B-ISDN Multipoint-to-Multipoint Call Configuration for CSCW Applications

Luca Cipriani
Ericsson Telecomunicazioni Italia
Italy

Giovanni Fruscio
CoRiTeL - Consorzio di Ricerca nelle Telecomunicazioni
Italy

Valentina Petrone
CoRiTeL - Consorzio di Ricerca nelle Telecomunicazioni
Italy

Marco Pugliese
CoRiTeL - Consorzio di Ricerca nelle Telecomunicazioni
Italy

"Foundation of this work is partially funded by the ACTS Programme of the CEC under the project AC012 MULTICUBE."

ABSTRACT

Unlike traditional applications, based on the exchange of data between a pair of end-point, multimedia applications of video-conference and co-operative tools, normally used for Computer Supported Co-operative Work, require simultaneous communications among a group of terminals.

Currently, ATM signalling standards only provide unidirectional multicast communication for point to multipoint call. On the other hand, the Multipoint-to-Multipoint Multiconnection call, which would represent the best answer to CSCW users requirements, is still far from a possible standardisation.

This work describes how to provide on demand Multipoint-to-Multipoint ATM connections for the support of multimedia applications, with a minimum upgrading of already existing services and signalling protocols.

The proposed solution has been adopted and implemented in the context of the MULTICUBE ACTS project.

KEYWORDS:

CSCW, SIGNALING, MULTIPOINT-TO-MULTIPOINT CALL, GENERIC FUNCTIONAL PROTOCOL.
1. INTRODUCTION

Three years ago, the availability of results from previous European Community projects, working on the development of multimedia applications and hardware/software tools for Computer Supported Co-operative Work (CSCW), raised the interest of the main community users, like those represented by European Automotive and Aerospace industries. As CSCW applications are by their very nature multipoint-to-multipoint and demand for very strict network parameters, like bandwidth and end-to-end delay, it is necessary to develop enhanced infrastructure, with new signalling and network functionality, to fulfil distributed requirements such as, fast response time, high communication throughputs and multipoint functionality. This enhancement makes interactive applications feasible over Wide Area Networks (WAN).

Broadband technologies, like ATM, can play a very important role in supporting CSCW applications. On the other hand, as current ATM signalling standards do not still provide multipoint-to-multipoint calls, from a communication point of view, the main problem to be solved is the multipoint-to-multipoint issue. In the context of the European ACTS projects, a sound proposal for solving such a problem, has been provided by MULTICUBE - Efficient MUltiPoint to MUltiPoint broadband switched network services for distributed MUltimedia application - project.

After an overview of the global network test-bed architecture and a short description of the solutions supported in the customer premises, to allow LAN-to-LAN interconnection through public B-ISDN, the paper will focus on the technical solution adopted in the project to provide on demand multipoint-to-multipoint ATM connections (hereinafter called MULTICUBE call).

2. TEST-BED ARCHITECTURE FOR MULIPOINT-TO-MULTIPOINT CALL

The need for multipoint communication, coming from aerospace and automotive user requirements, can be already realised, simply looking at the logical structure of the network’s trials related to the MULTICUBE experience (see Figure 1).

![Figure 1: MULTICUBE logical trials’ network](image-url)
The access network configurations highlight that two kind of topology, star and bus, have been considered for MULTICUBE applications. In all the cases, as shown in Figure 1, users’ LAN have been interconnected to the premises of the nearest National Host island, which in turn, have been linked through the ATM WAN, provided by the JAMES ACTS project. Depending on the customer premises configuration and the different running applications (IP or ATM based), involved users have adopted different LAN technologies.

Thank to the introduction, in the customer premises, of an entity called Signalling Server, multipoint communications can be performed on top of this network, integrating the necessary features to enlarge available broadband networks’ functionality and capabilities and developing proper signalling functionality. Under the assumptions that the Pilot ATM Network is reached via one Virtual Path and no more than one standard Signalling VC is supported per VP, all the signalling traffic from the user site shall be transferred through this unique Signalling VC. Consequently only a signalling endpoint per user site will be seen by the public network. This does not imply that signalling services can not be invoked from the other workstations, but it can be done elsewhere via TCP/IP sockets (or equivalent) through an appropriate interface. In this case, the signalling server has to be aware of the local physical configuration. As each user site is generally configured with more than one terminal, the Signalling Server must work like a signalling agent for the specific terminal involved in the call, so the signalling endpoint can be virtually located in that terminal. Note that more than one Signalling Server process, associated to each call, can run at the same time.

Without pretending the completeness and the reliability of a public B-ISDN network, the LAN interconnection infrastructure, achieved through the introduction of a Signalling Server in the customer premises, allows terminal equipment to easily co-operate with the public network in order to invoke signalling services and set on demand multipoint to multipoint call. In this way CSCW support is dynamically assured on wide scale, avoiding waste of resource. To this aim it is opportune to remember that currently, most of CSCW applications at the user site makes use of pre-established connections to send their ATM traffic to destination. With regard to MULTICUBE experience, in ATM based application, it is the application itself or other entity, managed by the user (let’s call it User Interface), to request the ATM connections or the services, through an appropriate API [Ref. 1].

In IP based applications (with or without RSVP), requests for connections come from the User Interface or from the IP layer depending on if RSVP is included (see Figure 2).

![Figure 2: Typical communication scenario at the CPE](image)

Although here we will not deal with it, it is worth to stress that the support of on-demand connections, as well as of the additional capabilities offered by the MULTICUBE switched network, previously not supported, has required an enhancement and an extension of the ATM Forum API primitives.

Before closing this section it is important to highlight the role and the tasks performed by the signalling server. As the vendor signalling do not support public multipoint-to-multipoint advanced signalling, the appropriate enhancements and integrating functions have been developed at the user sites, introducing them just in the Signalling Server. Due to the proprietary nature of the private ATM switch, which makes difficult providing it with internally software functions, the Signalling Server is located in a user workstation. As a result, the signalling messages shall pass through the switch transparently because they can not be dealt there properly. The switch is managed by the signalling server via management commands, as
a cross-connect equipment. Note that, for the communication among local workstations, it is used the vendor provided ATM LAN. An address resolution table has been defined on the signalling server for mapping the NSAP addresses into ATM (E.164) addresses of the possible local and remote destinations.

The Signalling Server is able to handle the whole signalling procedure, related to each terminal: point-to-point (pt-to-pt), point-to-multipoint (pt-to-mtp) and multipoint-to-multipoint (mtp-to-mtp) calls. In the case of a private star ATM network topology, it provides additional functions for accessing the public network interface through the private switch. The Signalling Server internal architecture is shown in Figure 3.

![Figure 3: Signalling Server internal architecture](image)

3. MULTICUBE APPROACH TO MULTIPOINT-TO-MULTIPOINT CALL

3.1 Basic Concepts

Generally multipoint applications require all parties, involved in a communication session, to send and receive information each other. Currently the standardised signalling Capability Set (CS2.1) only specifies the point-to-multipoint call, while the network capability and hence the control protocols to allow multipoint-to-multipoint connection, are only foreseen for Release 3 of B-ISDN. However multipoint-to-multipoint network capabilities can be emulated by opportunely assembling point-to-multipoint connections between interested parties. In particular the establishment of a MULTICUBE Call, that is a signalling association among $n$ users for the allocation of bearers and the provisioning of advanced services, is performed opportune establishing $n$ point-to-multipoint unidirectional connections. As each involved party should be able to send data towards the other $n-1$ parties and receive data from them, for each user we will have respectively $1$ outgoing point to multipoint connection and $n-1$ incoming point to multipoint connections (see Figure 4).

![Figure 4: MULTICUBE Call as co-ordination of $n$ point-to-multipoint connections](image)

3.2 Functional Model

From a functional point of view, the multipoint-to-multipoint capability can be achieved using the conceptual model described in [Ref. 2]. That recommendation specifies the functional protocol for the B-ISDN, for the application to a range of additional basic call capabilities and generic functionality, at the T_B or S_B / T_B reference point, by means of the Digital Subscriber Signalling System No. 2 (DSS2) protocols. Adopting that scheme for our goal and specialising the generic SS-control entity, which is service specific, in the multipoint-to-multipoint call control entity, the conceptual model results that shown in Figure 5.

Let us analyse briefly the model. In a multipoint-to-multipoint call users are connected each other through a mesh of standard point-to-multipoint connections. The master (the initiator of the call) establishes its point-to-multipoint connection towards the remote peers, which in turn are induced to establish further point-to-multipoint connections through the invocation of suited remote operations (i.e. $\text{setupMtpToMtpParty}$).
addMtpToMtpParty, etc.). The establishment of associations between remote peer entities (see [Ref. 2]) is provided by the GFP transport mechanism using Connection Oriented Bearer Related (COBR) procedures. Remote operations, carrying remote operation parameters (ROP), are coded and decoded by the Remote Operation Service Element (ROSE), using or the Abstract Syntax Notation One (ASN.1) or ad hoc rules, and carried through UNIs by means of the FACILITY message. On the other hand parameters and information, related to the multipoint-to-multipoint call, have been put in the Facility Information Element, inside UNI signalling messages. Finally the Co-ordination Function, beside to handle functions related to the multipoint-to-multipoint call, dispatches messages and primitives from and to the various signalling service elements (multipoint-to-multipoint Call Control, ROSE, GFP, UNI protocols). It is important to note that as the multipoint-to-multipoint service is provided using the already existing [Ref. 3] and [Ref. 4] protocols, that must not be modified, the above mentioned extra information (Facility IE, containing invoke, result or error components of an operation) has to be handled separately. 

The approach followed in the realisation of the MULTICUBE call has been that to re-use, as much as possible, existing standardised protocols, just introducing some new functions in the call handling. Table 3-1 sums up the set of recommendations selected in order to build the MULTICUBE call.

<table>
<thead>
<tr>
<th>B-ISDN CS2.1 (UNI and NNI) Recommendations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q.2931 Layer 3 specification for basic call/connection control</td>
</tr>
<tr>
<td>Q.2932.1 Generic Functional Protocol</td>
</tr>
<tr>
<td>Q.2971 Point-to-Multipoint Call/Connection Control</td>
</tr>
<tr>
<td>Q.2650 InterWorking of DSS2 and B-ISUP</td>
</tr>
<tr>
<td>Q.2610 Usage of cause and location in DSS2 and B-ISUP</td>
</tr>
<tr>
<td>Q.2762 B-ISUP general functions of massages and signals</td>
</tr>
<tr>
<td>Q.2722.1 Network Node Interface Specification for Point-to-Multipoint Call/Connection Control</td>
</tr>
</tbody>
</table>

Table 3-1: Recommendations selected to provide MULTICUBE Calls

### 3.3 Call Control Logic

The management of the point-to-multipoint connections to realise the MULTICUBE Call, has required a very careful definition of call control logic. The latter has been distributed in the Signalling Server and mainly in the Network Access Node related to the master of the call. On the other hand this choice allows the network to:

- easily have a “global view” of the MULTICUBE Call and co-ordinate different connections involved in a MULTICUBE call,
- easily face up troubles, related to concurrence, that occur among all parties,
- reduce the software complexity of the terminal

Moreover the network should have a global visibility of active connections because the user should be made paying more than $n$ connections for a MULTICUBE Call service, as there are additional functions provided.

In order to co-ordinate the different connections involved in a MULTICUBE Call, the Call Handling at the network node will make use of a data structure to storage the status of single connections, like a $nxn$ matrix, which rows keep the state of every point to multipoint connection in the MULTICUBE Call.
Such a table will be dynamically managed; each time a new party is added to the call, the size of the matrix will increase of one unit. For instance, when a confirmation of set-up of the connection $\text{M} \rightarrow \text{A}$ (Master $\text{M}$ toward Party $\text{A}$) is received, the Call Handler will set the element $\text{MxA}$ to Active (Figure 6). When a party is dropped, the whole row and column is set to 0 and the dimension of the matrix is reduced of one unit. Only the originating node processes the MULTICUBE call information and sends Bearer Related FACILITY messages [Ref. 2] to request the remote users, to perform required operations, exploiting the point-to-multipoint connection originated by the master user. At last it expects the remote destinations to do the same, as a confirmation of the completion of the operation. The complete set-up procedure of a MULTICUBE call involving three parties is depicted in Figure 6. As shown, only the Master of the call can request the addition of parties, whereas the Add Party procedures among other parties is explicitly requested by the network, by

Figure 6: MULTICUBE call set-up among three parties.
3.4 Timers related issues

From MULTICUBE call control point of view, some timers are required in order to bound the operations time, and in order to check the completion of the call:

- Timer TM3_Set (MULTICUBE CALL SETUP): it is started when the network sends the FACILITY containing addMulticubeParty component, and stopped upon receiving the SETUP message coming from the leaf;
- Timer TM3_Add (MULTICUBE CALL SETUP): it is started when the network sends the FACILITY containing addMulticubeParty component to the originating leaf, and stopped upon receiving the FACILITY message as a confirmation of the completion of the ADD PARTY operation towards the other leaf;
- Timer TM3_Drop (DROP OF PARTY REQUESTED BY THE NETWORK): it is started when the network sends the FACILITY message containing dropPartyConnection component to the originating leaf, and stopped upon receiving the confirmation of the dropping of the other leaf;
- Timer TM3_Rel (CALL RELEASE BY NETWORK): it is started when the network sends the FACILITY message containing dropMulticubeParty component to the originating leaf, and stopped upon receiving the RELEASE message as a confirmation of the release of the leaf’s outgoing point to multipoint connection.

4. MULTICUBE APPROACH VERSUS STANDARD

The multipoint-to-multipoint call is supported within this project with different linked point-to-multipoint calls, with some characteristics like the symmetry of the configuration. Actually, the available recommendations provide only the support of point-to-multipoint calls. The added feature in the MULTICUBE call is the “link” between the different ones, in order to get the real support of the CSCW service.

As the ‘MULTICUBE call’ involves several items susceptible of standardisation (Multiconnection, multipoint-to-multipoint), contributing to ITU would require to identify the most appropriated scope and not last the most appropriated time-frame for it.

ITU has been working on the point-to-point Multiconnection call since 1995, but the recommendation has not been concluded yet, as different approaches are being compared. Neither ITU is working on point-to-multipoint multiconnection call nor, even more, on multipoint to multipoint call.

In such a scenario MULTICUBE contributions could address the point-to-point multiconnection call recommendation, for long time delayed and not very mature yet. On the other hand, the transport information approach, followed by ITU for point-to-point multiconnection call, Q.2932 COBI (Connection Oriented Bearer Independent) plus TCAP (Transaction Capabilities Application Part), is quite similar to Q.2932 COBR (Connection Oriented Bearer Related) plus ROP, used in MULTICUBE.

Although the main effort in the MULTICUBE proposal has been spent in the definition of a set of procedures for parties and connection management, in a complex scenario where parties attached to various network nodes join and leave a ‘MULTICUBE call’, we believe our work too premature to impact the present proposal for standards, in this preliminary phase of study multipoint to multipoint call.

5. CONCLUSION

CSCW applications often require broadband network infrastructures, able to support multipoint-to-multipoint configurations. As current B-ISDN signalling standards do not provide multipoint-to-multipoint call, alternative solutions have to be considered. Following the principle of re-using as much as possible existing standard and just introducing some new co-ordination functions, the multipoint-to-multipoint call has been provided referring to the functional conceptual model described in ITU-T Q.2932.

The support of the new call handling functionality for MULTICUBE call, as well as
for the LAN to LAN interconnection through the public B-ISDN network, has required the introduction of a functional entity, called Signalling Server, able to manage the new complex realised configuration. It is worth to highlight that Signalling Server implementation is successfully running in MULTICUBE’s trials.

ACKNOWLEDGEMENTS

The authors wish to thank the members of the MULTICUBE ACTS project and in particular F. Argiolas, A. Ishaq (ERICSSON Telecomunicazioni), D. Lento (CSELT) and N. Lopez-Manzanares (TID) for the ideas, discussions, contributions to this work.

REFERENCES

Ref. 1 ATM Forum - Semantic Description of Native ATM Services version 1.0 - af-saa-0048.000;

Ref. 2 ITU-T - Q.2932 B-ISDN Digital Subscriber Signalling System n.2 (DSS2) – Generic Functional Protocol;

Ref. 3 ITU-T - Q.2931 B-ISDN Digital Subscriber Signalling System n.2 (DSS2) – UNI Layer 3 Specification for Basic Call / Connection Control;

Ref. 4 ITU-T – Q.2971 B-ISDN Digital Subscriber Signalling System n.2 (DSS2) – UNI Layer 3 Specification for Point-to-Multipoint Call / Connection Control;

Ref. 5 ACTS - AC012 MULTICUBE project and the standards - December 1997;

Ref. 6 MULTICUBE Annual Project Review Report - A012/CSE/MR/R/016/b0