Efficient and Scalable Architecture for Multiview Real-Time Media Distribution for Next Generation Networks

Hugo Marques

Submitted for the Degree of
Doctor of Philosophy
from the
University of Surrey

Institute for Communication Systems (ICS)
Faculty of Engineering and Physical Sciences
University of Surrey
Guildford, Surrey GU2 7XH, UK

February 2016

Abstract

With the massive deployment of broadband access to the end-users, the continuous improvement of the hardware capabilities of end devices and better video compression techniques, acceptable conditions have been met to unleash over-the-top bandwidth demanding and time-stringent P2P applications, as multiview real-time media distribution. Such applications enable the transmission of multiple views of the same scene, providing consumers with a more immersive visual experience.

This thesis proposes an architecture to distribute multiview real-time media content using a hybrid DVB-T2, client-server and P2P paradigms, supported by an also novel QoS solution. The approach minimizes packet delay, inter-ISP traffic and traffic at the ISP core network, which are some of the main drawbacks of P2P networks, whilst still meeting stringent QoS demands. The proposed architecture uses DVB-T2 to distribute a self-contained and fully decodable base-layer video signal, assumed to be always available to the end-user, and an IP network to distribute in parallel - with increased delay - additional IP video streams. The result is a decoded video quality that adapts to individual end-user conditions and maximizes viewing experience.

To achieve its target goal this architecture: defines new services for the ISP's services network and new roles for the ISP core, edge and border routers; makes use of pure IP multicast transmission at the ISP's core network, greatly minimizing bandwidth consumption; constructs a geographically contained P2P network that uses P2P application-level multicast trees to assist the distribution of the IP video streams at the ISP access networks, greatly reducing inter-ISP traffic, and; describes a novel QoS control architecture that takes advantage of the Internet resource over-provisioning techniques to meet stringent QoS demands in a scalable manner.

The proposed architecture has been implemented in both real testbed implementation and ns-2 simulations. Results have shown a highly scalable P2P overlay construction algorithm with very fast computation of application-level multicast trees (in the order of milliseconds) and efficient reaction to peer-churn, with no perceptually annoying impairments noticed. Furthermore, huge bandwidth savings are achieved at the ISP core network, which considerably lower the management and investment costs in infrastructure. The QoS based results have also shown that the proposed approach effectively deploys a fast and scalable resource and admission control mechanism, greatly minimizing QoS related signalling events by using a per-class over-provisioning approach and thus preventing per-flow QoS reservation signalling messages. Moreover, the QoS control architecture is aware of network link resources in real-time and supports for service differentiation and network convergence by guaranteeing that each admitted traffic flow receives the contracted QoS.

Finally, the proposed *Scalable Architecture for Multiview Real-Time Media Distribution for Next Generation Networks*, as a component for a large project demonstrator, has been evaluated by an independent panel of experts following ITU recommendations, obtaining an excellent evaluation as computed by Mean Opinion Score.

Key words: 3D, application-level multicast, content delivery networks (CDN), DVB-T2, IP multicast, multiview, peer-to-peer (P2P), P2P-CDN, Quality of Service (QoS), QoS overprovisioning, video streaming.

Email: hugo.marques@av.it.pt

WWW: https://www.surrey.ac.uk/faculty-engineering-physical-sciences

Acknowledgments

This thesis would not be possible without the unconditional support from my supervisors, hence my first words are for Prof. Rahim Tafazolli, Dr. Seiamak Vahid and Dr. Jonathan Rodriguez with whom I shared my doubts and obtained valuable guidance.

Secondly, to my family for their patience during this period of partial absence. Therefore my thanks go to Célia, Mariana, Miguel and my 1st tier family.

A very special thanks goes to Paulo Marques for always seeing the practical side of things and for teaching me the importance of simplicity.

Special thanks go: to Hélio Silva for always addressing problems as interesting challenges and his support in the implementation of the P2P components; to Evariste Logota for his guidance in the shaping of the QoS approach proposed in this thesis and support in the ns-2 implementations; to Ricardo Bassoli, Vahid Nazari Talooki and Victor sucasas for being my out-of-band PhD colleagues, with whom I learned much and published many scientific publications; to Rogério Dionísio for kindly accepting being overloaded with work during the final months of the writing of this thesis.

I would also like to thank all the other researchers who I have worked with, especially in the FP7 ROMEO project and the 4TELL Research Group, and to Cláudia Barbosa for all the management support, hotel bookings, travels, etc.

Finally, I would like to thank my hosting institution, Instituto Politécnico de Castelo Branco, and all my colleagues for supporting me towards the PhD conclusion.

Contents

List of Figures

Glossary of Terms (optional)

A/V	Audio/Video
AAA	Authentication, Authorization And Accounting
ACOR	Advanced Class-based resource Over-Reservation
ACS	Admission and Control Subsystem
AVC	Advanced Video Coding
BGP	Border Gateway Protocol
CAGR	Compound Annual Growth Rate
CDN	Content Delivery Network
CIB	Control Information Base
CoS	Class of Service
CR	Core Router
CRRM	Cooperative Radio Resource Management
CS	Chunk Selection
DiffServ	Differentiated Services
DNS	Domain Naming System
DVB-T2	Digital Video Broadcast – Second Generation Terrestrial
ER	Edge Router
ESM	End System Multicast
FIB	Forwarding Information Base
FTP	File Transfer Protocol
HEVC	High Efficiency Video Coding
IP	Internet Protocol
IRACS	Internet Resource and Admission Control Subsystem
ISP	Internet Service Provider
ITU	International Telecommunication Union Standardisation Sector

LTE	Long Term Evolution
MARA	Multi-user Aggregated Resource Allocation
Mbps	Mega bit per second
MIB	Management Information Base
MIH	Media Independent Handover
MOS	Mean Opinion Square
MPEG	Moving Picture Experts Group
MRIB	multicast Routing Information Base
MTM	Multicast Tree Manager
MTU	Maximum Transmission Unit
MVC	Multiview Video Coding
MVD	Multiview Video plus Depth
NAL	Network Abstraction Layer
NALU	Network Abstraction Layer unit
NAT	Network Address Translation
NCDP	Network Control Decision Point
NFV	Network Functions Vitrtualization
NMS	Network Monitoring Subsystem
NSIS	Next Steps in Signalling
NUT	Network Abstraction Layer unit type
OSFP	Open Shortest Path First
P2P	Peer-to-peer
PA-CDN	Peer-Assisted Content Delivery Network
PCR	Program Clock Reference
QoE	Quality of Experience
QoS	Quality of Service
RACF	Resource Admission Control Function

RAM	Resource and Admission Manager
RAO	IP Router Alert Option
RC	Resource Controller
RIB	Routing Information Base
RRO	Record Route Object
RRS	Resource Reservation Subsystem
RSVP	Resource Reservation Protocol
RTP	Real-Time Protocol
SDN	Software Defined Networks
SEI	Supplemental Enhancement Information
SLA	Service Level Agreement
SOMEN	Self-Organizing Multiple Edge Nodes
SVC	Scalable Video Coding
TB	Topology Builder
TC	Topology Controller
TISPAN	Telecommunications and Internet converged Services and Protocols for Advanced Networking
UDP	User Datagram Protocol
UE	User Equipment
VCEG	Video Coding Experts Group
VCL	Video Coding Layer
VoIP	Voice over IP

Chapter 1

1 Introduction

1.1 Motivation

In recent years, speed for Internet access has experienced enormous upgrades both in wired and wireless links. This is especially true in the majority of the cities of modern countries, where fibre-to-the-home and Long Term Evolution (LTE) access is a reality. In addition, computers and mobile devices have also become faster and lighter, which in fact empowered them for a new type of use – the visualization of social networking media, also known as user-generated content. According to Alexa Internet [1], Facebook and YouTube are the two top contributors for this kind of content distribution. In fact, according to[2] Internet video from sites such as YouTube (short-form†), Hulu (long-form†), Netflix (video-to-TV), online video purchases and rentals, webcam viewing, and web-based video monitoring (excluding P2P video file downloads) are expected to have a compound annual growth rate (CAGR) of 34% till 2016. Video mobile data is predicted to have a CAGR of 75% in the period 2012-2017 according to [3]. Netflix, by itself attracts more than 23 million subscribers in the North America, streaming high-definition quality video with an average bitrate reaching almost 4Mbps. According to [4] Netflix is the single largest source of Internet traffic in the US, consuming 29.7% of peak downstream traffic.

Thus it is widely accepted that consumers will expect more features from their viewing experience. First, they want on-demand services so that they can watch the contents at the moment of desire. Second, they want to watch content anytime, anywhere, and regardless of the device type. It could be wide-screen display in a living room, a navigation screen in a car, or a handheld device such as smart phone or tablet. Third, high definition content has already gained popularity in most of European countries, and even ultra-high definition quality is expected to attract more people considering that the resolution of some tablets and laptops is already far better than the high definition. The trends of on-demand, mobile, and ultra-high definition quality impose formidable challenges for the delivery network of the future.

† **Short form**: User-generated video and other video clips generally less than 7 minutes in length; **Long form**: Video content generally greater than 7 minutes in length

Such challenges include the capability of satisfying the stated and implied needs of different existing network applications or services (therefore contributing to their overall acceptability as perceived subjectively by the end-users), the provision of broadband, seamless and heterogeneous connectivity and the availability of mechanisms capable of providing an effective use of networking resources, Such delivery network of the future should be perceived as a unified communication channel, where all the legacy access and wide area coverage networks are able to interplay in harmony, providing the best available connection to end-users. It also must be smart in nature, allowing resources to be adaptively allocated on demand to suit the application at need.

In contrast, today´s communication ecosystem constitutes a plethora of communication networks independently engineered for the application at hand with some attempts at integration towards this utopian highway. On one hand, 4G mobile networks aims to provide a unified mobile networking island allowing seamless connectivity between 3GPP mobile standards, and non-compliant systems through mobility anchors; so called gateways to the IEEE networking systems. Moreover, it attempts to provide a networking bridge towards the IP world to provide some medium rate data services with limited coverage. On the other hand, we have the IP world for fixed users that provides very high speed connectivity, but without service guarantees since the current Internet is not optimized for real time traffic.

So what does the end-user and ISP provider experience today? The end-user can today receive IP services on their mobile handset, but coverage is potentially limited and the viewing experience could be subject to glitches with delay; moreover the expectation of receiving 3D or multi-viewing experience would be thrashed. The effect would be even more pronounced for the ISPs that have invested significantly in networking infrastructure. Without smart mechanisms in place, that can adaptively allocate resources to users demand, wastage of resources can be foreseen in terms of surplus signalling and ineffective use of bandwidth. Furthermore, the ability to compete in the market towards providing new services with higher definition would be severely diminished.

Taking a step towards the future, concepts such as Software-Defined Networking (SDN), Virtualization and P2P networking are being actively investigated towards providing more effective use of networking resources. SDN allows centralised control of networking resources at the touch of a button, whilst virtualization allows networking resources to be shared between different service providers and services pretty much in a similar fashion to the cloud paradigm. Finally, P2P networking has been seen as a way towards providing application connectivity and distributing large amounts of data over the IP network in a scalable manner. However, existing approaches are either limited in the way they can handle dynamic networking characteristics, heterogeneous devices with varying capability, and real-time traffic.

In this thesis, a step towards this network of the future is taken by investigating a new architecture that can provide efficient and scalable multiview media distribution for real-time media. In this context, two main research challenges are addressed related to P2P networking and QoS resource provisioning on the network. The first research challenge aims to rethink P2P networking, as a complementary technology to DVB-T2 in the distribution of additional video streams to end-users that will improve the quality of the decoded video. Considering the time-stringent characteristics of real-time media, the P2P network paradigm needs rethinking in order to ensure minimum delay while providing fast response to its inherent dynamic characteristics (e.g., peer churn). P2P also needs to be seen as a helper solution in freeing resources at the ISP core network. With this in mind this thesis proposes a hybrid client-server and P2P approach in an also novel synergy between pure IP multicast [5] and application-level multicast. In this setup, by using IP multicast at the ISP core network, containing the P2P network to the ISP access-network level and using the application level multicast trees as distribution paths for the P2P traffic, the approach greatly reduces the related traffic at the ISP core network, practically eliminates related P2P inter-ISP traffic and greatly improves response time to the peer-churn effect. The second research challenge tackles how to achieve "scalable QoS provisioning over the IP network" using QoS over-provisioning with per class reservations, where current implementations are limited in terms of providing real-time knowledge of link resource statistics and handling large networking loads, which have led to excess, per IP flow, protocol signalling.

1.2 Structure of Thesis

The thesis is divided in six chapters, starting with this introductory chapter. The following chapters provide background information, thoroughly describe the proposed architecture, present results, concluding remarks and future directions. Furthermore each chapter includes an abstract and chapter conclusions to facilitate reading. In particular:

Chapter 2 provides background information on the technologies that are relevant for this thesis. It includes summary descriptions for scalable video coding (SVC), stereoscopic video, multiview video coding (MVC), MVC plus depth (MVD) and procedures for content representation and coding. Furthermore, it also describes relevant video streaming approaches, including client-server and P2P approaches, content delivery networks (CDN), peer-assisted CDNs (PA-CDNs) and existing QoS solutions.

Chapter 3 is the core chapter of the thesis and thoroughly describes in detail the architecture for multiview real-time media distribution, including its components, modules and network operations.

Chapters 4 and 5 present component validation and integrated system validation. Component validation is obtained through objective measurements whilst the integrated system validation includes both objective and subjective test measurements. The evaluation of results in Chapter 5 is independently performed by a set of external subjects.

Finally, Chapter 6 presents overall conclusions for the implemented architecture and its results, and foresees future directions of research.

1.3 Novel Work Undertaken

This thesis presents an efficient and scalable architecture for multiview real-time media distribution over a next generation network. The proposed architecture merges the concept of terrestrial digital video broadcast (DVB-T2) with novel hybrid client-server and P2P distribution systems and a novel QoS overprovisioning solution.

The goal is to facilitate over-the-top multiview real-time media distribution over current Internet architecture including the provisioning of guidelines for inter-ISP operation, therefore minimal architecture updates are foreseen at the ISP's core network.

The scientific outcomes from this work have led to the following publications:

Book Chapters

- H. Marques et al, "Next Generation Communication Systems For PPDR – The SALUS Perspective", in book: Wireless Public Safety Networks 1, Elsevier-ISTE, December 2015. [ISBN: 978-1-7854-8022-5].
- H. Silva, H. Marques, J. Rodriguez, "3D media distribution over the Internet with hybrid client-server and P2P approach", in Book: Wireless Internet, Springer, November 2014. [ISBN: 978-3-319-18801-0].

  [Note: The first two author's appearance order in this paper is alphabetical, with equal main contributions].
- E. Logota, F. Saghezchi, H. Marques, J. Rodriguez, "Cooperative Strategies for Energy Saving and End-to-End QoS Control", in book: Novel 3D Media, edited by Tasos Dagiuklas and Ahmet Kondoz, Springer, October 2014 [ISBN: 978-1-4939-2025-9]
- G. Mantas, N. Komninos, J. Rodriguez, Evariste Logota, H. Marques, "Security for 5G Communications", in book: Fundamentals of 5G Mobile Networks, Wiley, October 2014 [ISBN: 978-1-118-86752-5].
- E. Logota, H. Marques, J. Rodriguez, F. Pascual, M. Sanz, "Dynamic QoS Support for P2P Communications", in book: 3D Future Internet Media, edited by Tasos Dagiuklas and Ahmet Kondoz, Springer, April 2013 [ISBN 978-1-4614-8372-4].

- A. Lykourgiotis, R. Bassoli, H. Marques and J. Rodriguez, “IP-based Mobility Scheme Supporting 3D Video Streaming Services”, in book: 3D Future Internet Media, edited by Tasos Dagiuklas and Ahmet Kondoz, Springer, April 2013 [ISBN 978-1-4614-8372-4].

Journals

- H. Marques, H. Silva, E. Logota, J. Rodriguez, S. Vahid, R. Tafazolli, “Multiview Real-Time Media Distribution for Next Generation Networks”, the International Journal of Computer and Telecommunications Networking (Computer Networks), Elsevier, *[Accepted, to be published in 2017]*.

Peer Review Conferences

- V. Monteiro, K. Huq, S. Mumtaz, H. Marques, C. Thomos, T. Dagiuklas, J. Rodriguez, "3D Video Streaming Transmission over OFDMA-based Systems", 21st IEEE International Conference on Image Processing (ICIP 2014), Paris, France, 27-30 Oct 2014.
- E. Logota, H. Marques, J. Rodriguez, “A Cross-layer Resource Over-Provisioning Architecture for P2P Networks”, 18th International Conference on Digital Signal Processing (DSP 2013), Special Session on 3D Immersive & Interactive Multimedia Over the Future Internet, Santorini, Greece, 1-3 July 2013.
- V. De Silva, E. Ekmekcioglu, S. Dogan, A. Kondoz, A. Akman, F. Pascual, I. Politis, H. Marques, X. Shi, Mulsys, “Immersive 3D Media Delivery for Collaborating Users over Multiple Networks” 18th International Conference on Digital Signal Processing (DSP 2013), Special Session on 3D Immersive & Interactive Multimedia Over the Future Internet, Santorini, Greece, 1-3 July 2013.
- S. Dogan, E. Ekmekcioglu, A.M. Kondoz, A. Akman, F. Pascual Blanco, I. Politis, T. Dagiuklas, H. Marques, “Real-time immersive multimedia experience for collaborating users over hybrid broadcast networks”, Future Network & Mobile Summit (FUNEMS 2013), Lisbon, Portugal, July 2013.
- H. Silva, H. Marques, J. Rodriguez, C. Verikoukies, “Multiple Multicast Trees for 3D Real-Time Content Distribution over P2P Networks”, 1st ACM Workshop on: “High Performance Mobile Opportunistic Systems” HP-MOSys, Cyprus, Oct.2012.
  [Note: The first two author’s appearance order in this paper is alphabetical, with equal main contributions].
- G. Matthew, H. Marques, J. Rodriguez, "Cross-Layer Approach in P2PSIP MANETs", Proc. of the IEEE Consumer Communications and Networking Conference (CCNC 2011), Las Vegas, USA, January 2011

Chapter 2

2 Background Information

Chapter Outline: this chapter provides baseline information on the technologies that are relevant for the presented thesis. Video distribution over the Internet is by itself far from being novel. The evolution of microprocessors and advances in video compression algorithms such as the H.264/MPEG-4 Advanced Video Coding (AVC) standard or, more recently, the HEVC/H.265 standard, have largely contributed for the increase consumption of video over the Internet. Their capability to achieve a considerable bandwidth reduction whilst still providing high video quality, together with faster processors, increased Internet availability and access speeds, were key factors for the video boom that we have seen in the past years. In fact, video will still be the main responsible for data traffic in the Internet for years to come. End-user expectation as also increased, besides access everywhere users also expect better video quality and new experiences (e.g. 360º virtual reality videos, already available in YouTube). Distribution of multiview video to end-users in a large scale, was something been forecast many times in the past, mainly because it enables the visualization of 3D content. The selling of 3D technology enabled devices and content, besides the noted interest in the past 3-4 years, still is below expectations and with the exception of cinema, there are few adepts of the technology in personal or home environments. This thesis is presented in the assumption the multiview technology will be more user friendly in the upcoming years, providing a comfortable user experience in smaller devices such as personal computers, laptops and mobile phones (and also the home TV set). When such milestone is achieved, multiview can be expected to be implemented in full force, therefore the main issue will be on how to distribute it timely and in a resilient, pervasive and bandwidth efficient manner. Towards this goal: Chapters 2.1 to 2.6 provide background information on video coding techniques; Chapter 2.7 describes server based video streaming technologies; Chapter 2.8 explains the packetisation structure used for the system concept and validation; Chapter 2.9 identifies different Peer-to-Peer (P2P) video streaming approaches; Chapter 2.10 provides a description of Content Delivery Networks (CDNs) whilst Chapter 2.11 focus on Peer-Assisted CDNs (PA-CDNs); Chapter 2.11, provides a summary on current existing QoS techniques needed to ensure proper treatment of the video packets is given at network level and, finally; Chapter 2.12 presents the overall remarks on this chapter.

2.1 Basics of video compression and the H.264/MPEG-4 AVC

Overall, video content can be seen as a sequence of pictures (called frames) played at a specific speed, commonly known as frames per second (fps). Frames are composed by many pixels arranged on a specific horizontal and vertical resolution (e.g., 1.280x720 for High Definition (HD), 1.920x1.080 for Full HD (FHD), 2.048x1080 for 2K or 3.840x2.160 for 4K), where each pixel occupies 3 bytes (for a 24 bit colour coded image), one for each of the following colours: red, green and blue (also known as RGB). If no compression is used, the size of each frame is the total numbers of pixels multiplied by 3 Bytes and the raw video bitrate would be that result multiplied to the fps used in the encoding. Table 2-1 depicts some values for the abovementioned resolutions.

Table 2-1: Transmission rates for uncompressed video

Video resolution	Horizontal Pixels	Vertical Pixels	Frame Pixels (Total)	Bytes per pixel	Total Frame size (Bytes)	Frames per second	Required raw transmission rate
HD	1.280	720	921.600	3	2.764.800	60	~166MB/s
FHD	1.920	1.080	2.073.600	3	6.220.800	60	~375MB/s
2K	2.048	1.080	2.211.840	3	6.635.520	60	~400MB/s
4K	3.840	2.160	8.294.400	3	24.883.200	60	~1.5GB/s

Such values would be very difficult to achieve in today's Internet. The solution to this problem is to use spatial and temporal compression. Spatial compression is achieved through: (i) an entropy encoder to remove redundant information in each frame (e.g., pixels that are spatially close to each other typically have similar values); (ii) processing the image in the frequency domain and applying a mask to the high frequencies, which are known to provide small and sharp image details, and; (iii) using Chroma Subsampling to reduce smooth colour variations, which are imperceptible to the human eye, within the sampled image. Temporal compression, on the other hand, performs compression through predictive coding by analysing a group of frames across time. It is common for consecutive frames in a video shot to be almost identical. Redundant information between previous and next frames is removed and the difference stored in the form of a (shift) vector. This process is known as motion compensation and the 'amount' of shift is given by a motion vector. Hence the concept of I-frame (Intra frame), P-frames (predicted) and B-frames (bi-directionally predicted) is introduced. I-frames correspond to a full resolution of a reference image in a scene and do not depend on other video frames to be decoded. P-frames contain predictive information of a scene based on the information provided in previous frames and as such depend on these to be decoded. B-frames also contain predictive information of a scene, but are based on the information provided by both past and future video frames.

H.264/MPEG-4 Advanced Video Coding (AVC) standard* [6] makes use of all these compression techniques plus a number of new features that allow it to compress video much more efficiently. It is currently one of the most used video coding standards used to distribute video. For reference, the H.264/AVC has been officially released in 2003 by the Joint Video Team of the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG). Since then, many revisions, corrigenda, and amendments have been released. Of relevance for this Thesis are the ones depicted in Table 2-2.

Table 2-2: H.264/AVC releases of relevance for this Thesis

Version	Date	Release description
Version 8	November 22, 2007	Major addition to H.264/AVC containing the amendment for **Scalable Video Coding** (SVC)
Version 11	March 16, 2009	Major addition to H.264/AVC containing the amendment for **Multiview Video Coding** (MVC) extension
Version 18	April 13, 2013	Amendment to specify the **coding of depth map data for 3D stereoscopic video**
Version 21	February 13, 2014	Amendment to specify the **Enhanced Multiview Depth High profile**

When compared to previous standards, H.264/AVC gains in compression efficiency can be up to 50%, whilst the decoder complexity is about four times that of MPEG-2 [7].

Some of the major features that make H.264/AVC a successful video coding standard are:

- *Intra Prediction*: as with previous video coding standards, H.264/AVC divides the image into small square blocks called macroblocks. Typically these are 16x16 pixel Y blocks, related with luminance information and 4x4 pixel Cr and Cb blocks, related with colour information. The reason why the colour blocks are smaller is because the human eye is less sensitive to colour than to luminance. *Intra prediction* means that the samples of a macroblock are predicted by using only information of already transmitted macroblocks of the same image (frame). For the prediction purpose, nine different prediction modes are supported (e.g., vertical prediction, horizontal prediction, DC-prediction and plane-prediction), where blocks are predicted from previously decoded blocks to the north, north-east, and west (in a frame structure). For further information on intra prediction modes, please refer to [8].
- *Motion Compensated Prediction*: contrary to previous standards where only blocks of the size 16x16 and 8x8 were supported, H.264/AVC allows each macroblock to be divided into smaller partitions and even sub-macroblock partitions, as depicted in Figure 2-1 below.

* In this thesis the short acronym H.264/AVC will be used

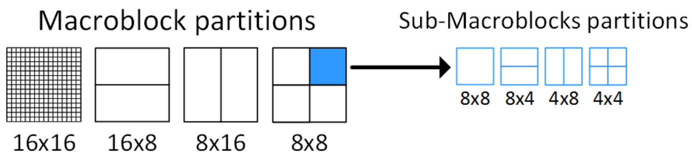

Figure 2-1: H.264/AVC macroblock partitions

Such fine-graining enables more efficient motion-compensated prediction, since with smaller blocks better matches can be found and used with motion vectors (which provide the corresponding position of the block in a previous frame). It is important to note the H.264/AVC allows multiple previous reference images to be stored in memory which further increases the coding efficiency of the *motion compensated prediction*.

- *Transform Coding*: when predicting the current image, motion compensation is used from already reconstructed images, as seen before. In order to reduce the spatial redundancy of the prediction error, transform coding is applied to each block. Contrary to the previous video coding standards, where a Discrete Cosine Transform (DCT) was used, H.264/AVC uses (different) Integer Transforms. Compared to a DCT, the Integer Transforms have only integer numbers ranging from −2 to 2 in the transform matrix, which allows computing the transform and the inverse transform in 16-bit arithmetic using only low complex shift, add, and subtract operations [7]. Further information on the H.264/AVC transform coding can be found on [9].
- *Entropy Coding Schemes*: H.264/AVC specifies two new entropy coding techniques which represent major improvements in terms of coding efficiency, these are: Context Adaptive Variable Length Coding (CAVLC) - an adaptive variant of Huffman coding - and Context Adaptive Binary Arithmetic Coding (CABAC) - based on arithmetic coding techniques. The latter achieves compression ratios that are 10% to 15% higher than CAVLC [10]. For further information on CABAC please refer to [10].
- *Adaptive Deblocking Filter*: when using macroblocks to process a frame, artefacts can appear at the block edges, a common fate in previous standards, such as the H.263, which did not mandatory-enforced the filtering of the block edges. To ensure the quality intended by the video producer reaches the consumer, H.264/AVC mandates the filtering process to be carried out in the coding loop, where three levels of the *Adaptive Deblocking Filter* are foreseen: *slice level*, *block edge level* and *sample level*. According to [11] loop filtering typically improves both objective and subjective quality of video streams with significant reduction in decoder complexity when compared to post filtering.
- *Profiles and Levels*: H.264/AVC was designed to suite the requirements of a large range of video application domains such as video conferencing, mobile video, consumer and high definition broadcast. This includes different bit rates, resolutions and video quality.

Hence profiles and levels were specified to ensure conformance of encoders and decoders with the standard (profiles define sets of coding tools and algorithms that can be used, while levels place constraints on the bitstream parameters). Profiles also allow a decoder to quickly recognize the requirements to decode a specific stream. In its initial release, only two profiles have been initially defined, currently there are more, some were already depicted in Table 2-2. For reference, the following are the most well-known H.264/AVC profiles:

- *Baseline Profile* (BP): the simplest profile, targeting low-cost applications - mainly used for video conferencing and mobile video;
- *Main Profile* (MP): targeted for consumer broadcast (standard-definition digital TV broadcasts that use the MPEG-4 format) and storage applications, but has been overtaken by the High Profile (see below);
- *Extended Profile* (XP): intended for streaming video, achieves high compression with improved robustness to data loss and has special characteristics for server stream switching;
- *High Profile* (HiP): intended for high definition broadcast (HDTV) and disc storage (e.g., Blu-ray)

For a complete and updated list of the H.264/AVC Profiles and levels, please refer to [6].

2.2 Scalable Video Coding (SVC)

Scalable Video Coding (SVC) provides the ability to recover acceptable video quality by decoding only parts of a video bitstream. Technically it enables decoding partial bitstreams in order to provide lower temporal (frame rate) or spatial (frame size) resolutions, or reduced fidelity (signal-to-noise ratio (SNR)) video, while retaining a high reconstruction quality (relative to the rate of the partial bitstreams). Hence, scalability refers to the capability of removing parts of the video bitstream in order to adapt it to the heterogeneous network conditions, terminal capabilities and end-user preferences. The different types of scalability can also be combined so that different spatio–temporal resolutions and bit rates can be supported within a single scalable bit stream.

One of the most important features of SVC is, the source content has to be encoded only once (for the highest required resolution and bit rate), resulting in a scalable bit stream from which representations with lower resolution and/or quality can be obtained by discarding selected data. In comparison to single-layer coding, bit rate increases of 10% to 50%, for the same fidelity, might be tolerable [12].

Temporal scalability can be achieved in different ways, but the main principle is the same, there is a temporal base layer and multiple temporal enhancement layers, where each layer depends on the previous layers. Figure 2-2 illustrates 3 different approaches for achieving temporal scalability, where the set of pictures between two successive pictures of the temporal base layer, together with the succeeding base layer picture, is referred to as a *group of pictures* (GOP). T_k represents a specific temporal layer.

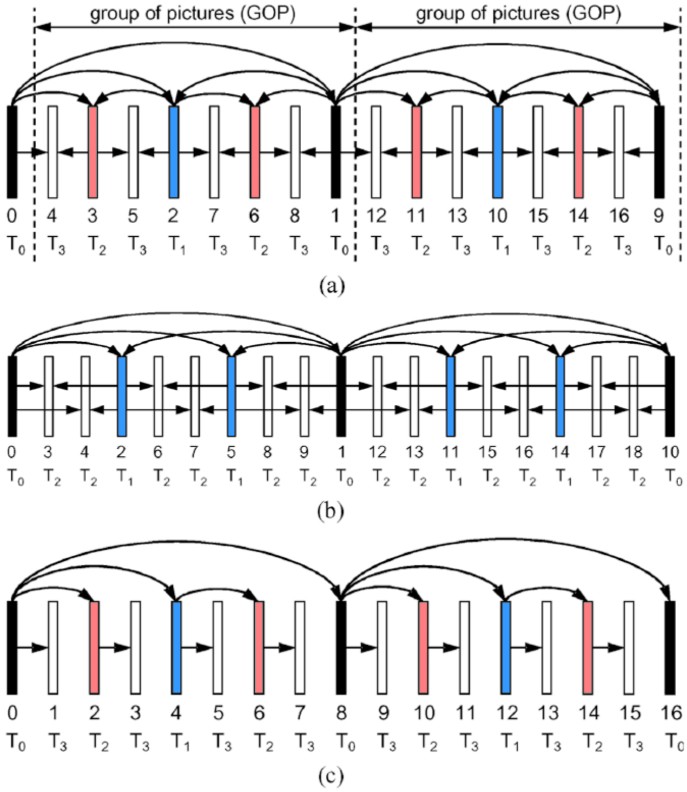

Figure 2-2: Temporal scalability with hierarchical prediction structures, adapted from [12]

Figure 2-2 (a) depicts temporal scalability with hierarchical B-pictures. By using the depicted coding scheme (a dyadic prediction structure) it is easy to see that any temporal layer T_K can be obtained/decoded independently of other temporal layers greater than *k* (e.g., T_1 can be decoded independently of having received the temporal layers T_2 or T_3 - on the other hand, T_2 and T_3 depend on T_1 which depends from T_0, note that T_3 does not depend on T_2). In this scheme we have three independently decoded bitstreams with 1/8th (T_0 only), 1/4th (T_0+T_1) and 1/2 (T_0+ T_1+T_2) of the full frame rate.

Figure 2-2 (b) depicts temporal scalability with a nondyadic hierarchical prediction structure. In this case there are only two independently decodable bitstreams with 1/9th (T_0 only) and 1/3rd (T_0+T_1) of the full frame rate. To improve coding efficiency, modification of the prediction structure for the temporal base layer may be modified across time.

Figure 2-2 (c) provides an example of a hierarchical prediction structure that does not use future frames for motion compensated prediction, hence the decoding is faster (with less delay). Taking as an example T_1 in Figure 2-2 (a), it can be seen that this particular temporal layer needs to wait for 9 frames to be transmitted in order to be decoded.

In what respects spatial scalability, different spatial layers are used, each with a different resolution (the most common configurations adopts the 2:1 relation between neighbour layers, however different aspect ratios can be used by H.264/SVC). The principle is similar to that of temporal scalability, a higher spatial layer (e.g., D_1) may be obtained (reconstructed/upsampled) from a lower one (e.g., D_0), using a technique called *Inter-Layer Prediction*. The word ‘may’ was used because in some cases, higher spatial layers can also be obtained by temporal prediction within the same spatial layer - this concept is shown in Figure 2-3. Generally the temporal prediction provides a better approximation of the original signal than the upsampled lower layer reconstruction.

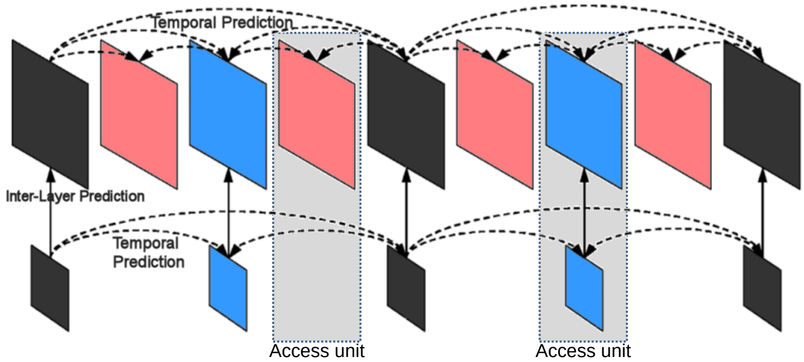

Figure 2-3: Spatial scalability with inter-layer and temporal prediction, adapted from [12]

As depicted in Figure 2-3, the lower spatial layer may have a different frame rate than the upper spatial layer, in other words, lower layer pictures do not need to be present in all *access units*. This means inter-layer prediction can only take place inside a given access unit using a layer with a spatial layer identifier inferior to the spatial layer identifier of the layer to be predicted.

For further information on SVC, please refer to the H.264/AVC standard [6] or [12].

2.3 The 3D Video Format Concept

Stereoscopic video mimics human binocular view and consists of coding scenes captured by two slightly separated cameras. It basically consists on a time sequence transmission of two sets of images (left view and right view), forming a two-dimensional arrangement. Figure 2-4 illustrates how video data from a multiple camera scenario may be arranged after being captured.

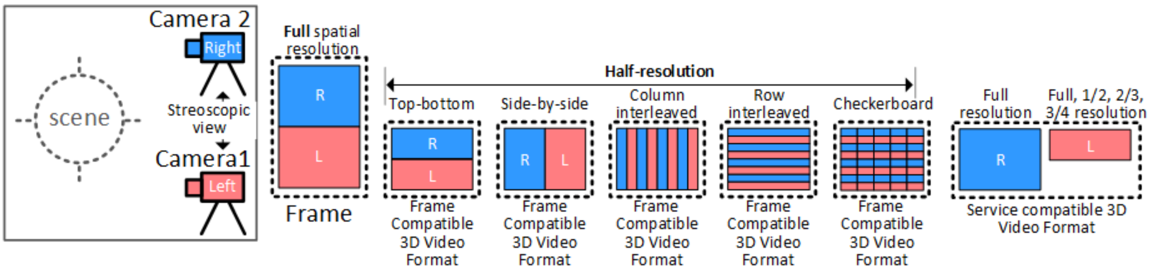

Figure 2-4: Representation formats for 3D video

In full spatial resolution the video information from each camera is captured at the full spatial resolution with a temporally synchronized camera array - this produces very high video quality with a very high bitrate. In frame compatible 3D video formats the two stereo views are multiplexed into a single coded frame, where each coded view has half the resolution of the full coded frame and the frame rate is equivalent to that of a single view – this format is compatible with existing encoding, decoding and delivery infrastructure. In service-compatible 3D video formats, left and right views are transmitted as independent video elementary streams that can be consumed simultaneously by both legacy 2D and 3D devices with high 3D video quality.

The coding of stereoscopic images has been researched for many years, mainly in exploiting the correlation between the two views. Since both cameras capture the same scene, the basic idea is to exploit the interview redundancy for compression, which can be of two types: interview similarity across adjacent camera views, and temporal similarity between temporally successive images of each video. A coder that can take advantage of these redundancies is usually called a multiview coder.

2.4 Multiview Video Coding (MVC)

Multiview Video Coding (MVC) is fundamental to reach acceptable compression efficiency for the representation of multiple views of a video scene, such as in multiple synchronized video cameras (i.e., stereo-paired video for 3D viewing). The compression efficiency is obtained through inter-view, temporal and spatial prediction, as illustrated in Figure 2-5:

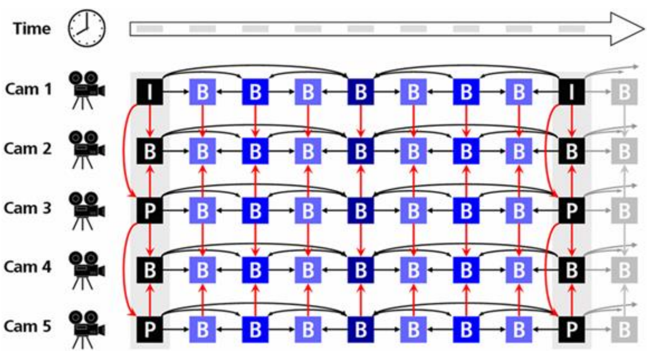

Figure 2-5: Frame prediction in H.262/MPEG-2 video multiview profile, adapted from [13]

Since the cameras capture the same scene from nearby viewpoints, a considerable inter-view redundancy is present. While intra-coded pictures (I-frames) provide full information of a view in a specific time, the bi-directionally predicted frames (B-frames) are computed from temporally neighbouring images and adjacent views. This brings a significant compression gain and lower bitrate. In recognition of this high quality encoding capability and support for backward compatibility, the Stereo High profile of the MVC extension was selected by the Blu-Ray Disc Association as the coding format for 3D video with high-definition resolution.

MVC, however, assumes fixed view positions and does not support viewpoint adjustment or additional viewpoint generation in the receiver-side. In MVC is normal to transmit the complete set of captured views throughout the media session, being therefore broadcast delivery of content its natural application.

The use of MVC also mandates the compressed multiview stream to include a base view bitstream coded independently from all other views - this ensures compatibility with decoders for single-view profile of the standard. This is especially important for the television broadcast use case, where the base view could be extracted and decoded by legacy receivers, while newer 3D receivers could decode the additional 3D bitstream including non-base views.

To address this need for flexibility and customizability, the H.264/AVC design, covers a video coding layer (VCL) designed to efficiently represent the video content, and a Network Abstraction Layer (NAL), see Figure 2-6.

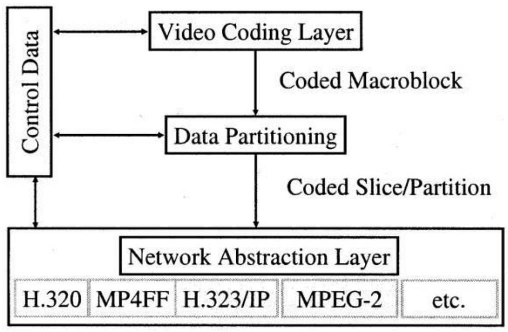

Figure 2-6: Structure of H.264/AVC video encoder [28]

The NAL formats the VCL representation of the video and provides header information in a manner appropriate for conveyance by a variety of transport layers or storage media. Different types of NAL units exist (also known as NUT), some are for coded video pictures, while others for non-picture data such as parameter sets and supplemental enhancement information (SEI) messages. MVC makes use of the NAL unit type structure to provide backward compatibility for multiview video, where the video data associated with a base view is encapsulated in NUT defined for the 2D video, while the video data associated with the additional views are encapsulated in a different NUT, that is used for both scalable video coding (SVC) [6] and multiview video†.

A scalable video sequence may hence be achieved, composed of a so-called base-layer and of one (or more) enhancement-layer(s): compared to a single-layer sequence, the base-layer is self-contained and fully decodable to a signal of lower quality and/or lower resolution in terms of pixel or time. Enhancement layers, on the contrary, cannot be decoded if the base layer is lost or damaged and can only be used to improve the overall quality. Therefore, scalable coders allow a stream to be sent over channels with different bandwidth constraints or to devices having different capabilities. However, the bit rate that is required for a given level of video quality still increases approximately linearly with the number of coded views [14] hence using MVC to transmit a large number of video views for multiview displays is not feasible.

2.5 Multiview Video Coding plus Depth (MVD) and the new 3D-HEVC

In multiview plus depth (MVD) coding a layer can represent texture, depth, or other auxiliary information of a scene related to a specific camera perspective, thus multiview plus depth coding provides redundancy reduction among different views at the same time instance, for which the content is usually rather similar and only varies by a slightly different viewing position [18].

MVD enables to effectively adapt the video content on different displays (e.g., multistereoscopic displays, capable of displaying multiple views at the same time) and viewing conditions, such as viewing distance, as well as for meeting individual preferences. This is ensured through disparity adjustment between views [15], the concept is illustrated in Figure 2-7.

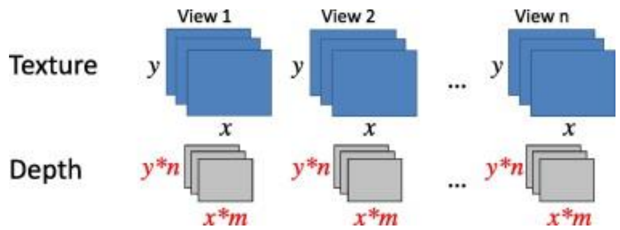

Figure 2-7: Example of resolutions of texture and depth view components [16]

In contrast to video or texture data, depth maps contain sharp edges at boundaries between objects with different scene depth. They are also characterized by having large areas of similar or slowly varying sample values. Depth data are a monochromatic signal provided by grayscale images, where each pixel indicates the distance of the corresponding 3D-point from the camera (also called depth map). By relying on the available texture (conventional 2D colour sequences, providing colour information) and depth data, using projective geometry [27], it is possible to

† A flag is specified to distinguish whether the NAL unit is associated with an SVC or MVC bitstream

synthetize a novel view of the scene from a viewpoint which differs from those captured by the cameras.

The multiview video plus depth (MVD) representation is one of the most prominent techniques to enable flexible view synthesis in the rendering end. In MVD, each texture image sequence is accompanied by the corresponding per-pixel depth map sequence.

More recently, as a result of significant advances in video coding, the High-Efficiency Video Coding (HEVC) standard was released as ITU-T Recommendation H.265 and ISO/IEC 23008-2 (MPEG-H Part 2). Its video coding layer design is based on conventional block-based motion compensated hybrid video coding concepts, but with some important differences relative to the prior standards: increased video resolution and increased use of parallel processing architectures. When correctly configured, the features of the new design can encode videos with 50% less bit rate (on average) for equal perceptual video quality [17].

Based on the new features of HEVC, a 3D video coding extension was developed to enhance the depth performance of 3D video formats such as the MVD. The 3D extension of HEVC (3D-HEVC) is included in the third version of HEVC (February 2015) and provides increased coding efficiency by joint coding of texture and depth for advanced 3D displays.

3D-HEVC makes use of three different tools for the coding of dependent views, these are: *Disparity-Compensated Prediction*, *Inter-View Motion Parameter Prediction* and *Inter-View Residual Prediction*. The *disparity-compensated prediction* (DCP) enables inter-picture prediction using already coded pictures from other views at the same time instance. The approach is different from the one used by *motion compensated prediction* (described in Chapter 2.1), where the inter-picture prediction was achieved using already coded pictures from the same view at different time instances. The *inter-view motion parameter prediction* enables the prediction of motion parameters in a dependent view from an already coded and transmitted view at the same time instance, thus it assumes that different views of the same scene have similar motion parameters. Finally, the *Inter-View Residual Prediction* assumes that different views of the same scene also have similar residual signals hence the coding efficiency in a currently coded dependent view can be improved from a reconstructed residual signal of an already coded view.

Furthermore 3D-HEVC introduces new intra coding techniques, called *depth modelling modes*, and modifies the *motion-compensated prediction* and *motion vector coding* for the coding of depth maps.

2.6 3D Video Representation and Coding

The scene set-up used a 4 camera array (camera 1 to camera 4) in a linear arrangement, as shown in Figure 2-8.

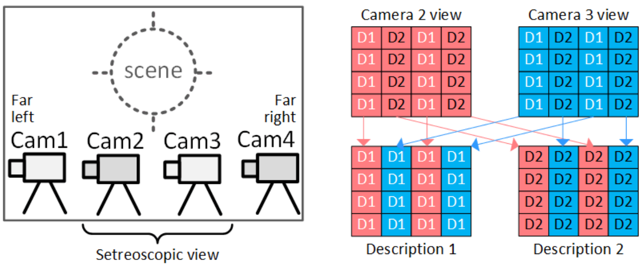

Figure 2-8: 3D video representation and coding with multiple description

The two cameras in the centre (cam2 and cam3) form a stereoscopic pair and the other two cameras (cam1 and cam4) were located on either side of the central stereoscopic video pair at a farther distance than stereoscopic baseline distance. A compressed video stream that results from the format illustrated in Figure 2-8 was obtained by using the SVC extension of MPEG-4 Part 10/ H.264 Advanced Video Coding (H.264/AVC) standard. A poly-phase spatial sub-sampling approach was adopted to create a description out of a single camera view. Both of the descriptions are combined column-by-column to create a full resolution camera pair (or the colour-plus-disparity pair, if side cameras are under concern). If either of the descriptions is lost or discarded, the full resolution representation is reconstructed by interpolating the even (odd) columns from odd (even) columns. Each formed description is effectively treated as a video stream and SVC is applied on each description.

Table 2-3: Complete list of the video streams needed for the provided (4 camera) set-up

#	Content type	Description	Tx means
Camera 1			
1.	Base Layer Half colour + Half depth	1	IP
2.	Base Layer Half colour + Half depth	2	IP
3.	Q. Enhancement Half colour + Half depth	1	IP
4.	Q. Enhancement Half colour + Half depth	2	IP
Camera 2 + Camera 3			
5.	Base Layer Half colour	1	**DVB-T2**
6.	Base Layer Half colour	2	IP
7.	Base Layer Half depth	1	IP
8.	Base Layer Half depth	2	IP
9.	Q. Enhancement Layer Half colour	2	IP
10.	Q. Enhancement Layer Half depth	1	IP
11.	Q. Enhancement Layer Half depth	2	IP
Camera 4			
12.	Base Layer Half colour + Half depth	1	IP
13.	Base Layer Half colour + Half depth	2	IP
14.	Q. Enhancement Half colour + Half depth	1	IP
15.	Q. Enhancement Half colour + Half depth	2	IP

The centre stereoscopic camera pair (cam2 and cam3) is split to 1 DVB-T2 stream and 6 P2P/IP streams (3 descriptions – 2 quality layers in each). Side cameras (cam1 and cam4) are split to 4 streams each (2 descriptions for each camera – 2 layers in each description). Different from the central stereoscopic camera pair, the descriptions are formed by combining the disparity map and the camera in the same video frame. The full set of produced video streams is depicted in Table 2-3.

2.7 Video Streaming Technologies

Video streaming in the current Internet can mainly be achieved in three different ways: Real-Time Streaming Protocol (RTSP) with Real-time Transport Protocol (RTP) streaming (through UDP or TCP), HTTP streaming or adaptive HTTP streaming (TCP only). Today, the vast majority of Internet applications use HTTP streaming and adaptive HTTP streaming, mainly because: (i) it avoids the need for a media control server, such as an RTSP server (reducing the cost of a large-scale deployment over the Internet); (ii) of the pervasive characteristics of HTTP approaches (e.g., can bypass security rules imposed by firewalls) and; (iii) the client (rather than the server) maintains the intelligence to determine which piece of information to send next, considerably improving server-side scalability.

2.7.1 *Streaming using RTSP with RTP*:

RTSP [19] was one of the first solutions for the setup and control of the delivery of data with real-time properties (such as audio and video). It was developed to control multiple data delivery sessions (play/pause/resume/teardown, repositioning of playback, fast forward and rewind), provide a means for choosing delivery channels such as UDP, multicast UDP and TCP, and delivery mechanisms based on RTP [21]. The current version, RTSP 2.0 (released in December 2016), is not backwards compatible with the previous version. The reasons why RTSP 2.0 was decided not to be backwards compatible with RTSP 1.0 can be found on [21].

In the vast majority of cases, RSTP uses RTP as the media delivery mechanism, hence RTSP is considered to be an out-of- band protocol – the RTSP messages are sent over a separated channel (different ports) from the media stream, whose packet structure is defined by RTP. Thus, when using RTSP, the actual media/video content is encapsulated in RTP packets which include payload type identification, sequence numbering, timestamping and source synchronization (each source may have its own independent stream of packets). RTP is commonly used for transporting formats such as PCM, ACC, and MP3 for sound and MPEG and H.263, H.264 and HEVC/H.265 for video. RTP, however, does not provide mechanisms to ensure time/sequence/order delivery or provide any quality-of-service guarantees, as it relies on lower-layer services. If streaming stored media (i.e., not live media) RTP over TCP is used and the transfer of the media content can be compared to a file transfer. If streaming real-time media, than RTP over UDP is the preferred method, however it may not work on the current Internet due to firewall rules (historically, UDP traffic is blocked by firewalls). As an alternative, RTSP has a specific mode of interleaving the RTP and RTCP packets onto the existing TCP connection being used for RTSP, see section 14 in [19]. When this mode is used, the client connects to the RTSP server and all communications flow over a single TCP connection.

2.7.2 HTTP streaming

In HTTP streaming the media/video file is stored in an HTTP server as an ordinary object with a specific URL. The video content is divided into smaller fragments, called chunks, and a video client plays it by fetching (through HTTP GET messages) the *chunks* in a chronological order. The server sends (through HTTP Response messages) the chunks as fast as the TCP congestion protocol allows it. Since the video is treated as a file, video prefetching (video is download at a higher rate than the consumption rate) is naturally achieved through the use of TCP. It has been shown [20] that when the average TCP throughput is roughly twice the media bit rate, streaming over TCP results in minimal starvation and low buffering delays. Application buffering, TCP buffering, and playout delay complement the video prefetching feature in order to ensure a smooth reproduction of the content. If the client wants to navigate in the video sequence, the *HTTP byte-range* header is used in the HTTP GET request message, specifying a specific range of bytes matching the location in the video the client currently wants to retrieve. It is important to note that HTTP is stateless, so when the server receives such type of request, it forgets about the previous bytes in sequence for transmission, and instead begins sending the bytes indicated in the *byte-range* request. Such condition represents a waste of network bandwidth and server resources [22]. For this reason, many HTTP streaming solutions use only a moderate-size client application buffer, or limit the amount of prefetched video using the *byte-range* header in HTTP requests [23]. Overall, HTTP streaming solutions were widely deployed, but have been consistently replaced by *adaptive streaming* to solve the major HTTP streaming shortcoming - all the clients receive the same encoding of the video, independently of their network conditions.

2.7.3 HTTP Adaptive Streaming (HAS)

HTTP Adaptive Streaming is an umbrella term for various HTTP-based streaming technologies that allow a client to adaptively switch between multiple bitrates, depending on current network conditions [24]. A wide variety of adaptive streaming protocols have emerged recently, being the most well-known proprietary solutions the Apple's HTTP Live Streaming (HLS), Microsoft's HTTP Smooth Streaming (HSS), and Adobe's HTTP Dynamic Streaming (HDS), and the standardized solutions the 3GPP Adaptive HTTP Streaming (AHS) and MPEG Dynamic Adaptive Streaming over HTTP (DASH). From these, DASH is the solution that will be used to describe the concepts of adaptive streaming. In DASH the videos are also split in chunks, and each chunk is now encoded at multiple bitrates. The chunks from different streams (thus different video quality) are transmitted in a way that the client can switch between different streams (different bitrate) if necessary (e.g., if suffering from increased delay or if the end-to-end bandwidth changes during the session). The client selects different chunks one at a time with HTTP GET request messages [25]. Hence, with DASH, the HTTP server has to have a *manifest file* containing the URL and the bit rate for each of the video versions. The client learns about the various versions by first requesting the manifest file. Afterwards the client then selects one chunk at a time by specifying a URL and a byte range in an HTTP GET request message for each chunk. By dynamically monitoring the available bandwidth and client buffer level, and adjusting the transmission rate with version switching, DASH can often achieve continuous playout at the best possible quality level without frame freezing or skipping [26].

2.8 Packetisation Structure Used in This Thesis

In any of the abovementioned mechanisms, when using TCP, the transmission rate (from server to client) can significantly vary due to the TCP congestion control mechanism (a “saw-tooth” variation is common). Packets can also suffer significant delay. Even though it is acknowledged the use of prefetching with application and TCP buffers considerably improves client playout and end-user watching experience, the architecture proposed in this Thesis is built upon the following principles: (i) a DVB-T2 signal will always be available to the P2P clients; (ii) there should be a super-peer (server) with a copy of the content at each enabled ISP; (iii) traffic at ISP core network needs to be drastically reduced; (iv) inter-ISP traffic should be avoided whenever possible.

These principles, mandate the usage of UDP (for pure IP multicast at the ISP’s core network, as explained in Chapter 3), furthermore, Since DVB-T2 standard will be used for DVB broadcasting and it is using MPEG-2 Transport Stream (MPEG2 TS) packets, using the same encapsulation for P2P has an advantage of having a common standard that can be handled by the same tools at the receiving side.

With these in mind each stream identified in Table 2-3 is to be transmitted through MPEG2 TS. The effective use of MPEG2-TS built-in Program Clock Reference (PCR) also facilitates the synchronisation task of the streams received from DVB-T2 and P2P network. The generation of chunks is performed on the encrypted MPEG2-TS packets that contain the encoded content. These chunks are then distributed over the P2P network. The P2P chunks will be small-sized (<MTU size), to avoid IP fragmentation, but variable in size. According to this scheme each video

frame is sliced during the encoding process into NAL units (NALUs), as explained in Chapter 2.4. Each NALU will be encapsulated into a chunk with a unique ID. The resulting packetisation scheme is shown in Figure 2-9. The encapsulation of the NALU directly on top of UDP reduces the overhead in 12 bytes, when compared to the usage of RTP over UDP.

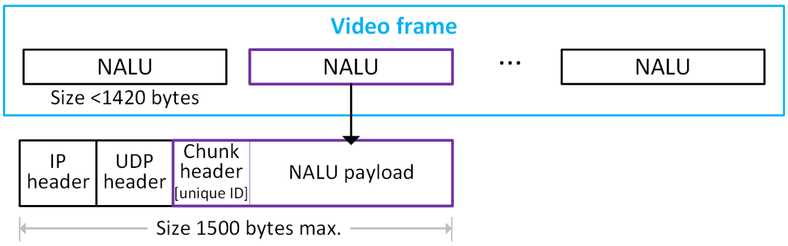

Figure 2-9: Video frame packetisation structure as used by this thesis

The end-user will receive the 'Base Layer Half Colour (cam2+cam3)' stream through DVB-T2 while receiving at the same time (with increased delay) the other IP streams from the P2P network. The decoded video quality will therefore be proportional to the number of received IP streams. It is important to note that each description is sent to the peer in a different IP stream, such that if packets of the corresponding description are lost, then the associated packet in the other description are used to reconstruct the video frame with a slightly downgraded quality.

Figure 2-10 provides an example of the possible streams being received at a decoder. Taking in consideration results from [30][31], it was initially assumed average bitrates between 4Mbps and 5Mbps per stream, meaning each user would require a considerable amount of bandwidth just to consume the stereoscopic view with depth adjustment and fast viewpoint navigation capability (either towards Camera 1 or Camera 4). If the user moves to a viewpoint between Camera 1 and Cameras 2+3, then additional IP video streams will be needed, resulting in additional bandwidth consumption.

For the specific case of the end device to be a multi-view display with multiple view capability (e.g. 8 views at the same time), the streaming server would need to send 14 IP streams which could imply a total of 70Mbps, just for this device. As it can easily be understood, when considering a client-server approach in today's Internet, these numbers can rapidly consume the server's bandwidth. A possible solution to this problem would be the use IP multicast transmission towards the clients, but after many years of experimentation, IP multicast is not currently a ubiquitous service on the public Internet, being mostly deployed on private/corporate networks. The main reasons behind this lack of support are related to inter-domain routing issues, lack of standardized congestion control mechanisms for multicast traffic and inherent multicast security issues, which are essential if multicast applications are to be safely deployed.

2.9 P2P Video Streaming Approaches

Peer-to-Peer (P2P) video streaming approaches are based on their file-sharing equivalents. A (video) content server enables end-users, called peers, to obtain specific video content and make it available to other peers through the use of a diffusion mechanism which is implemented as software called *peer-to-peer client*. In the P2P file sharing paradigm, the peer selection creates an overlay network formed by logical connections between peers. This overlay network can be structured or unstructured, which has resulted in two fundamental architectures: *Tree-based Push Systems* (structured) and *Mesh-based Pull Systems* (unstructured).

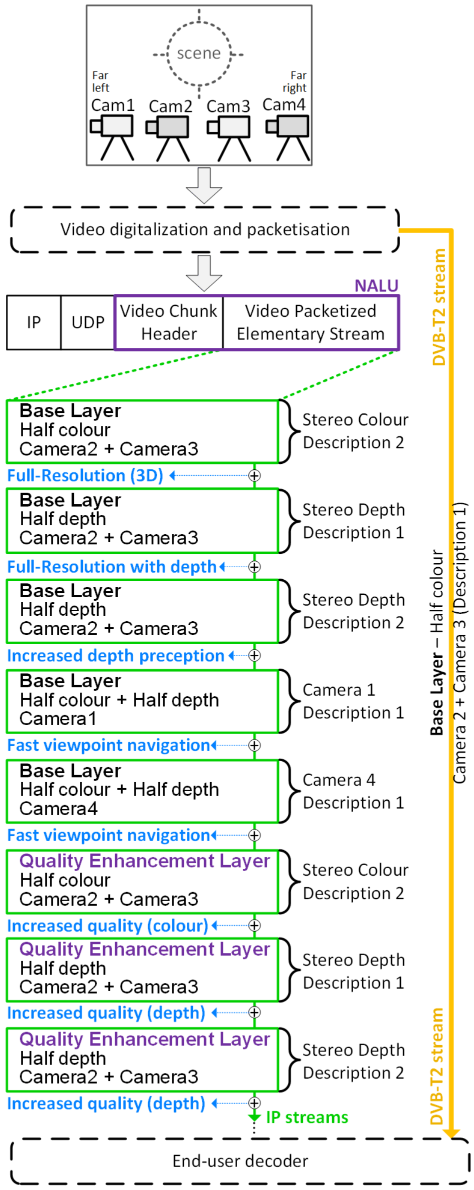

Figure 2-10: Possible subset of IP streams to be transmitted to a decoder.

Studies devoted to the discussion of the design of the aforementioned tree-based push and mesh-based pull schemes have been presented in prior literature surveys [59][60][61]. In a *Tree-based Push System*, a tree-based overlay is formed as an application-layer multicast tree where the video content is pushed from the root (the server) towards the leaves of the tree - as such tree-based push systems are a natural extension of Content Delivery Networks (CDN), discussed at the next subchapter. The concept has been proposed in some seminal works including [62][63][64], and also implemented (e.g., the end system multicast (ESM) [65]). The main research issue in tree-based systems is the overlay formation and maintenance. More specifically, the structure of the trees must withstand peer dynamics and exploit the peer's available resources effectively. Tree-based systems perform well in terms of delay because of the predefined topology. On the other hand, they are prone to churn because the overlay has to be repaired after peer arrival and departures.

On the other hand, a *Mesh-based Pull System* relies on the dynamic formation of logical links between peers. The establishment of logical connections is performed upon mutual agreement of peers, so it requires peers to share the information about their media repository, which guides a peer to pull its desired media chunk from other peers. An example commonly referred to as a P2P stream solution based on the mesh-based system is *PPLive* (http://www.pptv.com). Two main issues are present in mesh-based systems: peer selection and chunk scheduling. Peer selection is the process of selecting peers to establish partnerships with and chunk scheduling is defined as the process of scheduling the pull requests to other peers in order to obtain missing chunks. Mesh based systems can tolerate frequent peer arrival and departures, however, they perform poorly in terms of delay.

Table 2-4 summarizes the characteristics of both these two P2P distribution architectures.

Table 2-4: Feature comparison - *Tree-based Push Systems* vs *Mesh-based Pull Systems*

Tree-based Push Systems (Structured overlay)	**Mesh-based Pull Systems (Unstructured overlay)**
Both tree-based and mesh-based overlays are characterized by a tree-like structure	
In both approaches, peers receive different pieces of video content from different parents and forward these pieces to children peers	
Compared with CDN, the client upload bandwidth is better utilized, and the traffic load on the video server is significantly reduced, leading to a more scalable system.	
Based on application-layer multicast trees on top of which resides the video server.	Connections between peers are formed dynamically according to certain peer selection criteria.
The server maintains the structure of the trees, monitors the overlay, and distributes video chunks in the tree	Distributed construction mechanism
A central server reconstructs the overlay in case of peer departures	The peers self-reconfigure in case of peer departures
The video content is pushed from the root towards the leaves of the tree	The video content is pulled by the peers from their neighbours
In each tree, a number of peers are directly connected to the server, they are called Top-level parents	Peers are randomly connected
Each peer may participate in all the trees. Consequently, each tree is formed by the same set of peers. However, each peer is placed in a different position in each tree.	Peers also participate in multiple distribution trees, but without hierarchical definition
In each tree, a peer has a single source peer called parent and a set of destination peers called children. If the peer is *fertile* it receives video chunks from its parent peer and forwards the received chunks to its children peers.	There is a high peering degree - the topology is a mesh
Due to upload bandwidth restrictions, a peer can be *fertile* in some trees and *sterile* in the others	Similar but in a mesh topology
The upload bandwidth is not fully utilized, as the leaf nodes that account for the major part of the system never share their upload bandwidth.	The suboptimal distribution of chunks due to random pulling may lead to a certain extent of bandwidth waste.
Leaf nodes of the tree are consumers without uploading any content to the other peers (also known as *sterile* peers).	Fair distribution of resources amongst peers
A peer can be repositioned several times throughout the duration of its session due to peer arrivals and departures	A peer can re-compute its peering connections on the swarm several times throughout the duration of its session due to peer arrivals and departures
When a peer is disconnected from a subset of the available trees, it can still receive chunks via the rest of the trees	It is very robust against peer churn, due to the randomness embedded in the peering procedure and also the high peering degree in a system.
Resilience to the peer churn can be improved by using multiple multicast trees - video chunks are sent to each tree in a Round-Robin fashion.	Since there is no central server, chunks arrive according to their availability on the swarm
Within a stable streaming tree, the delay is strictly bounded: its maximum value is determined by the longest overlay path from the root (server) to the leaf nodes.	It cannot guarantee a deterministic delay bound, as the path from one peer to another can be arbitrarily long. In addition, the transmission delay is not the only source of delay and the available bandwidth plays a significant role in delay performance. This results in unpleasant playback experiences, such as long start-up delays and playback freezes.
The complexity of maintaining a stable tree is high in the face of peer churn. In particular, when internal nodes leave, streaming disruptions may happen due to a slow recovery of the streaming tree.	The maintenance complexity is low, thanks to its robustness against system dynamics and also the less rigid logical structure in the overlay.
Buffering is essential in order to accommodate delay diversity among flows traversing different trees	Buffering is required to accommodate chunks that are received at random order and also to schedule requests for missing chunks
Some key references: [64][68][69][70][71][72]	Some key references:[63][66][73][74][75][76][77]

Further analysing key references for P2P distribution systems and checking their suitability for the proposed distribution of the multiple IP video streams the most relevant and inspiring solutions are described below.

DONet (also known as *CoolStreaming*) [66], proposes a P2P based data-driven overlay network for efficient live media streaming; video is divided to segments of uniform length and a buffer map is used to identify each video segment and to indicate if it is available - each node continuously exchanges its buffer map with partners where a special node, called deputy node, is responsible for providing the list of partners to new (joining) nodes. *DONet* also proposes a scheduling algorithm that calculates the number of potential suppliers for each video segment; basically the algorithm starts from those with only one supplier and so forth. If there are multiple suppliers, then the algorithm starts by selecting the one with highest bandwidth and enough available time. According to the authors, the average distance from origin node to a destination node is bounded by O(log n), where 95% of nodes can be reached up to 6 hops. An Internet-based *DONet* implementation, called *CoolStreaming* v.0.9, was released on May 30, 2004.

An unstructured P2P network (called *iGridMedia*) for interactive applications (such as online auction, person interview or video sharing) using a push-pull approach is proposed in [78]. *iGridMedia* aims to provide delay-guaranteed P2P live streaming service over the Internet - safeguarding the ISPs have dedicated servers to support the delay guaranteed interactive applications. For overlay construction, joining nodes must first contact a rendezvous point which is a server maintaining a partial list of current online nodes - then each node randomly finds other 15 nodes to establish a partnership. For the streaming delivery, in the pull mechanism, the video streaming is packetized into fixed-length packets called streaming packets marked by sequence numbers. Each node periodically sends buffer map packets to notify all its neighbours what streaming packets it has in the buffer and then explicitly requests its absent packets from neighbours. Once a packet fails to be pulled, it will be requested again. In the push mechanism, *iGridMedia* evenly partitions the stream into 16 sub streams, and each sub stream is composed of the packets whose sequence numbers are congruent to the same value modulo 16. Once a packet in one sub stream is successfully pulled from a peer, the remaining packets in this sub stream will be relayed directly from this peer. When a neighbour quits or packet loss occurs, the pull mechanism is started again. *iGridMedia* is fully implemented in C++.

Focusing only on application level multicast solutions and tree construction algorithms, CoopNet [68][69] and *SplitStream* [63] are the most interesting. CoopNet is a mechanism for distributing streaming media content using P2P cooperative networking; it uses a centralized approach where a central server is responsible to determine the path of distribution, indicating joining peers to which parent they should connect - the peer hierarchy is decided based on each peer available bandwidth (reported upon connection to the server – periodically afterwards) and their proximity (based on IP/BGPP prefix). *CoopNet* also employs multiple description coding to address the interruptions caused by the frequent joining and leaving of individual peers.

SplitStream is also a multicast mechanism for distributing content in P2P cooperative environments, but contrary to *CoopNet*, it uses a distributed approach; there is no central server and all nodes have the same responsibility - new nodes try to find a parent and join directly to the tree; trees are constructed in a distributed fashion using each peer's upload and download bandwidth. In the distribution of content *SplitStream* divides data in multiple stripes and distributes each stripe per each tree – the source makes stripe selection and multicast each per a designated tree.

Parallel to academic research on P2P networking for the distribution of media content there were several EU projects on 3D media processing, delivery and presentation to the end-user. Some notable results have been found on SEA, 20-20 3D MEDIA, 3D4YOU, P2P-Next, MUSCADE, MOBILE3DTV, DIOMEDES, SKYMEDIA and ROMEO. SEA had aspects related to context-aware networking delivery platform for 2D/3D media. 20-20 3D MEDIA focused on creating complete 3D media-capable chain while considering distribution and networking of spatial media extensively. 3D4YOU had features of defining 3D media delivery formats for broadcasting for home entertainment applications. P2P-Next has aspects on P2P content delivery for user centric media distribution. MOBILE3DTV worked on delivery of stereoscopic 3D video over DVB-H. MUSCADE considered scalable 3D multi-view video via broadcast to home. DIOMEDES concentrated on combining DVB with P2P to deliver high quality 3D media to home. SKYMEDIA has aspects of delivery of 3D multi-view video and finally, ROMEO [31] was built on top of DIOMEDES, further improving the results of combining P2P with DVB-T2 in the delivery of 3D media to home users and by providing a real-time audiovisual communication overlay for remote collaborating parties' interaction. ROMEO was the inspiration for the current thesis.

Overall, the current consensus is that using P2P systems for video streaming is an approach that inherently suffers from various problems including streaming discontinuity due to peer churn and heterogeneity in resources, which impact quality of service, Furthermore existing P2P solutions are based on the distributed nature of P2P networking which inevitably results in randomized, wide-reaching traffic where content existing on topologically close peers is downloaded from distant peers, as illustrated in Figure 2-11.

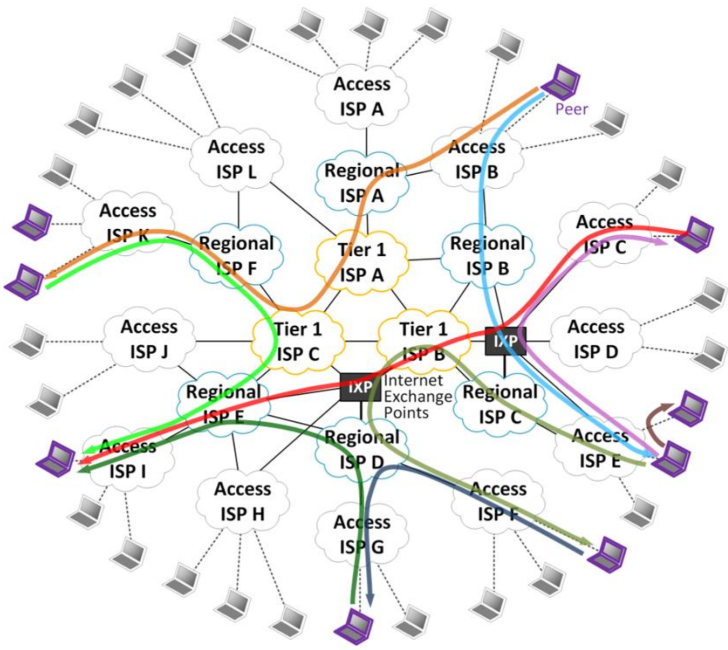

Figure 2-11: Inter-ISP traffic caused by P2P applications (highly simplified view)

Research has shown that the existing P2P systems employ a peer-selection strategy which in the large majority of cases is unaware of the network underlay information, resulting in a largely random connected topology. As a consequence P2P applications provide cheap scalability for content providers at the expense of high network costs for Internet Service Providers (ISPs), a fact which reportedly polarized ISPs attitude towards peer-to-peer systems [32]. P2P traffic throttling reduce costs for ISPs but increase download times for users, inevitably, as stated in [33], ISPs and P2P developers play a game of cat and mouse - with this ‘unfair treatment’ of P2P applications by ISPs, the legality of P2P throttling is in question as ISPs and federal regulators try to reach decision from courts [34].

The biggest challenge for P2P applications is therefore to find a solution that retains the inherent advantages of P2P networking, such as its scalability and efficient use of bandwidth whilst ensuring quick reaction to network dynamics (caused by peer churn) and minimizing cross-ISP traffic, which have led ISPs to throttle down P2P traffic in peak hours.

2.10 Content Delivery Networks (CDN) and Peer Assisted CDN (PA-CDN)

The simplest and easiest way to distribute video content is to use the typical client-server paradigm and stream the videos directly from the server to clients. This approach, however, has some major drawbacks: (i) single point of failure – if the server fails the service is disrupted; (ii) clients may be too far from the server - implying increased end-to-end delay and jitter, increased susceptibility of link failure and heavy inter-ISP traffic signature and; (iii) if many clients connect at the same time, server resources and correspondent subscribed link bandwidth (or traffic quotas) become a bottleneck.

To address these issues, in the late 90s, the concept of a Content Delivery Network (CDN) started to emerge. CDNs make use of multiple servers, scattered around multiple geographically distributed locations. These servers, also known as *cache servers*[‡] store copies of the content and serve clients nearest to their location. This concept is illustrated in Figure 2-12.

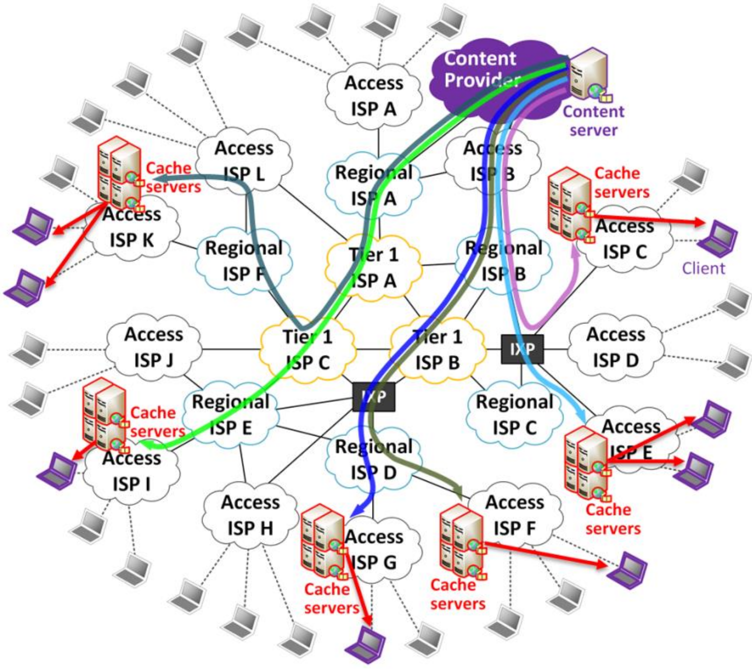

Figure 2-12: Concept for a Content Delivery Network

Well known CDNs are *Google CDN* (YouTube), *Akamai's CDN* (also used by YouTube and Hulu), Netflix Open Connect (Netflix), Limelight Networks (also used by Hulu) and Level3 (also used by Hulu). An overview of these and other commercial CDNs can be found on [35], whilst an interesting study of Netflix, Hulu and their CDNs is provided in [36].

In what respects the location of the *cache servers*, there are mainly two different approaches [37]:

- *Enter Deep*: consists on deploying the *cache servers* in ISPs all over the world. This approach allows content to be available close to the end-users with the best possible bandwidth and avoiding the issues mentioned at the beginning of this chapter.
- *Bring Home*: consists on a less dense approach, by deploying the *cache servers* in a much smaller number of locations around the world, but providing each location with increased resources and connecting these using a private high-speed network.

‡ *Cache servers* are also known by other names, such as, *edge servers* or *edge points of presence*

When comparing both, the *Enter Deep* approach has a highly distributed design which may provide optimum throughputs with lowest delays. This design however considerably complicates the managing of the remote locations. On the other hand the *Bring Home* design provides a lower management overhead but end-users may suffer marginal lower throughput and a slight higher delay.

One important aspect of CDNs is related to the selection of the best cache servers for a specific end-user. The following are three possible approaches:

- *Geographically based*: this approach can be addressed in two different ways: (i) the CDN service provider uses the client IP address to map its geographic location (some providers of such service are Neustar [38] and MaxMind [39]) and hence determine the cache servers closest to it – this, however, may not always work, as the precise location of the client IP addresses is not always known -, and; (ii) use Domain Name System (DNS) resolution to intercept and redirect requests to the appropriate cache servers (detailed information on how this can be achieved is provided in [40]) – unfortunately, this approach will not work when clients are configured to use remote DNS servers (such as the well-known Google DNS servers 8.8.8.8 and 4.4.4.4). In these situations the DNS server and the client will mostly be far away from each other.
- *Performance based*: this approach also makes use of the DNS interception as described in the geographically based approach to provide clients with different cache server locations (from time to time). The goal is to measure network conditions between specific cache servers and clients. Such measurements can be done directly through specific tools or, indirectly by measuring response times in common client-server transactions (e.g., during TCP three-way-handshake) – even though the redirection to different cache server locations does not occur frequently, when it does, it may cause unsatisfying viewing conditions.
- *IP Anycast based*: the usage of IP anycast [41] enables a particular address to be available in multiple locations. Packets destined to an IP anycast address are hence routed to one of several available locations. When using this approach, the CDN service provider needs to assign the same IP address in each of its cache servers' locations. Since Internet routing is performed through the Border Gateway Protocol (BGP), the CDN service provider will also need to advertise this IP address in at least one BGP router per location. The behaviour of BGP is to treat these route advertisements as different paths to the same location when, in fact, they refer to different locations. When receiving a packet for this IP anycast address, the first router will route it according to its BGP routing table, if there is only one path, the BGP router forwards the packet through that path, if there are multiple equal cost paths, then the BGP router forwards the packet to the different

interfaces in the path. This procedure goes on until the packet reaches the closest cache server location. After that a server in the cache server's cluster responds to the client and the video streaming process starts.

Other important aspects when designing CDNs are: (i) the location of the cache servers [42]; (ii) load balancing algorithms, to ensure clients are redirected to the cache server location capable of providing the best service; (iii) provisioning of enough resources to each cache server cluster to ensure highest throughput and minimum delay; (iv) safeguarding proper service level agreements (SLAs) are in place between the CDN service provider and the anchoring ISPs and; (v) what content to replicate [43][44][45]. Additional information on these aspects can be found on [46].

Even considering all the advantages brought by CDNs, current and future demands for video traffic on the Internet is/will be able to push CDNs to their limit [47][48]. The major drawback of the CDN model is its inability to take advantage of the upload bandwidth of the clients, which effectively puts the entire load onto the CDN infrastructure.

The search for alternative solutions has led to an emerging and attractive solution known as *Peer Assisted CDN* (PA-CDN[§]). The core principle of PA-CDNs is to merge the advantages of the two worlds, P2P and CDN. Thus, PA-CDNs enable the distribution of (video) content over the Internet at a very large scale, with good resiliency and quality of service assurances whilst considerably lowering infrastructure and bandwidth associated costs of the CDN service providers. PA-CDNs solve the major issue of P2P distribution networks (which is content throughput being considerably crippled by peer churn) by making the CDN cache servers having a copy of the content. In such cases, the cache servers will only have to support few clients. If the content is being requested by multiple peers, there is sufficient capacity to deliver content in the P2P swarm, and the load on the cache servers can even drop, as the content could be mainly shared through centrally managed P2P swarming algorithms [49][50][51]. Furthermore, based on the findings in [52][53][54], PA-CDNS can offload between 50% to an astounding 88% traffic from CDNs.

The great majority of PA-CDNs use a *centralized architecture*, whose basic operation proceedings can be summarized as follows: upon redirection to the nearest cache server location, the peer starts receiving the first video chunks from one of the cache servers. It also receives a list of peers to whom it should connect in the P2P swarm. If the connection to the swarm is unsuccessful, the cache server will send a refreshed list with new peers and, if still the capacity of the swarm is still insufficient, the cache server serves the peer directly. Thus the CDN has full control of the P2P network, which facilitates management and increases security. This approach is

---

[§] The concept of PA-CDN is also known for other names, such as *Hybrid P2P* or *Hybrid CDN and P2P* used by well-known PA-CDNs, such as KanKan, YouTube and Spotify as documented in [55]-[58].

2.11 QoS in Packet Switched Networks

The IP network convergence [79] has imposed the Internet as being the core infrastructure to attach various access technologies and to assure the transport of all kind of services and applications (e.g., data, audio, and video). A major challenge in the convergence environment is that each service or application has its own QoS requirements for bandwidth, latency, jitter, and loss. For example, applications like Voice over IP (VoIP) require 150ms of (mouth-to-ear) delay, 30ms of jitter and no more than 1% packet loss [80]. The interactive video or video conferencing streams embed voice call, and thus have the same service level requirements as VoIP. In contrast, the streaming video services, also known as video on-demand, have less stringent requirements than the VoIP due to buffering techniques usually built in the applications. Other services such as File Transfer Protocol (FTP) and e-mail are relatively non-interactive and delay-insensitive. This means that the Internet must be able to treat each service according to the service requirements and efficiently utilise the network resources for operators to maximise revenue. However, the legacy Internet system was not designed for these purposes; it treats all the services equally in a best-effort fashion. In order to make the Internet more attractive, the QoS provisioning consists of designing algorithms and protocols to enable the network to provide predictable, measurable, and differentiated levels of quality guarantees.

As a first enhancement to the best-effort service, is the classification of traffic into classes, and use these to provide different levels of service. Even though such feature was available since the first release of the IP protocol (1979), by using the *Type of Service* field in the IP header (currently renamed to DSCP/ECN[**] field), its adoption has not been successful till the late 90s, when the first networks integrating audio, video and data start to emerge. There are 3 main QoS principles associated with this vision of service differentiation: (i) *Classification & Marking* – enables edge/ingress routers to recognize to which application packets belong and mark them according to a QoS policy; (ii) *Isolation & Policing* – provides isolation between the different traffic classes and guarantees each will behave according to a specific policing mechanism (e.g., leaky bucket) – if classes misbehave, their packets may be delayed or dropped if no sufficient resources exist, and; (iii) *High resource utilization:* the network resources are used as efficiently as possible, through different scheduling mechanisms (e.g., First-in First-Out (FIFO), Priority

[**] DSCP/ECN – Differentiated Services Codepoint / Explicit Congestion Notification

Queuing (PQ) and Weighted Fair Queuing (WFQ)) – applications may use more resources than predefined if such resources are available.

One well-known QoS architecture making use of these QoS principles is *DiffServ* [81]. In the DiffServ architecture vision, when traffic enters a DiffServ network, called DiffServ (DS) Domain, it is classified and conditioned at the boundaries of the network, by *edge routers* (also called DS boundary nodes which can be *ingress* or *egress* routers depending if traffic is entering or exiting the DS Domain, respectively) or, other DS-compliant entity (e.g., a computer application or a voice over IP phone). When reaching the core network, packets are forwarded by *core routers* (also called DS interior nodes) according their tagged classification (called DS *codepoint*) in the DSCP field[††], as mentioned before. When a DS-marked packet arrives at a core router, the packet is forwarded to the next link according to a per-hop behaviour (PHB), associated with that packet's DS codepoint (Standardized PHBs have a recommended codepoint). The PHB is the forwarding behaviour of the router and it basically influences how the router's buffers and link bandwidth are shared among the competing classes of traffic (hence influence delay and packet loss). Since the routers' forwarding policy is on a hop-by-hop manner, according to the DS codepoint value, in the Diffserv architecture there is no need for routers to keep state of the multiple existing source-destination flow pairs. This is one of the major advantages of the DiffServ architecture when compared to other solutions – it scales very well.

Furthermore, to facilitate the usage of the PHB, the Internet Engineering Task Force (IETF) has standardized two main PHB groups:

- *Assured Forwarding* (AF) [82]: is a forwarding behaviour that divides traffic into four AF classes, as depicted in Table 2-5, where each is guaranteed to be provided with some minimum amount of bandwidth and buffering.

Table 2-5: DifSerf recommended DSCP values for AF classes

	Class 1	**Class 2**	**Class 3**	**Class 4**
Low Drop	001010 **AF11** DSCP value: 10	010010 **AF21** DSCP value: 18	011010 **AF31** DSCP value: 26	100010 **AF41** DSCP value: 34
Medium Drop	001100 **AF12** DSCP value: 12	010100 **AF22** DSCP value: 20	011100 **AF32** DSCP value: 28	100100 **AF42** DSCP value: 36
High Drop	001110 **AF13** DSCP value: 14	010110 **AF23** DSCP value: 22	011110 **AF33** DSCP value: 30	100110 **AF43** DSCP value: 38

NOTE: the dropping algorithm must treat all packets within a single class and precedence level identically.

[††] Diffserv uses six bits of the IP header to identify the DS Codepoint (DSCP), which selects a PHB

- *Expedited Forwarding* (EF) [83]: is a forwarding behaviour that intends to provide low delay, low jitter, low packet loss and assured bandwidth. In a sense the goal is to provide a PHB in which EF marked packets will usually encounter short or empty queues (hence packet loss will be minimum). The recommended DSCP value for the EF class is 101110 (46 in decimal).

Thus, Diffserv is typically deployed in a single administrative domain (such as a single ISP). With proper configuration and network dimensioning, the highest priority class can achieve extremely low packet loss and delay. However, to achieve end-to-end QoS between customers of two different ISPs, Service Level Specifications (SLS) and Traffic Conditioning Specifications (TCS) would need to be established between each ISP in the path between these two ISPs, a task not easy to achieve. Furthermore, the QoS mechanisms used by DiffServ can only ensure each class service over some reasonable amount of time therefore it cannot ensure that a specific class (even the higher priority classes) will be able to enjoy their desired service throughout the entire duration of the flow.

This issue is addressed by another well-known QoS architecture, known as the *Integrated Services* (IntServ). The *Intserv* operation principle is very similar to that of the landline telephone network – when two users want to communicate: first, the caller signals the network about his intention to establish a call, providing a destination address (phone number); second, the network checks to see if there are enough resources, and if there are, it signals the called party (if there are not, the call is terminated – this is called call admission); third, the called party either accepts or rejects the call – if accepting, the network is signalled on the acceptance; fourth, the circuit is established and the call takes place. The QoS is maintained throughout the entire duration of the call and all the devices involved on the call will only release the resources associated with this call, after it ends (which can be terminated by either party).

The most well-known example of the IntServ Architecture is the Resource Reservation Protocol (RSVP) [84], which allows reserving appropriate amount of bandwidth on a path between a source and destination, as depicted in Figure 2-13. With RSVP the following operations apply:

- *Call Signalling*: specific RSVP signalling messages are used by end-devices to request specific QoS parameters from the network for a particular flow. Routers also use RSVP signalling messages forward these QoS requests to all other routers along the path(s) of the flows. These signalling messages allow to establish and maintain state to provide the requested service and carry the overall end-to-end decision of whether or not the call has been able to reserve sufficient resources at each and every router on the end-to-end path.

- *Call Admission*: in order to ensure no degradation occurs on already established RSVP sessions, and considering resources are not infinite, a resource reservation will be denied if the requested resources are not available.
- *Resource Reservation*: RSVP reservation requests consist of a flow descriptor which is constituted by a flow specification (*flowspec*) together with a filter specification (*filter spec*) - the *flowspec* specifies a desired QoS whilst the *filter spec identifies* the flow. Importantly, the reservation request message originates at a receiver and is passed upstream towards the sender. Once resources are reserved, the call enjoys these resources throughout its duration, regardless of the demands of all other calls.

In this sense, IntServ mandates that not only the edge and core routers support these new functionalities (signalling, admission and resource reservation) but the end-points (e.g., servers and personal computers) too. Even more, all the devices need to keep the state of all the connections/flows where they participate. These, together with the excessive number of QoS control signalling events, and therefore the related long session setup time, are the “Achilles’ heel” of the Intserv architecture.

There are however other key relevant standards for QoS approaches, these include the IP Multimedia Subsystem (IMS) from the 3GPP, the Resource Admission Control Function (RACF) from the ITU-T (International Telecommunication Union Standardisation Sector) and the Resource and Admission Control Sub-system (RACS) from the TISPAN (Telecommunications and Internet converged Services and Protocols for Advanced Networking) [85].

Traditionally, the resource reservation is performed on per-flow basis, meaning that the QoS signalling messages are triggered upon every service request [86]. As a consequence, the approach has been criticized in the research community due to the lack of scalability [88].

To ease the understanding, Figure 2-13 illustrates the processes related to QoS-enabled path setup for establishing a new service session, the path maintenance during the session lifetime, and the path release when the session terminates.

Hence, when an end-user wants to consume media content from a remote media server, see Figure 2-14, the UE needs to send a service request to a QoS Broker located in the Network Operations Centre. Note that the QoS Broker is responsible for the overall control of the network.

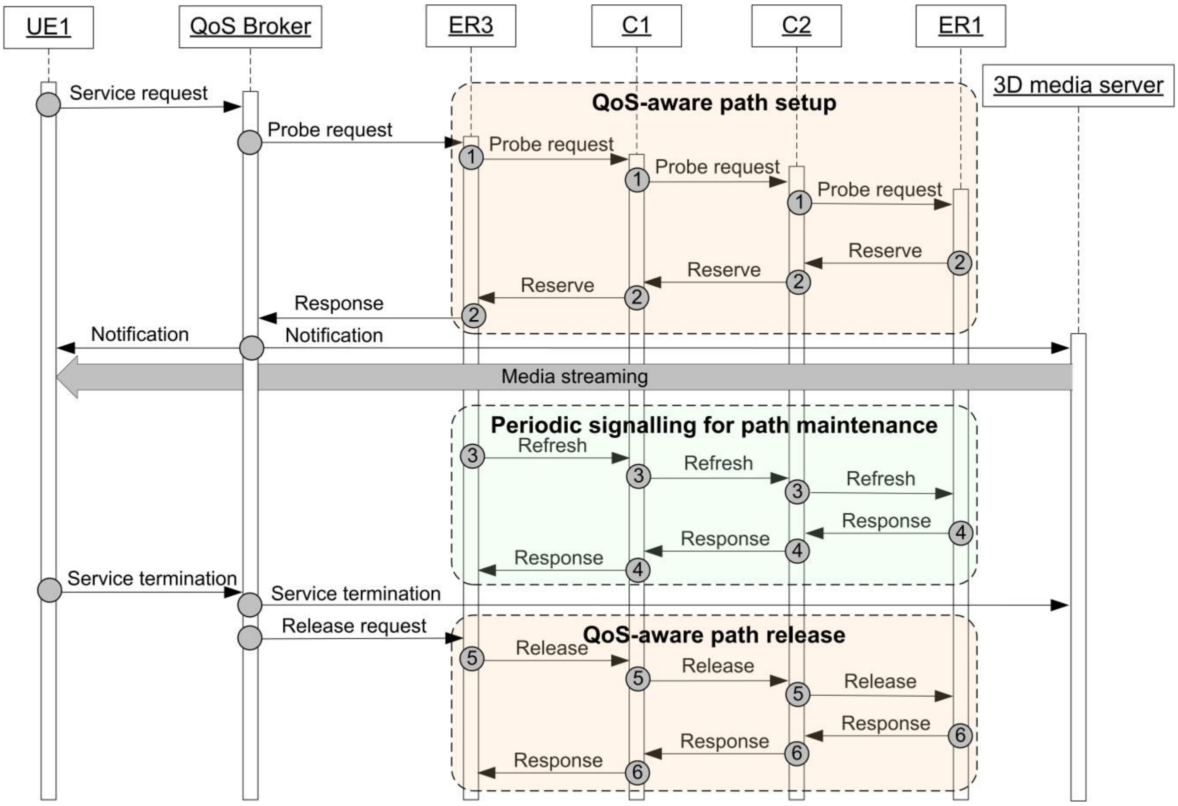

Figure 2-13: Per-Flow QoS-aware communication path control.

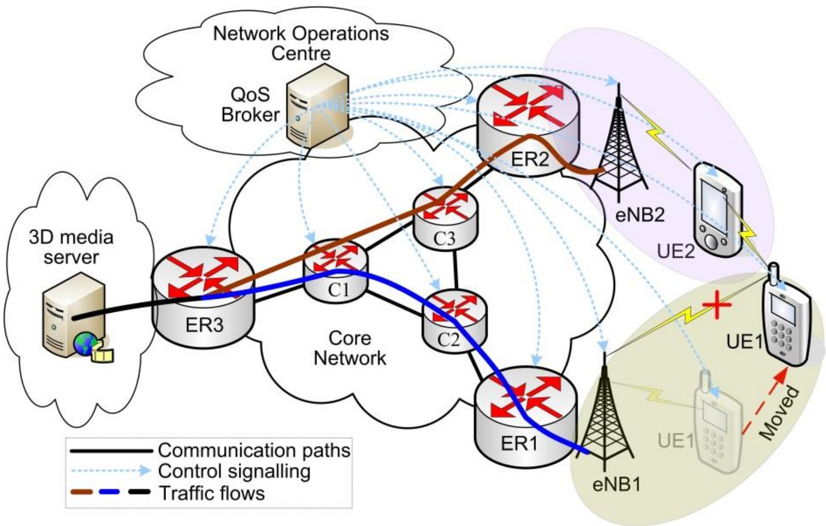

Figure 2-14: Illustration of users and networks dynamics.

In this sense, the QoS Broker triggers the Edge Router 3 (ER3) to check whether the available network resource is sufficient to transport the requested media with acceptable quality. For this purpose, ER3 sends a Probe message on a path, which may be provided by the underlying routing protocol, (e.g. Open Shortest Path First (OSPF) [89]), to acquire the minimum unused resource on the path. Assuming that there is sufficient available resource, the receiving ER1 signals the path in the reverse direction, so the required resource is reserved on each node on the path and the session is established afterward. Otherwise, the request must be denied or admitted in best-effort Class of Service (CoS), depending on the Service Level Agreement (SLA) between the provider and the end-user. Then, as long as the media session is active (running), the Refresh and Response signalling messages are periodically (e.g., every 30 seconds) issued in such a way as to maintain the reservation parameters on the path. When the media session terminates, the Release and Response messages are used to release the reserved resources for future use. It is noted that periodic messages generated by independent network nodes can become synchronized and contend with other network traffic for link and forwarding resources, as shown in [87]. Any mechanism that makes use of periodic messages must avoid (or minimize) message synchronization by including some randomness in the central periodicity value. Also, the refresh time speeds up adaptation to routing or path changes, but increases network and protocol operation overhead.

It becomes clear that the per-flow QoS reservation control is not acceptable from the scalability perspective. In order to address these challenges, resource over-reservation, consisting of reserving more resources than a CoS may need, can be explored so several service requests can be processed without instant signalling as long as the reservation surplus is sufficient to accommodate new requests [88][90][91]. However, the approach imposes a trade-off between signalling overhead reduction and waste of resources as well as QoS violations [92]. ITU-T G.1081 [93] suggests five monitoring points in networks, allowing service providers for monitoring networks and services performance to improve resource utilization. However, existing proposals are mostly based on path probing techniques [94] and show limitations in terms of computational complexity, accuracy, and undue signalling [95].

Pan et al. [96] proposed to over-reserve bandwidth surplus as a multiple of a fixed integer quantity for aggregate flows destined to a certain domain - a Sink-Tree-Based Aggregation Protocol. This solution also does not comply well with network dynamics and fails to efficiently utilise the network resources. The work in [97] over-provisions virtual trunks of aggregate flows based on a predictive algorithm (e.g., past history) without an appropriate mechanism to dynamically control the residual bandwidth between various trunks. Resource over-reservation is also studied in [98] in an attempt to reduce control states and the signalling overhead. Sofia et al. [99] proposed the use of resource over-reservation to reduce excessive QoS signalling load of the Shared-segment Inter-domain Control Aggregation Protocol (SICAP). However, these proposals lead to undesired waste of bandwidth. The Multi-user Aggregated Resource Allocation (MARA) [91] proposed functions to dynamically control bandwidth over-reservation for CoSs, to improve system scalability. However, MARA also shows serious limitations in its resource distribution capability.

In this scope, recent findings proposed new ways for scalable, reliable, cost efficient control design of IP-based network architectures and protocols [100]. In particular, the Self-Organizing

Multiple Edge Nodes (SOMEN) [101] enables multiple distributed network control decision points to exploit network paths correlation patterns and traffic information in the paths (obtained at the network ingresses and egresses) in such a way as to learn network topology and the related links resource statistics on real-time without signalling the paths. As such, SOMEN provides a generic network monitoring mechanism. The Advanced Class-based resource Over-Reservation (ACOR) [103] effectively demonstrated the breakthrough that, it is possible to design IP-based networking solutions with significantly reduced control signalling overhead without wasting resources or violating the contracted quality in the Internet. The Extended-ACOR (E-ACOR) [104] advances the ACOR's solution through proposals of multi-layer aggregation of resource management and a new protocol to efficiently track congestion information on the bottleneck links inside a network without undue signalling load.

In this view, recent research effort claimed that the Internet resources can be efficiently over-provisioned (booking more resources in-advance) in such a way to allow differentiation of QoS control with reduced signalling overhead and increased resource utilization. **These approaches, however, need further investigation to support real-time knowledge of links resource statistics, which cannot be provided through periodic probing, and achieve on-the-fly QoS adaptation as requested by the innovative networking architectural design proposed in this thesis.**

2.12 Conclusions

The Scalable Video Coding (SVC) extension of H.264/AVC, with multiview, was the chosen solution in this thesis for coding the multiple video sequences. This resulted in a total of 15 video streams, being 1 transmitted through DVB-T2 and the other 14 through IP via a P2P overlay network. The stream provided by the DVB-T2 is assumed to be always present at the decoder and will provide a clock reference and a high quality base layer, which can be improved by further decoding the received IP streams.

The architecture proposed by this thesis also builds on top of the PA-CDN concept, with some changes, as will be described in the next Chapter. There are a number of challenges associated with the distribution of the content through the P2P network, namely, the transmission and propagation delay, jitter, the network stability and resilience to peer churn, and its impact to congestion on the ISP's core network.

Furthermore, the existing QoS signalling solutions do not seem to be scalable, or viable in a large scale deployment scenario. The DiffServ approach is incapable to ensure end-to-end QoS and the IntServ/per-flow QoS approach is incapable to scale as desired, due to much overhead signalling overhead - a new solution needs to be found.

Chapter 3

3 Advanced Architecture for Multiview Real-Time Media Distribution

Chapter Outline: knowing the stringent delivery limitations the IP video streams will face when distributed to the end-users, this chapter details a novel architecture, including its components, that potentiates the advantages of both geographical client-server and P2P network paradigms, similar to PA-CDNs. The approach minimizes P2P traffic to cross inter-ISP boundaries and hugely reduces its volume at the ISP core network. The proposed architecture fully exploits the concept of IP multicast, but only at the ISP core network, being the ISP Edge Routers (ER) responsible to convert the multicast IP video streams in unicast streams for the Top-level parents at a local P2P overlay network. These Top-level parents will be the roots for multiple distribution trees at the access network level. Architecture components allow for fast creation of these P2P application-level multicast trees, fast peer repositioning in the overlay and peer self-awareness to quickly react to network failures, such as the ones caused by peer churn. Furthermore, this approach lowers ISP infrastructure and maintenance costs, with few modifications requested at the ISP network. The proposed changes are mainly associated with a novel Quality of Service approach, of upmost importance, to ensure real-time knowledge of link's resources while still maintaining a fast response and low overhead.

3.1 The High-Level Vision

With the proposed *Architecture for Multiview Real-Time Media Distribution*, this chapter proposes to address the following 10 research challenges:

- Distribute bandwidth-hungry real-time media content to a large population in a scalable manner (up to 70 Mbps per consumer, but could be higher in the future);
- Dynamically adapt video quality according to consumer network conditions;
- Lower Service Provider costs in infrastructure and its management;
- Achieve minimum impact on existing traffic load at the Service Provider core network;

- Ensure the distribution of multiview content achieves the minimum inter-ISP traffic as possible;
- Provide almost real-time full awareness of resource availability at the ISP core links;
- Perform QoS resource reservation at the ISP core network with minimum delay (possibly on-the-fly), with the least waste of resources;
- Achieve QoS with minimum control signalling events;
- Achieve minimum end-to-end delay, jitter and packet loss (related to the consumption of the multiview content) even considering the dynamics of network behaviour;
- Re-use existing technologies and standards as much as possible for easiness of adoption.

The study builds upon the PA-CDNs principle and includes merging the multidisciplinary concepts of P2P networking (at access-level) with IP multicasting and network QoS over-provisioning (at core network) to provide an integrated solution that is able to deliver very high speed content over next generation networks in a scalable and effective manner. As mentioned in Chapter 2.10, P2P networking can offload up to 88% of traffic in a content service provider network. In fact, as shown in Figure 3-1, the proposed architecture can be perceived as a large-scale (hybrid) multihoming media distribution system, with high quality stereoscopic 3D content being received from DVB-T2 at all times. To complement and/or enrich the consumer quality of experience (QoE), the download of additional media content (e.g. additional viewpoints or increased quality) is made available in the Internet through a CDN service provider. This content can be directly streamed from a server or from another customer/peer in the network.

The proposed system is developed around a new architecture focused on P2P and networking components, as shown by Figure 3-1. Pivotal to this architecture is a Main Server where the content is stored. In fact, it should be seen as a cluster of servers, for redundancy, load-balancing and scalability reasons, but for simplicity only one will be considered in the subsequent descriptions. The Main Server is associated to a specific DNS domain (typically associated with the service brand) and it is property of the CDN service provider. The architecture also depicts Super-peers that are also property of the CDN service provider, that act as proxies/replicas of the Main Server and are placed in the premises of every ISP that has an agreement with the CDN provider. The Super-peers are responsible to serve peers from a specific geographical area or ISP. The peers are located at the ISP's access networks and represent the end-users who consume the content, whilst the ISP core network is responsible to transport traffic between the Super-peer and the ISP access networks and also traffic to and from other ISPs through the proper implementation of Service Level Agreements (SLAs).

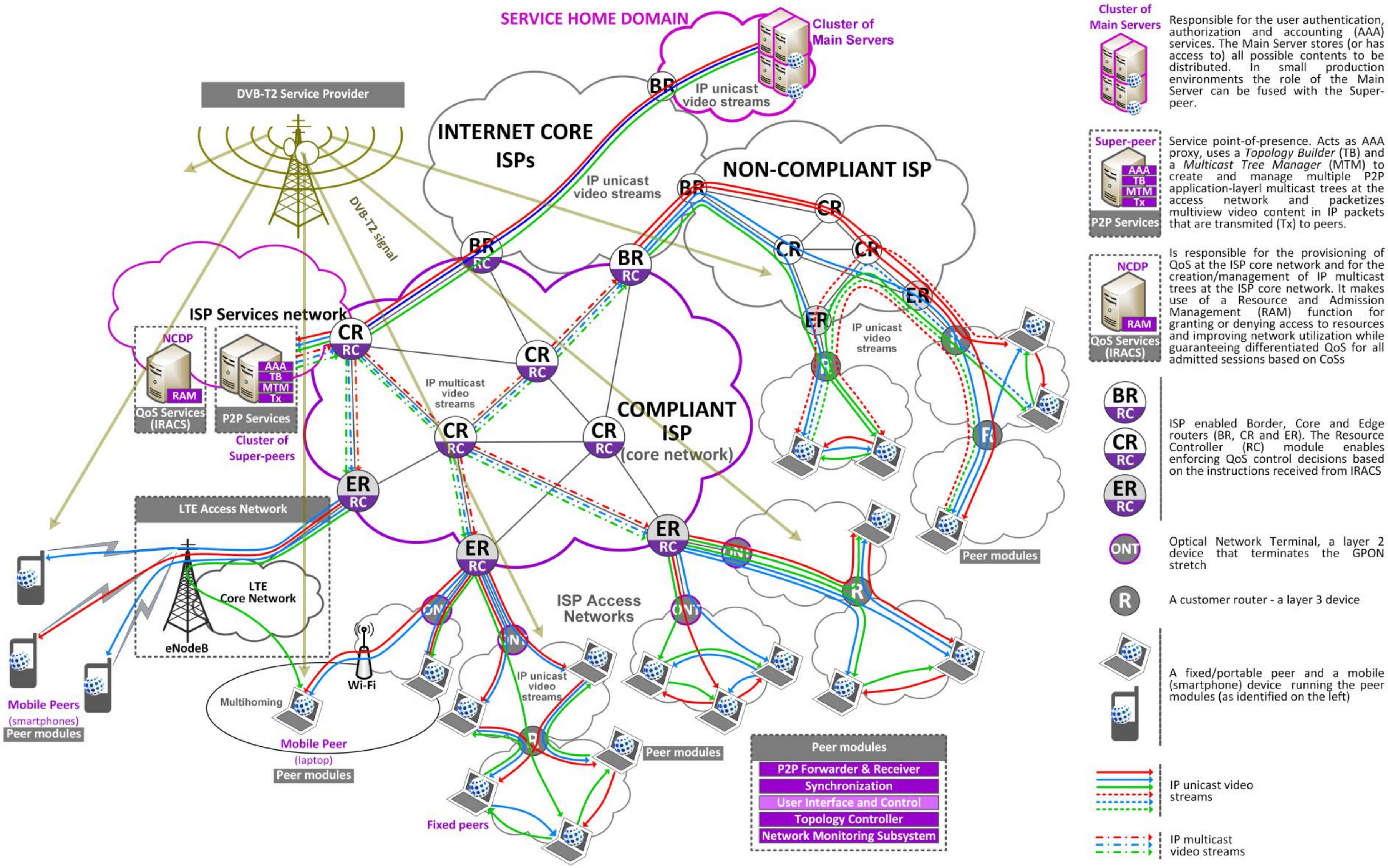

Figure 3-1: Efficient and Scalable Architecture for Real-Time Media Distribution – high-level concept

At the ISP level, for proof-of-concept purposes, a modular point-of-presence topology is considered, where the ISP services are located in a dedicated Services Network – this is the place where the Super-peers should be housed. In such topology, border routers (BR) connect the ISP to other ISPs, core routers (CR) provide internal ISP high speed connections and the edge routers (ER) - also known as distribution or access routers - are high port density routers connecting customers (peers) with the ISP core network. As in any ISP topology, customers also have their own routers, which are not managed by the ISP (these are depicted as an ONT[‡‡] and R[§§] in Figure 3-1).

Subscribers of the service, called consumers, or peers in the presented approach, will connect to the Main Server for authentication, authorization and accounting (AAA) purposes. Upon success, the server will redirect the peers to their nearest Super-peer - this decision takes in consideration the peer's geographic location and ISP (based on IP address information). Peers will then use the provided Super-peer address to request specific content. For each new peer requesting content the Super-peer will compute the peer's position in an application-level multicast tree, effectively distributing the content via a P2P network.

Peers can either assume the role of a parent, a child/parent or a child. Parents receive the content directly from the Super-peer (in fact from the serving ER, as later explained) and occupy the highest level on the P2P multicast tree. Child/parents are the peers that receive the content from another peer and also feed other peers – they occupy intermediate levels on the multicast tree. A child is a peer that receives content from other peers and does not feed other peers - they occupy the lowest level on the P2P multicast tree and can be considered leafs (mobile (smartphone) users are a clear example of leaf peers, since they have considerable restrictions in their download/upload and battery capacity).

In order to improve resiliency and redundancy, the video streams mentioned in Table 2-3, are transmitted in specific multicast trees. The streams that complement each other (to improve visualization quality) are transmitted in the same multicast tree. On the other hand, the streams that provide redundancy to other streams are transmitted in a different multicast tree. Table 3-1 shows the mapping between IP video streams and the correspondent multicast tree.

This concept is summarized and illustrated in Figure 3-2.

[‡‡] Optical Network Termination (ONT): a layer 2 device that terminates the Gigabit Passive Optical Network (GPON) stretch.

[§§] Residential Gateway (R): a layer 3 device in charge of the traffic routing for all services, packet marking, network address translation (NAT) and firewall.

Table 3-1: Relation between the IP video streams and their associated multicast distribution tree

Tree	Content type
1	Base Layer Half colour (Cam2+Cam3)-Desc.2 Q. Enhancement Layer Half colour (Cam2+Cam3)-Desc.2
2	Base Layer Half depth (Cam2+Cam3)-Desc.1 Q. Enhancement Layer Half depth (Cam2+Cam3)-Desc.1
3	Base Layer Half depth (Cam2+Cam3)-Desc.2 Q. Enhancement Layer Half depth (Cam2+Cam3)-Desc.2
4	Base Layer Half colour + Half depth (Cam1) – Desc.1 Q. Enhancement Layer Half colour+Half depth (Cam1)-Desc.1
5	Base Layer Half colour + Half depth (Cam1) – Desc.2 Q. Enhancement Layer Half colour+Half depth (Cam1)-Desc.2
6	Base Layer Half colour + Half depth (Cam4)-Desc.1 Q. Enhancement Layer Half colour+Half depth (Cam4)-Desc.1
7	Base Layer Half colour+Half depth (Cam4)-Desc.2 Q. Enhancement Layer Half colour+Half depth (Cam4)-Desc.2

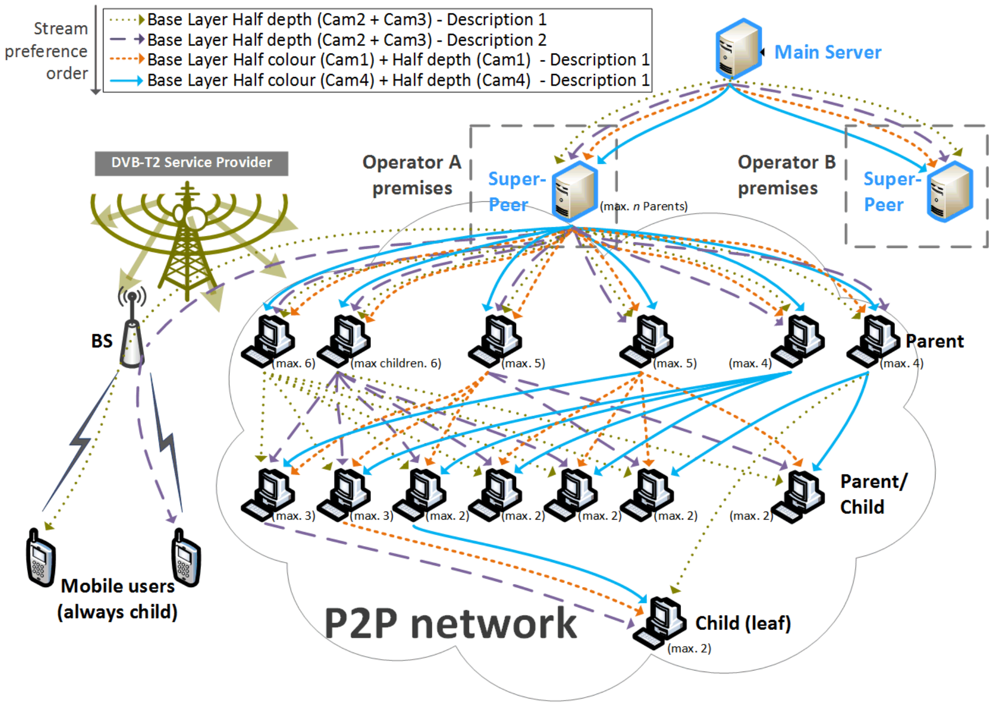

Figure 3-2: Distribution of content using P2P concept and application-level multicast trees. Peers with better resources will occupy a higher position on the tree.

3.2 P2P components

3.2.1 The Main Server and Super-peer

The Main Server is responsible for the AAA services. The Main Server stores (or has access to) all possible contents to be distributed. In small production environments the role of the Main Server can be fused with the Super-peer.

The Super-peer is responsible to serve peers at a specific geographical area or ISP domain. The Super-peer can operate either in a reactive or proactive manner. If a reactive behaviour is used, the Super-peer does not store new content unless specifically requested by a customer peer. If, on the other hand, a proactive behaviour is used, the Super-peer will use non-peak hours to synchronize with the Main Server by storing new content (e.g., based on content request statistics provided by the Main Server). Proactive behaviour has the advantage of providing peers with lower response time and lighter operation on network peak times. The disadvantage would be the need for higher storage capacity.

After peer grouping (by common Edge Router) and peer sorting by evaluation (a concept described in Chapter 3.2.2.1), the Super-peer creates multiple P2P application-level multicast trees split per each ER and content type. This concept is illustrated in Figure 3-3.

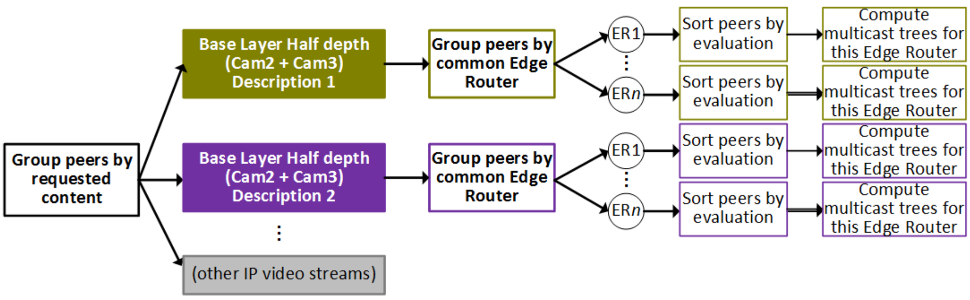

Figure 3-3: Content and location aware construction of P2P application-level multicast distribution trees

To perform its functions the Main Server/Super-peer has three components, a *Topology Builder*, a *Multicast Tree Manager* and a *P2P Packetiser/Transmitter*. Their modules and interrelations are depicted in Figure 3-4.

3.2.1.1 Topology Builder

The *Topology Builder* (TB) is responsible to listen for new peer connection requests, while acting as an authentication proxy (authenticator) for user authentication with the Main Server/Super-Peer. It creates multiple P2P application-level multicast trees for content distribution and decides where peers should be inserted on the P2P trees.

When a peer is redirected to a Super-peer, it is the responsibility of the TB to compute the peer position in the P2P application-level multicast tree at the access network. The steps in the computation are depicted in Figure 3-3 and are: (i) to group peers according to the requested content, see Table 2-3; (ii) group peers according to their common ER - geographical aggregation and; (iii) sort peers by evaluation, a metric explained in Chapter 3.2.2.1.

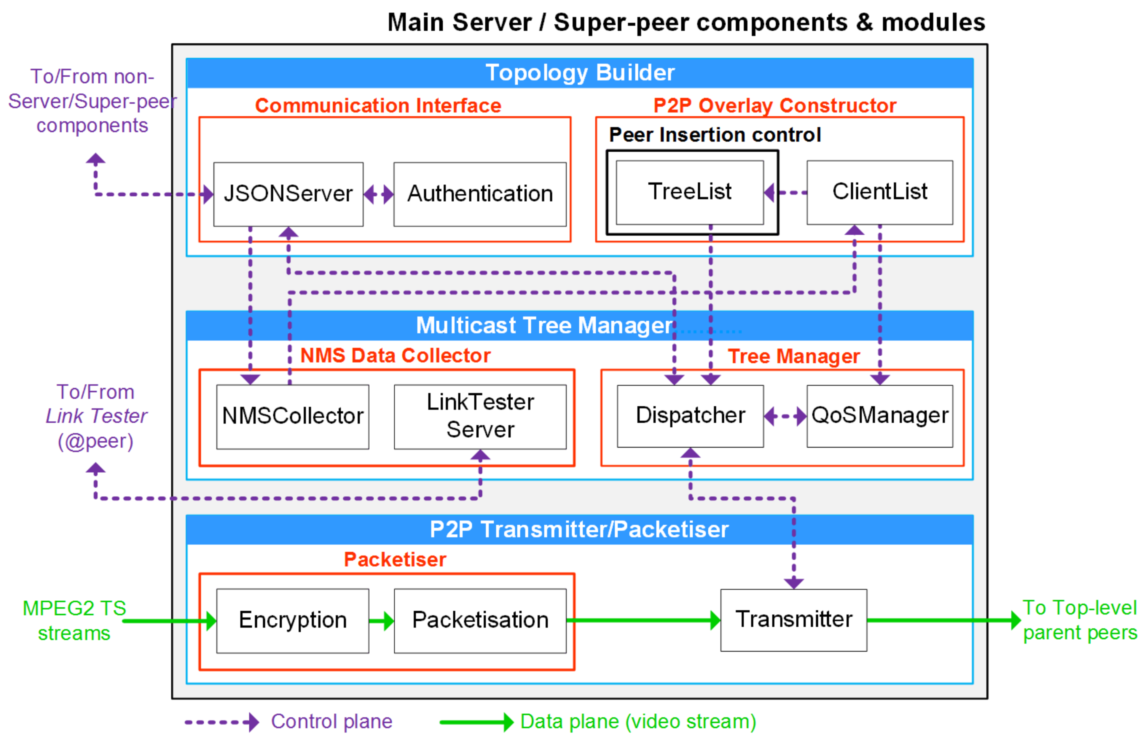

Figure 3-4: Main Server/Super-peer components, software modules and their relations

After grouping and sorting operations the multiple P2P multicast trees are computed, one per each requested content and ER. As it will be explained in Chapter 3.3, traffic at the ISP core will be transmitted using pure IP multicast and it will be the ER's responsibility to map each requested content multicast address to specific parent(s) IP addresses(s) – the ER will effectively act as a replicator. To optimize the ER resources, the Super-peer predetermines how many top-level peers (parents) can be directly fed by one ER. This means that, when constructing each P2P tree, the Super-peer positions a predetermined number of highest ranking peers at the top of the tree and delegates in these the distribution of content to other peers in the same access network. Every time a peer is selected to forward content, its resources are diminished, and if its evaluation becomes lower than other peers, the new highest ranking peer will take the role of parent for additional content streaming. This concept is illustrated in Figure 3-5.

If a peer has insufficient network resources, it will not receive some of the streams, such is the case of peers P10, P14 and P15. Also note that peer P18 will take the role of parent for the content represented in solid line; this may occur because the upload capacity of previous highest ranking peers has been diminished (P11, P12 and P13). To minimize the issues associated with peers joining and leaving the system, also known as *churn*, the TB uses the following mechanisms:

- *Grounding*: new joining peers are always inserted at the bottom of the P2P tree. The algorithm then suffers periodically updates (every *t* seconds) to maximize the efficiency of the P2P tree – promoting and demoting peers. The default periodicity value for the tree maximization process is 15 seconds, this is related to the tree re-insertion delay which is expected to not exceed the 50% of the buffering delay, as explained in Chapter 3.2.3. Furthermore, testbed results (Figure 4-4) have shown the tree computation time to be in the order of milliseconds.

- *Graceful leaving*: whenever possible, peers inform the TB about their imminent disconnection, and should only stop forwarding content to their children when confirmed by the TB or upon a timeout.
- *Redundancy*: when inserted on a tree, all peers will be informed of their active parent and a backup parent. If the active parent is not reachable within a timeout the peer switches to the backup parent and informs the TB. This behaviour is further explained in Chapter 3.2.3.

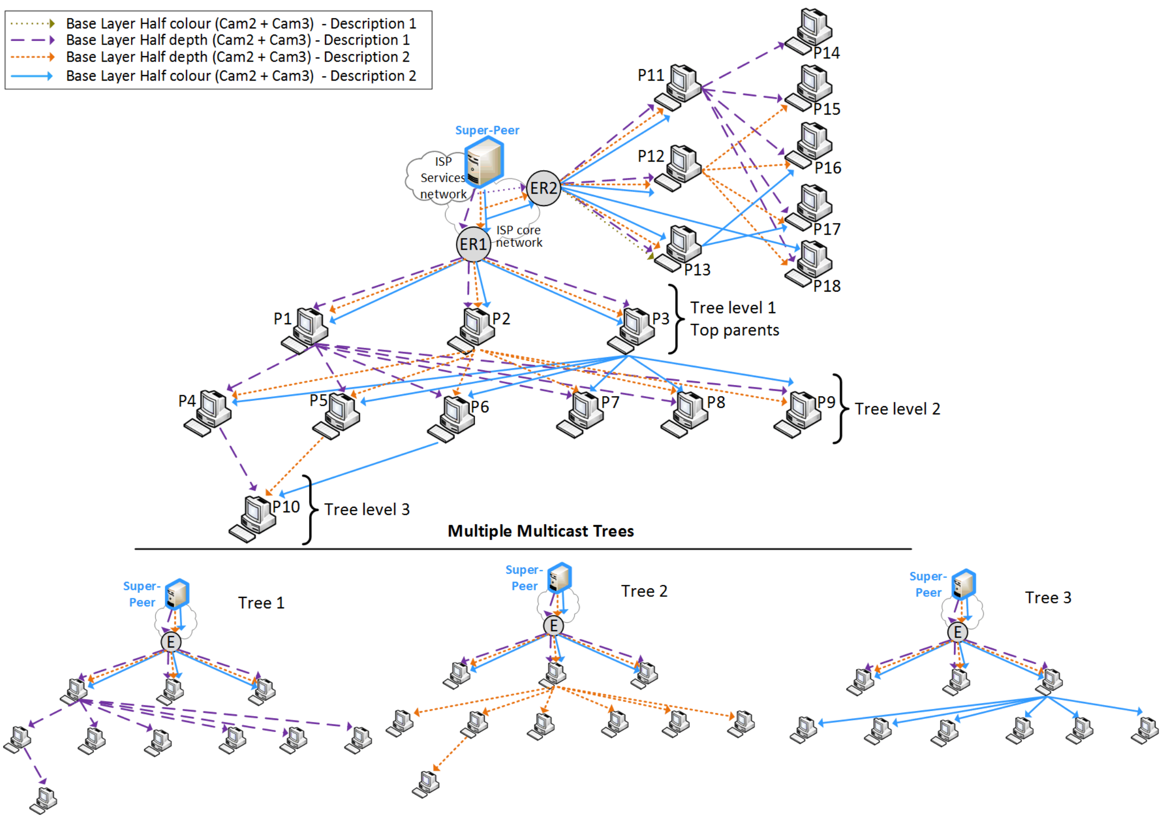

Figure 3-5: Usage of multiple application-level multicast trees at the access network (concept)

This approach brings the following advantages:

- *Resiliency*: by using tree separation, a major fault in a specific zone of the network will not affect other zones.

- *Scalability*: by grouping peers by common ER, tree depth is significantly reduced since peers sharing the same access network have improved downstream and upstream bandwidth, which allows more children per parent.
- *Performance/Quality*: the total number of hops between top level parents and their children is significantly lower, which contributes to reduce the packet/chunk delay, jitter and loss. Recovery from minor faults (such as peer churn) can also be achieved in a faster way, since the backup parent is on the same access network.

Figure 3-6 provides a flowchart description of the P2P tree computation procedures. When a new peer joins the system (peer A in this case), the TB parses the peer request to retrieve the peer evaluation and the requested content, amongst other data. It then creates a peer record and inserts the peer at the bottom of the tree by allocating a parent and a backup parent. It also updates the timestamp associated with this operation and flags the dispatcher that further actions are needed for this peer. The dispatcher then sends a message to the peer, informing on its parent and backup parent's addresses. The TB then further analyses the peer record to identify if this peer can be a parent and if it is stable (a concept explained in Chapter 3.2.3).

After insertion operations are complete the *ChangeNeeded* flag is updated. To maximize the P2P tree efficiency, the algorithm periodically re-constructs each tree using each stored peer record. If changes are needed (e.g., peers needs to be re-allocated) this information is once more signalled to the dispatcher and the process is repeated. These operations are achieved by the following TB's submodules, also depicted in Figure 3-4:

- *Authentication*: responsible to exchange authentication, authorization and accounting messages with the AAA services running at the Main Server.
- *ClientList*: is responsible for maintaining an updated list of all connected peers. If the peer disconnects (graceful or ungraceful), this list is updated in order to remove the peer from the tree.
- *TreeList*: uses the P2P tree construction and maintenance algorithm to compute the multiple P2P application-level multicast trees. As it will be explained in Chapter 3.2.3, peers are responsible to sending network statistics reports towards the *Multicast Tree Manager*. The algorithm uses the updated peer data to compute each peer position on the tree.
- *JSONServer*: is responsible for sending and receiving messages from/to the Main Server/Super-peer components.

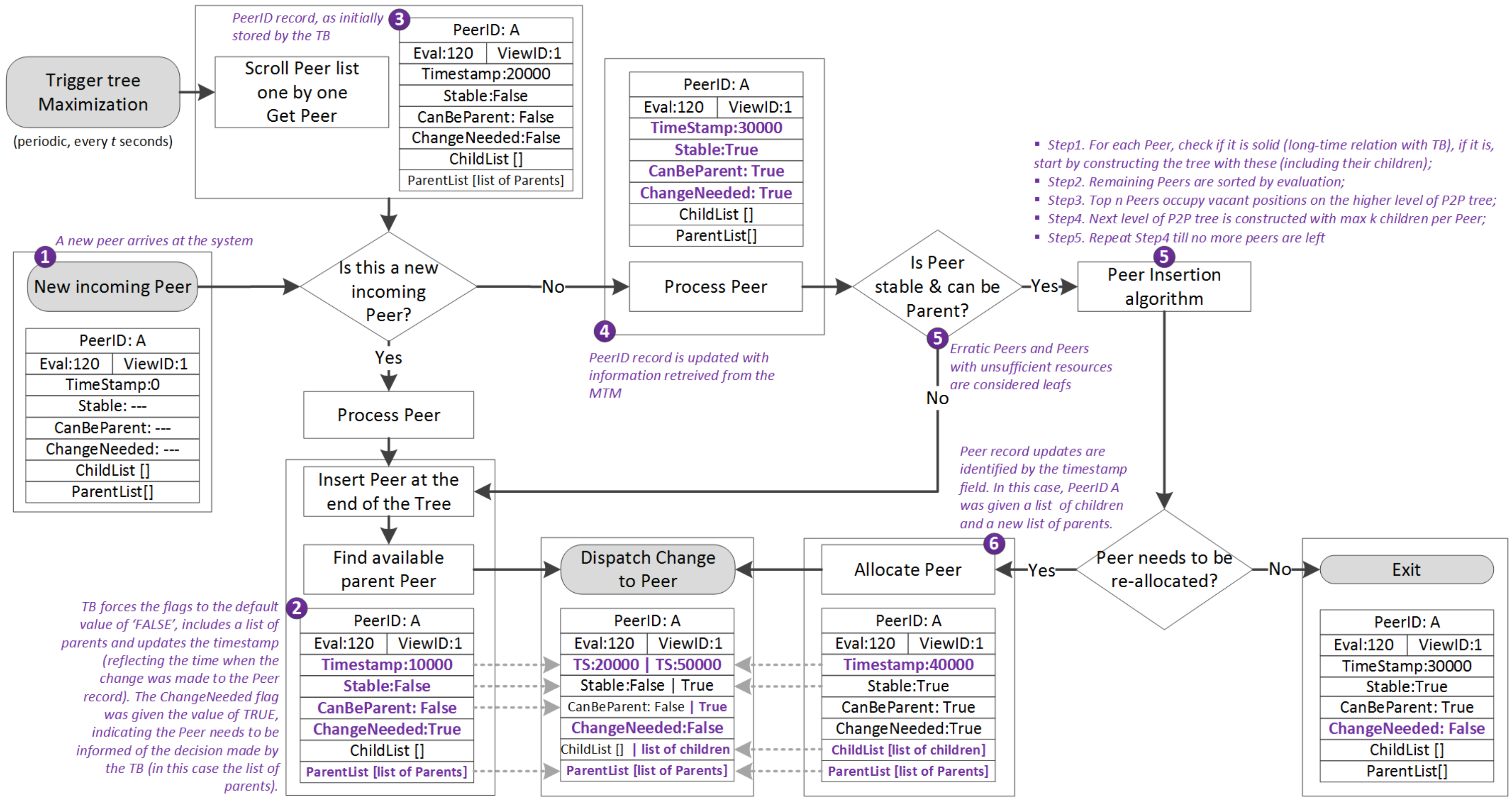

Figure 3-6: Peer insertion procedure at the Topology Builder running at the super-peer

3.2.1.2 Multicast Tree Manager

The *Multicast Tree Manager* (MTM) is intrinsically related with the TB operations. The MTM allows peers to perform bandwidth tests with the super-peer and collects/aggregates network monitoring data (percentage of packet loss, average round-trip delay, jitter and available bandwidth), from all connected peers, providing the TB an updated view of the network conditions. Alternatively, the MTM may choose a passive approach in what respects the measurement of the average round-trip delay and available bandwidth associated with each peer, by analysing response times in common client-server transactions (e.g., during the TCP three-way-handshake, or during the HTTP Response and the client's TCP ACK). However for the network monitoring data specific to the peer's network, it should be done locally by each pear, as it will be described in Chapter 3.2.3.

The MTM is also responsible to inform the ISP's QoS mechanisms, on the endpoints of IP multicast trees at the ISP core network (as later explained in Chapter 3.3.1).

The MTM performs its functions using the following submodules, also depicted in Figure 3-4:

- *NMSCollector*: collects network monitoring data periodically sent by the *Network Monitoring Subsystem* (NMS) running at every connected peer, as it will be explained in Chapter 3.2.3 This information is also shared with the TB for quick P2P tree maintenance operations. It is noted the *NMScollector* runs normally in passive mode (that is, it waits for the NMS reports to be sent), however, the NMS may also adopt an active behaviour and request a specific peer for a NMS report, in order to update its database or resolve database inconsistency.
- *LinkTesterServer*: allows authorized peers to perform bandwidth tests. For the download test it sends a predefined fixed size binary file, for the upload test it expects to receive the exact same file. The results based on this transfer are then used on a composite metric, equation (3.1), to compute its evaluation. The bandwidth test should be performed the first time a peer connects and upon super-peer request (for troubleshooting/maintenance purposes).
- *QoSManager*: as peers connect to the TB, this submodule is responsible to signal the *Internet Resource and Admission Control Subsystem* (IRACS) - an ISP QoS reservation mechanism described in Chapter 3.3.1 on the peers' addresses, so it can check if an IP multicast distribution tree exists from the super-peer to the ERs serving these peers (at ISP's core network level). If yes, no changes need to be done at ISP level. If not, either a new IP multicast tree is created or new branches are added to an existing tree.

- *Dispatcher*: is used to inform the *P2P Packetisation/Transmitter* of the Top Level peers to which it should send the IP video streams. It also interfaces with the TB's *JSONServer* for sending and receiving messages from/to the Main Server/Super-peer components.

3.2.1.3 P2P Packetisation/Transmitter

The *P2P Packetisation/Transmitter* component, also depicted in Figure 3-4, is responsible to encrypt and packetize MPEG2 TS in UDP chunks. Each chunk corresponds to a single Network Abstraction Layer Unit (NALU) - which is a self-contained and decodable video unit, as explained in Chapter 2.6. The chunk header carries information needed for content selection and retransmission mechanisms by the peer modules, this includes the use of unique chunk identifiers (*ChunkID*) per stream.

3.2.2 The Peer

In order to comply with its tasks, four components need to be installed at the peer, a *Topology Controller,* a *Network Monitoring Subsystem,* a *P2P Receiver & Forwarder* and an *Audio/Video Decoding & Renderer,* which also includes an user interface. These components, their modules and interrelations are illustrated in Figure 3-7.

3.2.2.1 Topology Controller

The *Topology Controller* (TC) is the module responsible to establish the peer's initial contact with the Main Server for user authentication and redirection to the nearest (geographical) Super-Peer. After computing the *peer evaluation* (see next paragraph*)* it then informs the TB which specific content the peer is interested in obtaining (e.g. viewpoint changes through the user interface).

The TC is therefore responsible to comply with the P2P tree operations as requested by the TB such as establishing connections with parents (for content request) and accepting connections from children peers (for content forwarding through the *Chunk Selection* module.

The *peer evaluation* is a metric that takes in to consideration the peer's hardware (memory and CPU), the peer's network capabilities (upload and download throughput) and the peer's *stability*, a concept explained in Chapter 3.2.3.

Peer evaluation is calculated according to equation (3.1) and indicates how valuable a peer is in the P2P distribution system.

$$Evaluation = \mathrm{K}_1 \cdot U + \mathrm{K}_2 \cdot D + \mathrm{K}_3 \cdot M + \mathrm{K}_4 \cdot C + \mathrm{K}_5 \cdot S \tag{3.1}$$

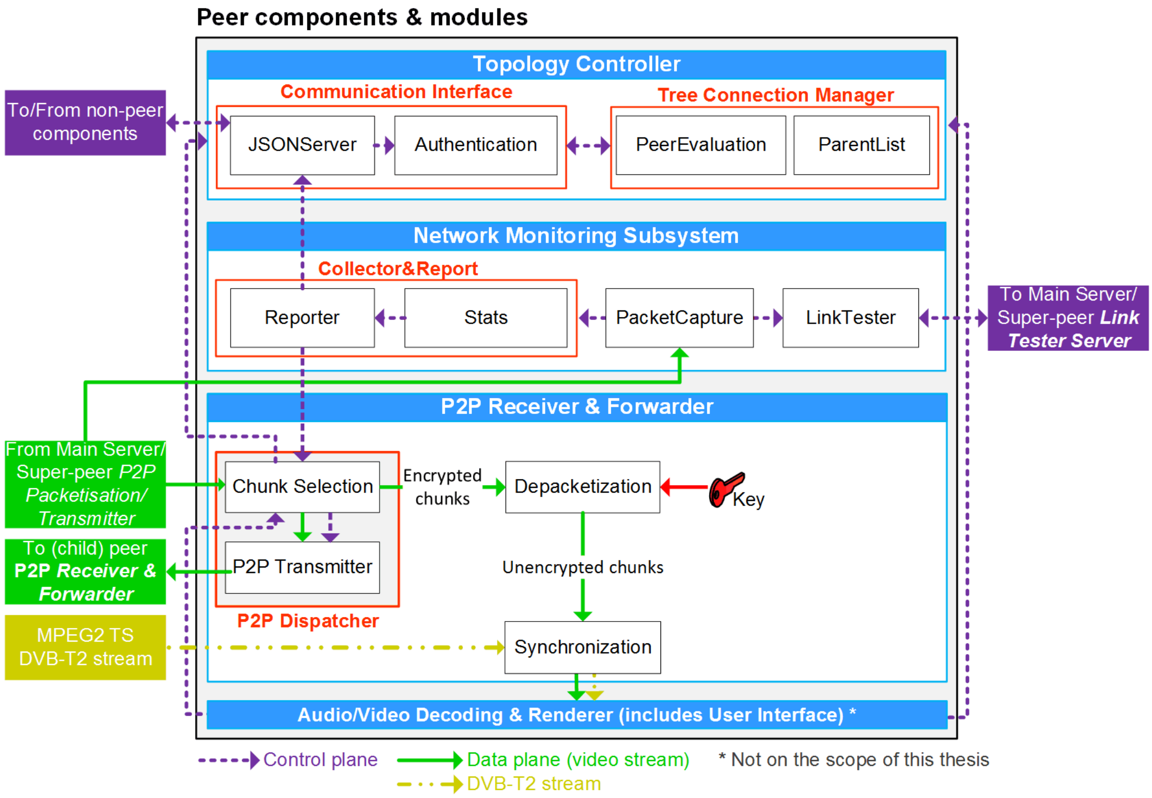

Figure 3-7: Peer components, software modules and their relations

In equation (3.1), K_1 to K_5 are weights that allow normalization and prioritization of the different types of metrics; *U and* D correspond to the peer upload and download bitrates (in kbps) respectively; *M* represents the peer random access memory (RAM) in megabytes; *C* identifies the CPU brand and model, in a scale from 1 to 8, according to Annex 1, and; *S* represents the peer's *stability* as described in Chapter 3.2.3. Considering the high complexity in obtaining an analytical model for computing the optimal values for the K_i weights, the validation approach was based on experimentation. Table 3-2 provides the possible range of values for each K_i weights and the ones that have provided reasonable results for the system during the validation trials.

Table 3-2: Range of values for the K_i weights

	K_1	K_2	K_3	K_4	K_5
Possible range of values	0 to 1	0 to 1	0 to 3	0 to12500	0 to 40
Values used in the Trials	0.5	0.05	0.45	1875	16.4
Suggested fine increment steps	every 0.01	every 0.01	every 0.03	every 125	every 0.4

Note: the proposed incremental steps are based on a 1% ratio, however coarser intervals may be used to ease the fine-tuning of the Evaluation.

These functionalities are enabled through the following modules, as depicted in Figure 3-7:

- *Authentication*: this submodule interacts with the *Authentication* module running at the Main Server/Super-peer. It allows the user to be authenticated and obtain content decryption keys according to its subscriber profile.

- *JSONServer*: is responsible for sending and receiving messages from/to the peer components.
- *PeerEvaluation*: computes the peer evaluation as described in (3.1);
- *ParentList*: contains this peer's list of parents and backup parents (for each content type) as indicated by the TB.

3.2.2.2 P2P Receiver & Forwarder

This component is responsible for achieving synchronization between the DVB-T2 and IP video streams and to forward, in case the peer is a parent, the IP video streams to its child(ren). It has multiple modules and their purpose is as follows:

- *Chunk Selection* (CS): is responsible for controlling the flow of chunks from a peer to its children. There are a few well-known chunk selection strategies (in this context it is important to define the concept of *piece* - chunks are aggregated in *pieces*, thus a *piece* contains multiple chunks): (i) *Strict Priority*, a peer must first download all the chunks of one *piece* before requesting another piece to the same source peer – the reason is that only full copies of *pieces* can be traded with other peers; (ii) *Rarest First*: amongst all the existing *pieces*, the peer will select the one which is more rare, that is, the one with fewer peers sharing it – the goal is to enhance the file availability in the swarm and avoid ending with an incomplete file, in case the *piece* disappears from the P2P swarm (e.g., though peer churn); (iii) *Random First Piece*: the first *piece* to download should be chosen randomly in order to improve the odds of download it quickly – this is important because the peer needs *pieces* to trade and at the beginning it has none; (iv) *Endgame Mode*: the last *piece* of a file should be requested to multiple peers, however once the first chunk of that *piece* arrives, the request to the other peers should be cancelled to avoid duplication and waste of resources – this strategy is used to minimize the probability of waiting too long for the last *piece* of a file, if downloading it only at the end. These strategies however are applicable to regular P2P file distribution applications (delay and jitter tolerant) and not to real-time media distribution. The CS strategy used in the CS module is based on [105][106]. The CS adapts the streams to the network conditions and the upload capacity of the peer. It also implements a failover mechanism that increases delivery of chunks during disconnection events. The CS requests and retrieves the values of the upload capacity, average round-trip delay, jitter and packet loss of the peer from the *Reporter* module on the NMS. The CS also receives topology updates from the TC to update the list of the peer's children in order to determine which streams are going to be forwarded (i.e. according to the properties of the stream (importance, bitrate), the peer capability (upload capacity) and the network conditions (average round-trip delay, jitter, packet loss). The CS is able to communicate with the CS instance running at a parent or child peer (through JSON server) to request chunks of missing streams.

- *P2P Transmitter*: transmits selected IP video streams (or chunks) to the peer's children according to the list of IP addresses provided by the *Chunk Selection.*
- *Depacketization*: extracts and decrypts the video content from the P2P chunks and feeds the resulting data to the *Synchronisation* module. The decryption key is obtained after successful peer authentication when requesting a specific content to the Main Server/Super-peer).
- *Synchronization*: receives one video stream from DVB-T2 broadcasting network and other IP video streams from the P2P delivery network. Because the DVB-T2 frames will be received sooner (less delay) they will be stored in a buffer so they can be synchronized with the IP video streams being received. In fact, both DVB-T2 frames and the IP video stream packets will be stored in buffers (one buffer for the DVB-T2 frames and multiple buffers, for the IP video stream packets – one per each stream being received. Buffering of the frames for synchronization causes a delay between timestamps and the clock reference, and the time of some frames may elapse while they are stored in the buffer. To prevent such losses, the Synchronization module adds an offset to the timestamps of frames at the time of forwarding to the renderers. The amount of this offset depends on the time difference between the arrival time and decoding time of a frame. After these operations the play-out synchronisation is achieved.

3.2.3 Network Monitoring Subsystem

The *Network Monitoring Subsystem* (NMS) collects peer hardware and software characteristics, network traffic statistics (packet loss, average round-trip delay, jitter, available bandwidth) for each received stream and periodically reports these data to its parents (it chooses a different parent in each iteration using a round-robin approach) or to the MTM (in case this is a Top-Level peer). NMS reports can also be triggered by a request from the MTM or when changes in the peer's network conditions cross a specific threshold.

The NMS is also responsible to compute the *peer stability*, a metric that reflects the stability (S) potential of a peer based on previous sessions and computed according to equation (3.2):

$$S = \propto \cdot \sum_{i=1}^{c} t_i - \beta \cdot \left(\sqrt{\frac{\sum_{i=1}^{c} (t_i - \bar{t})^2}{c - 1}} \right) \tag{3.2}$$

Where, α and β are weights (between 0 and 1), c refers to a predefined number of previous connections for which the duration has to be memorized, t_i represents the establishment time (in complete minutes) of each of the c previous connections and $\bar{t}$ is the average value over the c connections.

Before explaining equation (3.2), it is important to bring into the discussion that the system needs to deal with the low completion ratio that occurs when users stop consuming the video content and abandon the P2P swarm just after a few minutes of start watching. Thus, equation (3.2) determines how stable a peer is in the P2P network, based *on a sample* of all its past connections, more specifically in the last c connections. The reason why only a sample of c connections is used instead of all connections is because, for the vast majority of cases, user's behaviour is cyclic and its viewing pattern is repeated every day, this behaviour has been extensively studied in [107] - [110]. By estimating an average of 50 connections per week, we reach the *default* value of c used in equation (3.2), however this value may be changed to adapt the formula to emerging consuming behaviours. Furthermore, since strong behavioural correlation exists amongst users, the formula is valid for the general user population.

The first part of equation (3.2) provides the total connected time for the c connections while the second part provides an *estimation* for the standard deviation of all the user connections, by only using c connections. The α and β factors are used to weight the importance of the total time the peer was connected during the c connections versus the peer erratic behaviour. In case of insufficient information (e.g., peer joining the system for the first time), the NMS uses the predefined default value of *stability* = 100 (assuming α=0.1, β=0.3 and 20 minutes per each connection).

Table 3-3 shows the structure for the periodic NMS report. To save resources and simplify socket management at the receivers, reports are sent to one of the parents, who then collects all the received reports during a time-window and sends all collected reports to its own parent (one by

Table 3-3: Updated structure for the NMS data report

Field	**Description**	**Field Size** (in Bytes)
ID	Message ID (identifies this is a report)	4
PeerID	The unique peer ID as given by the TB	32
LocalIP	The local IP address of the Peer (IPv4/v6)	16
NetMask	The local IP subnet mask	4
DL	The download capability in (Kbps)	4
UL	The upload capability (Kbps)	4
nChildren	The total number of children of this peer	4
CpuID	The CPU brand and model	4
TotalMem	The size of RAM memory in the peer (MB)	4
FreeMem	The size of available memory(MB)	4
ConsMem	The size of consumed memory(MB)	4
OS	The operating system identification	variable
Delay	The average round-trip packet delay (μs)	4
Jitter	The delay jitter (μs)	4
PacketLoss	The packet loss (in %, content specific)	4

one in a persistent TCP connection). This process goes on until the data gets to the highest peers in the P2P tree hierarchy, who then send the bundle of all collected reports to the *NMSCollector* submodule at the MTM. This procedure is illustrated in Figure 3-8.

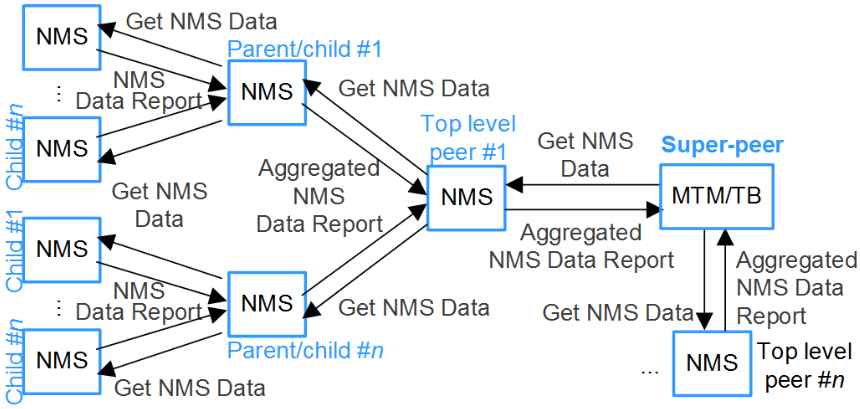

Figure 3-8: Procedure for collecting network monitoring statistics

The NMS is also responsible for detecting, reporting, and possibly solving peer connectivity problems. If a major failure occurs in a stream reception, the NMS first contacts the NMS of the parent responsible to stream that specific content and if it is reachable, it is up to that parent to solve the problem. If the parent wouldn't solve the problem in a timely manner, or cannot be contacted, the TC is notified in order to immediately switch to the backup parent and inform the MTM - this set of procedures is illustrated in Figure 3-9. The amount of time a peer waits for its parent reaction is related to the amount of buffering configured at the synchronization module (see Chapter 3.2.2.2). In IPTV services new peers may have to wait up to 10–15 seconds before they can join a P2P overlay, and it can take another 10–15 seconds to launch the media player and store the video frames in the buffers [108]. A similar value was expected and an upper-bound delay for the initial play-out of 30 seconds was considered. The tree re-insertion delay is expected to not exceed the 50% of the buffering delay in order to avoid interruptions on the content visualisation.

NMS functionalities are implemented through the following submodules, as was depicted in Figure 3-7:

- *PacketCapture*: this submodule passively collects network data such as, connection history (start time, duration) or traffic statistics (packet loss, average round-trip delay, jitter), by exploiting the *libpcap* library [111].
- *LinkTester*: provides link testing functions with the MTM to determine link characteristics such as the peer's downlink and uplink capacity. The downlink indicates how many streams the peer is able to consume (more streams provide higher video quality and faster

viewpoint navigation, see Figure 2-10) and the uplink determines how may children peers can be fed by this peer (with an average bitrate of 4Mbps per stream).

- *Stats*: is responsible for collecting hardware and software information, as depicted in Table 3-2, and for computing the statistics associated with the network data collected by the *PacketCapture* module.
- *Reporter*: formats the information collected by the *Stats* submodule according to a report template. It is also responsible for sending to the TC's *JSONServer* the report to the selected peer parent. If this peer is a parent, this submodule is also responsible for collecting NMS reports from all of its children and to send the report bundle to the selected parent in the hierarchy.

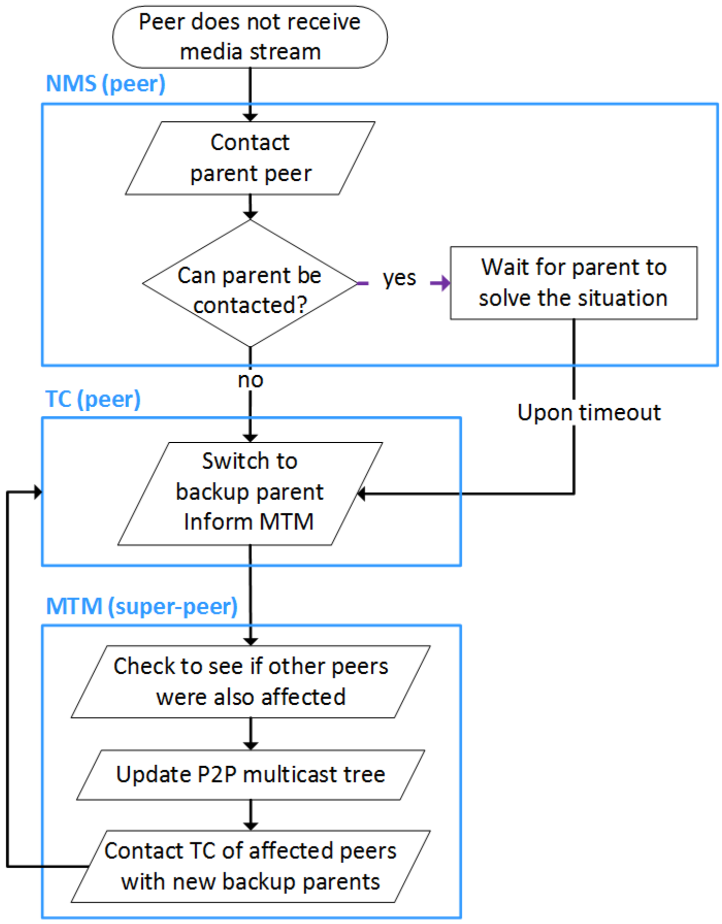

Figure 3-9 - Set of procedures performed by a peer upon detecting major failure in the media stream delivery

3.3 Network Components

Besides Main servers/Super-peers and clients/peers, there are other components/modules that complete the proposed architecture. More specifically, the ones needed to be deployed at the ISP network. These include a QoS controller/broker (and its modules/submodules) that coordinates the QoS operations at the ISP domain and the module that runs at the ISP core and edge routers, responsible for enforcing QoS decisions. Thus the ISP has its own core and access networks, besides having multiple peering connections with other ISPs, as was illustrated in Figure 3-1 and described at the beginning of Chapter 3.

3.3.1 Internet Resource and Admission Control Subsystem (IRACS)

To ensure the IP video streams will get the needed QoS demands, ISP routers would need to be configured to support IP multicasting and QoS overprovisioning [112]. The *Internet Resource and Admission Control Subsystem* (IRACS) is the overall mechanism deployed to coordinate the establishment of IP multicast trees and enforcement of proper QoS policies at the ISP network level. IRACS is compliant with Next Steps in Signalling (NSIS) framework [113] (IEEE RFC4080), hence all IRACS signalling operations use the structure depicted in Figure 3-10.

IRACS operations are based on two main components:

- *Resource and Admission Manager* (RAM): a central control entity - implemented in a server named *Network Control Decision Point* (NCDP) - responsible for defining proper control policies and managing the distribution or access to the underlay network resources.
- *Resource Controller* (RC): enables the ISP to enforce the control decisions based on the instructions received from the RAM. In particular, the RAM includes the following functional modules: a *Resource Reservation Subsystem* (RRS), a *Control Information Base* (CIB), an *Admission and Control Subsystem* (ACS), and one interface for communications with external nodes.

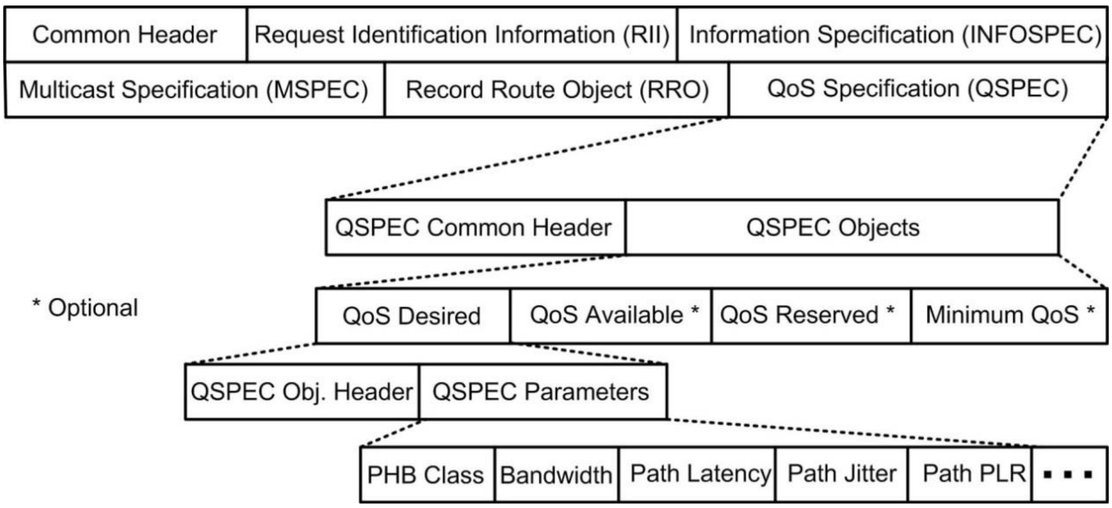

Figure 3-10 – IRACS signalling structure, compliant with NSIS

Figure 3-11 depicts the physical mapping of IRACS components. IP multicast and QoS overprovisioning are controlled by the NCDP, whilst the RAM, a component of NCDP, configures the policies at each router RC who then locally enforces the QoS and multicast routing/mapping operations.

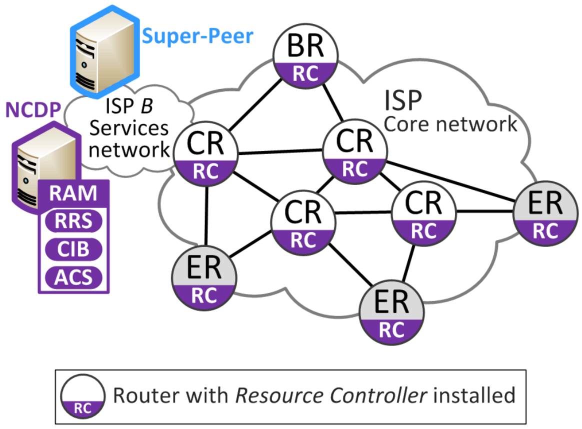

Figure 3-11 - Physical mappings of IRACS components.

The IRACS functional architecture is depicted in Figure 3-12.

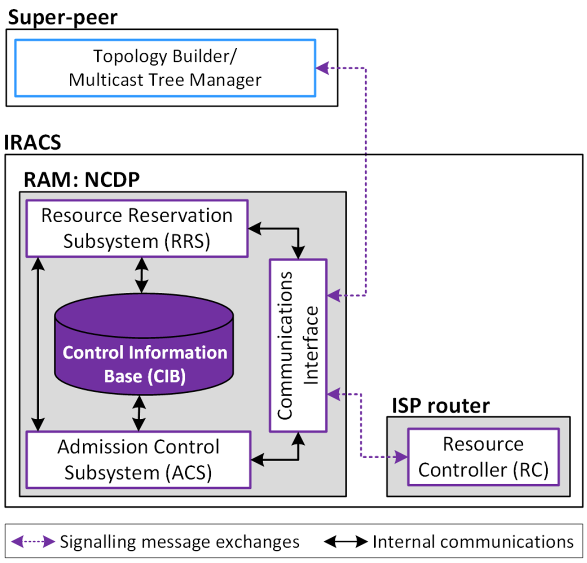

Figure 3-12 - IRACS architecture

3.3.1.1 Resource and Admission Management (RAM) Functions

The RAM is responsible for network access and resource allocation to support IP transport with heterogeneous QoS requirements. In other words, the RAM controls the ISP's network infrastructures by granting or denying access to the related resource consumption in a way to improve the network utilization while guaranteeing differentiated QoS for all admitted sessions based on CoSs. Such differentiation of QoS control is a must for seamless convergence of all types of services over the Internet. Hence, the RAM is responsible for defining appropriate control policies and dictating them for the enforcement on the ISP routers to prevent some Internet applications from starving other applications.

The RAM also performs traffic load balancing to avoid unnecessary congestion occurrence in support for packet delay and jitter control. As such, it facilitates the synchronization operations of multiple views that end-users receive from multiple communication paths. Also, the RAM uses efficient resource over-reservation techniques to improve control scalability, that is, without overwhelming the ISP network with QoS reservation signalling, states and processing overhead while guaranteeing SLAs in terms of bandwidth demands. The RAM achieves these functions by means of a good knowledge of the underlay network topology, the related link resource statistics and control through interactions between various modules such as the RRS, the CIB and the ACS which are respectively described in the next subchapters.

In purely centralized networks, a single RAM will take overall control of the network as in Figure 3-11. In hierarchical scenarios spanning multiple domains, as described in , a RAM will need to be deployed per ISP for scalability reasons. In such cases to ensure correct QoS signalling is performed in inter-domain connections, pre-established SLAs must have been configured.

3.3.1.2 Resource Reservation Subsystem (RRS)

In the RAM architecture, the RRS is exploited to create and manage bandwidth-aware IP multicast trees across the ISP's core network domain in a way to allow for connection between access networks and peers. The precise amount of IP multicast trees and associated QoS state a RRS is able to manage is dependent on the resources (CPU, memory and interface bandwidth) available on the router where the Resource Controller resides (see Section 3.1.5 of Chapter 3). Considering the hardware characteristics of today's core routers and ISP core interconnections, dozens of thousands of multicast trees are possible. Assuming each multiview content needs 14 IP streams and that Base and Enhancement layers share the same application-level multicast tree, as described in Table 3-1, only 7 application-level multicast trees are needed per content, leaving room for plenty of content diversity. The limit will surely be imposed by the bandwidth available on the core network links. The RRS module deploys an aggregate resource over-reservation control scheme [114] to dynamically define appropriate policies for resource sharing among various classes of service (CoSs) on network interfaces upon need. Aggregate over-reservation means that a CoS may be reserved more bandwidth than it currently requires, according to the local control policies. This approach is used to prevent excessive QoS control signalling, states and processing overhead to achieve scalability and to reduce session setup time as well. Moreover, it enables RRS to avoid CoS starvation by means of proper readjustment of reservation parameters dynamically upon need, such that the performance can be achieved without increasing session blocking probability unnecessarily.

3.3.1.3 Control Information Base (CIB)

The RAM uses the CIB to maintain a good knowledge of the underlying network topology and the related resources statuses. It stores the multicast trees created inside the ISP network under the control of the RAM and the IDs of the outgoing interfaces that belong to the trees. Moreover, it maintains a record of the overall capacity of each interface, the amount of bandwidth reserved and used in each CoS on the interface [100]. The CoSs configured on the interface are also maintained along with relevant information about the active sessions inside the network. An active session's information includes, but is not limited to, the bandwidth required by the session, the session ID, the IDs of the flows that may compose the session, the ID of the CoS to which the session belongs, the source ID, the destination ID, the ports IDs, associated multicast tree's ID and the multicast channel.

3.3.1.4 Admission Control Subsystem (ACS)

The ACS enables the RAM to accept or reject service requests, depending on the service requirements in terms of QoS (e.g. bandwidth) and the network resource availability reported by the CIB local database. Therefore, the RAM provides an interface for interactions with the Super-peer (TB and MTM). It is worth mentioning that this interface allows also for receiving service demands from the Main Server/Super-peer or a peer upon need, depending on the overlay specific control mechanism and requirements. Whenever a session is admitted, terminated or the QoS requirements of a running session are readjusted in a CoS on a communication tree, the ACS process updates the resource utilization status (e.g. the bandwidth usage in the concerned CoS) on the related outgoing interfaces in the local CIB. As stated in Chapter 3.3.1, these operations are compliant with the NSIS framework [113].

Considering that resources are over-reserved throughout the network, the ACS is able to admit, terminate or readjust the QoS demands of several sessions without signalling the nodes inside the network as long as the over-reservation is not exhausted on the distribution trees involved in the process, leading to scalable admission and QoS control. When the over-reservation is exhausted on a tree, the ACS triggers the RRS to define new reservation parameters such as the amount of resources to be reserved and the reservation thresholds for the relevant CoSs along the tree as detailed in [114]. After the new reservation parameters have been successfully computed, the RAM conveys them to the nodes on the tree so that the new control policies are enforced on the nodes using the RC modules. This way, the reservation parameters are defined and readjusted dynamically in a way to prevent CoS starvation or unnecessary waste of resources while the QoS control signalling frequency is reduced for scalability.

3.3.1.5 Resource Controller (RC) Functions

The RC module implements the basic control functions and mainly operates at the ISP routers, as illustrated in Figure 3-11. In particular, it deploys elementary transport functions to enable UDP port recognition (routers are permanently listening on a specific UDP port) or *IP Router Alert Option* (RAO) [115] on the ISP routers to properly intercept, interpret and process control messages. It interacts with Resource Management Functions (RMF) [113] to properly configure schedulers [116][117], thus ensuring that each CoS receives the amount of bandwidth allocated to it to provide QoS-aware data transport across the network. For flexibility, it interfaces with legacy protocols (e.g., routing protocols, existing system databases) in order to improve performance. For example, the RC is able to exploit legacy control databases such as, but not limited to, the Management Information Base (MIB), Routing Information Base (RIB), multicast Routing Information Base (MRIB) and Forwarding Information Base (FIB), according to the control instructions received from the RAM.

When deployed at the ISP network border, the RC is enabled to learn inter-domain routing information from the traditional Border Gateway Protocol (BGP) [118] for proper packet delivery between various domains. It also interacts with traffic control and conditioning for traffic shaping and policing according to operator's local control policies to force admitted traffic flows to comply with the SLAs between network users and the providers, which functions are available in most of the *Differentiated Services* (DiffServ) [119] based frameworks. Consequently, the RC is used to enforce multicast trees decisions upon receiving instructions from the RAM. It is also used to allow control messages to collect the IDs of outgoing interfaces and their capacities on trees as being *Record Route Object* (RRO)[120][121]. When instructed by the RAM through a control message, the RC enables ISP routers to record the ID of the previous outgoing interface visited by the message so that asymmetric route issues can be avoided in reverse direction of trees [122].

As a result, the IRACS approach pushes network control complexity to the RAM at the NCDP server and the ISP core routers are left simpler by implementing the RC for scalability reasons.

3.4 Network Operations

When a router boots-up, the RAM, especially the ACS module, gets network topology information by importing such information from existing (resident) link state routing protocols. The ACS then uses an appropriate algorithm (e.g., Dijkstra [89]) to compute all possible edge-to-edge routes inside the core network under its control. As in [91], a combination of the edge-to-edge routes leads to all possible edge-to-edges branched routes. Amongst these computed routes, the ACS selects the best routes that can be used for service delivery. A route can be selected based on, but not limited to, its number of hops and bottleneck outgoing interface capacity. It is worth mentioning that a bottleneck outgoing interface on a route is the outgoing interface which has the smallest capacity on the route. Then, the ACS allocates a unique multicast channel (S, G) for each selected route where S is the IP address of the edge router at which the route originates and G is the IP multicast address assigned to the concerned tree. Besides, the ACS triggers the RRS and the latter defines initial over-reservation parameters to be enforced on interfaces inside the network. After that, the ACS encapsulates this information together with the route record object RRO in a control message and sends the message to the nodes on each route.

As the control message is travelling along a route, every visited router hosting the RC module intercepts the message and configures its local multicast routing table as well as the initial over-reservation parameters destined to its interfaces accordingly. Also, the control message is forced to follow the desired route by means of the route record RRO which enables source routing such that multiple QoS-aware multicast trees are thus initialized for use inside the network. In a pure centralized scenario, a single RAM maintains knowledge on the entire network and related trees. In a hierarchical control scenario, an end-to-end route may be a concatenation of trees from each of the domains that lie on the route. Hence, every RAM in a domain properly maps traffic flows to its local trees according to its local QoS control model, independently of the other RAMs and end-to-end control is assured in a scalable manner across heterogeneous network environment.

As the network is operating, incoming authorized peers' connection requests to a super-peer must be forwarded to the RAM for the purpose of resource and admission control. To ease the understanding of the interactions among a joining peer, the Super-peer, the RAM and the Main Server, Figure 3-13 is used to depict a use case for the IRACS session setup.

These operations and the message flows are based on the network topology introduced in Figure 3-1. In particular, a peer wants to enjoy a 3D content service. Hence, it first interacts with the Main Server, which performs authentication, authorization and accounting operations. During this control phase, the peer and the server agree on the required views QoS requirements (e.g., bandwidth) as well as the related traffic characteristics. In case authentication operations are successful, the server redirects the peer to the Super-peer by providing the address of the latter. This is important to prevent unauthorised peers from affecting the media delivery performance. Hence, the joining peer issues a connection request to the Super-peer. The request carries the desired session's QoS and the involved traffic characteristics (e.g., codec and peak rate, etc.). Upon receiving the request, the Super-peer contacts the RAM to request the underlay network resource capabilities in order to establish QoS-aware overlay connectivity for the networked peers.

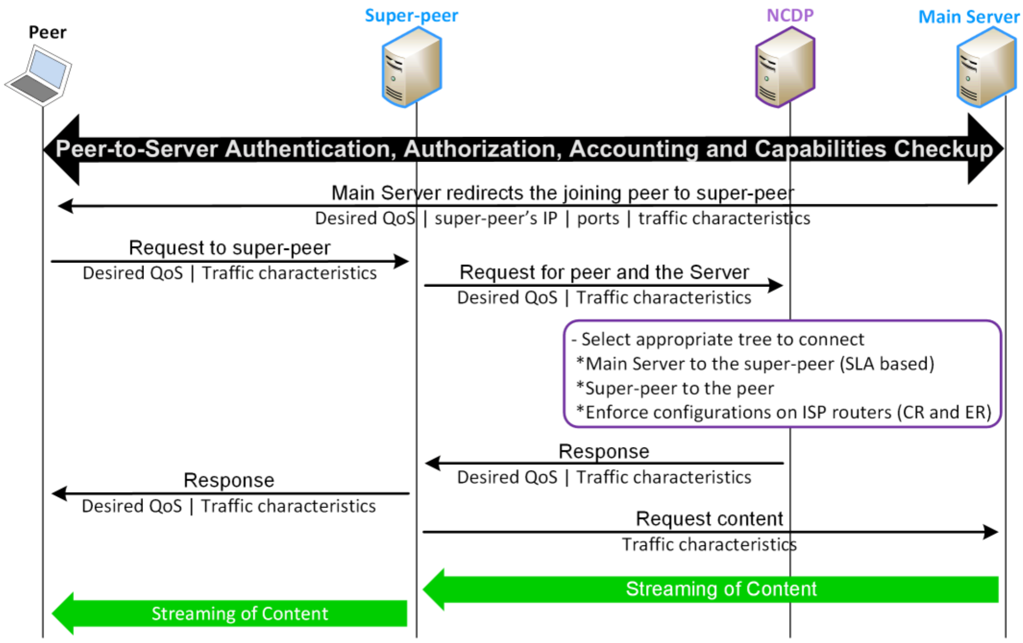

Figure 3-13 - IRACS related signalling (simplified) for peer insertion.

The request to the RAM includes the IP address of the joining peer and that of the main server as well as the QoS requirements and the traffic characteristics. Thus, the NCDP/RAM, selects the appropriate multicast trees (pre-established as in previous subchapter) to connect the Super-peer to the Main Server and the appropriate multicast trees to connect the Super-peer to the joining peer. In case the trees selection succeeds, the NCDP instructs the ERs that lie on the selected trees so that the latter can properly enforce the selected trees and the requested QoS. Hence, the ERs bind the incoming contents parameters (e.g., source' and destination' IP addresses, ports and transport protocol) with the selected tree's multicast channel. This is to assure that, incoming media packets will be correctly encapsulated at ingress ER to follow the trees allocated, so they enjoy the QoS reserved for them.

The same way, the packets will be decapsulated at the relevant egress ER for delivery to the peer/consumer/end-user. Besides, the NCDP provides the underlay resource availability information to the TB/MTM, as shown in Figure 3-12. Based on this feedback and the peers' resource conditions the Super-peer is enabled to define or update the appropriate P2P application-level multicast trees with less signalling exchanges, which is crucial for scalability.

In this sense, the construction of the multicast distribution trees is performed by two different system components, the IRACS at the ISP core network level, and the TB at the access level.

Figure 3-14 illustrates this concept and Figure 3-15 presents a step-by-step description of the events in a sample scenario.

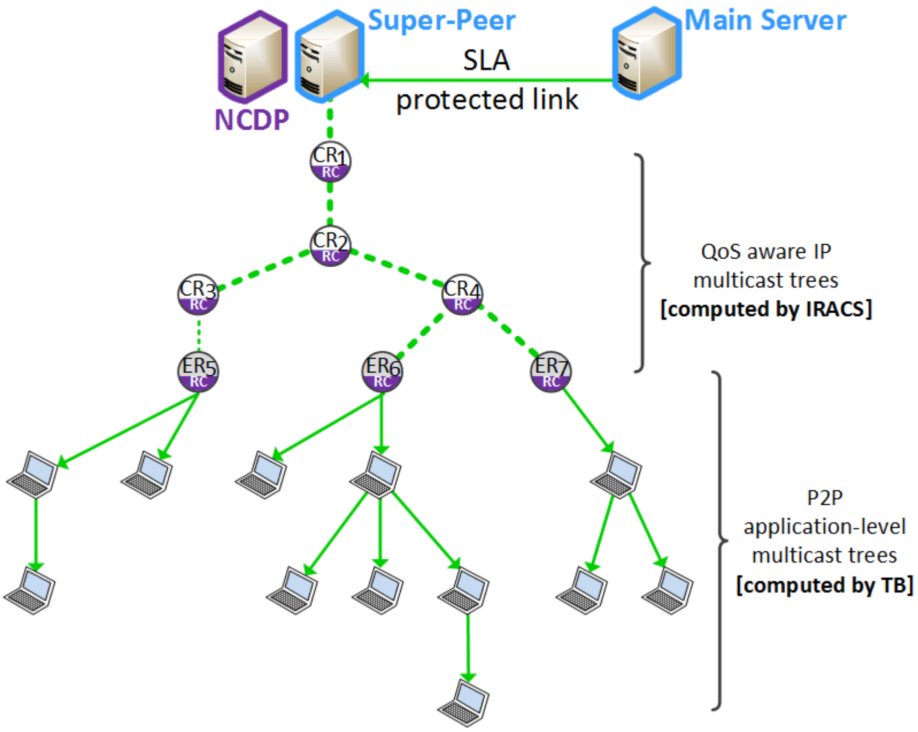

Figure 3-14: Delegation of the multicast tree construction mechanisms in our proposed framework. Bottom part depicts the constructed tree hierarchy at the access network.

Using Figure 3-15 as a reference:

- When peer 1 connects, IRACS computes the tree from the Super-peer to the Edge Router serving the peer. TB computes the tree from the ER5 to the peer;
- A new peer arrives (peer 2) and no changes are needed in the IP multicast tree at the ISP core level. TB computes peer insertion on access-level existing tree – peer 2 becomes child of peer 1;
- Additional peers (peers 3 and 4) connect from a different ER. IRACS adds new branches to the existing IP multicast tree at ISP core level and the TB computes new P2P tree at access level;
- Finally, new peers arrive at the same access network (peers 5, 6, 7). The only changes are at the access-level P2P tree - it now consists of one parent (peer 3), one parent/child (peer 4) and three children (peers 5, 6, 7).

Figure 3-16 provides an overall view of the signalling involved when a new peer joins the system.

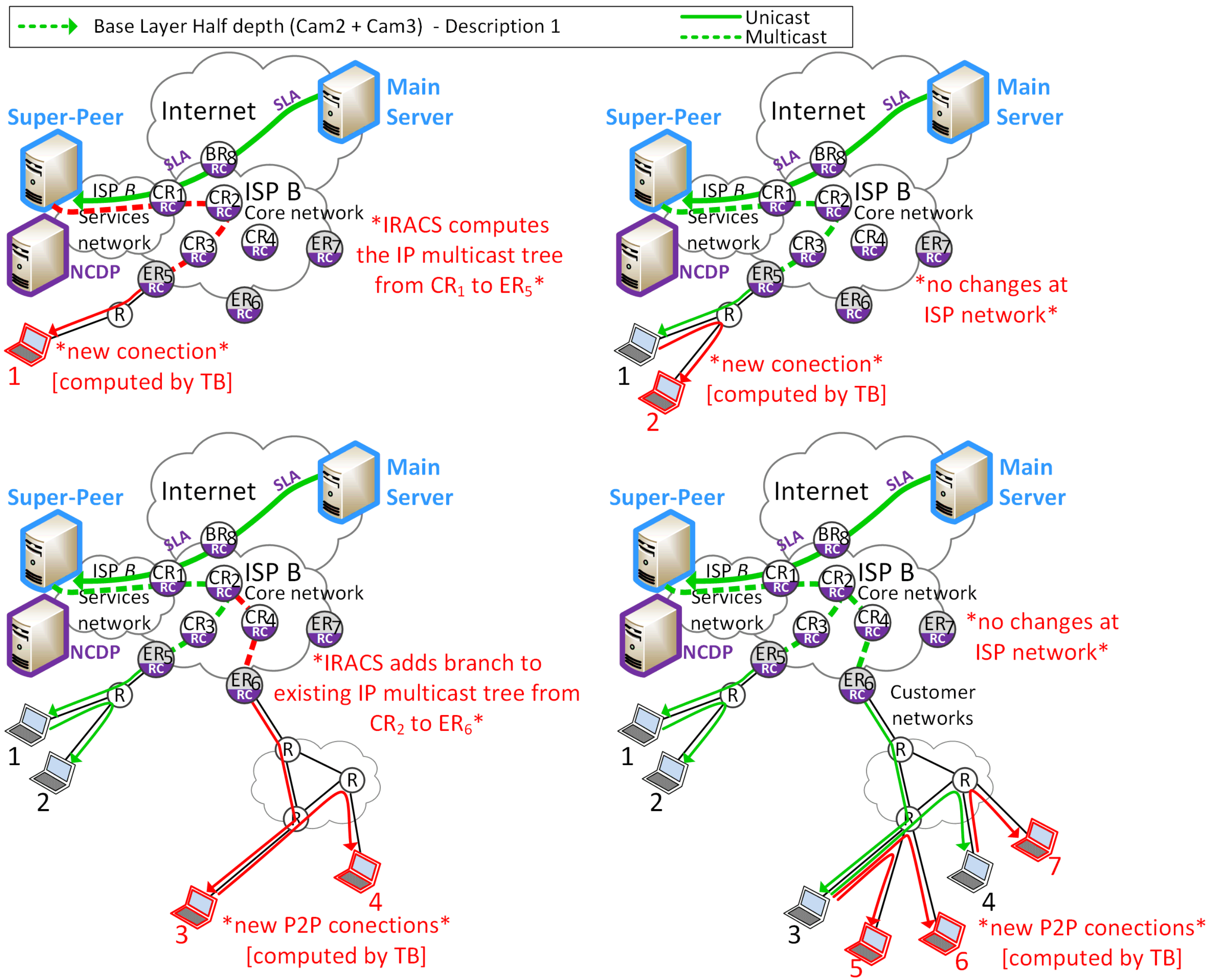

Figure 3-15: Step-by-step procedures on the multicast tree construction.

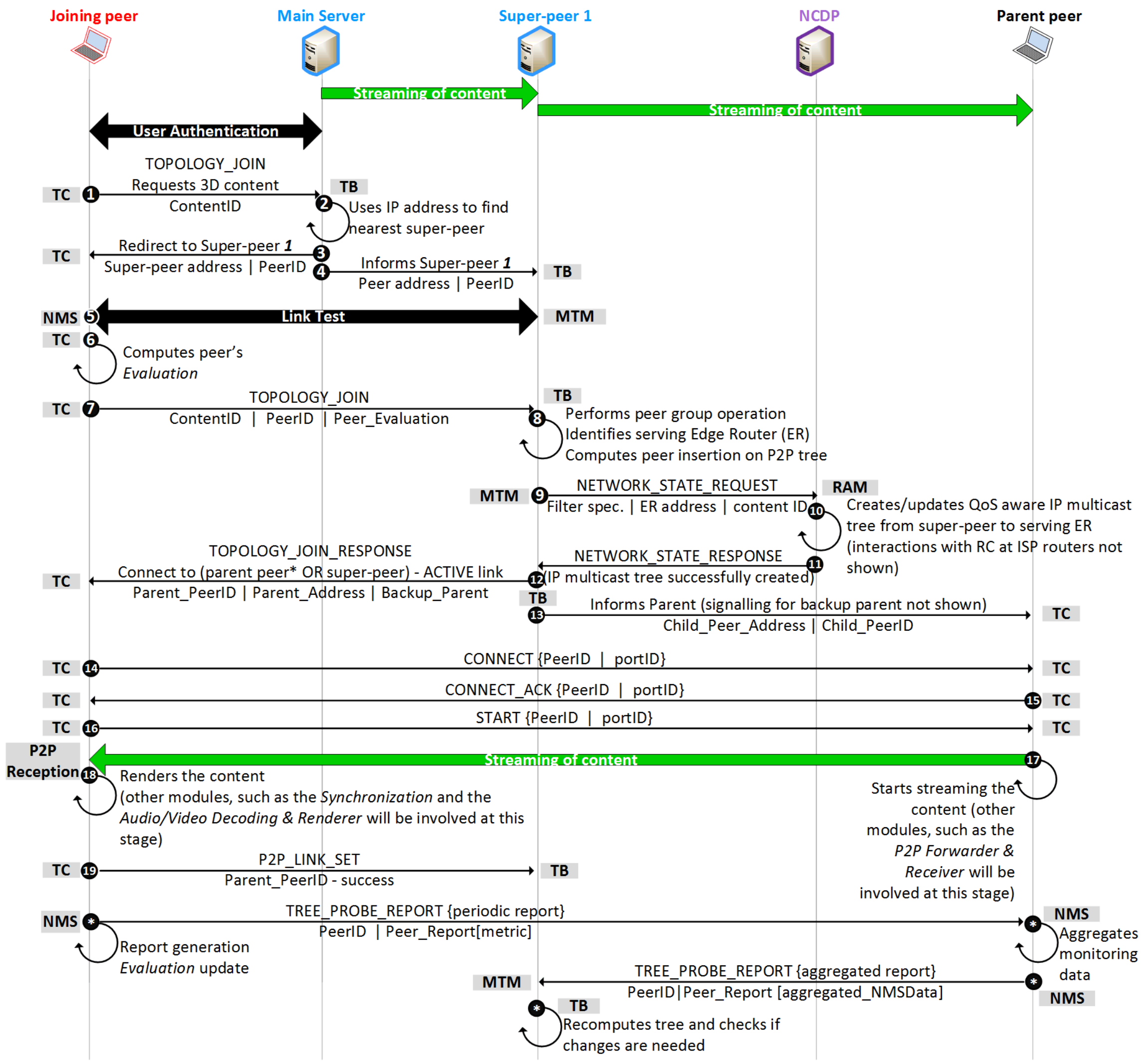

Figure 3-16: Signalling amongst the system components when a new peer joins the system - the scenario considers the construction of one tree only and assumes a peer (on the right) is already receiving content.

3.5 QoS Assurance across multiple network domains

In order to facilitate the understanding of the IRACS operations in support for the distribution of the IP video streams in a multiple ISP network domain scenario, the description is based on the network topology in Figure 3-17 and also on the message sequence chart in Figure 3-18.

In particular, Figure 3-17 encompasses three ISPs' networks (ISP A, ISP B and ISP C). For simplicity, we assume that the Main Server/Super-peer is hosted in ISP A's infrastructure which is connected to ISP B's network via the Border Routers - BRs - (BR1 in ISP A and BR2 in ISP B). The ERs bridge between an access and a core networks while a BR is used to connect two core networks. Besides, the ISP B's network is composed of two access networks (Access B1 and

Access B2) and one services/control network, which are connected through a common core network. Likewise, the ISP C's network encompasses two access networks (Access C1 and Access C2) and one services/control network. Also, a Super-peer is present in each ISP's network so as to allow for a proper coordination of the overlay media streaming among the peers.

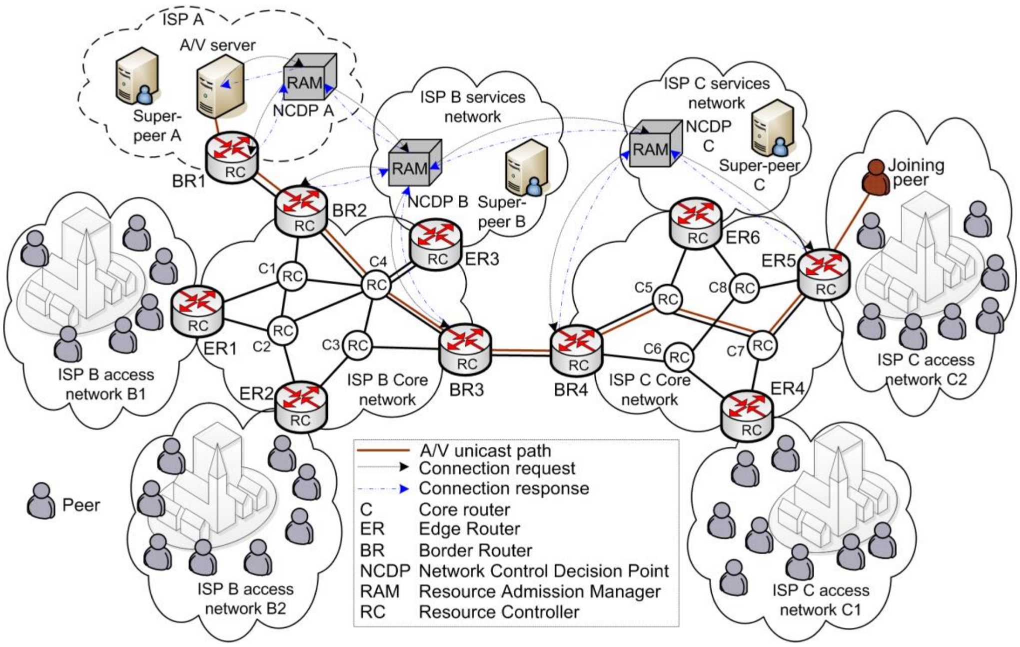

Figure 3-17 - QoS support for IP video streams across multiple domains

Hence, let's suppose that a joining peer, with brown/shaded colour in the access network requests content to the Main Server/Super peer. A connection request specifies the desired QoS parameters in terms of bandwidth and buffer size together with the related traffic characteristics (e.g., flow IDs, source and destination IP addresses and ports, etc.). This will be performed in compliance with NSIS protocol, as described in the beginning of Chapter 3.3.1. Thus, the way that QoS-aware path is set up, is detailed in the rest of this subchapter using the message sequence chart in Figure 3-18.

First, the joining peer in the access network C2 (see Figure 3-17) issues connection request to the Main Server/Super-peer, specifying the desired QoS and traffic characteristics as explained earlier. After that, the Main Server/Super-peer contacts RAM A with the IP address of the joining peer along with the desired QoS and traffic characteristics, and requests a bandwidth-aware path to the joining peer. This means that a QoS-enabled path must be set up all over the way between the Main Server/Super-peer and the joining peer.

Upon receiving the request, RAM A checks its candidate unbranched trees (created at system initialization phase for unicast purposes) that connects the ISP A's network to the neighbouring ISP B's network on the route towards the joining peer (see BR1 to BR2 in Figure 3-17). A Service Level Agreement (SLA) should have been pre-established between the involved ISPs while the proper BRs (ingresses and egresses in a domain) are obtained based on the inter-domain routing information populated by BGP (Border Gateway Protocol) [118]. As such, whenever a candidate unbranched tree presents sufficient over-reservation, the RAM A books the best tree amongst them and forwards the request to its neighbouring domain towards the joining peer, that is, to the RAM B without QoS reservation signalling. However, if the over-reservation is insufficient, the RAM A invokes the *Resource Reservation Subsystem* (RRS) to decide new reservation parameters policies based on the resource information in its local *Control Information Base* (CIB).

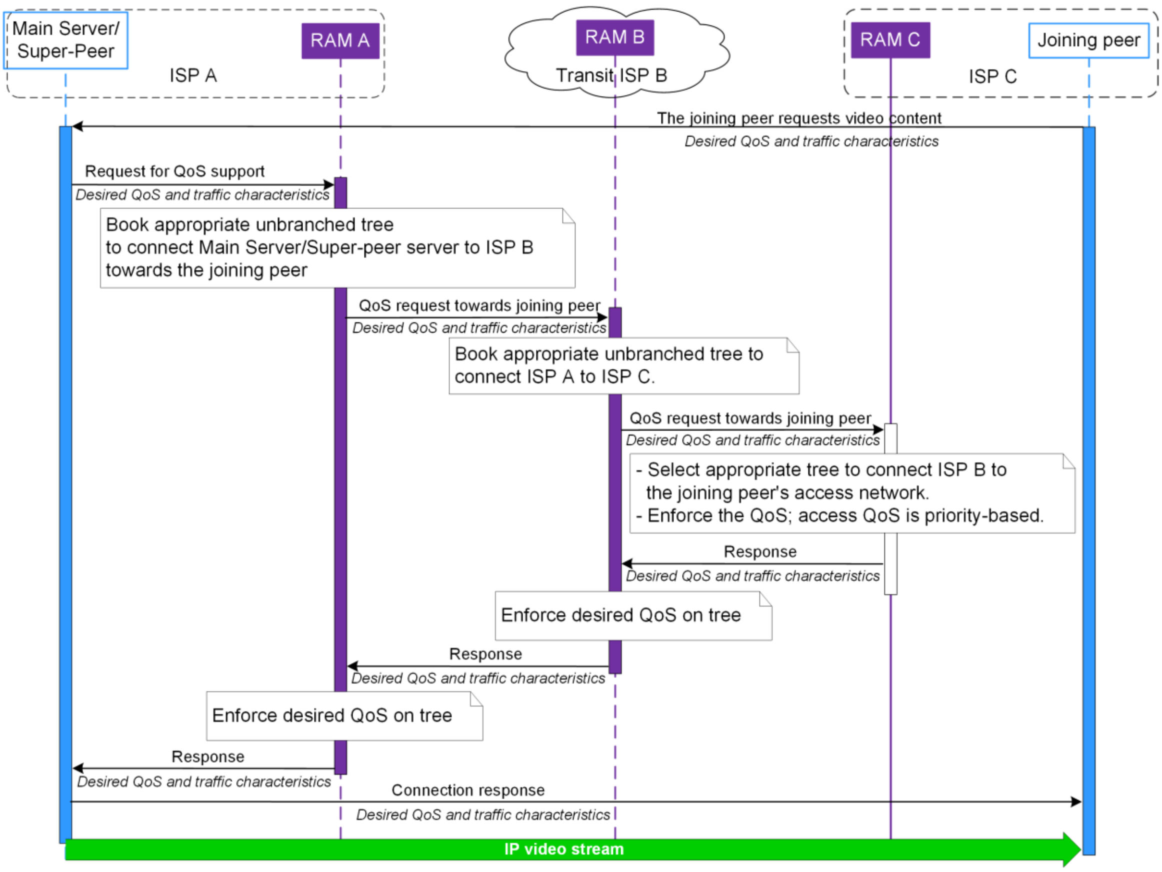

Figure 3-18 - A use case for the QoS signalling messages sequence chart in a cross domain set-up

In case the resource readjustment is successful, the tree is booked and the message is automatically forwarded to the next RAM B without QoS reservation signalling. This way, RAM A avoids per-flow QoS reservation signalling on trees so as to scale. In case the currently unused network resources are not enough to connect the joining peer with acceptable QoS, RAM A denies the request and sends a notification back to the Main Server/Super-peer, or it maps the request to a lower QoS Class of Service (CoS) according to pre-defined SLAs between the provider and the customer. This mechanism is repeated at each RAM visited on the route (e.g., RAM B) till the message reaches the RAM C in the ISP domain where resides the joining peer.

Hence, RAM C selects its best unbranched tree to extend the path being created across the ISP C's core domain to the edge router (ER5) of the joining peer's access network. Then, it enforces the QoS in its local database and instructs the related ingress and egress routers (BR4 and ER5) to configure the sessions-to-trees binding so that media packets/chunks can be correctly encapsulated at ingress and decapsulated at egress points. Afterwards, it sends a response message in the reverse route back to the Main Server/Super-peer. It is important to recall that, at the edge node (ER5), a *Policy and Charging Rules Function* (PCRF) is deployed to enforce priority of the traffic flows over the background services in the access domain of the joining peer.

As the response message is travelling back to the Main Server/Super-peer, the BR3 intercepts it and forwards it to RAM B. Hence, in case RAM B has sufficient available over-reservation on the booked tree, it simply confirms the desired QoS enforcement in its local database and instructs the related ingress and egress routers (BR2 and BR3) to configure the sessions-to-trees binding so that media packets/chunks can be correctly encapsulated at ingress and decapsulated at egress points. In this case, RAM B forwards the message upstream to RAM A without issuing QoS reservation signalling messages. Otherwise, the RAM B must first signal the Resource Controllers (RCs) implemented in the routers on the selected tree in order to readjust the reservations according to the local control policies before forwarding the message towards the A/V server. In the same way, BR1 will intercept the message and forward it to the RAM A. Then, RAM A will enforce the QoS on its tree and send the response message to the A/V server. As a result, end-to-end QoS-enabled unicast path is set up between the A/V server and the joining peer. Finally, the server informs the joining peer and starts streaming the media.

IRACS effectively deploys a fast and scalable resource and admission control mechanism, considerably lowering signalling events using a per-class over-provisioning approach thus preventing per-flow QoS reservation signalling messages. Moreover, it is aware of network link resources in real-time and supports for service differentiation and network convergence by guaranteeing that each admitted traffic flow receives the contracted QoS.

3.6 Conclusion

This chapter provided the details on the core novelties claimed by this thesis. These novelties included a new vision for PA-CDNs applicable to the distribution of multiview real-time media that merges the concepts of client-server and P2P overlay networks in a synergy between IP multicast and application-level multicast, retaining the advantages of both paradigms while still improving bandwidth efficiency and considerably lowering packet loss, delay and jitter. A summary of the research challenges and specific contributions to address them in provided below in Table 3-3.

Table 3-4: A summary of the main research challenges and contributions

Research challenge	Proposed solution (Key achievement)
Distribute bandwidth-hungry real-time media content to a large population in a scalable manner (up to 70 Mbps per consumer)	Use a multihoming distribution network composed of DVB-T2 and IP network. Distribution in the IP network uses both client-server and P2P paradigms. The client-server is modified to client-ER using pure IP multicast. The ER replicates the IP multicast traffic to multiple P2P Top-level peers who then distribute the traffic to other peers in the same access network. The P2P network is hence geographically contained within the boundaries of a particular ER.
Dynamically adapt video quality according to consumer network conditions	Video content is split in multiple streams - Base layer and enhancement layer streams - , obtained by using the SVC extension of H.264/AVC encoding standard. Each base layer and its correspondent enhancement layer are sent over the same multicast tree and thus different views are sent over different multicast trees. Therefore a peer only receives a specific number of streams, which can be limited by its network conditions or to a subscription fee.
Lower Service Provider costs in infrastructure and its management	P2P networking is proposed to assist the distribution of content, offloading traffic from the ISP core network. Besides achieving considerable bandwidth savings for the ISP, this means less investment on equipment and management.
Achieve minimum impact on existing traffic load at the service provider core network	IP multicast is proposed to be used only at the ISP core network. This approach lowers any concerns on the usage of multicast technology, providing huge bandwidth savings and low impact on existing traffic patterns.
Ensure the distribution of multiview content achieves the minimum inter-ISP traffic as possible	P2P traffic is contained in to a single ER, therefore no P2P traffic exists outside its access networks. The exception occurs when peers are located on non-compliant ISPs, as shown in Figure 3-1. Peers located in such ISPs may be allowed to access if the CDN policy allows it.
Provide almost real-time full awareness of resource availability at the ISP core links	The IRACS RC (see Chapter 3.3.1.5) module, running in the ISP core and edge routers has real-time knowledge of the resources for all the router's interfaces. This information is periodically updated to the CIB module (see Chapter 3.3.1.3) who gathers the knowledge of the underlying network topology.
Perform QoS resource reservation at the ISP core network with minimum delay (possibly on-the-fly), with the least waste of resources and in an adaptive manner	The IRACS is proposed as a novel over-provisioning QoS mechanism, that performs class based QoS reservations (contrary to PHB used in DiffServ or per-flow reservations used by IntServ). When compared to other existing over-provisioning approaches that mostly waste bandwidth and increase session blocking probability unnecessarily, IRACS allows minimizing undesired QoS control states, processing and signalling overheads by decentralizing operations at distributed entities throughout the ISP network.
Achieve QoS with minimum control signalling events	

Achieve minimum end-to-end delay, jitter and packet loss (related to the consumption of the multiview content) even considering the dynamics of network behaviour.	By containing the P2P network to small geographical areas, changes (such as peer churning) are self-contained to these areas, furthermore, since parents and peers are closer to each other and share the same access network, the propagation delay and jitter are shorter, and there is less probability of packet loss. In addition the usage of IP multicast drastically reduces the bandwidth needed to distribute the content to the multiple ERs which further improves delay. Finally, by offloading traffic to the P2P network the Super-peer has less load and traffic queues are shorter.
Re-use existing technologies and standards as much as possible for easiness of adoption	The proposed architecture is built upon the existing concept of PA-CDNs. IP multicast is a stable and very well-known standardized technology from IETF [5]. The same is applicable to IRACs who is built in the foundations of NSIS [112] and DiffServ [81].

New concepts have been introduced, such as the *Topology Builder*, the *Multicast Tree Manager*, the *Topology Controller*, the *Network Management Subsystem*, the *Internet Resource and Admission Control*, the IP-multicast–to-IP-unicast replication function of the ISP's Edge Routers, amongst others. The concept of splitting the distribution of content in pure multicast trees at the ISP core network and geographical self-contained application-level multicast trees at the access network level, together with a new algorithm for P2P overlay construction and maintenance synchronized with a fast QoS approach with low signalling overhead, are the true novelties presented.

This thesis claims that the acceptance of the assumptions made in this chapter by an ISP would greatly improve its network efficiency and considerably facilitate its management in the support for multiview real-time media.

Chapter 4

4 Component Validation

Chapter Outline: component validation is a logical step when designing, developing and integrating a new system. It allows verifying the correct operation of a component or its submodules before proceeding to full system integration. Chapter 4.1 describes the testbed and presents the validation results for Server and Peer components and modules under specific real tests; Chapter 4.2 provides real performance results for a subset of inter-ISP connections in the current Internet infrastructure; Chapter 4.3 describes the simulation scenario and performance results for the IRACS network component. Finally, Chapter 4.4 highlights the conclusions from the obtained results.

4.1 P2P networking performance

For the P2P network performance the following components were integrated and tested on a real testbed:

- *P2P Packetisation/Transmitter* (Main Server): multi-threaded UDP sender application, running at the Server/Super-peer, responsible for the packetisation, transmission and encryption of chunks for the different IP video streams. It receives, at runtime, the IP address list of the top-level parents provided by the *Topology Builder*.
- Topology Builder (Main Server): is the entity that creates the multiple application-level multicast trees at the access network level and inserts joining peers on specific positions, according to their rank, on these trees. The highest ranking peers, called Top-level peers, are at the root of each tree and receive the content directly from the *P2P Transmitter*. Lower-level peers receive the content directly from the Top-level peers or from other higher ranking peers.
- *Multicast Tree Manager* (Main Server): is the entity that collects/aggregates network monitoring data (percentage of packet loss, average round-trip delay, jitter and available bandwidth), from all connected peers, providing the *Topology Builder* (TB) with updated peer's network conditions. It also allows peers to perform bandwidth tests with the Main Server/Super-peer.

- *Topology Controller* (peer): performs the initial contact with the Server for user authentication and redirection to the nearest super-peer. It computes the peer evaluation and performs P2P tree operations as commanded by the TB (parent, parent/child or child).
- *Network Monitoring Subsystem* (Peer): this module collects peer hardware and network traffic statistics (packet loss, average round-trip delay, jitter and available bandwidth) and, periodically reports the collected data towards the Multicast Tree Manager (MTM). This module is also responsible to compute the peer stability, a metric that reflects the stability potential of this peer based on previous sessions.
- *P2P Receiver & Forwarder* (Peer): multithreaded UDP receiver application, running at the peer, responsible for the reception/chunk selection, depacketization, decryption of each received chunk. It also forwards each received chunk to the children peers. After decrypted the chunks are also forwarded to the local Synchronization module.

The evaluation tests performed are related, with the peer joining and leaving the P2P overlay.

4.1.1 Test environment

The test bed simulates a single ISP core network and access network, with one Super-peer and seven peers, running the software modules as described in the previous subchapter. There are a total of 4 IP networks, with one 1Gbps layer 2 switch in each network. The overall testbed topology is depicted in Figure 4-1, lines are representatives of IP video stream flows. In the presented scenario it was considered the construction of a single multicast application-level tree with a single Top-level parent.

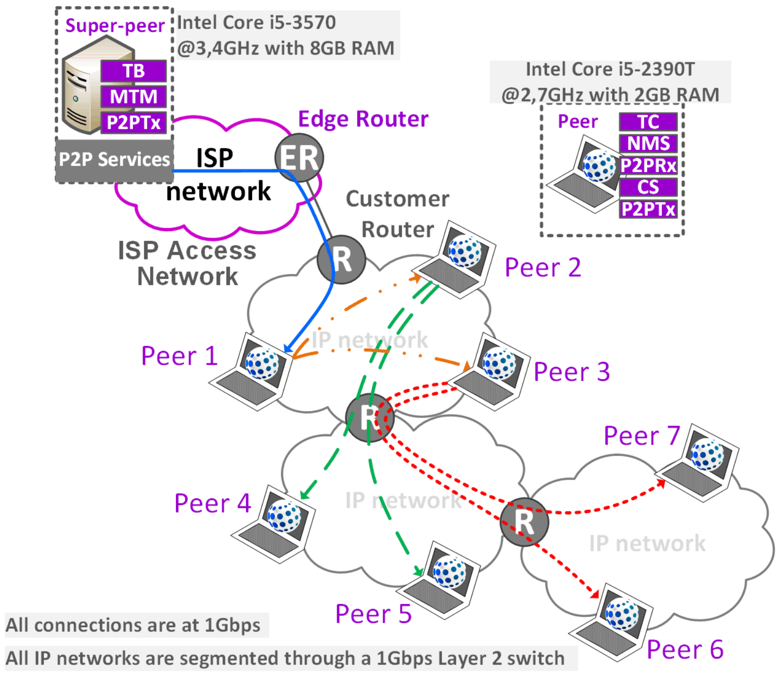

Figure 4-1: P2P network test bed topology overall look

During the tests, peers join the service and the Super-peer updates the application-level multicast trees (all share the same topology) accordingly. To measure the P2P network performance, the tree depth consists of three levels. Top-level *Peer 1* starts to receive the media and forwards it to its children (peers 2 and 3), and the process continues to the remaining peers. Table 4-1 depicts the IP video streams used, in a total of approximately 28Mbps. Following this set up, the test bed allowed for the measurement of the following performance metrics:

- CPU consumption by the different modules – relevant as a baseline, to estimate minimum system requirements for correct system operation (Server and Peer) and also to gather feedback for the Peer Evaluation metric, equation (3.1), as described in Chapter 3.2.2.1;
- Memory footprint at the TB – enables to estimate how the algorithm used for the creation of the multiple P2P application-level multicast trees scales in terms of memory consumption when the number of peers grows (up to 10^5 peers);
- Tree computation time at the TB – same as the previous, but applicable to CPU consumption;
- Consumed bandwidth by the NMS periodic reporting system – provides insight on how much overhead is consumed by the NMS component of a parent peer as the number of children peers increase and the reporting period decreases (is faster).

Table 4-1: Streams used for P2P networking performance experiment

PID	Stream Name
1001	Base Layer Half colour (Camera 2 + Camera 3) – Description 1
1002	Base Layer Half colour (Camera 2 + Camera 3) – Description 2
1003	Q. Enhancement Layer Half colour (Camera 2 + Camera 3) – Description 2
1004	Base Layer Half depth (Camera 2 + Camera 3) – Description 1
1005	Q. Enhancement Layer Half depth (Camera 2 + Camera 3) – Description 1
1006	Base Layer Half depth (Camera 2 + Camera 3) – Description 2
1007	Q. Enhancement Layer Half depth (Camera 2 + Camera 3) – Description 2

4.1.2 Results

4.1.2.1 CPU consumption by the different modules

The results obtained, reflect an average of 20 rounds. Each round took a duration of 15 minutes (2 minutes per each increase in the number of streams, plus 1 minute pause between rounds). Figure 4-2 depicts the results for the P2P modules that contribute the most for the CPU consumption (server and peer).

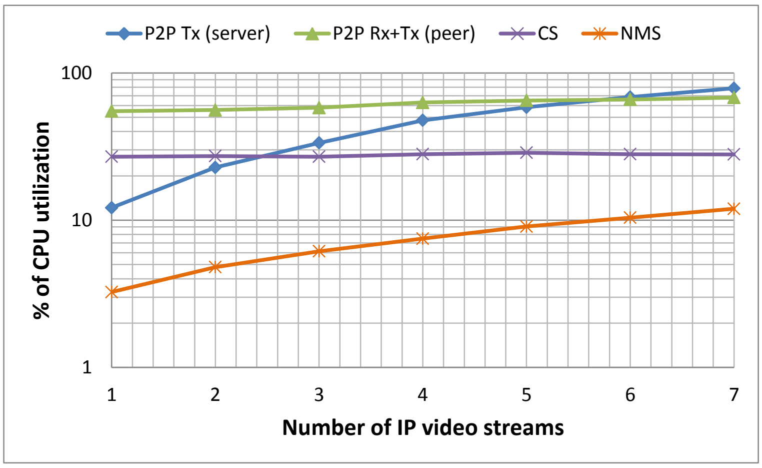

Figure 4-2: P2P modules' CPU utilization versus number of IP streams transmitted

From the analysis of Figure 4-2, the following results can be drawn:

- Small fluctuations may exist since neither the size of chunks or the bitrate is constant throughout the entire video sample being transmitted.
- The CPU consumed by the P2P Transmitter module running at the Server/Super-peer (P2P-Tx of the server) has a direct dependency on the number of IP streams it needs to generate from the source MPEG-2 transport stream containing the full 3D media multiplex. Hence, associated file operations, packetisation (chunk stream generation) and encryption operations consume the most CPU. It was observed that stream replication had a minimal impact on the CPU consumption for each new top-level peer added (less than 1%).
- The CPU consumption of the P2P Receiver & Forwarder, running at the peer (P2P Rx+Tx of the peer) is relatively high. Upon reception of each stream from a different port, It performs depacketisation, decryption; checks integrity, and forwards each chunk to both the Synchronization module (for media players) and to the network, using the instructions received from the CS module.
- The CS module's CPU overhead, due to its nature, is basically independent from the number of streams flowing through a peer. It shows a steady behaviour.
- The NMS module monitors every stream received and for each it calculates its bandwidth, jitter and packet-loss. The CPU resources consumed are considerably low, linearly depending on the number of streams being received. This low value was only possible to achieve by running the *libpcap* [111] at the operating system Kernel level with a single instance responsible for all the traffic related monitoring operations.
- The TC and the MTM bring a negligible load on the CPU (less than 0.1%), and hence are not shown in Figure 4-2.

4.1.2.2 Memory footprint at the Topology Builder

For this evaluation, the peers in the topology run a script that allows them to send multiple joining requests (up to a total maximum of 100K requests) to the Super-peer, actually emulating thousands of peers. Figure 4-3 depicts the memory footprint for the Super-peer in light and heavy conditions. The results were computed for P2P tree construction and maintenance operations only.

At its maximum, the total memory consumed at the Super-peer was approximately 48 Mbytes, which indicates a highly scalable algorithm. Note that due to the (geographical) distributed nature of the system, Super-peers are expected to support far lesser number of peers and only in very rare occasions a Super-peer would need to support such a high number of peers. ISPs in such conditions may split clients by zone, or perform load-balancing techniques amongst two or more Super-peers.

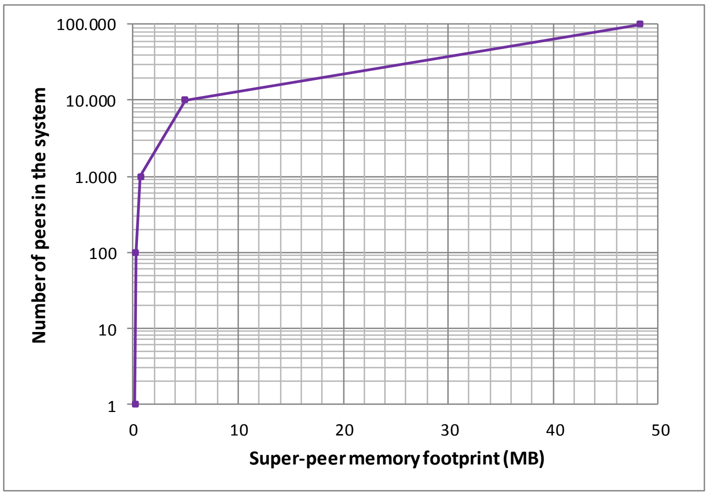

Figure 4-3: Memory footprint for P2P tree construction and maintenance operations at the Super-peer versus number of peers in the system.

4.1.2.3 Computation time for the construction and maintenance of P2P application-level multicast trees

The evaluation of this metric was crucial to understand how the proposed system would perform under constant changes in the access network. Using the same set up as before, Figure 4-4 shows the evaluation of this performance indicator when computing a single tree versus the number of peers in the system.

In the most demanding case, where all 7 trees have 100k peers, the computation would take approximately 3.5 seconds. For more general cases, the computation of all P2P multicast trees will be under a second.

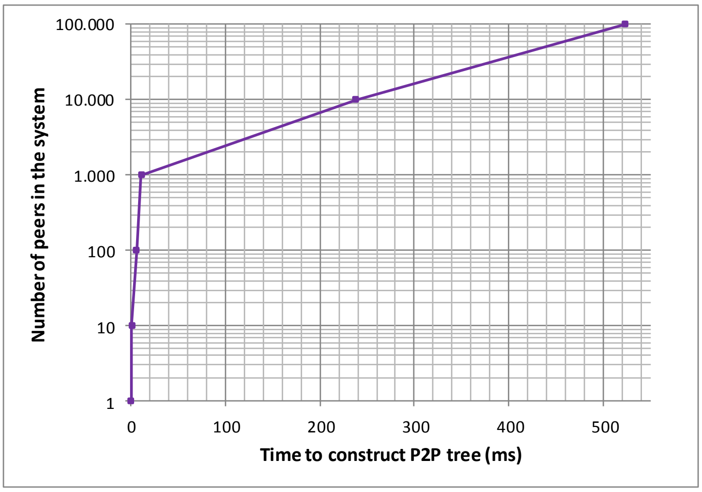

Figure 4-4: Time to compute the P2P multicast tree versus the number of peers in the system

4.1.2.4 NMS bandwidth consumption

As explained in Chapter 3.2.3, for tree maintenance purposes, each peer periodically sends a network monitoring data report to the MTM.

In our test bed environment, a single report has a size of 387 Bytes (58 bytes from Ethernet, IP and TCP overheads, and the remaining 329 bytes from the *JavaScript Object Notation* (JSON) plus the report structure).

In an access network that has a capacity of 1Gbps, considering each media stream to be approximately 4 Mbps and that at least a minimum of 3 streams are needed to be received by each peer, a worst case scenario was considered, where one single parent has to support 75 children (leaving 10% of the available bandwidth for other networking operations). In such a scenario, the average bandwidth consumed by the NMS depends on the periodicity of the report and the number of children being supported.

Through analytical calculation it can be seen, from the plot in Figure 4-5, the NMS module operating in a peer with 75 children consumes an average of 45Kbps bandwidth if the report is to be sent every 5 seconds. Even though this is the worst case scenario, it still corresponds to an acceptable bitrate.

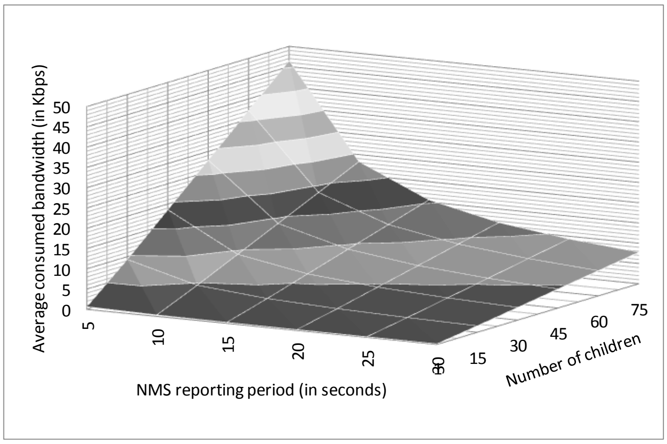

Figure 4-5: Average consumed bandwidth by a parent peer when sending the aggregated NMS reports towards the MTM on the Super-peer

4.2 Inter-ISP tests for the service suitability in the existing Internet infrastructure

4.2.1 Test bed description

This Chapter describes the tests used to measure real-life performance of a subset of inter-ISP connections in the current Internet infrastructure. By default, the core and access networks of the involved ISPs are agnostic to the IP video service's traffic, treating it with a best effort policy.

The performance evaluation is performed using the *iperf* [131] tool to measure jitter and packet loss in the set up scenarios presented in Table 4-2.

Table 4-2: Inter-ISP scenarios

	Country				
	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
Super-peer	Germany	UK	Greece	Portugal	Portugal
Peer	Portugal	Portugal	Portugal	Spain	Turkey

4.2.2 Test environment

The inter-ISP network topology applicable for this test is depicted in Figure 4-6. For each scenario, the *iperf* tool was configured to send multiple IP streams with the following configuration:

- Transport protocol: UDP;
- Size of the protocol data unit: 1380 Bytes (average size as measured in test-bed);

- Sending bitrate: constant bitrate of 3.5 Mbps. (average bitrate as measured in the test-bed).

The first three scenarios present no bandwidth link limitation and the measurements intend to evaluate the performance of the inter-ISP connection for best-effort service. Scenarios 4 and 5 present bandwidth link limitation and the goal is measuring how much it could affect the visualization of content in today's Internet.

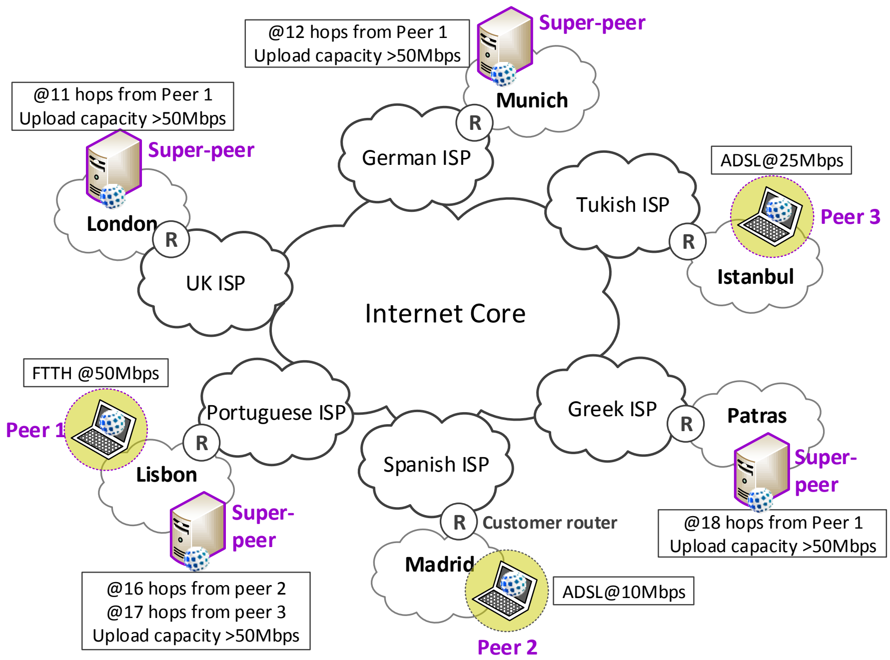

Figure 4-6: Inter-ISP topology set up

All tests have been performed on working days during peak hours, between 10:00h and 17:30h (CET), hence they reflect the worst case scenario for such transmissions. The only exception was scenario 5, performed at 21:00h CET, in which the goal is to measure the performance in a end-of-day residential use case.

4.2.3 Results

The results obtained, reflect an average of 2 rounds. Each round took a duration proportional to the number of supported streams at the client (4 minutes per each increase in the number of transmitted streams). Figure 4-7 and Table 4-3 depict the obtained results for scenarios 1 to 3, whilst Figure 4-8 and Figure 4-9 depict the results obtained for scenarios 4 and 5, respectively.

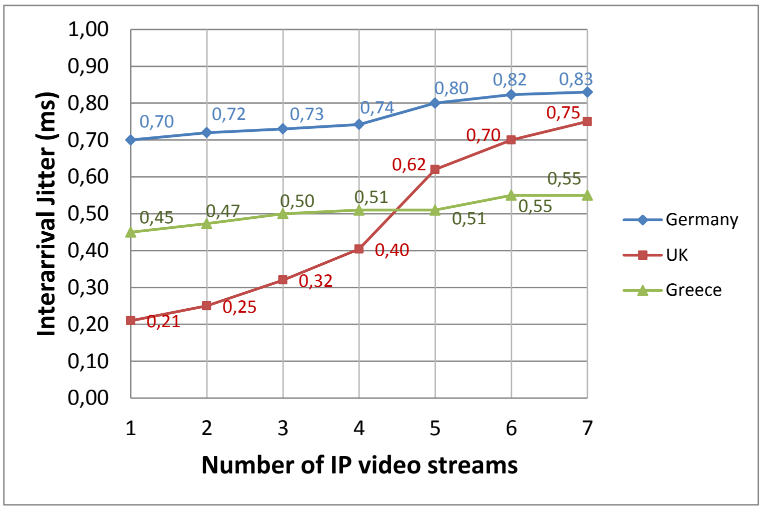

Figure 4-7: Scenarios 1-3 - client inter-arrival jitter (ms) versus Super-peer geographic location

Table 4-3: Scenarios 1-3 - client percentage of packet loss versus Super-peer geographic location

	Germany	UK	Greece
Stream 1	0,00	0,00	0,00
Stream 2	0,00	0,00	0,10
Stream 3	0,00	0,00	0,17
Stream 4	0,00	0,00	0,20
Stream 5	0,00	0,00	0,20
Stream 6	0,00	0,03	0,30
Stream 7	0,00	0,26	0,29

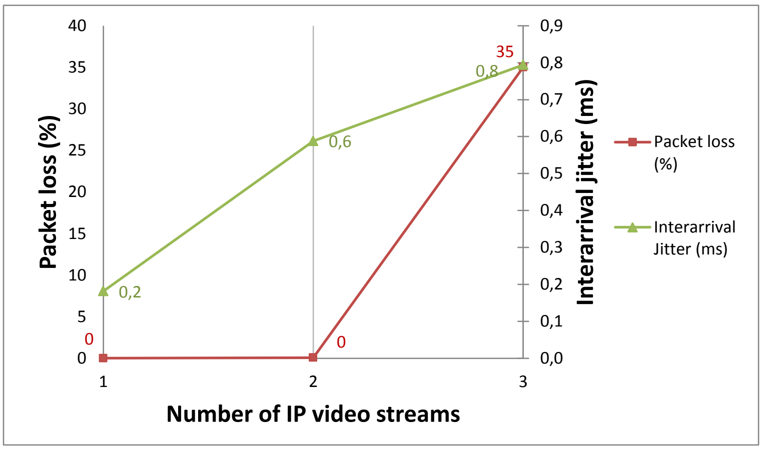

Figure 4-8: Scenario 4 - inter-arrival jitter (ms) and percentage of packet loss in a residential 10Mbps ADSL connection under peak hours

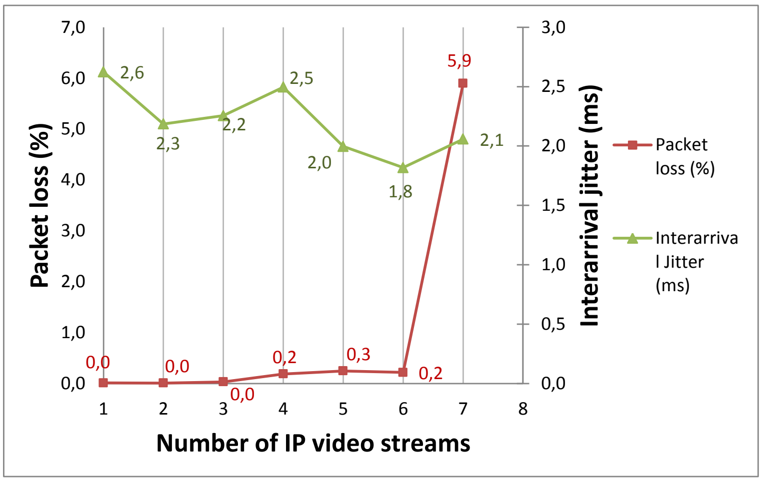

Figure 4-9: Scenario 5 - inter-arrival jitter (ms) and percentage of packet loss in a residential 25Mbps ADSL+ connection during non-peak hours

From the performance graphics it can be concluded that the packet inter-arrival jitter does not constitute a major issue in any of the tested scenarios. This is the case even when the traffic is traversing multiple ISPs under the best-effort treatment.

From Figure 4-7 and Table 4-3, it can be concluded that the service, if deployed in a small to medium scale, could be supported by today's Internet without any change in the ISPs equipment or topology. In larger-scale deployments, with the increase in the number of clients, ISPs would be forced to take improvement measures, and the proposed P2P overlay approach is a possible solution.

From Figure 4-8 it can be expected that a residential 10Mbps ADSL connection will not suitable for users during peak hours. The server was only able to deliver 2 IP streams, above which the packet loss rate increases to unacceptable levels. The service could possibly be scaled down to support the fixed/portable terminals with a stereoscopic display without disparity based rendering (i.e., without 3D depth adjustment control), but with high-quality HD stereoscopic video. However, delivering stereoscopic-plus-disparity or full four-views-plus-disparity is infeasible.

For out-of-the-peak hours, Figure 4-9 suggests that a 25Mbps connection can provide acceptable conditions for up to 6 IP streams, which implies that a stereoscopic service accompanied by disparity maps for 3D depth adjustment can be feasible.

4.3 IRACS performance evaluation

4.3.1 Description

This chapter presents the performance evaluation for the IRACS component in a simulation environment. IRACS is a cross-layer resource over-provisioning approach for the proposed architecture and this evaluation relates with the QoS overprovisioning at the Internet Service Provider (ISP) core network.

4.3.2 Simulation environment

In order to evaluate the benefits of the cross-layer resource over-provisioning support for the dynamic P2P networking, IRACS was implemented and simulated in the Network Simulator 2 (ns-2), version 2.31 [123] using the existing SOMEN [101][102] implementation. The simulated topology is provided in Figure 4-10 and the simulation parameters are given in Table 4-4.

Table 4-4: IRACS simulation parameters

Variables	**Scenario 1** (used for measuring signalling events and number unnecessary requests denied)	**Scenario 2** (used for computing packet dropping statistics)
Link Capacity (C)	1 Gbps	10Mbps
Session Requests	20.000	
Classes of Service (CoS) [124]	▪ 1 Control; ▪ 1 Expedited Forwarding (EF); ▪ 1 Assured Forwarding (AF); ▪ 1 Best-effort (BE).	
Scheduling discipline at the routers [116]	Weighted Fair Queue (WFQ)	
Traffic Characteristics	Based on a Poisson process: ▪ Randomly generated CBR ▪ Pareto ▪ Exponential	
Bandwidth requests	Between 128Kbps and 8Mbps	▪ Between 128Kbps and 1Mbps
Type of requests	▪ 5% are long-lived sessions (lifetime of the whole simulation duration); ▪ 25% are relatively long-lived sessions (lifetime of 30 minutes); ▪ 70% are short-lived sessions (lifetime of 5 minutes).	
Number of Simulation replications	5 times with different seeds of random mapping of requests to CoSs and to peers from different access domains	
Note: the traffic was mapped to the different CoS in order to simulate both the multiple video streams (layers/views) and other traffic, with unpredictable traffic dynamics.		

The simulations results were obtained as mean values for all replications. Input values for the simulation were provided from a bash script file that passes the parameters to the ns-2 OTcl simulation file and the ns-2 simulator automatically to obtain all the results. Then, Matlab is used to plot the graphics shown in the next subchapters.

The results focus on three key performance indicators: scalability, resource utilisation efficiency and QoS differentiation.

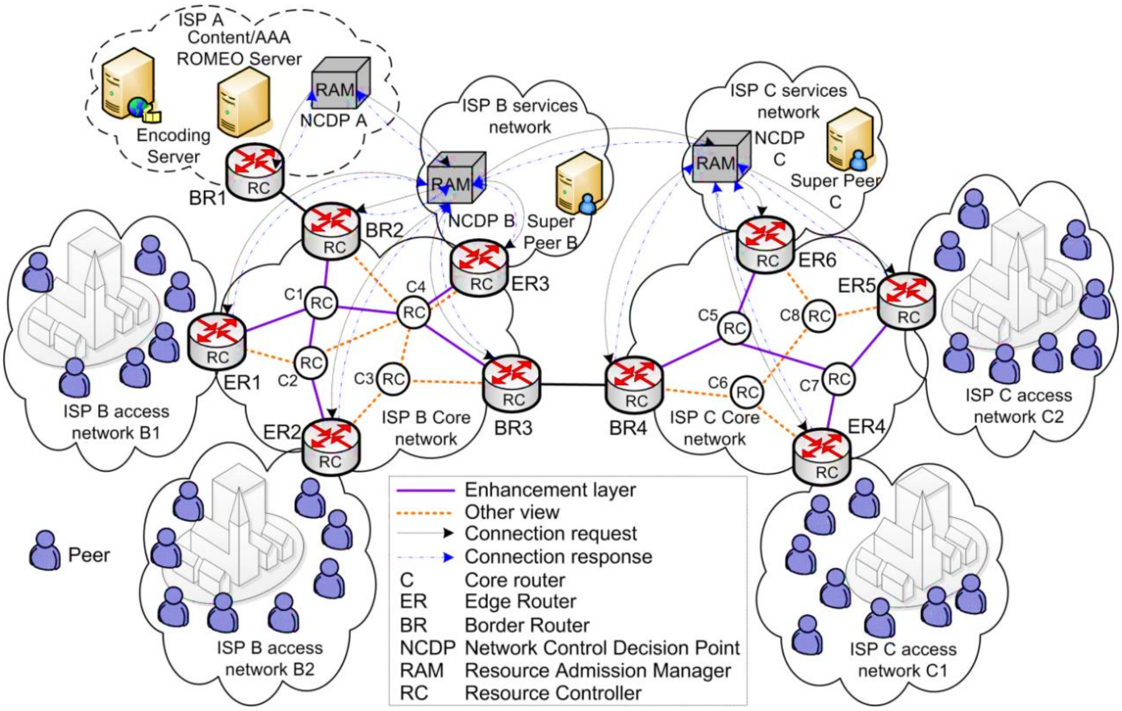

Figure 4-10: Proposed P2P networking approach using IRACS resource control approach

4.3.3 Results

Figure 4-11 compares the IRACS resource over-provisioning with per-flow QoS mechanisms, using simulation scenario 1.

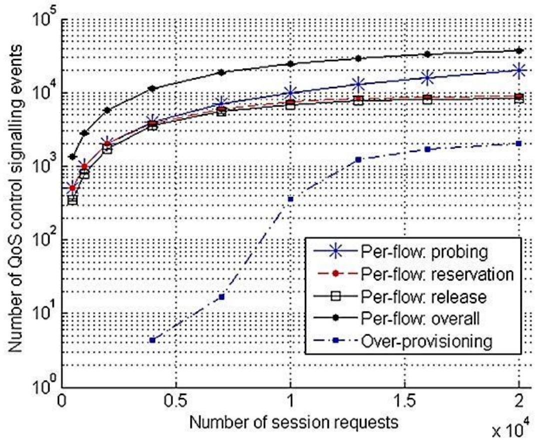

Figure 4-11: QoS over- provisioning vs. per-flow control: signalling events number (scenario 1)

It is therefore worth mentioning that, in per-flow approaches, a multicast tree is probed (probing events) to acquire its resource availability before admission of a new request in the tree. In case there is sufficient resource available on a multicast tree, the latter is signalled, so the requested amount of resource is reserved (reservation events) to guarantee the QoS contracted for the incoming request. When a session terminates in a tree, appropriate signalling is triggered to release (release events) the related reservations for future use. Hence, we observe that the IRACS over-provisioning is able to reduce the overall per-flow signalling events (i.e., probing + reservation + release) above 90%, depending on the network congestion level (see Figure 4-11). The over-provisioning does not trigger signalling when the network is less congested (session requests below 4,000). The signalling is triggered for reservations re-adjustment only when the over-reservation in a requested CoS is exhausted. With respect to per-flow, the probing and the reservation events numbers overlap when the network is less congested (request number below 4,000); all requests are admitted then. As the resource availability decreases (request number beyond 4,000), the number of probing goes higher than that of the reservation; some requests are denied since there are not enough resources to guarantee the demanded QoS.

Additionally, Figure 4-12 shows that the IRACS over-provisioning guarantees QoS differentiation similar to per-flow approach when there is link congestion. These results have been obtained by activating the real traffic in the simulation environment, thus the need to use a simplified version of simulation scenario 1, called scenario 2 (10Mbps links only, with maximum reservation being 1Mbps).

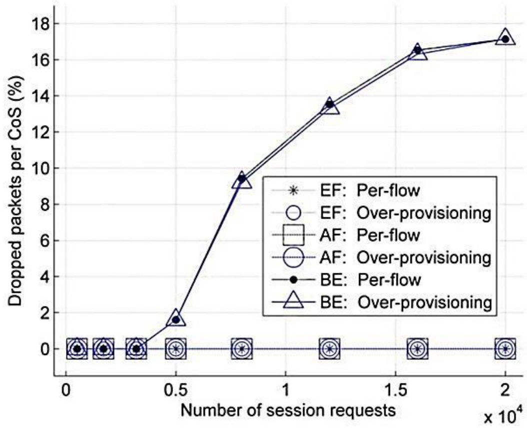

Figure 4-12: Packet dropping statistics of IRACS vs. per-flow approach (scenario 2)

It is possible to see that when link congestion occurs, both IRACS and the per-flow approach have a similar behaviour, that is, BE traffic (which has the higher drop precedence) starts to be dropped in order to ensure enough resources are available for the EF and AF classes. Further increase in traffic would have caused higher packet drop for BE immediately followed by AF traffic (in a lower proportion). As explained in Chapter 2.11, due to the stipulated EF class traffic characteristics [83], it would only be dropped for very high congestion levels.

Considering that over-provisioning is usually challenged by the utilisation of the residual resources (over-reserved and unused), Figure 4-13 is used to compare IRACS with a competing over-provisioning approach, MARA [91]. MARA has been chosen as the most closely related solution to IRACS. In Figure 4-13 it is observed that IRACS does not deny any request when residual resources are sufficient for a new request while MARA denies many requests, especially as the network demands increase. This means that IRACS reuses the residual resources more efficiently in contrast to MARA.

Based on the performance evaluation described in this simulation, it becomes clear that the traditional per-flow QoS approaches incur undue signalling and therefore the related processing overhead, which poses serious scalability issues. In this sense, they are not suitable for dynamic scenarios like the one analysed in this thesis. As an alternative, IRACS demonstrates that efficient resource over-provisioning is fundamental to guarantee differentiated QoS not only in a scalable fashion, but also with increased network resource utilisation efficiency.

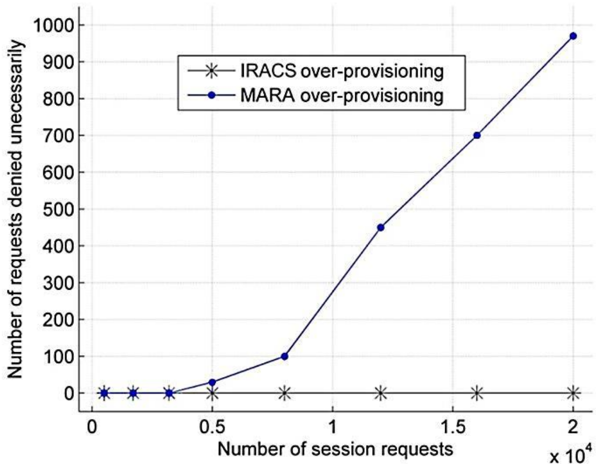

Figure 4-13: IRACS over-provisioning vs. competing previous over-provisioning approach (scenario 1)

4.4 Conclusions

In the P2P networking tests, there were a total of 7 different video streams with data rates of about 28 Mbps for the downstream and 8 Mbps for the parent upstream (2 children per parent, 4Mbps per stream). At component scale it was observed there is still room for improvement on the software development as the CPU consumption in components like the *P2P Packetisation/Transmitter* and the *P2P Receiver & Forwarder* can be a problem in a real implementation scenario. However, the test results have also shown highly scalability and fast response time for the other components, including the P2P overlay construction algorithm and the tree maintenance operations, achieving a proof-of-concept status.

The inter-ISP tests allowed understanding how good the presented solution would work on the current Internet. From the results it can be concluded that the service, if deployed in a small to medium scale, could be supported by today's Internet without any change in the ISPs equipment or topology. There would however not be any QoS safeguarding which could cause some service instability in lesser good ISPs. It was also observed that a 25Mbps connection would provide acceptable conditions to receive up to 6 IP video streams, which implies that a stereoscopic service accompanied by disparity maps for 3D depth adjustment can be feasible.

Furthermore, the simulation results for the IRACS have shown that each connected session will receive a minimum guarantee of its desired QoS in terms of bandwidth and delay through P2P overlay system. The results have also shown that the use of efficient resource over-provisioning ensures scalability since the control states and signalling overhead (and therefore the related CPU consumption, energy, and memory) can be significantly reduced.

Chapter 5

5 Integrated System Validation

Chapter Outline: this chapter describes the validation testbed, the experiences and the methodology used for the interpretation of the results for the integrated system validation. The system has been validated as a component of the ROMEO project [31] where end-user validation has been performed on the project demonstrator. Chapter 5.1 highlights the tested features, Chapter 5.2 describes the testbed, the composition of the test panel and the test stimuli, while Chapters 5.3 and 5.4 present and analyse the subjective and objective evaluation results. For the subjective assessment, ITU recommendations have been followed and the Mean Opinion Score (MOS) for each test event has been computed by averaging subjects' normalised scores.

5.1 Tested features

Chapter 4 presented objective measurements (based on testbed, simulation and analytical work) for singular architecture components (Server, P2P and Network). This chapter, on the other hand, presents both subjective and objective measurements for the full system validation. The test conditions identified and described in Table 5-1, will enable to measure the impact of the Server and P2P components/modules in the full system, which will be under subjective evaluation by a panel of independent experts. These scenarios were presented to the participating subjects according to the sequence illustrated in Figure 5-1.

5.2 Used topology and test procedure

For each subject, the tests were shown one after the other, with a certain time interval between each. The time interval is kept wide enough to let the subjects perceive the immediate and transient effects of the test case in question and also vote on the score sheet in front of them. Figure 5-1 shows the timeline of the subjective tests during the validation.

The topology used for the user trials is shown in Figure 5-2.

Table 5-1: Tested P2P scenarios

Scenario/Condition	Description	Purpose
0 - Perfect network conditions	This corresponds to the higher quality to be presented to the subjects. There are no bandwidth limitations, no congestion and the P2P network is stable (no churning).	Provides the ideal viewing conditions. There are no networking impairments and all anticipated streams are delivered with the best quality.
1 - Graceful disconnection of a Top-level peer in the P2P distribution system	A Top-level peer realizes it is going to be shut down and immediately performs a graceful disconnection, alerting the *Multicast Tree Manager* for this condition. Its children are expected to swift react and switch for the backup parent, before the Top-level peer disconnects. Changes are also expected to be immediately reflected on the Topology Builder.	Evaluates the immediate and transient effects on the perceived quality of the video when a Top-level peer (highest in the P2P hierarchy) is shutdown in a graceful manner (e.g., the user shuts down the computer or closes the player).
2 - Reconnection of a peer into the system	A new peer with excellent *evaluation* joins the P2P distribution system. It is expected this peer to start as a leaf and to transition to a Top-level peer during the *tree maximization* procedure periodically performed by Topology Builder. It is also expected that at least one children of another Top-level parent will switch parents (to the new Top-level peer).	Evaluates the immediate and transient effects on the perceived quality of the video that occurs when there are multiple changes simultaneously occurring at the P2P network.
3 - Ungraceful (sudden) disconnection of a Top-level peer in the P2P distribution system	A Top-level peer is disconnected before signalling the *Multicast Tree Manager* of the condition. It is expected the NMS module in the Top-level peer's children to detect the fault and, upon time-out, switch for the back-up parent before the P2P receiver buffer gets empty. Changes on the P2P distribution system are expected to be updated on the Topology Builder upon the time-out. Furthermore, it should also be seen that one of the children of the disconnected Top-level peer will be promoted to Top-level peer.	Same as in Condition 1, but occurring when a Top-Level peer is abruptly disconnected from the network (e.g., connectivity lost due to network cable removal or operating system/application crash)

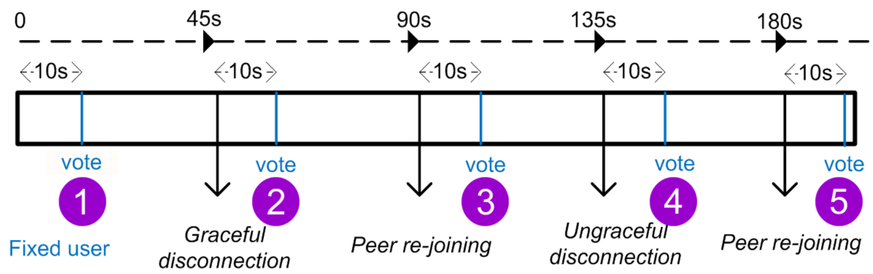

Figure 5-1: The timeline of the subjective tests (6 minutes and 10 seconds overall per subject) during the final trials. The fixed terminal user and the portable terminal user, experience the corresponding events simultaneously and give their votes separately (adapted from [31])

During the voting times, subjects were asked to put their opinion on the voting sheet based on a rating scale, as it will be explained in Chapter 5.3.

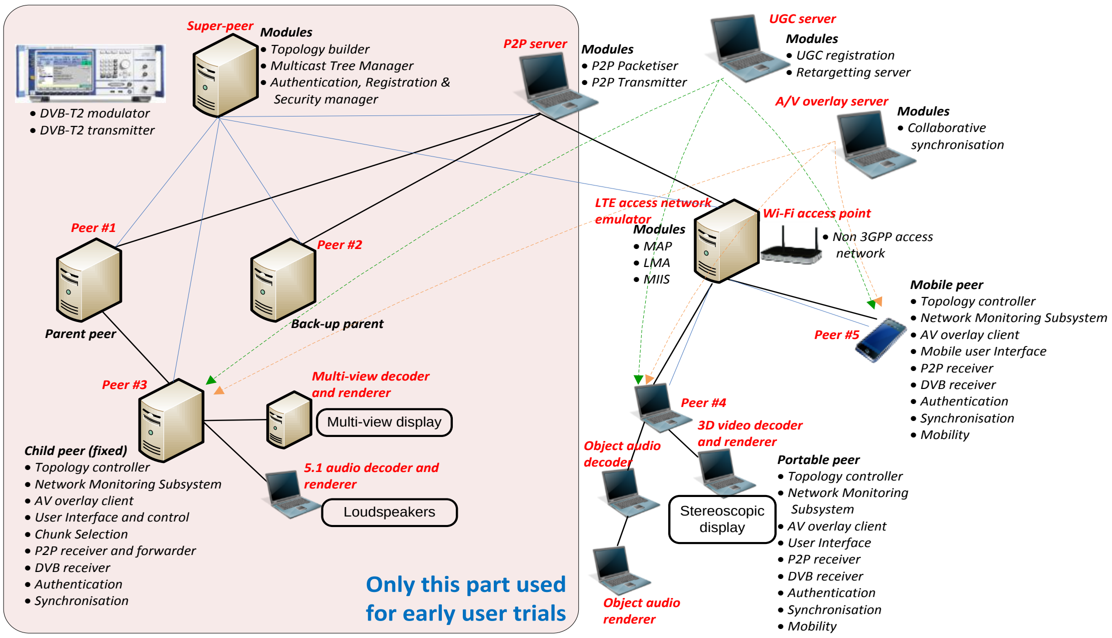

Figure 5-2: Setup used for the full system validation [31]

5.2.1 Test panel

For the system validation, the subjects were mostly non-experts with no or little experience with multi-view 3D video and auto-stereoscopic display technology. The complete test panel consisted of 53 subjects out of which 19 were female and 34 were male. The minimum age was 21 and the maximum age was 62 and the average age of all subjects was 35. The number of subjects guarantees the statistical validity of the evaluation.

5.2.2 Test stimuli

Two different multi-view plus disparity content tests were used, one named "Musicians"[***] and the other "Acting scenes"[†††], which were generated and processed as part of the ROMEO project [31]. Both contents were encoded according to the SVC extension of the H.264/AVC encoding standard (Annex G), using the CABAC entropy encoding scheme, at 30 frames per second in Full High Definition resolution (1920x1080) using a chroma/YUV format of 4:2:0 with 8 bits for chroma and 8 bits for luminance. To ensure the content duration for the subjective tests, the original scene has been concatenated ten times and encoded. For clarity, only the video evaluation is presented below. In the case of spatial audio (5.1 or object based audio), the major influential factors on the quality were the encoding scheme applied and the rendering

[***] Musicians – 3D video footage of a small band of acoustic music players, indoors. The camera rig and the musicians are on the same level

[†††] Acting scene - 3D video footage of a small theatrical act, indoors.

environment, including the spatial alignment with the video, which are out-of-scope for this thesis. The network transmission has negligible impact, owing to the considerably lower transmission overhead of spatial audio compared to multi-view video (i.e., no adaptation is applied).

5.3 Subjective test results - 3D video experience evaluation results

As identified in Table 5-1, different target network events have been applied/triggered on the P2P topology to find out their perceptual impact on the 3D video experience and assess the capacity of the system in maintaining service quality and continuity. The used evaluation rating scale, compliant with ITU-R scales, is shown in Figure 5-3.

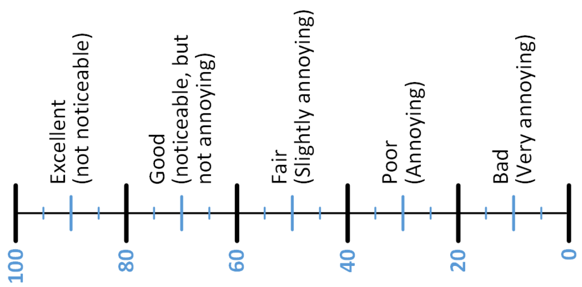

Figure 5-3: Evaluation rating scale

Subjects have been asked to mark their opinion on certain time intervals (see Figure 5-1), which constitutes a continuous quality evaluation. This had been defined in the ITU recommendation BT.500 [126] as *Single Stimulus Continuous Quality Evaluation* (SSCQE) methodology that does not require a reference. After the completion of tests, for each subject, the scores have been normalised to the range of [0, 1]. The normalisation process has been performed for each subject's own group of scores. The normalisation process ensures that the entire range from the lowest grade to the highest grade is utilised for each subject.

The Mean Opinion Score (MOS) for each test event has been computed by averaging subjects' normalised scores. However, it has been necessary to remove outliers in the scores for a test in order not to negatively affect the computed QoE figure. In order to ascertain whether the distribution of scores for any test presentation was normal or not, the β_2 test, as explained in the aforementioned ITU-R BT.500 [126], has been used. This test refers to the calculation of the *kurtosis coefficient* of the function (i.e. the ratio of the fourth order moment to the square of the second order moment) allowing, for a particular test, to detect any outlier scores by any subject that should be removed from the average score computation.

Figure 5-4 shows the calculated MOS for the applied tests conditions (see Figure 5-1) for both the fixed terminal user and the portable terminal user. The first vote refers to the ideal viewing conditions, i.e., there are no networking impairments and all anticipated streams are delivered steadily to both user terminals with the best quality.

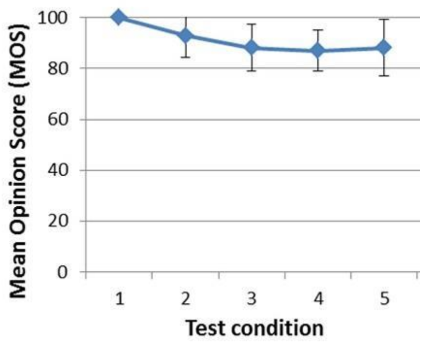

Figure 5-4: MOS for fixed and portable terminals during the system validation with 95% confidence intervals ([0,1] interval is rescaled to [0,100] to represent MOS). The order of the test conditions is the same as in Figure 5-3 [31]

Figure 5-4 shows that none of the test conditions for any of the two user terminal types leads to a perceptually annoying impairment that can distract the subjects. It also shows that the following three network conditions (2, 3 and 4) were detected, having a similar level of impact on the perceptual 3D video quality. This lower classification was caused by a minor video freezing effect (a few milliseconds) that occurred every time the peer changed parents, or its position in the P2P hierarchy. The video player was not fully optimized to deal with the UDP socket change - the IP address from the source of the stream changes, hence a new socket is created and the old is destroyed (after all data has been passed to the player buffer). Furthermore, the observed glitches towards the end of the test, which indeed corresponds to the fourth test case (see Figure 5-1), where the transient impact of parent peer's ungraceful disconnection is seen. Because, in the incident of an ungraceful parent peer disconnection, the child peer immediately switches to its backup parent and tries to maintain packet reception continuity. However, at this instance, several packets are received and stored in the P2P receiver's input buffer twice (i.e., received from the disconnected parent and the back-up parent) and even due to the jitter, some packets are received out of order. This situation is handled more efficiently in the graceful disconnection scenario, where the handovers are done following handshakes. This indicated that there was a need for further improvement in the child peer's packet reordering capability.

this impact is negligible as the MOS remains in the interval of [80,100] taking into consideration the confidence interval. **This shows the fact that the P2P clients developed and installed in the user terminals are able to mitigate the impact on the 3D streams' delivery continuity fairly quick.**

A slight degradation in condition 4 (ungraceful parent disconnection) is observable as the child peer immediately switches to its backup parent and tries to maintain packet reception continuity.

However, at this instance, several packets are received and stored in the P2P receiver's input buffer twice (i.e., received from the disconnected parent and the backup parent) and even due to the jitter, some packets are received out of order. This situation is handled more efficiently in the graceful disconnection scenario (2), where the parent switch is done prematurely (anticipatory).

5.4 Objective test results – P2P performance results

The P2P 3D content distribution system is evaluated according to the ITU recommendations J.241 (Quality of service ranking and measurement methods for digital video services delivered over broadband IP networks) [127] and Y.1541 (Network performance objectives for IP-based services) [128]. The following is a summary of the relevant performance parameters as specified in these recommendations:

- *IP packet Loss Ratio* (IPLR): is the ratio between the number of the packets lost in the network and the total number of packets transmitted by the sender.
- *IP packet Transfer Delay* (IPTD): is the time interval between initial transmission and final reception of a packet.
- *IP packet Delay Variation* (IPDV): is the IPTD variation between consecutive packets of the same flow.

Table 5-2, Figure 5-5, Figure 5-6 and Figure 5-7 provide the measured results for the P2P distribution system. In accordance with ITU recommendation Y.1541 [128], the measurement of the results was performed using multiple intervals of 1 minute. The reason behind this choice is that the evaluation intervals should be sufficiently long to include subsets of the packet population of interest whilst being at the same time sufficiently short to ensure a balance of acceptable performance throughout each interval (also addressing the practical aspects of the measurement).

Table 5-2: Measured IPLR values versus P2P tree depth

P2P tree depth	IPLR	Comments
1st level (parent)	0%	No packet loss was observed for the transmission of the 14 IP video streams under ideal network conditions.
2nd level (parent/children)	0%	
3rd level (child)*	0%	

** 3rd level is a later addition to the demonstrator set-up and served to evaluate the performance of the P2P content distribution system for an extended tree depth level.*

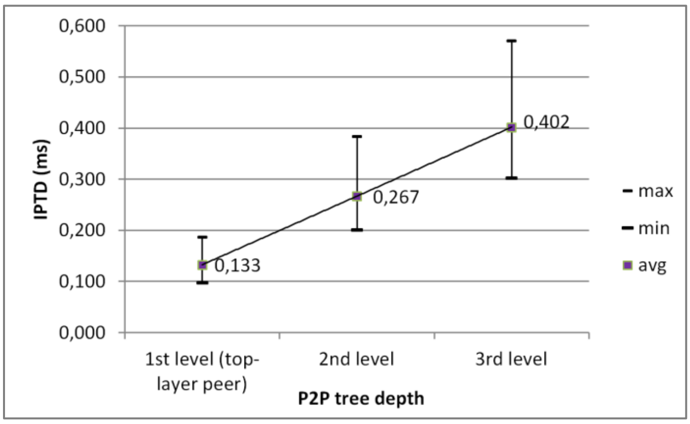

Figure 5-5: Measured IPTD values versus P2P tree depth

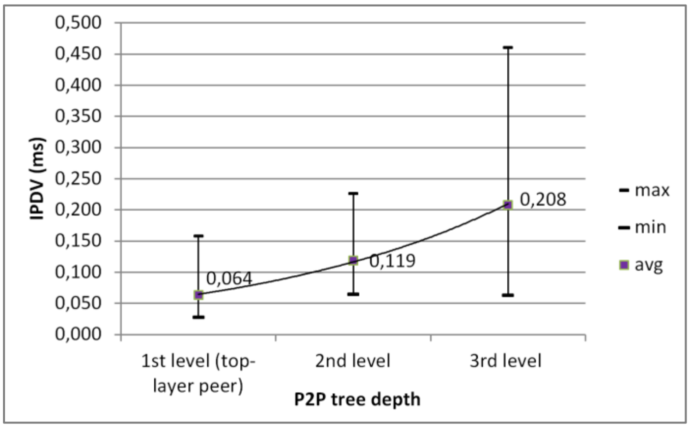

Figure 5-6: Measured IPDV values versus P2P tree depth

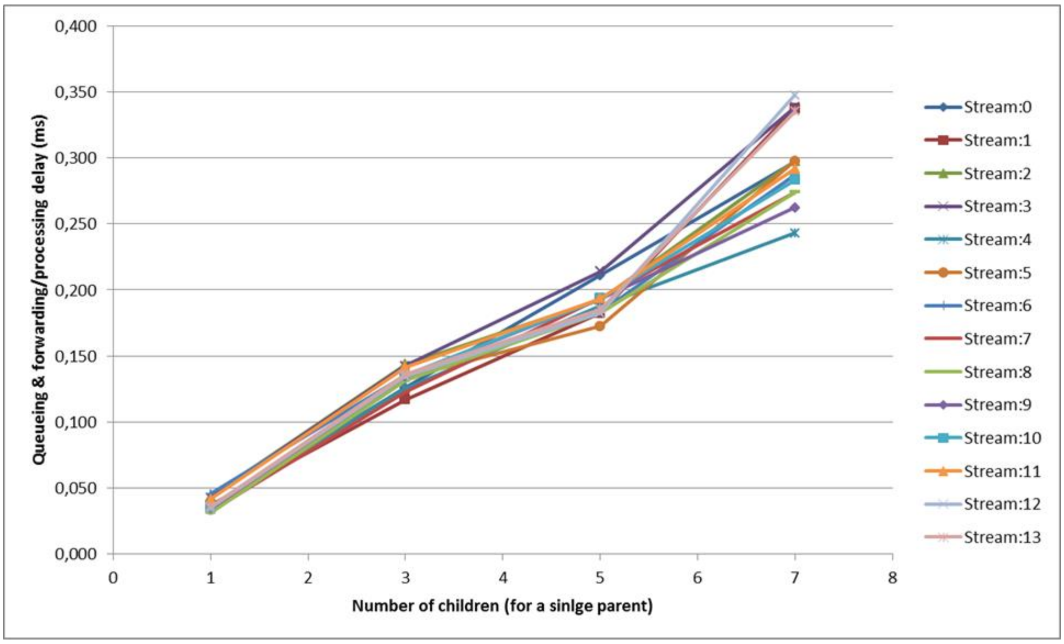

Figure 5-7: Queueing and processing/forwarding delay for the demonstrator

To understand these values it is important to remind the test setup used link speeds of 1Gbps, with transmission of chunks of variable size (estimated average of 1380 bytes) in optimal network conditions (no load). Furthermore, the results depicted in Figure 5-7, correspond to the queuing and forwarding delay at the peer, that is, the amount of time between a chunk arrival (fully) and the moment the first bit of that chunk is forward to a child peer. The goal is to show how much delay is added by each forwarding peer in the P2P network and how that increases with the number of children peers it has to feed. It is thus related to time in queue and processing delay, mostly related to the peer's CPU processing capacity and code optimization at the *P2P Receiver & Forwarder* module.

5.5 P2P results analysis

Under the ideal network conditions provided by the validation set-up, the performance results for the IPLR (see Table 5-2), IPTD (see Figure 5-5) and IPDV (see Figure 5-6) are within excellent service quality for IP television or streaming services as specified by ITU recommendations J.241 [127] and Y.1541 [128] – see Table 5-3:

Table 5-3: Subset of IP network QoS class definitions and network performance objectives

Network performance parameter	QoS Classes		
	Class 0	Class 6 (provisional)	Class 7 (provisional)
IPLR	10^{-3}	10^{-5}	
IPTD	100ms	100ms	400ms
IPDV	50ms (small variations are allowed, dependent on the capacity of the network links)		
Being Class 0 (Real-time, jitter sensitive, high interaction) and Class 6 (later addition from Y.1541) the most relevant to the evaluation.			

Figure 5-7 depicts the measured results for queueing and processing/forwarding delay at each peer. Basically it reflects the amount of time a IP packet takes from the moment it is received by a peer till the moment it is relayed/forwarded to a downstream child.

Considering each additional tree level (hop) increases the IPDV and the IPTD, the measured values in Figure 5-6 and Figure 5-7, associated to an administrative decision on the maximum number of hops, makes it possible to estipulate the total amount of delay caused by the proposed P2P distribution system – please note that the distance will have a negligible impact, as the peers will always be in close proximity due to the way the P2P application level multicast tree algorithm works.

Considering the goal is to target a real deployment scenario, let us consider a hypothetical reference path from the Super-peer to the Edge Router (ER) serving a service customer, as depicted in Figure 5-8.

In the hypothetical reference path, in accordance with ITU-T recommendation Y.1540 (Internet protocol data communication service – IP packet transfer and availability performance parameters) [129], a single network section is defined as a set of devices and their interconnecting links responsible for providing ‘a part’ of the IP service between the source (the Super-peer) and the destination (the customer). Also, each network section represents a different Internet Service Provider (ISP) / network operator.

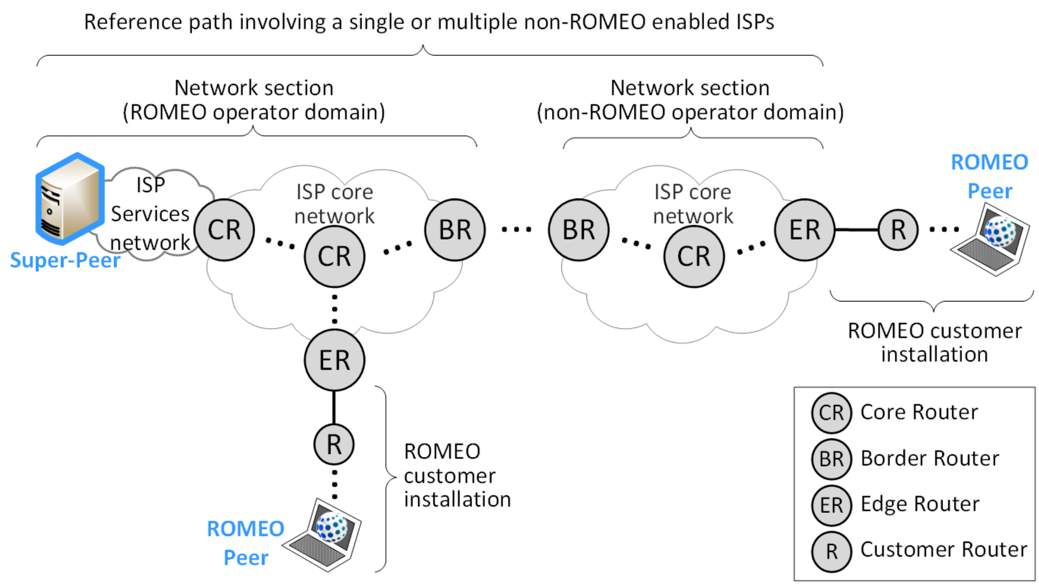

Figure 5-8: Hypothetical reference path for measuring networking performance objectives under typical commercial deployment usage scenarios

For the hypothetical reference path it is considered the typical delay and jitter values for Core Routers (CR), Border Routers (BR) and Edge Routers (ER) used in ITU recommendation Y.1541 [129] (revised in December 2011), as depicted in Table 9.

Table 5-4: Examples of typical delay contribution by router role (as in ITU recommendation Y.1541)

Type of device	Average total delay (sum of queueing & processing)	Delay variation (jitter)
Core Router	2ms	3ms
Border Router*	3ms	3ms
Edge Router	10ms	16ms
* in recommendation Y.1541 the *Border Router* functionality is performed by a *Internetworking gateway*.		

For route lengths greater than 1,200Km (satellite links excluded), the route length calculation as defined in another ITU-T recommendation, G.826 (End-to-end error performance parameters and objectives for international, constant bit-rate digital paths and connections) [130] will be used, which is obtained by multiplying 1.25 by the air distance between the source (location where the Super-peer is located) and the destination (location where the service customer resides):

$$R = 1.25 \times air\ distance\ \ (in\ Km),$$ where R is the route length.

Table 5-5 gives the results for three hypothetical references taken from Figure 4-6, in terms of number and type of routers, distance, and contribution of all these components to IPTD and IPDV - please note that IPDV values assume the worst case addition of each node and therefore are pessimistic.

Table 5-5: Analysis on the network performance results on real deployment scenarios

	Munich, DE → Lisbon, PT			**London, UK** → Lisbon, PT			**Patras, GR** → Lisbon, PT		
	Value	**Avg. IPTD (ms)**	**Max. IPDV (ms)**	**Value**	**Avg. IPTD (ms)**	**Max. IPDV (ms)**	**Value**	**Avg. IPTD (ms)**	**Max. IPDV (ms)**
Air distance (Km)	2010			1610			2690		
R - Route length (Km)	2513	13	---	2013	10	---	3363	17	---
Insertion time[1]	<0ms	---	---	<0ms	---	---	<0ms	---	---
Number of hops[2]	12	---		11	---		18	---	
Number of Core Routers	6	12	18	5	10	15	8	16	24
Number of Border Routers[3]	4	12	12	4	12	12	8	24	24
Number of Edge Routers	2	20	32	2	20	32	2	20	32
Last hop (customer installation)[4]	<0ms	---	---	<0ms	---	---	<0ms	---	---
	Total	**44**	**62**	Total	**42**	**59**	Total	**60**	**80**

[1] the amount of time a IP packet takes to be transmitted on the wire (also known as serialization time). The access network (customer connection to the Internet) is assumed to be at least 10Mbps (minimum requirements to have full resolution stereoscopic view) and packets to have 1200 bytes;
[2] the number of hops is taken from real executed traceroutes (as explained in Chapter 4.1);
[3] the number of Border Routers is extracted from the abovementioned traceroutes;
[4] as observed in Figure 5-5, Figure 5-6 and Figure 5-7, the impact of last hop is negligible.

Using the provided hypothetical reference paths as examples of real-case deployment scenarios, the values show that existing QoS rules can support the proposed over-the-top service for a subset of ISP customers located on non-modified ISPs. Nevertheless for more stringent values for the IPDV on longer hypothetical reference paths, additional service-level agreements (SLAs) need to be established.

5.6 Conclusions

Independent evaluation has shown that none of the test conditions had led to a perceptually annoying impairment, the three network conditions (Table 5-1) that were triggered had negligible impact on the perceptual 3D video quality. **This validates the algorithms and performance of the P2P components. It was shown that network dynamics caused by events such as peer arrivals, or appearance of better parents, could be easily and quickly adapted by the proposed solution.**

Analytical analysis has also shown the delay and jitter (IPDV) applicable to the proposed architecture (worst case conditions) is comfortably within the margins defined by ITU to support digital video services delivered over broadband IP networks. One remaining important factor that needs to be taken in consideration is the available bandwidth for Internet consumers. In the full system proposed approach, there will be a total of 14 different IP video streams. The views differentiate according to their content (base layer, quality enhancement layer, colour, depth and description). A rough estimate of the required bandwidth that includes all these IP streams leads to data rates of about 56 Mbps for the downstream and 20 Mbps for the upstream (parent peers feeding an average of 5 children). In laboratory environment, the provisioning of such data rates is not a problem but in today's access networks, the limitations of bandwidth are still an obstacle for testing the proposed concept on an international scale. It can be expected that the 'Future Internet' will be able to provide these resources but the key question is when this is likely to happen. Starting from this question, a brief analysis was carried out to establish the current situation and extrapolate the current trend into the future access rates. Akamai's state-of-the-Internet website [132] and reports were used to obtain average connection speeds (including average peak rates) for most European countries, as depicted in Table 5-6. It was also possible to obtain the trend in both the connection speeds and Internet connections capable of supporting 15 Mbps or more. The analysis shows that a sufficient number of users currently have an Internet connection capable of supporting some level of the intended services.

Table 5-6: Average connection and speed in Europe (Q3'15) (source Akamai [132])

Country	Avg. connection speed (Mbps)	Avg. peak speed (Mbps)	Country	Avg. connection speed (Mbps)	Avg. peak speed (Mbps)
Sweden	16.1	62.8	Austria	10.9	43.5
Switzerland	15.6	59.4	Germany	10.7	46.8
Netherlands	15.2	60.9	Portugal	10.4	48.2
Norway	14.3	50.0	Slovakia	10.3	44.0
Finland	14.0	53.2	Hungary	10.0	51.7
Czech Republic	13.9	48.7	Poland	10.0	43.5
Denmark	12.9	48.1	Spain	9.7	47.4
Romania	12.8	72.1	Russia	9.6	54.2
Belgium	12.4	57.3	France	7.9	37.2
United Kingdom	11.8	50.9	Italy	6.4	30.2
Ireland	11.0	46.4	Turkey	6.3	37.5

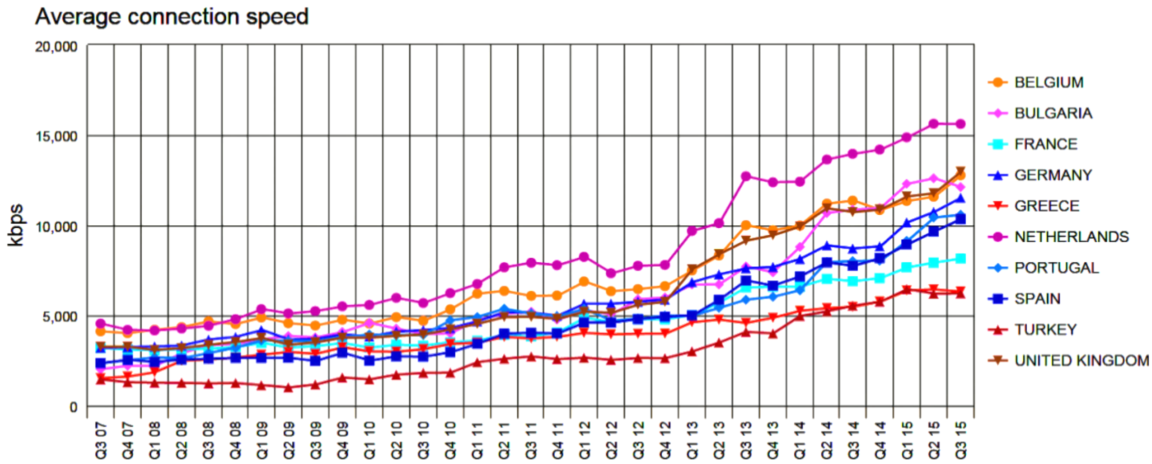

Figure 5-9: Average Internet connection speed in Europe (source Akamai [132])

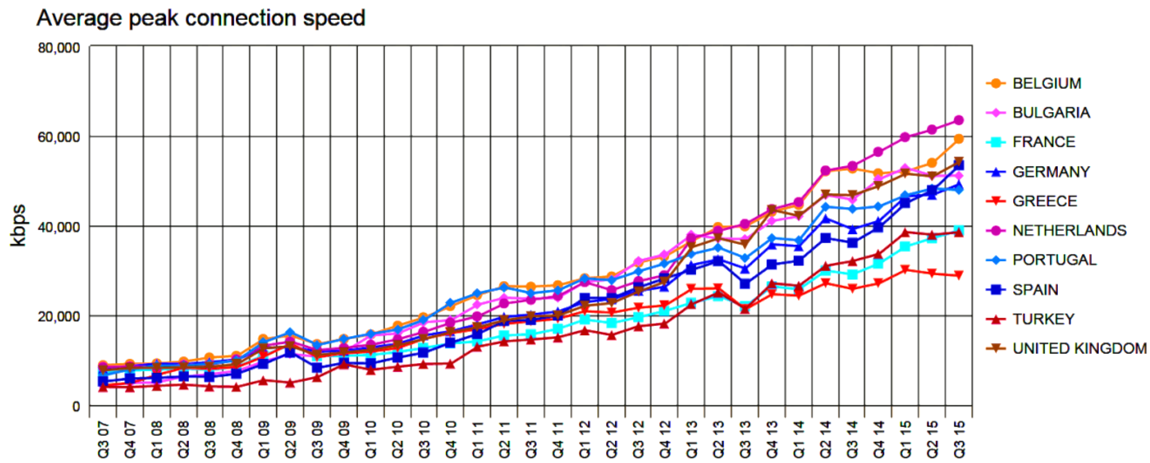

Figure 5-10: Average peak Internet connection speed in Europe (source Akamai [132])

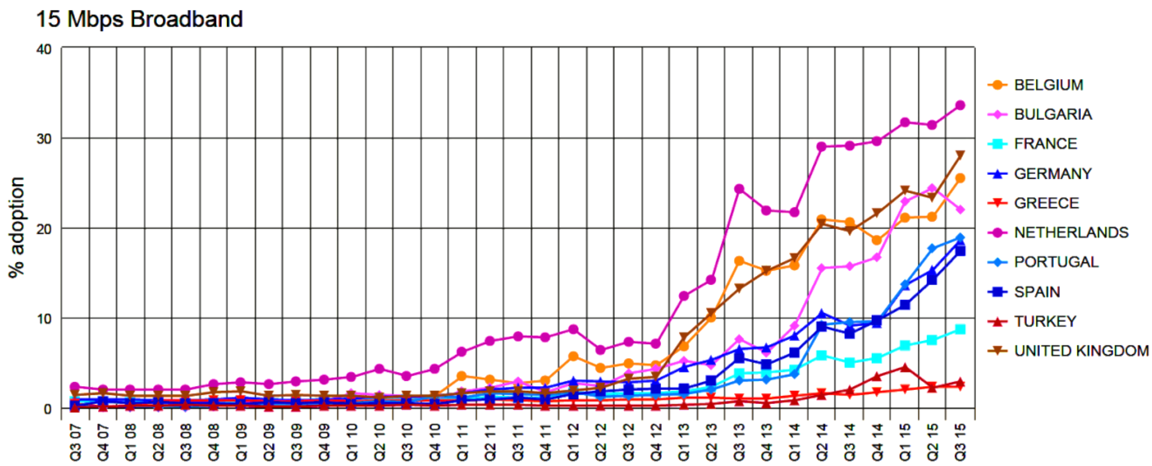

Figure 5-11: Percentage of clients with at least 15Mbps access to the Internet (source Akamai [132])

The data rates that are supposed to be necessary to support the full service (all 15 IP streams with 56Mbps/20Mbps) could, in this scenario, be expected to become available in the next two or three years, provided that other variables do not change. These other variables would include the video coding standard for which H.264 was used. It can be expected, that the HEVC standard (H.265) would result in data rates about 50 % lower than those required for H.264.

Chapter 6

6 Conclusions

6.1 Main Findings

Major challenges for service providers lie ahead, especially in what regards the efficient use of their network resources. IP multicast technology may have its opportunity to finally be deployed in large scale. Meanwhile the top sites on the Internet continue to use unicast client-server approach, which is known to be inefficient when distributing large volumes of the same data to multiple recipients. However, it is not possible to predict with any certainty how the Internet will mature - it is unclear what would be the globally adopted solution in the next years.

With this in mind this thesis is proposing a new architecture for multiview real-time media distribution over a next generation network. It creates a new concept of multihoming PA-CDNs delivering content from both television broadcast services, through DVB-T2, and the Internet. It also introduces the concept of IP-multicast-to-IP-unicast replicator in a novel approach for geographically contained hybrid client-server/P2P distribution network, retaining the desirable properties of each while improving both. Furthermore, the proposed architecture unleashes the power of IP multicast at the ISP core network together with a new P2P overlay construction algorithm for the use of multiple application layer multicast trees at network access level. When compared to pure client-server or P2P distribution scenarios, the proposed approach is able to effectively distribute the load among most participating nodes, avoiding inter-ISP (unwanted) traffic, while respecting individual node bandwidth constraints and achieving a fast insertion and tree reconstruction time. The use of multiple distribution trees differs from many other approaches, and enables a fair share among participating peers – proportional to their rank - being the only exception the leaf peers. Leaf (child) peers are considered the lowest rank peers and in most situations their participation on the distribution of content is therefore minimized – mobile terminals are provided as a good example of leaf peers since they have considerable restrictions in their download/upload capacity, traffic quotas, and battery.

When compared to the well-known CoopNet [68][69] and SplitStream [63], the proposed architecture uses of a more complete rank methodology, which allows a better adaptation to the heterogeneity of P2P networks, considers different trees to distribute different content and also different is the approach to provide resilience to peer failures - backup links are calculated offline

- and selective updates are made upon peer failure. The approach also allows each peer's upload rate to be controlled, contrary to CoopNet (and many other proposals), and proposes a hybrid centralized/decentralized approach, contrary to SplitStream which uses a pure decentralized approach. These characteristics better enable content administration and searching functions, while also allowing a faster tree update mechanism.

In addition, this thesis also describes IRACS, a novel QoS control architecture, capable of obtaining real-time knowledge of link's resources and providing QoS adaptation in a timely manner. IRACS further improves existing QoS resource over-provisioning techniques, by lowering the signalling overhead whilst still differentiating classes of service. In particular, the proposed architecture attempts to ensure that each connected session will receive at the ISP core network, a minimum guarantee of its desired QoS in terms of bandwidth and delay. On one hand, this is important to reduce synchronisation design complexity and buffering requirements on end-users' devices while the perceived quality of live multimedia content can be improved in heterogeneous networks. On the other hand, the use of efficient resource over-provisioning allows scalability since per-flow QoS reservation control overhead can be avoided without incurring waste of resources.

Individual component validation performed in Chapter 4 and independent evaluation performed by an external panel of subjects in Chapter 5, has validated the concept of the proposed architecture. Furthermore, it is demonstrated in Chapter 4.2 that the current Internet (as is) is capable of supporting an over-the-top solution for the provision of the multiview real-time media.

6.2 Future Work

The following are topics of interest that could be studied in order to further enhance the proposed architecture for multiview real-time media distribution:

- Instead of considering the deployment of Super-peers in the ISP's dedicated Services Network, a multiview service provider could choose to use cloud services for deploying the Main Server/Super-peer. Cloud services enables globally distributed services, quickly deployment, easy management and optimized computing infrastructure. The use of cloud services would have the potential to facilitate service replication across multiple globally distributed instances allowing the services to move closer to consumers. It could also improve service fault tolerance. One important factor is that in some geographical regions the cloud provider would need to be selected based on proximity and/or SLA agreements with ISPs on the intended geographical coverage.
- To use of software defining network (SDN) potentiated by network functions virtualization (NFV). NFV is an enabler of dynamic service provisioning. By replacing

some multiview service elements with virtual network functions, new functions can be added or improved just by updating a software image, rather than waiting for a vendor to develop or manufacture a dedicated appliance. If integrated with SDN, service providers can express and enforce application traffic management policies and application delivery constraints at the required level of granularity.

- One such example of SDN/NFV applicability is to move the *Topology Builder* and *Multicast Tree Manager* functionalities, as described in Chapters 3.2.1.1 and 3.2.1.2, to the *Edge Routers*. Providing these routers will have enough resources, it is possible to improve the performance of the system. Each access network will then be a fully self-contained distribution system and all the P2P signalling would be dealt by the ER. This approach creates a distributed PA-CDN where each ER acts as a proxy for its access networks. This principle is also applicable to the *Border Routers* that interface with non-compliant ISPs.

- To further validate IRACS as a highly scalable QoS overprovisioning solution:
  - By study its resilience in dynamic scenarios with links and nodes failures;
  - By studying its performance with dynamic control of the weights of classes;
  - By implementing IRACS in a real testbed. This will include the embedding of the *Resource Controller* module into an existing open source router image, such as VyOS;
  - By exploiting header compression in its signalling messages to further reduce its overhead.
- To extend the QoS mechanisms to the ISP access network. With the network equipment virtualization, the typical residential gateway (typically a layer 3 device) is being replaced by a layer 2 device (e.g., ONT). Network functionalities, such as QoS control and firewall configuration would be the responsibility of a virtualized broadband access server, running at the ISPs edge router. This will allow the ISP to easy deploy firewall or QoS rules based on the customer traffic.
- To improve the *P2P Receiver&Forwarder* behaviour by making it perform *cut-through* switching of chunks, instead of *store-and-forward*. This will allow to significantly optimize the delay per forwarding peer, reducing the cumulative delay for peers at the bottom of a particular distribution tree.
- To provide a solution for social watching - consumers in a specific group should be able to watch the content at the same time. Distributing the content through an IP network (multicast plus P2P) means different users will get the video streams at slightly different times. Some contents, such as a football game is best experienced by a group of people if viewed at the same time. Research on how to optimize the buffering delay in order to

achieve synchronization amongst multiple peers would be a nice add-on to the proposed architecture.

Bibliography

[1] Alexa - The Web Information Company, “The top 500 sites on the web”. [online] Available: http://www.alexa.com/topsites

[2] Cisco Systems, “Cisco Visual Networking Index: Forecast and Methodology”, 2011-2016. [online]. Available: http://ec.europa.eu/newsroom/dae/document.cfm?doc_id=2037

[3] Cisco Systems, “Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015–2020”. [online]. Available: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-forecast-qa.pdf

[4] Sandvine, “Global Internet Phenomena Report, Spring 2011”, 2011. [online]. Available: http://www.sandvine.com/news/globalbroadbandtrends.asp

[5] Deering, S., "Host extensions for IP multicasting", STD 5, RFC 1112, August 1989 [online]. Available :http://www.rfc-editor.org/info/rfc1112 doi : 10.17487/RFC1112

[6] ITU-T and ISO/IEC JTC 1, "H.264: Advanced video coding for generic audiovisual services", ITU-T Recommendation H.264 (10/16) and ISO/IEC 14496-10 (MPEG-4 AVC), 2016.

[7] J. Ostermann et al., "Video coding with H.264/AVC: tools, performance, and complexity," in IEEE Circuits and Systems Magazine, vol. 4, no. 1, pp. 7-28, First Quarter 2004. doi: 10.1109/MCAS.2004.1286980

[8] ISO/IEC 14496–10:2003, “Coding of Audiovisual Objects—Part 10: Advanced Video Coding,” 2003, also ITU-T Recommendation H.264 “Advanced video coding for generic audiovisual services.”

[9] H. S. Malvar, A. Hallapuro, M. Karczewicz and L. Kerofsky, "Low-complexity transform and quantization in H.264/AVC," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 598-603, July 2003. doi: 10.1109/TCSVT.2003.814964.

[10] D. Marpe, H. Schwarz and T. Wiegand, "Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard," in IEEE Transactions on Circuits and

Systems for Video Technology, vol. 13, no. 7, pp. 620-636, July 2003.
doi: 10.1109/TCSVT.2003.815173.

[11] P. List, A. Joch, J. Lainema, G. Bjontegaard, and M. Karczewicz. 2003. Adaptive deblocking filter. IEEE Trans. Cir. and Sys. for Video Technol. 13, 7 (July 2003), 614-619. DOI=http://dx.doi.org/10.1109/TCSVT.2003.815175.

[12] H. Schwarz, D. Marpe and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 17, no. 9, pp. 1103-1120, Sept. 2007.
doi: 10.1109/TCSVT.2007.905532.

[13] J. Chakareski, "Adaptive multiview video streaming: challenges and opportunities," in IEEE Communications Magazine, vol. 51, no. 5, pp. 94-100, May 2013.
doi: 10.1109/MCOM.2013.6515052

[14] P. Merkle, A. Smolic, K. Muller and T. Wiegand, "Efficient Prediction Structures for Multiview Video Coding," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 17, no. 11, pp. 1461-1473, Nov. 2007.
doi: 10.1109/TCSVT.2007.903665.

[15] M. K. Kang and K. J. Yoon, "Depth-Discrepancy-Compensated Inter-Prediction With Adaptive Segment Management for Multiview Depth Video Coding," in IEEE Transactions on Multimedia, vol. 16, no. 6, pp. 1563-1573, Oct. 2014.
doi: 10.1109/TMM.2014.2323939

[16] Y. Chen, M. M. Hannuksela, T. Suzuki, and S. Hattori, “Overview of the MVC+D 3D video coding standard”, J. Vis. Comun. Image Represent. 25, 4 (May 2014), 679-688.
doi: http://dx.doi.org/10.1016/j.jvcir.2013.03.013.

[17] J.-R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, “Comparison of the coding efficiency of video coding standards— Including high efficiency video coding (HEVC),” IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, pp. 1669–1684, Dec. 2012.

[18] K. Müller et al., "3D High-Efficiency Video Coding for Multi-View Video and Depth Data," in IEEE Transactions on Image Processing, vol. 22, no. 9, pp. 3366-3378, Sept. 2013. doi: 10.1109/TIP.2013.2264820

[19] H. Schulzrinne, A. Rao, R. Lanphier, M. Westerlund, and Ed. Stiemerling, "Real-Time Streaming Protocol Version 2.0", RFC 7826, December 2016. [online].
Available: http://www.rfc-editor.org/info/rfc7826
doi: 10.17487/RFC7826

[20] B. Wang, J. Kurose, P. Shenoy, D. Towsley, “Multimedia Streaming via TCP: An Analytic Performance Study,” ACM Transactions on Multimedia Computing Communications and Applications (TOMCCAP), Vol. 4, No. 2 (Apr. 2008), pp. 16:1–22

[21] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003. [online]. Available: http://www.rfc-editor.org/info/rfc3550 doi: 10.17487/RFC3550

[22] S. Ihm and V. S. Pai. 2011. Towards understanding modern web traffic. In Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference (IMC '11). ACM, New York, NY, USA, 295-312. doi: 10.1145/2068816.2068845

[23] A. Rao, A. Legout, Y. Lim, D. Towsley, C. Barakat, and W. Dabbous. 2011. Network characteristics of video streaming traffic. In Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies (CoNEXT '11). ACM, New York, NY, USA, , Article 25 , 12 pages. doi: 10.1145/2079296.2079321

[24] van Brandenburg, R., van Deventer, O., Le Faucheur, F., and K. Leung, "Models for HTTP-Adaptive-Streaming-Aware Content Distribution Network Interconnection (CDNI)", RFC 6983, July 2013. [online]. Available: http://www.rfc-editor.org/info/rfc6983 doi: 10.17487/RFC6983

[25] S. Akhshabi, A. C. Begen, and C. Dovrolis, “An experimental evaluation of rate-adaptation algorithms in adaptive streaming over HTTP”, in Proceedings of the second annual ACM conference on Multimedia systems (MMSys '11). ACM, New York, NY, USA, 157-168. doi: 10.1145/1943552.1943574.

[26] J. F. Kurose and K. W. Ross, “Computer Networking: A Top-Down Approach (6th Edition)”, Pearson. 2012

[27] S. Li, C. Hou, L. Yu, C. Zhu, J. Lei, “Pixel-Based Inter Prediction in Coded Texture Assisted Depth Coding”, Signal Processing Letters, IEEE, vol. 21, no. 1, pp. 74-78, 2014

[28] A. Vetro, T. Wiegand and G. J. Sullivan, "Overview of the Stereo and Multiview Video Coding Extensions of the H.264/MPEG-4 AVC Standard," in Proceedings of the IEEE, vol. 99, no. 4, pp. 626-642, April 2011. doi: 10.1109/JPROC.2010.2098830

[29] H. Schwarz, D. Marpe, and T. Wiegand, “Overview of the Scalable Video Coding Extension of the H.264/AVC Standard”, IEEE Transactions on Circuits and Systems for

Video Technology, Special Issue on Scalable Video Coding, vol. 17, no. 9, pp. 1103-1120, September 2007.

[30] MUSCADE MUltimedia SCAlable 3D for Europe, large scale integrated project funded by the European Union Framework Program 7. [online]. Available: http://www.muscade.eu

[31] ROMEO Remote Collaborative Real-Time Multimedia Experience over the Future Internet, large scale integrated project funded by the European Union Framework Program 7. [online]. Availabe: http://www.ict-romeo.eu/

[32] D. Qiu and R. Srikant, “Modeling and performance analysis of BitTorrent-like peer-to-peer networks”, SIGCOMM Comput. Commun. Rev. 34, 4 (August 2004), 367-378.
doi: 10.1145/1030194.1015508

[33] S. James and P. Crowley, "IMP: ISP-Managed P2P," 2010 IEEE Tenth International Conference on Peer-to-Peer Computing (P2P), Delft, 2010, pp. 1-9.
doi: 10.1109/P2P.2010.5569959

[34] Torrentfreak website, “Comcast can block BitTorrent again Court rules”, April 2010. [online]. Available: https://torrentfreak.com/comcast-can-block-bittorrent-again-court-rules-100406/

[35] T. Leighton, “Improving performance on the internet”, Commun. ACM 52, 2, 44-51, February 2009.
doi: 10.1145/1461928.1461944.

[36] V. K. Adhikari et al., "Measurement Study of Netflix, Hulu, and a Tale of Three CDNs," in IEEE/ACM Transactions on Networking, vol. 23, no. 6, pp. 1984-1997, Dec. 2015.
doi: 10.1109/TNET.2014.2354262

[37] C. Huang, A. Wang, J. Li, and K. W. Ross, “Understanding hybrid CDN-P2P: why limelight needs its own Red Swoosh”, In Proceedings of the 18th International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV '08). ACM, New York, NY, USA, 75-80.
doi: http://dx.doi.org/10.1145/1496046.1496064.

[38] Neustar website. [online]. Available:www.neustar.biz/

[39] MaxMind website. [online]. Available: www.maxmind.com/

[40] P. Vixie, “What DNS Is Not”, Queue 7, 10, Pages 10 (November 2009), 6 pages.
doi: http://dx.doi.org/10.1145/1647300.1647302.

[41] McPherson, D., Oran, D., Thaler, D., and E. Osterweil, "Architectural Considerations of IP Anycast", RFC 7094, January 2014. [online].

Available:http://www.rfc-editor.org/info/rfc7094/
doi: 10.17487/RFC7094

[42] R. Buyya, A. m. K. Pathan, J. Broberg and Z. Tari, "A Case for Peering of Content Delivery Networks," in IEEE Distributed Systems Online, vol. 7, no. 10, pp. 3-3, Oct. 2006.
doi: 10.1109/MDSO.2006.57

[43] Y. Jin, Y. Wen and K. Guan, "Toward Cost-Efficient Content Placement in Media Cloud: Modeling and Analysis," in IEEE Transactions on Multimedia, vol. 18, no. 5, pp. 807-819, May 2016.
doi: 10.1109/TMM.2016.2537199 6

[44] G. Gao, Y. Wen, W. Zhang and H. Hu, "Cost-efficient and QoS-aware content management in media cloud: Implementation and evaluation," 2015 IEEE International Conference on Communications (ICC), London, 2015, pp. 6880-6886.
doi: 10.1109/ICC.2015.7249422

[45] S. Scellato, C. Mascolo, M. Musolesi, and J. Crowcroft, “Track globally, deliver locally: improving content delivery networks by tracking geographic social cascades”, in Proceedings of the 20th international conference on World wide web (WWW '11). ACM, New York, NY, USA, 457-466.
doi: 10.1145/1963405.1963471.

[46] A. Passarella, “Review: A survey on content-centric technologies for the current Internet: CDN and P2P solutions”, Comput. Commun. 35, 1 (January 2012), 1-32.
doi: 10.1016/j.comcom.2011.10.005

[47] X. Liu, F. Dobrian, H. Milner, J. Jiang, V. Sekar, I. Stoica, and H. Zhang, “A case for a coordinated internet video control plane”, in Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication (SIGCOMM '12). ACM, New York, NY, USA, 359-370.
doi: 10.1145/2342356.2342431.

[48] P. Wendell and M. J. Freedman, “Going viral: flash crowds in an open CDN”, in Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference (IMC '11). ACM, New York, NY, USA, 549-558.
doi: 10.1145/2068816.2068867

[49] R. S. Peterson and E. Gün Sirer, “Antfarm: efficient content distribution with managed swarms”, in Proceedings of the 6th USENIX symposium on Networked systems design and implementation (NSDI'09). USENIX Association, Berkeley, CA, USA, 107-122.

[50] M. Zhao, P. Aditya, A. Chen, Y. Lin, A. Haeberlen, P. Druschel, B. Maggs, B. Wishon, and M. Ponec, “Peer-assisted content distribution in akamai netsession”, in Proceedings of the 2013 conference on Internet measurement conference, pages 31–42. ACM, 2013.

[51] R. Roverso, A. Naiem, M. Reda, M. El-Beltagy, S. El-Ansary, N. Franzen, and S. Haridi, “On the feasibility of centrally coordinated peer-to-peer live streaming”, in Consumer Communications and Networking Conference (CCNC), 2011 IEEE, pages 1061–1065. IEEE, 2011

[52] D. Karamshuk, N. Sastry, A. Secker and J. Chandaria, "ISP-friendly peer-assisted on-demand streaming of long duration content in BBC iPlayer," 2015 IEEE Conference on Computer Communications (INFOCOM), Kowloon, 2015, pp. 289-297. doi: 10.1109/INFOCOM.2015.7218393.

[53] A. Balachandran, V. Sekar, A. Akella, and S. Seshan, “Analyzing the potential benefits of cdn augmentation strategies for internet video workloads”, in Proceedings of the 2013 conference on Internet measurement conference, pages 43–56. ACM, 2013.

[54] C. Huang, J. Li, and K. W. Ross, “Can internet video-on-demand be profitable?”, ACM SIGCOMM Computer Communication Review, 37(4):133–144, 2007.

[55] G. Zhang, W. Liu, X. Hei and W. Cheng, "Unreeling Xunlei Kankan: Understanding Hybrid CDN-P2P Video-on-Demand Streaming," in IEEE Transactions on Multimedia, vol. 17, no. 2, pp. 229-242, Feb. 2015. doi: 10.1109/TMM.2014.2383617

[56] M. Zink, Kyoungwon Suh, Y. Gu, and J. Kurose. 2009, “Characteristics of YouTube network traffic at a campus network - Measurements, models, and implications”, Comput. Netw. 53, 4 (March 2009), 501-514. doi: http://dx.doi.org/10.1016/j.comnet.2008.09.022

[57] Z. H. Lu, X. H. Gao, S. J. Huang and Y. Huang, "Scalable and Reliable Live Streaming Service through Coordinating CDN and P2P," 2011 IEEE 17th International Conference on Parallel and Distributed Systems, Tainan, 2011, pp. 581-588. doi: 10.1109/ICPADS.2011.113

[58] M. Goldmann and G. Kreitz, "Measurements on the spotify peer-assisted music-on-demand streaming system," 2011 IEEE International Conference on Peer-to-Peer Computing, Kyoto, 2011, pp. 206-211. doi: 10.1109/P2P.2011.6038737

[59] X. Hei, Y. Liu and K. W. Ross, "IPTV over P2P streaming networks: the mesh-pull approach," in IEEE Communications Magazine, vol. 46, no. 2, pp. 86-92, February 2008. doi: 10.1109/MCOM.2008.4473088

[60] J. Liu, S. G. Rao, B. Li and H. Zhang, "Opportunities and Challenges of Peer-to-Peer Internet Video Broadcast," in Proceedings of the IEEE, vol. 96, no. 1, pp. 11-24, Jan. 2008. doi: 10.1109/JPROC.2007.909921

[61] Y. Liu, Y. Guo, and C. Liang, “A survey on peer-to-peer video streaming systems”, Peer-to-Peer Netw. Appl. (2008) 1: 18.
doi:10.1007/s12083-007-0006-y

[62] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, “Scalable application layer multicast”, in Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '02). ACM, New York, NY, USA, 205-217.
doi: http://dx.doi.org/10.1145/633025.633045

[63] M. Castro, P. Druschel, A. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, “SplitStream: high-bandwidth multicast in cooperative environments”, in Proceedings of the nineteenth ACM symposium on Operating systems principles (SOSP '03). ACM, New York, NY, USA, 298-313.
doi: 10.1145/945445.945474.

[64] D. Kostić, A. Rodriguez, J. Albrecht, and A. Vahdat, “Bullet: high bandwidth data dissemination using an overlay mesh”, in Proceedings of the nineteenth ACM symposium on Operating systems principles (SOSP '03). ACM, New York, NY, USA, 282-297.
doi: 10.1145/945445.945473.

[65] Y. Chu, S. G. Rao, and H. Zhang, “A case for end system multicast (keynote address). In Proceedings of the 2000 ACM SIGMETRICS international conference on Measurement and modeling of computer systems (SIGMETRICS '00). ACM, New York, NY, USA, 1-12.
doi: 10.1145/339331.339337

[66] X. Zhang, J. Liu, B. Li and Y. S. P. Yum, "CoolStreaming/DONet: a data-driven overlay network for peer-to-peer live media streaming," Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies., 2005, pp. 2102-2111 vol. 3.
doi: 10.1109/INFCOM.2005.1498486

[67] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, “Scalable application layer multicast”, in Proceedings of the 2002 conference on Applications, technologies, architectures, and

protocols for computer communications (SIGCOMM '02). ACM, New York, NY, USA, 205-217.
doi: http://dx.doi.org/10.1145/633025.633045

[68] V. N. Padmanabhan, H. J. Wang and P. A. Chou, "Resilient peer-to-peer streaming," 11th IEEE International Conference on Network Protocols, 2003. Proceedings., 2003, pp. 16-27.
doi: 10.1109/ICNP.2003.1249753

[69] Padmanabhan VN, Wang HJ, Chou PA, Sripanidkulchai K, “Distributing streaming media content using cooperative networking”, Proceedings of the 12th international workshop on network and operating systems support for digital audio and video. Association for Computing Machinery, Miami, FL, U.S.A., pp 177–186

[70] N. Ramzan, H. Park, E. Izquierdo, “Video streaming over P2P networks: Challenges and opportunities”, Signal Processing: Image Communication, Volume 27, Issue 5, May 2012, Pages 401-411, ISSN 0923-5965.
doi: 10.1016/j.image.2012.02.004

[71] M. Castro, P. Druschel, A. M. Kermarrec and A. I. T. Rowstron, "Scribe: a large-scale and decentralized application-level multicast infrastructure," in IEEE Journal on Selected Areas in Communications, vol. 20, no. 8, pp. 1489-1499, Oct 2002.
doi: 10.1109/JSAC.2002.803069.

[72] V. Venkataraman, K. Yoshida and P. Francis, "Chunkyspread: Heterogeneous Unstructured Tree-Based Peer-to-Peer Multicast," Proceedings of the 2006 IEEE International Conference on Network Protocols, Santa Barbara, CA, 2006, pp. 2-11.
doi: 10.1109/ICNP.2006.320193

[73] Y. Guo, C. Liang, Y. Liu, “AQCS: Adaptive Queue-Based Chunk Scheduling for P2P Live Streaming”, in NETWORKING 2008 Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet: 7th International IFIP-TC6 Networking Conference Singapore, May 5-9, 2008. Springer, Berlin, pp 433–444

[74] Z. Liu, Y. Shen, K. W. Ross, S. S. Panwar and Y. Wang, "LayerP2P: Using Layered Video Chunks in P2P Live Streaming," in IEEE Transactions on Multimedia, vol. 11, no. 7, pp. 1340-1352, Nov. 2009.
doi: 10.1109/TMM.2009.2030656

[75] M. K. H. Yeung and Y. K. Kwok, "Game Theoretic Peer Selection for Resilient Peer-to-Peer Media Streaming Systems," 2008 The 28th International Conference on Distributed Computing Systems, Beijing, 2008, pp. 817-824.
doi: 10.1109/ICDCS.2008.69

[76] J. F. Paris and P. Shah, "Peer-to-Peer Multimedia Streaming Using BitTorrent," 2007 IEEE International Performance, Computing, and Communications Conference, New Orleans, LA, 2007, pp. 340-347.
doi: 10.1109/PCCC.2007.358912

[77] A. Vlavianos, M. Iliofotou and M. Faloutsos, "BiToS: Enhancing BitTorrent for Supporting Streaming Applications," Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications, Barcelona, 2006, pp. 1-6.
doi: 10.1109/INFOCOM.2006.43

[78] M. Zhang, L. Sun, X. Xi and S. Yang, "iGridMedia: Providing Delay-Guaranteed Peer-to-Peer Live Streaming Service on Internet," IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference, New Orleans, LO, 2008, pp. 1-5.
doi: 10.1109/GLOCOM.2008.ECP.337

[79] ITU-T, “Y.2001: General overview of NGN”, ITU Recommendation Y.2001, 2004.

[80] Cisco Systems, “Enterprise QoS Solution Reference Network Design Guide”. [online]. Available:http://www.cisco.com/en/US/docs/solutions/Enterprise/WAN_and_MAN/QoS_SRND/QoSIntro.html/

[81] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, December 1998. [online].
Available: http://www.rfc-editor.org/info/rfc2475
doi: 10.17487/RFC2475

[82] Heinanen, J., Baker, F., Weiss, W., and J. Wroclawski, "Assured Forwarding PHB Group", RFC 2597, June 1999. [online]. Available: http://www.rfc-editor.org/info/rfc2597
doi: 10.17487/RFC2597

[83] Davie, B., Charny, A., Bennet, J., Benson, K., Le Boudec, J., Courtney, W., Davari, S., Firoiu, V., and D. Stiliadis, "An Expedited Forwarding PHB (Per-Hop Behavior)", RFC 3246, March 2002. [online]. Available: http://www.rfc-editor.org/info/rfc3246
doi: 10.17487/RFC3246

[84] Braden, R., Ed., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, September 1997. [online]. Available: http://www.rfc-editor.org/info/rfc2205
doi: 10.17487/RFC2205

[85] C. Yun and H. Perros, “QoS Control for NGN: A Survey of Techniques”, Journal of Network and Systems Management, Volume 18, Issue 4, pp 447–461, Springer, 2010.
doi:10.1007/s10922-010-9157-x

[86] Rand, D., "PPP Reliable Transmission", RFC 1663, July 1994. [online]. Available: http://www.rfc-editor.org/info/rfc1663. doi: 10.17487/RFC1663

[87] S. Floyd and V. Jacobson; “The synchronization of periodic routing messages”, IEEE/ACM Trans. Netw. 2, 2, 122-136, April 1994. doi: 10.1109/90.298431

[88] Manner, J. and X. Fu, "Analysis of Existing Quality-of-Service Signaling Protocols", RFC 4094, May 2005. [online]. Avilable: http://www.rfc-editor.org/info/rfc4094 doi: 10.17487/RFC4094

[89] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998. [online]. Available: http://www.rfc-editor.org/info/rfc2328 doi: 10.17487/RFC2328

[90] Baker, F., Iturralde, C., Le Faucheur, F., and B. Davie, "Aggregation of RSVP for IPv4 and IPv6 Reservations", RFC 3175, September 2001. [online]. Available: http://www.rfc-editor.org/info/rfc3175 doi: 10.17487/RFC3175

[91] A. Neto, E. Cerqueira, M. Curado, E. Monteiro and P. Mendes, "Scalable Resource Provisioning for Multi-User Communications in Next Generation Networks," IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference, New Orleans, LO, 2008, pp. 1-6. doi: 10.1109/GLOCOM.2008.ECP.453

[92] R. Prior, S. Sargento, “Scalable Reservation-Based QoS Architecture — SRBQ”, in Encyclopedia of Internet Technologies and Applications, 2008. doi: 10.4018/978-1-59140-993-9.ch067

[93] ITU, “G.1081: Performance monitoring points for IPTV”, ITU-T Recommendation G.1081, ITU-T Study Group 12, 2008.

[94] S. R. Lima and P. Carvalho, "Enabling self-adaptive QoE/QoS control," 2011 IEEE 36th Conference on Local Computer Networks, Bonn, 2011, pp. 239-242. doi: 10.1109/LCN.2011.6115198

[95] K. M. Salehin and R. Rojas-Cessa, "Combined methodology for measurement of available bandwidth and link capacity in wired packet networks," in IET Communications, vol. 4, no. 2, pp. 240-252, January 22 2010. doi: 10.1049/iet-com.2009.0249

[96] P. P. Pan, E. L. Hahne and H. Schulzrinne, "BGRP: Sink-tree-based aggregation for inter-domain reservations," in Journal of Communications and Networks, vol. 2, no. 2, pp. 157-167, June 2000.
doi: 10.1109/JCN.2000.6596736

[97] P. Pinto, A. Santos, P. Amaral and L. Bernardo, "SIDSP: Simple Inter-Domain QoS Signaling Protocol," MILCOM 2007 - IEEE Military Communications Conference, Orlando, FL, USA, 2007, pp. 1-7.
doi: 10.1109/MILCOM.2007.4454767

[98] R. Bless, “Dynamic Aggregation of Reservations for Internet Services”, in Proceedings of 10th Int. Conf. on Telecommunication. Systems – Modeling and Analysis (ICTSM’10), 1, 26-38.
doi:10.1023/B:TELS.0000029019.73186.aa

[99] R. Sofia, R. Guerin and P. Veiga, "SICAP, a shared-segment inter-domain control aggregation protocol," Workshop on High Performance Switching and Routing, 2003, HPSR., 2003, pp. 73-78.
doi: 10.1109/HPSR.2003.1226683

[100] E. Logota, S. Sargento, A. Neto, “Um Método para Controlo Avançado de Sobre-reservas Baseado em Classes de Serviço e Sistema para a sua Execução” - A Method and apparatus for Advanced Class-based Bandwidth Over-reservation Control -, Patent 105305, 2010

[101] E. Logota, A. Neto and S. Sargento, "A New Strategy for Efficient Decentralized Network Control," 2010 IEEE Global Telecommunications Conference GLOBECOM 2010, Miami, FL, 2010, pp. 1-6.
doi: 10.1109/GLOCOM.2010.5683572

[102] C. Campos, “Sobre-Reserva em Redes com Controlo Centralizado e Distribuído”, MSc Thesis, Universidade de Aveiro, 2010.

[103] E. Logota, C. Campos, S. Sargento, and A. Neto. 2013. Advanced multicast class-based bandwidth over-provisioning. Comput. Netw. 57, 9 (June 2013), 2075-2092.
doi: 10.1016/j.comnet.2013.04.009

[104] E. Logota, C. Campos, S. Sargento and A. Neto, "Scalable resource and admission management in class-based networks," 2013 IEEE International Conference on Communications Workshops (ICC), Budapest, 2013, pp. 1095-1100.
doi: 10.1109/ICCW.2013.6649400

[105] K. Birkos, C. Papageorgiou, T. Dagiuklas and S. Kotsopoulos, "A failover mechanism for peer-to-peer streaming over multiple multicast trees," 2013 IEEE International Conference

on Communications Workshops (ICC), Budapest, 2013, pp. 672-676.
doi: 10.1109/ICCW.2013.6649318

[106] K. Birkos, F. Andriopoulou, C. Papageorgiou, T. Dagiuklas and S. Kotsopoulos, "Enhanced failover mechanisms for tree-based peer-to-peer streaming," 2015 IEEE International Conference on Communications (ICC), London, 2015, pp. 7024-7029.
doi: 10.1109/ICC.2015.7249446

[107] M. Vilas, X.-G. Paneda, R. Garcia, D. Melendi, and V.-G. Garcia, “User behavior analysis of a video-on-demand service with a wide variety of subjects and lengths,” in Proceedings of the 31st EUROMICRO Conference on Software Engineering and Advanced Applications, 2005, pp. 330–337.

[108] D. Manzato and N. Fonseca, “A survey of channel switching schemes for IPTV”, Communications Magazine, IEEE , vol.51, no.8, pp.120,127, August 2013,
doi: 10.1109/MCOM.2013.6576349

[109] H. Yu, D. Zheng, B. Y. Zhao, and W. Zheng, “Understanding user behavior in large-scale video-on-demand systems”, in Proceedings of the 1st ACM SIGOPS/EuroSys European Conference on Computer Systems 2006 (EuroSys '06). ACM, New York, NY, USA, 333-344.
doi: 10.1145/1217935.1217968

[110] B. Chang, L. Dai, Y. Cui, and Y. Xue, “On feasibility of P2P on-demand streaming via empirical VoD user behavior analysis,” in Proceedings of the 28th International Conference on Distributed Computing Systems, 2008, pp. 7–11

[111] The libpcap project. [online]. Available: http://sourceforge.net/projects/libpcap/

[112] E. Logota, H. Marques and J. Rodriguez, "A cross-layer resource over-provisioning architecture for P2P networks," 2013 18th International Conference on Digital Signal Processing (DSP), Fira, 2013, pp. 1-6.
doi: 10.1109/ICDSP.2013.6622787

[113] Hancock, R., Karagiannis, G., Loughney, J., and S. Van den Bosch, "Next Steps in Signaling (NSIS): Framework", RFC 4080, June 2005. [online].
Available: http://www.rfc-editor.org/info/rfc4080
doi: 10.17487/RFC4080

[114] E. Logota, A. Neto and S. Sargento, "COR: An efficient Class-based resource Over-pRovisioning mechanism for future networks," The IEEE symposium on Computers and Communications, Riccione, Italy, 2010, pp. 298-301.
doi: 10.1109/ISCC.2010.5546524

[115] Katz, D., "IP Router Alert Option", RFC 2113, February 1997. [online]. Available: http://www.rfc-editor.org/info/rfc2113 doi: 10.17487/RFC2113

[116] A. Demers, S. Keshav, and S. Shenker. 1989. Analysis and simulation of a fair queueing algorithm. In Symposium proceedings on Communications architectures & protocols (SIGCOMM '89). ACM, New York, NY, USA, 1-12. doi: 10.1145/75246.75248

[117] S. J. Golestani, "A self-clocked fair queueing scheme for broadband applications," INFOCOM '94. Networking for Global Communications., 13th Proceedings IEEE, Toronto, Ont., 1994, pp. 636-646 vol.2. doi: 10.1109/INFCOM.1994.337677

[118] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006. [online]. Available: http://www.rfc-editor.org/info/rfc4271 doi: 10.17487/RFC4271

[119] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, December 1998. [online]. Available: http://www.rfc-editor.org/info/rfc2475 doi: 10.17487/RFC2475

[120] Manner, J., Karagiannis, G., and A. McDonald, "NSIS Signaling Layer Protocol (NSLP) for Quality-of-Service Signaling", RFC 5974, October 2010. [online]. Available: http://www.rfc-editor.org/info/rfc5974 doi: 10.17487/RFC5974

[121] Vasseur, J.-P., Ed., Ali, Z., and S. Sivabalan, "Definition of a Record Route Object (RRO) Node-Id Sub-Object", RFC 4561, June 2006. [online]. Available: <http://www.rfc-editor.org/info/rfc4561 doi: 10.17487/RFC4561.

[122] Braden, R., Ed., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, September 1997. [online].Available: http://www.rfc-editor.org/info/rfc2205 doi: 10.17487/RFC2205

[123] The Network Simulator , ns-2.31. [online]. Available: http://nsnam.sourceforge.net/wiki/index.php/User_Information

[124] Babiarz, J., Chan, K., and F. Baker, "Configuration Guidelines for DiffServ Service Classes", RFC 4594, August 2006. [online].

Available: http://www.rfc-editor.org/info/rfc4594
doi: 10.17487/RFC4594

[125] MathWorks: Matlab. [online]. Available: http://www.mathworks.com/products/matlab/.

[126] ITU-R, “BT.500: Methodology for the subjective assessment of the quality of television pictures”, ITU-R Recommendation BT.500, 2012.

[127] ITU-T, "J.241: Quality of service ranking and measurement methods for digital video services delivered over broadband IP networks", ITU-T Recommendation J.241, 2005.

[128] ITU-T, "Y.1541 Network performance objectives for IP-based services", ITU-T Recommendation Y.1541, 2011;

[129] ITU-T, "Y. 1540: Internet protocol data communication service - IP packet transfer and availability performance parameters", ITU-T Recommendation Y.1540, 2011;

[130] ITU-T, “G.826: End-to-end error performance parameters and objectives for international, constant bit-rate digital paths and connections”, ITU-T Recommendation G.826, 2012

[131] The iperf tool. [online]. Available at: https://iperf.fr/

[132] Akamai, “EMEA HIGHLIGHTS (Q3 2015)”, [online].
Available: https://www.stateoftheinternet.com/downloads/pdfs/2015-q2-state-of-the-internet-report-infographic-emea.pdf

[133] VyOS – Linux based network operating system. [online]. Available at: https://vyos.io/

Annex 1 - CPU Hierarchy

This Annex provides the table with the CPU rankings used for the calculation of the *peer stability*, a metric described in Chapter 3.2.3. The table groups CPUs with roughly similar gaming performance into tiers based on the average performance each CPU achieved in the reference site *tom'sHardware* (http://www.tomshardware.com/).

Value	CPU Type	
	Intel	AMD
8	Core i7-3770, -3770K, -3820, -3930K, -3960X, -3970X, -4770, -4771, -4790, -4770K, -4790K, -4820K, -4930K, -4960X, -5775C, -5820K, 5930K, -5960X, -6700K, -6700, -6800K, -6850K, -6900K, -6950X Core i5-6600K, -6600, -6500, -5675C, -4690K, 4670K, -4590, -4670, -4570, -4460, -4440, -4430, -3570K, -3570, -3550	---
7	Core i7-2600, -2600K, -2700K, -965, -975 Extreme, -980X Extreme, -990X Extreme Core i5-3470, -3450P, -3450, -3350P, -3330, 2550K, -2500K, -2500, -2450P, -2400, -2380P, -2320, -2310, -2300	FX-9590, 9370, 8370, 8350, 8320, 8300, 8150
6	Core i7-980, -970, -960 Core i7-870, -875K Core i3-4370, -4360, -4350, -4340, -4170, -4160, -4150, -4130, -3250, -3245, -3240, -3225, -3220, -3210, -2100, -2105, -2120, -2125, -2130	FX-6350, 4350 Phenom II X6 1100T BE, 1090T BE Phenom II X4 Black Edition 980, 975
5	Core i7-860, -920, -930, -940, -950 Core i5-3220T, -750, -760, -2405S, -2400S Core 2 Extreme QX9775, QX9770, QX9650 Core 2 Quad Q9650	FX-8120, 8320e, 8370e, 6200, 6300, 4170, 4300 Phenom II X6 1075T Phenom II X4 Black Edition 970, 965, 955 A10-6800K, 6790K, 6700, 5800K, -5700, -7700K, -7800, -7850K, 7870K A8-3850, -3870K, -5600K, 6600K, -7600, -7650K Athlon X4 651K, 645, 641, 640, 740, 750K, 860K
4	Core 2 Extreme QX6850, QX6800 Core 2 Quad Q9550, Q9450, Q9400 Core i5-650, -655K, -660, -661, -670, -680 Core i3-2100T, -2120T	FX-6100, -4100, -4130 Phenom II X6 1055T, 1045T Phenom II X4 945, 940, 920 Phenom II X3 Black Edition 720, 740 A8-5500, 6500 A6-3650, -3670K, -7400K Athlon II X4 635, 630
3	Core 2 Extreme QX6700 Core 2 Quad Q6700, Q9300, Q8400, Q6600, Q8300 Core 2 Duo E8600, E8500, E8400, E7600 Core i3 -530, -540, -550 Pentium G3470, G3460, G3450, G3440, G3430, G3420, G3260, G3258, G3250, G3220, G3420, G3430, G2130, G2120, G2020, G2010, G870, G860, G850, G840, G645, G640, G630	Phenom II X4 910, 910e, 810 Athlon II X4 620, 631 Athlon II X3 460

2	Core 2 Extreme X6800 Core 2 Quad Q8200 Core 2 Duo E8300, E8200, E8190, E7500, E7400, E6850, E6750 Pentium G620 Celeron G1630, G1620, G1610, G555, G550, G540, G530	Phenom II X4 905e, 805 Phenom II X3 710, 705e Phenom II X2 565 BE, 560 BE, 555 BE, 550 BE, 545 Phenom X4 9950 Athlon II X3 455, 450, 445, 440, 435, 425
1	Core 2 Duo E7200, E6550, E7300, E6540, E6700 Pentium Dual-Core E5700, E5800, E6300, E6500, E6600, E6700 Pentium G9650	Phenom X4 9850, 9750, 9650, 9600 Phenom X3 8850, 8750 Athlon II X2 265, 260, 255, 370K A6-5500K A4-7300, 6400K, 6300, 5400K, 5300, 4400, 4000, 3400, 3300 Athlon 64 X2 6400+