MANAGING MULTIMEDIA SESSIONS ON A PRIVATE BROADBAND COMMUNICATION SYSTEM

An object-oriented transaction-oriented approach including signalling aspects

Ralf Cordes **Dieter Wybranietz** **Rolf Vautz**

Telenorma GmbH
Bosch Telecom
Advanced Systems Development Division
Mainzer Landstraße 128 - 146
D-6000 Frankfurt/Main 1

1 Scope

Multimedia communications comprising the exchange of audio, video and data information will play an increasingly important role in the telecommunications market (2,4,8,12,15). These new applications set up new requirements for the communication infrastructure ranging from the physical interconnection to the management and control of these services (1,3,5,7,11,16). Different classes of multimedia communication services are discussed as a case study which is based on a prototype of a distributed broadband communication system. By an advanced distributed multimedia application we show how timing and structuring requirements, session management and synchronization are supported by distributed control software including signalling and call control.

2 Scenario and Outline

Assume an enterprise-wide communication network of a publishing company. Using a multimedia terminal a reporter initiates a video conference from site T2 with his partner, who is the editor of a journal for which the reporter had written an article (see Figure 1).

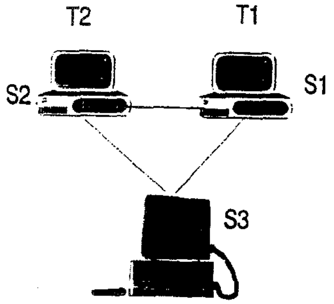


Figure 1: The Publishing Scenario

Before, the reporter had received his reviewed and annotated document (voice and text annotations are possible) from a multimedia document server (site S2). The editor sitting at site T1 accepts the video conference request and the communication starts. During this session the editor mentions that he has made some additional comments on the paper which he would like to discuss with the reporter. This version of the article is sent via a file-transfer to site T1 from site S1 where it was stored. By comparing the different versions using a joint pointing device which is visible on both screens they work cooperatively on the document. Assume further that during this conference the editor proposes to look up information stored in a database to clarify controversely discussed items. The editor (T1) initiates a query against a distributed database management system (DDBMS) located at sites S1, S2, and S3; whereas the global management resides at site S2. The request and the response from the DDBMS are displayed simultaneously on both screens. During the whole conference, both partners can see each other in a dedicated window on the screen.

This is one example of a typical future application scenario employing multimedia information exchange. There are a variety of other examples including distributed language laboratories, travel agencies, hospitals, education etc. as compared to todays communications facilities (16). All of these applications have in common that during one "logical" call several basic connections exist. For example, in the scenario presented above one-to-one audio/video and data connections have to be established or to be released. Thus, a multimedia call does not just consist of several connections, but these connections are subject to change dynamically. Take the database query of the sample application given, the signalling and call control software of a future communication system therefore has to offer primitives and mechanisms to handle these new requirements.

In particular, there are some main problems to be solved. Take as one example the establishment of the videophone connection. If the two videophone terminals offer different qualities, both sites have to agree on a minimum context with a special kind of Q.o.S. Therefore, we have to use a special type of negotiation by setting up the connection provided by signalling primitives. On the other hand the multimedia retrieval call against a distributed database has to be supported by special protocol primitives offering a multilevel transaction protocol. These protocol primitives should provide the facility of setting up connections from a third site, in our case the DDBMS has set up connection to its subsystems which are located somewhere across the network. To avoid a bottleneck at the DDBMS site, every subsystem should be able to set up connections to the requestor sites. With these functionalities we are able to cascade connections in a multicast- and broadcast-oriented manner providing different levels of end-to-end control.

To summarize, we have to use new techniques for system software supporting multimedia communication sessions on private communication systems. Examples are:

- in-call bandwidth modification
- negotiation of communication contexts with suitable Q.o.S.
- integration of servers providing cascaded connections with end-to-end control
- transaction-oriented protocols providing logging, roll-back, and security aspects
- object oriented structuring of generic application and service elements.

After giving a rough impression about our object oriented approach, we show the timing diagramme of the possible transactions supporting the scenario showing above. After discussing the architecure of our underlying communication system we present the appropriate primitives for the multimedia signalling protocol.

3. Structuring of generic service and application components

To gain more flexibility by integrating services on a multimedia terminal we have to structure the multimedia service components in an appropriate manner (4). This approach should be independent from a dedicated application and should provide generic components for different applications which are easily configurable to complex services and/or applications. These components are called generic application and service components. For us the concept of hypertext in combination with an object oriented approach of structuring the multimedia information units offers the suitable features meeting the requirements mentioned above. These generic multimedia application components are running on multimedia terminals (5) connected to a basic communication system.

In particular, we distinguish between object classes for composite (complex) objects (so called *PAGES)*, for monomedial objects (called *PARTICLES*), for linking and anchoring objects, and for telecommunication service elements. Excerpts from our class structuring is given in the Figure 2. The conceptual basis is given by the DEXTER model (6,10) serving as a framework for a network wide information information service (open link server) and the object oriented structuring of multimedia material given by the MHEG proposals (9). According to the DEXTER model we distinguish between the

- presentation,
- storage, and
- physical

layer for the generic components.

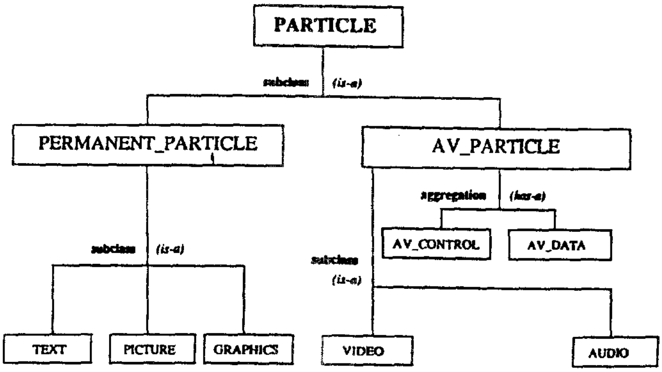


Figure 2: Excerpts from the Class Structure

The instantiated objects in the storage layer of our system have a dedicated slot providing the default presentation of this object. The presentation manager decides whether the default presentation should used or to present it according to the calling application or service context. Each *PARTICLE* contains the information about connections (links) to other particles and/or pages. These *LINKS* are also first class objects and can be subdivided into two classes. One class is dedicated to the traversal of the network wide information pool (*NAVIGATIONAL LINKS*); the other one supports the synchronization of media streams, especially for connecting time independent particles with time dependent ones (*SYNCHRONIZATIONAL LINKS*). In more detail, we subdivide the *NAVIGATIONAL LINKS* according to the source and destination of the link like *PAGE*, *PARTICLE*, and *TELECOMMUNICATION ELEMENT*. Using this extra structuring of linking information we are able to encapsulate and hide typical telecommunication related information for services and components to provide dialing, setting up connections, and accessing protocols like FTAM or MHS.

4 Transaction oriented Support for Multimedia Services

The object oriented structuring of service and application components is embedded in a multilevel transaction system (13,14) which offers:

-- *Usage of the flexibility of an ATM network*: To provide flexible allocation and release of bandwidth for parallel or modified virtual channels during one physical multimedia session.

-- *Support of parallel processing* according ot the object oriented structuring of the generic service components based on message passing.

-- *Avoiding of bottlenecks* during one multimedia session by flexible routing and flexible allocation of network pathes using multipoint protocol primitives.

-- *Usage of fine grained set-up and roll-back functionalities* according to the touched objects by appertaining of subtransactions on one special level.

We will demonstrate our approach by analyzing the scenario described in Section 2 and presenting a diagramme (see Figure 3) of possible transactions and their levels.

The begin of the multilevel transaction is the request from T1 at the global management of the DDBMS at site S2: *req(S2)*. S2 has to invoke the subsystems at S1, S2, and S3 with the dedicated query for each site: *req(Si)*. In parallel on each site the search process runs. These processes produce the result of the query which will be delivered in parallel to the sites T1 and T2: *send(Ti)*. If the information has arrived, an acknowledgement is send to the sites Sj from every Ti: *ack(Sj)*. If the processes running at sites Sj dealing with the query are finished without an error, an acknowldegement is send the global management at S2 from every site: *ack(S2)*. At the end if there are no errors the calling site T1 will receive an *ack(T1)* and the whole distributed query has been managed by the system.

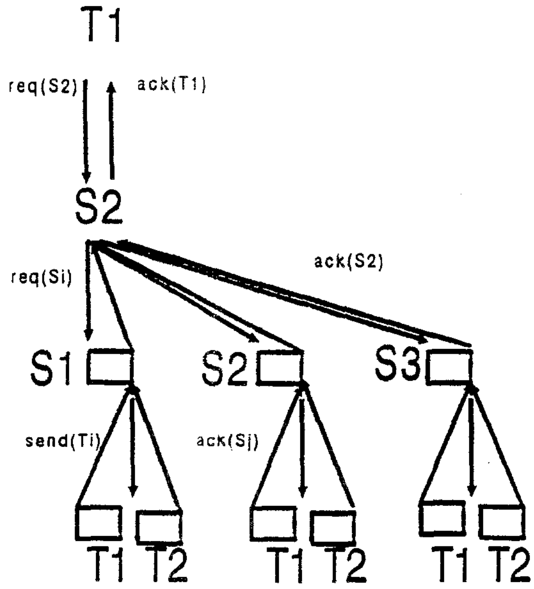


Figure 3: Timing Diagramme of Transactions in Scenario

5 Architecture of the communication system

Our considerations are based on an advanced distributed communication system for the customer premises market. Such a system is currently being developed in the course of the DAMS project (Dynamically Adaptable Multiservice System). The main objective of DAMS is to address the problems associated with the integration of services which are based on either circuit-switched or packet-switched techniques within the business environment (see Fig. 4). DAMS will provide for both delay-sensitive telephony services and time-indepented data services a fully integrated and cost-effective system which will fulfill future customer requirements. Further goals are flexible use of bandwidth, adaptability to changing user demands, increased realibility and availability.

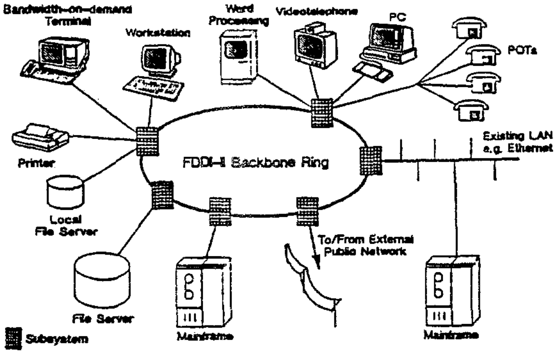


Figure 4: The DAMS Architecture

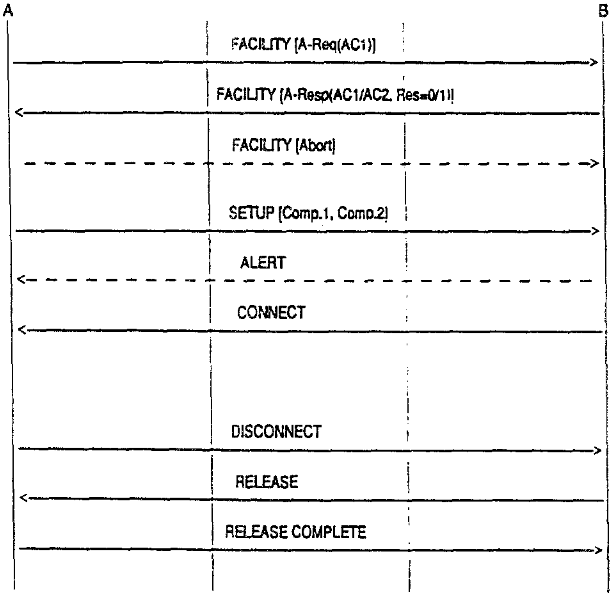
In order to cater for evolving customer requirements, flexibility has been a major concern in the specification of the system. It consists of autonomous subsystems, which can operate in a stand-alone mode or can be connected by a backbone system to build large networks or extend the system to greater distances. So-called DAMS nodes which can adapt to various interface types (e. g. ISDN basic access, ISDN Primary Rate Access, Ethernet, Token Ring, IEEE 802.9 IVDTE Terminals etc.) offer a local switching capacity and can be interconnected by a backbone system. The DAMS backbone unit is based on the FDDI-II standard defined by ANSI-subcommittee X3T9.5. In principle, the DAMS nodes can be interconnected by different topologies and transfer technologies. The bandwidth for the interconnection of the subsystems can be extended by the use of parallel backbone systems or a backbone with higher transmission rate.

The DAMS node internal switching is realized by a special hybrid mechanism operating on a bus system. In order to be prepared for the new standard in telecommunication, the evolution of the DAMS system concept towards ATM has already been considered. The switching principle of DAMS allows the integration of isochronous as well as non-isochronous data switching. The hybrid ATM/STM local switch will be achieved by introducing a connection-oriented switching technique based on the ATM-header structure defined by CCITT. An ATM-header is added to each incoming packet which is then switched based on a contents of the virtual channel identifier (VCI). The DAMS node internal signalling and control procedures are supported by signalling protocols based on the ATM-cell structure where these cells are routed using the VPI/VCI contents. The allocation of non-isochronous slots is controlled by a distributed media access protocol. Packets gain access to the bus by using non-isochronous slots. The isochronous traffic is circuit-switched between the related units allowing fixed and variable bandwidths (in terms of multiples of the basic access rate 64 kbit/s).

The distributed hardware is controlled by a distributed operating system called PordOS which fulfils the above mentioned requirements. In this context, support of real-time processing, migration of system modules during operation as well as load sharing capabilities are provided. The operating system is composed of the kernel and a set of services running on top of the kernel. The kernel provides process management, communication facilities, exception handling, I/O-support, memory management as well as test and debug aids. The services on one hand implement the environment needed to run the application programs and, on the other hand, mechanisms and strategies to support advanced systems features. PordOS also provides a good basis for the implementation of high level structuring facilities as, for example described in (17).

6 Signalling Aspects

The system described above provides basic mechanisms to reserve isochronous bandwidth for arbitrary bitrates and the transfer of data. The hybrid approach allows the software-controlled allocation of isochronous and non-isochronous bandwidth. The multimedia scenario given above requires more advanced facilities. Several of these basic connections have to be combined into one multimedia call. We will discuss possible modifications to the Q.931 protocol to make it compatible with the new requirements of broadband communications. Although, at this moment, it is not likely that an enhanced Q.931 protocol will be the final solution for B-ISDN, the identification of the parameters to be inserted to handle ATM connections and complex multimedia services will make easier the development of a new protocol. In the following discussion, the message structure as defined in Q.931 is maintained. Therefore, every message may comprise a protocol discriminator, call reference, message type, and other information elements as required. While most of the Q.931 messages may remain such as "call information phase" messages (resume, suspend, user information etc.), the "call clearing" messages (disconnect, release etc.) and the "miscellaneous" messages (facility, information, notify etc.) changes are needed in the "call establishment" messages (alerting, call proceeding, connect, set up etc.). In this paper, we will focus on two new protocol procedures: Context negotiation and the request for allocation of new channels during an established call.



AC = Application Context
A-Req = Associate-Request
A-Resp = Associate-Response
Res = Result = 0 (positve) / 1 (negative)
Comp. = Service Component

Figure 5: Context Negotiation before Call Establishment

The "context negotiation" procedure is described as a supplementary service which can be invoked before the request to establish a call (see Fig. 5). The introduction of "context negotiation" as a supplementary service will have the following advantages: Invoking this supplementary service before call set up will provide the user with the possibility to negotiate with the remote user terminals all necessary parameters required for the call with the distinguished service. If there are terminals at the called user site which are fully compatible to the requested service the call will be established only to one of these terminals. If the distinguished service cannot be supported at the called user site but if there are terminals which provide an alternate compatible service, the calling user will have the opportunity to decide if a call with this alternate service shall be established. Without initiation of the "context negotiation" there is no guarantee that the call will be established with a remote terminal with the highest level of compatibility. In this case, if the requested service is not mandatory, the call would also be established with a terminal with a lower level of compatibility that responses faster. In Figure 6 we show the structuring of the Facility elements for context negotiation.

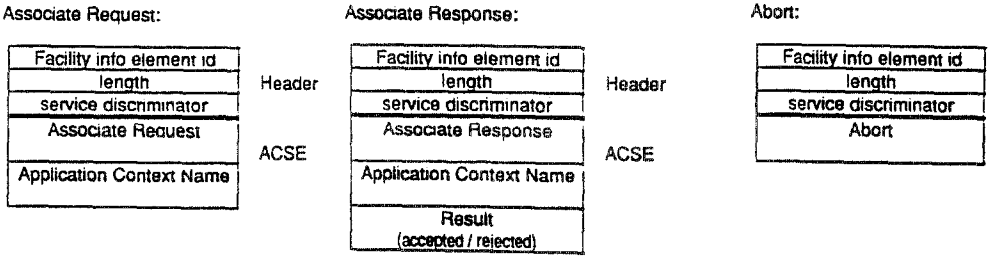


Figure 6: Facility information elements for Context Negotiations

The "allocate" message is sent from the user or the network to request the allocation of new channels during an established call. In the allocate message more than one connection can be requested at the same time and the information elements needed to characterize them are the same as those presented in the set-up message. As an enhancement of the allocate message, a "modify" message is introduced which can request the modification of active channels. Both, the modifiy and allocate messages, are especially needed for multimedia calls as in the application scenario described above. In Figure 7 the modification during an active call is shown in detail.

7 Current Status of Work and Outlook

A first demonstrator of a distributed multimedia application based on a private broadband switching system has been presented at the TELECOM 91 in Geneva in October. The described prototype of the distributed communication system will be available in 1992 within in ESPRIT and RACE programme for supporting multimedia communication sessions.

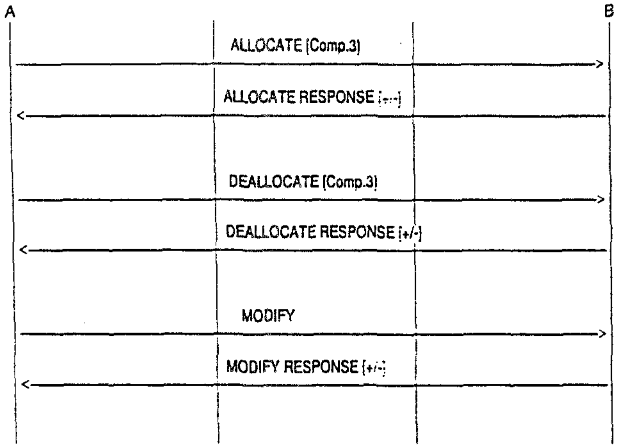


Figure 7: Modification during active Call

Acknowledgement

Most of the work presented has been supported by the European Community within the projects DAMS (ESPRIT 2146), MCPR (RACE 1038) and BUNI (RACE 1044). We would like to thank all the partners and institutions for fruitful discussions and their cooperation.

References

(1) Berra, P.B. et al.; Architecture for Distributed multimedia Database systems, Computer Communications, vol.13, no.4, May 1990

(2) Bulick, S. et al.; The US West Advanced Technology Prototype multimedia communication System, GLOBECOM89, Texas Nov. 1989

(3) CACM Special Issue: Standards and the Emergence of Digital Multimedia Systems, April 1991

(4) Cordes, R.; Kummerow, T.; Multimedia Communication and Information Management Based On Available and Emerging Standards; Proc. IEEE Workshop on Telematics, Cheju Island, Korea, 1991

(5) Cordes, R.; On the Way to Hypermedia and Multimedia Services and Terminals; Proc. Interactive Communication Tools, Paris, May 1990

(6) Halasz F., Schwartz M; The Dexter Hypertext Reference Model; in (10)

(7) Little T.D., Ghafoor A.; Synchronization and Storage Models for Multimedia Objects. IEEE Journal on Selected Areas in Communication, Vol.3, No.3, April 1990

(8) Ludwig, L., Dunn D.F; Laboratory for the Emulation and Study of Integrated and Coordinated Media Communications, Proc. SIGCOMM Conf., Aug. 1987, Stowe Vermont

(9) MHEG Document S.3, Summer 1991

(10) NIST, Hypertext Standardization Workshop, Jan. 1990

(11) Sheperd W.D., Salmony M.; Extending OSI to support Synchronization Required by Multimedia Applications; Computer Communications Vol.13, No.7, Sept. 1990

(12) Watanabe K. et al.; Distributed Multiparty Desktop Conferencing System: MERMAID, Proc. Conf. on CSCW, Los Angeles, 1990

(13) Weikum G., Principles and Realization Strategies of Multilevel Transaction Management, ACM TODS Vol.16, No.1, 1991

(14) Weikum G, Haase C.; Multi-Level Transaction Management for Complex Objects: Implementation, Performance, Parallelism, ETH Zürich, No. 162, July 1991

(15) Wybranietz, D.; Cordes, R.; Stamen, F.-J. ; Support for Multimedia Communication in Future Private Networks; 1st Workshop on Network and Operating System support for Digital Audio and Video, Berkeley, Nov. 1990

(16) Wybranietz, D.; Strategies for Future Private Communication Systems, to appear: 3rd International Symposium on Systems Research, Informatics and Cybernetics, Baden-Baden, Aug. 1991

(17) Wybranietz, D.; Buhler, P.; The LADY Programming Environment for distributed operating systems; Future Generation Computer Systems, Vol.6, North Holland, 1990, pp 209-223