Project: **Internet Protocols over Digital Video Broadcasting Media (IP over DVB)**

Study of Encapsulation and Protocol Performance

ESTEC Contract N.: 17471/03/NL/ND

Document ID: ESTEC 17471/03/NL/ND-D5.1

Final Report

Authors:	*Organisation:*
Marie-José Montpetit, Michael Schmidt	Joanneum Research
Gorry Fairhurst	University of Aberdeen
Bernhard Collini-Nocker	University of Salzburg
Hilmar Linder	GCS
Hillar Tork	EMS

Approved:	*Date:*	*Signature:*

Document File Name: final report V1.doc
Revision: 0.1
Date: 26.3.2004 Number of Pages: 37

Contents

1 Introduction

The project has developed new IP over DVB encapsulation to provide Internet services over DVB broadcast networks. The current common encapsulation, Multiprotocol Encapsulation, while providing basic services, adds non-necessary overhead to the IP packet while not fully supporting the IP requirements for next generation networks. Amongst those are the supports to the IP version 6 (IPv6) packets and Quality of Service (QoS). Based on the existing Internet Drafts (IDs) of the IP over DVB mailing list, the project has provided a full implementation, from software to satellite demonstration of the Ultra Light (ULE) encapsulation on the forward channel using realistic traffic models and service scenarios. In addition, to fully provide a full networking solution, the study has also evaluated the encapsulation on the return channel. This has looked at performance and address quality of service issues. Since network performance cannot be abstracted from the network architecture the study has also evaluated a number of realistic end user scenarios. Hence the result of the study has provided a comprehensive assessment of the new encapsulation and provides a roadmap for future implementation.

IETF background:
In January 2004, the IESG approved the formation of a new IETF working group (in the Internet Area) to develop protocols for the transmission of IP packets over MPEG-2 transmission networks (the Charter of this WG is included in the attached appendix). The activities conducted under this project provided a direct contribution to the formation of this WG.
The ipdvb working group (WG) aims to build open standards above the ISO MPEG-2 Transport Stream, and will, wherever possible, use methods that are compatible with the range of deployed systems (ATSC, DVB, etc) to provide the IP services. This goal includes definition of a new encapsulation scheme that was developed within this project and specification as an Individual Internet Draft, ULE (see next section). ULE is has been adopted as the WG encapsulation method. Following adoption of the ID as a WG work item, it will undergo working group review, review by the Internet Area Chairs and finally be submitted to an email last call to both the WG and IETF. This is the normal standards-track procedure and will take 6-12 months to complete. Following this, the document will receive IESG and IANA review, and after refinement of the document, the final document is expected to be published as an RFC by the RFC Editor.

The WG is also tasked to provide a set of supporting protocols to provide/support the general process of address resolution. QoS announcement/control protocols are anticipated as future work items, or as extensions to the core address resolution protocol.

2 Overview of status of standardization efforts relevant to IP over DVB

The Digital Video Broadcasting Project has been established since 1992, with the purpose of developing standards for digital television systems. The transmission standards are based on the ISO MPEG-2 standard [S11] and are applicable over a wide range of distribution technologies. Since the data being transferred is all digital, DVB systems are capable of being used for all types of information that can be digitised. The original emphasis was on TV signal broadcasting, carrying both a video and audio component, but the technology has increasingly been used for data delivery as well. Particular strengths of interest are the broadband and broadcasting aspects.

In order to support IP (Internet Protocol) based services, which is increasingly becoming an essential requirement, a return channel needs to be provided as well. The DVB reference architecture (see Fig. 1 below, [A35]) defines the broadcast forward channel independently from the return interaction channel, allowing the return channel to be implemented over a completely different network and technology than that used for the forward channel. Signalling in the forward direction may be either out of band, using the Forward Interaction Path, or in band, embedded in the broadcast transport stream as defined in [A35]. A number of standards in this area have already been produced and as a result, the support and demand for "IP over DVB" continues to increase. This in turn has led to an increase in the amount of standardisation activities related to this area.

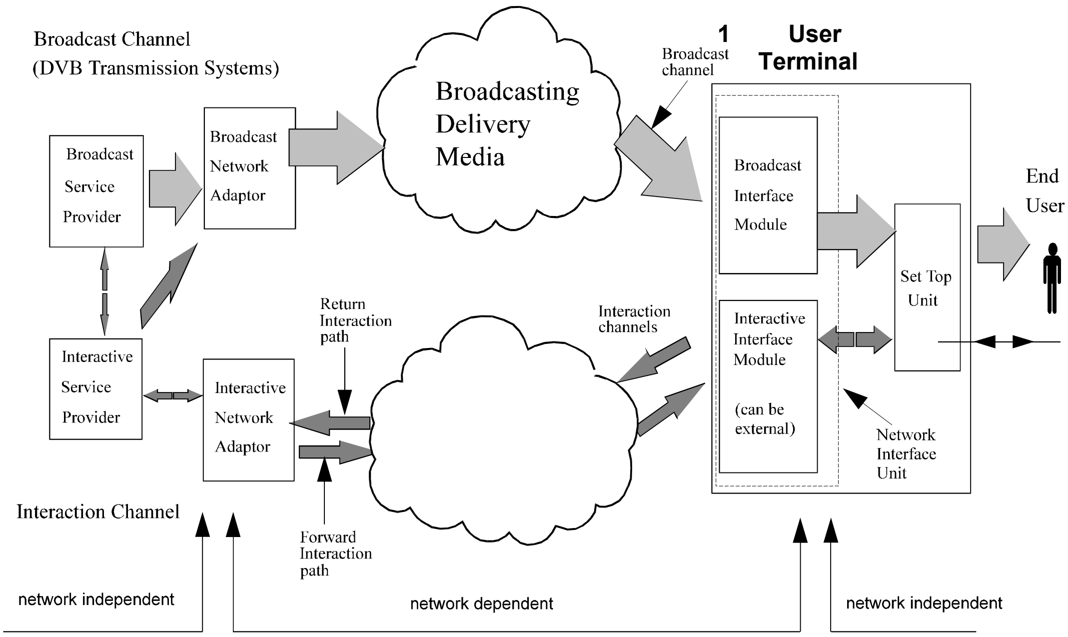


Figure 1: DVB system reference model for interactive services

Table 1 below summarises the various standards bodies investigated in this study. The relevance to the work of the present study is briefly described, with references to the section of the report D1.1 where more information can be found. Suggested actions on those participating in the current study or follow-ons hereof are also listed in the "Impact" column.

Body	Working Group	Section reference	Impact on the IP Encapsulation Study
MPEG	MPEG-2	2.1 – 2.4	Stable standard, basis for IP transport over DVB, new encapsulation must be compatible with this.
IETF	IP-DVB	3.1	Discussions in this forum should be reflected in the encapsulation proposed in this study. Conversely, output from the study should be submitted as an Internet Draft to this group for approval.
	AVT	3.2	Working on support for MPEG-2 over IP networks. Applicable if considering IP over MPEG-2 over IP.
	IPCDN	3.2	Coordinating to avoid overlap with IP-DVB.
	MMUSIC	3.2	Working on Service Discovery and Selection, most relevant for MPEG over IP.
	MSEC	3.2	Working on Multicast Security key exchange protocols for broadcast media.
	ROHC	3.2	Looking at efficient encapsulation of small packets in mobile environment. Output from this study could be relevant.
DVB-CM	IP Datacast	4.2.1	Focus on wireless and mobile IP over DVB-T/X. If proposed encapsulation is to be considered, it must support the power saving transmission mode to be defined.
DVB-TM	DVB-S	4.4.1	The baseline for satellite transmission, including protocol stacks for transmission of IP (either ATM/AAL5 or MPE/MPEG). New encapsulation must be compatible with these formats.
	DVB-S2	4.4.2	Defining new generation of satellite transmission scheme, with adaptive modulation and coding. New encapsulation must be compatible with this transmission scheme. Due to inherent flexibility of DVB-S2, this should not be an issue.
	Return Channels Group	4.5	Main work on cable and LMDS systems. Specify out of band interactive link based on ATM/AAL5, while in-band forward link is MPE/MPEG-2. New encapsulation could be applicable for forward link as for DVB-S. On interactivity link would need extension of the standard to allow non-ATM based connections. Group currently inactive.
	DVB-RCS	4.6	The open standard baseline for interactive satellite systems. Uses DVB-S forward link, see comments above. On return link, new encapsulation immediately useful for MPEG option, not applicable to ATM. With reawakening of DVB-RCS, may be possible to include in next revision of standard.

	GBS	4.7	Working on mechanisms for both streaming and non-streaming data. Will be compatible with existing DVB mechanisms where applicable. An obvious group to which the new encapsulation scheme should be submitted.
	IPDC	4.8	see comments under DVB-CM, IP Datacast above.
	IPI	4.9	Working on DVB over IP, so not directly applicable. May be applicable if considering IP over MPEG-2 over IP. Should monitor work on service discovery and selection.
ETSI	BSM WG	5	ETSI will eventually be asked to approve any DVB accepted standard. As some of the BSM WG work may be affected by the encapsulation method, there should be ongoing liaison with this body.
Satlabs		6	Main body looking into interoperability aspects of DVB-RCS. This body should approve any proposed modifications to the DVB-RCS standard first.
ATSC		7	North American equivalent of DVB, (similar systems, but not interoperable). IP also normally sent by MPE, so new encapsulation should be equally applicable to ATSC. Liaison with ATSC should be initiated.
DAB		8	DAB is carefully following the work that has been done in DVB for data transfer and intends to be as compatible as possible. Differences may arise due to the different transport stream structure. The new encapsulation should be applicable. No action required, as DAB is not trying to be a leader in this field, but rather will follow DVB.
DOCSIS		9	DOCSIS encapsulation is well defined for the established cable network environment. No new developments are expected.

Table 1 - Summary of relevance of various standards bodies

Further information on this chapter can be found in document:
WP1000: D1.1 “Overview of status of standardization efforts relevant to IP over DVB”

3 ULE Spec

An Ultra Lightweight Encapsulation (ULE) [7] has been defined. This provides efficient and flexible support for IPv4, IPv6 and other network layer packets over networks built upon the MPEG-2 Transport Stream (TS). The encapsulation is also suited to transport of other protocol packets and bridged Ethernet frames. Examples of appropriate use include the Digital Video Broadcast (DVB) architecture, the Advanced Television Systems Committee (ATSC) system (ATSC; ATSC-G), and other similar MPEG-2 based transmission systems.

Such systems typically provide unidirectional (simplex) physical and link layer standards. Bi-directional (duplex) links may also be established using these standards (e.g., DVB DVB-RCS). Protocol Data Units, PDUs, (e.g. Ethernet Frames, IP datagrams for transmission over an MPEG-2 Transport Multiplex are passed to an Encapsulator. This formats each PDU into a Sub-network Data Unit (SNDU) by adding an encapsulation header and an integrity check trailer. The SNDU is then fragmented into a series of TS Packets) that are sent over a single TS Logical Channel (i.e. using a common PID value).

The current Multi-Protocol Encapsulation (MPE), defined by DVB is based on the MPEG-2 Control Plane (using DSM-CC). This scheme has seen widespread use in a variety of MPEG-2 transmission networks, including DVB-S. In contrast, ULE places packets directly into the MPEG-2 transport stream. DVB calls this "Data Piping".

ULE provides a significant functional benefit over MPE, allowing the encapsulation of a wide variety of packet types using a standard encapsulation header. The key characteristics of ULE are:

- a simple header with few fields - easing processing
- a frame type field - allowing the receiver to identify the protocol being transported
- a length field allowing packets up to 32 KB.
- a CRC-32 integrity check
- provision for future extensions (as and when required).

Demultiplexing of packets carried by ULE is performed using the ULE type field (which follows the IANA Ethernet assignments). Use of this field provides native support for IPv6, MPLS, and other network layer protocols. (The recorded datasets showed only IPv4 Unicast traffic.) ULE also provides an IANA controlled type-space reserved for the ULE protocol. One application of the type field is to support efficient layer-2 bridging. The ISP feed scenario is unlikely to require the use of bridging headers.

MPE provides optional TS Packet Packing (with an under-specified algorithm), whereas ULE is being developed with TS Packet Packing as the default mode of operation. Packing can provide a significant benefit for IP traffic (reducing overhead from 8-80% to 1-30% - the largest gain for the smallest packets). However, in some cases, the additional benefit may be small.

ULE reduces the SNDU overhead by 15-67% compared to the MPE/LLC encapsulation method. (The largest saving when the link destination address is suppressed). This is accompanied by a reduction in header complexity, reducing the 18 values carried in the MPLE/LLC header to only 3 or 4 in ULE. For most patterns of Internet traffic, this gives a transmission performance gain over MPE (of at least 1-2%). The actual gain depends on the amount of additional overhead added by MPE and is greatest (up to 50% for small IPv4 packets) when MPE LLC/SNAP and bridging headers would be required

MPE has been compared with ULE within a DVB-S style link. ULE provides transmission performance improvements and a significant functional benefit over MPE, allowing the encapsulation of a wide variety of packet types using a standard encapsulation header. MPE provides optional Packing (with an under-specified algorithm), whereas ULE is being developed with TS Packet Packing as the default mode of operation. Packing can provide a significant benefit for IP traffic (reducing overhead from 8-80% to 1-30% - the largest gain for the smallest packets). ULE may further improve transmission efficiency, because the ULE header is smaller than the MPE header. In all cases, the performance of ULE is better than that of MPE. While the strengths of MPE lie in its compatibility to the DSM-CC control plane and the large deployed user base; the strengths of ULE lie in its efficient support for a range of network protocols and it's potential to support new applications.

3.1 Comparison of MPEG-TS Packet Padding and Packing

The efficiency offered by Padding and Packing is dependent on the SNDU size, and hence the IP packet size. To compare the two schemes, the term "Transmission Efficiency" is defined, as below:

Transport Efficiency = Encapsulated Payload Bytes/ Total Bytes Transmitted

The overhead introduced by UDP is an additional 8 bytes of header per IP packet. The figure below shows MPE with TS Packet Padding and Packing for a range of UDP packet payload sizes.

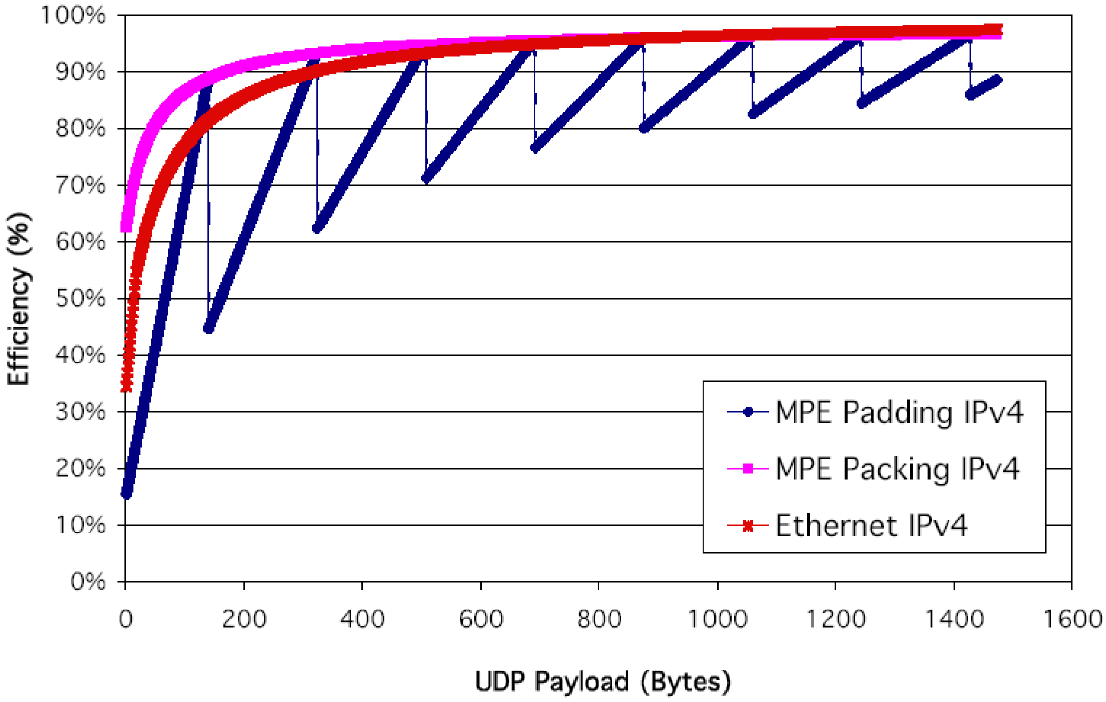


Figure 2: Calculation of Transmission Efficiency as a function of packet size.

MPE with TS Packet Padding and Packing both offer good efficiency for large UDP payloads. Packing is more efficient than Padding, although for a range of (larger) selected UDP payload sizes the two are equivalent. The results for TCP are almost identical: The UDP header is 8B whereas the TCP header is 20B, so the results for TCP can be deduced by adding 12B to the x-axis. For large payloads, the efficiency of MPE is comparable with that of Ethernet. Ethernet is less efficient for small packets since it was not conceived as a wireless technology. MPE with Padding has an efficiency that varies with a saw tooth pattern (figure 2). Similar performance is also exhibited in other sub networks using fixed size packets/cells such as AAL5/ATM [13]. For MPEG-2, a larger TS Packet size (e.g. compared to ATM cell size) IP Traffic Characterisation exaggerates this (more padding bytes). With a smaller size, the saw tooth is much less severe, less padding is transmitted, but more sub network headers are required for the additional fragments.

This graph is constructed based on a number of assumptions: The IP Maximum Transfer Unit (MTU) is assumed to be 1500B (the maximum for traditional Ethernet). It is assumed that the IP packets form a continuous stream (i.e., the Gateway always has data ready to send – this impacts TS Packet Packing, and will be investigated further in a later work package). An MPE overhead of 16B is assumed (see table 1 below, and appendix A). The IPv4 header is 20B (i.e. neither IPSec nor tunnel encapsulations are used). The overhead due to MPEG-2 signaling and control data in the TS is also not considered.

Overhead (Bytes)	Encapsulation Header	Header Fields and Function
16	MPE	MPE, No LLC/SNAP Header No Ether type - Assume IPv4 (0x0800)
16+8 = 24	MPE	MPE, with LLC/SNAP Header Ether type allows use of other protocols – ARP, IPv6 etc
16+24 = 40	MPE	MPE, LLC/SNAP Header containing Ethernet bridging header, but excluding Ethernet FCS.
16+28 = 44	MPE	MPE, LLC/SNAP Header containing Ethernet bridging header with FCS.
8	ULE (D=1)	ULE omitting destination receiver address
8+6 = 14	ULE (D=0)	ULE, including destination receiver address facilitating routing and L2 filtering
8+14 = 22	ULE (D=1)	ULE, containing Ethernet bridging header, but excluding Ethernet FCS.
8+6+14 = 28	ULE (D=0)	ULE, including destination receiver address and Ethernet bridging header, but excluding Ethernet FCS.

Table 2: Summary of SNDU overhead

For IPv6, the basic IP packet header size is 40B (twice as large as IPv4). While IPv4 options are seldom used (and are ignored in these calculations), IPv6 networks may reasonably be expected to use IPv6 extension headers (although to simplify analysis these are also ignored). The results for IPv6 also resemble those for IPv4. This larger IPv6 header leads to the MPEG-2 TS efficiency for smaller IPv6 payloads being higher for the same size of IP/UDP payload data.

Further information on this chapter can be found in document:
WP3100: Performance of ULE

4 Characterization of TCP/IP Traffic

The D1200.1 report provides analysis of a set of example TCP/IP applications and characterizes the traffic they produce with respect to the IP packet Length. This is the key parameter that influences encapsulation Transmission Efficiency. The outputs of this work package are a series of tcpdump files, corresponding to each of the types of traffic, and analysis of the distribution of packet sizes observed for each traffic class. To provide results that are not sensitive to the sampling interval or duration over which packets are captured, this data is presented in the form of a Cumulative Probability Density (CPD) function.
The report also describes the Multi-Protocol Encapsulation (MPE) and the Ultra-Light Encapsulation (ULE) and compares their header efficiency. This analysis will be completed in a subsequent Work Package using a self-designed JAVA tool.

The information presented in the following paragraph is relating to the packet sizes produced by a range of applications. The packet captures of protocols running over TCP (smtp, http and ftp) were gathered from the Electronics Research Group (ERG) mail and www server at the University of Aberdeen. These are Sun servers running the Solaris Operating 8 system (SunOS 5).

The forward traffic (from the server to the client) generated by TCP bulk applications consists mainly on large segments, that is packets of 576 B or 1500 B (depending on the use of PMTUD and the maximum PMTU). The presence of tunnels and Virtual Private Networks has introduced a variety of PMTU within the general Internet. Future deployments of links with increased MTU and evolution of PMTUD are likely to raise this value over the next decade. The congestion window dynamics and the TCP client ACK policy largely determine the inter-arrival time of IP packets carrying TCP DATA.

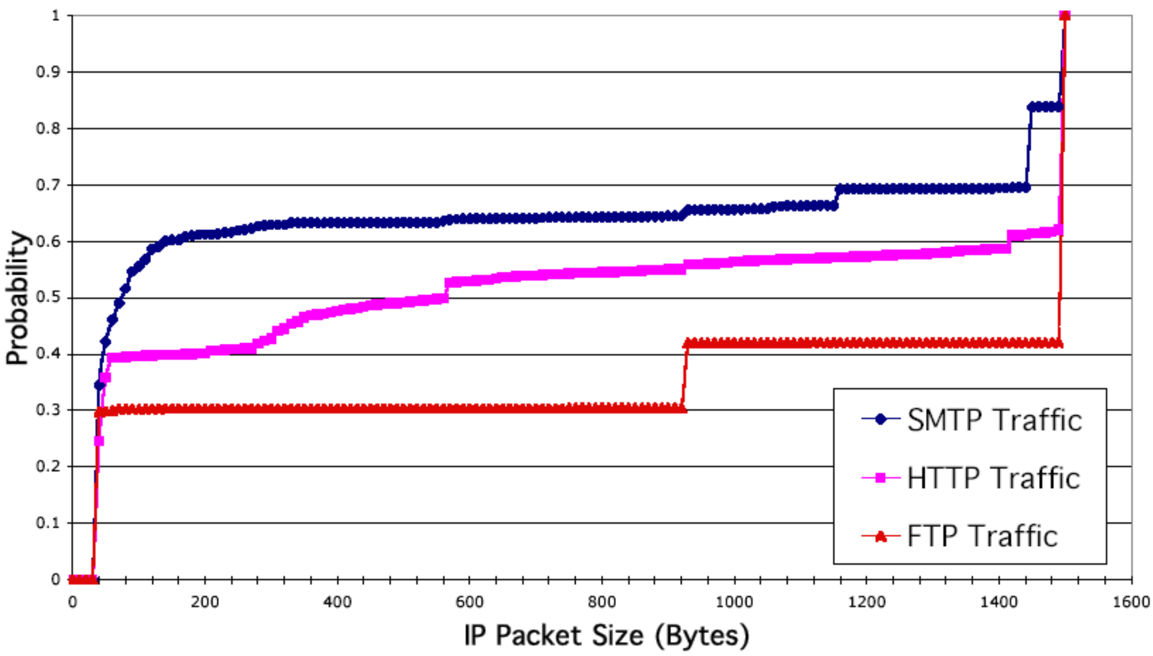


Figure 3: Summary of CPD of IP Packet size for TCP Examples

A number of distinct trends can be observed in the three sample sets. All show a large proportion of IP packets of size 40 B and 1500 B. There are also a number of other common packet sizes, such as 932 B for ftp. Other common sizes were observed during HTTP exchanges: 576 B (minimum IPv4 MTU) and 1420 B (a common maximum packet size for IPSec when used with a Virtual Private Network (VPN) tunnel). Interactive TCP applications can generate arbitrary size packets (even with no data payload). The distribution of packets lengths is a function of the application and/or user behaviour and is not easily characterised. The return traffic (returned by the recipient of the data transfer towards the source) consists mainly on minimum sized segments, that is packets of 40 B or slightly larger (if TCP options are used, such RTTM). Some TCP control packets (such as SACKs) can be larger. However, all TCP control packets are small compared to application level control messages (e.g. HTTP GET Requests). The latter can rise to many hundreds of bytes. The TCP client ACK policy largely determines the inter-arrival time of IP packets carrying TCP ACKs.

Further information on this chapter can be found in document:
WP2200: characterisation of TCP/IP traffic

5 ULE Performance Analysis

ULE reduces the SNDU overhead by 15-67% compared to the MPE/LLC encapsulation method. (The largest saving when the link destination address is suppressed). This is accompanied by a reduction in header complexity, reducing the 18 values carried in the MPLE/LLC header to only 3 or 4 in ULE. For most patterns of Internet traffic, this gives a transmission performance gain over MPE (of at least 1-2%). The actual gain depends on the amount of additional overhead added by MPE and is greatest (up to 50% for small IPv4 packets) when MPE LLC/SNAP and bridging headers would be required

Analysis of MPE/ULE performance with JAVA tool

We define the Packing Threshold as the period of time that the Encapsulator is willing to defer transmission of a partially filled TS-Packet to accumulate more SNDUs, rather than use Padding. After the Packet Threshold period, the Encapsulator uses Padding to send the partially filled TS-Packet. A Packing threshold of zero is equivalent to Padding.

Analysis of the data showed an increase in Packing efficiency, with increasing Packing Threshold. A Packing Threshold of 20-40 ms provided a good comprise between time and transmission efficiency. Figure 4 shows the transmission efficiency for ftp data using MPE with Packing (according to the ULE-Specified technique). It is interesting to observe that using MPE with Padding, only a very small proportion (i.e. 0.35% for the http sample) of IP packets fitted exactly into an integer number of TS Packets (i.e. no Padding required). The graph shows that a Packing Threshold greater values 20ms produced an negligible gain to warrant a further increase in transit delay. In the remainder of this report, the Packing Threshold was therefore chosen as 20ms.

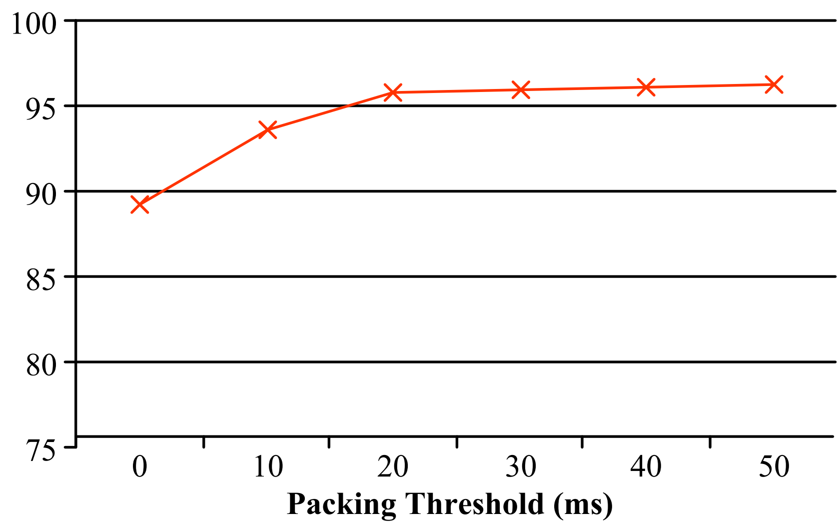


Figure 4: Packing Threshold against Transmission Efficiency (%).

Table 3 tabulates the transmission efficiency for each application data set (in the direction of forward transmission). Results are provided for the default MPE Encapsulator configuration where each SNDU is Padded (the % of TS-Packets/Cells that are Padded is also indicated).

The Performance of MPE with Packing is also shown, both for the default case, and the case where LLC/SNAP encapsulation is used. Mixed IPv4/IPv6 networks requiring a type field require this additional overhead when using MPE. The final two columns show the transmission performance of the two ULE modes: with a link layer address (D=0) and with the link layer address omitted (D=1).

Data is not presented for MPE or ULE where a MAC source address is required and/or bridged L2 networks are used, although the SNDU overhead in these cases are summarised in table 1.

Table 3 compares performance for each of the individual protocols in each captured data set. Columns are provided for MPE with Padding (showing transmission efficiency and the percentage of incomplete, i.e. Padded TS Packets), and the two MPE encapsulation methods – basic (IPv4 only) and LLC (with EtherType). The final two columns show the performance of the two ULE encapsulation schemes (D=0, and D=1).

	Transmission Efficiency (%)					
	MPE Padding		MPE Pack	MPE LLC	ULE (D=0)	ULE (D=1)
	TSPac Padded	Efficiency				
MPEG-4 HQ	14.9	88.8	91.8	88.8	92.6	95.9
MPEG-4 LQ	17.6	87.5	98.2	97.4	98.5	99.1
MP3	13.2	83.9	98.5	97.8	98.7	99.2
H.261	19.6	87.0	98.1	97.1	98.3	99.0
PCM	50.2	53.3	92.5	89.2	93.4	99.1
GSM audio (20ms)	100	39.7	81.2	75.2	83.9	90.1
GSM audio (40ms)	100	57.5	86.8	81.4	88.2	92.9
HTTP	14.4	89.2	98.6	97.9	98.7	99.3
FTP	11.8	89.8	98.8	98.3	99.1	99.4
SMTP	19.5	84.2	98.1	97.0	98.2	98.9

Table 3: Transport efficiency for sample data – forward path

In the following discussion, we assume the baseline of a multi-protocol link, that is for MPE we assume LLC/SNAP is employed. To ensure fairness in the comparison with ULE, the same (ULE) Packing algorithm is used for both MPE and ULE.

The UDP-based multimedia applications are presented in the upper part of the above table. The video-based applications and MP3 streaming content utilised a range of (larger) packet sizes, and showed similar transmission efficiencies. Packing for MPE improved transmission efficiency by 0-36% dependent on the application. Packing for ULE improved transmission efficiency by 4-40% dependent on the application.

The three real-time audio codecs (PCM and GSM) all emitted small packets, and showed considerable improvement in performance when Packing was used. The GSM traffic in particular consists of fixed sized RTP media payloads, all of which were less than a TS-Packet payload in size (i.e. when Padding is used, 100% of TS-Packets were Padded).

The lower portion of the table shows performance for the TCP-based applications in the forward direction. The TCP-based flows are characterised by large packets in the forward direction, and show a performance gain when Packing is used (~8% for FTP and HTTP, and ~13% for SMTP using MPE; ~10% for FTP and HTTP, and ~15% for SMTP for ULE).
The following table provides data for the traffic types commonly encountered in the return direction (i.e. towards a content owner).

	Transmission Efficiency (%)					
	MPE Padding		MPE Pack	MPE LLC	ULE (D=0)	ULE (D=1)
	% Cells Padded	Efficiency				
PCM	50.2	53.3	89.1	85.6	90.8	94.7
GSM audio (20ms)	100	39.7	81.2	75.2	83.9	90.1
GSM audio (40ms)	100	57.5	86.8	81.4	88.2	92.9
RTCP	77.0	58.9	87.1	82.1	89.9	92.7
HTTP ACK	83.5	39.8	84.5	78.5	86.2	91.6
FTP ACK	100	21.8	71.4	62.5	74.1	83.3
SMTP	48.6	65.5	93.8	91.1	94.6	96.8

Table 4: Transport efficiency for sample data – return path

The multimedia traffic is bi-directional, these data sets therefore exhibit identical performance to those in table 3.

RTCP is a control protocol used by some RTP-based multimedia codecs. Although many codecs generate RTCP packets, these typically contribute only a very small total of the traffic on actual Internet links.

The traffic generated by the TCP-based applications (HTTP and FTP) primarily consisted of TCP ACK and control packets (i.e. packets of 40 B or slightly larger if TCP options were used, such RTTM). Some TCP control packets (such as SACKs) can be larger. However, all TCP control packets are small compared to application level control messages (e.g. HTTP GET Requests). The latter can rise to many hundreds of bytes. The data above suggests that Packing can offer significant efficiency benefits over Padding; the gain for MPE using Packing is approximately 40% higher transmission efficiency. The improvement rises to 52% with ULE (D=0) and 62% with ULE (D=1).

The case of SMTP must be considered separately. The SMTP traffic consists of both data (uploading messages) and ACKS (from down-loading messages). This traffic resembles that encountered on the return link – the actual traffic distributions will depend upon the way the email client is used and the ratio between down-loaded and up-loaded emails.

Finally, it should be noted that all of the data analysed was for a single packet flow using only one application. Multiplexed packet flows have been analysed in a separate report, and are not considered in this report. The following figure (figure 5) presents a summary of the data presented in the previous tables.

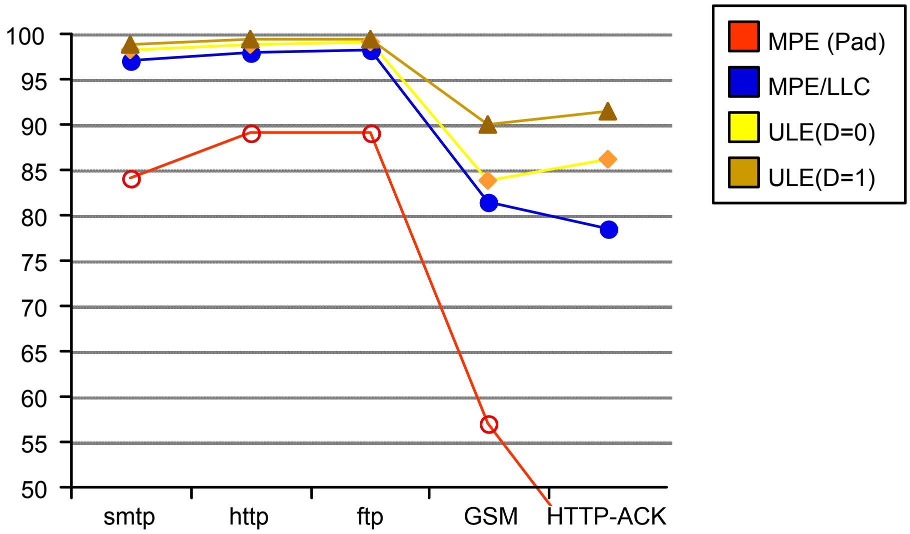


Figure 5:: Shows Packing efficiency for various protocols

The summary plot compares the transmission efficiency of four key protocols. It can be observed that in all cases, the performance of ULE was better than that for MPE (≤40% better than MPE with Padding and ≤4% better compared to the best MPE method). If the ULE MAC address was omitted (D=1), transmission efficiency of ULE is further improved by 0.3-6%.

The current Multi-Protocol Encapsulation (MPE), defined by DVB is based on the MPEG-2 Control Plane (using DSM-CC). This scheme has seen widespread use in a variety of MPEG-2 transmission networks, including DVB-S. This report compares performance of this with the Ultra Lightweight Encapsulation (ULE), a new encapsulation protocol, which is a work item of the ipdvb WG. Unlike MPE, ULE places IP packets directly into the MPEG-2 transport stream. Demultiplexing of packets carried by ULE is performed using the ULE type field, this field provides native support for IPv6, MPLS, and other network layer protocols. ULE also supports efficient layer-2 bridging.

ULE provides transmission performance improvements and a significant functional benefit over MPE, allowing the encapsulation of a wide variety of packet types using a standard encapsulation header. In MPE Packing is optional (with an under-specified algorithm), whereas with ULE TS Packet Packing is the default mode of operation. Packing can provide a significant benefit for IP traffic. As in any scheme that reduces the per-packet overhead, the gain is largest for small packets. The actual gain depends on the amount of additional overhead added by MPE and is greatest (up to 50% for small IPv4 packets) when MPE LLC/SNAP and bridging headers would be required. ULE may further improve transmission efficiency, because the ULE header is smaller than the MPE header. This gives a performance gain over MPE (of at least 1-2%).

In all cases, the performance of ULE is better than, or at least comparable to, that of MPE. While the strengths of MPE lie in its compatibility with the DSM-CC control plane and the large deployed user base; the strengths of ULE lie in its efficient support for a range of network protocols and it's potential to support new applications.

Further information can be found in the document:
WP3100: Performance of ULE

6 Header compression with ULE

This chapter describes a preliminary investigation of the RObust Header Compression (ROHC) method and examines the potential benefits use when used in combination the Ultra-Light Encapsulation (ULE) defined by the IETF ipdvb WG. The ULE encapsulation format does not currently provide a code-point for ROHC, but the design permits future extension of the protocol using the Sub-network Data Unit, SNDU, type field. ULE also provides the other two-prerequisites for ROHC: a mandatory SNDU Cyclic Redundancy Check (CRC) and a SNDU length field. ULE currently has no link configuration protocol to negotiate the profile or configuration of the compressor and de-compressor, although such a protocol may yet emerge as a part of the “address resolution” work item within the ipdvb WG.

Using analytical approaches, it has highlighted the performance gains to determine the predicted transmission efficiency for several types of TCP/IP Traffic. The analysis showed that a considerable performance gain was achievable when the traffic is characterised by small packets.

The above results make several assumptions, including that all packets are compressible, and therefore the achievable performance may be less than that estimated (especially for short-lived flows). However, the benefits of using ROHC are expected to increase significantly with the use of IPv6 (where the header is larger, but much more easily compressed). To determine actual performance, the ROHC protocol needs to be implemented to evaluate the trade-offs between increased transmission efficiency and robustness of the compression protocols.

No data has been presented for UDP-based multimedia traffic, although clearly the same methodology could be applied in this case. Analysis of traffic aggregates combining both UDP and TCP packet flows is likely to be much more complex, and would require development of new methods.

Given that ULE could be extended to support ROHC on operational satellite links, what would actually be required to allow this? Several key areas need to be addressed:

1) The ULE protocol will have to be updated to include an appropriate code-point or code points for ROHC.

2) The ROHC protocol (and specifically negotiation methods) will need to be specified for non PPP links.

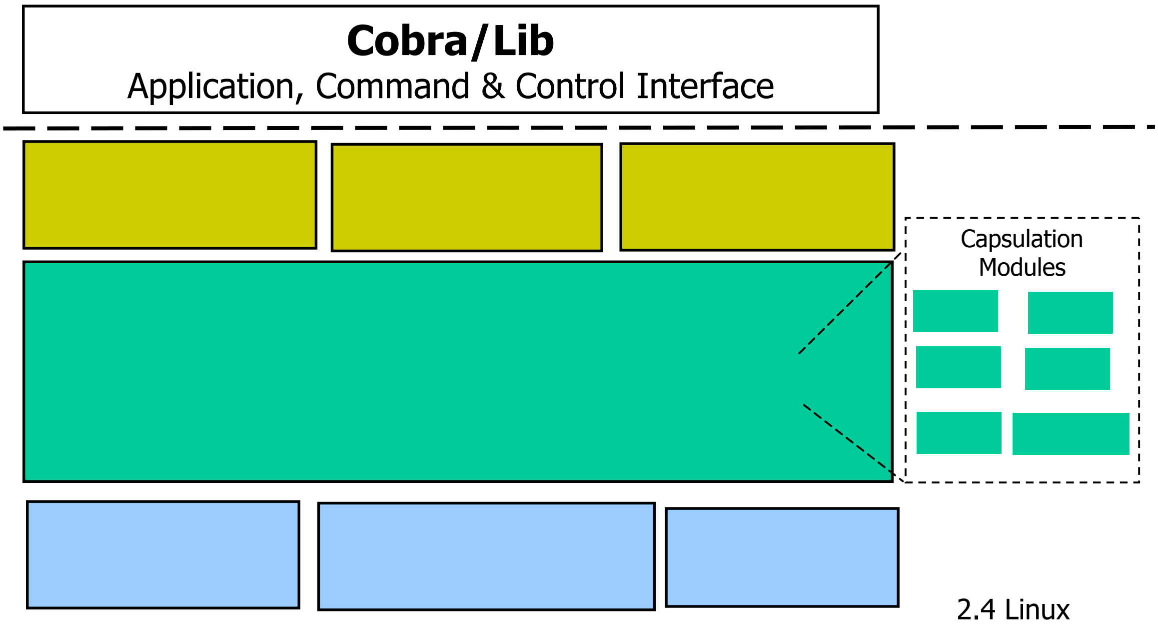
3) Most current uses of ROHC anticipate bi-directional communication. Use over unidirectional links may be desirable, especially for UDP-based applications and hybrid networks that have asymmetric routing. This is possible, but would require further study.

Further information can be found in the document:
WP4300: Header compression with ULE

7 ULE Implementation

The ULE implementation phase focused on the design and implementation of ULE encapsulator/ decapsulator.

The ULE encapsulator was implemented for the gcs IP/DVB gateway and the ODG gateway. These gateways utilize the Cobra/DVB driver stack and the PCI Cobra/i960 or the C-PCI *flex*:converger adapters, respectively. The Cobra/DVB driver features a very modular architecture, as shown in the following figure.

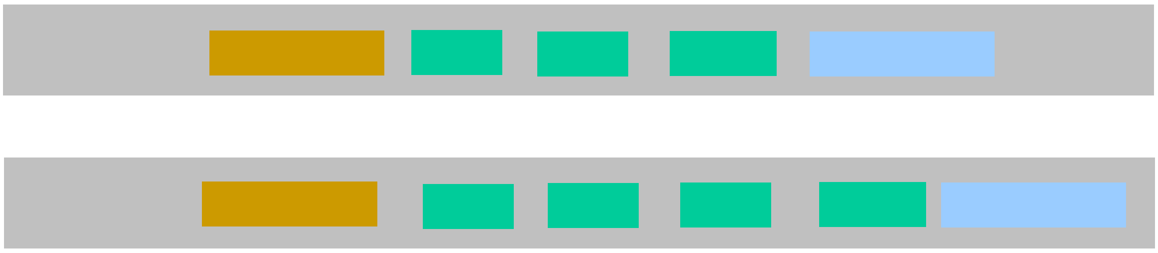


Each separate solid box represents a Cobra/DVB kernel module. Functionality that is independent of each other is separated into a module of its own. The Cobra/DVB Core module provides the basic management code, framework and runtime environment for the other modules. The Cobra/DVB Core can be accessed via a character device (Cobra/Char) or via a network interface (Cobra/Net).

Special sets of modules, in a way, are the capsulation modules, shown in the dotted box in the right part of the figure. They are used to build the capsulation chains. These chains then can be attached to network interfaces (Cobra/Net) or character devices (Cobra/Char), which act as data sources for the encapsulation process.

Encapsulation of payloads – network packets via Cobra/Net or user data via Cobra/Char – is achieved by sending the payload through a so-called "*caps chain*". A caps chain can be seen as a transformation that is applied to the payload (attaching additional headers and/or trailers, segmentation into fixed sized cells …) before it is handed off to the actual hardware device responsible of putting the bits onto the wire.

A caps chain is executed by calling certain capsulation modules in a specified order, where the output of the previous capsulation module becomes the input of the next one. The actual number and order of capsulation modules that make up a certain caps chain can be configured at run-time; the chain is then referred to by its (user defined) name. In the framework of the IPEncaps project two new chains, called TxULE/RxULE were added to the Cobra/DVB driver. The following figure gives an illustration of the TxULE and TxMPE3e chain. The data (i.e. packet) flow is from left to right, starting at the Cobra/Net module, which gets the packets from the Linux kernel's networking stack.



The chains are implicitly terminated by a "data sink", which, most likely, is a hardware device like the Cobra/i960 or the *flex*:converger card.

The ULE decapsulator has been implemented for two COTS DVB-S receiver cards. The Technotrend DVB-S card (also known under different names such as Hauppauge DVB-S, or Siemens DVB-S card) is the most frequently used digital satellite receiver card in Europe. The card is available in two versions: 1) the TT-PCline premium family has an integrated hardware MPEG-2 decoder and is mostly used for watch digital TV or building personal digital video recorders while 2) the TT-PCline budget family is optimized for data reception. Technotrend GmbH offers Windows drivers for their TT-PCline cards. Linux drivers are also available from the LinuxTV open source project (www.linuxtv.org). In the IPEncaps project gcs used the LinuxTV 1.0.1 driver for implementing the ULE decapsulator. Because of the modular driver structure, this was straightforward. The TS demultiplexer was reprogrammed to deliver plain TS cells to the ULE decapsulator routine. The ULE decapsulator encapsulates the TS cells according to the pseudo algorithm shown below. The full implementation can be found in dvb_net.c in the LinuxTV driver.

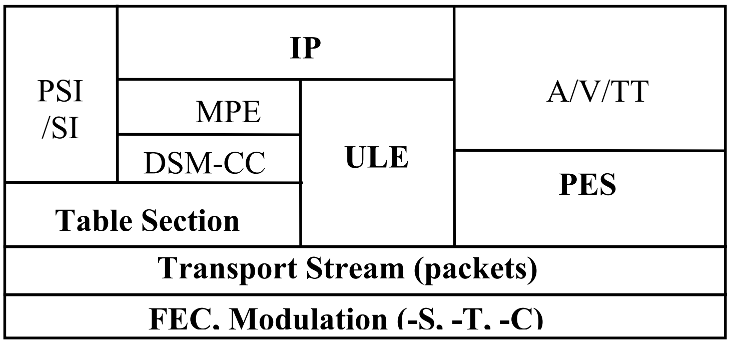
The second implementation of the DVB-S decapsulator was for the Pent@value card from Pentamedia Corp. The main difference in terms of hardware between the Pent@value and the Technotrend card is the quality of the DVB-S tuner. The tuner of the Pent@value card locks to SCPC carriers with symbols rates as low as 2 MSymbols, while the Technotrend card needs roughly 6 MSymbols for a stable receiver lock. Hence, only the Pent@value card was used for the interoperability tests via the ESTEC satellite carrier.
Pentamedia Corp offers both Windows and Linux drivers for the Pent@value card. The Linux driver is available partly in source code (the HW depended modules are provided in binary format). The implementation of the ULE decapsulator has to go into the HW depended part of the driver. Therefore, a NDA with Pentamedia Corp. was necessary to get the full source code. After studying the complete source code of the driver it turned out the integration of the ULE decapsulator was not possible without additional knowledge of the hardware. This additional information was not provided by Pentamedia. gcs therefore chose to port the hardware dependent module form the original Pent@value driver to their Cobra/DVB driver stack. As a result, the Pent@value card is now fully integrated into the Cobra/DVB driver stack and supports all functionality and encapsulations provided the framework.

To summarize, the ULE implementation phase results in the following driver modules:

- TxULE and RxULE caps chains for the Cobra/DVB driver stack
- ULE decapsulator for the LinuxTV driver stack. All modifications made by gcs will be released into open source at the end of the project.
- ULE decapsulator for the Pent@value card. This was achieved by integrating the Pent@value hardware in to Cobra/DVB driver stack. The driver will be available in binary format for the Linux 2.4.24 kernel.

Impact on IRD Design

Conceptually the standard Multi Protocol Encapsulation (MPE) is situated as an extension on top of the Digital Storage Media Command and Control [9] using table type 0x3E (DSM-CC Sections containing private data) with DSM-CC stream type 0x0A (Multi-protocol Encapsulation). The table sections also carry other DSM-CC data, such as carousels, and the Program Specific Information/Service Information (PSI/SI) [10]. Therefore it is required, that Integrated Receiver Decoder (IRD) hardware, firmware, or software is able to reassemble and analyse sections.
The Ultra Light Encapsulation (ULE) directly works on the Transport Stream layer and has to reassemble Protocol Data Units (PDU) using a similar semantics as the section table, interpreting a dedicated pointer field to obtain the start of a payload in a transport packet, when the Payload Unit Start Indicator (PUSI) is set to 1.



The implementation of the ULE decapsulator requires the hardware/firmware of an IRD card to permit access to full TS cells (including the TS header). Usually, all DVB card allow access at this level. The implementation of the ULE decapsulator is straightforward and follows to pseudo code give in the previous section.

Having access to TS cells does so far not have any impact on the IRD design. However, the way how transport stream packets are transferred from the IRD card memory to the PC host memory does have influence on the driver performance, specifically on less powerful PCs.

The simplest transfer method is to generate one interrupt per transport stream packet and to transfer one transport stream packet per DMA. At higher rates this introduces a substantial amount of interrupts and DMA transfers.

The better method is to transfer a larger DMA buffer at once (holding several or many transport stream packets) and generate interrupts only per DMA transfer. A certain time-out is necessary to transfer after a given time also non-full DMA buffers. With this approach fewer transfers and less interrupts are necessary at higher rates.

The optimal solution would be to reassemble ULE protocol data units on the IRD card in the IRD card's firmware and only transfer whole ULE payloads. Since the semantics of the ULE decapsulation is to a certain extent very similar to the section reassemble it should be straightforward to implement this in the IRD card's firmware. However, the firmware of the IRD cards is mostly confidential and proprietary and only the IRD card manufacturer can do such modifications.

PID filtering remains unchanged on the IRD card firmware. Since ULE has to encapsulate payloads on one PID for corresponding packets, there is no need to change this.

Hardware address filtering cannot be performed without changing the IRD card's firmware, because ULE puts the MAC addresses at a different location compared to MPE. However, the MAC addresses are not spread over the header anymore, and the hardware filtering should be easier to implement. In most cases ULE will require the driver of the host PC to filter addresses, specifically if network addresses are to be filtered.

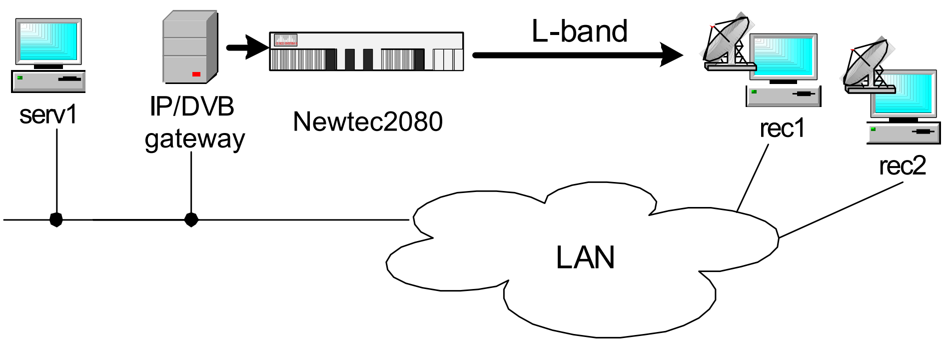
A comparison of the ULE and the MPE decapsulator at a processing level is very hard as some of the IRD cards have a table section reassembly mechanism already is implemented in their hardware/firmware. Hence, the code required for MPE decapsulation is minimal; i.e. it just requires checking the table-id for a value of 0x3e, to strip of the table section header and pass the received SNDU as an Ethernet frame to the operating systems network stack. The Technotrend firmware supports section filtering in firmware. On the other hand the Pent@value firmware does not support section filtering and hence the ULE decapsulation code is no more (and may be less) "expensive" than the MPE decapsulation code in terms of processing requirements in the host machine.

Further information can be found in the technical Note:
D2.1 on "implementation of ULE encapsulation" as well as in Technical Note
D2.2 "Prototype Software Description including a User Manual for IRD and IP/DVB gateway and initial Test Report"

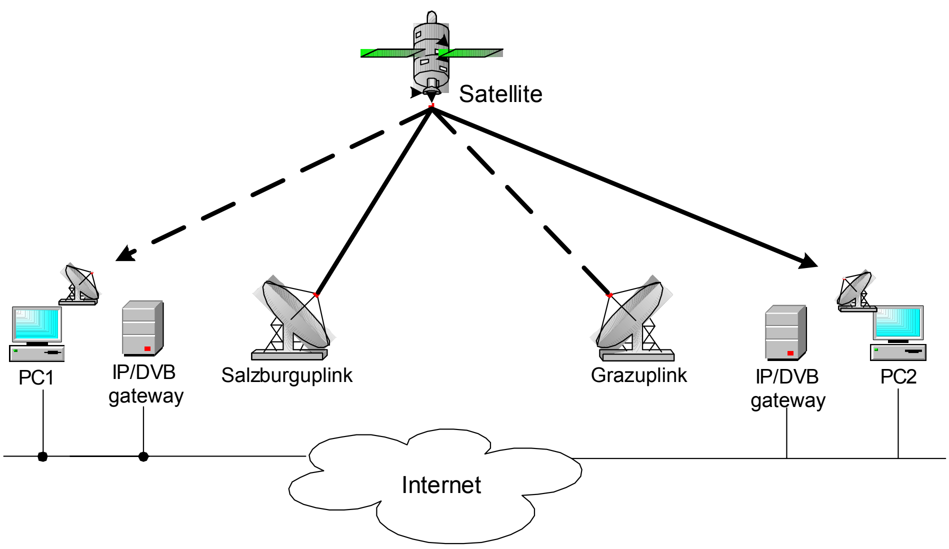
8 Interoperability Test

A parallel team with the companies 6Wind and IABG developed also an implementation of a ULE.The interoperability tests have initially been carried out in a laboratory test bed. The first set-up only involved the encaps/decaps of the IP/DVB gateway driver.

The second set-up involves the Technotrend and Pent@value drivers on the forward link.



Afterwards the tests have been performed using the two available uplink stations in Salzburg and Graz utilizing ESA space capacity. Both in Salzburg and Graz one IP/DVB gateway and one receiving PC have been installed. The Salzburg gateway was sending MPE towards Graz, whereas the Graz gateway was sending ULE towards Salzburg. **Proper operation was verified**.



9 ULE Modelling (Simulation)

9.1 Traffic Characterisation

UoA provided a set of traffic characterisations that illustrate the typical IP packet lengths produced by some well-known Internet applications. The outputs of this work package were a series of tcpdump files, corresponding to each of the types of traffic, and analysis of the distribution of packet sizes observed for each traffic class. To provide results that were not sensitive to the sampling interval or duration over which packets are captured, this data has been presented in the form of a Cumulative Probability Density (CPD) function**.**

9.2 Simulation Network Model

The network model used for the simulation is a simple satellite access network. Traffic is transmitted via the satellite network between a subnetwork behind the source terminal to a subnetwork behind the gateway. There is no Internet wide routing as queuing and overall network performances are being monitored. De-encapsulation is implemented to get a flavour of end to end performance for comparison to WP3310 and to verify performance improvements in ULE due to packing of short TCP ACK packets. The model could accommodate future extensions such as multicasting, routing and QoS monitoring in the future and support full mesh implementations and alternative topologies.

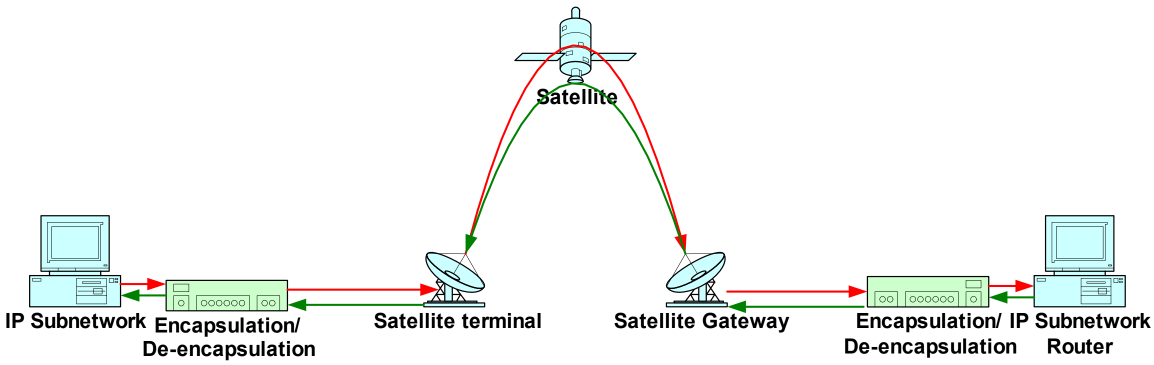


Figure 6 - Network Model

Encapsulation is performed according to a simple rule:

- Take a IP packet out of the queue
- Add MPE or ULE overhead and CRC to form a SDU
- Fragment the SDU into TS cells

For MPE the encapsulation node implements the standard ISO/IEC 13818-1 [10] and EN 301 192 V1.1.1 (1997-12) as well as section packing. These specifications restrict the number of new packet headers in an MPEG cell to one new SDU header for each TS cell (no use of packet length) which results in reduced efficiency for short ACKs and ACK bursts.

For ULE the two options, D=0 and D=1 have been implemented to simulate either a receiver with layer 2 filtering or alternatively a single receiver or a receiver that filters at layer 3. This is a decision that would be implemented by a gateway provider.

Finally the architecture supports limited QoS management in the form of timer and the possibility of multiple queues. For example, multiplexing small packets in a larger MPEG container may improve bandwidth efficiency but add some delays which in turn may break QoS contracts or lead to queue build-up; a timer will force the encapsulation to proceed even if packets are still sent to a non full link queue (indicating non empty IP queue).

9.3 Queuing and Performance Analysis

The queuing analysis was performed to investigate generic performance parameters between the ULE and MPE encapsulation. The basis of the analysis is a queuing model with vacations where for the DVB network the vacation is due to either an IP queue becoming empty or the encapsulation timer has expired because of lack of capacity or QoS parameter. The analysis was performed on small queues and arrival rates that are representative of the return channel but the model is generic. While the queuing cannot be a full representation of the reality, the model has nonetheless showed that under constant arrivals ULE will get its fullest performance improvements over MPE when the queues remain full, hence the arrival rate and the network load is fairly high.

9.4 Network Layer Analysis and Simulation Conclusions

The analysis and simulation of ULE (with and without destination) and MPE have been conducted on the return channel. The queuing analysis was based on well known work on queues with vacation, in this case either a vacation due to the end of encapsulation (timer expired or MAC is not available) or the queue being empty. What the simulation confirms is that packet loss reduction and bandwidth utilisation gains of ULE are dependant on traffic type, traffic load and encapsulation time as was demonstrated also by the queuing analysis. Going back to the scenarios of WP2100, ULE will have the best performance over the inter WLAN scenario especially for remote access. In those scenarios, interLAN traffic is dominated by asymmetrical TCP based protocols such as ftp: that are used for remote database access or multimedia transmission. This generates many ACK packets that are not efficiently encapsulated by MPE. For those scenarios the same goodput (application level throughput) is generated with lower MAC throughput (better utilisation). For other applications the gains are not as important. The simulation results also show the effect of the encapsulation time on performance, a parameter that could be varied in the future to study quality of service.

The results for CBR traffic indicate that for a 200 Bytes packet, typical of for example voice traffic, MPE and ULE without destination have a fairly steady overhead difference of 10-15% for traffic loads of 1 to 1.8 Mbps (Figure 7). This confirms the hypothesis that under steady state conditions and fairly high traffic rates the encapsulation gain will tend to approach the theoretical limits.

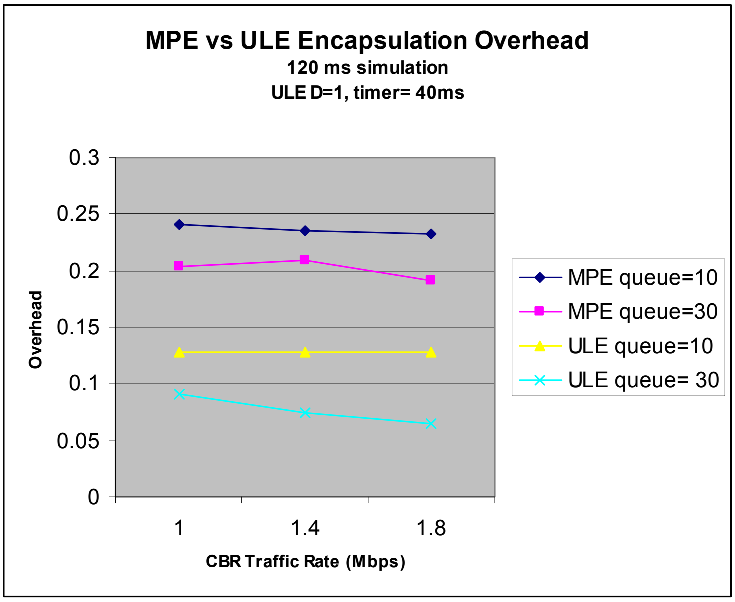


Figure 7: - Overhead for MPE and ULE

The simulation confirmed that ULE never under performed MPE and also showed that it added flexibility in the encapsulation.

9.5 End-to-End MAC Layer Simulation

OPNET simulations have been carried out to compare the end-to-end performance of carrying IP traffic over either ULE or MPE (with section packing) when used on a DVB-RCS return link. The results reflect the behaviour of the MAC layer BoD scheduling, which is responsible for the return link capacity management of the DVB-RCS network and its impact upon the performance of the two encapsulation schemes
The purpose of the simulations is to compare and evaluate the new encapsulation scheme (ULE) against MPE with section packing. This comparison addresses a comprehensive range of aspects and measures of performance and efficiency of the encapsulation schemes and also the interactions with and efficiency with respect to BoD. Statistics were taken at the MAC layer and relate to end-to-end application performance.
It can be concluded from download simulations, when small packets are encapsulated, that ULE is never worse than MPE with section packing in terms of TCP performance, performance variation and packet latency and jitter.

When network resources are constrained and when encapsulating small packets, ULE outperforms MPE with section packing in terms of TCP performance, performance variation and packet latency and jitter. Since operational networks are rarely over-provisioned, for economic reasons, ULE is expected to evidence these improvements in deployment.

When encapsulating small packets, TCP performance variation is especially reduced, providing a more consistent user experience across the entire user population with ULE and also a more consistent experience as load is increased. Combined with reduced packet latency and jitter ULE effectively offers an improvement in the quality of service experienced.

ULE can be said to provide negligible improvements for the encapsulation of medium or large packets versus MPE with section packing, when examining performance, performance variation and packet latency and jitter. In combination with the above conclusions for small packets, ULE indicates a greater utility in the return links of satellite systems, which are more frequently used to transport the TCP ACK stream and particularly for consumer systems where bandwidth use is typically less symmetric.

When comparing TCP performance, ULE employing VBDC can be seen to perform marginally better than MPE with section packing employing VBDC plus a single slot of RBDC. This illustrates that a more efficient capacity assignment method (VBDC) may be chosen by a satellite network operator to reduce costs but without sacrificing performance, if ULE is used.

When employing VBDC with a minimal CRA assignment to improve performance and perceived service quality factors such as packet latency and jitter, ULE is seen to maximise the effects of CRA as compared to using MPE with section packing.

Irrespective of the DVB-RCS capacity allocation type or combination used, ULE was observed to be at least equivalent and in most cases better than MPE with section packing in terms of TCP performance, performance variation and packet latency and jitter. In many cases it could be seen that the use of ULE allowed a more cost effective use of the return link bandwidth, taking advantage of lower cost bandwidth allocation mechanisms.

Hence, it can be concluded that ULE provides useful benefits and no disadvantages when used on the return link of a DVB-RCS system. As such it is expected to be of general interest to the manufacturers of DVB-RCS equipment and indeed to the entire DVB-RCS community.

9.6 ULE Flexibility

The ULE type field allows transmission of all types of frames supported by IEE 802.3 Ethernet, using the industry standard IANA-assigned Ethertype value. No modification is required to the protocol to support new protocols.

The ULE specification also reserves a set of unused values for link-specific code points. These code points may be used for testing, operator-specific packet types or for future specification of new standards-based encapsulations, such as the development of header compression (e.g. ROHC). A further use of these code-points has been proposed, to allow the encapsulation to provide "optional" extension headers a flexible and future-proof method for extension of the SNDU header. This style of operation was successfully included in the Ipv6 network header specification, and much could be inherited from this approach

Further information about the simulation can be found in the documents:
WP3320 Report: Link Layer Simulations
WP3310 Report: IP Encapsulation OPNET Simulation Report

10 Future Work

In proposing future work, care must be taken not to overlap with work conducted in industry and other standardisation bodies. In particular:

- IETF, where IP over DVB is going to be further developed, not specifically for satellite however.
- ETSI/BSM, where address resolution and QoS are on the agenda for future work; this will be based on a generic satellite architecture with DVB as one case study, pending members bring forward contributions.
- Satlabs, which is concerned by current industry implementations of DVB-RCS and interoperability between systems.

Based on this information, the follow up to ULE encapsulation work is foreseen building protocols and services above ULE, for the DVB satellite community and in a medium term (2005-2006) implementation and contribution to set of coherent IP-enabled standards. This includes resolution of addresses and other link properties, starting with an Informational RFC to identify design options and contrasting table-based (INT, MMT) and dynamic ND/ARP type protocols. The trade-off parameters need to be based on ease of implementation with MPE and ULE, as well as operational and network management interoperability. This work could become the basis for a set of standards-based uniform IP address resolution mechanisms over all DVB and broadband satellite systems.

The gain from the future work based on the simulation tools and implemented hardware developed in the current study is important. It will allow comparison of solutions (dynamic vs. table based, use of extension headers vs. table etc.) and suggest at least one proposed implementation that could be directly applied to the European satellite industry and selected operators. That implementation could become the basis of the BSM work (any BSM solution applied to DVB satellites would have to comply) as a proposed standard from ESA and the industrial consortium. It would undoubtedly lead to more IETF drafts and possible RFCs and extend the buy-in of t the industry at large.

Key areas recommended for future work include:

(i) *Progression of the ULE ID along standards track* – A small activity is necessary to complete the submission of the ID as an RFC. This includes working group calls, IANA considerations, implementation Reports and attendance at meetings (2 man months prior to the next IETF). This could be completed in parallel with other activities. UoA would be well-placed to perform this work

(ii) *Research, specification and definition of address resolution mechanisms relating to resolution of the MPEG-2 PID to/from IP address mapping function*. This work falls within the charter of the ipdvb working group, and requires detailed consideration of the INT and MMT approaches for DVB (see mapping consideration below). Marie-Jose Montpetit and UoA are co-authors of an ID providing a first attempt to work in this area.

(iii) *Research, specification and definition of address resolution processes relating to resolution of the IP address to/from the subnetwork NPA (MAC) layer 2 address.* For IPv6, this work must be coordinated with practical consideration of ND implementation and the implications of using ND within a satellite environment. This work falls within the charter of the ipdvb working group, and requires liaison with the IPv6 working group.

Applications of ULE (and MPE) within a globally routed IP multicast network. Important issues relate to the management of the service, and support for operational procedures to control multicast propagation – this work has been previously proposed to ESTEC, by UoA and was well-received. While the expertise to guide this work certainly exists within the PIM and MBONED WGs, the required practical work is to specify interfaces, procedures and good practice – which would not directly contribute to an RFC. This work could be co-funded with (i) or independently funded.

10.1 Encapsulation

The remaining work required to complete ULE centres on completion of the protocol specification – specifically to address the concerns raised by early implementers, to remove any ambiguity in the specification and to add any missing features.

The only proposed additional feature has been the suggestion of an extension header format (this is further discussed in appendix 1). This would provide a simple path to future upgrades and permit "safe" experimentation with new features (e.g. to support more dynamic protocols for address resolution, QoS and bandwidth brokerage). A standard extension mode would encourage the development of new services. The usage of extended headers to complement current service/session advertisement table also creates the opportunity for future work, linked to QoS and flow management as well as the definition of novel DVB services for switching satellites. Since extended headers allow dynamic management, this work could lead in an interesting trade-off between the loss of bandwidth due to the extension vs. the added efficiency in managing the DVB flows.

The primary focus should the completion of the standardisation activity, namely to complete the ID following adoption as a working group work item, and to progress it through working group and Internet Area Chairs review in preparation for the IETF last call process. This should finally result in RFC publication (targeted for 2004).

10.2 PID Mapping

Address resolution mechanisms are required to complement the encapsulation to provide a true edge to edge protocol. Address resolution occurs at two levels when using an MPEG-2 link to support DVB services: This section describes the mechanisms used to identify a DVB-level flow. The uni-directional address resolution mechanisms offered by DVB have evolved as needs emerged within the DVB community

Both the standard DVB Project Internet Notification Tables (INT) and the DVB-RCS Multicast Mapping Table (MMT) provide means to bind IPv4 of IPv6 addresses to DVB PIDs. The INT is the current standard and the address resolution features introduced probably offer a good match to the requirements when used to deliver additional internet content as part of an IP overlay. A particular strength lies in the ability to use standard MPEG-2 multiplexor functions that update SI tables as they remultiplex the PIDs used by a Transport Stream.

Weaknesses of the INT method are:

(i) The inherent complexity in supporting remultiplexing of PIDs

(ii) Since this is a sub-IP function, the same code has to be written by each developer for each new interface card.

(iii)The approach does not map well onto the use of multicast as a service offerings

Other issues to be considered include:

(i) impact of multicast scoping on addressing

(ii) impact of applications that use dynamic multicast addreses (such as ND, PIM)

(iii) Architectural issues – the INT method requires sub-IP functions to specify IP services, this gives no control of this mapping to an IP-level protocol, requiring ISP policy decisions to be implemented independently at the various levels.

(iv) There is no security/authentication method

An IP-level control protocol could counter these weakness and offers the opportunity of consideration and solutions to the issues described above. Although it could seem attractive to design a new protocol, the ipdvb WG has long realised that the intention should not be to replace the INT and MPE approaches (this co-existence was a key feature of the Charter negoatiations of the WG).

The first goal must therefore be a critical review of the INT approach, its strengths, issues and expected operation environment. This would be welcomed by the IETF as an ID for the ipdvb WG. Theipdvb WG has the goal of providing an Informational RFC on address resolution by 2005. This RFC will review the current practices and propose a consistent approach that unites the current static binding (in the cable industry), the standard table based approaches (INT)

Following this, there is a need to identify a candidate protocol to provide an IP-level mechanism. This would require protocol engineering, understanding of DVB and other MPEG-2 systems and contributions from the community at large. A dynamic protocol also relates to future work in ETSI/BSM on address resolution based on generic address interface. This is however out of the scope of the current study to address this issue.

The final stage would be to define the mapping(s) between the above and the under-lying DVB equipment (encapsulators, gateways, receivers, multiplexors, etc). It’s interesting to note that where necessary, a MPEG-level device could use the IP-level protocol to generate an INT (all the key information would be present at the IP-level – the converse is not necessarily true, since the IP-level protocol may be used to establish other subnetwork capability information, such as the mapping of QoS parameters and link layer service attributes (priority/pre-emption, use of compression, choice of encapsulation, differentiated services options, etc). More work in however necessary to evaluate exactly how these mechanisms interact. This is currently not addressed in IETF.

Next generation satellites supporting multi-spot beam also systems raise additional issues – namely how to associate traffic flows with switching parameters. These can be via address-resolution mechanisms, or via tags attached as L2 encapsulation headers. Both methods have merits, and hybrid approaches are also feasible. A key advantage of the ipdvb framework is that standards-based methods may be used to build such schemes, this will in future allow re-use of standard hardware and software components.

The development of IP Layer address resolution would have several merits, particularly for IP-only services and two-way MPEG-2 transmission networks. Not only would it release a Receiver from performing MPEG-2 table processing, it would also allow much more dynamic association of PIDs to traffic. Examples of dynamic associations include: association/freeing of PIDs in response to join or prune actions taken by multicast routing protocols, or on assignment of new IP addresses using DHCP/DHCPv6.

10.3 ND Work

In many subnetworks supporting IP packets, individual nodes are assigned a layer 2 address. This address is known variously as a Medium Access Control (MAC) address or Network Point of Attachment (NPA). Such addresses may be defined for receivers in both ULE and MPE. A mechanism is required to bind IPv4/IPv6 addresses with these layer 2 addresses. Protocols are required to signal these addresses to the link receivers and transmitters, known as Address Resolution, or Neighbour Discovery (for IPv6). MPEG-2 transmission networks often utilize broadcast media (e.g., satellite and cable) where the mapping may take into account issues related to network operations and traffic engineering, and specifically mechanisms that scale (i.e. can support 100s to 10000's of downstream receivers), are insensitive to satellite propagation delay and tolerant of expected levels of packet loss. While the Neighbour Advertisement Protocol [RFC2461] could be used as a basis for such a design for IPv6 addresses, the extensive use of broadcast messages to request and transmit layer 2 addresses would prove inefficient for DVB systems using a wireless physical layer. Modifications/enhancements may therefore be required to allow ND to operate efficiently over a satellite-based DVB network.

Both ARP and ND also allow unsolicited advertisements of bindings by a sender that are broadcast/multicast to the network, without requiring the overhead of a client request. However, both ND and ARP are currently restricted to advertising a single association per message. To achieve efficient transmission and receiver processing over broadcast physical layer, a method needs to be found that advertises several associations in a single message (e.g., following the method used in MPEG-2 Tables).

Implementing such protocols above the IP layer (e.g. using multicast IP transport, as used by ND), would allow this protocol to be implemented in a portable way, not dependent on specific receiver hardware/drivers and would allow future integration of the functions within IP routers.

11 Achievements

Based on the study proposal, and the work performed, the main goals of the project have been achieved, namely:

- The formation of the IP over IETF WG and its approval by the IESG (see section 5)
- A review of the data over DVB standardisation activities
- The definition of use cases for IP over ULE/DVB
- The issuing of current ULE drafts to the IETF community for implementation within and outside the study consortium
- The implementation of ULE on 2 platforms with conformance and interoperability testing
- IP over DVB traffic analysis tools
- Analysis of ULE performance vs. MPE using spreadsheets and queuing analysis
- Analysis of ULE performance vs. MPE using captured Internet traffic
- Simulation of ULE as a general purpose RCST protocol in ns2 and OPNET
- The dissemination in the DVB, Internet and other networking communities

These achievements are detailed in the different sections of this report

12 Document History

Date	Version	Comments
15.3.2004	0.1	Document Created

13 References

[1] ETSI EN 300 421 Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for 11/12 GHz satellite services.

[2] ETSI EN 301 192: Digital Video Broadcasting (DVB): DVB specification for data broadcasting; European Standard (Telecommunications Series).

[3] ETSI EN 301 790: Digital Video Broadcasting (DVB): Interaction channel for satellite distribution systems.

[4] ETSI TS 102 292: Functional Architecture for IP interworking with BSM terminals.

[5] IETF draft-fair-ipdvb-ar; G. Fairhurst, and M.J. Montpetit, "Address Resolution for IP datagrams over MPEG2 networks." October 2003.

[6] IETF draft-fair-ipdvb-req; G. Fairhurst, "Requirements for transmission of IP datagrams over MPEG2 networks." June 2003.

[7] IETF draft-fair-ipdvb-ule; G. Fairhurst and B. Collini-Nocker, "Ultra Lightweight Encapsulation (ULE) for transmission of IP datagrams over MPEG2/DVB networks", Oct. 2003.

[8] International Journal of Computer Networks; information at http://www.sciencedirect.com/science/journal/13891286

[9] ISO/IEC 13818-6: Information Technology – Generic coding of moving pictures and associated audio information – Part 6: Extension for Digital Storage Media Command and Control (DSM-CC) – International Standard (IS).

[10] ISO/IEC 13818-1: Information Technology – Generic coding of moving pictures and associated audio information – Part 1: Systems.

[13] Heinanen, J., *Multiprotocol Encapsulation over ATM Adaptation Layer 5*. 1993, IETF.

[S11] ISO/IEC 13818-1: "Information technology - Generic coding of moving pictures and associated audio information - Systems", 2000.

[A35] ETSI EN 301 790 v1.3.1, Digital Video Broadcasting (DVB); Interaction Channel for Satellite Distribution Systems, 2003-03,

14 Abbreviations

ATSC	Advanced Television Systems Committee
B	Byte = 8 bits
BoD	Bandwidth on Demand
BOF	Birds of a Feather
BSM	Broadband Satellite Multimedia
Cell	Fixed length data unit
FEC	Forward Error Correction
DAB	Direct Audio Broadcasting
DAVIC	Digital Audio-Visual Council
DOCSIS	Data of Cable Service Interface Specification
DSMCC	Digital Storage Media Command and Control (also DSM-CC)
DVB	Digital Video Broadcasting
DVB-C	DVB for Cable
DVB-RCS	DVB Return Channel for Satellites
DVB-S	DVB for Satellite
DVB- T	DVB for Terrestrial
ESA	European Space Agency
ETSI	European Telecommunications Standards Institute
HAP	High Altitude Platform
ID	Internet Draft
IETF	Internet Engineering Task Force
IJCN	International Journal of Computer Networks
INT	Internet Notification Table (DVB)
internet	Any network based in the IP suite of protocols
Internet	The global public internet
IP	Internet Protocol
IPv4/v6	Internet Protocol version 4/6
ISP	Internet Service Provider
ITT	Invitation to Tender
ITU	International Telecommunication Union
ITU-R	ITU Radiocommunication sector
ITU-T	ITU Telecommunication Standardization Sector
LAN	Local Area Network
LISSY	LAN Interconnection Satellite System
LLC	Logical Link Control
MAC	Medium Access Control
MMT	Multicast Mapping Table (DVB-RCS)
MPE	Multiprotocol Encapsulation
MPEG	Moving Pictures Expert Group
MPEG cell	188 byte cell, with 4 byte header
MPLS	Multiprotocol Label Switching
NDA	Non Disclosure Agreement
Ns2	Network Simulator version 2
NSF	National Science Foundation (US)
OPNET	Simulation Tool
OBP	Onboard Processor
PDU	Protocol Data Unit
PEP	Performance Enhancing Proxies
PID	DVB Packet Identifier
pilc	Performance Implications of Link Characteristics
PUSI	Payload Unit Start Indicator
QoS	Quality of Service
RFC	Request For Comments
RTP	Real Time Protocol
SNAP	Subnetwork Access Protocol
SNDU	Subnetwork Data Unit
STF	Special Task Force
Tcl	Tool Command Language
TCP	Transport Control Protocol
TS	Transport Stream
VoIP	Voice over IP
WG	Working Group
WP	Work Package

Appendix 1: Final Charter of the ipdvb WG in the IETF Internet Area

IP over DVB (ipdvb)

Chair(s):
Gorry Fairhurst (□ulf@erg.abdn.ac.uk)

Responsible Area Director:
Margaret Wasserman <□ulfil□t@thingmagic.com>

Mailing Lists:
General Discussion: ip-dvb@erg.abdn.ac.uk
To subscribe: subscribe ip-dvb at majordomo@erg.abdn.ac.uk
Archive: http://www.erg.abdn.ac.uk/ip-dvb/archive/

Description of Working Group:

The WG will develop new protocols and architectures to enable better deployment of IP over MPEG-2 transport and provide easier interworking with IP networks. Specific properties of this subnetwork technology include link-layer support for unicast and multicast, large numbers of down-stream receivers, and efficiency of transmission.

These properties resemble those in some other wireless networks. The specific focus of the group is on the use of MPEG-2 transport (examples include the Digital Video Broadcast (DVB) standards: DVB-RCS; DVB-S and DVB-T and related ATSC Specifications) in next generation networks and is not concerned with the development, replacement, or retention of existing protocols on the existing generation of networks.

The WG will endeavour to reuse existing open standard technologies, giving guidance on usage in IP networks, whenever they are able to □ulfil requirements. For instance, we acknowledge the existing Multiprotocol Encapsulation (MPE) [ATSC A/90;ETSI EN 301192] and that this will continue to be deployed in the future to develop new markets. Any alternative encapsulation would need to co-exist with MPE.

Appropriate standards will be defined to support transmission of Ipv4 and Ipv6 datagrams between IP networks connected using MPEG-2 transport subnetworks. This includes options for encapsulation, dynamic unicast address resolution for Ipv4/Ipv6, and the mechanisms needed to map routed IP multicast traffic to the MPEG-2 transport subnetwork. The standards will be appropriate to both MPE
and any alternative encapsulation method developed. The developed protocols may also be applicable to other multicast enabled subnetwork technologies supporting large numbers of directly connected systems.

The current list of work items is:

Specify the requirements and architecture for supporting Ipv4/Ipv6 via MPEG-2 transmission networks. Such requirements should consider the range of platforms currently (or anticipated to be) in use. This draft will be an Informational RFC.

Define a standards-track RFC defining an efficient encapsulation method. The design will consider the need for MAC addresses, the potential need for synchronisation between streams, support for a wide range of Ipv4/Ipv6 and multicast traffic.

Provide an Informational RFC describing a framework for unicast and multicast address resolution over MPEG-2 transmission networks. The document will describe options for the address resolution process, relating these to appropriate usage scenarios and suggesting appropriate protocol mechanisms for both the existing Multi-Protocol Encapsulation (MPE) and the efficient encapsulation (2). Consideration will be paid to existing standards, and the cases for Ipv6 and Ipv4 will be described.

Define standards-track RFC(s) to specify procedures for dynamic address resolution for Ipv4/Ipv6. This will describe the protocol and syntax of the information exchanged to bind unicast and multicast flows to the MPEG-2 TS Logical Channels. This will include specific optimisations appropriate for networks reaching large numbers of down-stream systems.

Goals and Milestones:

- JAN 04 Draft of a WG Architecture ID describing usage of MPEG-2 transport for IP transmission.
- MAR 04 Draft of a WG ID on the new Encapsulation.
- JUL 04 Draft of a WG ID on the AR Framework, specifying mechanisms to perform address resolution.
- JUL 04 Submit Architecture to IESG
- OCT 04 Draft of a WG ID or the AR Protocol, defining a protocol to perform IP address resolution.
- OCT 04 Submit Encapsulation to IESG
- APR 05 Submit AR Framework to IESG
- AUG 05 Submit AR Protocol to IESG
- AUG 05 Progress the Encapsulation RFC along the IETF standards track.

Appendix 2: Potential Additions to ULE

In addition to the current syntax of the ULE packet it will be proposed to the IETF WG that ULE allows extension headers (as in IPv6). Such extension headers are part of the DOCSIS specifications. The extensions must be added in such a way that receivers can skip all (or some) extensions without having to interpret them and still process the packet. Optionally they can understand all/some of the extensions and use them to help process the packet according to the information contained in the extensions.

The format will follow the IPv6 example, e.g.:

```
+---------------------------------------------+
 |  Base Header                                |
 ---------------------------------------------
 |  Payload type = IP Packet                   |
 +---------------------------------------------+
 |    Actual Packet Payload                    |
  +---------------------------------------------+
```

In addition a sender could choose to use an extension header with one or more extensions; the packet would then become something like:

```
+---------------------------------------------+
 |  Base Header                                |
 +---------------------------------------------+
 |  Payload type = extension                   |
 +---------------------------------------------+
 | +-------------------------------------------+|
 | |     - total length of all extensions     ||
 | |    +----------------------+               ||
 | |    |  + extension type    |               ||
 | |    |  + extension length  |               ||
 | |    |  + extension value   |               ||
 | |    +----------------------+               ||
 | |    |  .... more extension|               ||
 | |    +----------------------+               ||
 |+-------------------------------------------++
 ||  Payload type = IP packet                 ||
 |+-------------------------------------------++
 |    Actual Packet Payload                    |
 +---------------------------------------------+
```

Actual use of the extension headers needs further use case analysis. The examples from the DOCSIS specs include flow management, piggy back bandwidth requests (which could be useful in OBP satellite systems using bandwidth on demand or to support RSVP, RSVP-TE and MPLS), privacy policy etc.