of mechanical parts, a back panel and plug-in units. Central module incorporates hard disk or flash, PCI (Peripheral Component Interconnect) bus, Ethernet bus, four TPx (Symmetrical Primary Rate Access Interface) units, two PCM (PCI Mezzanine Card) slots and other units. TPx unit provides connection to the digital network (ISDN primary rate access (PRA), V5.2i interfaces).

Peripheral units are inserted in the MLC sub-rack starting from the left to the right after the central unit. In the module 20 slots are available for the peripheral plug-in units. They are connected in a form of a star, thus minimizing the mutual influence among the plug-in units and allowing plug-in unit replacement under voltage.

The system connects to the IP network via Ethernet adapter. IP subscribers can use PC application software or IP phones. Support for H.323 is already implemented and has successfully substituted ISDN telephones inside one of the company's branch offices. In this paper we will propose the solution for the additional support of the SIP terminals. The main objective is to provide the same user experience for the basic call and execution of the supplementary services as for the ISDN subscriber. Our intention is not to provide SIP-based services. AN has to provide signalling and media gateway. We will focus on signalling gateway only.

The core of the modern switching system is its software code. It provides the functionality, control, and management of the system. Software is divided in system and application part. Most of the application is written in SDL. Various drivers for the peripheral devices and smaller parts of the application are written in programming languages C and C++. In the next two sections a brief introduction to the differences between ISDN and SIP subscriber will be given.

3 ISDN user

ISDN user connects to the AN of the telecommunication network with his/her terminal equipment. Protocol layers used by the user terminal equipment and the network are illustrated in Figure 2. For call control DSS1 signalling is used. DSS1 is defined by ITU-T

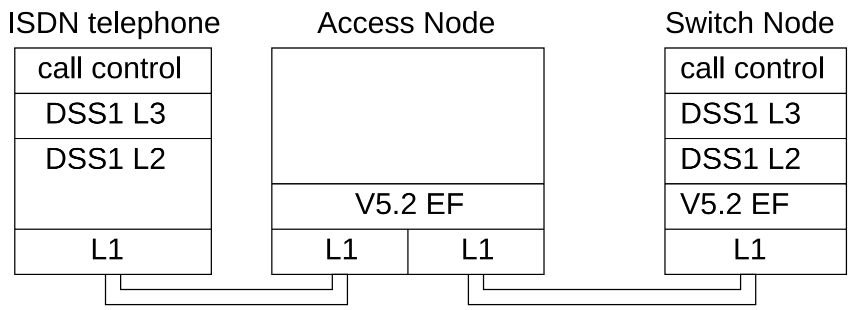

Figure 2: Protocol layers of the communicating parties for ISDN subscriber.

recommendation Q.931 [1] and is used for establishing, maintaining, and clearing of network connections at the ISDN user-network interface. These procedures are defined in terms of messages exchanged over the D-channel of basic and primary rate interface. An example of a successful call establishment consists of the following message flow: "SETUP", "CALL PROCEEDING", "ALERTING", "CONNECT", and "CONNECT ACKNOWLEDGE" (Fig. 7). DSS1 message is encoded into the bytes following the Q.931 recommendation.

ISDN subscriber signalling is terminating at SN. AN uses V5.2 envelope function to forward the L2 DSS1 messages. Call control, switching, and provision of supplementary services is provided by SN.

4 SIP user

SIP was developed under the IETF (Internet Engineering Task Force). It is an application-layer signalling protocol used for creating, modifying, and terminating multimedia sessions [2]. It can be used with UDP (User Datagram Protocol) or TCP (Transmission Control Protocol). SIP follows the Internet's client/server architecture. The main logical entity is user agent, which is an end-point for communication. It acts as user agent client and user agent server for the duration of the call. SIP is request-response protocol. User agent clients send requests and user agent servers respond to that requests based on the current state of the call and user preferences. Requests include method, which defines the nature of the request and the address to which the request should be send. The response includes status code, which defines the type of the response (success or failure).

Protocol layers used by the SIP terminal equipment and the telecommunication network are illustrated in Figure 3.

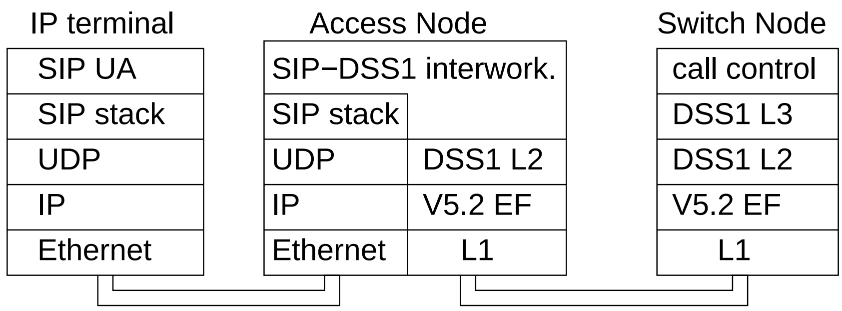

Figure 3: Protocol layers of the communicating parties for the SIP subscriber.

A successful SIP invitation consists of two requests, “INVITE” followed by “ACK”. The “INVITE” request invites the callee to join a session. After the callee has agreed to participate in the session with a “200 OK” response, the caller confirms reception by sending an “ACK” request. The “INVITE” request typically contains a session description usually written in SDP (Session Description Protocol). Description provides information about media formats that the caller is willing to use and where it wishes the media to be sent. If the callee wishes to accept the call, he/she responds to the invitation by returning a similar description listing of the proposed media types.

It is evident from the Figure 3 that some additional protocol layers have to be developed for the AN’s native support of SIP subscribers. Section 6 will discuss selected solution for the missing components. First, basic properties of Specification and Description Language will be given for better understanding of the proposed software architecture.

5 Specification and Description Language

SDL was developed by the switching systems industry and was first standardized by CCITT (Comité Consultatif International Télégraphique et Téléphonique) in 1976 [3]. It is based on finite state automata, but it uses graphical representation of flowcharts to show allowed transitions. In the development cycle, SDL is employed for the formal specification and design of the system. SDL supports the specification and description of structural and behavioural aspects of the application under development. It is a formal description technique (FDT). Its dynamic semantic is formally defined by the combination of Meta-IV and CSP (Communicating Sequential Processes) [4].

SDL may serve a number of purposes from reasoning about systems at an abstract level to the automatic derivation of implementations. Most of the SI 2000 V5 application code is written with SDL. Today several commercial and academic tools supporting the development of the systems with SDL are available. Tool support comprises graphical editing, validation, verification, simulation, animation, code generation, and testing. We are using Telelogic’s ObjectGeode v4.0 for the academic work and Geodedit v.2.2.4 for the work on the SI2000 product family.

At the highest level of the SDL hierarchical specification is *system* object (Fig. 4). System is the entry point to the SDL specification. It comprises a set of *blocks* and *channels*. Blocks are connected to each other and with the environment by channels. The hierarchy in SDL is static and has only very little influence on the dynamic behaviour of the system.

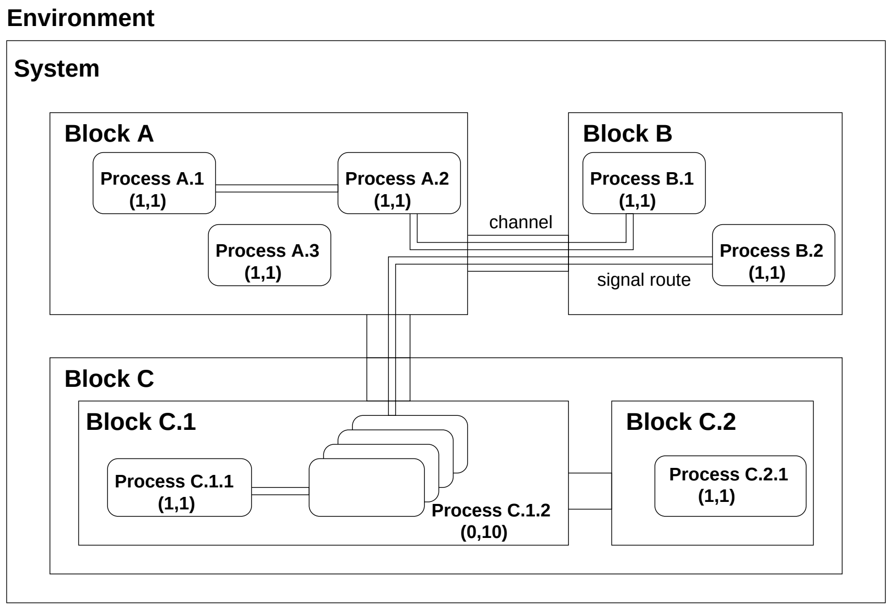

Figure 4: Top level description of SDL system.

Block is described by sub-blocks or set of processes. Blocks are static entities — they are created during the initialization of the system. Communication between blocks is only possible along the defined channels. Channels are asynchronous and can be unidirectional or bidirectional. Blocks are finally refined by processes.

A process is defined by a process graph that usually consists of several pages of state transitions. Each branch of the graph represents a possible execution of the process. Processes describe system's behaviour. Communication between different processes and between processes and the block interface is done via signal routes.

Signal routes are non-delaying. In the case the signal is sent to a process within a different block, it travels along the signal route in the same block, the channels connecting the blocks, and finally the signal route defined at the block where the receiving process in located (Fig. 4). Signals are the primary communication mechanism in the SDL. Each process has an unbounded queue at the input port. The port allows received signals to be queued until they are consumed or discarded by the process instance.

Process instances can be created either during the initialization of the system or dynamically during the execution of the system. Within the process declaration a dynamic range of allowed number of instances can be set in parenthesis (Fig. 4). Each process instance in an SDL system represents an independent asynchronously executing CEFSM (Communicating Extended Final State Machine). A process instance may be executed as soon as one of its trigger conditions holds. The transitions of a single process instance are executed sequentially. Interleaving of different actions concurrently executed by different processes are eliminated in SI2000 V5 system by implementation of atomic transitions — transition of a process instance can not be interrupted.

6 SIP-DSS1 Signalling Gateway

Due to the limited space we will not provide full description of the mapping function. Only the basic architecture of SIP signalling gateway will be provided and mapping of "INVITE" request will be given. Due to the nature of IP networks special care has to be given to the user authentication and authorization for the access to the network services. Basic requirements for Signalling Gateway (SG) are:

- user authentication/authorization,
- SIP to ISDN user mapping,
- SIP-DSS1 mapping,
- DSS1 message creation (encoded),
- SIP message creation (text based).

After user authorization SIP to ISDN user mapping should be performed. Each SIP user is assigned an E.164 telephone number and other ISDN like properties that will be considered during the call. User mapping should be managed by the management part of the network, so user's configuration can be changed during the system uptime. Based on individual user properties, mapping from SIP to DSS1 and vice versa is performed. Due to the different procedures for call establishment and network properties (circuit-switched versus packet based interface) SG has to provide appropriate procedures for successful SIP or DSS1 message creation and transmission. Proposed software hierarchy (Fig. 5) of the SIP-DSS1 signalling gateway consists of:

- SIP-DSS1 signalling gateway (mapping function),
- SIP/SDP stack (transaction layer, parser, constructor),
- UDP client/server,
- IP/Ethernet stack.

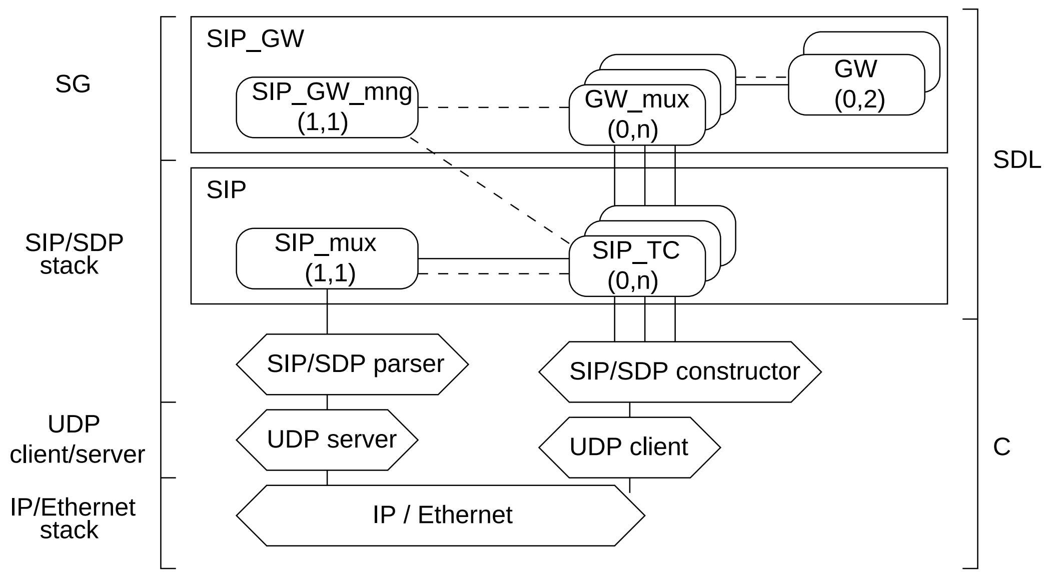

Figure 5: SIP-DSS1 signalling gateway and protocol layers on IP interface.

IP stack and Ethernet drivers are available as a part of the operating system. The V5.2 interface transports ISDN L2 signalling. This layer is not needed at AN for the ISDN subscriber (Fig. 2). It had to be provided if we wanted to present IP subscriber as an ISDN user to the SN (Fig. 3). Parts of the code have been developed during the development of the

H.323-DSS1 signalling gateway by Iskratel's research team. That includes L2 layer of the DSS1 protocol and corrections of the V5.2 interface.

For the proper operation SIP protocol stack has to be developed. It consists of UDP server/client, SIP/SDP parser, SIP/SDP constructor, and SIP transaction layer. During the packet network overload or link problems transaction layer provides message retransmission procedures and elimination of duplicated messages. This functionality is needed due to the use of unreliable transport protocol.

SIP stack is going to be developed partly with parser generation tools (Bison and Flex) resulting in C code and partly with SDL. Structure of the parser is presented in Figure 6. Parser interacts with upper SDL layers with the ADT (Abstract Data Type) operators implementing call back functions that result in SDL signals with parsed message. Construction of the SIP/SDP message is going to be provided through the function calls.

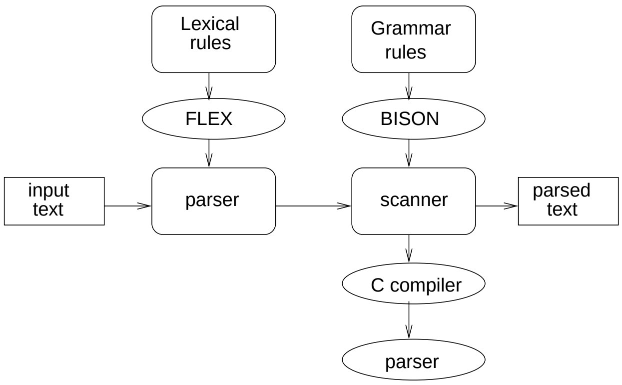

Figure 6: Structure of the parser developed with BISON and FLEX.

SDL block SIP is composed of two processes. Process SIP_mux is involved with all the communication with the parser. It additionally provides authentication and authorization. Its static nature assures constant presence during the uptime of the system. During the establishment of a new call SIP_mux creates one instance of process SIP_TC for each SIP user. Process SIP_TC describes behaviour of the SIP transaction layer and communicates with constructor.

Block SIP_GW has three different process descriptions. SIP_GW_mng is a static process. It manages instances of the process GW_mux. Each SIP user has one GW_mux and one DSS1_L2 process (not shown in the figure 5). If new call is initiated from the IP interface, a request for GW_mux creation is received from SIP_TC after its creation. In the case of ISDN call initiation, request for GW_mux process creation is received from the peer DSS1_L2 process. Since we are following ISDN subscriber requirements, we will not provide more than two active calls to the SIP user. Each call has its own GW process. Next, an example of SIP "INVITE" message mapping to DSS1 "SETUP" message is presented.

6.1 SIP-DSS1 Mapping Function Example

An example of a message flow is shown in Figure 7. A SIP user is calling an ISDN user. When new message arrives at the IP interface UDP server forwards received data to the SIP/SDP parser. Parser calls appropriate call-back function and forwards parsed data to the process SIP_mux. Before we proceed with the description of the execution, let us briefly describe an example of the "INVITE" message:

```
INVITE sip:2207269@gw.iskratel.si;user=phone  SIP/2.0
Via: SIP/2.0/UDP matrix.uni-mb.si:5060
```

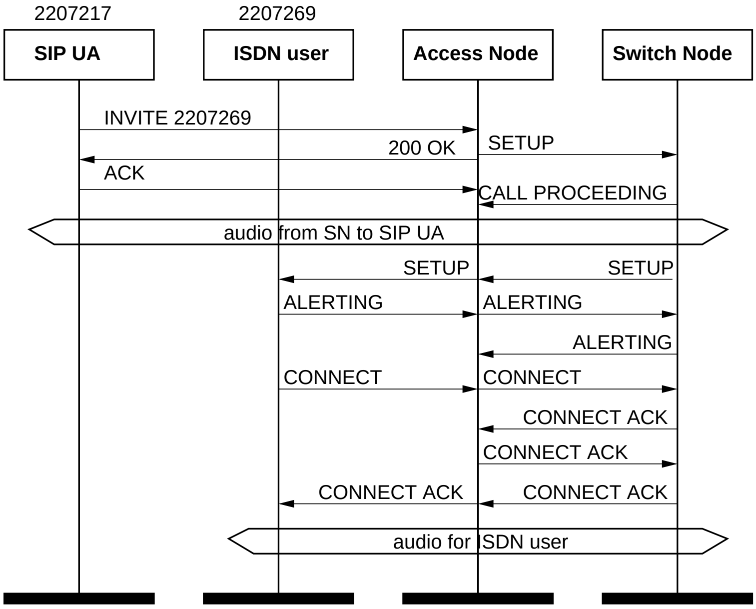

Figure 7: Call setup between SIP UA and ISDN subscriber.

```
From: Bostjan Vlaovic <sip:bostjan.vlaovic@uni-mb.si>
To: Simon Nedok <sip:2207269@gw.iskratel.si;user=phone>
Call-ID: 12345600@matrix.uni-mb.si
CSeq: 1 INVITE
Contact: Bostjan Vlaovic <sip:bvla@matrix.uni-mb.si>
Content-Type: application/sdp
Content-Length: 140

v=0
o=Bostjan Vlaovic 2890844526 2890844526 IN IP4 matrix.uni-mb.si
s=Session SDP
c=IN IP4 matrix.uni-mb.si
t=0 0
m=audio 49172 RTP/AVP 8
a=rtpmap:0 PCMA/8000
map:0 PCMA/8000
```

The first part of the message is made of SIP header fields with the call control information. From the header names of the SIP message one can reason about its contents. “From” header field includes information about the caller. Header value shows that user “bvla” is calling from the IP terminal “matrix.uni-mb.si”. IP terminal can be properly equipped personal computer or IP phone. The address of the callee is found in the “To” header field. In the example user “2207269” at “gw.iskratel.si” is being invited to the SIP session. Any valid SIP address could be used in the “To” header. “Contact” header field value contains the URI (Universal Resource Identifier) at which the UA would like to receive subsequent requests. Procedures for SIP user/virtual ISDN subscriber mapping has to be provided. In the example shown in figure 7 call to “2207217” would be mapped to sip URL “sip:bostjan.vlaovic@uni-mb.si”, which would be subsequently resolved to “sip:bvla@matrix.uni-mb.si” by SIP procedures which are outside of the scope of SIP-DSS1 signalling gateway.

After the blank line, SDP description of the media follows. The most important line in SDP message describes proposed codec and port information (G.711 A-law on port 49172). Detailed description of SIP and SDP messages can be found in [2] and [5], respectively. RTP/AVP map is provided in [6].

SIP_mux proceeds with the examination of the table of active users. If “INVITE” represents initiation of the first call from the SIP user, request for a new instance of SIP_TC is issued by SIP_mux. After successful initiation of SIP_TC a request for creation of GW_mux is issued to SIP_GW_mng. GW_mux creates new GW process for every call associated with the SIP user/virtual ISDN subscriber. Process GW receives parsed SIP message in the form of SDL signal, performs mapping function, creates appropriate DSS1 message and forwards it through the SIP_mux to the DSS1_L2 process.

To enable transfer of the original ring back tone or possible error tones from the SN SIP session is established immediately (Fig. 7). From the “INVITE” header fields and user configuration data DSS1 “SETUP” message is formed (Tab. 1). DSS1 messages “ALERTING”

Table 1: DSS1 SETUP message.

8 7 6 5 4 3 2 1	Bits
0 0 0 0 1 0 0 0	Q.931 Protocol discriminator
0 0 0 0 0 0 0 1	Length of call reference value
0 0 0 0 0 0 0 1	Call reference value
0 0 0 0 0 1 0 1	Message type (SETUP)
0 0 0 0 0 1 0 0	Bearer capability identifier
0 0 0 0 0 0 1 1	Length of the bearer capability contents
1 0 0 0 0 0 0 0	Coding standard (ITU-T) and information transfer capability (speech)
1 0 0 1 0 0 0 0	Transfer mode (circuit mode) and information transfer rate (64 kbit/s)
0 0 1 0 0 0 1 1	Layer 1 identifier and user information layer 1 protocol (G.711 A-law)
0 1 1 0 1 1 0 0	Calling party number information element identifier
0 0 0 0 1 0 0 1	Length of calling party number contents
0 0 1 0 0 0 0 1	Type of numbers (national number) and numbering plan identification (ISDN numbering plan E.164)
1 0 0 0 0 0 0 1	Presentation indicator (allowed) and screening indicator (user-provided, verified and passed)
0 0 1 1 0 0 1 0	2
0 0 1 1 0 0 1 0	2
0 0 1 1 0 0 0 0	0
0 0 1 1 0 1 1 1	7
0 0 1 1 0 0 1 0	2
0 0 1 1 0 0 0 1	1
0 0 1 1 0 1 1 1	7
0 1 1 1 0 0 0 0	Called party number information element identifier
0 0 0 0 1 0 0 0	Length of called party number contents
1 0 1 0 0 0 0 1	Type of numbers and numbering plan
0 0 1 1 0 0 1 0	2
0 0 1 1 0 0 1 0	2
0 0 1 1 0 0 0 0	0
0 0 1 1 0 1 1 1	7
0 0 1 1 0 0 1 0	2
0 0 1 1 0 1 1 0	6
0 0 1 1 1 0 0 1	9

and “CONNECT ACKNOWLEDGE” are ignored by GW, since SIP session is already active. User is receiving ring back tone through the media channel. When called party (ISDN subscriber) accepts the call, SN connects both media paths and call goes to the active state.

That concludes our example.

7 Conclusion

Introduction to SIP-DSS1 signalling gateway showed that functional specification can not be trivially defined. There are many options to consider. While ISDN terminology follows subscriber number which is mapped to physical interface, SIP's subscriber "number" is acctualy user's personal address. We can choose to build ISDN-like SIP network, where IP terminals would represent SIP users with predefined subscriber number/SIP address pairs. That would enable "user" authentication through IP terminal's MAC (Media Access Control) address and would greatly simplify the user location. On the other hand, SIP's native user mobility would be lost. Security is of major concern in traditional telecommunication networks. Network equipment should always doubtlessly know if subscriber (terminal) is authorized for requested services. If we map users to physical terminals, authorization becomes trivial.

Implementation of suplementary services in ISDN and SIP networks differ. We have decided to follow ISDN model. For successful suplementary service activation additional support of INFO method will have to be supported by UA. SIP recommendations are still under active development, so we expect our implementation to change during the development.

Convergence of traditional telecommunication networks and IP based networks is in the eyes of the beholder. One can adopt existing circuit switching equipment to interact natively with SIP network elements. In most cases existing services would have to be reimplemented due to the differences in service executions. Development, verification, and field testing takes time and a lot of resources. Our approach proposes provision of existing services to IP users as the first step of network transition to full packet based operation.

8 Acknowledgements

The authors acknowledge the support of the IskraTEL in the funding of the work reported in this paper. We would like to thank the Telelogic for providing ObjecGeode for our research and educational activities.

References

[1] ITU-T. *ISDN user-network interface layer 3 specification for basic call control*. ITU, 1999.

[2] Jonathan Rosenberg et al. *SIP: Session Initiation Protocol*. IETF Standards Track RFC 2543, 1999.

[3] ITU-T. *CCITT specification and description language (SDL)*. ITU, 1993.

[4] Andreas Mitschele Thiel. *Systems Engineering with SDL*. John Wiley & Sons, Ltd, 2001.

[5] Mark Handley et al. *SDP: Session Description Protocol*. IETF Standards Track RFC 2327, 1998.

[6] Henning Schulzrinne. *RTP Profile for Audio and Video Conferences with Minimal Control*. IETF Standards Track RFC 1890, 1996.