An End-System-Oriented Approach to Implementing Telephony Supplementary Services

Marcel Graf, Hong Linh Truong, and Johnny W. Wong*

IBM Research Division
Zurich Research Laboratory
8803 Rueschlikon / Switzerland

(*) on sabbatical leave from the University of Waterloo, Canada

Abstract

Traditionally, telephony supplementary services are implemented using a centralized approach for which a public switch or a private branch exchange (PBX) plays a crucial role in the implementation. With advancements in personal computer, networking, and digital signal processing technologies, an alternative approach is now possible. This approach exploits the power of the workstation (or end system) and allows the implementation of basic telephony services (i.e., call establishment and termination) and supplementary services (e.g., call transfer, call deflection, and camp-on busy) without requiring the switch to support any specific features. The network is only required to provide two types of end-to-end connection, realtime and non-realtime. Any signaling required is accomplished by control information sent over a peer-to-peer signalling channel between end systems. Encoded voice samples are exchanged over another peer-to-peer voice channel. This end-system-oriented approach offers advantages such as scalability, potentially faster introduction of new supplementary services, and the elimination of some centralized functions. Using a reference architecture, this paper describes the signalling messages required and the details of how the basic and some supplementary services can be implemented.

1 Introduction

A telephone system provides basic services (call establishment and termination), and a variety of supplementary services such as call transfer, call deflection, and camp-on-busy. A list of supplementary services defined for the public switched telephone network (PSTN) can be found in [1]. Traditionally, supplementary services are implemented using a centralized approach for which a switch or a private branch exchange (PBX) plays a crucial role in the implementation.

In recent years, advances in personal computer, networking, and digital signal processing technologies have spurred interest in computer-telephony integration (CTI) [2]. The basic technique in CTI is to connect a workstation to a LAN for data service and to a PBX (and PSTN) for voice service. A user can then place or receive calls from the workstation. This technique utilizes the services offered by two separate networks (for data and voice), and any telephony services (basic or supplementary) are implemented at the switch (i.e., using a centralized approach).

More recently, we have witnessed significant interest in the implementation of telephone applications on a single, integrated-services network, thus eliminating the need for separate networks for voice and data. This is substantiated by the various products that allow users to place phone calls over the Internet [3]. The basic technique is to package encoded voice samples into messages and transmit these messages over the network (the Internet in this case) like any other data message. In addition to voice transport, the network can be used to transmit signalling information directly between workstations (end systems) for call control purposes. This end-system-oriented approach exploits the direct signalling capability between end systems, and allows one end system to place a call to another end system or to request supplementary services without imposing any specific requirement on the network infrastructure, except that it provides end-to-end transport of the voice and signalling messages between the end systems.

The end-system-oriented approach is an option specified in the ITU-T recommendation H.323 [4] [5], which is an industry-wide accepted protocol standard for providing audiovisual communications on local area networks. Specifically, in the "Direct Endpoint Call Signalling" option, signalling messages are exchanged directly between endpoints without involving the gatekeeper (a centralized component in H.323). Regarding supplementary services, H.323 refers only to the Q.931, Q.932, and Q.95X series of recommendations. However, it does not specify how supplementary-services-related signalling messages are handled. It is therefore optional as to whether an end-system-oriented approach is used in the implementation of supplementary services.

In this paper, we describe a method (based on the end-system-oriented approach) for implementing supplementary services. In Section 2, we illustrate the centralized approach by means of an example. In Section 3, the basic principle of our end system-oriented approach is described. Some assumptions and a reference architecture to facilitate our discussion are also presented. The signalling messages as well as the details of how the basic services can be implemented are described in Section 4. This would provide the framework required for the implementation of supplementary services. Section 5 is concerned with implementating supplementary services, for which we illustrate our method by means of several examples. Finally, Section 6 contains some concluding remarks, including a discussion of the advantages of the end-system-oriented approach.

2 Centralized Approach

In this section, the centralized approach is illustrated by means of an example, which shows how the "camp-on busy" service is typically implemented in a PBX. The sequence of events is shown in Figure 1. Suppose user B is busy, and user A would like to place a call to user B. User A's call request first goes to the PBX, which checks user B's status, and informs user A that user B is busy. User A subsequently

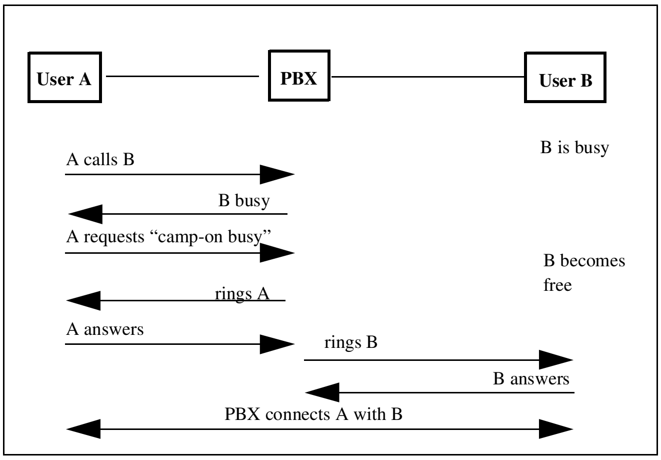

Figure 1: Camp-on Busy Supplementary Service

requests "camp-on busy". The PBX then monitors user B's status, and when user B becomes free, it informs user A by ringing A. When user A answers, the PBX rings user B. Finally, when user B also answers, the PBX sets up a phone connection between users A and B.

From the above description, it can be seen that the PBX is intimately involved in the execution of supplementary services. Status information is kept at the PBX, and a signalling protocol between the PBX and the user is required.

3 End-System-Oriented Approach

3.1 Basic Principles

In our end-system-oriented approach, a peer-to-peer signalling channel is used by end systems involved in a call to exchange the control information required for the implementation and execution of basic and supplementary services (see Figure 1). In addition, a peer-to-peer voice channel is used for the transfer of voice packets. The switches or routers (within the integrated-services network) are not involved in the execution of supplementary-service-related functions; they only provide the means for the end systems to exchange signalling messages and voice packets.

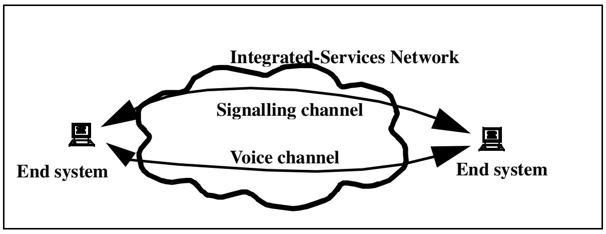

Figure 2: Basic Principles

3.2 Assumptions

In this subsection, we briefly discuss the system features required for the implementation of basic and supplementary services. A number of assumptions made to facilitate the discussions in the rest of the paper have no impact on the generality of our approach.

We assume that each user is identified by an electronic mail (email) address, and that the signalling channel is realized by a TCP connection. To establish the signalling channel, the calling user needs to determine the TCP address (i.e. IP address and port number) of the called user. We assume that this TCP address is either known to the calling user, or can be obtained from an address mapping function. The details of the address mapping function are beyond the scope of this paper.

We also assume that the networking technology used to provide the transport service is ATM or IP subnet. In both cases, the TCP connection for the signalling channel is set up over IP (IP over ATM is assumed for the ATM case). Concerning the voice channels, voice communication often requires a quality of service (QoS) guarantee from the transport service, e.g., acceptable end-to-end delay and delay jitter. Such guarantees could be provided by a virtual channel connection for the case of ATM or by RSVP for the case of IP subnet. In all cases, we assume that the calling end system is responsible for initiating the voice channel setup. The address information required is sent over the signalling channel as part of the call establishment procedure. The details of the setup or release of a voice channel are dependent on the subnetwork technology used, and will not be discussed in this paper.

3.3 Reference Architecture

The software architecture shown in Figure 3 is used to facilitate the discussion of our method to implement basic telephony services and supplementary services. Three software modules are defined at the end system: call manager, user application, and transport system. The call manager (CM) is responsible for call processing. It interacts with its peer by exchanging signalling messages. It also processes requests from the user application. The user application supports a graphical user interface. It also plays a role in the implementation of basic and supplementary services. For example, if the user is prepared to accommodate camp-on busy, the information is kept at the user application. In our discussion, the terms user and user application are used interchangeably. The transport system provides the services required by the call manager to set up the signalling channels and voice channels.

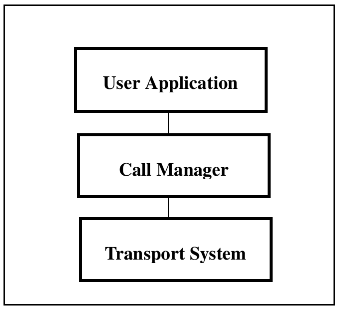

Figure 3: Reference Architecture

Interface primitives exist between the user application and the call manager, and between the call manager and the transport system. The details of these primitives are beyond the scope of this paper.

4 Basic Telephony Services

As mentioned above, the details of how basic services (call establishment and termination) are implemented will provide the framework required for the implementation of supplementary services.

In this section, we first introduce some signalling messages that are sent between peer call managers, and then illustrate how these messages are used to implement basic services.

4.1 Signalling Messages

Four signalling messages are defined for the implementation of the basic services: SETUP, CALL_PROCEEDING, CONNECT, and RELEASE. The meanings of these messages are similar to those defined in Q.931 [6]. Each message may have a number of parameters associated with it. Throughout our discussion, the notation MESSAGE_NAME (param1, param2, ...) will be used to refer to a signalling message with name MESSAGE_NAME and parameters param1, param2, ...

4.2 Call Establishment

We first describe the procedures of the call manager for the case of successful call establishment. The sequence of events is shown in Figure 4. Upon receiving a place call request from user A, the call manager at end system A (CM-A) sets up a signalling channel to CM-B (the call manager of the called user), and sends a SETUP signalling message over this channel. When CM-B receives this message, it indicates the incoming call to user B. User B's status is that the call should proceed. So CM-B returns a CALL_PROCEEDING message to CM-A, with a parameter indicating "user free". CM-A, upon receiving this message, indicates to user A that the call is in the ringing state. Some time later, user B answers the call. When this happens, CM-B sends a CONNECT message to CM-A. The address information required for setting up the voice channel (i.e. the ATM address in the case of an ATM subnet and the UDP address in the case of an RSVP/IP subnet) is included in this message. Upon receiving the CONNECT message, CM-A proceeds to setup a voice channel. When the voice channel is established, both CM-A and CM-B indicate to their users that the call is now active.

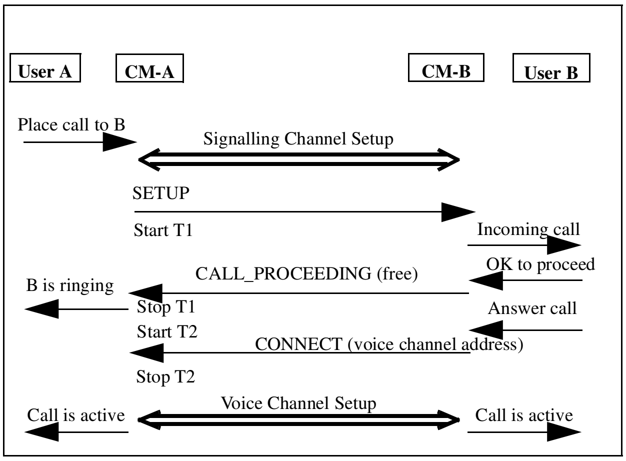

Figure 4: Call Establishment - Successful Case

Two timers, T1 and T2, are used by CM-A to handle exceptional conditions (see Figure 4). The expiration of T1 means that a response to the SETUP message has not been received. This indicates severe network congestion because a reliable TCP connection is used to exchange signalling messages. On the other hand, the expiration of T2 indicates that user B is not answering the call. In both cases, call establishment is not successful, and CM-A will terminate the call establishment procedure. Details of call termination are given in Section 4.3.

Another scenario in which call establishment is unsuccessful is when user B is busy. In this case, a CALL_PROCEEDING message with a parameter indicating "user busy" is returned to CM-A instead of "user free", (see Figure 4). When this happens, CM-A indicates "B is busy" to user A, and allows user A to request certain supplementary services (e.g., call back or camp on busy), or simply to terminate the call. Details of the provision of supplementary services are described in Section 5).

Note that the above description concerns the basic procedure of call establishment. Other features can be added. For example, in the SETUP message, one can include the email address of the called user as a parameter. This would allow CM-B to check whether the called email address matches the email address of user B. One can also include the email address of the calling user or further information (e.g. subject of the call) in the SETUP message. This would allow user B to screen calls based on caller identity. Furthermore, one can use the signalling messages to negotiate quality of service parameters for the voice channel.

4.3 Call Termination

At any time, both CM-A and CM-B can initiate the termination of a call. This may be due to a request from its user or some time-outs. Suppose CM-A initiates call termination. It sends a RELEASE message on the control channel to CM-B. Any running timers are stopped, and the signalling control channel and any existing voice channel of the call are released (see Figure 5).

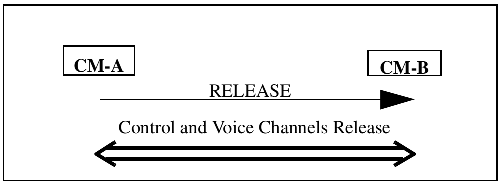

Figure 5: Call Termination

Upon receiving the RELEASE message, CM-B stops any running timers, and releases the control channel and any existing voice channel of the call.

5 Supplementary Services

In this section we describe how supplementary services can be implemented using our end-system-oriented approach. For ease of exposition, our description will be based on a number of selected supplementary service examples.

We first introduce four additional signalling messages designed to support the implementation and execution of supplementary services.

SS_INFO	Inform the peer CM about certain supplementary services.
SS_REQUEST	Request the peer CM to perform actions relating to a certain supplementary service.
SS_CONFIRM	Response to a SS_REQUEST to confirm the processing of a supplementary service that the peer CM has requested.
SS_REJECT	Response to a SS_REQUEST to reject the processing of a supplementary service that the peer CM has requested. The reason for the rejection is included.

Each message may have zero or more parameters, and can be sent as a separate message at any time between a CALL_PROCEEDING and a RELEASE message. Sometimes an SS_INFO message is required in parallel to a call establishment or termination message. In such cases, the parameters to be carried in the SS_INFO message will be carried instead in the SETUP, CALL_PROCEEDING, or RELEASE message to reduce the number of control messages to be exchanged.

We next describe how some examples of supplementary services can be implemented.

5.1 Camp-on Busy

In Section 2, the centralized approach to implementing camp-on busy was described. Our first example is concerned with the same supplementary service, and our description will show the difference between the centralized approach and the end-system-oriented approach.

Suppose a user B has activated "camp-on busy" (i.e., user B would accept camp-on calls). At end system B, CM-B keeps a record for camp-on users. User B can review this record and decide which camp-on call to accept when the current call is terminated.

The procedure for the execution of camp-on-busy is illustrated in Figure 6. Suppose that user B is busy when user A places a call. CM-B returns a CALL_PROCEEDING message with parameters indicating that the user is busy, but that camp-on is possible. If user A wishes to request camp-on busy, CM-A will send a SS_REQUEST message to CM-B. When CM-B receives this message, it enters user A's email address into user B's camp-on record and replies with a SS_CONFIRM message. Suppose user B, after having terminated the current call, examines his/her camp-on record and decides to answer user A. CM-B then sends a CONNECT message to CM-A to establish the call.

Note that CM-A uses a timer T2, which will expire if user A does not wish to request any supplementary service.

A comparison of the procedures shown in Figure 1 and Figure 6 shows that the end-system-oriented approach requires no involvement of the PBX, and that camp-on busy can be implemented with

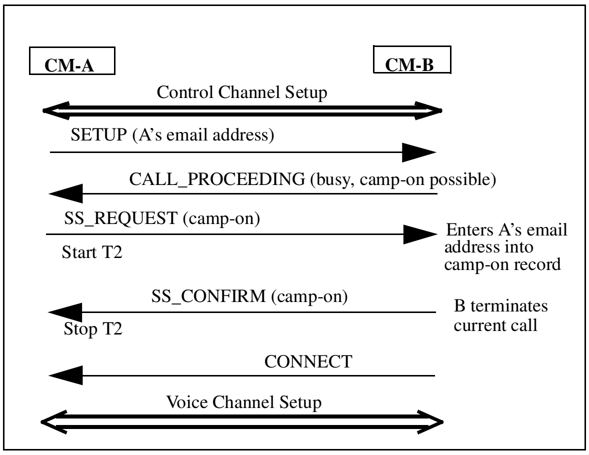

Figure 6: Camp-on Busy

the appropriate definition of signalling messages, as well as the appropriate design of procedures at the end systems. Using the same design approach, other supplementary services can readily be implemented.

5.2 Call Deflection

Our next example is call deflection, which allows a user to forward a call to another user. Suppose user B is busy and has activated "deflect to user C on busy". If CM-B receives a SETUP signalling message from CM-A, it will return a RELEASE message with SS-INFO parameters "deflect to C" and user C's address. CM-A then terminates the call placement to CM-B and initiates a new call to user C. Other variants of call deflection are deflection on no answer and immediate deflection. It is easy to see that the procedures for these other variants are similar. For example, the procedure for deflection on no answer is illustrated in Figure 7.

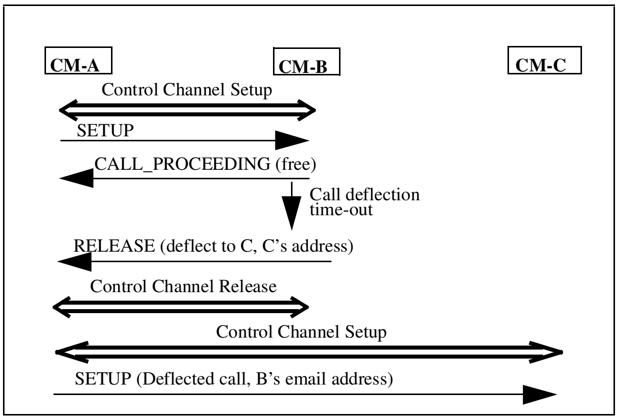

Figure 7: Call Deflection on No-Answer

5.3 Hold and Activate Call

In general, a user can have several calls set up simultaneously. At most one of these calls can be "active", while the others are "on hold".The status of a call can be changed by means of an SS_INFO message. Specifically, CM-A can indicate change of status to hold (or active) by sending an SS_INFO(hold) (or SS_INFO(active)) message to CM-B (see Figure 8).

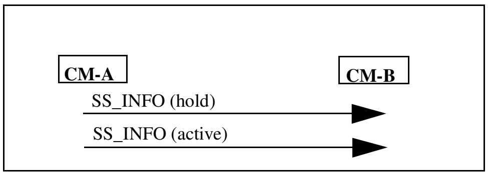

Figure 8: Hold & Activate Call

5.4 Call Transfer

A more complex example is call transfer. The typical scenario is that user A, having an "on hold" call to user B and an "active" call to user C, would like to request the establishment of a new call connecting users B and C. The procedure at the call manager is illustrated in Figure 9. When a call transfer request is received from user A, CM-A

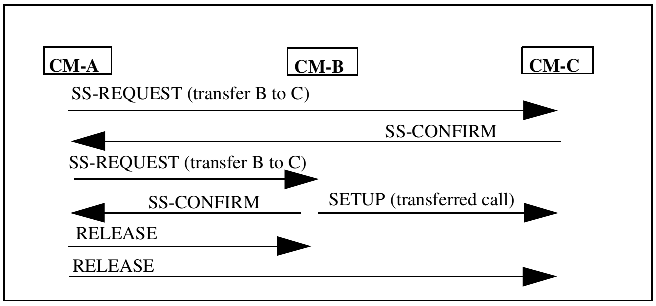

Figure 9: Call Transfer

sends an SS_REQUEST message to the CM-C to ensure that CM-C will give priority to the transferred call coming from user B. This request is subsequently confirmed by an SS_CONFIRM from CM-C. CM-A then sends an SS_REQUEST to CM-B requesting the placement of a "transferred" call to user C. This is again confirmed by the corresponding SS_CONFIRM message. At this point, CM-A can terminate both calls to users B and C.

In the meantime, CM-B initiates a call to user C according to the procedure described in Section 4. The only exception is that the SETUP message sent by CM-B contains the indication that it is a "transferred" call, thus allowing CM-C to distinguish this call from any other incoming calls that may arrive while it is waiting for call transfer. This feature would help ensure success of the transfer, e.g. user C would not become busy before the transferred call from CM-B has arrived.

5.5 Messaging service

Finally, we consider a new supplementary service called "messaging service", which can be developed as a result of our end-system-oriented architecture. A user A can use this service to transfer to user B a short message in case he/she finds that user B is busy or does not answer the call. The message may be a text or an audio message.

It should be noted that this messaging service is synchronous in the sense that the sending user knows that the message is already stored in the remote end system. It is different from email because one does not know in advance when the message will arrive at the destination. In our end-system-oriented architecture, the text or encoded voice messages are carried in SS_REQUEST messages and confirmed with SS_CONFIRM messages as soon as they are stored on the remote end system. In the case of encoded voice, it is also different from the conventional "answering machine" service because the calling user records the voice message on his local system and transfers it to the remote end system via the end-to-end signalling control channel. Thus no voice channel is required.

A variation of this messaging service is to provide users with a means of exchanging short text messages in parallel with their voice conversation. This can be used, for example to send a name to the remote user instead of spelling that name over the voice channel. Another variation is the ability to send a text message within the SETUP message, thus allowing a calling user to inform the called user about the purpose of the call.

6 Concluding Remarks

In this paper, we have developed an end-system-oriented approach to implementing telephony supplementary services, and have shown that, with proper definition of signalling messages and proper design of procedures at the end systems, a wide range of supplementary services can be implemented without the involvement of a switch or PBX. It offers a number of advantages over the traditional approach of a centralized implementation. First, because the supplementary service-related functions are moved to the end system, the processing power of the end system can be used to provide a more flexible user interface. Moving these functions to the end system also permits the implementation of new supplementary services without changing the underlying network infrastructure. This may result in the faster introduction of new supplementary services or of a modified version of existing supplementary services. Finally, our end-system-oriented approach leads to lower complexity and cost of centralized components because some centralized functions have been eliminated. It is also scalable in terms of supporting many users.

7 References

[1] ITU-T Recommendation Q.950, "Digital Subscriber Signalling System No. 1 (DSS 1) - Supplementary Services Protocols, Structure and General Principles"

[2] "Standard Issue", Byte Magazine, September 1995, 201-207.

[3] VON - Internet Telephony, http://www.von.com/teleph.html

[4] ITU-T Draft Recommendation H.323, "Visual Telephone Systems and Equipment for Local Area Networks Which Provide a Non-Guarenteed Quality of Service", decided May 28, 1996

[5] G. A. Thom, "H.323: The Multimedia Communications Standard for Local Area Networks", IEEE Communications Magazine, Vol. 34, No. 12, December 1996

[6] ITU-T Recommendation Q.931, "Digital Subscriber Signalling System No. 1 (DSS 1) - ISDN User-Network Interface Layer 3 Specification for Basic Call Control