The Information Footprint – A satellite-based Information on Demand Teleservice

Karl Jonas, Helmut Jungblut, Joachim Kaeber, Manfred Kaul, Ina Müller, Horst Santo, Josef Schäfer, Ralf Wegner

EMail: <first name>.<last name>@gmd.de

Abstract

An INFORMATION ON DEMAND (IoD) Teleservice developed at the German National Research Center for Information Technology (GMD) provides remote access to multimedia information consisting of audio, video, and text. It uses a bidirectional narrow-band message link between the end user and the service provider, and a unidirectional broadband data link from the service provider to the end user.

Hyperstructure of the multimedia database, clear information presentation (on SUN workstations and on PCs) and high speed communication with long round-trip-delays were the major problems to be solved.

Since the IoD teleservice uses a satellite connection (among others), it turned out to be necessary to implement an access protocol optimized for the access of real-time multimedia data across a long-delay high-bandwidth link, a LONG FAT PIPE [2] ; we called it the MEDIASERVICE PROTOCOL (MSP).

This paper introduces the Information on Demand Teleservice and describes a first implementation.

Keywords: Satellite communication, communication protocols, information retrieval, video coding, Information on Demand.

I. Introduction

In the emerging field of multimedia applications new partners from previously diverging fields as computer science, communication technology and media industry are coming together to explore new ways of using technology and define completely new products and new markets.

In the melting pot of multimedia the rules of the game are just beginning to be defined. A whole range of basic questions are still unsolved.

Since the processing power begins to allow for multimedia applications, computer science starts embracing audio and video media as well. Video on Demand (VoD) teleservice is discussed as a possible killer application for multimedia systems.

In the area of teleconferencing, some research and development concentrates on using existing net-working protocols, such as IP/UDP, as a transport for audio and video data. The Multicast Backbone (MBONE) provides the infrastructure necessary for world-wide audio/video conferencing and telecooperation services and comprised some 750 subnets in april 1994 [3].

Still, there is a technology lack enabling mass applications. And it remains still unclear, how a sufficient market for VoD could evolve.

Therefore, pilot projects are launched all over the world testing the technological intricacies, and looking for interesting applications, for which a market might exist.

On the other hand, traditional media industry begins to discover the new range of applications enabled by emerging multimedia systems. They wonder what the technological possibilities are, how they could make use of them and how they will sell their products in the multimedia marketplace.

From a technology point of view, the target of the project is a user friendly integration of the three technology families information, communication and media technology: Reflecting the state of the art, only some first steps have been achieved up till now.

Starting from a user’s point of view, the multimedia interface –either under DOS/Windows or SUN/Open Windows– needs to be intuitive, supporting traditional features like fast forwarding of video streams, and still offering added value (jumping, hyper links…) without confusing the user.

On the other end (of the OSI stack), the communication subsystem should be optimized for guaranteed delivery (of user requests) and real-time provision (of the requested information). The trade-off between high quality and high speed needs has to be handled.

Many R&D projects are on the line, tackling various aspects of the problem stated. Among the most successful applications that are already used is the World Wide Web (WWW).

During the soccer worldcup 1994, Sun, EDS and Sprint provided the WWW server http://www.worldcup.com. After the worldcup, this server has been decommissioned and the database has been archived.
The Information Footprint Proceedings JENC6

K. Jonas

Opposite to the Sun project, our IoD project laid emphasis on complete inclusion of all news agencies' information and high quality full size video covering whole games indexed in such a way that by content based retrieval particular video scenes can directly be accessed.

Section 2 describes the application scenario used for the demonstration of the IoD Teleservice. Section 3 introduces the MediaService Protocol (MSP) which was designed and implemented for the transmission of real-time information across a satellite link. Section 4 discusses coding issues for audio and video information and section 5 describes information retrieval problems and solutions.

The final chapter summarizes conclusions and gives an outlook on future work.

II. An Application Scenario

In many projects one of three components, the computer science side, the communication or the media part dominats, thus causing an imbalance in the research topics. Therefore, the National Research Center for Information Technology, GMD, and Westdeutscher Rundfunk, WDR, one of the largest public broadcasters in Germany, came together in the Photokina project described in this paper.

Central issues of our architecture are the interactive retrieval structures allowing for direct access to text, images, audio, and jumping into video clips. E.g. asking for all direct free kicks results in a textual list consisting of all corresponding texts, images and video clips related to this subject. Going through this list the user is jumping within the video back and forth in a way that would be impossible with conventional VCR.

WDR collected all available mediaware provided by news agencies and by WDR itself during the world cup, including news agencies' texts, very high resolution Color JPEG images, and the videos in PAL format on Betacam tapes. GMD provided all hardware and software for the IoD teleservices. In particular, a video server has been installed on the GMD campus in Schloß Birlinghoven near Bonn, storing all soccer video digitized in MPEG-1 format.

On the Photokina fair, two multimedia clients (Sun workstation and PC) provided interactive hypermedia interfaces for retrieving text, image and video information about the soccer worldcup. Information requests issued from these clients were transmitted via satellite to the video server, which in turn transmitted MPEG streams via satellite back to both clients. Video and audio data was delivered in real time on demand. MPEG hardware (Reel Magic on PC, Barney1 on Sun) decoded the incoming MPEG streams in realtime.

The asymmetric communication links of the IoD Teleservice use a bidirectional narrowband access for message exchange between client and server, and a one-way broadband link for information providing from the server to the client. The typical scenario uses a telephone or ISDN-64 connection for the first, and a broadband link ( fibre-based or satellite-based) for the latter.

During the photokina fair, both links used the satellite, with (ISDN-like) 128 kBit/s for the message service and 3.5 Mbit/s for the information service. The bandwidth was limited by the serial link to the satellite modems.

Endsystems (mediaserver and mediaclient) were connected to a local area network, with routers and satellite modems providing the connection between the networks (see figure 1).

The real scenario went even further. A conference of the Society of Motion Picture and Television Engineers (SMPTE) was broadcasted from the fair, and a live videoconference was established with partners from Sankt Augustin and from Moskow.

Since the IoD teleservice uses a satellite-based approach, participation is possible for users with ground-stations between Lisabon and Moskow, Finland and Sicilia – the Information Footprint.

III. Filling the pipe

III.A. The MediaService Protocol

The application scenario drove the need for a realtime-capable communication protocol across a satellite link. After some experiments with existing

1. prototype board from SUN for MPEG-1 video
The goal of the MediaService Protocol (MSP) is the provision of high-speed data connection for the transmission of real-time multimedia data (with focus on coded audio/video streams) across a communication link with a high \( \frac{\text{transmission-delay}}{\text{bandwidth}} \) product. [2] describes the problems TCP/IP connections experience across such a long fat pipe.

The goal of the MediaService Protocol (MSP) is the provision of high-speed data connection for the transmission of real-time multimedia data (with focus on coded audio/video streams) across a communication link with a high \( \frac{\text{transmission-delay}}{\text{bandwidth}} \) product. [2] describes the problems TCP/IP connections experience across such a long fat pipe.

The throughputs we had measured for conventional protocol stacks across our satellite link were

- 328 kBit/s with ftp;
- 115 kBit/s with ttcp (tcp)
- 38 kBit/s with rep
- 28 kBit/s with NFS / cp
- 22 kBit/s with ttcp (udp, receiver).

While a reliable transmission of all information will always be our goal, effects of retransmission (delays) can be much worse than effects of missing data in the case of continuous real-time data like audio or video. Assuming a typical roundtrip delay (RTD) of 0.5 seconds (across a satellite connection), a lost packet in a video stream would lead to a delay where either a high capacity for buffered video sequences has to be provided at the client, or a presentation delay would be recognized by the user.

Alternatively just accepting that packets might get lost requires a more sophisticated decoder (capable to handle corrupted input streams). In the case of MPEG video streams, a low packet loss rate should not be visible to the user.

Besides a (fast and unreliable) connection for data transmission (including real-time audio and video streams), the MSP as it is implemented in our client-server environment supports a reliable connection for message exchange.

After connection establishment and initialization of the two channels, MSP provides protocol functions for the (de)selection of some piece of information (like a video clip) on the server and for start and interrupt (end) of a transmission.

Window based and rate based rate control are implemented in order to prevent the server from flooding the client with packets. Both values can be set by the client.

The server provides access to some server information, such as the length of a data file and the (byte-) position of a certain frame within the file (mpeg only). This allows the implementation of sophisticated clients.

**Figure 1: The GMD IoD communication scenario (as demonstrated on Photokina ’94)**
III.B. MSP Performance

All measurements in this chapter were made across a 2'4 Mbit/s satellite link, with a bit-error-rate of less than $10^{-9}$ (after FEC) and about 500 ms RTD.

During the tests, an audiofile of 2 MBytes was transferred from the mediaserver across the satellite link to the mediaclient. In a series of tests we optimized some of the parameters that could influence throughput across the link (the maximum sender data rate, the windowsize at the server, size of a data packet). Starting with an unlimited windowsize (on the sender side), and a packetsize of 8 kBytes (of data), the maximum datarate was increased from 512 kBit/s to 2.4 Mbit/s:

![Graph showing the relationship between maximum datarate and real datarate](image)

The maximum throughput (2,070 kBit/s) was achieved with no packetloss at a maximum (sender’s) datarate of 2,300 kBit/s. With higher sending rates, packetloss occurred (due to buffer limits at the gateways to the satellite link) and decreased the real datarate on the receiver’s side.

As could be expected, the size of the sending window has a similar effect on the real throughput as the limitation of the sender’s bandwidth. A window of 30 packets was optimal for 8kByte datapackets if the sender’s throughput was higher than 2,300 kBit/s.

Using larger (16kBytes) or smaller (4kBytes) packets did not influence the throughput across the satellite link.

III.C. PC Implementation

Originally the MSP was written on the workstation. Therefore it had to be transported on the PC from C to Microsoft C++. Besides several syntax problems to be solved there were different sets of parameters necessary. For example the winsock interface can only deal with a window size equal to one packet. The maximum packet length could be set to a maximum of 8 Kbytes only.

Nevertheless, on workstation and PC we got a bit rate of around 1.2 Mbit per second taking the upper limits defined by the PC. In comparison to the results of the optimized parameter set of the unix protocol which handles bit rates of more than 2 Mbit per second, the winsock protocol needs to be optimized in forthcoming test phases.

The experiences with both platforms are strongly connected to the different properties given by the mpeg decoder cards of workstation and PC. Although there are more possibilities of changing the parameter set on the socket interface on the workstation, the PC has one advantage concerning the available hardware. The reelmagic card on the PC turned out to be more stable in decoding corrupted mpeg files, which can be the case when playing via satellite.

IV. Audio and Video Coding for IoD Services

Audio and Video for IoD Services were coded in MPEG1 (ISO/IEC 11172). This standard was mainly intended for CDs and other Digital Storage Media with reliable access and constant bit rate.

MPEG streams consist of I-, B- and P-frames, where only I-frames contain the full information for

![Figure 3: Satellite spectrum during Photokina](image)

1. 3.5 Mbit/s IoD data service
2. 128 kbit/s IoD message service
3. 1 Mbit/s Conference from St. Augustin
4. 2 Mbit/s Conference from SMPTE
5. 2 Mbit/s Conference from Moskow
recomputing the full image contents. If a B- or P-frame is hit when jumping back and forth within an MPEG stream, currently available MPEG hardware and software do not support the reconstruction of exactly the frame which has been hit. So the typical workaround is to search for a nearby I-frame and to display that frame instead. This is different to Motion-JPEG, where every single frame can be addressed and contains the full information.

On the PC client the MPEG decoder card dealt well with missing data, resulting in short flickers in the presentation. On the SPARC station we encountered more problems with lost packets or bit errors and this problem has not been totally solved yet.

If video doesn’t play smooth, it’s annoying but tolerable for human eyes. But if audio cannot be played with acceptable quality it should be skipped completely.

We experimented with various bit rates for video and audio and finally decided to choose a video bitrate of 1.4 Mbit/sec and 192 kbit/sec for audio. Higher bitrates didn’t provide better quality on our decoders, especially on the PC, due to hardware limitations.

One main difficulty was random access in the MPEG files, which we solved with an index file that contains the byte positions corresponding to time codes.

We offered the videos with multiple audio tracks (multilingual IoD), which caused additional problems for the PC client, because the decoder could (up to now) not handle the MPEG system streams that had several audio streams multiplexed in it. Thus we had to produce several system streams for the same video and then switch between the files when the user wanted to change the language. This again required exact positioning to allow continuous video playing.

As MPEG1 is mainly intended for access on CD or hard disk, the PC decoder offers as a standard interface the Microsoft MCI command set, that directly opens the MPEG file. This works well with locally stored or files accessible via NFS.

To use the satellite link with the MSP presented in the previous section, we had to write our own API using the developer kit of the MPEG decoder card. The card got its input from a buffer that was provided by the application and filled by the MSP.

We used the SUN platform as an alternative flexible testbed for experimenting.

Based on the application scenario, we can state our requirements on a MPEG-1 decoder as follows:

- video quality has to be good enough to attract non-computer scientists,
- VCR functions like „fast forward“, „seek frame“, „play from.. to“ have to be supported,
- decoding for standard MPEG-1 systems,
- realtime decoding for 25 frames/sec,
error tolerance to support lossy networks.

There are today some SW systems for MPEG playback available but they suffer from a set of limitations. None of them can decode a video stream with 25 frames/sec in realtime on present day workstations, all of them with the exception of the CMS system [4] offer the simplest VCR functions like PLAY/PAUSE/STOP only.

Unlike the PC where we had a standard „MPEG-1 system stream“ decoder, our own workstation decoder was based on a mix of HW and SW components. A SW module was used for the demultiplexing of MPEG-1 system streams to feed a HW video-only decoder1 and a SW based MPEG-1 Layer 2 PD audio decoder2.

The integration of separate video and audio decoders has turned out to be more difficult than expected. Components intended as separate programs react to severe errors mainly with program abort. They can therefore be used as a library only with reengineering the complete system. In addition, restart functionality has to be added to the video and audio engines to recover from transport errors and to support the seek functionality.

Error tolerance is the most critical part in our view, because neither the standards nor the current implementations provide the necessary support. The MPEG-1 standard was defined for storage on CDs with an error rate $<10^{-12}$, the MPEG-2 transport streams (ISO/IEC 111138-1) are intended for more lossy networks with random errors $<10^{-8}$, but they are of little help in high speed packet networks [1].

On the demonstration of the IoD Services the user could switch between local (NFS) and remote (satellite) access.

We emphasize that there still is a need for a standard interface for communicating with MPEG devices and multimedia servers. One step forward is done with the DSMCC (Digital Storage Medium Control and Command) protocol presented in the MPEG2 specification.

Decoders should offer the possibility to cooperate with network protocols suitable for specific situations like satellite link.

MPEG2 will provide so called Transport Streams for the delivery of video and audio data to take unreliable connections into account. But there are still doubts if this concept is sufficient for lossy network connections.

For the hierarchical model of MPEG2 we also envisage an „intelligent“ MediaServer and a suitable protocol, that should be able to drop less important information in critical traffic situations, i.e. the additional streams for high-quality video and audio are not sent and basic data, especially header information for the decoder, has higher priority.

V. Information Retrieval

Video on Demand projects have mostly been confined to the video rental business, restricting interactivity to the choice of the video to be rented and the VCR buttons interface. This kind of interactivity is relatively poor compared to the rich wealth of new capabilities offered by the emerging technologies of interactive multimedia. In order to exploit interactivity to a far broader extent, in our IoD project we have experimented with a hypermedia client interface for integrated access to text, images, and real-time video over the network.

The intended audience of the SoD interface has been the sports TV consumer familiar with PC like interfaces. Therefore the user input has been restricted to mere pointing and clicking, which could have been implemented on a touch screen, a TV remote control, or a set-top-box as well. No keyboard for text input is necessary at any time. For the same reason, no sophisticated text or concept retrieval interface has been included, which could be of interest to another audience as e.g. sports editors, or sports scientists. But the system has a sufficiently general design to include more sophisticated interfaces later.

The interface consists of a single window (mostly full screen), in which several subwindows and buttons are exchanged dynamically depending on the context. The central subwindow displays the hyper-text, the images and the video. When a video is played back, the corresponding VCR buttons appear below the video subwindow and disappear, when play back is finished. Beyond VCR functionality, digital video capabilities are exploited by a number of additional features: First, an additional slider allows for easy rewinding and positioning of video. Rewinding within a single video cutout, a complete video file, or within multiple video files adds additional complexity. Second, a slow motion button allows for playing back at slower speed. Third, start and end buttons allows for exact positioning of the video at a beginning or at the finish of a scene. Fourth, an audio button allows for choosing of the four different audio tracks presenting different languages and commentators. Other displayed objects on the screen are mouse sensitive as well: Clicking on the GMD logo, a hypermedia presentation of the GMD including its flagship video is started.

1. SUN’s BARNEY card
2. PD MPEG-1 audio decoder maplay (Thomas Bading, TU Berlin)
The initial screen introduces the user into the subject of the soccer world cup 1994 by displaying the names of the participating soccer teams and inviting the user to choose his favorite subject by clicking. The user can enter the hypermedia network via text links (alphabetical list of catch words), list of all images, and lists of short descriptions of all scenes contained in the chosen soccer game. All media are cross referencing between each other so that you arrive at video cutouts from texts, images and other video cutouts. According to our experience, the amount of cross referencing has to be well tuned to the contents: Too few links leads the user into dead ends, too many links overwhelms the user with an intractable information flood.

A click on any item on the screen leads to a mixed list of text, image, and video items, from which the user may select the item of choice. At any point he can follow the individual chain of references backward. This functionality is already known from other distributed multimedia browsers, e.g. WWW browsers, NCSA Mosaic or Netscape. Unfortunately, these browsers suffer some drawbacks which made them inadequate for our IoD project: First, video is not well integrated. Video files are copied over the network completely before the player starts. The video window on the screen is not integrated into the multimedia browsers. Second, the browsers have a fixed layout, and look-and-feel, and cannot be adapted to arbitrary corporate design. The browsers need a more general mechanism for layout and look-and-feel adaptation, which is available e.g. with SGML browsers via DTDs, or DSSSLs.

Of course, all multimedia information chunks in an IoD teleservice have to be collected, to be transformed, indexed, annotated or enriched in order to transform them into interactive hypermedia. In our IoD project emphasis has been laid upon abstraction and generality of this transformation process, which is briefly described for interactivity, text, images, and video in the following four paragraphs.

**Text:** All news agencies’ texts that entered WDR offices over the news agencies’ tickers during the soccer world cup ’94 have been collected by the WDR. Using standard text retrieval technology, irrelevant words have been eliminated from the indexing procedure and the texts have been turned into hypertexts with cross references between each other. Afterwards, final manual adaptation remained necessary for names: E.g. in the Italian team Roberto Baggio (often referred to as Baggio, R. Baggio, or Baggio R.) and Dino Baggio (often referred to as Baggio, D. Baggio, or Baggio, D.) played together and the granularity of automatic indexing algorithms turned out not to be well adjusted in some cases.

**Images:** High resolution Color JPEG pictures (ca. 1/2 Mbyte per image, resolution of 2048 x 846, 24 Bit True Color, which is good enough even for print media). On the available platforms the JPEG viewers turned out not to be fast enough (2 - 5 minutes start up time on the best currently available PC hardware and software). Therefore the JPEG images have been transformed into MPEG videos with one frame length, using the MPEG hardware for accelerating the image viewing process by a factor of more than 100.

**Videos:** Videos have been provided by WDR on Betacam tapes with four audio tracks (Italian TV (RAI), Original Audio, German TV, German Radio), one tape per halftime. Each tape has been digitized separately, once or more than once: Some MPEG decoder hardware could not play more than two audio tracks. Therefore, the different audio tracks had multiply to be encoded. One major goal of the IoD project has been to provide direct access to all the video clips, cuts and breaks: not only retrieval of complete videos, but direct access to video sections contained within larger video sequences as subparts. The enabling technology here is adequate video indexing. For the purpose of video indexing, a student of sport sciences skipped over all video tapes, associating time codes with short descriptors of the scene (like goal, red card, corner kick, direct free kick), name of primary actor (i.e., the player in focus, e.g. R. Baggio), and cross references (e.g. to related texts, images, or other video clips), thus building up a database, which became a central component of our system.

On the other hand, we did not choose a multimedia authoring system for two reasons: First, the available authoring systems at project begin did not support hypertexts, and hypertext generation from linear texts and from databases. Second, there is no adequate multimedia authoring tool that supports both platforms, Sun workstations and Pcs.

Creating databases has been in the past and will be in the future the most labour intensive part of work in multimedia service construction. Therefore, an important question is, where, and in which application field, indices could be generated automatically. The authors of this paper have found at least one such field: A “News on Demand” service can be generated fully automatically by deploying the videotext information supplied together with the video signal for the auditive unimpaired people. By attaching time stamps to the incoming videotext, a correlation can be established to the video timecode. Thereby, contents based retrieval of news video becomes manageable. In USA, and GB, law guarantees videotext for the auditive unimpaired people for every broadcasting. In Germany, only once a day the main news broadcasting is accompanied by
videotext for the audibly unimpaired people. So the sources, from which we can derive interactive multimedia, very much depends on the nation and the language.

The IoD project aimed at more than just multimedia authoring. In order to integrate different video decoder hardware on different platforms and to integrate the media service as underlying realtime protocol, we expected the programming effort be a substantial part of our work and decided to choose a general object-oriented software development environment rather than a mere multimedia authoring system. The project deadlines being rather tight, rapid prototyping environments were a must.

Several different platforms, PC with Reel Magic MPEG-1 decoder and Sun with a beta release of the Barney card, had to be supported. ParcPlace Visual Works 2.0 is available for all platforms and allows for platform independent code of the non platform specific part of the software, e.g. databases, object-oriented data models, and interfaces. This choice turned out to foster platform independent multimedia system development. Of course, the Smalltalk environment can ask for the platform on which it is currently running, so the program itself can switch between the platform dependencies seamlessly.

VI. Conclusions and Outlook

The most important lesson learned from the SoD project is that clear separation of the different system layers, in particular between interface design and underlying video system elements, is impossible with the current multimedia technology. Paraphrased in other terms, top-down design or bottom-up system construction do not meet, they do not come up with the same final product. For building multimedia systems it is an important advice to do both, top-down and bottom-up, simultaneously in an iterative way.

That’s why the ongoing standardization efforts for MMSS\(^1\) (like IMA\(^2\) or MMCF\(^3\)) encountered heavier difficulties than expected, and why open multimedia systems for heterogenous platforms are more difficult to realize (compared to communication platforms, e.g.).

From the user interface point of view the most desirable interface for operating a video player is the imitation of the well-known VCR interface, which is used nowadays in almost all living rooms at home: This interface consists of play, pause, stop, rewind and fast forward buttons. But MPEG is designed for normal video replay, not for rewind and fast forward, or even jumping back and forth.

On the other hand, the digital representation of video streams allows for other functionality not available on the VCR, e.g. random access, immediate access to certain timecodes without winding the tape (therefore without loss of time), display every second/third/... frame, transform the displayed image in some computable way (e.g. through a looking glass). Constructing a video player for MPEG bottom-up therefore often results in unfamiliar interfaces differing very much from the VCR buttons.

Another important point was that the existing protocol technology was not suitable for continuous media transmission over satellite, so that an appropriate new MediaService Protocol had to be designed. It became obvious that currently available implementations of protocols like TCP/IP were not suitable for multimedia communication across broadband satellite links. Specific features like long round-trip-times and weather-dependencies have to be considered.

The current implementation of MSP has proven to supersede the performance of a TCP service when used over a long fat pipe (i.e., a satellite link). The first goal was to get things running, the code will be subject to changes in terms of robustness (i.e., against unexpected messages, out of range parameters) and performance tuning. Experiments with the IoD teleservice lead to a number of requirements for additional protocol features, requiring additional features like Forward Error Correction, security mechanisms, multicast capabilities, prioritized data transmission and more.

Furthermore, the protocol needs to be compared to similar approaches (like L. Rowe’s „Continuous Media System”), and alternatives to the use of TCP and UDP need to be evaluated (XTP, RTP, ATM/AAL).

References

  Internet Draft
draft-hoffman-rtp-mpeg-encap-00.txt


[3] Michael R. Macedonia, Donald P. Brutzman: MBone provides Audio and Video Across the Internet
  IEEE COMPUTER April 1994


---

1. Multimedia System Services
2. Interactive Multimedia Association
3. Multimedia Collaboration Forum
Author Information

Karl Jonas joined the German National Research Center for Information Technologie (GMD) in 1994. His research focuses on high speed protocol issues, with special attention to high-bandwidth-long-delay communication networks for multimedia applications.

Helmut Jungblut joined GMD in 1972. His responsibilities and research interests focus on mobile and stationary satellite communication, SCPC-connectivity, and the datahighway between Germany and Russia.

Joachim Kaeber joined GMD in 1985; main activities are in systems/network management with some participation in multimedia application development.

Manfred Kaul joined GMD in 1987. His research focuses on multimedia databases, object-oriented frameworks for multimedia applications, and IoD.

Ina Müller worked at GMD on digital image processing and coding. She left the research center in spring 1995.

Horst Santo’s research focuses on multimedia application aspects within distributed information teleservice scenarios.

Josef Schäfer is head of GMD’s research group for „Multimedia Applications in Telecooperation“ (MAT).

Ralf Wegner studied physics in Bonn. He graduated in astrophysics at the Max-Planck Institute for Radioastronomie. Since September 1994 he joined the GMD-MAT project at Birlinghoven. His work addresses multimedia platforms as well as high speed protocols especially on personal computers.

Further information about the authors and their work in the Multimedia Applications in Telecooperation (MAT) group of GMD can be obtained via http://mats.gmd.de/.