

Project Number: **Telematics for Research 4007**  
Project Title: **Multimedia Education and Conferencing Collaboration  
over ATM Networks and Others (MECCANO)**

Title of Deliverable: **Support for network technologies in the MECCANO tools  
and performance characteristics of the MECCANO systems  
and their effect on system parameters**  
Deliverable ID: **R5.1/5.2**  
Produced by Workpackage: **5**  
Contractual Date of Delivery: **March 31, 2000**  
Editor: **Colin Perkins**  
Contributors: **Roy Bennett, Kristian Hasler, Peter Kirstein, Jörg Ott,  
Andreas Rozek, Radoslaw Ruchala, Thierry Turletti**

### **Abstract:**

The aim of the MECCANO project was to provide a robust networked multimedia education and conferencing environment, suitable for use over a wide range of network technologies. To cope with the heterogeneity this implies, the Internet Protocol (IP) was adopted as a unifying platform over a range of layer 2 technologies (Ethernet, ATM, direct broadcast satellite, ISDN). Where it is not possible to use native IP, a number of gateways have been developed to other network architectures.

This report describes the underlying network technologies supported and the way in which the tools have been developed to adapt to these technologies. It discusses the performance required of an IP network for the successful use of the MECCANO tools and then reports on the observed performance in the context of IPv4 (unicast and multicast) and IPv6.

Finally the report describes the tool support provided for non-IP networks, such as ISDN, ATM and DBS, and the performance of such networks in the context of MECCANO.

### **Keyword list:**

Multimedia Conferencing, adaptation to network performance, IP-Multicast, IPv6, ATM, DBS, ISDN, PSTN.

# Contents

1	Introduction.....	4
2	Relation to other deliverables .....	4
3	Supported network technologies.....	4
3.1	Internet Protocol .....	5
3.2	ISDN/Voice telephony networks .....	5
3.3	Asynchronous Transfer Mode .....	6
3.4	Direct Broadcast Satellite .....	6
4	Tool support for IP networks .....	6
4.1	Use of IPv4 .....	6
4.1.1	Adaptation to packet loss .....	7
4.1.2	Adaptation to network jitter .....	10
4.2	Use of IP multicast.....	10
4.2.1	Support for scalability.....	11
4.2.2	Support for heterogeneity .....	11
4.2.3	Interworking with non-multicast capable networks.....	11
4.2.4	Visualization of multicast loss characteristics: RQM.....	13
4.2.5	Visualization of multicast loss characteristics: MeshTrace .....	14
4.3	Use of IPv6 .....	14
4.3.1	Support for IPv6 in audio/video tools.....	15
4.3.2	Support for IPv6 in shared workspace tools .....	15
4.3.3	Support for IPv6 in session announcement tools.....	16
4.4	Use of IPsec .....	16
4.4.1	Introduction.....	16
4.4.2	Problems with IP Multicast and IPsec .....	17
4.4.3	Problems with the use of relays .....	17
4.4.4	Problems in the use of IPsec with servers.....	18
5	Required IP network performance .....	18
6	Observed IP network performance .....	19
6.1	IPv4 Multicast.....	19
6.1.1	Overloaded Routers, Links and Tunnels.....	20
6.1.2	Routing Loops and Flapping Routes .....	21
6.1.3	Limitations of RQM and MeshTrace.....	24

6.1.4	Conclusion .....	25
6.2	IPv4 Unicast.....	26
6.3	IPv6.....	26
7	Tool support for non-IP networks.....	26
7.1	ISDN/Voice telephony networks .....	27
7.2	Asynchronous Transfer Mode .....	27
7.3	Direct Broadcast Satellite .....	28
7.3.1	Standards activities .....	28
7.3.2	Validation activities .....	28
7.3.3	Driver development .....	28
7.3.4	Deployment of services .....	28
8	Performance of non-IP networks .....	29
8.1	ISDN/Voice telephony networks .....	29
8.2	Asynchronous Transfer Mode .....	29
8.2.1	Introduction.....	29
8.2.2	The alpha test procedure .....	30
8.2.3	Experience .....	30
8.2.4	Conclusion .....	32
8.3	Direct Broadcast Satellite .....	32
9	Conclusions.....	32
	References.....	33

## Figures

Figure 1:	Repair using redundant audio.....	8
Figure 2:	The interleaving process.....	9
Figure 3:	RTP Reception Quality Matrix .....	13
Figure 4:	MECCANO Network Status on December 14th, 1998.....	20
Figure 5:	MECCANO Network Status on January 18th, 1999.....	21
Figure 6:	MECCANO Network Status on November 29th, 1999 at 15:10 .....	22
Figure 7:	MECCANO Network Status on November 29th, 1999 at 15:20 .....	23
Figure 8:	MECCANO Network Status on November 29th, 1999 at 15:30 .....	24
Figure 9:	Number of successful Path Traces after three Trials.....	25

# 1 Introduction

This document is the combination of MECCANO deliverables R5.1, *Support for network technologies in the MECCANO toolset* and R5.2, *Performance characteristics of the MECCANO systems and their effect on system parameters*. The two have been combined due to the large degree of overlap between them. The deliverable is available on the MECCANO web site [MECWEB] under *Deliverables*.

The aim of the MECCANO project was to provide a robust networked multimedia education and conferencing environment, suitable for use over a wide range of network technologies. To cope with the heterogeneity this implies, we adopted the Internet Protocol (IP) as a unifying platform over a range of layer 2 technologies (Ethernet, ATM, direct broadcast satellite, ISDN). As a consequence of this, the majority of our tools work over IP. Where it is not possible to use native IP, we have implemented a number of gateways to other network architectures.

## 2 Relation to other deliverables

The architecture adopted to achieve our aims is described in deliverable R3.1, *The MECCANO Internet Multimedia Conferencing Architecture*.

We discuss the MECCANO toolset in three deliverables, including this one, in each case in a specific context. We are currently releasing fourteen tools which are described in deliverable D4.3, *Second major release of MECCANO tools*, in a standard format, including release notes, platforms supported and installation instructions. We preface this deliverable with an assessment of the complete set of all the tools we have delivered during the project.

In the deliverable D6.2/7.2, *Session announcement, management facilities, gateways and relays in MECCANO Release 2*, we describe and discuss those tools which either form the gateways and relays or which implement the work on session announcement and conference control.

In this deliverable we address the ability of the tools to operate in the many network environments which we support, emphasising the features which suit these environments. In the *best-effort* IP environment, we note the adaptations to reduce the effects of network losses. With the need to join in a single conference users with widely differing network bandwidth, we discuss the features which allow such conferences to proceed without diminishing the quality to that available to the least well provided user. We discuss also the ability to interface with non-IP networks using gateways and the tool support for IPv6.

## 3 Supported network technologies

The main focus of network activities in the MECCANO project has been IP networks such as the public Internet, TEN-155 European backbone network, and the national research networks.

Other network technologies such as ATM, ISDN/voice telephony networks and direct broadcast satellite have been used also, primarily as bearers for IP traffic but also in native mode on some occasions. One of the primary focus points of MECCANO has been the convergence of a range of network technologies, with IP acting as a glue between traditionally disparate worlds. This can be seen in our choice of the networks supported.

In this section we discuss the role of IP and its bearer technologies within the MECCANO project. This sets the scene for later discussion of the toolset support for those networks, and their performance.

### **3.1 Internet Protocol**

The primary network technology employed in the MECCANO project is IP. We chose to use IP as our base technology because of the widespread deployment of Internet technologies, and because of its suitability as a glue protocol, combining several lower-layer technologies into a uniform service platform (the so-called “hourglass model”).

The basic transport provided by IP is unreliable, best-effort, packet delivery. Higher-level protocols provide for flow control and error recovery, for example TCP for reliable unicast transport and RTP for real-time flows.

Several enhanced services are available, in addition to the base IPv4 transport, these include:

- IP multicast, provides support for group communication by allowing packets to be delivered to a group of receivers in an efficient manner.
- IP security, which allows for packets to be authenticated and/or encrypted for private communication
- Differentiated and/or integrated services which provide some degree of reliability and quality assurance

Finally, IPv6 bundles all those additions to the base IPv4 model together, and provides for an extended address space.

The majority of the MECCANO work has been devoted to support of the basic IPv4 service, since that is what is used in the majority of the Internet. However, we have also devoted considerable effort into support for IP multicast, and have prototype implementations which utilize the other advanced features offered.

### **3.2 ISDN/Voice telephony networks**

As noted previously, one of the aims of MECCANO is to provide a converged conferencing environment, based around Internet standard protocols. This implies that gateways must be built, to provide access to the converged network from legacy systems such as the PTSN and ISDN based conferencing.

Our use of ISDN/voice telephony networks within MECCANO is therefore two fold:

- as a bearer for IP traffic
- as a native transport, with a gateway into the IP conferencing environment

The first of these requires little elaboration, since it uses standard protocols (e.g. PPP over ISDN/voice modems) to hide the underlying network. It may require limited gateway support, for example when the IP interface transported by the ISDN/PSTN link does not support IP multicast - this is discussed in Section 4.2.3.

Of more interest is the support provided for gatewaying between IP and the ISDN/voice telephony world. There are two aspects to this: support for plain voice calls, and support for H.320 multimedia sessions.

The previous MERCI project included development of a component to gateway H.320 based ISDN conferences into the H.323 based IP world. We have not, therefore, studied such gatewaying in MECCANO, focussing instead on the conversion between H.323 and other IP-based conferencing systems with our StarGate system.

The preliminary AudioGate system shows that the simpler case – gatewaying an ISDN voice call into IP - can be completed, validating the approach and paving the way for a complete solution. It also shows that the large legacy base of ISDN/POTS phones which we need to gateway into Mbone/H.323 conferences can be accommodated within the MECCANO framework.

### **3.3 Asynchronous Transfer Mode**

Asynchronous Transfer Mode (ATM) provides a high-capacity cell switched environment, which is commonly used to engineer wide area networks (e.g. the TEN-155 European research network runs IP over an ATM backbone). It also provides for native voice and video transport.

We have primarily used ATM as a bearer technology for IP, in particular to provide QoS. Examples include the trial over the TEN-155 MBS and testing on the London East Anglia Research Network (LEARNet). We have also used native ATM, to a limited extent, with a gateway to IP; the robust-audio tool includes a gateway to ATM networks in much the same manner as its ISDN gateway.

### **3.4 Direct Broadcast Satellite**

Direct broadcast satellite provides an effective means of distributing digital data, in the form of an MPEG transport stream, to a very large area. The MECCANO project has continued work begun previously to use such a network as a transport for IP data, allowing multicast-based lectures and seminars to be distributed to very large audiences, whilst still giving the benefits of a converged network.

## **4 Tool support for IP networks**

This section describes the features of the tools which have been implemented specifically to support IP networks.

### **4.1 Use of IPv4**

The basic environment assumed by the MECCANO toolset is a packet network offering support for IP version 4. Examples include the public Internet, the TEN-155 European research network backbone, and the various national research networks.

Networks such as these have unique channel characteristics, due to their shared infrastructure and best effort forwarding. In particular, we note that the following problems affect transport of multimedia traffic:

- Packets may be lost
- Packets may be duplicated
- Packets may be reordered in transit
- Packets will be fragmented if they exceed the MTU of the underlying network

Although fragmentation is not a disastrous phenomenon if it is a rare occurrence, relying on IP fragmentation is a bad design strategy; it significantly increases the effective loss rate of a network and decreases goodput. This is because if one fragment is lost, the remaining fragments (which have used up bottleneck bandwidth) will then need to be discarded by the receiver. It also puts additional load on the routers performing fragmentation and on the end-systems re-assembling the fragments.

In addition, the transit time between two hosts on an IP network will not be constant. This is due to two effects - jitter caused by being queued behind cross-traffic, and routing changes. The former is possible to characterise and compensate for by using a playout buffer, as discussed in Section 4.1.2, but the latter is impossible to predict and difficult to accommodate gracefully.

The MECCANO project aims to provide support for conferencing on nodes which have a reasonable connection to the Internet, and where the Internet has reasonable performance. The key word here is reasonable: some degree of media degradation due to poor network channel characteristics is inevitable, the aim of the support provided for IP networks is to minimise that degradation.

#### **4.1.1 Adaptation to packet loss**

Much research on Internet audio and video applications has focused on issues arising from the best-effort nature of network, such as packet loss concealment and forward error correction (see [Perkins98] for a recent survey by MECCANO partners).

If it is desired to repair a media stream subject to packet loss, it is useful to have some knowledge of the loss characteristics which are likely to be encountered. A number of studies have been conducted on the loss characteristics of the Mbone [Bolot00, Handley97, Yajnik96] and although the results vary somewhat, the broad conclusion is clear: in a large conference it is inevitable that some receivers will experience packet loss. Packet traces taken by Handley [Handley97] show a session in which most receivers experience loss in the range 2-5%, with a somewhat smaller number seeing significantly higher loss rates. Other studies have presented broadly similar results.

It has also been shown that the vast majority of losses are of single packets. Burst losses of two or more packets are around an order of magnitude less frequent than single packet loss, although they do occur more often than would be expected from a purely random process. Longer burst losses (of the order of tens of packets) occur infrequently. These results are consistent with a network where small amounts of transient congestion cause the majority of packet loss. In a few cases, a network link is found to be severely overloaded, and large amount of loss results.

The primary focus of a repair scheme must, therefore, be to correct single packet loss, since this is by far the most frequent occurrence. It is desirable that losses of a relatively small number of consecutive packets may also be repaired, since such losses represent a small but noticeable fraction of observed losses. The correction of large bursts of loss is of considerably less importance.

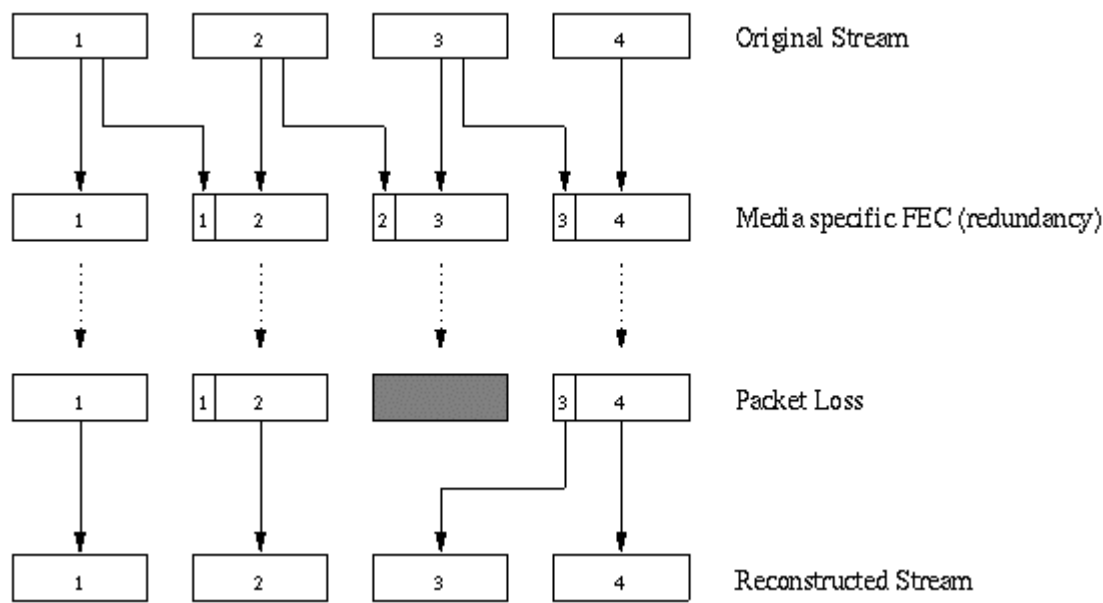


Figure 1: Repair using redundant audio

A simple means to protect against packet loss is to transmit each unit of audio in multiple packets. If a packet is lost then another packet containing the same unit will be able to cover the loss. The principle is illustrated in Fig. 1. This approach has been advocated by Hardman et al [Hardman95] and Bolot et al [Bolot00] as part of the predecessor MERCI project and extensively simulated by Podolsky et al [Podolsky98].

The first transmitted copy of the audio data is referred to as the primary encoding and subsequent transmissions as secondary encodings. It is the sender's decision whether the secondary audio encodings should be the same coding scheme as the primary, although usually the secondary is encoded using a lower-bandwidth, lower-quality, encoding than the primary.

The choice of encodings is a difficult problem and depends on both the bandwidth requirements and the computational complexity of the encodings.

Erdöl *et al.* [Erdöl93] consider using short term energy and zero crossing measurements as their secondary scheme. When loss occurs the receiver then interpolates an audio signal about the crossings using the short-term energy measurements. The advantage of this scheme is that it uses computationally cheap measures and can be coded compactly. However, it can only cover short periods of loss due to the crude nature of the measures. Hardman *et al.* [Hardman95] and Bolot *et al.* [Bolot00] advocate the use of low bit-rate analysis-by-synthesis codecs, such as LPC (2.4-5.6kbps) and GSM (13.2kbps); although these are computationally more demanding, they can tolerably cover the loss periods experienced on the Internet.

If large end-to-end delay can be tolerated, it is possible to delay the redundant copy of a packet, achieving improved performance in the presence of burst losses. This has been studied by project partners, and implemented in the RAT and FreePhone audio tools [Kouvelas97].

We have designed also an adaptive FEC-based error-control algorithm [Bolot99] which provides very good performance with the “signal processing” FEC scheme for audio recently standardized in the Internet Engineering Task Force (IETF) [RFC2198]. Of course, even this “optimal” scheme cannot provide guaranteed quality given the best effort service model of the current Internet. However, it puts us one step closer to quasi-constant quality audio even over connections with high or highly varying loss rates. The algorithm has interesting features:

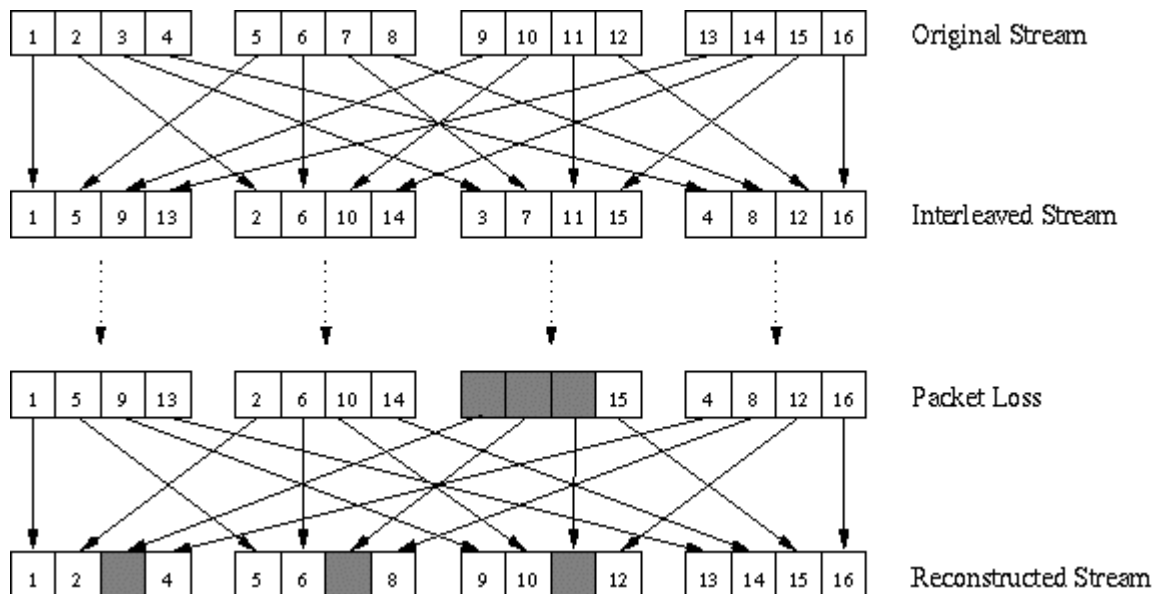
- i) it optimizes a subjective measure of quality (such as the perceived audio quality at a destination) as opposed to a non-subjective measure (such as the packet loss rate at a destination),
- ii) it incorporates the constraints of rate control and playout delay adjustment schemes, and
- iii) it adapts to varying (and estimated on line with RTCP) loss conditions in the network.

The standard RTP payload format for redundant audio [RFC2198] was developed by project partners in the predecessor project. During the course of MECCANO this has been adopted for a number of additional uses, besides audio conferencing [RFC2733], showing the wide applicability of the technique.

In addition to work on redundant audio transmission, project partners have been active in the development of robust video transport, working on the RTP payload format for H.263+ video [RFC2429] which includes many provisions for robust transport in the presence of packet loss.

Unfortunately, all these error protection schemes result in a bandwidth increase for the streams (due to the extra protection information carried). This increase in bandwidth may be unacceptable in some cases, for example, low-speed dialup or wireless links may not be able to support the bandwidth requirements of such a stream. Interleaving provides an effective means by which audio streams may be protected which trades latency, rather than bandwidth, for such protection.

An interleaver is a device which permutes the order of a sequence of symbols. The corresponding device which restores the original order of the symbols is a deinterleaver. An interleaver is employed in a transmission system when it is desired to randomize the distribution of errors after reception: a burst of loss on the channel is transformed into a sequence of isolated losses by the interleaving process. This is illustrated in Fig. 2.



**Figure 2: The interleaving process**

It can be seen that a burst of consecutive loss in an interleaved stream will result in multiple small gaps in the reconstructed stream. This spreading of the loss is important for two similar reasons: firstly, packet voice applications typically transmit packets which are similar in length to phonemes in human speech. Loss of a single packet will therefore have a large effect on the intelligibility of speech. If the loss is spread out so that small parts of several phonemes are lost, it becomes easier for listeners mentally to patch-over this loss [Miller50] resulting in improved

perceived quality for a given loss rate. In a somewhat similar manner, error concealment techniques perform significantly better with small gaps, since the amount of change in the signal's characteristics is likely to be smaller.

The majority of speech and audio coding schemes can have their output interleaved. Provided the channel exhibits bursts of loss, rather than isolated loss events, interleaving provides an effective means by which a media stream may be protected. The disadvantage of interleaving is that it increases latency. The major advantage of interleaving compared to other means of protecting media streams is that it does not increase the bandwidth requirements of a stream.

In the MECCANO project we have implemented interleaving in the experimental releases of the Robust-Audio Tool. Experience has shown that it performs well for non-interactive applications, where the additional latency occurred is unimportant.

We have also studied the interactions between interleaving and RTP/UDP/IP header compression [Perkins00].

A final outcome of our work on adaptation to packet loss the documentation of options for repair of streaming media [RFC2354] and a series of and guidelines for authors of RTP payload format specification, recently published as “best current practice” in the IETF [RFC2736].

#### **4.1.2 Adaptation to network jitter**

Another area where considerable effort has been expended is in adaptation to network jitter and audio sampling clock skew. Existing work has studied the calculation of suitable playout points in the absence of synchronization mechanisms [Moon98, Ramjee94] and on the detection of clock skew between hosts [Moon99, Paxson98]. The work undertaken in MECCANO is the first work we are aware of that addresses the issues of transparent skew adjustment for network audio applications.

We have observed that, when sampling an audio signal for digital transmission, the sample clock will differ from its nominal rate due to variations in the quartz crystal oscillators and the components that regulate their frequency. During the implementation of the Robust-Audio Tool [RAT-4] for workstations and personal computers we observed a 0.5% variation between nominally similar clocks. This has undesirable effects, since when the sender's clock is faster than the receiver's, samples will accumulate; consuming memory and increasing the delay. As the memory available for the audio buffer is exhausted interruptions occur as frames are dropped from stream. Conversely, when the sender's clock is slower than that of the receiver, audio played out at the receiver becomes interrupted as the playout buffer runs dry.

Further details of this work are presented in [Hodson00].

## **4.2 Use of IP multicast**

The changes needed to enable IP multicast support in an application which supports unicast IP are, at one level, trivial: senders must choose an address from the multicast range as the destination for UDP packets they transmit, but require no other changes. Receivers must use the same address, and perform a *multicast join* on creating the receive socket, in addition to normal UDP operation. This amounts to one extra system call on typical systems.

On a subtler layer, the introduction of multicast has significant consequences for designers of networked multimedia applications, primarily relating to heterogeneity and scalability.

We note also that all the features needed for robust operation with unicast IP, as discussed previously, apply also to multicast.

### **4.2.1 Support for scalability**

The basic problems relating to scalability of multicast applications are now well understood and relate to avoidance of feedback implosion and limitation of control traffic. The toolset we provide implements the standard algorithms in this area.

Recent work relating to scalability of audio/video transport via RTP has introduced a timer reconsideration algorithm [Rosenberg] which has also been implemented in our toolset, enhancing the scalability of the tools to very large sessions.

### **4.2.2 Support for heterogeneity**

The Internet is very heterogeneous, with link speeds ranging from several Kbps up to several Gbps, and very varied levels of congestion. How can a single multicast source satisfy a large and heterogeneous set of receivers?

If the sender layers its media stream, different receivers can choose to receive different amounts of traffic and hence different qualities. To do this the sender must code the media as a base layer (the lowest quality that might be acceptable) and a number of enhancement layers, each of which adds more quality at the expense of more bandwidth. Each layer is sent to a different multicast group, and receivers can decide individually how many layers to subscribe to, by joining and leaving multicast groups.

During the course of the MECCANO project we have integrated support for layered coding into the media tools RAT and Vic.

Support for layered audio in RAT comprises a layered “wide-band speech” ADPCM codec, sending 16kHz sampled audio as a 64kbps stream or as two layers: a 48kbps base layer with a 16kbps enhancement layer. The framework is general, so other layered codecs can be incorporated as they become available.

Support for layered video in Vic comprises the layered PVH codec originally developed at UC Berkeley for the MASH project, and ported to Vic as part of the integration work in MECCANO. The PVH codec supports an arbitrary number of layers at a range of data rates.

Our implementation of layered audio and video provides a sound basis for future experimentation and is useful in its own right at present. The current mode of operation of layered coding in the MECCANO toolset is with manual control of the receiver subscription to multiple layers, allowing for adaptation between high quality within an intranet and low quality wide area, from the same source.

Further work is required in order to design and implement automatic control of layering, for receiver-driven congestion control. Members of the MECCANO project are active in the IETF and Internet Research Task Force (IRTF) groups studying this problem, but there is no well-understood standard control mechanism as yet.

### **4.2.3 Interworking with non-multicast capable networks**

Whilst the portion of the Internet which is multicast enabled is large and growing, there still exist regions where multicast is not supported. It is necessary to allow users in these parts of the network to join conferences which are ongoing in the multicast capable region.

Furthermore, it is possible that regions which are nominally multicast enabled are found to provide insufficient quality of service, whereas the unicast quality is excellent. This can happen, for example, due to misconfiguration in routers or due to defined bandwidth limits for multicast traffic. A multicast-to-unicast gateway is useful in these cases.

In view of the situation described above (poor international multicast connectivity but excellent unicast links) the MECCANO partners had to consider the use of a packet reflector in order to still be able to hold seminars and electronic meetings. Such a tool receives incoming packets and retransmits them to a list of registered unicast and/or multicast addresses. In contrast to a (transcoding) 'gateway' a reflector does not touch the traffic, e.g. for the purpose of bandwidth economisation, it just distributes the packets among the participants of a session.

In the beginning, several instances (one per media stream) of Henning Schulzrinne's "rtpttrans" tool were started at one site which distributed incoming RTP and RTCP packets between a fixed multicast group and an explicitly given set of unicast addresses. While performing quite well, it turned out to be rather uncomfortable having to supply the IP addresses of each unicast terminal manually - not taking into consideration, that the program had to be stopped and restarted with new addresses whenever a new site expressed its interest in using the reflector.

As a consequence, it was decided to write a reflector which should be able to manage its list of unicast peers automatically. Fortunately, this plan turned out to be relatively easy as even "silent" RTP applications (i.e. those which "listen" only to incoming traffic) periodically emit RTCP packets. The program "Mtransit" that resulted from that activity therefore just listens to all incoming packets watching out for new senders which it then automatically adds to an internal distribution list - and removes again after a several minutes of inactivity. It was written in Java when a brief study showed that both a PC and a SPARCstation were able to handle the traffic of a typical MECCANO session. (Similar tools - "rug" and "mug" - have been written in C by Julian Highfield. They use less memory and processor capacity than "Mtransit", but cannot be used on PCs running under Windows)

Although the strategy of packet reflection imposes a relatively high traffic load on the site running the reflector and does not scale well, it provides a simple and robust way to include (a limited number of) sites with sufficient network capacity (i.e. without the need for transcoding and mixing) in multicast sessions or conferences.

A similar approach was adopted in the development of the UiO reflector system. This is composed of a data forwarding application (Reflector) and a separate control unit. An RTSP-based control unit has been developed and tested as a part of the MECCANO project; an H.323 controller is currently under development.

Reflector associates a multicast address/port with a set of unicast addresses/ports and forwards data packets received from any of these addresses to all others. Several such mappings can coexist with many multicast channels being mapped simultaneously. The number of simultaneous channels is limited by network adapter performance. The reflector is externally configured using a simple, TCP-based, control protocol known as Reflector Session Protocol (RSP).

The RTSP controller (RTSPC) is an RTSP server that can operate as an RSP client to configure and control one or more reflectors. RTSPC acts as a bridge between the WWW-based announcement of multicast sessions and the end-user multimedia client tools.

The general architecture of the UiO reflector is described in deliverable R3.1. Current implementations of the RTSP client and the server (MURS) are described in deliverable D4.3. Further details of these implementations and a description of current H.323 activities are given in D6.2/7.2.

An alternative approach is that of the UTG gateway, developed at UCL. Unlike the other two interworking devices, the UTG is not simply a forwarder between different networks; it can also transcode media streams to reduce their bandwidth requirements. This allows effective interworking with users on bandwidth-constrained networks (e.g. mobile) whilst not limiting the

quality for those on higher capacity systems. More details of the UTG are presented in deliverable D6.2/7.2.

#### 4.2.4 Visualization of multicast loss characteristics: RQM

The Real-time Transport Protocol, RTP, provides quality of service feedback with reception reports sent alongside the media stream. If the media is sent via IP multicast it is possible for a third party to snoop on these reception reports, displaying reception quality for all members of a group. The RQM application has been developed to perform such monitoring (a simplified version of RQM is integrated into the robust-audio tool), allowing easy detection of network problems. The distributed binaries support IPv4 addresses; it is also possible to compile RQM to support IPv6.

When running RQM displays a matrix with participant details on the left, and a number of cells to the right of these. Each row of cells denotes the packet loss rate observed for data sent from the participant indicated at the left of that row (point to a cell and a popup will appear giving the names of the source and destination of the traffic represented by that cell).

The colours of the cell start as green (no loss) and fade to red (20 % loss). A white cell indicates that no information is available. A light blue cell indicates that the receiver is not receiving media data from a particular sender (at present light blue is only used when an empty reception report is received, indicating that a receiver can hear no-one). Clicking on a cell will initiate a *multicast traceroute* between participants, if mtrace is installed and available on the system.

A sample screen-shot of the RQM application is provided in Fig. 3.

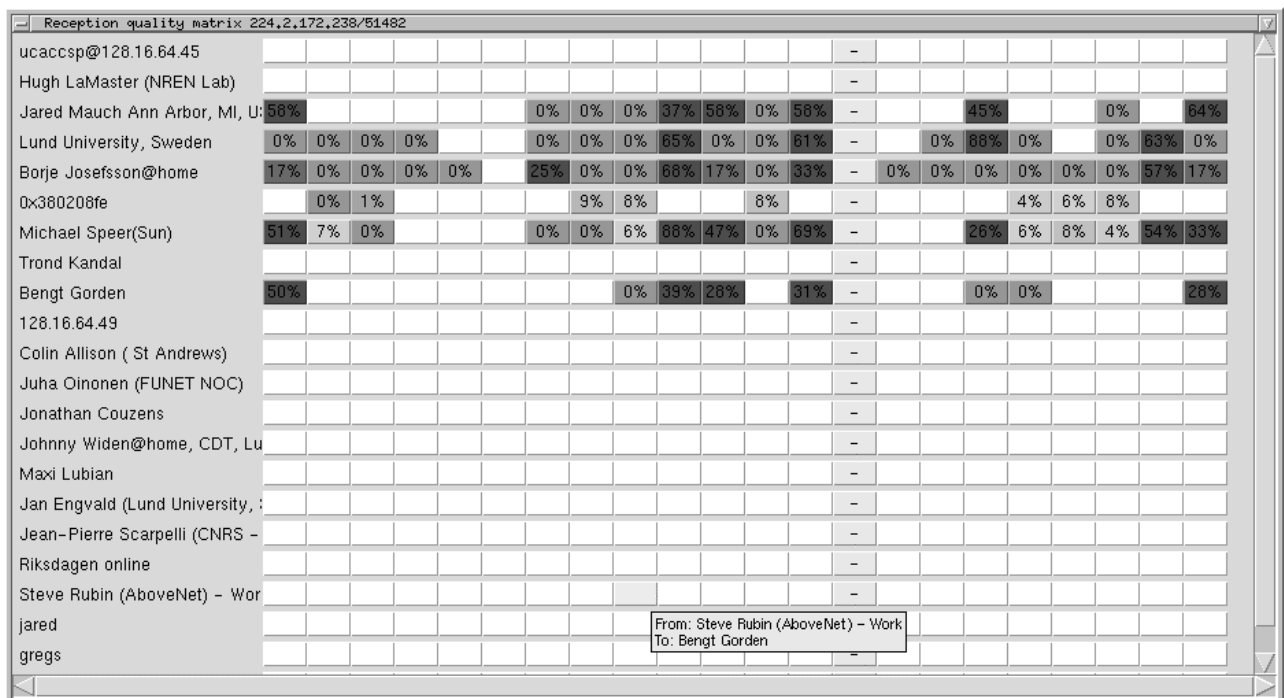


Figure 3: RTP Reception Quality Matrix

The RQM application allows “at-a-glance” viewing of a multicast session, showing clearly those receivers who have poor reception quality, even by those with little expert knowledge. The inclusion of support for multicast traceroute allows a skilled operator to determine the exact location of the fault.

#### 4.2.5 Visualization of multicast loss characteristics: MeshTrace

Driven by frustration with the poor quality of MECCANO conferences on the (pan-European) multicast network, we carried out experiments to discover how existing commands like mtrace and minfo might help.

These experiments led to the development of MeshTrace, a tool which simply runs multicast traceroute (mtrace) between all possible pairings of a given set of session participants. The output of these mtrace invocations contains all multicast routers on the distribution path between one of the given sites and the other and gathers information on packet losses and duplications reported by these routers. Obtaining all these mtrace results in a short time span creates an almost instantaneous overview of the multicast connectivity between the observed sites. By combining the measurements for routers and links shared by several distribution paths, maps can be drawn which visualise the multicast network status at the time of a MeshTrace run (see section 6.1 for a number of examples). Given one or more of such maps, drawn for a series of MeshTrace observations and the underlying measurement figures, it is relatively easy to locate the source of transmission problems and to contact the service provider responsible for that router or network.

The combination of the reception quality matrix, RQM, and the meshtrace scripts provides a powerful system for the detection and location of problems with the IP multicast infrastructure. The reception quality matrix shows at-a-glance the reception quality for all participants in a session, and allows for limited problem tracing. The meshtrace scripts provide a more in-depth view of the network, allowing accurate fault tracing.

### 4.3 Use of IPv6

During the course of the MECCANO project a number of our tools have been adapted to support IPv6, in addition to IPv4. There are a number of steps to providing such support:

- The operating system has to provide an IPv6 protocol stack;
- The low level socket code in the application has to be written in a protocol independent manner, to support both IPv4 and IPv6;
- The high level application protocol has to be address agnostic, and operate irrespective of the underlying transport mechanism

Operating system support for IPv6 has improved greatly during the project; we now have IPv6 support in Red Hat Linux 6.2 (other Linux distributions may support this also, but have not been tested in the project), FreeBSD 3.x with KAME, FreeBSD 4.0, Solaris 8, Windows NT 4.0 and Windows 2000.

It is straight forward to convert the low-level socket interface code written to support IPv4 into protocol independent code which can support both IPv4 and IPv6 and the revised interface is defined in [RFC2553]. The main problem we encountered during the ports was inconsistent implementation of the standard API by various operating systems, since both the API and its implementations were under development at the time we were working. At the time of writing, the majority of systems have very good support for IPv6<sup>1</sup> and the APIs have now been stable for over a year, so this is unlikely to affect future development.

In the next sections, we discuss the high-level protocol changes needed for particular applications.

---

<sup>1</sup> The exception being the Thompson Detexis MUSICA stack, which lags significantly in its support of the standard API

### 4.3.1 Support for IPv6 in audio/video tools

The Real-time Transport Protocol, RTP, was designed from the start to be independent of the underlying transport. As such, it can operate equally well over IPv4, IPv6 and ATM. The reason for this independence is the omission of transport level addresses from the protocol, and their replacement by a session level SSRC identifier<sup>2</sup>.

The audio tool, RAT, and video tool, Vic, have been ported to use IPv6. In both cases, all that was needed was to update the low-level socket code and update command line parsing routines, to recognize IPv6 addresses. No changes were needed to the RTP stack in either application.

### 4.3.2 Support for IPv6 in shared workspace tools

Support for IPv6 was concentrated on the shared text editor NTE since there was full access to the source code and an understanding of its implementation. Unlike the video and audio tools, NTE had its own application protocol that was dependent of the IPv4 stack. Adding support for IPv6 involved changes to NTE's protocol that made it incompatible with previous versions. Two options were considered. The first was to simply replace version 4 addresses with version 6 addresses, thus making a special NTE compatible only with IPv6 hosts. This was considered inflexible for our needs as we required interoperability between hosts using different IP stacks. This temporary change was also considered not to be useful for the continuing development of the protocol. The second option was to remove the reliance on IP addresses from the protocol and to create an IP-independent protocol, similar to the Real Time Protocol (RTP) used in the audio and video tools, RAT and Vic. This would allow NTE to be used on any stack and also to interoperate with different stacks.

The NTE protocol version was changed to version two; as stated above, this is incompatible with previous versions. IPv4 addresses were used extensively within the protocol for two different uses. The main use was for the address to make up part of a 64bit unique identifier for block, line and pointer messages. The 128bit IPv6 address was not suitable for the existing identifier length and using an IP address did not guarantee a unique identifier. For example a host with multiple instances of NTE would have the same identifiers and obvious collision problems. Making the identifier protocol independent was the preferred option and the address part of the identifier was replaced with a 32bit random but unique source identifier. Allocation and collision detection of the identifier is described below.

IPv4 addresses were used also in the retransmission protocol to indicate the source of the requesting host, with the source address derived from the packet header of a retransmission request message. This made retransmission messages incompatible with packet relays where the source address of the packet was that of the relay and not of the requesting NTE. In addition, retransmission messages were not directed towards specific instances of NTE; this resulted in all instances running on a single host processing each other's retransmission messages. Again the choice was made to replace the address with the unique identifier described above. Since the IP address is no longer used, request messages are multicast to the group, rather than unicast to the source. Each receiver then filters the messages before processing them.

The unique source identifier is allocated upon start-up, consisting of a 32bit random number. The identifier is required to be unique for the current session. It is essential that this number is kept unique, as a duplicate identifier could overwrite previously created data and corrupt the

---

<sup>2</sup> RTP uses an IPv4 address in the CNAME if running over IPv4, but this can be replaced by any unique persistent identifier without affecting protocol operation. Implementations which run over IPv6 use an IPv6 address, for example.

internal structures holding such data. As the changes propagate between each NTE, this will eventually affect all hosts and make the session unusable. Although the chances of collision are remote the effect is undesirable and so collision detection and resolution of the source identifier has been implemented.

The solution is similar to the implementation of collision detection and source allocation used in the RTP protocol. If a source receives a message with the same identifier, then it regenerates its own identifier and sends a new *sessionleave* message for the old identifier. This informs other NTEs that may be using this source for retransmissions to renegotiate a new source. If a source receives two messages with the same unique identifier but different hostnames, the source discards all packets from those sources until a *sessionleave* message from one of them has been received. This prevents the processing of messages that could possibly corrupt the session.

Further changes to the NTE protocol were made and have been summarised in deliverable D4.3.

### **4.3.3 Support for IPv6 in session announcement tools**

The session advertisement protocol (SAP) used in the session directory (SDR) tool has been modified to accommodate IPv6 addresses. This required increasing the size of the SAP header for IPv6-only announcements, announced on an IPv6 SAP address. IPv4 announcements are not affected by this change and IPv4 announcements from SDR remain compatible with older IPv4-only SDR versions.

In addition to the protocol change, SDR has been extended to listen for sessions advertised on the IPv6 session advertisement (SAP) address, in addition to listening on the IPv4 SAP address. IPv6 address scopes (link local, site and global) have been added for creating sessions requiring IPv6 addresses. No changes have been made to the SIP user interface so it does not have support for IPv6.

The Secure Conference Store (SCS) uses SDR for listening to SAP advertised sessions using both IPv4 and IPv6; therefore only an update of the SDR version was required. Minor changes to the SCS server were required for it to support the creation and storing of IPv6 sessions. Creating new sessions required IPv6 address options in the web interface. Since the SCS does not actively announce sessions, but instead stores sessions, changes to the internal database structure were required to accommodate IPv6 addresses. The session description (SDP) presented to the SDP Parser (SPAR) applet, already had fields suitable for storing an IPv6 address. The SPAR applet needed improvements to be consistent with the SDP specification and therefore work with the IPv6 addresses.

The applet itself does not contain any networking code; instead it uses the browser's HTTPS transport mechanism to receive SDP and tool configurations from the server. It was therefore necessary only to use an IPv6-enabled web browser and server to upgrade the communications to use IPv6. Browser support was limited to a patched version of Microsoft's Internet Explorer. This was the only browser that supported the correct environment for SPAR including HTTPS, Java and IPv6 support. The patch was written by Microsoft Research as part of their IPv6 stack package. The Apache SSL web server was patched using the KAME IPv6 Apache network patches.

## **4.4 Use of IPsec**

### **4.4.1 Introduction**

Over the last few years, there has been a concerted activity in the IETF to standardise network-level security techniques. This has led to the development of the IPsec set of standards

[RFC2401]. IPsec operates at the network level, and has support for integrity, authentication and confidentiality. Fundamental to IPsec is a Security Association (SA) formed between a sending and a receiving party. In negotiating the SA, two symmetric encryption keys may be derived; one is used for authentication, but a different one may be used for confidentiality. If only authentication and integrity is desired, then a digital signature is formed by encrypting, with the authentication key, a combination of the source/destination and a hash of the packet contents. This combination is appended to the packet to form an authentication header. If confidentiality is desired, then the packet contents are also encrypted (often including the packet header), and a new security header is appended. The Security Association is formed by a negotiation phase between the sending and receiving parties. During the negotiation, some asymmetric exchanges are carried out, which allow authentication of the two parties of each other; one version of this negotiation is IKE [RFC2409].

If there had been widespread availability of IPsec implementations, and widespread deployment of these implementations, then we would have considered their use in MECCANO. Even then, for reasons given below, we might have had difficulties. As it is, there has been no attempt to use it in this project. There is ongoing activity amongst several of the MECCANO partners, in particular UCL, to investigate where it is would be suitable for multimedia conferencing. None of the current developments or deployments has considered it, however.

#### **4.4.2 Problems with IP Multicast and IPsec**

Even if there were wide availability of the current implementations, these still have considerable problems for our applications. There are a number of problems of this approach for multicast, secure conferencing. First, the IKE mechanism:

- The SA normally depends only on IP numbers; multicast conferencing normally uses also port numbers. While annoying, this would not alone be a showstopper.
- There is no agreed method of doing IKE in the multicast environment – and hence of forming multi-way SAs.
- There is some difficulty in linking personal names to IP number – yet this is the link needed in the Security Association.
- Some IPsec implementations just do not work with Multicast.
- Many implementations of IPsec do not allow user programs to enter encryption keys.

In view of the above, we believe that there are major unresolved standards and implementations issues before we would be comfortable using IPsec. For the moment, therefore, we are using only application level security – even in the development versions of our software.

#### **4.4.3 Problems with the use of relays**

There are other problems with the use of IPsec in relays. In a relay, there is normally some change in the IP Header between the sending and receiving side. Because the digital signature depends on both the Header and the data, it is not possible to relay a secured packet without knowing the encryption keys in the Header. Normally it is considered a violation of end-end security to inform an intermediate node of these keys. It is possible to have a destination start up a particular secured stream across a relay. While this could lead to compromise of the particular stream, it would not be as serious as automatically informing the relay of the result of all Security Associations - for instance by permitting the relay to automatically act as an end-point of each part of the stream in its own right.

With some relays, it would be possible to have the multicast side of a relay run secure conferencing with application-level encryption; the encryption key could be entered in the

normal way by a user on the other side. Particularly the multicast-unicast conversion relays come into this category. The user could derive the SDP parameters, including encryption key, from the SCS via SPar or from SDR, both delivered in D4.3. It would then be possible to open unicast streams over IPsec on the unicast side of the relay using the normal IKE. We have not carried out this mechanism, but there would be no problems in principle in so doing.

#### 4.4.4 Problems in the use of IPsec with servers

Similar considerations arise with multicast and unicast access to media servers. If the media servers are running in a multicast mode, the same considerations arise as in 4.4.3 above. They can run in Unicast mode, being attached to conferences through a multicast/unicast relay. Again the unicast side could use standard IPsec, the multicast side using application-level security. We have not thought through why this hybrid form of access would be an advantage.

## 5 Required IP network performance

As noted in Section 4.1, the performance of an IP network can vary significantly. The primary factors which affect this are packet loss and jitter (variation in end-to-end delay).

Packet loss is the primary cause of poor reception quality in IP-based conferencing systems. The quality observed depends on the loss rate and pattern of loss (e.g. evenly spread loss is easier to repair than burst loss), upon the amount of error protection coding applied to the media stream, and upon the error concealment algorithm applied by the receiver.

The two audio tools developed in the project, RAT and FreePhone, both use redundant packet transmission to protect the media stream from packet loss. In addition, RAT employs sophisticated error concealment. Accordingly, for audio, we define a *reasonable* quality connection to be one which experiences less than 15 % packet loss. This value is based on experience and listening tests [Hardman95], but is clearly dependent on the loss patterns observed. Higher loss rates can be tolerated, but the signal then becomes “intelligible” rather than “pleasant” to listen to (indeed, operation at 20-30 % loss is possible, with the resulting speech being metallic and unpleasant sounding, but totally understandable).

The video tool used in the project, Vic, deals less well with the effects of packet loss (since audio was considered more important for the usability of the conferencing system). Informal experience shows that 10 % loss is the maximum which allows acceptable performance.

Delay variation has two effects: it either forces the media tool to increase the size of the playout buffer, to accommodate late packets, or it forces the tool to discard late packet to retain a low end-to-end delay. The choice must be made based on the expected use of the session (i.e. interactive or not).

When running over the TEN-155 European backbone network, we did not experience significant problems due to delay variation (i.e. the variation was small enough that the systems could still be used in an interactive fashion, with negligible packet loss due to late arrival). Measured jitter was on the order of a few milliseconds on average, with extremes in the order of 10s of milliseconds.

We therefore define a reasonable quality Internet to be one which exhibits loss rates less than 10 % and with jitter on the order of milliseconds.

## 6 Observed IP network performance

### 6.1 IPv4 Multicast

In general, the performance of IPv4 multicast has been poor during the lifetime of the MECCANO project, and conducting Europe-wide multicast sessions has been difficult, at best.

This can be attributed to a number of problems. Insufficiently developed routing protocols, poor support by router vendors of those protocols which exist, transition problems during the switch from a single routing domain to a network with true inter-domain routing, and lack of support from the network operators (due to the difficulty of getting multicast working and the limited user base once it is operational).

Despite this, it must be said that multicast has proved itself very capable in some regions of the network. The regional and national multicast connectivity usually was sufficient for decent multimedia conferences. Over a long period, the University of Erlangen has been transmitting audio/video streams showing actual TV broadcasts into the German Mbone at rates around 500 Mbps without major problems. However, the pan-European multicast backbone has proved to be very difficult to use. From an application user's point of view, the following problems emerged:

- SDR announcements did not reach other partners,
- other partners often did not appear in the user lists of MECCANO tools (or temporarily disappeared during a session) although they were present and transmitting,
- the quality of audio/video transmissions suffered from high packet loss rates and
- some partners were only able to receive incoming packets while the traffic they sent did not reach the other conference participants (or vice-versa)

For an ordinary user, this situation is quite frustrating and may lead to the decision not to use MECCANO applications (or Mbone conferencing tools at all) but to prefer unicast-based solutions such as H.323 applications or proprietary tools such as CU-SeeMe instead. Indeed, the lack of multicast infrastructure, especially for SOHO users, and the poor performance of the international Mbone have led to the rather bad reputation of multicast-based conferencing solutions.

As described earlier, several tools have been written within MECCANO in order to be able to *look deeper* into the Mbone and to locate any problems. RQM gives a quick real-time overview of packet-loss rates between conference participants and can be used to start a multicast traceroute (mtrace) to check the distribution path between sites. MeshTrace runs such mtraces between all possible pairings of a given set of conference participants to get an overview of the overall network situation. With the output from these programs, it was possible to understand the problematic behaviour of the pan-European Mbone, contact the proper service providers and convince them that they should look more carefully at their equipment and its configuration...

A few examples, generated from MeshTrace outputs, illustrate typical situations with which MECCANO had to cope. Since the relevant figures are colour-coded, they are best viewed on colour displays or from colour printouts. Users with black-and-white displays or printouts may have some difficulties understanding the diagrams.

# MECCANO Network Status

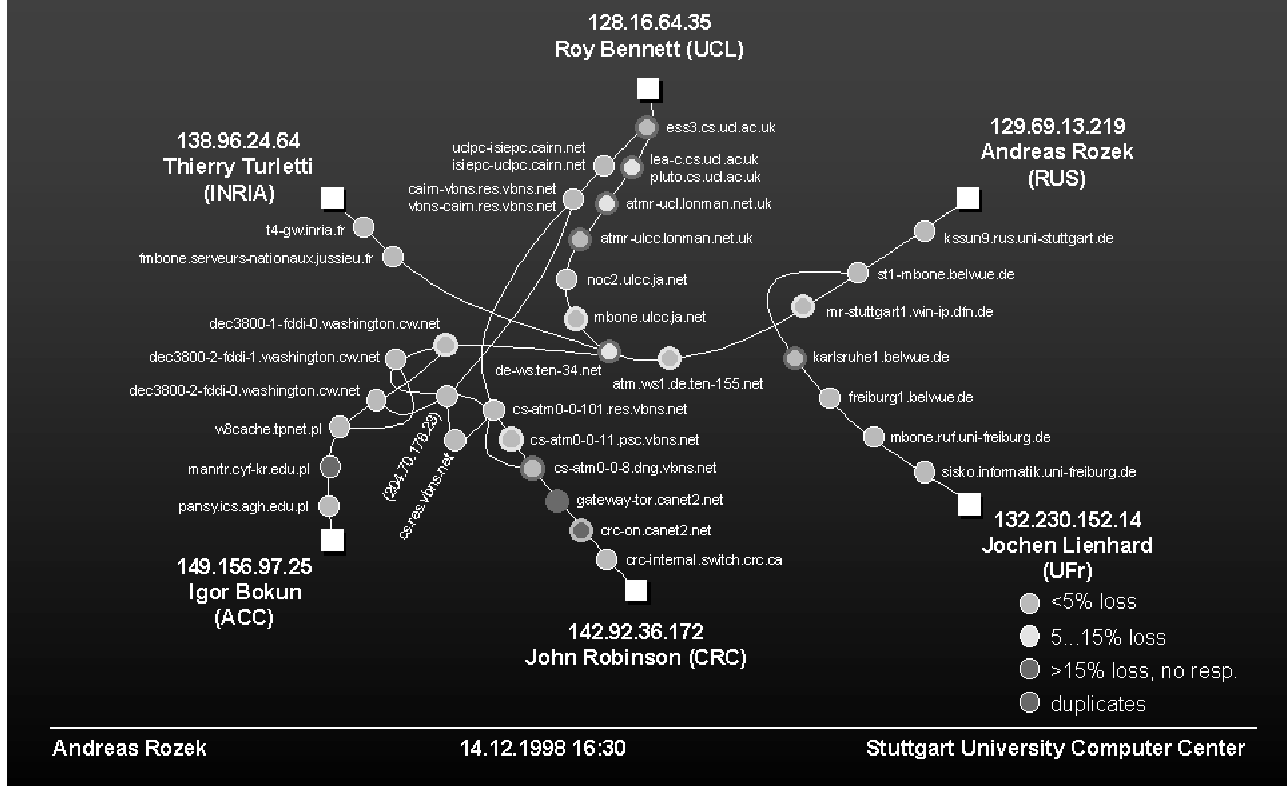


Figure 4: MECCANO Network Status on December 14th, 1998

## 6.1.1 Overloaded Routers, Links and Tunnels

In the beginning, packet losses resulted mainly from overloaded routers, crowded links between routers or DVMRP tunnels with too restrictive bandwidth settings. Fig. 4 shows the star-like topology of the pan-European Mbone, centred at “de-we.ten-34.net”, which could not handle the multicast traffic properly. Additionally, some regional and local multicast routers (some of them just being workstations with multicast routing software) appeared to be overloaded. Note also the weird routing between Poland and the rest of Europe which goes via the U.S. and the complicated topology between Canada, the U.S. and Poland.

After indicating the situation to the technicians responsible, they were able often to solve the problem by upgrading their routers, increasing the throughput between adjacent routers or simply modifying their configuration files. Fig. 5 shows a few improvements of that kind.

UCL had been able to improve its routers which resulted in reduced packet loss. However, other routers could not be upgraded within a month's time (especially over the winter break) and the ongoing connectivity problems led to an additional tunnel short cut between Germany and the U.S.

# MECCANO Network Status

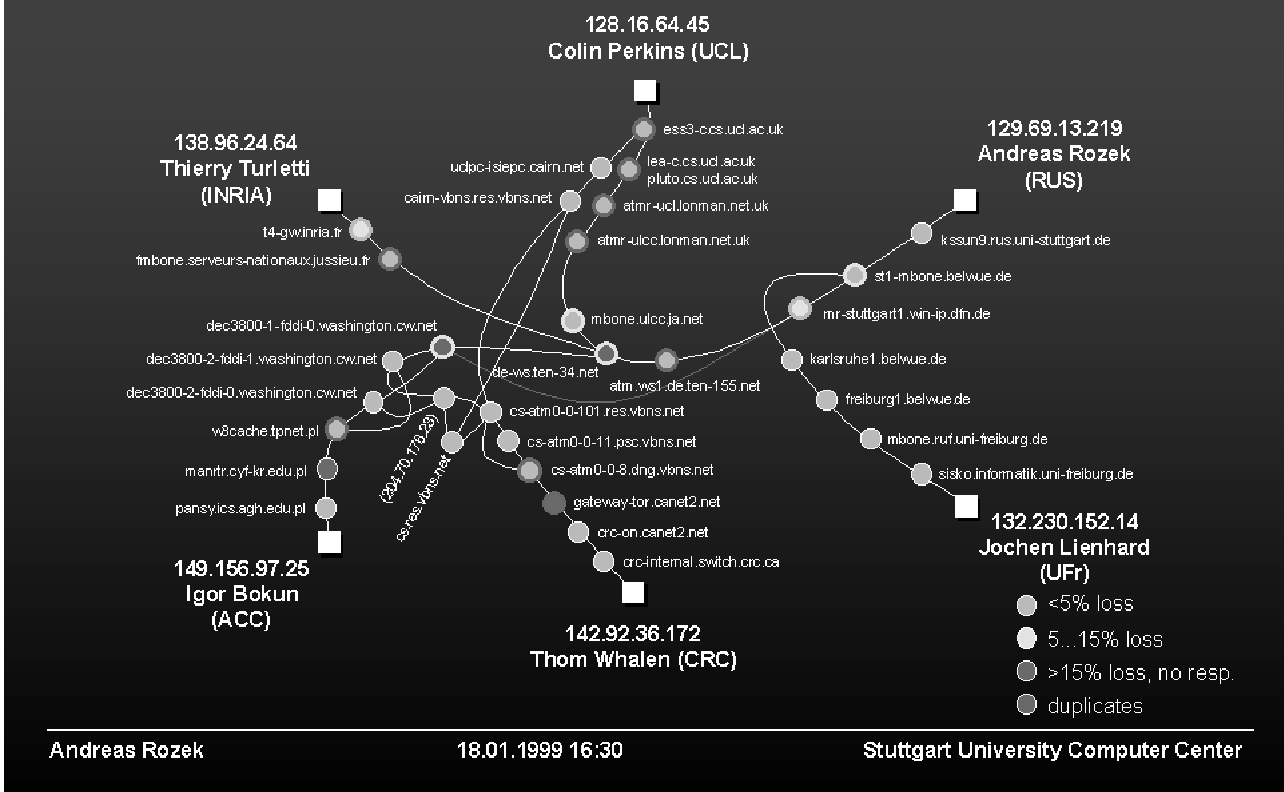


Figure 5: MECCANO Network Status on January 18th, 1999

## 6.1.2 Routing Loops and Flapping Routes

The situation changed when TEN-155 links became operational and PIM was introduced as the new routing protocol instead of DVMRP. Although the routers were now powerful enough to handle the multicast traffic in principle, flapping or misleading routes prevented multicast packets from being delivered properly (and in-time!) and unidirectional effects became common.

As an annoying side effect, the Mbone behaviour became unpredictable on a shorter time scale than before. A sequence of three MeshTrace observations, figures 6, 7 and 8, recorded with a break of 10 minutes between each, illustrates this problem.

The situation illustrated in the Fig. 6 looks almost ideal: most routers are rendered green, indicating normal operation without significant packet loss. Why four Norwegian routers seemed to have severe transmission problems remains unclear. That they recovered from this situation during the next few minutes, as shown in the following figures, leads to the assumption that these routers were rebooted or otherwise reconfigured. (Note: Jörg Ott, Thierry Turlletti and Martin Mauve had not joined the session at that time)

# MECCANO Network Status

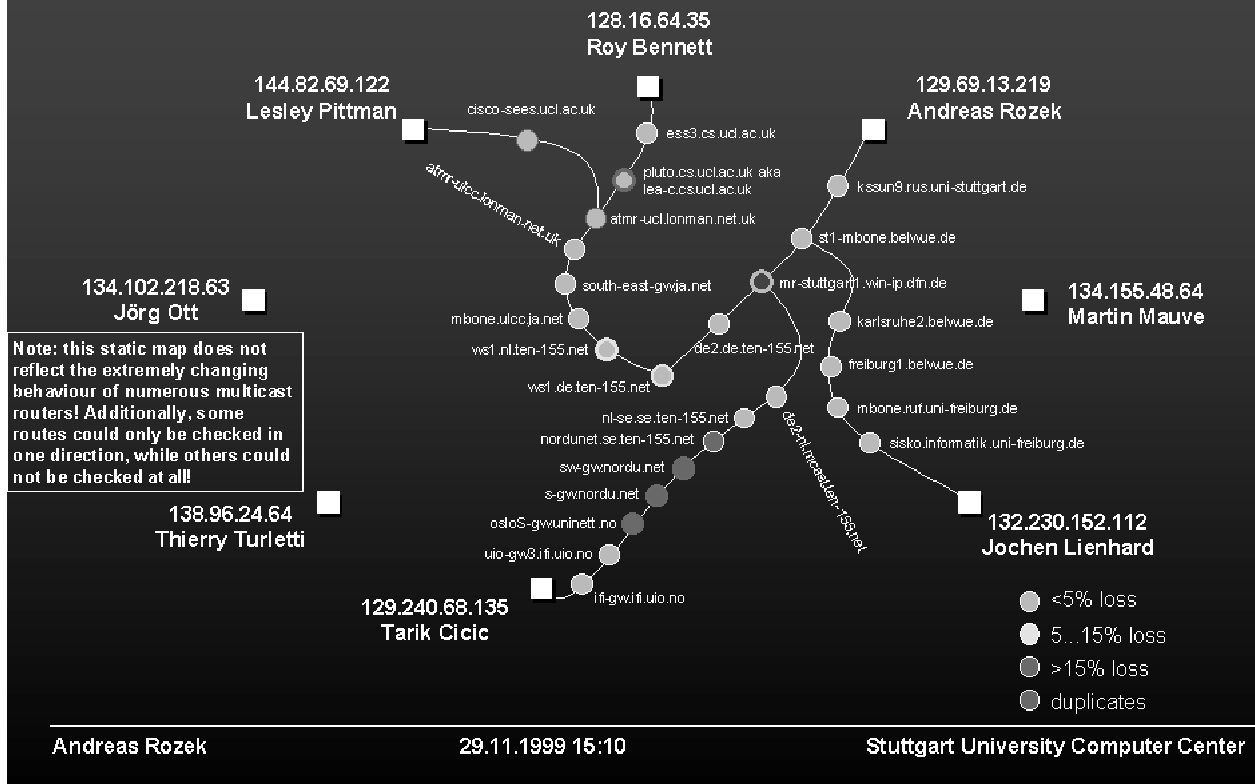


Figure 6: MECCANO Network Status on November 29th, 1999 at 15:10

A surprising detail in Fig. 6 is the role of mr-stuttgart1.win-ip.dfn.de as a centre for the routes to Norway - usually, all packets to and from Norway should pass ws1.de.ten-155.net rather than the relatively small router in Stuttgart. It might be because of this unusual load that the Stuttgart router sometimes did not respond to mtrace requests (as indicated by the transparent interior of the corresponding node).

Unfortunately, the figure does not properly reflect the problems MeshTrace had when collecting information about the multicast distribution paths between all participating sites: some multicast traceroutes failed in one direction but succeeded in the other direction and packet statistics for different paths with overlapping sections sometimes had contradictory contents. Such results of themselves indicate some instabilities in the routing - a perception which becomes clearer when looking at Fig. 7 (Note: Thierry Turletti and Martin Mauve had meanwhile joined the session, whereas Jörg Ott had not).



# MECCANO Network Status

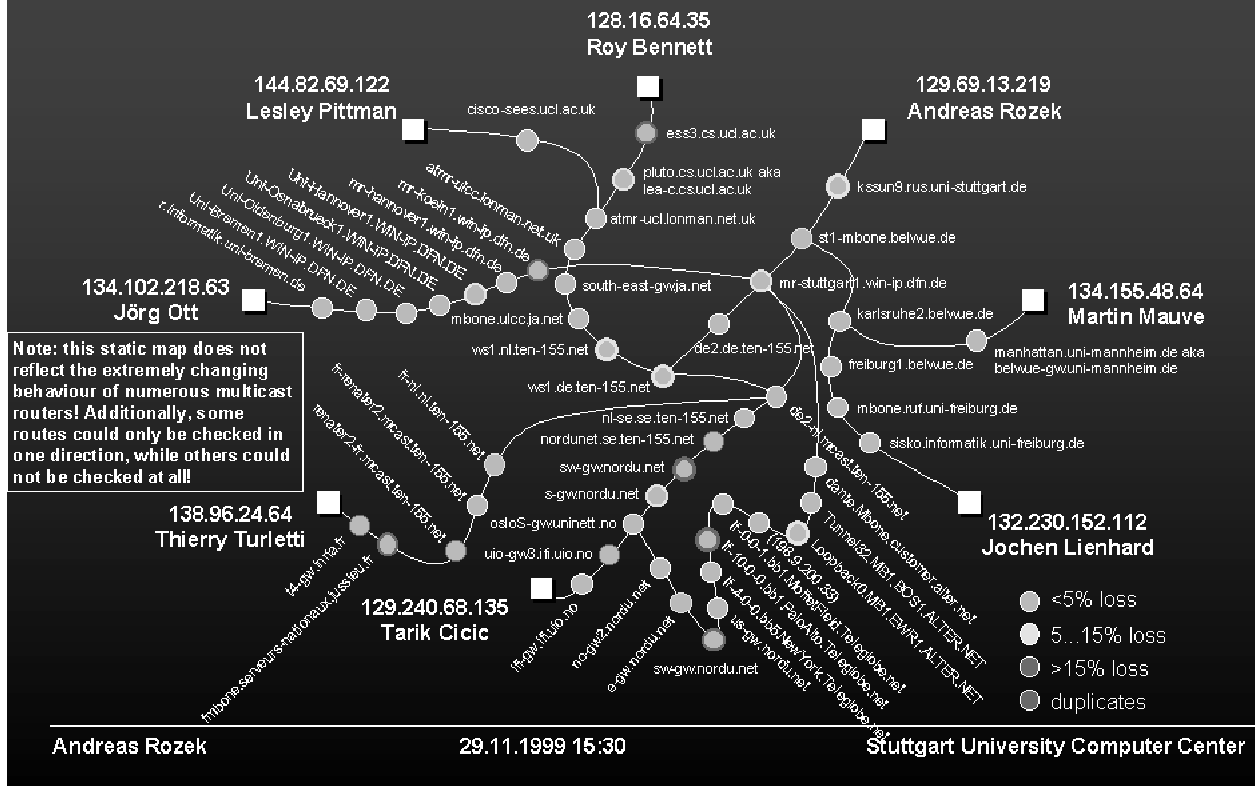


Figure 8: MECCANO Network Status on November 29th, 1999 at 15:30

### 6.1.3 Limitations of RQM and MeshTrace

In spite of their usefulness, both debugging aids have clear limitations. The most important one is that, in order to get valuable information, there must be at least some kind of multicast connectivity between the observed sites. Without any connectivity, due to a broken router, lost packets or any other problem that leads to 100 % end-to-end packet loss, mtrace is unable to determine the multicast distribution path and generate packet statistics. As a consequence, it is impossible to identify the source of the problems.

Fig. 9 illustrates how important it is to keep this restriction in mind (i.e. to keep in mind that missing paths in the diagrams shown above actually indicate extremely poor performance). It shows how many mtrace observations between participants succeeded during the period of some 30 minutes during which the three figures were constructed. The three observations are denoted as *a*, *b* and *c* in the table.

# MECCANO Network Status

Number of successful Path Traces after three Trials (a/b/c)

From:	To:	Roy Bennett (128.16.64.35)	Thierry Turletti (138.96.24.64)	Tarik Cielik (129.240.68.135)	Radek Ruchala (149.156.97.16)	Andreas Rozek (129.69.13.19)
Roy Bennett (128.16.64.35)		0	0	0	0	
Thierry Turletti (138.96.24.64)		0	0	0	0	
Tarik Cielik (129.240.68.135)		abc	0		ab	abc
Radek Ruchala (149.156.97.16)		0	0	0		0
Andreas Rozek (129.69.13.19)		bc	ab	bc	ab	

Andreas Rozek

10.02.2000 16:45-17:15

Stuttgart University Computer Center

Figure 9: Number of successful Path Traces after three Trials

While the high number of zeroes simply indicates the low quality of the network, the numerous *ab* and *bc* entries give a slight impression of the dynamic nature of the multicast connectivity. By comparing corresponding entries above and below the diagonal, the table in Fig. 9 also shows the unidirectional behaviour of some connections.

## 6.1.4 Conclusion

Our experiences with multicast connectivity have shown that there are dramatic deficiencies at the network level. These show themselves in poor performance, rapidly changing state, difficulties in locating problematic routers and links. The need to persuade service providers to debug their network has prompted us to make a large investment in trying to identify the problems and provide evidence of failure to the providers.

The poor performance results mainly from immature multicast routing protocols and (apparently) centralised topologies which turn out to be less robust against local failures than current unicast networks. On the other hand, long-lasting test sessions with a regional service provider have also shown a lack of proper monitoring and management tools for multicast networks on the provider's side. As the actual consequence, running a multicast network is a difficult task and not yet well understood and running multicast conferences is somewhat like playing dice without a die...

In future, the situation will become even more difficult if, as expected, more and more sites install packet filters which (silently!) drop incoming packets according to filter rules based on

protocol ids, sender addresses and port numbers. The ignorance of operators or filter misconfiguration may lead to unidirectional packet losses; these will be very difficult or even impossible to locate should such packet filters affect also tools like mtrace and mrinfo.

Thus, there is little prospect of the network situation improving from a user's perspective. It will become increasingly difficult to use multimedia conferencing tools, and multicast applications in general, unless proper routing protocols with stable and interoperable implementations within routers are made available. There will also be a need for multicast management systems and debugging tools. Until multicast networks become as reliable as current telephone systems, there will be also some need for debugging aids which help with the examination of network connections on an end-to-end level - either for the application user or, better still, for service providers.

## 6.2 IPv4 Unicast

In spite of the problematic pan-European multicast performance, MECCANO had to carry out a number of network conferences and seminars in order to coordinate project activities and to fulfil the contract. Experiments with a central packet reflector and unicast connections between project partners showed much better performance, so that MECCANO had to rely on the European unicast infrastructure rather than the Mbone for many of its sessions (see also Section 4.2.3).

Practical experience with this reflector setup has shown, that, in principle, enough bandwidth is available within Europe for international multimedia conferences and that the probability of satisfactory transmission quality is much better today than it was a few years ago (i.e. during the MICE and MERCI projects). As described before, the poor multicast performance is mainly a matter of routing rather than network capacity.

Unfortunately, concrete (numeric) comparisons between the unicast and the multicast setup could not be made. As described in Section 6.1.3, relevant multicast measurements can only be performed if there is at least *some* kind of multicast connectivity between the participating sites - but this was not the case whenever the project partners decided to use a packet reflector instead of the multicast infrastructure.

## 6.3 IPv6

We have limited ourselves to providing support for IPv6 in our toolset, and conducting limited small-scale tests of that support. In a LAN environment the tools have been observed to work reasonably well, although a number of bugs in the IPv6 protocol stacks supplied with various systems have been observed (in particular, multicast support has been less than reliable in some cases, particularly as the number of participants rises). These problems are to be expected, since the vendors, in the main, label their IPv6 support "experimental".

We have not conducted wide area IPv6 trials in the MECCANO project, although some of the projects we support (e.g. COIAS) have done so. Feedback we have received from those projects indicates that the tools we provide work as expected, but there are significant problems with the IPv6 network infrastructure which have to be resolved before IPv6 can be considered to be ready for full service.

# 7 Tool support for non-IP networks

As noted previously, the MECCANO project has focussed primarily on the use of IP as a convergence technology, with other networks being used as bearers for IP. In order to support

legacy systems, we also provide a number of gateways between IP-based conferencing systems and ISDN/PTSN/ATM systems.

These gateways are discussed in more detail in deliverable D6.2/7.2; hence we outline them only briefly here.

## **7.1 ISDN/Voice telephony networks**

The MECCANO Audio Mbone-Telephony Gateway “AudioGate” provides users on an arbitrary telephone network (PSTN, ISDN and GSM) with access to the audio channel of multicast conferences. On connection setup, functions such as dynamic conference selection are provided. As soon as a “connection” to an Mbone session is established, additional services such as user identification, muting, most recent speaker indication, etc. are provided.

AudioGate was developed in several phases. The initial phase focussed on implementing, testing, and enhancing the core components, with less focus on the overall architecture. This phase, completed with the delivery of AudioGate 0.1, showed all the base technology working. Several further steps were needed to convert AudioGate 0.1 into a gateway based upon the Mbus architecture and to enrich the range of functionality offered to the users.

In the MECCANO project we have put much effort into the development of standards for multimedia conferencing, and of reference implementations of those standards. The ACC IP-PSTN gateway shows the power of these activities by integrating a number of components developed outside the project into a useful gateway device which can interoperate with components developed by the project.

This gateway is designed to enable a user to call a regular phone from a computer connected to Internet. It makes use of a Dialogic D/41ESC card for access to telephone lines from a PC running Windows NT and gateway software developed in the project from a combination of the Columbia University sipd, the Elemedia RTP library and the Dialogic Gatenet package. The resulting gateway allows a user with a standard SIP terminal and an RTP-based audio tool (e.g. sdr with RAT/FreePhone) to communicate with a user on a regular telephone.

In a similar way, the UiO Multicast-Unicast Reflector Servers (MURS) have been extended into a prototype H.323-Mbone Gateway using a lightweight architecture for integration of H.323 audio equipment (Cisco’s VoIP technology) with RAT. The gateway can be used, in a similar way to AudioGate, to access IP multicast conferences from telephone networks. It differs in its approach from that of AudioGate by the ad hoc use of existing tools for a specific purpose, rather than the design of a system capable of extension to other media, such as StarGate.

## **7.2 Asynchronous Transfer Mode**

We have provided little explicit support for ATM within our toolset, using it instead as a bearer for IP. In this context, we used the TEN-155 managed bandwidth service to provide loss-free PVCs over which IP tunnels were implemented.

There is one area where our toolset does contain specific support for ATM networks. This is the Robust-Audio Tool, which contains an “audio” driver which can interface with a native ATM AAL1 audio system. This driver, contributed by Nortel Networks, follows the model developed in the project by the AudioGate system, showing how the gateway-integration concept advocated there can be applied to systems other than ISDN.

## **7.3 Direct Broadcast Satellite**

INRIA DBS system uses the Eutelsat skyplex satellite named Hot-Bird 5, located at 13 degrees East. Hot-Bird 5 provides onboard multiplexing that allows multiple up-link frequencies to be multiplexed within a single downlink frequency. As a result, a receiver can listen to multiple IP streams from several feeds within the same frequency.

The main use by the MECCANO project of direct broadcast satellite has been in the development of a routing protocol for unidirectional links. We have focused on these aspects: standardisation, validation, driver development and the deployment of services.

### **7.3.1 Standards activities**

INRIA has been actively participating for 4 years in the Unidirectional Link Routing (UDLR) working group at the IETF, working on the standardisation of a "Link Layer Tunnelling Mechanism". This mechanism emulates a bi-directional connectivity between node connected with a unidirectional link. It is implemented on every node at link layer level and is therefore transparent to the IP layer. Every node at IP level sees the link as bi-directional and can communicate with any other nodes. As a result, the mechanism allows unidirectional links to be transparently integrated in the Internet; furthermore routing protocols do not need any modification to supports these links. This mechanism has been agreed within the UDLR group and is now in the process of a last call as a proposed IETF standard.

### **7.3.2 Validation activities**

In order to validate the protocol, interoperability tests were conducted between INRIA's UDLR implementation and that of the SONY Corporation. These tests were necessary to gain the proposed standard status. In June 1999 at INRIA, SONY equipment was connected to the INRIA satellite network and communicated with that of INRIA, demonstrating complete interoperability of the implementations.

### **7.3.3 Driver development**

INRIA has developed a series of drivers to support the UDLR mechanism. Development started three years ago with the COMATLAS cards. These DVB-decoder cards have several drawbacks, mainly due to use of the ISA bus and the interrupt design. For these reasons INRIA decided to support other DVB cards such as Harmonic Data Systems's NMC-model 040 (PCI bus). The drivers are available for the FreeBSD 3.x and Linux 2.0 operating systems.

Since September 1999, we have been transmitting IPv6 packets over the satellite link, making it necessary to update the drivers (transmission and reception) to support IPv6. We have made several tests (unicast and multicast) to validate the IPv6-UDLR support. Drivers are available for the COMATLAS card with FreeBSD 2.2.x using the IPv6 KAME stack.

### **7.3.4 Deployment of services**

In July 1999, the Conseil General du Tarn, which owns transmission equipment and satellite capacity on Hot-Bird and whose equipment implements the INRIA link layer tunnelling, demonstrated that receivers can receive the transmissions from INRIA and the Conseil General du Tarn at the same time. Currently, the satellite network includes two transmission stations and several receivers able to receive simultaneously the IP transmissions from the two stations (TDMA access). We have used a multicast routing policy for this architecture. Since both INRIA and the Conseil General du Tarn have enabled multicast routing protocols in their routers offering Mbone connectivity to the receivers, these receivers can forward multicast

traffic on the reception sites, if requested. Both feeds transmit IP traffic to a maximum of 1Mbps.

The network architecture and the available bandwidth are particularly suited to experimenting with reliable multicast file transfers as well as video conferencing applications. INRIA has developed a Reliable Multicast implementation, based on adaptive Forward Error Correction, that makes use of the special satellite features. It allows distribution over a great number of receivers: high bandwidth, high delay and unidirectional lossy links. Some tests have been made over the satellite link in order to validate this implementation.

We plan the following work on DBS:

- Drivers: Support of IPv6 - UDLR for the new DVB cards under FreeBSD 3.x and KAME stack.
- Reliable Multicast: Multilayered support and congestion control.
- Routing: Test of multicast routings protocols (PIM-DM, PIM-SM) with UDLR.
- QoS: Test of an IP management approach.

## **8 Performance of non-IP networks**

### **8.1 ISDN/Voice telephony networks**

The MECCANO project used ISDN/Voice telephony networks only as bearers for standard voice traffic, gatewaying that traffic onto an IP network via systems such as the University of Bremen's AudioGate. Accordingly, there is little to say about the performance of such networks.

We did experience difficulty with the Linux programming interface to ISDN cards, but this was satisfactorily overcome.

### **8.2 Asynchronous Transfer Mode**

The primary experience of the MECCANO project with ATM networks was the use of the TEN-155 managed bandwidth service to provide a high quality network path between partner sites. The MECCANO project was the alpha-test site for this service: the following is adapted from our report [T155-alpha] on this test for DANTE.

#### **8.2.1 Introduction**

The Managed Bandwidth Service (MBS) enables connectivity among research projects in countries connected to TEN-155, the European Research Network. The service allows projects to request bandwidth for specified periods of time, connecting specified nodes, with various service guarantees. Access, via ATM, is available to institutions connected to a National Research Network (NRN) or directly at a Point of Presence of TEN-155.

Deployment of the MBS was scheduled to occur in three phases: alpha test, beta test and full service. The aim of the alpha test was to verify procedures for establishing an MBS “account”; to verify working of the ATM NOC service; to identify and resolve issues re NRN interworking; and to identify and resolve issues re NRN internal workings. Testing of the service included these tasks:

- Assess usability of the service
- Assess ATM NOC's interaction with the alpha test user and DANTE
- Assess DANTE's interaction with alpha test user

- Assess NRN's ability to provide service
- Assess interaction with NRNs
- Determine procedural adjustments needed

and the success criteria for the alpha test were defined to be:

- Connections established through to end systems
- Service usable by applications
- Change requests handled correctly
- Scheduled connections on time 95% of time
- Appropriate interaction with NRNs established

On successful completion of the alpha test, the beta-test started. This extended coverage to additional countries with multiple sites in each country, extended the service to multiple projects, verified that revised procedures developed as a result of the alpha test worked, and verified interaction with the additional NRNs. The sole alpha tester was the MECCANO project, acting on behalf of ERCIM.

### **8.2.2 The alpha test procedure**

The initial plan for the alpha test was that DANTE would produce a series of web pages which would allow for service to be requested and for ongoing requests to be tracked. Each project would nominate a single Group Network Manager (GNM) who will liaise with DANTE, via these web forms and email.

The national path was provided by the NRNs, who nominated an Access Port Manager (APM) as their liaison. The international path was provided by TEN-155 via their ATM-NOC. DANTE was the coordinating partner.

It was planned that the projects would request service from DANTE, who would liaise with the NRNs to arrange connection. The Project (via its GNM) was allowed to liaise directly with the ATM NOC if problems occurred, but had to copy DANTE on all correspondence to ensure that all issues were correctly addressed.

Getting access to the managed bandwidth service had four stages, which involved the following people:

- Agreement stage (DANTE, GNM, APM)
- Delivery stage (DANTE, GNM, APM, ATM NOC)
- Production stage (DANTE, GNM, ATM NOC)
- Finish stage (DANTE, GNM)

Much of this interaction was managed by electronic mail and web forms. The Authorised contact (GNM) was able to reach the ATM NOC and ask for changes or last minute modifications to the service. A policy for *change requests* was defined, establishing boundaries for changes. Within those boundaries, it was planned that changes should be made within a two hour response time for the TEN-155 part of the network path.

### **8.2.3 Experience**

The agreement phase of the alpha test proceeded smoothly. We completed the forms produced by DANTE giving details of the GNM, sites to connect and technical contacts at those sites, together with details of the time period for which we desired connectivity. The information requested was reasonable and simple to provide.

We initially requested three ATM VPs linking UCL, INRIA Sophia Antipolis and RUS. An additional link was later requested linking RUS and Messe-Essen for the purpose of providing a demonstration of the MBS at the launch of the 5th Framework programme.

The timing of these links was for 14:00 to 17:00 (UK time) every Monday, between 1 March and 26 April 1999 and continuously from the 22nd to 25th February 1999. We requested a CBR service at 2Mbps. Some confusion arose due to the changing requirements, but on the whole the agreement phase went well.

As was expected, the delivery stage of the alpha test was considerably more difficult. There are a number of stages involved in getting connectivity:

- Connections between partner sites and the NRN
- International connections between the NRNs and TEN-155
- National connections between multiple partner sites over an NRN

It was intended that each NRN would designate a single contact point, the APM, who would liaise with partner sites to arrange connectivity between the NRN and the partner site. This process was to be initiated by DANTE, after the project GNM requested access to the MBS. In practice, possibly due to a shortage of time, the initial request for service to the NRN was made by the individual partner sites in parallel with the request to DANTE. This led to some confusion, since there was no clear chain of responsibility within the NRNs and it was unclear who should be contacted to resolve problems. Even after the alpha test was complete, we were unsure who the designated APM was for each NRN - this area had to be clarified before the beta test started, with procedures being more strictly enforced.

In a similar manner, it was difficult to ensure that each NRN configured the same level of service to each partner site. Again, we used email to coordinate this, but on a flat basis: we had a single mailing list. If the plan for each NRN to designate a single APM had been realised, this would have reduced the communication overload, since they could have communicated with each other, the GNM and DANTE only, rather than having to involve all participants. This also overlapped onto the setup of international connections. Once all the sites and NRNs agreed on a common service, connections between them were established smoothly. However, it took some time for a common service to be agreed.

We have had experience of only a single national connection between multiple sites across an NRN - from RUS to Essen. This connection was unusual since the site at Essen was not a MECCANO project site, but rather a temporary addition to the alpha test for the purpose of the demonstration at the launch of the TEN-155 network. Accordingly, coordination between the MECCANO project, DANTE and Essen was more difficult than between the MECCANO partners and DANTE.

Once the connections were in place, they performed well and very few problems were observed during the launch demonstration. We had a number of problems setting up the demonstration, but these were not the fault of the TEN-155 network or DANTE - the service provided worked very well, and the demonstration was a success.

We must note, however, that a pure ATM service, whilst allowing for great flexibility in terms of its use - since it is a level 2 service – places a heavy burden on the users of that service, who must configure the IP level routing and setup a VPN themselves. With the current level of maturity of the IP QoS and VPN services it is unclear that it would be possible for DANTE to offer a layer 3 managed bandwidth service, but if such a service could be offered, it would simplify the job of the users significantly.

This is one of the main areas of difficulty we had in using the MBS - we wished to use IP multicast and trial IPv6 over this infrastructure and it is non-trivial to configure the IP routing to allow this to take place. This is not strictly an issue with the MBS, but is something that potential users of the MBS must consider: if a level 3 service more complex than unicast IPv4 is desired, then the users of the MBS must be prepared to put considerable effort into configuring this service.

The charging relationship between the project, DANTE and the NRNs needed to be clarified before further tests took place. Certain partner sites received considerable pressure from their NRNs which wished to charge for the use of the MBS. This pressure was made worse by the apparent difficulty the NRNs have in providing occasional access to a link: the links we had requested to be active on Monday afternoons were active continuously for some time, before being disabled.

We must note that, owing to links only being established late in the period, parts of the notional procedure (changes to schedule/bandwidth) went untested.

#### **8.2.4 Conclusion**

The TEN-155 MBS did, in general, perform well and the network was very useful for the MECCANO project. There were clearly a number of rough edges in the procedure which were sorted out before it became a production service, but this was only to be expected during an alpha test. On the whole, the alpha test was considered a success.

### **8.3 Direct Broadcast Satellite**

The UDLR system developed for use over DBS satellite has been in continuous use with IPv4 throughout the MECCANO project, and with IPv6 since September 1999. Performance of the system has been excellent, with negligible loss rates observed. A number of seminars have been conducted over the system, showing its utility for this activity (of course, the latency induced by the satellite channel makes it unsuitable for interactive use).

In addition, interoperability tests have been conducted between the INRIA implementation and a commercial implementation from Sony Corporation. These tests were successful, showing complete technical interoperability, and demonstrating commercial interest in this standard.

## **9 Conclusions**

It is clear that the basic idea of converged networks with IP as a bearer is viable and that the MECCANO toolset provides acceptable performance in the presence of a reasonable quality IP network.

The performance of unicast IP within the current networks has been good. When using a set of unicast flows with a centralized packet reflector, the conferencing toolset we have produced has worked very well, to the extent that the majority of problems we experience are due to poor microphone and camera placement, not the underlying network.

Unfortunately, the performance of multicast IP has been less impressive, with considerable amounts of packet loss being observed, together with numerous routing problems. We must emphasise that these problems are not due to lack of support by DANTE and the NRNs, rather we believe that the products available in the market-place today do not provide effective multicast support. To some extent this is to be expected as standards for multicast routing are still evolving and the code deployed is first generation. It is, however, disappointing.

The non-IP networks used in the project have, to a large extent, worked as designed.

## References

- [Bolot00] Bolot J-C and Vega-García A, The Case for FEC Based Error Control for Packet Audio in the Internet. To appear in ACM Multimedia Systems
- [Bolot99] Bolot J-C, Fosse-Parisis S and Towsley D, Adaptive FEC-Based error control for Internet Telephony. In *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, New York, March 1999.
- [DTMF] Schulzrinne H and Petrack S, RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals. Work in progress, May 2000.
- [Erdöl93] Erdöl N, Castelluccia C and Zilouchian A, Recovery of Missing Speech Packets Using the Short-Time Energy and Zero-Crossing Measurements. *Transactions on Speech and Audio Processing*, 1(3):295-303, July 1993
- [Handley97] Handley M, An Examination of Mbone Performance. USC/ISI Research Report: ISI/RR-97-450, April 1997.
- [Hardman95] Hardman V, Sasse M A, Handley M and Watson A, Reliable Audio for Use over the Internet. In *Proceedings of INET'95*, 1995.
- [Hodson00] Hodson O, Perkins C S and Hardman V, Skew Detection and Compensation for Internet Audio Applications. *To be presented at the IEEE International Conference on Multimedia and Expo*, New York, July 2000.
- [Kouvelas97] Kouvelas I, Hodson O, Hardman V and Crowcroft J, Redundancy Control in Real-Time Internet Audio Conferencing. In *Proceedings of AVSPN'97*, Aberdeen, Scotland, September 1997.
- [MECweb] The MECCANO Project web site  
<http://www-mice.cs.ucl.ac.uk/multimedia/projects/meccano/>
- [Miller50] Miller G A and Licklider J C R, The Intelligibility of Interrupted Speech. *Journal of the Acoustical Society of America*, 22(2):167-173, 1950.
- [Moon98] Moon S B, Kurose J and Towsley D, Packet audio playout delay adjustment: performance bounds and algorithms. *ACM/Springer Multimedia Systems*, 5(1):17-28, January, 1998
- [Moon99] Moon S B, Skelly P and Towsley D, Estimation and Removal of Clock Skew from Network Delay Measurements. In *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, New York, March 1999.
- [Paxson98] Paxson V, On Calibrating Measurements of Packet Transit Times. In *Proceedings of the ACM Sigmetrics Conference on Measurement and Modeling of Computer Systems*, pages 11-21, Madison, Wisconsin, June 1998.
- [Perkins00] Perkins C S and Crowcroft J, Effects of Interleaving on RTP Header Compression. In *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, Tel Aviv, Israel, March, 2000.
- [Perkins98] Perkins C S, Hodson O and Hardman V, A Survey of Packet Loss Recovery Techniques for Streaming Audio. *IEEE Network*, 12(5):40-48, September 1998.
- [Podolsky98] Podolsky M, Romer C and McCanne S, Simulation of FEC-Based Error Control for Packet Audio on the Internet, In *Proceedings IEEE INFOCOM'98*, San Francisco, April 1998.

- [Ramjee94] Ramjee R, Kurose J, Towsley D and Schulzrinne H, Adaptive playout mechanisms for packetized audio applications in wide-area networks. In *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, Toronto, Canada, June 1994.
- [RAT-4] Hodson O and Perkins C S, Robust Audio Tool (RAT) version 4. Software available online at <http://www-mice.cs.ucl.ac.uk/multimedia/software/rat/>
- [RFC2198] Perkins C S, Kouvelas I, Hodson O, Hardman V, Handley M, Bolot J-C, Vega-García A and Fosse-Parisis S, RFC 2198: RTP Payload for Redundant Audio Data. IETF Audio/Video Transport Working Group, 1997.
- [RFC2354] Perkins C S and Hodson O, RFC2354: Options for Repair of Streaming Media. IETF Network Working Group, June 1998.
- [RFC2401] Kent S and Atkinson R, RFC 2401: Security Architecture for the Internet Protocol. November 1998
- [RFC2409] Harkins D and Carrel D, RFC 2409: The Internet Key Exchange (IKE). November 1998
- [RFC2429] Bormann C, Cline L, Deisher G, Gardos T, Maciocco C, Newell D, Ott J, Sullivan G., Wenger S. and Zhu C, RFC2429: RTP Payload Format for the 1998 Version of ITU-T Rec. H.263 Video (H.263+). IETF Network Working Group, October 1998.
- [RFC2553] Gilligan R, Thomson S, Bound J and Stevens W, RFC2553: Basic Socket Interface Extensions for IPv6. IETF Network Working Group, March 1999.
- [RFC2733] Rosenberg J and Schulzrinne H, RFC2733: An RTP Payload Format for Generic Forward Error Correction. IETF Network Working Group, December 1999.
- [RFC2736] Handley M and Perkins C S, RFC2736: Guidelines for Writers of RTP Payload Format Specifications. IETF Network Working Group, December, 1999.
- [Rosenberg] Rosenberg J and Schulzrinne H, Timer reconsideration for enhanced RTP scalability, IEEE Infocom, San Francisco, California, March/April 1998.
- [T155-alpha] Perkins C and Cooper C, MECCANO Experience with the TEN-155 Managed Bandwidth Service Alpha Test, report to QUANTUM project, 26 April 1999. <http://www-mice.cs.ucl.ac.uk/multimedia/projects/meccano/publications/#oth>
- [Yajnik96] Yajnik M, Kurose J and Towsley D, Packet Loss Correlation in the Mbone Multicast Network. In *Proceedings IEEE Global Internet Conference*, November 1996.

Note: RFCs are available from the IETF web site at: <http://www.ietf.org/rfc.html>