Remote Collaborative Real-Time Multimedia Experience over the Future Internet

ROMEO

Grant Agreement Number: 287896

D7.4

Integration Report

Document description	
Name of document	Integration Report
Abstract	This document provides a record of the progress during the six integration sessions organised in Munich and Istanbul between March 2013 and June 2014.
Document identifier	D7.4
Document class	Deliverable
Version	v1.0
Author(s)	H. Marques, H. Silva, J. Rodriguez, J. Ribeiro (IT) D. Florez Rodriguez (TID) V. Ağ, H. Gokmen, B. Demirtaş, O.Tanık, O. Dincer, H. Keskin, Y. Yildiz (ARC) D. Doyen (TEC) X. Shi (MUL) E. Ekmekcioglu, S. Dogan (US) N. Tizon (VITEC) M. Weitnauer (IRT) J. Lauterjung, H. Weigold (R&S) O. Altunbas, C. Ozkan, E. Cimen, A. Akman (ARG) I. Politis, , A. Lykourgiotis (UP) K. Georgiev (MMS)
QAT team	B. Demirtas (ARC), A. Akman (ARG), E. Ekmekcioglu (US)
Date of creation	11-June-2014
Date of last modification	30-Jun-2014
Status	Final version
Destination	European Commission
WP number	WP7
Dissemination Level	Public
Deliverable Nature	Report

TABLE OF CONTENTS

LIST OF TABLES

1 INTRODUCTION

1.1 Purpose of the Document

This document provides a record of the progress during the six integration sessions organised in Munich and Istanbul between March 2013 and June 2014.
The groups described in Section 2 represent intermediate integration steps on the way to the complete the demonstrator set-up. For each of these groups in Section 2, a detailed list of achievements and improvements is given that allows an assessment of the technical work carried out mainly in project years 2 and 3.

1.2 Scope of the Work

The work described in this document led to the set-up of the complete demonstrator of the ROMEO project. This demonstrator is used as an example that the concept at the basis of the ROMEO project is valid and is workable with state-of-the-art hardware and software implementations. The demonstrator is also used for user trials, which aim to verify that the proposed concepts work successfully.

1.3 Structure of the Document

In Section 2, for seven groups of modules/ sub-modules, which represent the important intermediate steps of the overall integration process, the technical details of the progress during the integration sessions are reported.
Section 3 focuses on the mapping of the described groups to the use cases as described in Deliverable D7.3 ‘Final Trials Plan and Validation Methodologies’. The conclusions are summarised in Section 4.

2 Integration levels

The planning of the integration in WP7 (Integration and Demonstration) started with an overview of all sub-modules and modules under development in WP6 (System Components Development). This overview was reported in Deliverable 7.2 [1] (Pilot Set-up and Evaluation Methodologies). The tables below (Table 1, Table 2, Table 3) provide an overview of the functionalities that were tested in consecutive integration steps.

The numbering of groups has been re-arranged in comparison with Deliverable D7.2 [1], since some groups were merged in the course of the integration steps and the original numbering needed modification.

Platform	Group	Module/ Sub-module
Server		Multi-view video and spatial audio capturing Depth map extraction Saliency map extraction Creation of audio object streams Creation of audio scene description Video Encoding Audio Encoding Transport Stream (TS) Generator
	Group01: Content Generation + Media Encode & TS Generation	
		DVB Transmission
	Group02: Content Generation + Media Encode & TS Generation + DVB Transmission	
		P2P Transmission P2P Overlay Multicast Tree Manager Topology Builder P2P Packetisation Authentication, Registration and Security Encryption User Authentication Key Management Content Integrity
	Group03: P2P Transmission + P2P Packetisation + Authentication, Registration and Security	
	Group04: Group01 + Group03	
		User Generated Content (server side) User Generated Content Registering and Storage User Generated Content Search and Discovery User Generated Content Upload and Download A/V Communication Overlay Server
	Group05: User Generated Content + Authentication, Registration and Security (server side) + A/V Overlay	

Table 1 – Groups of modules/ sub-modules for the server platform

Platform	Group	Module/ Sub-module
Peer		P2P Receiver & Forwarder Topology Controller P2P Chunk Selection P2P Depacketisation Network Monitoring Subsystem Authentication, Registration and Security Decryption User Authentication Key Management Content Integrity
	Group06: P2P Reception + P2P Chunk Selection + P2P Forwarding + Topology Controller + Network Monitoring Subsystem	
	Group07: Group06 + DVB reception + P2P Depacketisation + Authentication, Registration and Security + Synchronisation	
		Synchronisation A/V Communication Overlay Client Collaborative Synchronisation Controller Video Decoding Video Rendering Video Adaptation
	Group08: Synchronisation + Video Decoding + Video Rendering + Video Adaptation	
		Synchronisation A/V Communication Overlay Client Collaborative Synchronisation Controller Audio Decoding Object-based audio scene demultiplexing Audio Rendering
	Group09: Synchronisation + Audio Decoding + Audio Rendering	
	Group10: Group08 + Group09	
		User Generated Content (peer side) User Generated Content Capture User Generated Content Upload / Download User Generated Content Register and Store User Generated Content Search and Discovery User Interface and Control
	Group11: UGC functionalities (terminal side)	

Table 2 – Groups of modules/ sub-modules for the P2P terminal platform

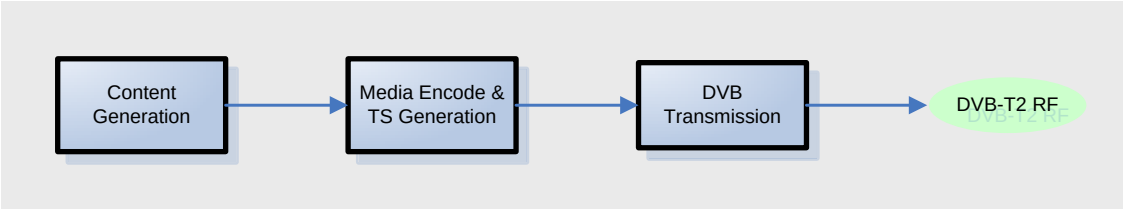
Platform	Group	Module/ Sub-module
Network-related		Mobility Media Aware Proxy Media Independent Handover Proxy Mobile IP
	Group12: P2P Transmission + Mobility + P2P Reception	
		Resource Admission Manager and Resource Controller Virtualisation Platform OLT development DHCP Server development NATIE development Configuration Portal development VSEE development ROMEO QoS enforcement PCRF development QoS platform Integration
	Group13: PCRF+Topology Builder	
Complete demonstrator related	Group14: Network Management Subsystem + P2P Topology Controller + P2P Topology Builder/Multicast Tree Manager	

Table 3 – Groups of modules/ sub-modules for the network-related platform

Most of the development steps for components and sub-modules were completed shortly after the end of the second project-year.

In the following subsections, the status of a subset of the groups shown in Tables 1 – 3 are described in detail. These groups represent the intermediate integration levels and are combined into the final demonstrator set-up. Other groups not described below are either a combination of other groups (e.g., Group04, Group10), or have a lot of integration steps in common as the described groups (e.g., Group01, Group07, Group11). Note that the demonstration of Group13 involving the virtualisation platform is done separately as the final demonstrator.

2.1 Group 02: Content Generation, Media Encode & TS Generation, DVB Transmission



Status after March 2013 integration:

- A third capture session in Paris has been organized in January 2013 providing two new sets of content: a scenario with three musicians (several tracks of around 1.5 minutes) and a scenario with three actors moving on the stage (one shoot of around 5 minutes).
- Post-processing of these two sequences has been performed, including rectifications (geometrical, colour) and disparity map extraction.
- Scalable and multi-description encoding block (i.e., Media Encode – video part) has been completed to produce eight elementary stream files conforming to MPEG 4 Part 10 – Annex B format. Intra and predictive coded frames are included with a Group of Picture size equal to 8 (~0.34 second) and multiple slices (NAL) units are allowed for each frame as per the design constraint depicted in D4.1.
- Post-processing of all sequences for multi-channel (5.1) and multi-object audio has been performed. Multi-channel and object audios are encoded to AAC streams in ADTS format.
- All provided new sets of eight scalable video streams are passed to the TS generation block for multiplexing, including the streams coming from the audio encoder block.
- Extraction of object audio scene description from the audio capture sessions has started.
- The DVB transmission part has been set up and tested with different transmission modes to verify the availability of the bandwidth required for the transmission of all views for test purposes.

Status after July 2013 integration:

- A set of sequences has been selected and the corresponding disparity maps have been extracted to be used in the subsequent integration efforts and demonstration.
- Post-processing to the disparity maps has been applied prior to compression (i.e., disparity map post-processing in WP3). Since the improvement in the quality has been found out to be insufficient, the post-processing stage is skipped.
- Post-processing of object audio scene description has been completed. The extracted scene description corresponding to the integration videos are encoded and ready for TS multiplexing.
- The various generated Transport Streams were successfully tested with the content available from the latest recording session.

Status after October 2013 integration:

- Several sequences from the last shooting session have been selected to be final sequence for the demonstration. They have been adapted to include a global shift to compensate the parallel shooting effect. The convergence plane is defined to ensure a balance between positive and negative disparities.
- These selected contents have been re-compressed (views and disparity maps, after the regeneration of multiple descriptions) through the Media Encoder block and multiplexed in the TS Generation block.
- Multi-channel and multi-object audio content elementary streams as well as object audio scene description streams corresponding to the demonstration are feed to the TS generation module for multiplexing
- The Transport Streams based on the selected material were looped to obtain test stream that were long enough for subjective assessment and system testing.

Status after November 2013 integration:

- Transport Stream generation was modified with time-shifting of video vs. audio to compensate for different delays caused by different buffer sizes in both paths, and were successfully tested.

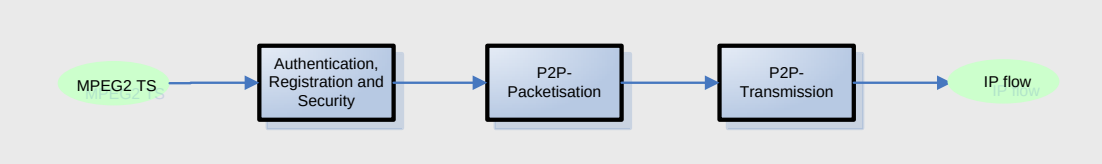
Status after March 2014 integration:

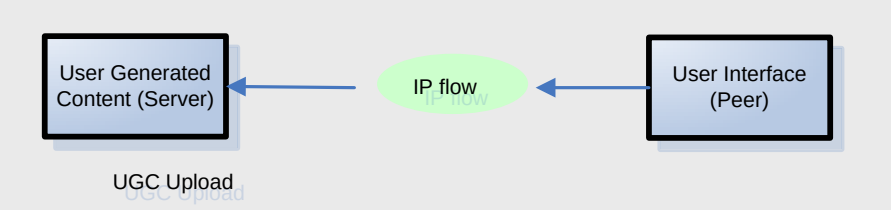
- No further steps were required.

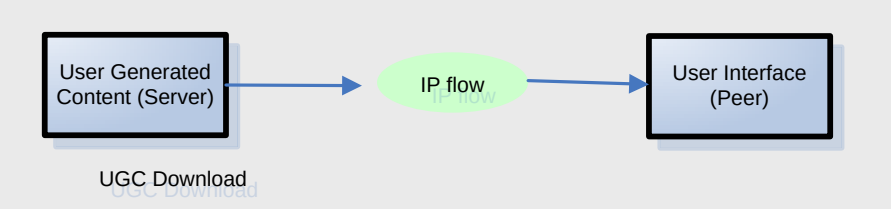
Status after June 2014 integration:

- No further steps were required.

2.2 Group 03 and Group 05: P2P Transmission, P2P Packetisation, Authentication, Registration and Security, A/V Overlay, UGC







Status after March 2013 integration:

- Mainly the integration between the packetisation and transmission components has been focused on.
- P2P chunk generation from the content file including MPEG2-TS packets has been tested.
- Generated chunks have been transmitted to the IP network (two streams only).
- Integration of A/V and UGC functionalities has not been started within the group components.
- One user could successfully connect to the A/V overlay server and stream at 5 fps.
- For the A/V Overlay Server, related partners decided to use the OpenMeetings platform, which is an open source video conferencing and instant messaging tool.
- Connection of one mobile and three portable terminals to the same OpenMeetings server has been achieved.
- Transmission of audio and video from OpenMeetings server to the peers has been tested successfully.

Status after July 2013 integration:

- P2P Transmitter has been updated and tested to transmit all 3D video streams (from Group01) to the network with an average rate of 25 fps.
- P2P Packetisation and P2P Transmitter integration has been completed.
- Content loop capability is added to the P2P Transmitter.

- Integration with Topology Builder component has been performed to get the IP addresses of all top layer peers in the ROMEO topology.
- Integration of the Authentication & Registration and Security component has been left to the following integration sessions.
- Integration of A/V and UGC functionalities has not been started within the group components.
- Two users could connect to the A/V overlay server and stream with 20 fps and audio.

Status after October 2013 integration:

- Integration between all components in this group has been completed. The P2P server has been able to generate chunks, encrypt the data and transmit to the network all streams in the given content.
- Integration bugs with the other ROMEO groups have been detected and fixed.
- Integration with user generated content, A/V overlay and portable/fixed User Interface components have been performed.
- Mobile terminal's User Interface and transcoding service integrations have been left to the following integration sessions.
- Three users could connect to the A/V overlay server in synchrony with the audio.

Status after November 2013 integration:

- Mobile peer stream selection functionality has been added to the P2P Transmitter.
- Integration with user generated content, A/V overlay and portable/fixed and mobile User Interface components have been performed.
- Transcoding service integration has also been performed.
- A collaboration group of three users has been formed by the A/V overlay where all of these users could share sample user generated content between each other. This includes uploading snapshots taken by the A/V application through UGC application to UGC server.
- A/V Overlay Server has been implemented with all the required functionality.
- A/V Overlay Client for all terminals (fixed, portable and mobile) has been implemented with all required functionality.
- Video and audio transmission and Instant Messaging have been tested with two portable clients and one mobile client.

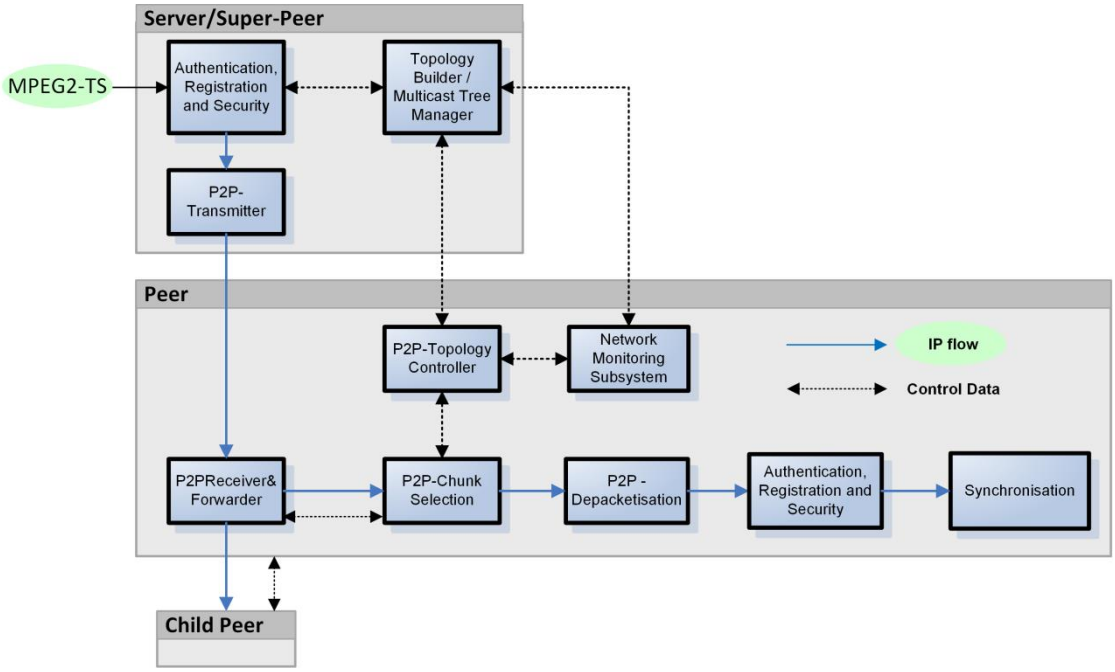
Status after March 2014 integration:

- Integration related bugs with other ROMEO groups have been detected and fixed.
- The same scenario as in the November 2013 meeting with new versions of A/V overlay application and UGC application (bugs fixed, interfaces changes).
- A/V Overlay Server and Client modules have been tested for all types of terminals to verify their stability.

Status after June 2014 integration:

- Stream selection functionality for portable peers has been added to the P2P Transmitter.
- Integration related bugs with the other ROMEO groups have been detected and fixed.
- UGC application has been completed.

2.3 Group 06: P2P Reception, P2P Chunk Selection, P2P Forwarding, Topology Controller, Network Monitoring Subsystem



The messages listed under the status information that refer to the progress after each integration session, have been successfully implemented and tested during the respective integration sessions.

Status after March 2013 integration:

- TB[1] informs the MTM[2] on the created multiple multicast trees (P2P overlay) – **internal message**
- TB sends *dummy* ACTIVE links to TC[3] – **message #307[4]**
- TB periodically informs the P2PTx[5] about the list of top-layer peers – **message #328**
- TB relays authentication message (user credentials) to the Authentication, Registry and Security module – **message #329**
- TC sends *dummy* message to the peer's CS[6] about its ACTIVE links - **message #323**
- TC sends *dummy* message to NMS[7] informing on the list of children – **internal message**

[1] TB: Topology Builder
[2] MTM: Multicast Tree Manager
[3] TC: Topology Controller
[4] The message numbers refer to D2.2 [3], D2.3 [4] and D6.6 [5].
[5] P2PTx: Peer-to-Peer transmitter
[6] CS: Chunk Selection
[7] NMS: Network Monitoring Subsystem

- TC sends *dummy* message to NMS to request networking statistics (packet loss, delay, jitter, upload and download capacities) – **internal message**
- TC sends *dummy* network statistics to the CS - **message #333**
- TC sends message to NMS to request networking statistics (packet loss, delay, jitter, upload and download capacities) – **internal message**
- NMS sends message to TC to report *dummy* networking statistics – **internal message**
- CS requests TC about the peer's link capacity (upload and download) – **message #110**

Status after July 2013 integration:

- TB requests the MTM sender module to dispatch topology messages to peers - **messages #307, #320, #323**.
- TB informs the TC on the PeerID; authentication status; port and IP address to use for the link tests – **message #330**.
- TB informs the TC about its ACTIVE links - **message #307**
- TC sends user credentials to the TB for authentication purposes –**message #329**
- TC requests (after authentication) the TB to join the ROMEO P2P distribution system – **message #306**
- TC informs the peer's CS about its real ACTIVE links - **message #323**
- TC informs NMS on the real list of children – **internal message**
- CS requests TC about the peer's link upload and download remnant bandwidth – **message #331**
- NMS sends message to TC to report on-the-fly real network monitoring statistics (packet loss, delay, jitter, upload and download capacities) – **internal message**
- Synchronization has been tested with P2P Transmitter & Receiver, and P2P video transmission to fixed and portable terminals successfully achieved.

Status after October 2013 integration:

- TB informs the TC about its READY/BACKUP links -**message #320**
- TB informs P2PTx on the list of Mobile peers – **updated message #328**
- MTM periodically refreshes shared resources (peer's records) with the TB to allow detection of peer disconnections – **internal behaviour**
- TC informs the CS about its READY/BACKUP links -**message #320**
- CS requests the TC about the peer's networking statistics (packet loss only) – **updated message #331**
- CS (child) requests the CS (parent) to forward specific ROMEO streams – **messages #342, #343, #344**

- CS informs the P2PRxTx[8] to forward specific ROMEO streams to a specific child peer

Status after November 2013 integration:

- CS requests the TC about the peer's networking statistics (jitter, delay, average incoming bitrate for the ROMEO streams) – **updated message #331**

Status after March 2014 integration:

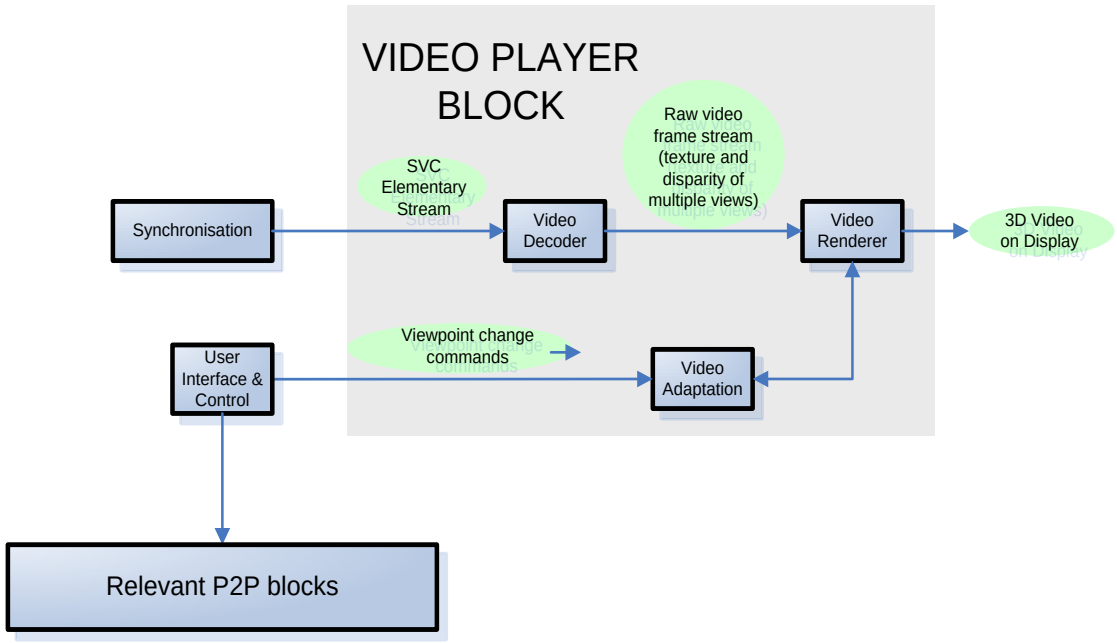
- TB no longer periodically informs the P2PTx for the list of top-layer parents. The message is sent once and then triggered again only when changes in the topology exist – **message #328**
- TC informs the CS about the networking (packet loss, delay, jitter, upload and download capacities) for a specific link – **message #333.**
- NMS sends (improved) on-the-fly network monitoring data to TC – **internal message**
- (update) CS (child) requests the CS (parent) to forward specific ROMEO streams – **updated messages #342, #343, #344**
- Capability of stream retrieving from P2P Transmitter / Receiver has been tested and verified in Synchronization in case of connection lost.
- CPU usage of Synchronization module has been optimized, tested and verified successfully.

Status after June 2014 integration:

- TB informs P2PTx on the list of Portable peers – updated message #328
- TC informs the CS the peers is going to abandon the P2P overlay (graceful disconnection) – **message #335** (new message, not defined in D6.6)
- TC sends to the CS information on the user viewpoint selection – **message #334** (new message, not defined in D6.6)
- CS request TC on the bitrate of each incoming bit stream – **updated message #331**
- (update) CS (child) requests CS (parent) to forward specific ROMEO streams – **updated messages #342, #343, #344**

[8] P2PRxFx: Peer-to-Peer Receiver & Forwarder

2.4 Group 08: Synchronisation, Video Decoding, Video Rendering, Video Adaptation'



Status after March 2013 integration:

- The connection between the Synchronisation module and the video player block as depicted in Table 2 could not be established yet.
- The core scalable video decoding functionality has been established (excluding multiple-descriptions merging).
- Mobile terminal's platform decoder performance (in terms of speed) has been tested using locally stored video elementary streams.
- The connection with the User Interface and Control through the video adaptation module has not been established yet.
- The definition of the data interface between the video decoder and the video renderer has been defined, including the video/disparity channel association and the packet structure.
- The reference Infiniband communication structure has been defined, assuming video decoding and video rendering functionalities are split between different platforms.
- The data packet structure between the Synchronisation module and the video decoders (elementary stream packets and PTS) and the video renderer (PCR) has been defined.

Status after July 2013 integration:

- The connection and the compressed video packet transfer between the Synchronisation module and the video decoders have been tested and verified (i.e., correct packet structure).

- Video rendering functionality has been achieved and tested for three different modes of operation: MVD2 (with depth adjustment on a stereoscopic screen), MVD2 (for Alioscopy autostereoscopic display), MVD4 (for Alioscopy autostereoscopic display).
- Video renderer functionality has been tested with a custom video decoding client.
- The data communication between the video renderer and the scalable multi-description video decoder has not been established yet.
- Mobile terminal's platform decoder performance has been tested locally with the stereoscopic renderer using locally stored video elementary streams.

Status after October 2013 integration:

- The multiple-TCP channel video frame data communication between the video decoders and the video renderer has been achieved and verified.
- It has been decided to unify the video decoder and the video renderer platform, such that no Infiniband is necessary (without compromising the video frame rate).
- The control messaging between the User Interface and Control module and the video renderer has been defined (i.e., the structure of the user controls on the user interface, and the JSON message structure for transferring these information to the video renderer).
- The switching functionality of the video renderer across modes of operation has been improved, such that depending on the frame availability on different input channels' buffers, the video renderer can adapt its rendering mode.
- Mobile terminal's platform video decoding and stereoscopic rendering has been tested with the DVB content retrieved through the DVB receiver (PCTV USB stick).

Status after November 2013 integration:

- The preliminary evaluation of the video adaptation process (i.e., video rendering based on user's depth and viewpoint preferences) has been performed on the Alioscopy multi-view display and on the stereoscopic monitor (attached to the portable terminal), and the provided depth and viewpoint range has been found to be sufficient.
- The integrity of the modules making the video player has been verified for the fixed and portable ROMEO peers.

Status after March 2014 integration:

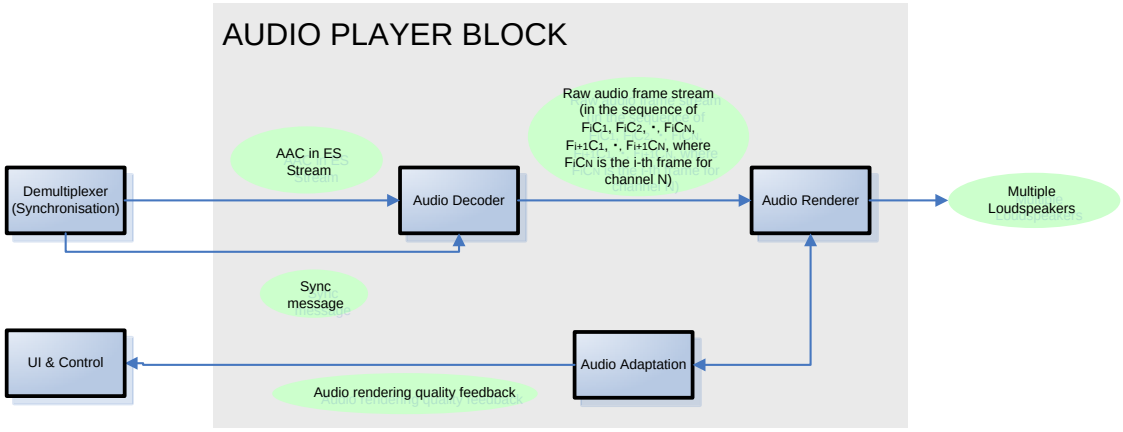
- The fixed terminal has been tested with two different multi-view displays (Alioscopy and Dimenco). The user interface is allowing the user to choose the display and send the adapted JSON message to the renderer.
- The mobile terminal's switching application between the DVB viewpoint and the requested left or right viewpoints (P2P viewpoints) has been tested.

Status after June 2014 integration:

- UI[9] can inform the relevant P2P block (TC[10]) on the user's viewpoint selection – **message #1008**
- The mobile terminal's platform video decoder and renderer modules have been tested with other client side modules, including the synchronisation module and the P2P sub-modules.

[9] UI: User Interface
[10] TC: Topology Controller

2.5 Group 09: Synchronisation, Audio Decoding, Audio Rendering



Status after March 2013 integration:

- A preliminary multi-channel audio decoder for fixed terminal has been put into test. The output is to a local PCM file and can be player by general purpose audio players supporting raw data playing.
- The data packet structure between the Synchronisation module and the audio decoders (elementary stream packets and PTS) and the video renderer (PCR) has been defined.
- Multiple objects audio streams data format for TS multiplexing has been proposed to and agreed with all related partners.
- The data communication between the fixed audio decoder and renderer has been tested and verified to be stable.
- An independent integration test for fixed-terminal audio decoding and rendering modules can be achieved by simulation of audio packets from the Synchronisation module with a streaming application. Partners have developed audio streaming application for this purpose. This made testing of object-based decoding and rendering independent of the Synchronisation module.
- The interface from the head-tracker to the mobile audio renderer has been successfully integrated and tested.
- The data communication between the mobile / portable decoder and the renderer has been tested. It was in a stable working state.

Status after July 2013 integration:

- Data communication between the fixed audio decoder and the synchronisation module has been tested. It showed that audio data streaming between fixed audio decoder and the synchronisation module is possible, but with missing audio frames.

- Test of extracting and using PTS in decoding the audio frames from synchronisation module has been made. The result has shown an inconsistent PTS value, which is caused by the mismatch of bytes order in Windows and Linux OS. It is subsequently addressed.
- The fixed-terminal audio decoding and rendering modules (without using PCR and PTS for actual synchronising with video) have been shown to work by simulating the audio frames from the synchronisation module.
- Object audio decoder API interface has been developed for initial test, where the decoded audio raw data is written to local file.
- Mobile audio decoder API, platforms and data format between the mobile audio decoder and renderer is proposed to and agreed with all related partners.
- The audio scene streaming implementation has been tested. It was working but had some minor issues to fix.

Status after October 2013 integration:

- The integration of fixed-terminal audio decoding and rendering modules with de-multiplexing and synchronisation module is completed and verified working properly. An Intermittent audio playback issue has been observed and solved by adjusting the buffering level at the synchronisation module.
- Synchronisation between fixed multichannel audio and video has been achieved.
- Multi-object audio decoder is integrated successfully with object audio renderer for portable devices. The object audio source is simulated and sent to the decoder. Attempts have been made to integrate the object audio decoder and renderer with synchronisation module without success. Issues have been identified in the generation of the object audio content and left to be addressed by partners after this integration session.
- Windows version of mobile audio decoder has been tested running fine on the mobile devices. The porting of code from Windows to Linux has been in progress and partners have agreed that the ported Linux object audio decoder will be tested in next integration session.
- Both mobile and portable decoder data communication with the according renderer has been tested. It was working mostly stable but still had some issues to fix.

Status after November 2013 integration:

- Fixed terminal: audio decoder and renderer integration with synchronisation module has been successful and synchronisation between fixed audio and video is also working fine.
- Multi-object audio master stream has been generated for test. It uses an object interleaving mechanism to multiplex all object audio into one object audio ES.
- Mobile audio decoder porting to Linux has been completed and tested. Tests have been done by simulation the audio source from synchronisation module.

High CPU usage on the mobile device has been observed and to be addressed by partners during the integration.

- The first end-to-end chain for mobile and portable audio renderer has been tested. It was working with limitations but mostly stable.

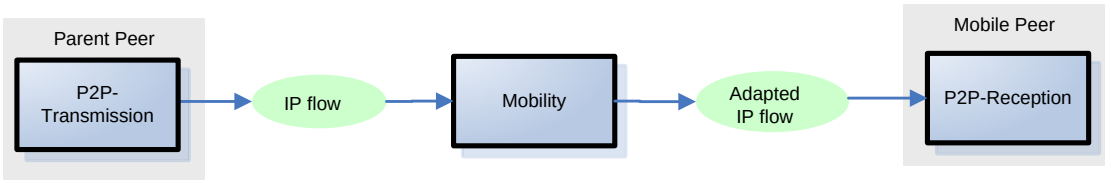
Status after March 2014 integration:

- Object audio decoder and renderer have been integrated with the synchronisation module. Intermittent audio is observed due to incoming audio frame rate is lower than specified and it is rectified in the P2P transmitter/receiver module.
- Synchronisation delay between object audio and video is observed during integration. It is identified that there is a constant PTS stamp differences between the object audio and video. It is rectified in a regenerated object audio master elementary stream.
- Mobile audio decoder is ported successfully to the Linux platform and integrated with the demux/synchronisation and P2P modules.

Status after June 2014 integration:

- A stable multi-object audio rendering system has been setup during the integration. Portable multi-object audio decoder, renderer and scene description decoder/renderer have all been tested working individually. Integration of all portable object audio modules with the synchronisation and P2P modules are completed successfully.
- Mobile audio decoder and renderer (along with the head-tracker for the mobile device) have been integrated into the Linux mobile devices and synchronised with audio and video renderers on fixed and portable devices.
- During this integration meeting, the intra-synchronisation of all renderers (fixed, portable and mobile) as well as the synchronisation to the video renderer has been tested successfully.

2.6 Group 12: P2P Transmission, Mobility, P2P Reception



Status after March 2013 integration:

- The Media Aware Proxy, Media Independent Handover and Fast Proxy Mobile IP functionalities of the Mobility module have been integrated over the LTE EPC platform and were tested for performance.
- P2P reception and depacketization have been tested with the synchronisation module.

Status after July 2013 integration:

- A successful vertical handover demonstration between the emulated E-UTRAN and Wi-Fi access during 3D video streaming has been performed. The vertical handover was system initiated and was forced by overloading the capacity of the LTE access network beyond a predefined threshold value.
- The P2P transmitter and receiver have been integrated with the LTE EPC platform and the vertical handover functionality has been executed successfully.
- At this point all the integration procedures have been based on the portable terminal.

Status after October 2013 integration:

- Based on the successful demonstration of the vertical handover, the integration process has been furthered with the fine tuning of the Media Aware proxy and Media Independent Handoff functionalities that are integrated over the LTE EPC platform, in order to successfully execute horizontal handovers.
- The P2P transmitter and P2P receiver modules have been tested successfully over the LTE platform. The Mobility module was at that time able to perform decisions for vertical and horizontal handovers based on the portable terminal.
- The functionalities developed for the portable terminal have been integrated on the mobile terminal successfully.

Status after November 2013 integration:

- The integration focus has switched to the multi-homing scenario for the portable terminal.
- A new horizontal handover buffering function has been developed in order to counter effect the impact of service discontinuity during the horizontal handovers.

- The integrated P2P Transmitter – Mobility – P2P receiver setup has been tested along with the synchronisation module.

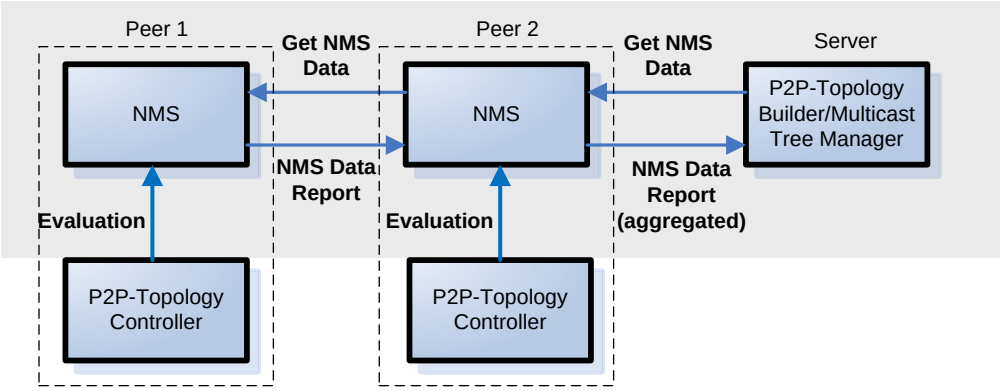
Status after March 2014 integration:

- The Mobility module has been integrated and tested successfully on the mobile terminal.
- The integration with the synchronisation module in order to fine-tune the pre buffering queue size has been successful.
- The machine learning based QoE model that is based on the network and physical layer parameters has been integrated with the Media Independent Handoff functionality of the Mobility module.

Status after June 2014 integration:

- P2P Transmission-Mobility-P2P Reception have been tested successfully for seven video streams transmitted over Wi-Fi and emulated LTE networks to the portable and the mobile terminal.
- The Mobility module has also been tested successfully with the Topology controller, during vertical handovers. The Topology Controller is able to identify the changes of the network interfaces during the handover and update the topology accordingly.

2.7 Group 14: Network Management Subsystem, P2P Topology Controller and Network Monitor, P2P Topology Builder/Multicast, Tree Manager



The messages listed under the status information that refer to the progress after each integration session, have been successfully implemented and tested during the respective integration sessions.

Status after March 2013 integration:

- TC[11] sends *dummy* message to NMS[12] informing on the list of children – **internal message**
- TC sends message to NMS to request networking statistics (packet loss, delay, jitter, upload and download capacities) – **internal message**
- NMS sends message to TC to report *dummy* networking statistics – **internal message**
- NMS (child) sends *dummy* NMS report to NMS (parent) - **message #314**
- NMS (top-layer parent) sends *dummy* networking report statistics (containing its own plus all children NMS reports) message to MTM -**message #316**

Status after July 2013 integration:

- NMS sends message to TC to report on-the-fly real network monitoring statistics (packet loss, delay, jitter, upload and download capacities) – **internal message**
- NMS (child) sends NMS report to NMS (parent) with real networking statistics (packet loss, delay, jitter, upload and download capacities) to NMS (parent) - **message #314**
- NMS (top-layer parent) sends to the MTM its own NMS Report plus all it's children NMS Reports - **message #316**

Status after October 2013 integration:

[11] TC: Topology Controller
[12] NMS: Network Management Subsystem

- Message #314 was deprecated, preference was given to **message #316** (NMS→MTM) for performance reasons

Status after November 2013 integration:

- MTM triggers notification to TB informing a specific peer is going to disconnect (graceful disconnection) – **internal message**

Status after March 2014 integration:

- NMS sends (improved) NMS reports to the MTM - updated **message #316**
- NMS sends (improved) on-the-fly network monitoring data to TC – **internal message**

Status after June 2014 integration:

- No additional integration work has been performed

3 Mapping of groups of modules/ sub-systems to use cases

For the three reference scenarios, the following use cases had been defined and described in D7.3.

Reference Scenario 1: Consumption of 3D audio/video over hybrid delivery paths
Use case 1.1: User Join
Use case 1.2: Live Stream reception of fixed user with optimal conditions
Use case 1.3: A/V adaptation
Use case 1.4: Peer Disconnect
Use case 1.5: Periodic User Condition Reporting
Use case 1.6: Intra-system handover
Use case 1.7: Multi-homing transmission
Use case 1.8: Inter-system handover

Reference Scenario 2: Collaborating group and sharing user generated content
Use case 2.1: Initiating and starting a Video Conference
Use case 2.2: Joining an existing collaborating group

Reference Scenario 3: P2P quality of service (QoS) over fixed access networks
Use case 3.1: QoS Adjustment

Table 4 gives an overview of the mapping between the group of modules/ sub-modules and the abovementioned use cases.

	Group 02 'Content Generation, Media Encode & TS Generation, DVB Transmission'	Group 03 and Group 05 'P2P Transmission, P2P Packetisation, Authentication, Registration and Security, A/V Overlay, UGC'	Group 06 'P2P Reception, P2P Chunk Selection, P2P Transmission, Topology Controller, Network Monitoring Subsystem'	Group 08 'Demultiplexing (Synchronisation), Video Decoding, Video Rendering, Video Adaptation'	Group 09 'Demultiplexing (Synchronisation), Audio Decoding. Audio Rendering	Group 12 'P2P Transmission, Mobility, P2P Reception'	Group 15 'Network Management Subsystem, P2P Topology Controller and Network Monitor, P2P Topology Builder/Multicast, Tree Manager'
Ref. Scen. 1							
Use case 1.1							
Use case 1.2							
Use case 1.3							
Use case 1.4							
Use case 1.5							
Use case 1.6							
Use case 1.7							
Use case 1.8							
Ref. Scen. 2							
Use case 2.1							
Use case 2.2							
Ref. Scen. 3							
Use case 3.1							

Table 4 - Mapping between use cases and groups of modules/ sub-modules

4 Conclusions

This deliverable presents a report on the progress during the six integration sessions of the ROMEO project between March 2013 and June 2014. These sessions were organised in different locations and regularly attended by all partners.
The groups described in the sub-sections of Section 2 are the most important intermediate integration steps on the way to the complete demonstrator, which will be delivered in M34 as D7.5 – Demonstrator.

5 References

[1] ROMEO Deliverable 7.3, “Final Trials Plan and Validation Methodologies”, 2014

[2] ROMEO Deliverable 7.2, “Pilot Set-up and Evaluation Methodologies”; 2013;

[3] ROMEO Deliverable 2.2, “Definition of the initial reference end-to-end system architecture and the key system components”; 2012

[4] ROMEO Deliverable 2.3, “Interim reference system architecture report”; 2014

[5] ROMEO Deliverable 6.6, “Report on Developed Final System Components”; 2014

APPENDIX A: GLOSSARY OF ABBREVIATIONS:

A	
AAC	Advanced Audio Coding
ACS	Admission Control Subsystem
ALSA	Advanced Linux Sound Architecture
AP	Access Point
AVC	Advanced Video Coding
AU	Access Units
A/V	Audio-Visual
B	
BRAS	Broadband Remote Access Server
C	
CIB	Control Information Base
CLD	Channel Level Differences
CPC	Channel Prediction Coefficients
CPE	Consumer-Premises Equipment
CPU	Central Processing Unit
CoS	Class of Service
CUDA	Computer Unified Device Architecture
D	
DBIR	Depth based Image Rendering
DHCP	Dynamic Host Configuration Protocol
DPX	Digital Picture Exchange
DVB	Digital Video Broadcasting
DB	Database
E	
eNodeB	Enhanced Node B
ER	Edge Router
ES	Elementary Stream
F	
fps	Frames per Second
FTTH	Fiber to the Home
G	

GbE	Gigabit Ethernet
GOP	Group of Pictures
GPL	General Public license
GPON	Gigabit Passive Optical Network
GPU	Graphics Processing Unit
GUID	Global Unique Identification
H	
HA	Home Agent
HD	High Definition
HDMI	High Definition Multimedia Interface
HGW	Home Gateway
I	
ICC	Inter-channel Coherences
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
IRACS	Internet Resource and Admission Control Subsystem
ISP	Internet Service Provider
J	
JSON-RPC	Javascript Object Notation – Remote Procedure Call
K	
L	
L2	Layer 2 (LLC and MAC level)
L3	Layer 3 (Routing, IP level)
LAN	Local Area Network
LGPL	Lesser General Public license
LMA	Local Mobility Anchor
LTE	Long Term Evolution
M	
MAG	Mobile Access Gateway
MAN	Metropolitan Area Network
MANE	Media Aware Network Element
MAP	Media Aware Proxy
MIH	Media Independent Handover
MIP	Mobile IP
MME	Mobility Management Entity
MN	Mobile Node

MPEG	Motion Picture Experts Group
MPS	MPEG Surround
MTM	Multicast Tree Management
N	
NACF	Network Attachment Control Functions
NALU	Network Abstraction Layer Unit
NAT	Network Address Translation
NATIE	Network Address Translation Implementer Equipment
NCDP	Network Control Decision Point
NMS	Network Monitor SubSystem
NTFS	New Technology File System
O	
OLT	Optical Line Terminator
ONT	Optical Network Termination
OS	Operating System
P	
P2P	Peer-to-Peer
PCEF	Policy and Charging Enforcement Function
PCM	Pulse Coded Modulation
PCR	Program Clock Reference (MPEG2-TS related)
PCRF	Policy Charging and Rule Function
PES	Packetised Elementary Stream
PMA	Proxy Mobile Agent
PMIP	Proxy Mobile IP
Q	
QoE	Quality of Experience
QoS	Quality of Service
R	
RAM	Resource and Admission Manager
RC	Resource Controller
RF	Radio Frequency
RGW	Residential Gateway
RRM	Radio Resource Management
RRS	Resource Reservation Subsystem
RTP	Real Time Protocol
S	

SBC	Session Border Controller
SDI	Serial Digital Interface
SDL	Specification and Description Language
SI	Service Information
SIP	Session Initiation Protocol
SLA	Service Level Agreements
SNR	Signal to Noise Ration
SVC	Scalable Video Coding
T	
TB	Topology Builder
TC	Topology Controller
TCP	Transmission Control Protocol
TS	Transport Stream
U	
UDP	User Datagram Protocol
UI	User Interface
UMTS	Universal Mobile Telecommunications System
USB	Universal Serial Bus
V	
VCL	Video Coding Layer
VoDSAR	Video on Demand Service Access Router
VRF	Virtual Router Forwarding
VSEE	Virtual Software Execution Environment
W	
WFS	Wave Field Synthesis
Wi-Fi	Wireless Fidelity
WLAN	Wireless - LAN
X	
Y	
Z	
Numbers	