CONVINcE

D5.3.1

Completion of full demonstrators' experiments and validation, demonstration of all use-cases with all measurement tools integrated

Editor: Dipti Adhikari (EXFO)

Reviewer: Adrian Popescu (BTH)

Authors: Dipti Adhikari (EXFO)
Juha Merikallio (Teleste)
Patrik Arlos, Markus Fiedler (BTH)
Samin Mohammadi, Reza Farahbakhsh (IMT-TSP)
Rickard Ljung (Sony)
Peter Barmann (Sensative)
Siwar BEN HADJ SAID (CEA)
Oguzhan Iren EROL (VESTEL)
Tenager Mekonnen (UO)
Mikko Uitto, Martti Forsell (VTT)
Anders Nilsson Plymoth (TelHoc)
Miika Komu, Jimmy Kjällman (ER)
Raoul Monnier (TVN)
Yannick Carlinet (Orange Labs)

CONVINcE confidential

1 EXECUTIVE SUMMARY

The objective of this report is to describe the use-case scenarios defined under the scope of CONVINcE and to validate the results of power consumption measurements for each scenario. The document also describes the data paths used in these scenarios to form the end-to-end integrated demonstration.

Five different use-case scenarios have been defined under the scope of CONVINcE:

- On Demand Video Streaming
- Video Surveillance
- Virtualized Video Surveillance
- Content Popularity
- Optimized Crowded WiFi

The data paths defined for the end-to-end integrated demonstration are:

- Data Path 1: Video on Demand
- Data Path 2: Video on Demand
- Data Path 3: Video Surveillance Network

The end-to-end demonstration and verification is done by using these five scenarios.

Deliverable D5.3.1 is part of the work package WP5 where the evaluation of the end-to-end delivery chain used for integrated demonstration is done.

The definition and identification of the use-case scenarios used for power saving mechanisms from the head end to network and to terminals are done within the work packages WP2, WP3 and WP5. These work packages refer to different components that are combined to form the end-to-end video delivery chain.

Table of Contents

Table of Figures

Table of Tables

2 DOCUMENT HISTORY AND ABBREVIATIONS

2.1 Document history

Version	Date	Description of the modifications
1.0	1.3.2017	TOC defined
1.1	21.3.2017	TOC Approved
1.1	5.4.2017	EXFO and Teleste updated the results
1.2	21.4.2017	Added contribution from Partners: SONY Mobile, IMT, BTH, Sensative, CEA
1.3	4.5.2017	Added contribution from partners: Ericsson and UO and VTT
1.4	5.5.2017	Added contribution from partners: Vestel
1.5	15.5.2017	Integrated version
1.6	1.6.2017	VTT modified contribution added
1.7	16.6.2017	Ready for review
2.5	04.09.2017	Final version

2.2 Abbreviations

CMTS	Cable Modem Termination System
CNOM	Cognitive Network Operation and Management
CSV	Comma Separated Value
DTTV	Digital Terrestrial Television
HDMI	High Definition Multimedia Interface
HD	High Definition
HEVC	High Efficiency Video Coding
IP	Internet Protocol
MOS	Mean Opinion Score
OF	OpenFlow
OS	Opinion Score
PMT	Power Measurement Terminal
QIP	QAM Internet Protocol
QoE	Quality of Experience
QoEJ	Quality of Experience per Joule
QoS	Quality of Service
SD	Standard Definitions
SDN	Software Defined Networking
STB	Set Top Box
UI	User Interface

VoD	Video on Demand

3 INTRODUCTION

The overview of the demonstrator scenario is as follows.

Integrated Demonstrator Scenario

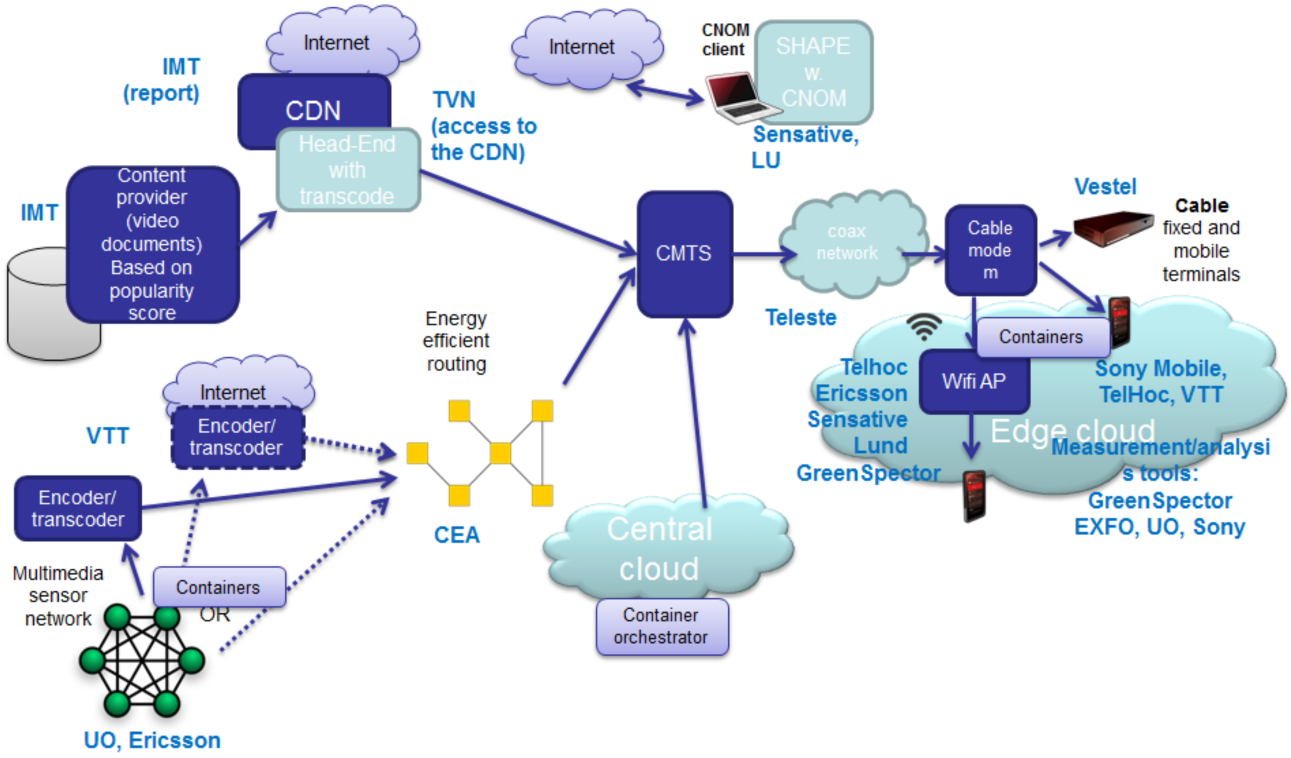


Figure 1: Integrated Demonstrator Scenario

Demonstrators - Synoptic Diagram

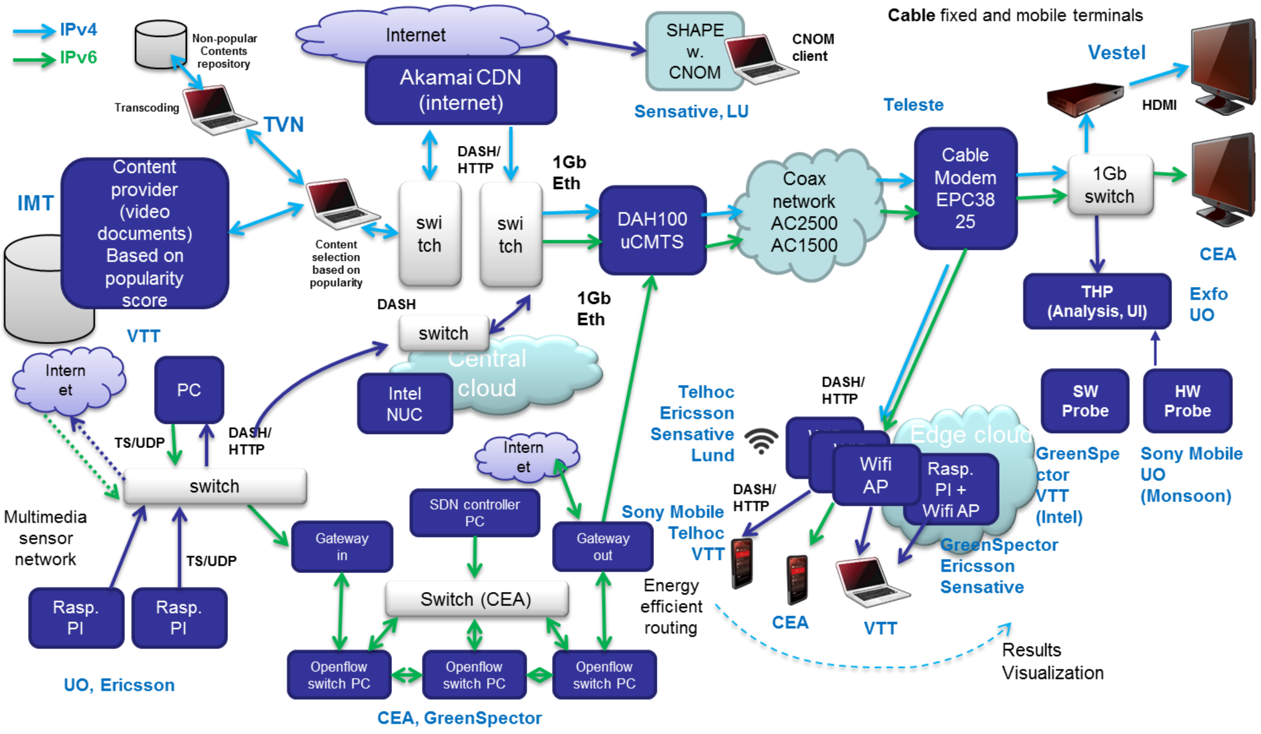


Figure 2: Demonstrators - Synoptic Diagram

Details about the tools used for measurements are defined in the document: https://bscw-convince.celticplus.eu/bscw/bscw.cgi/d20719/CONVINcE%20T5.3_Demonstrator%20Measurement%20Tools%20and%20Results_v1.6.xlsx

4 DEMONSTRATOR MEASUREMENT AND RESULTS

This section describes the use-case scenarios and reports the results of power consumption measurements defined under the scope of CONVINcE.

4.1 Source and Head End

4.1.1 Video Surveillance Network

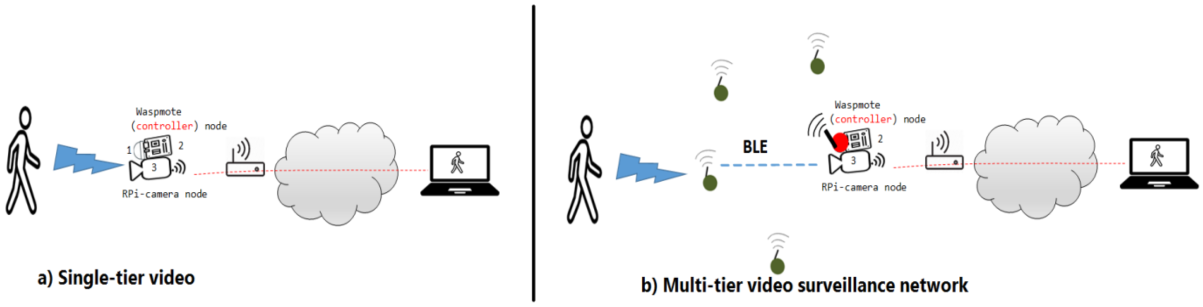
Partners: **University of Oulu, Ericsson and VTT**

This section describes the measured parameters and reports the results for the video surveillance scenario in the non-orchestrated scenario (i.e. no virtualization). The integrated, orchestrated version is described in section 5.3. Software encoding power consumption is investigated more in section 4.1.2.

4.1.1.1 Description

We have developed a system of low power video surveillance networks using open source software and hardware (Raspberry pi and Waspmote). The surveillance system streams the video to a remote server/laptop when motion is detected. We provide two-versions of the surveillance system: single-tier and multi-tier video surveillance system. While the single-tier is more energy efficient and cheap deployment scenario, the multi-tier provides extra-advantages such as monitoring wider geographical area.

The network architecture for both scenarios is presented below. The main goal in both scenarios is to reduce the idle power consumption of the system.



a) Single-tier video

b) Multi-tier video surveillance network

Figure 3: low power video-surveillance network. a) single-tier b) multi-tier

4.1.1.2 Measured Parameter

The parameter measured in the video-surveillance network is power.

4.1.1.3 Measured Results

a) Single-tier

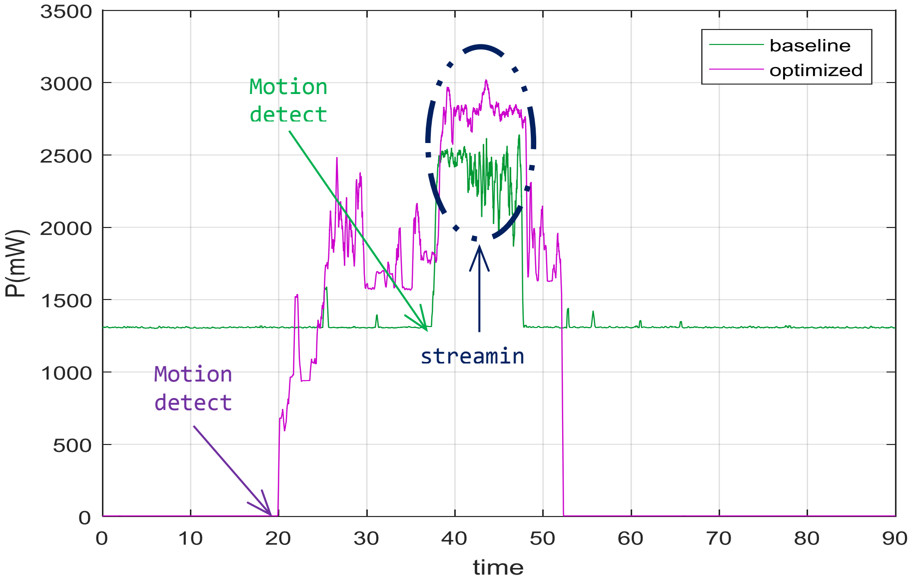


Figure 4: power consumption of single-tier video-surveillance network

As it can be seen in the figure 4, the optimized surveillance system (purple colour) is in low-power mode (almost zero mW consumption) when there is no activity in the area under surveillance. When motion is detected, the camera sensor node wakes up and starts streaming video to the laptop. The boot-up time for the camera in the present prototype is nearly 17 seconds. This can be improved by optimizing the operating system of the camera node and/or using multi-tier network architecture (figure 5). It is observed that the multi-tier network architecture can provide enough time for the camera sensor node to wake-up and to capture/stream the event on time.

b) Multi-tier

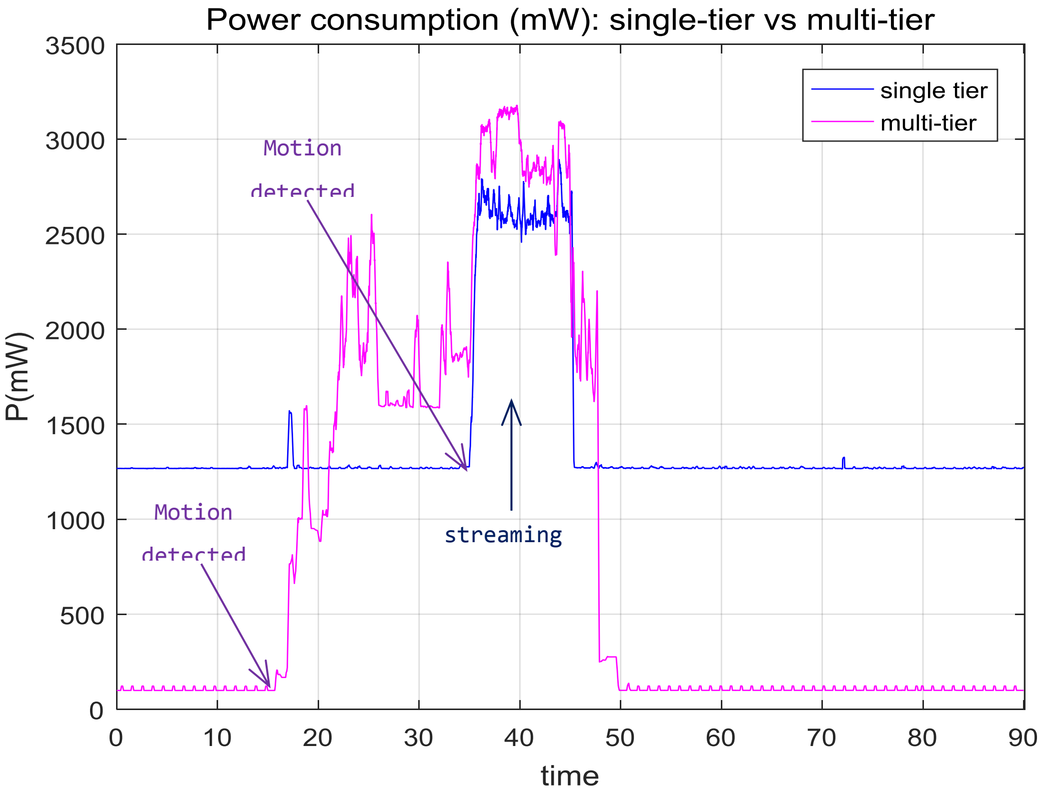
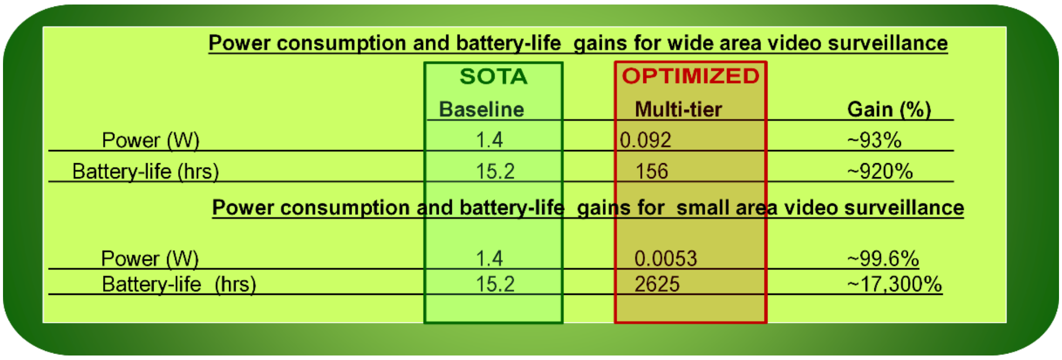


Figure 5: power consumption of multi-tier video-surveillance network

Summary: Power and battery life gains of low-power surveillance network



	SOTA Baseline	OPTIMIZED Multi-tier	Gain (%)
Power consumption and battery-life gains for wide area video surveillance			
Power (W)	1.4	0.092	~93%
Battery-life (hrs)	15.2	156	~920%
Power consumption and battery-life gains for small area video surveillance			
Power (W)	1.4	0.0053	~99.6%
Battery-life (hrs)	15.2	2625	~17,300%

4.1.2 Video Sources and Software Encoder

Partner: **VTT**

The demonstration path includes the VTT's video content server with software encoder. The server can be seen in Figure 1, on the left.

4.1.2.1 Description

The server uses two network interfaces. The first one is connected to the camera surveillance network for receiving the stream and the second one into the CEA energy efficient routing algorithm for outputting the stream over IPv6.

The encoding server has three main functionalities:

a) Acting as a HTTP web server for providing pre-encoded video-on-demand (VoD) content

b) Acting as live encoder for webcam input

c) Acting as a live transcoder for transcoding the content from surveillance network.

FFmpeg with x264 (H.264/AVC) and x265 (HEVC) encoding software libraries are used for encoding/transcoding the content into MPEG-DASH format suitable for HTTP streaming. For the transcoding task in functionality c), we mean encapsulation to the MPEG-DASH format without transcoding the input video into alternate resolution or bitrate. Intel Core i7-5820K@3.3 GHz PC is used for encoding the content into VoD at the VTT site with Eaton power board for measuring the outlet power. At the Orange site, Intel Core i7-3720QM@2.60GHz is used, which provides slightly different but aligned power consumption results with the first machine. Intel power gadget is used for measuring the processor power.

4.1.2.2 Parameters Measured

We measure the power and total energy consumption for encoding the content into VoD and live stream from the surveillance network. Measurement for the live webcam stream is considered as optional.

- **INPUT:**
  - Raw video (yuv/y4m) file.
  - Live (raw) video from webcam
  - Live video feed from surveillance network.

- **OUTPUT:**
  - H.264/AVC or H.265/HEVC encoded stream encapsulated into MPEG-DASH format
  - Live power consumption visualized in external display
  - CSV file containing power measurement results in format to be illustrated in EXFO user interface. The format is Comma Separated Values (CSV): Timestamp [Seconds]; Power[Watt]; Description [Text]
  - The output bitrate in bits/s and/or video quality in PSNR-Y (dB).

4.1.2.3 Measured Results

Figure 6 shows the results when using low or complex encoding compression for H.264 and HEVC test video sequence, which is a 60-second motion clip from "Tears of Steel". It is observed that the AVC encoding requires less power, but it also increases the required bitrate significantly when using similar compression tools. We measured 50W as idle power consumption for the encoding server at VTT. The results contain total power consumption of the server. It is also notable that HEVC high compression encoding consumed 122 KJ of energy at the encoding time of 650 seconds.

- **AVC**: Lowest average power consumption: **56W** for 61 seconds, total energy **3,4 KJ**, bitrate **15,6 MB/s**.
- **HEVC:** Lowest average power consumption: **94W** for 61 seconds, total energy **5,7 KJ**, bitrate **5,0 MB/s**.

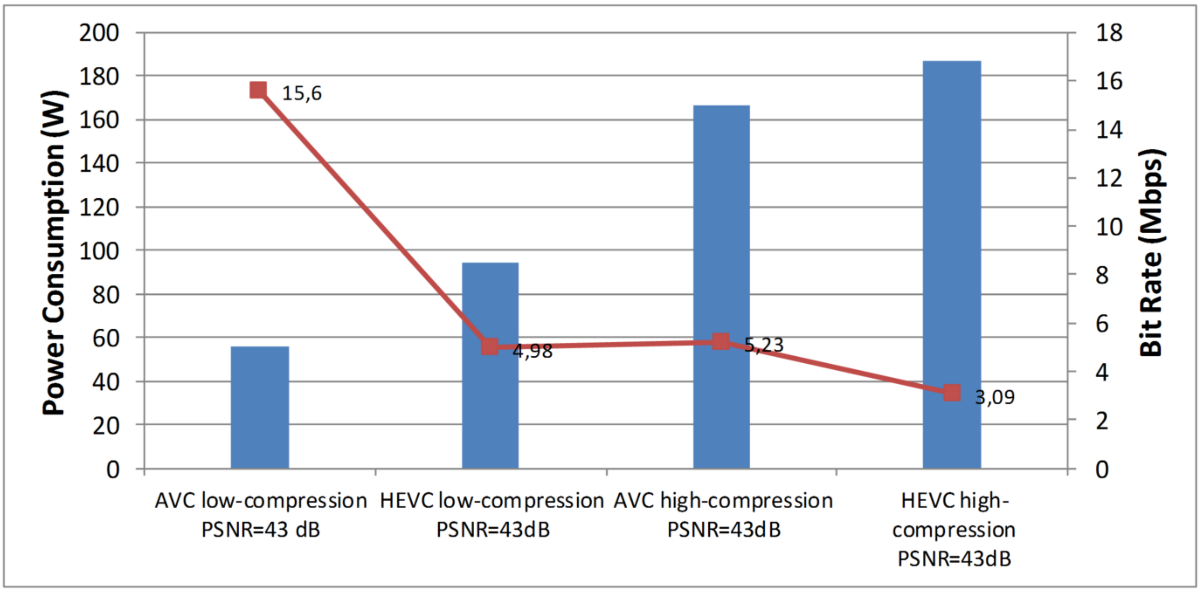


Figure 6: Power consumption values for VTT encoding server when using different compression techniques for H.264 and HEVC video.

4.1.3 Popularity of Chosen Video Streams

Partner: **IMT**

Popularity of Chosen Video Streams is a module that uses popularity metrics and factors to compute the popularity of videos. It gets the videos meta-data as a CSV file and assigns them a priority number representing their popularity value. This module helps to decide which videos should be cached in the networks that leads to the decrease of data traffic and consequently decrease of consumed energy in the network.

4.1.3.1 Description

Popularity is measured by different metrics depending on application. The most important factors considered in video popularity measurements are the number of views and number of likes. The publish date is also an important item to compare the videos popularity numbers. Here we compute the popularity of videos as follows:

$$\boldsymbol{without\ Date_{publish}}:\ Popularity = \alpha(\#views_{total}) + \beta(\#likes_{total}) + \gamma\left(\frac{\#likes_{total}}{\#views_{total}}\right)$$

$$\boldsymbol{with\ Date_{publish}}:\ Popularity = \alpha\big(likes_{daily}\big) + \beta\big(views_{daily}\big) + \gamma\left(\frac{likes_{daily}}{views_{daily}}\right)$$

$$likes_{daily} = \frac{\#likes_{total}}{\#days}\ ,\ views_{daily} = \frac{\#views_{total}}{\#days}$$

4.1.3.2 Parameters Measured

The module does not measure individually any parameter. Instead, it computes the popularity of the videos that provides a valuable metric for the CDNs to choose the most popular videos for catching.

4.1.3.3 Measured Results

We compute the popularity of 10 randomly selected videos from YouTube, which will be used in the Harmonic's Encoder.

The input is a CSV file of video's metadata as shown in Table 1.

Table 1 Metadata Information of Input Videos

Video id	Number of Views	Number of Likes	Publish Date	URL
1	598409	5650	2016-10-04	https://www.youtube.com/watch?v=fHLsGvAgvmE
2	9319	103	2016-03-15	https://www.youtube.com/watch?v=DiLwv-G9COQ&list=PLJxnQXiytA_RnvWf3ELztehOdauWxUuze&index=37
3	23352	192	2016-01-28	https://www.youtube.com/watch?v=tpubLPTaPjg
4	1732491	15654	2016-02-29	https://www.youtube.com/watch?v=AOoP56eXtzM
5	137510	339	2016-10-01	https://www.youtube.com/watch?v=q922TDCll6g
6	192417	2918	2016-07-28	https://www.youtube.com/watch?v=ZVnkBWeYAeY
7	1157288	5917	2014-09-01	https://www.youtube.com/watch?v=Q-4w5xYLwiU
8	268640	2736	2017-01-27	https://www.youtube.com/watch?v=dvBIkhxDMEc
9	17569	45	2017-01-27	https://www.youtube.com/watch?v=KB9e5Qb7Eks

10	7277008	7368	2009-03-01	https://www.youtube.com/watch?v=9QscURRuF0g

The output is a list of sorted videos based on their computed popularity as shown in Table 2.

Table 2 Sorted Popularity of Input Videos

Video Id	Popularity
8	4522
4	1494
1	1362
10	751
7	389
5	304
9	293
6	291
3	18
2	8

4.1.4 Transcoding

Partner: **TVN**

The content popularity use case is based on transcoding "on the fly" of non-popular contents instead of storing them "for eternity" in a CDN (see details in deliverable D2.2.2). This approach is also called "Just in Time Transcoding".

4.1.4.1 Description

In the headend, only one representation of the video content is produced in a single format (mezzanine format). This is stored in the cloud (CDN). In the edge, as close as possible to the end user, a transcoding operation is processed in order to provide the terminal of the end user with the requested format/bitrate. This is presented in Figure 7.

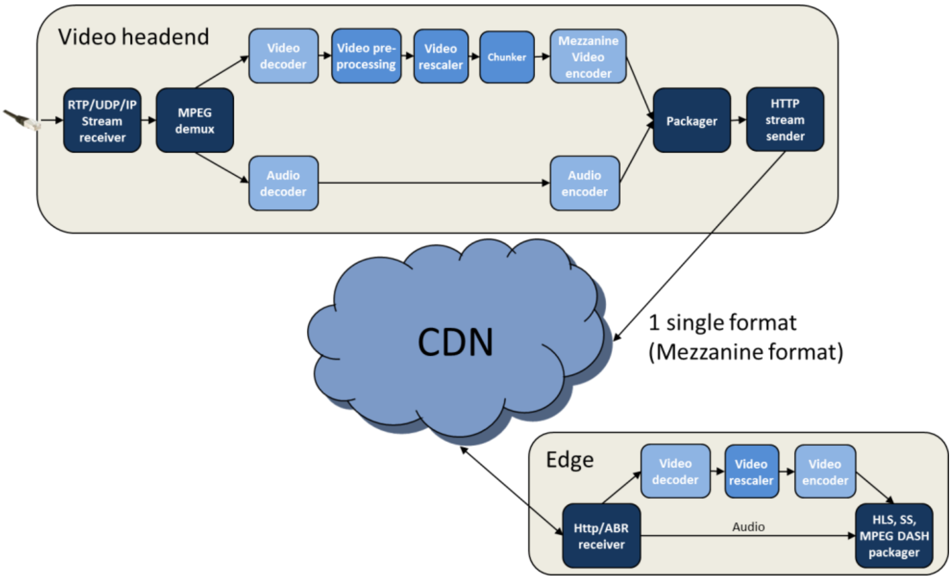


Figure 7: "Just In Time Transcoding" block diagram – Brute force approach

Figure 7 is based on the assumption that a full decode and a full encode is done. This is what it is called the “brute force approach”. The variant shown by Figure 8 can also be used to reduce the power consumption, making use of metadata generated by the encoder in the headend and of a simplified video transcoder.

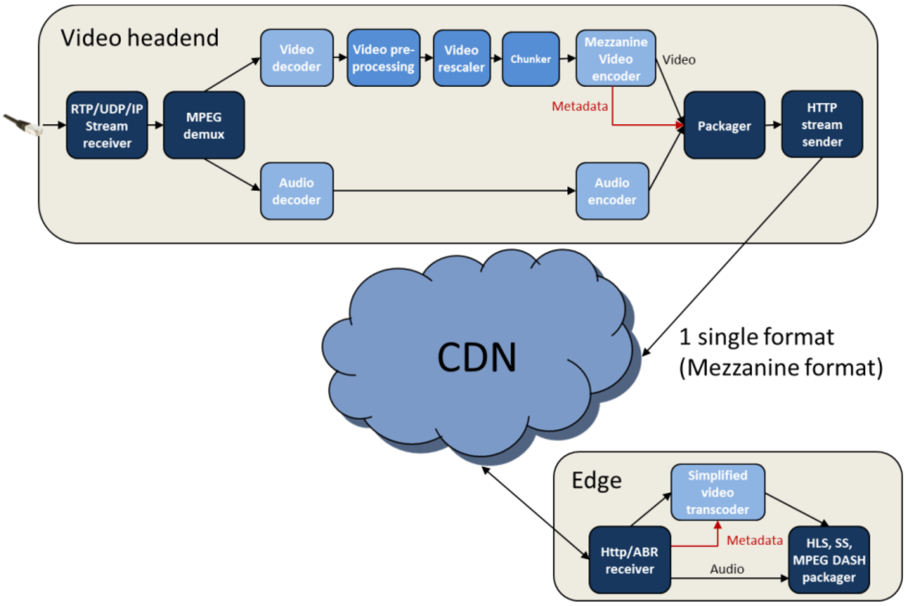


Figure 8: Optimized edge transcoding using metadata

4.1.4.2 Parameters Measured

The power consumption of the encoder is measured in both approaches.

Two techniques were used to measure the power consumption. The first one uses an external device (Ted Pro measurinig tool[1]). The second one calculates the consumption based on the value of the voltage and current given by the CPU of the device. The results obtained with both approaches have been compared and they showed similar results.

4.1.4.3 Measured Results

Table 3 gives the power consumption with the “brute force approach‟ and the optimized one.

Approach	Power
Brute force approach	1.05 W
Optimized approach	0.64 W
Gain	0.41 W (39%)

Table 3 – Transcoder power measurements

---

[1] See http://www.theenergydetective.com/downloads/TEDProCommercialSpecifications.pdf

4.1.5 Shape w.CNOM

Partner: **Sensative**

4.1.5.1 Description

The development of the CNOM technical system is done, and what now follows is to improve the very basic algorithm used so far to verify the technical system.

The overall goal with the CNOM system in Convince has been defined to lower the use of cellular connectivity for mobile devices by enabling the transfer of them to WiFi connectivity instead. This is obtained by increasing the availability of the WiFi-environment through optimal configuration.

Optimization of the current consumption will thus be evaluated through parameters describing the availability and capacity of WiFi communication, rather than the actual power consumption.

Currently, only the WiFi channel load is measured, but there may very well be other parameters defined ahead.

4.1.5.2 Parameters Measured

As described above, only the channel load is currently measured. At this stage, the algorithm is not suited for the CNOM optimization, which will follow ahead.

4.1.5.3 Measured Results

At this stage, there are no results from the algorithm optimization, only results that indicate the principles of the technical system.

4.2 Network

4.2.1 Energy Efficient Routing

Partners: **CEA and GreenSpector**

4.2.1.1 *Description*

We have integrated the CEA testbed where we have a SDN network (i.e. SDN controller, three OpenFlow switches and two gateways). We have installed in each OpenFlow switch the Greenspector probe to measure the power consumption and to store the results in CSV files. After that, we provide these CSV files to the EXFO tool in order to visualize them. In addition, we have hooked up all OpenFlow switches into the BTH probe. The SDN controller is hooked up, as well, into another BTH probe. The results of measurements are shown on the BTH Monitor Live.

In our testbed, we have implemented two routing approaches in the SDN controller:

- Conventional routing: upon receiving new traffic, the SDN controller computes the shortest path by using the hop count metric.
- GoGreen routing: upon receiving new traffic, the SDN controller computes a given number of shortest paths (in our case, it computes only two paths because the network is mall). After that, it selects, from these paths, the one with the lowest power consumption (which is given by the sum of power consumption related to network interfaces participating the path).

The measurement procedure is done as follows. First, we configure the controller to perform conventional routing. Then, we launch Greenspector probes in each OpenFlow switch. After that, we launch the video streaming through the testbed by using the VTT server and the terminal. When the measurements are done and the CSV files are generated, we shut down the SDN controller, configure it to do GoGreen routing and launch it again. We start Greenspector probes again and launch the video streaming. The measurement results are stored in new CSV files.

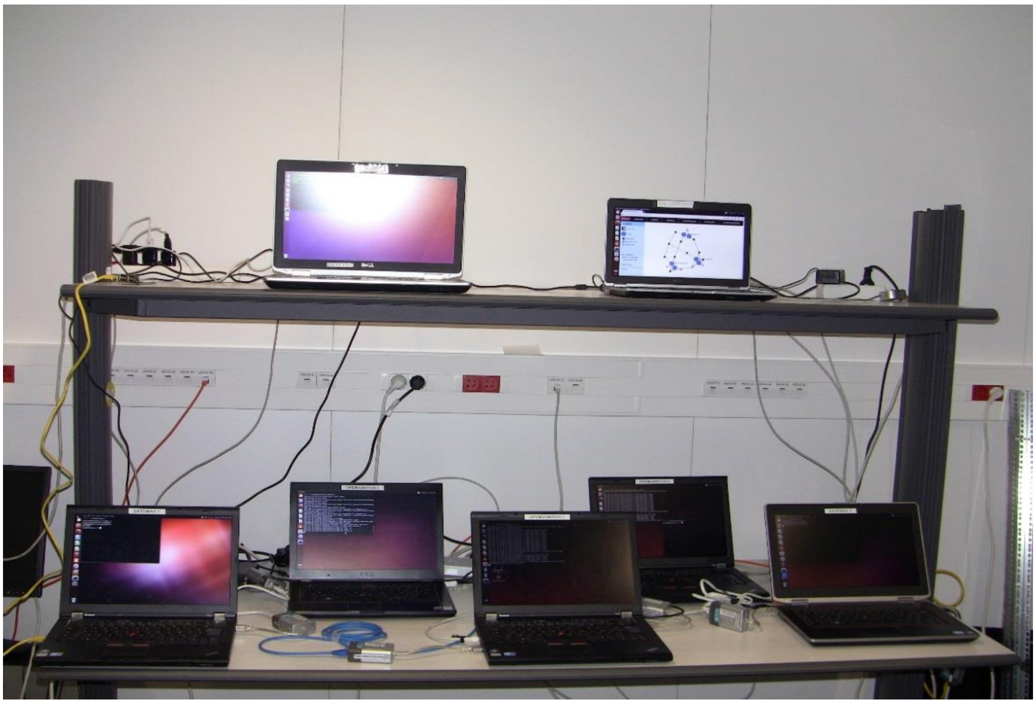


Figure 9: CEA Part in the Integrated Demonstrator

4.2.1.2 Parameters Measured

The instantaneous or average power consumption of the PC related to the openswitch software.

4.2.1.3 Measured Results

We have done several measurements with the Greenspector probes. However, these probes cannot reflect the energy gain that we get when the GoGreen routing approach is adopted. In fact, this algorithm considers the power consumption of all network interfaces power participating in the routing path. However, the Greenspector probe measures only the CPU power consumption in each OpenFlow switch and it cannot provide information about the power consumption related to each network interface.

We are now using the BTH probes, and the measurement results will be reported later.

Measured Results show with the EXFO UI

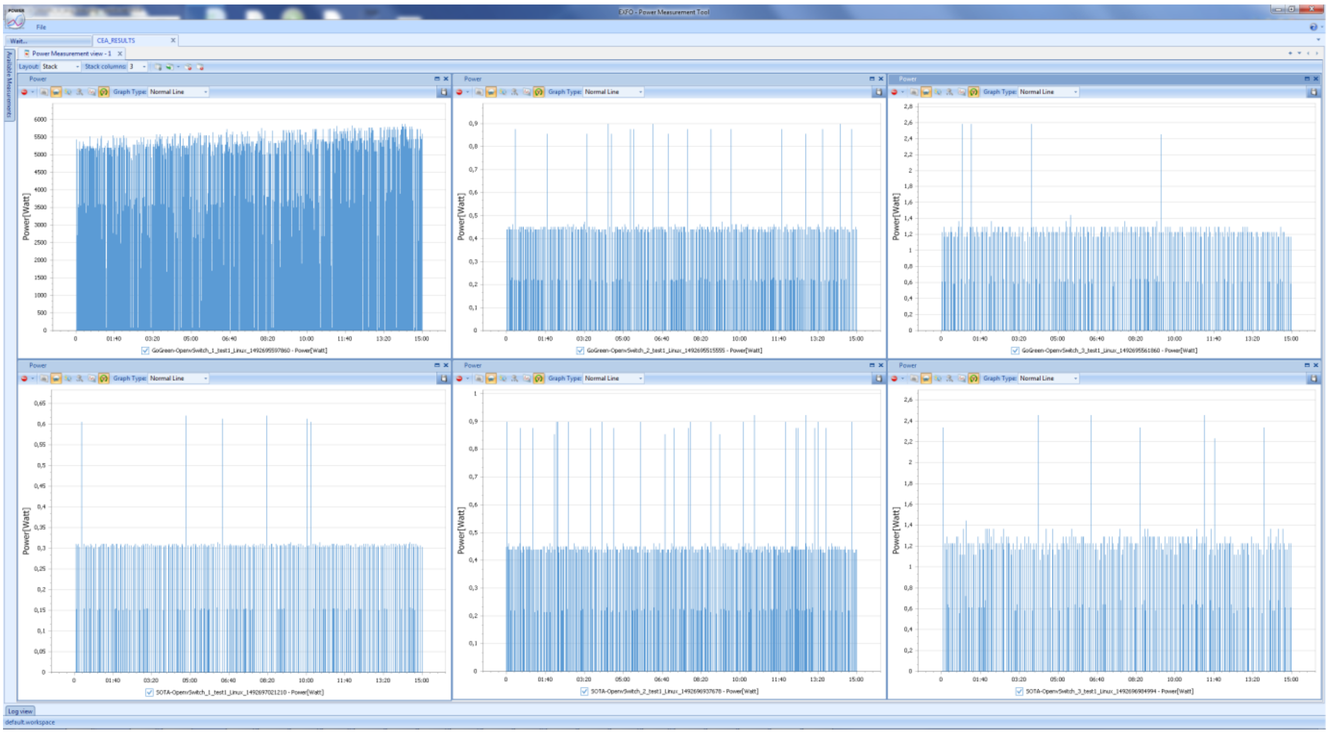


Figure 10: CEA Measurement Results

4.2.2 CDN

Partner: **IMT**

4.2.2.1 Description

The CDN evaluation module considers the CDN energy consumption methods available in literature to compute the consumed energy in a CDN. Different methods are suggested in the literature with variable number of parameters. We have compared the methods and chosen the most complete ones to use. Two following equations are used to compute the CDN's consumed energy:

$$E_{tot}{}^{CDN} = E_{tr}{}^{c} + E_{tr}{}^{e} + E_{st} + E_{sr}$$

Where $E_{tr}{}^{c}$, $E_{tr}{}^{e}$, E_{st}, E_{sr} are the amount of energy consumed in core, edge, storage, and server, respectively.

$$E_{tot} = E_{storage} + E_{server} + E_{synch} + E_{tx}$$

4.2.2.2 Parameters Measured

The module measures the amount of energy consumed by a CDN, which is given by the formula in the previous part considering different parameters such as number of hops, size and number of contents, number of surrogate servers, link's energy.

4.2.2.3 Measured Results

As there is no CDN provider in the project, this module only surveys the available methods for CDN evaluation and does not produce any numerical result based on a real scenario. However, by considering inputs for the mentioned values, this module is able to provide the output, which is the consumed energy in the CDN part of an end to end network.

4.2.3 Cable Network

Partner: **Teleste**

4.2.3.1 Description

Power efficiency figures for node and amplifiers are achieved through a combination of performance verification measurements and capacity calculation. Performance measurements that validate the maximum throughput capacity have been performed separately and no power measurement will take place at the review with integrated demo setup.

The separate test setup is chosen based on the experience from actual installations where such combination and its variations are considered typical, see the Figure below.

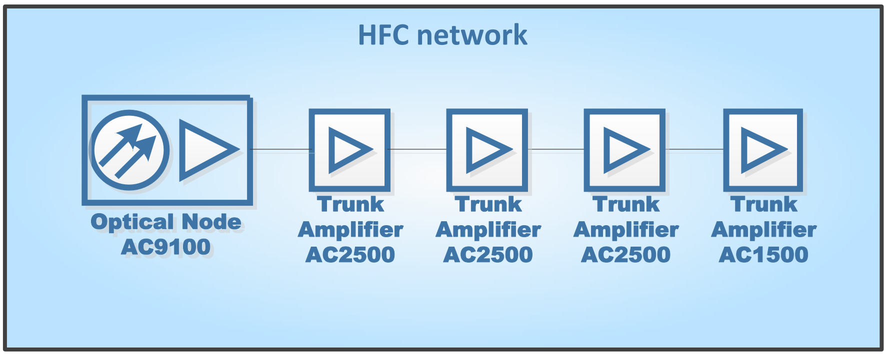


Figure 11: Separate DOCSIS 3.1 performance test setup

The review demonstrator's coax network consists of two DOCSIS3.1 amplifiers and a CMTS system. This setup is constructed to show the functionality and interoperability with the other Partners' systems, see block diagram below.

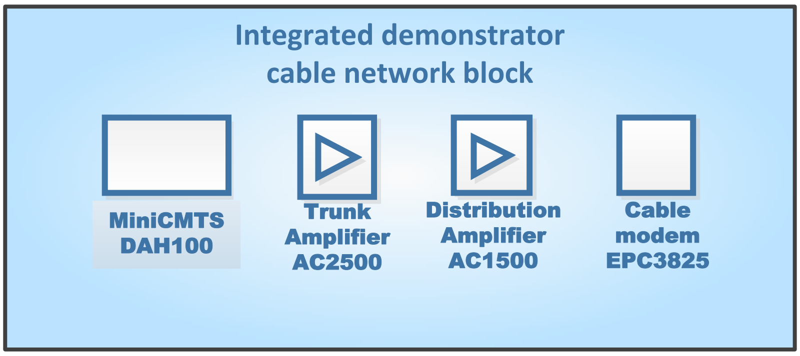


Figure 12: Integrated demonstrator cable network block

4.2.3.2 Parameters Measured

MER has been measured with two transmission scenarios using the separate test system setup. Based on the results power efficiency can be calculated using the maximum transmission capacity of the built devices.

The two spectrum allocation scenarios are described in CONVINcE deliverable D3.2.2.

4.2.3.3 *Measured Results*

The MER measurement report is attached to CONVINcE deliverable T5.3 on sheet "Cable network measurements".

Calculated power efficiency based on the validated performance is presented in table below.

Table 4: DOCSIS3.1 power efficiency comparison

Power efficiency	Node + 3* Trunk amplifiers + Distribution amplifier power consumption	Data throughput	Power efficiency
DOCSIS2.0 reference	**61W + 3*28W + 22W = 167W**	**4,9 Gbps**	**34,1 W/Gbps**
DOCSIS3.1 Scenario1	**65W + 3*38W + 26W = 205W**	**9,2 Gbps**	**22,3 W/Gbps**
DOCSIS3.1 Scenario2	**65W + 3*38W + 26W = 205W**	**11 Gbps**	**18,6 W/Gbps**

The results indicate that, by upgrading an HFC network from DOCSIS2.0 to DOCSIS3.1, the operator can achieve up to 35% improvement in power efficiency when adopting scenario 1 and up to 45% improvement when adopting scenario 2. This reveals that the higher the data capacity is the better power efficiency the system provides.

4.2.4 Edge Cloud

Partners: **Ericsson**

4.2.4.1 *Description*

The *Dash video burster by Ericsson* is a traffic shaping function that modifies the MPEG-Dash video streams by sending video segments (over HTTP) as relatively short and sparse bursts to terminal clients. I.e., the burster responds to clients' requests for segments only during short, recurring periods of time, with longer "silent" intervals in between. This saves some amount of energy in terminals because the wireless radio can be deactivated when no traffic is being sent. Thus, the related measurements focus is on the *terminal-side power consumption* (which affects the user experience, and it is in that sense more relevant), even though the bursting function itself is deployed into the edge cloud.

In general, the bursting function is an example of application-level energy saving mechanism that can be deployed, e.g., within a Docker container in the cloud. Deploying a traffic shaping function at the edge, close to the terminal, avoids subsequent in-network traffic shaping to be applied to packets in the stream.

4.2.4.2 *Parameters Measured*

Ericsson has measured power consumption in W at the terminal side, comparing two cases: Dash video streaming without and with stream bursting (the former is the baseline measurement and the latter is the optimized case).

4.2.4.3 *Measured Results*

Ericsson has measured the power consumption for receiving and viewing an on-demand Dash video stream (Tears of Steel, 1280x540 h264/avc1 5050 kbps video, aac/mp4a audio) on a 7" tablet: 1) without bursting and 2) with 8s bursts and 22s intervals (and a 20s initial burst; in this test set-up, these parameters allowed the client-side segment buffer to stay on a sufficiently high level all the time during streaming). The tablet was connected to a 2.4 GHz 802.11n Wi-Fi network. The Dash player in the terminal was the DASH-IF Reference Client (version 2.5.0, within Chrome on Android 5.0.1). An off-the-shelf KCX-017 USB power meter was used as the measurement tool. Without bursting, the average power consumption in the terminal was **1.95W**, and with bursting **1.88W** in this set-up. Thus, the improvement when using the bursting function was **~3.3%** (with identical stream quality).

4.2.5 WIFI Access Points

Partner: **TelHoc**

4.2.5.1 Description

TelHoc has measured power via a YoctoWatt chip embedded in the WiFi AP. This includes real time measurements displayed in a visualization tool as well as post analysis values provided to the EXFO tool. In addition, these measurement points will be integrated into the TelHoc software.

4.2.5.2 Parameters Measured

Power in (W) consumed by the WiFi AP for different use cases, including the usage of fountain and network coding under different wireless channel conditions, and different methods of encrypting/protecting the data.

4.2.5.3 Measured Results

It has been observed that there is a very strong correlation between the error rate of the wireless channel and the consumption of the wireless transmitter in the AP. This correlation is almost linear, with the consumption rising with increasing packet loss.

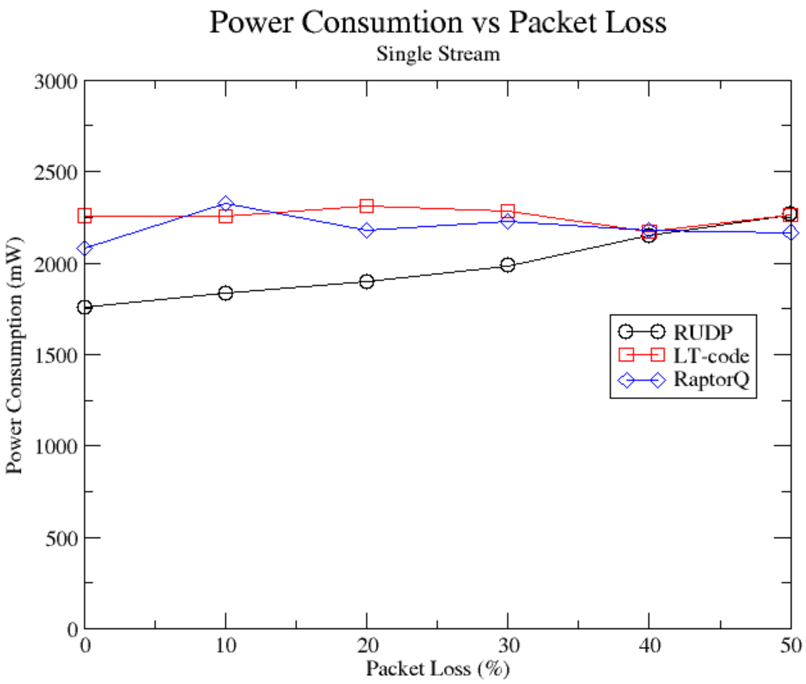


Figure 13: TelHoc measured results

However, when fountain coding is used, the consumption is almost flat and independent of the packet loss as long as the rate of the video is less than the available capacity. This means that there is an overhead to using fountain and network coding. While fountain and network coding are better at utilizing the available capacity and can accommodate more users with higher throughput streams, it comes at the price of higher power consumption for good channels. With a 10% packet loss and 10 receivers, the throughout gain is as much as 43%,

4.3 Terminals

4.3.1 Fixed Terminals

Partner: **Vestel**

Power measurement of Display units and STBs are demonstrated on fixed terminal part.

4.3.1.1 Description

Set Top Boxes and TV sets

Power measurement models of STB and Display Unit are covered on this demonstration.

In the power measurement model of STB, Active Mode processing (*See D4.1.1, Section 5.2.1 for details*) power of main IC is calculated first, and then the power spent by HEVC and AVC video decoding are measured.

On Display Unit side, during video playback, power consumption of it is calculated.
All these setups are demonstrated for both non-power optimized device and power optimized device.

The demonstration setup of power measurement of STB's and Display units are constructed as seen in the figure below. HEVC and AVC coded videos are included on server. Since measure the power consumption of Set Top Box during video decoding, STB is connected to PMT. At the same time, the Display Unit is also connected to PMT to gather measurement data. Measured average and instant power consumption values are transmitted from PMT to PC to evaluate them.

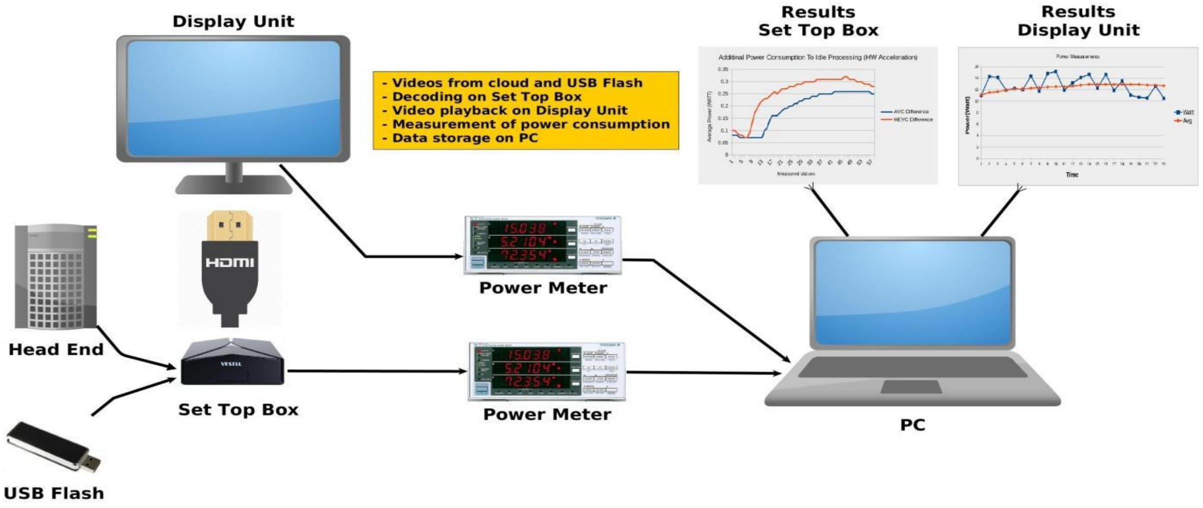


Figure 14: Fixed terminal power measurement setup

4.3.1.2 Parameters Measured

Inputs:
Video playback measurements are made by using AVC and HEVC encoded videos, which have properties as below.

- 1280x720p
- I-period 48, GOP 8
- 4.2 Mbps
- 50fps
- AVC-avg-PSNR 40.9 dB
- HEVC-avg-PSNR 43.3 dB

Outputs:

Average and instant power consumption values are received from PMT by PC over serial interface. Output values from the power meter tool towards to PC are

- Instant voltage
- Instant current
- Instant power consumption
- Average power consumption.

A script file has been coded to do two things. One is receiving data from PMT to PC over serial interface and the other is passing all received data into excel file.

4.3.1.3 Measured Results

The table below shows the power consumption results according to the power measurements models of STB Unit by playing video playbacks. The key point is that the Active Mode processing power of main IC is calculated first, and then the power spent by video decoding is measured. The decoding process is done by hardware block and software separately.

Process	Total Average Power	Percentage of video processing	Power Reduction
No video decoding	10.89 W	-	-
HEVC Soft-decoding	12.11 W	%10.10	-
HEVC Hard-decoding	11.23 W	%2.99	%7.26

Table 5: Power measurement results of STB

The next table shows the improvement of power consumption of 43’ Display Unit.

Size	**Standard Power**	Improved Power	Power Reduction
43’	56 W	36 W	%35.71

Table 6: Improvement of power consumption

4.3.2 Mobile Terminals

4.3.2.1 Description

Mobile Terminal (Scenario 4): Partners: **EXFO and University of Oulu**

This scenario is about video session power consumption estimation in different networks (3G/4G) using EXFO probes and Power consumption model of University of Oulu. The results are visualized in the EXFO Probe User Interface.

The EXFO probe analyses the video data of 3G/4G networks and provides network attributes needed for power consumption measurement by using configurable power consumption profiles. The analysis is post processing. Network attributes are provided in CSV format.

The University of Oulu power consumption model calculates an estimation for the terminal energy consumption of a communication scenario, based on the traffic profile of the scenario and the node energy profile. Pre-defined energy profiles, based on empirical measurements, are used to generate power consumption estimation for an application scenario.

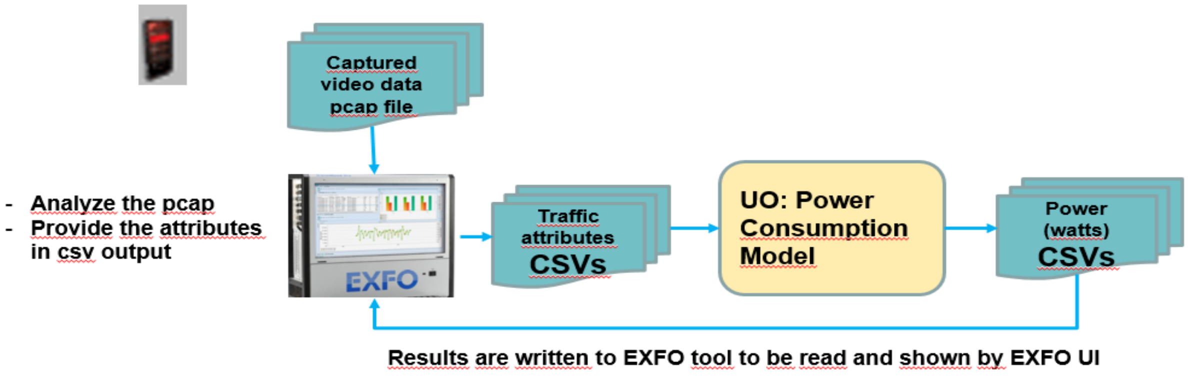


Figure 15: Mobile Terminal scenario 4 Block Diagram

The resulting mobile terminal power consumption may also be measured by using the power measurement tool developed by **Sony Mobile**. This tool captures power consumption characteristics based on the input source, which for a mobile terminal typically is on battery supply level. The results are stored in CSV files. The power consumption tool developed by Sony mobile can also provide correlation between instantaneous power consumption and debug logs from the terminal in order to characterize the power consumption from different use case specific events during the video streaming session.

Mobile Terminal (Scenario 6): Partners: **VTT**

This scenario studies the reduction of power consumption for the on-demand video streaming application with focus on Apple MacBook Pro 15 (late 2016) high-performance mobile terminal. The aim is to measure the power consumption of video processing with Intel Power Gadget tool in terminals playing back pre-encoded MPEG-DASH HD video streams from the VTT video source through the network and compare the baseline situation employing a popular off-the-shelf DivX video player with default settings assuming maximal H.264/AVC coding to the CONVINcE achievements employing an advanced GPAC Osmo player reconfigured and recompiled for low power consumption, executed in power-savvy execution settings and assuming maximal H.265/HEVC coding with the same objective video quality. The results of the power consumption and bitrate (video size) measurements are transferred to EXFO Travel Hawk Pro portable wireless network troubleshooting tool for visualization.

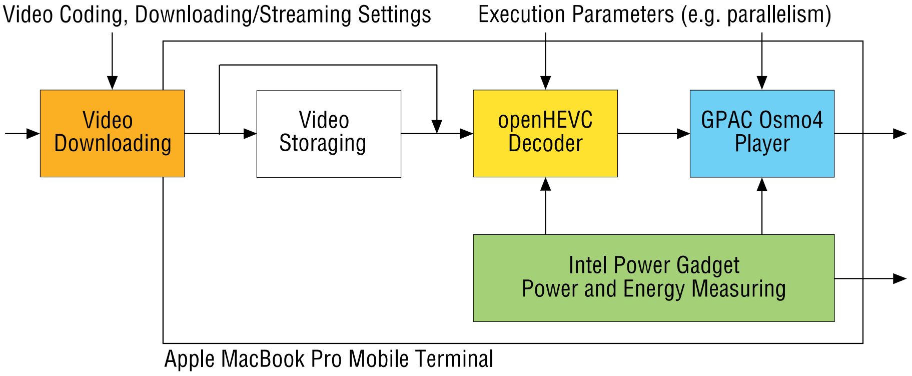


Figure 16: Mobile Terminal Scenario 6 CONVINcE setup.

4.3.2.2 Parameters Measured

Mobile Terminal (Scenario 4): Partners: **EXFO and University of Oulu**

- **INPUT:**
    - Captures video data pcap from Monsoon tool for different communication channels for 3G network.
    - Input to EXFO user interface is csv containing Timestamp [Seconds]; Power[Watt]; Description [Text]
- **OUTPUT:**
    - CSV file containing analyzed traffic data
    - power measured in watts with time duration

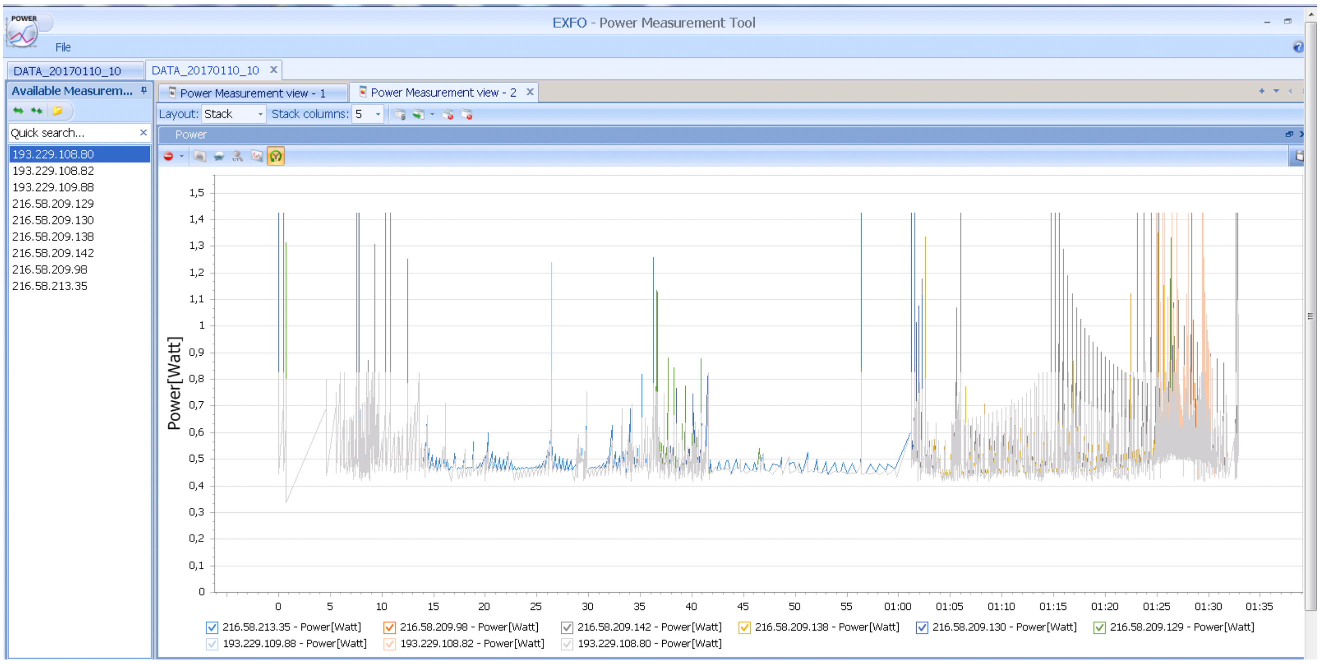
Mobile Terminal (Scenario 6): VTT

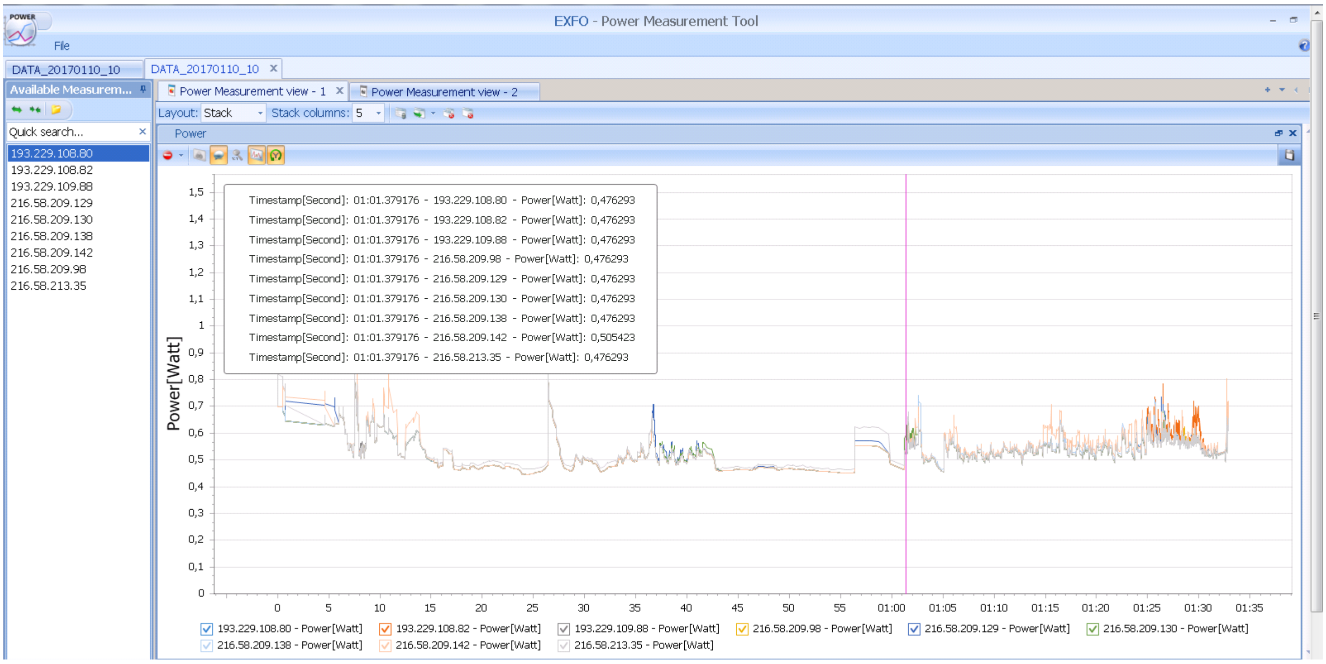
- **INPUT:**
    - Video stream encoded with either H.264/AVC or H.265/HEVC using the same objective quality in terms of the Peak Signal to Noise Ratio (PSNR).
- **OUTPUT:**
    - CSV file containing measurement results in format to be illustrated in EXFO user interface. The format is Comma Separated Values (CSV): Timestamp [Seconds]; Power[Watt]; Description [Text]
    - The amount of data transferred from the head end in bytes.

4.3.2.3 Measured Results

Mobile Terminal (Scenario 4): Partners: EXFO and University of Oulu

Power Measured: from **0.35watts – 1.42 watts** for video data in 3G Network.





```
Timestamp[Second]: 01:01.379176 - 193.229.108.80 - Power[Watt]: 0,476293
Timestamp[Second]: 01:01.379176 - 193.229.108.82 - Power[Watt]: 0,476293
Timestamp[Second]: 01:01.379176 - 193.229.109.88 - Power[Watt]: 0,476293
Timestamp[Second]: 01:01.379176 - 216.58.209.98 - Power[Watt]: 0,476293
Timestamp[Second]: 01:01.379176 - 216.58.209.129 - Power[Watt]: 0,476293
Timestamp[Second]: 01:01.379176 - 216.58.209.130 - Power[Watt]: 0,476293
Timestamp[Second]: 01:01.379176 - 216.58.209.138 - Power[Watt]: 0,476293
Timestamp[Second]: 01:01.379176 - 216.58.209.142 - Power[Watt]: 0,505423
Timestamp[Second]: 01:01.379176 - 216.58.213.35 - Power[Watt]: 0,476293
```

Figure 17: Mobile Terminal Scenario 4 Results

Mobile Terminal (Scenario 6): Partners: **VTT**

- **Baseline**: Average power consumption **7.6 W** for 141.9 seconds, total energy **1079 J**, transferred data **39 MB**.
- **CONVINcE:** Average power consumption **3.6 W** for 68.7 seconds, total energy **246 J**, transferred data **23 MB**.
- **Total improvements:** Energy reduction **77.2 %**, data reduction **41.0%**.

Note that the behavior of the baseline player is highly inefficient in this kind of streaming due to extensive and time consuming preloading of the video data for playback and number of error messages after the video playback. In the case of local playback, the energy reduction was **46.1 %** with the same video clip as reported in deliverable D4.1.2.

Note also that this does not include the power consumed by the external GPU of the MacBook Pro terminal, which we estimate to be 3.7 W in average. If this is added to the power of the baseline, the achieved energy improvement in video processing raises up to **81.5 %**.

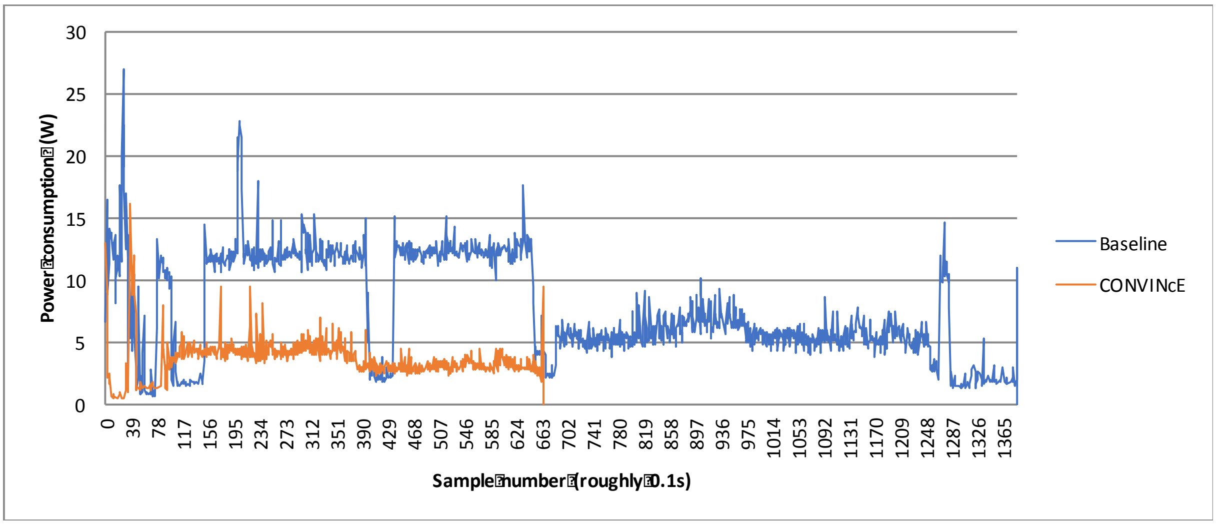


Figure 18: Mobile Terminal Scenario 6 Results

5 END-TO-END MEASUREMENTS AND RESULTS

This section describes the different end-to-end measurement data paths defined under the scope of CONVINcE and the results for power consumption measurements performed for each data paths.

Test-Bed

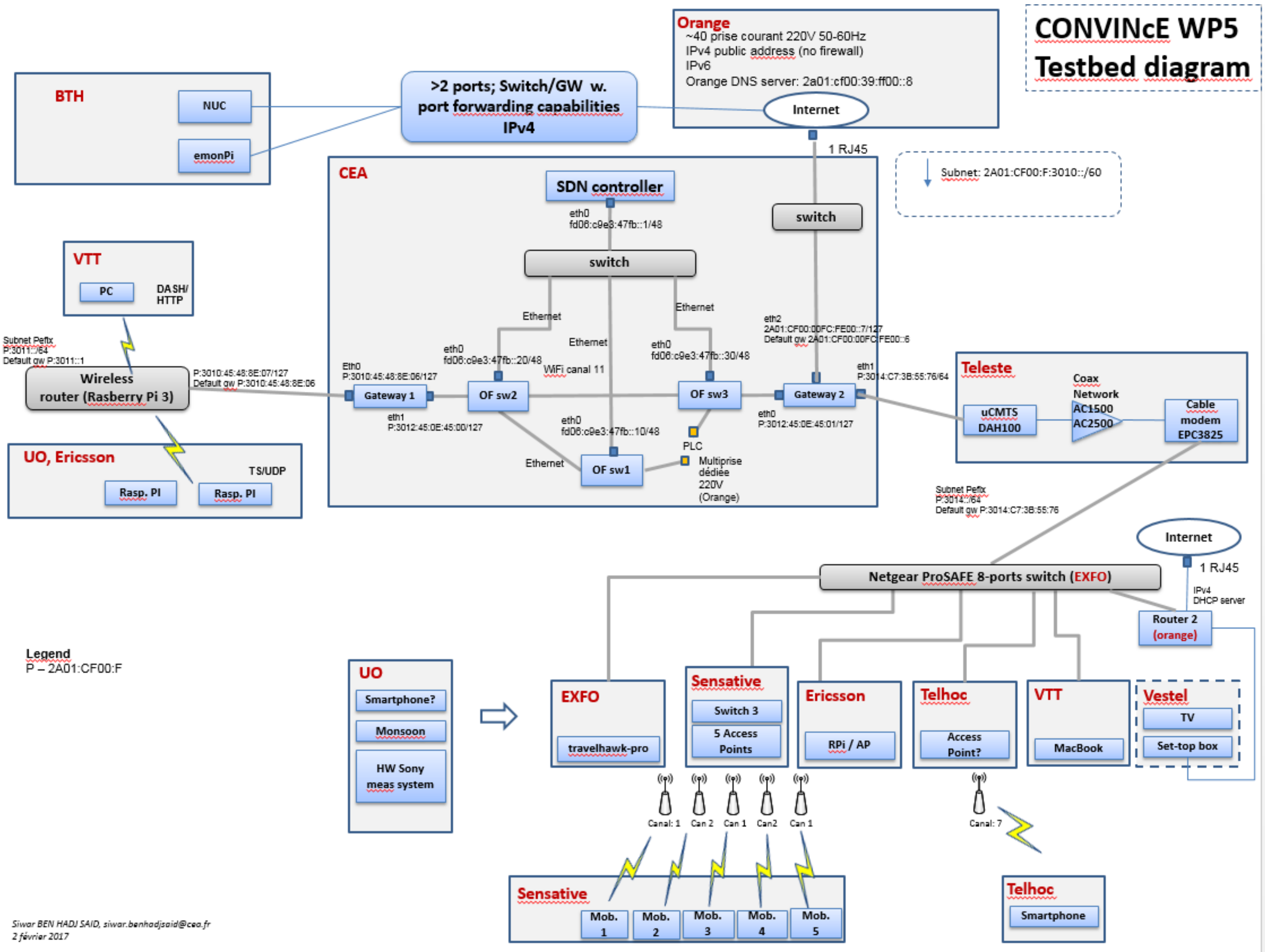


Figure 19: Integrated Test-Bed

https://bscw-convince.celticplus.eu/bscw/bscw.cgi/d20710/CONVINcE%20T%205.3_Final%20Integrated%20Demonstration%20Use%20Cases_v3.xlsx

5.1 Data Path 1 for Video on Demand

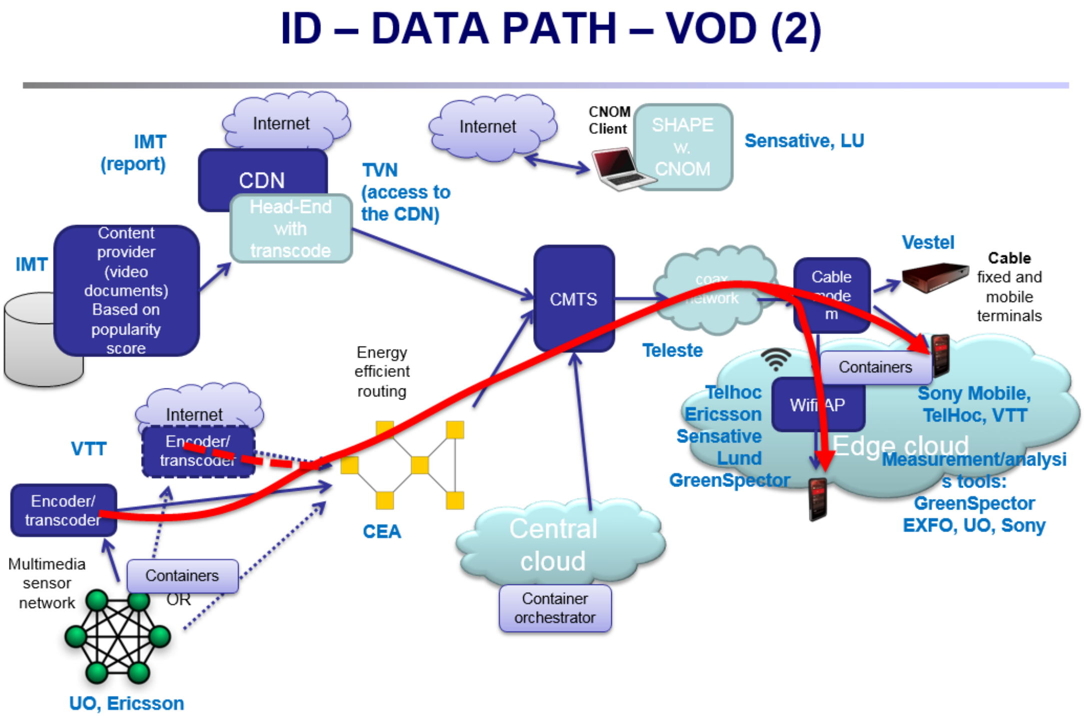


Figure 20: Data Path – Video on Demand

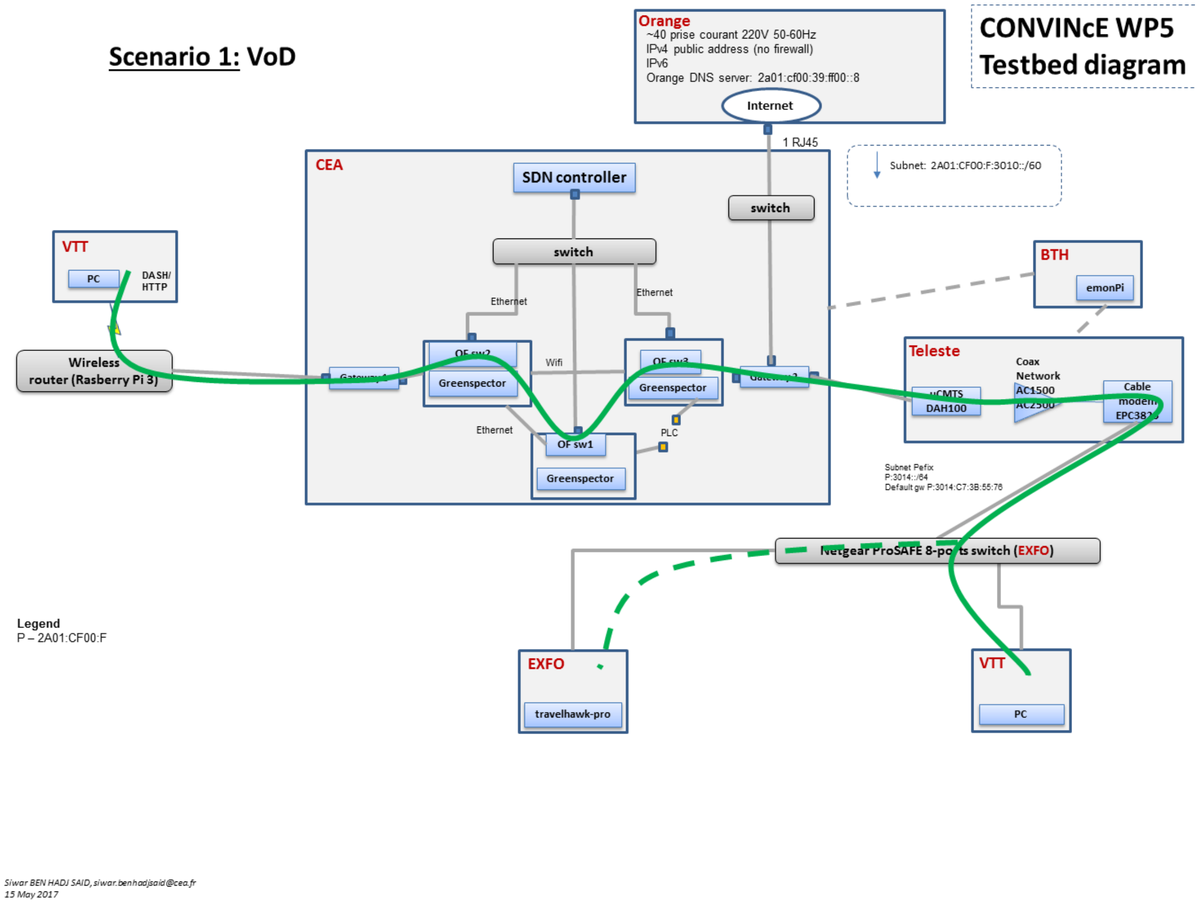


Figure 21: Test-Bed Video on Demand

5.1.1.1 Description

Partner: **VTT**

In this demonstrator, the pre-encoded video clips for H.264/AVC and H.265/HEVC coding with identical objective quality are streamed from the VTT head end server through the network to VTT's MacBook Pro 15 (late 2016) terminal (see scenario 6 of Section 4.3.2). The power consumption and the amount of video data are measured. The results of the measurements are converted to the Comma Separated Values (CSV) format and sent to EXFO TravelHawk Pro portable wireless network troubleshooting tool for visualization.

Partner: **CEA**

In this scenario, the CEA testbed ensures the traffic routing from VTT server to the VTT client. The CEA testbed is composed of two Linux-based PCs acting as gateways, 3 Linux-based PCs acting as OpenFlow switches and 1 Linux-based PC acting as an SDN controller. In the OpenFlow switches, we installed OpenvSwitch and NEONd softwares. NEONd is a component of the CEA in-house SDN software. This component allows a remote network interface configuration in OpenFlow switches. In the SDN controller, we installed the NEON controller, which is a component of the CEA in-house SDN software. The power consumption in OpenFlow switches and SDN controller are measured using the BTH tool.

Partner: **Teleste**

The coax network is constructed to show newly created DOCSIS3.1 amplifiers as part of the demo setup and other partner's demonstrations. A video stream provided by the distributor is conveyed through coaxial network to the subscriber. No measurement takes place here. See block diagram below.

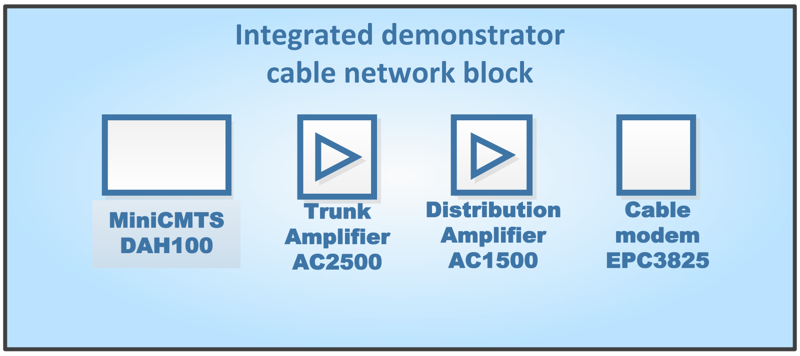


Figure 22: Coax network block diagram

New DOCSIS3.1 compatible devices enable 11Gbps data throughput capacity with relatively small power consumption increase, which results in an improvement of up to 45% in power efficiency in Cable TV networks when compared to DOCSIS2.0 installations.

Partner: **EXFO**

In this integrated demonstration scenario **EXFO** probes captured live network traffic for devices under test with different communication channels providing video data.

- Live video streaming data is send from VTT system and it was captured by EXFO TravelHawk Pro. Path used -> CEA system -> Teleste cable modem -> Netgear Switch -> TravelHawk Pro
- Captured data was analyzed and attribute csv file is provided to the power consumption model of University of Oulu installed and running in TravelHawk Pro.
- The output results are shown in EXFO UI.

Figure 23: Integrated demonstration EXFO probe

5.1.1.2 Parameters Measured

Partner: **EXFO**

Power in watts for the captured video data.

Partner: **VTT**

- Power consumption and energy of the video processing in the VTT's mobile terminal including the CPU (four general purpose Core i7 processor cores and on-chip GPU) but excluding the external GPU.
- Amount of data transferred to the terminal.

Partner: **GreenSpector**

Power consumption in watts of the CEA openflow (OF) switches.

5.1.1.3 Measured Results

Partner: **EXFO**

Power measured: from 0.46 watts -1.43 watts for the full 12 minutes live video data for 3G radio link.

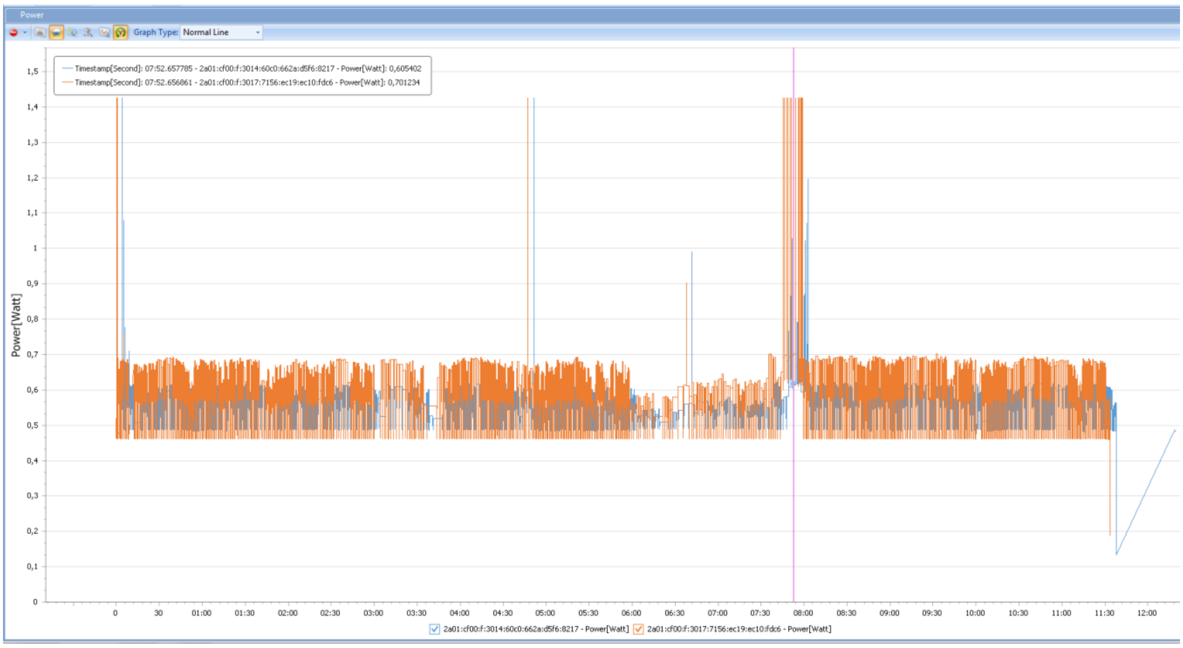


Figure 24: Measurement Results EXFO probe

Partner: **VTT**

- **Baseline**: Average power consumption **7.6 W** for 141.9 seconds, total energy **1079 J**, transferred data **39 MB**.
- **CONVINcE:** Average power consumption **3.6 W** for 68.7 seconds, total energy **246 J**, transferred data **23 MB**.
- **Total improvements:** Energy reduction **77.2 %**, data reduction **41.0%**.

Note that the behavior of the baseline player is highly inefficient in this kind of streaming due to extensive and time consuming preloading of the video data for playback and number of error messages after the video playback. In the case of local playback, the energy reduction was **46.1 %** with the same video clip as reported in deliverable D4.1.2.

Note also that this does not include the power consumed by the external GPU of the MacBook Pro terminal, which was estimated to be 3.7 W in average. If this is added to the power of the baseline, the achieved energy improvement in video processing raises up to **81.5 %**.

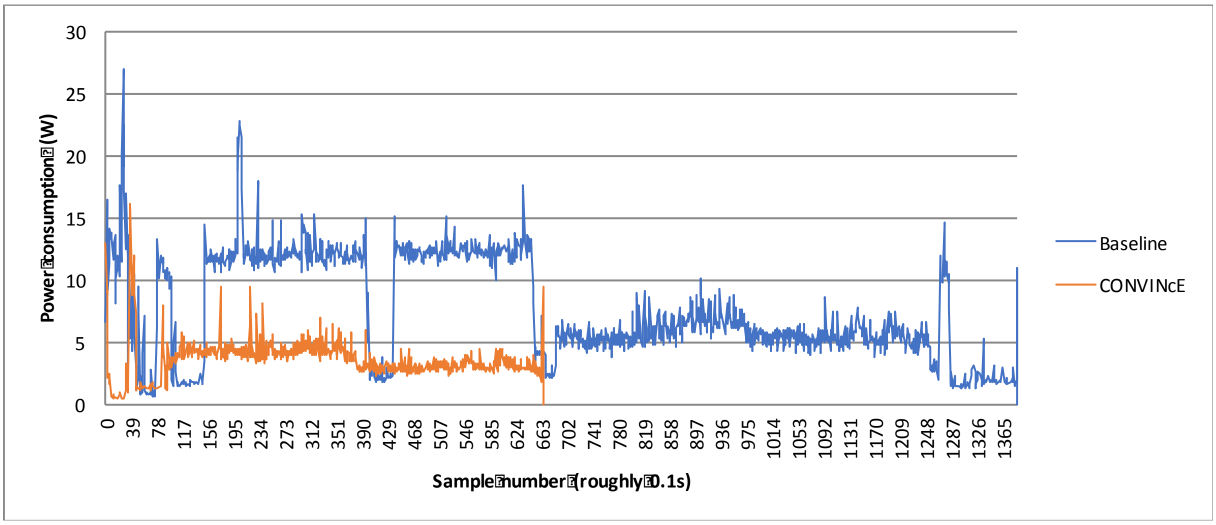


Figure 25: Power consumption in VTT's mobile terminal

5.2 Data Path 2 for Video Surveillance Network

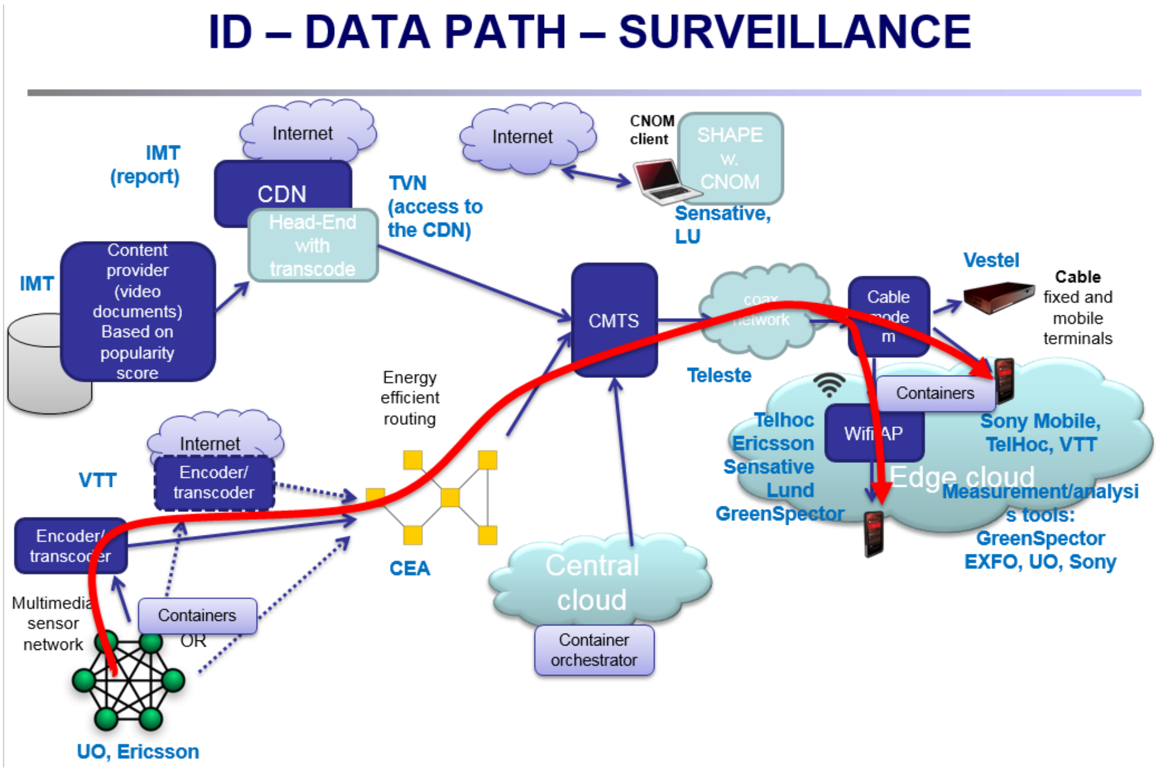


Figure 26: Data Path – Video Surveillance

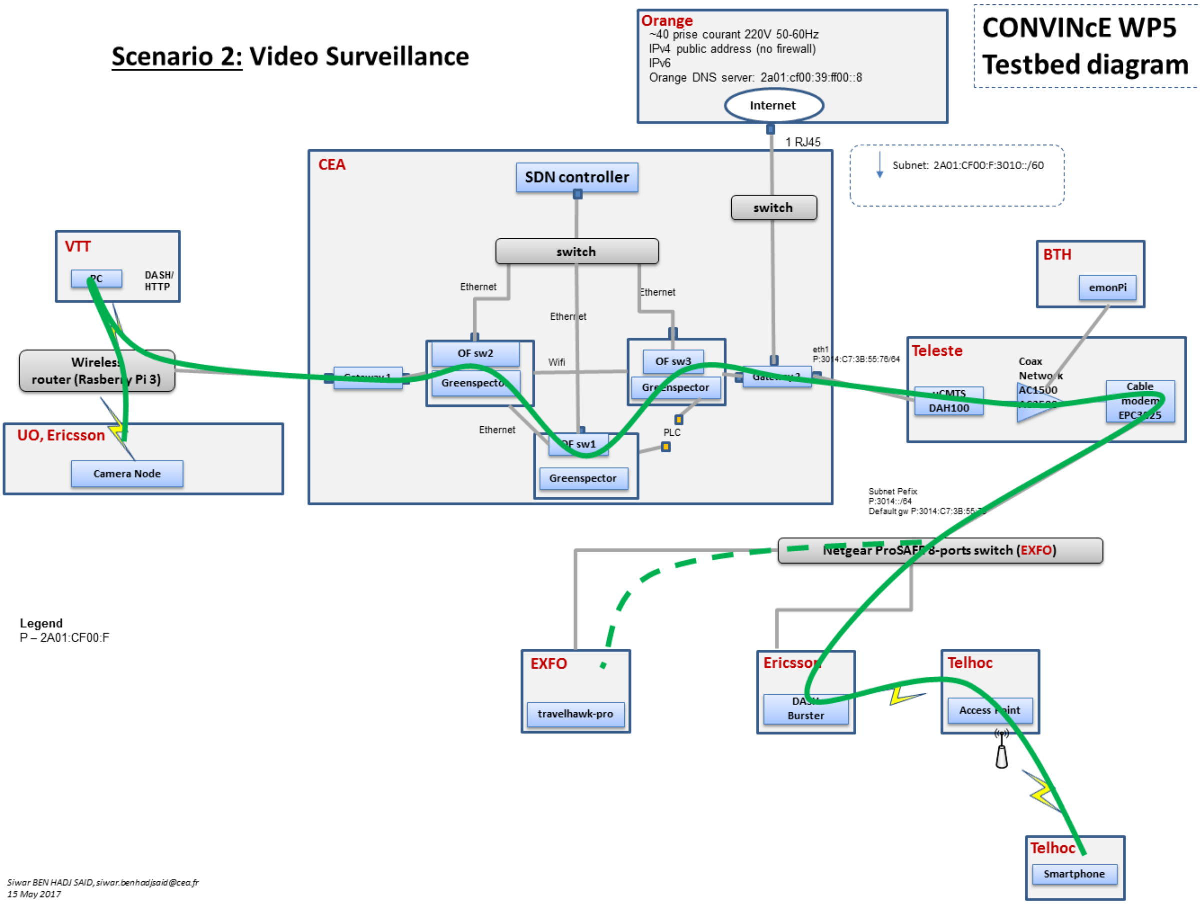


Figure 27: Test-Best Video Surveillance

5.2.1.1 Description

In the surveillance data path, the camera nodes (on the left) stream surveillance video to an encoder/transcoder node that translates the H264 video stream into HEVC encoded and MPEG-DASH encapsulated video. The encoder sends the traffic over the SDN-managed energy efficient routing network over IPv6.

Then the traffic passes the coax network via the cable modem to the Wifi Access Point (AP). Before routing the traffic to the terminals (the phones on the right), the Wifi AP can manipulate MPEG-DASH streams by two means. First, the Wifi AP can send the traffic in bursts so that the terminal can save energy while its radio is sleeping during bursts (this functionality is implemented by using containers). Second, the Wifi AP can use Fountain encoding to encode redundancy into traffic, so that the terminal can save energy in lousy networks.

Partner: **Ericsson / University of Oulu**

The energy consumption of the camera node is measured using a Monsoon power monitor.

Partner: **VTT**

In this demonstrator, the pre-encoded video clips for H.264/AVC and H.265/HEVC coding with identical objective quality are streamed from the VTT head end server through the network to VTT's MacBook Pro 15 (late 2016) terminal (see scenario 6 of Section 4.3.2). The power consumption and the amount of video data are measured. The results of the measurements are converted to the Comma Separated Values (CSV) format and sent to EXFO TravelHawk Pro portable wireless network troubleshooting tool for visualization.

Partner: **CEA**

In this scenario, the CEA testbed ensures the traffic routing from VTT server to the VTT client. The CEA testbed is composed of two Linux-based PCs acting as gateways, 3 Linux-based PCs acting as OpenFlow switches and 1 Linux-based PC acting as an SDN controller. In the OpenFlow switches, we installed OpenvSwitch and NEONd softwares. NEONd is a component of the CEA in-house SDN software. This component allows a remote network interface configuration in OpenFlow switches. In the SDN controller, we installed the NEON controller, which is a component of the CEA in-house SDN software. The power consumption in OpenFlow switches and SDN controller are measured using the BTH tool.

Partner: **Teleste**

The demonstrator’s coax network is constructed to show newly created DOCSIS3.1 amplifiers as part of the demo setup and other partner’s demonstrations. A video stream provided by the distributor is conveyed through coaxial network to the subscriber. No measurement takes place here.

New DOCSIS3.1 compatible devices enable 11Gbps data throughput capacity with relatively small power consumption increase, which results in an improvement of up to 45% in power efficiency in Cable TV networks when compared to DOCSIS2.0 installations.

Partner: **TelHoc**

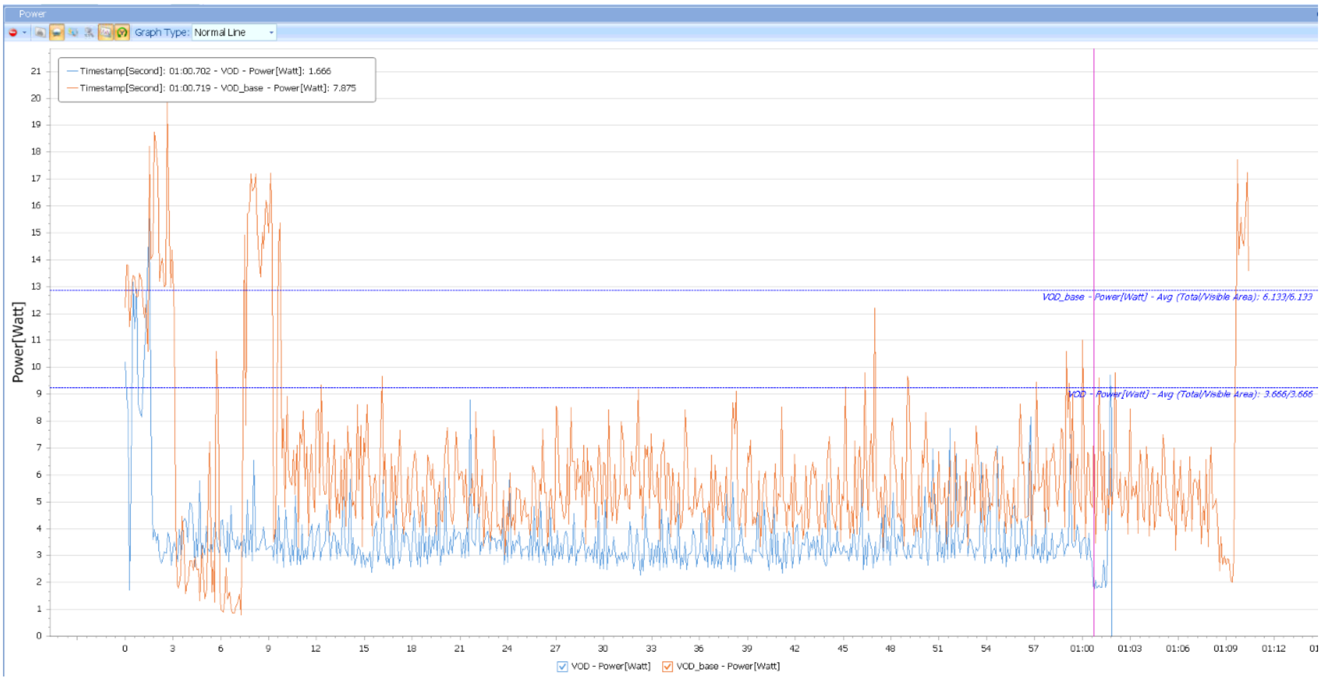
TelHoc measures the power consumption of the wireless AP using the Yoctopuce module and in the mobile terminal using measurement chipsets provided in certain Android phones. For the wireless AP this is measured directly from the power source, while in the mobile phone it is measured power the embedded chipset whose data is provided to the linux kernel for use in the demonstration application.

5.2.1.2 Parameters Measured

EXFO UI is used for visualizing the measured results of the scenario. Also, live visualization for the power consumption is visible from the VTT encoding server. Monsoon power monitor is used for visualizing the power consumption of the integrated camera node of University of Oulu and Ericsson.

5.2.1.3 Measured Results

Showing the power consumption results provided by VTT. The Power measured: **from 1watt-20watts**



Graph Type: Normal Line

Timestamp[Second]: 01:00.702 - VOD - Power[Watt]: 1.666
Timestamp[Second]: 01:00.719 - VOD_base - Power[Watt]: 7.875

VOD_base - Power[Watt] - Avg (Total/Visible Area): 6.133/6.133
VOD - Power[Watt] - Avg (Total/Visible Area): 3.666/3.666

Figure 28: Measurement results visualized in EXFO UI

5.3 Data Path 3 for Virtualized Video Surveillance

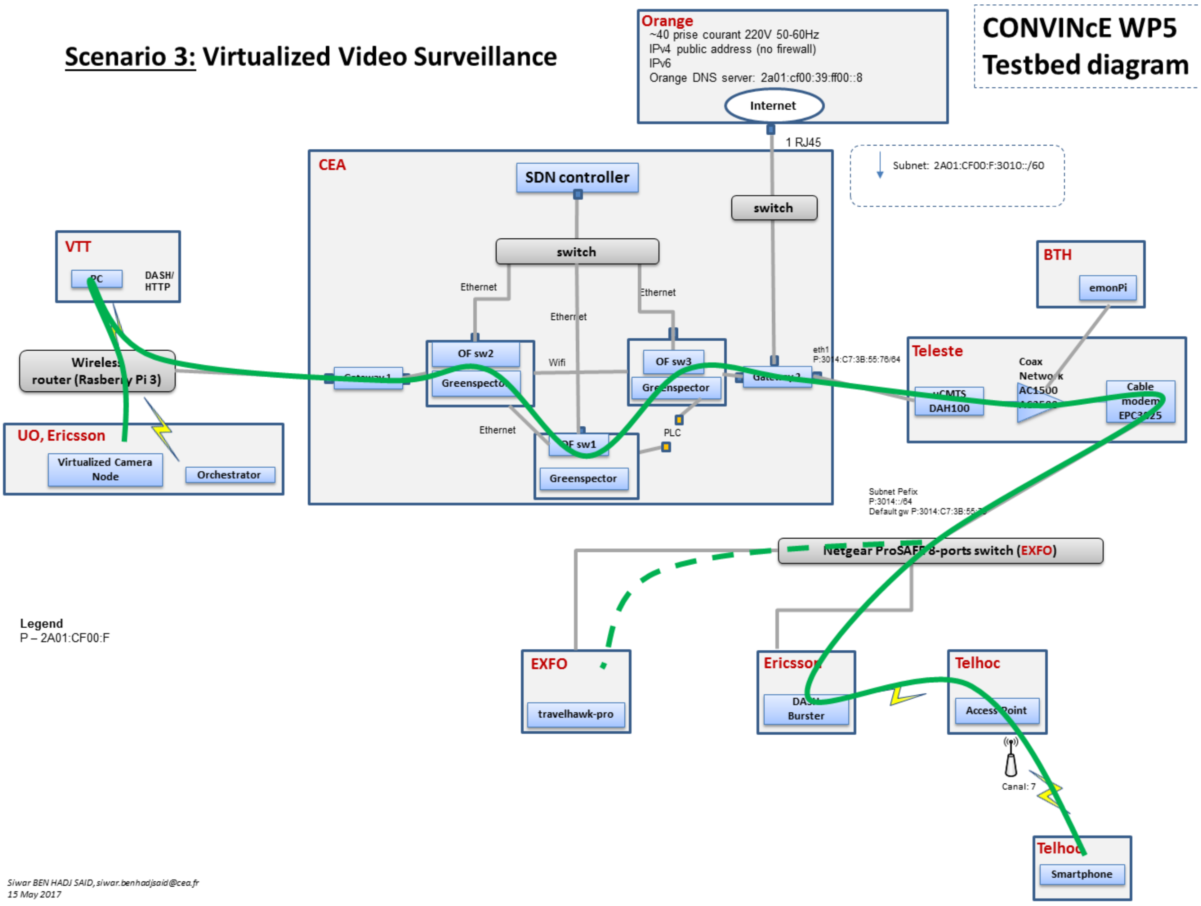


Figure 29: Test-Bed Virtualized Video Surveillance

5.3.1.1 Description

This scenario is a second variant of the video surveillance camera scenario. Instead of using a physical camera sensor, in this data path, we use a virtualized camera node that is managed by an orchestrator. In this data path, the containerized multimedia sensors (on the left) stream surveillance video to an encoder/transcoder node that translates the H264 video stream into HEVC encoded and MPEG-DASH encapsulated video. The encoder sends the traffic over the SDN-managed energy efficient routing network over IPv6.

Then the traffic passes the coax network via the cable modem to the Wifi Access Point (AP). Before routing the traffic to the terminals (the phones on the right), the Wifi AP can manipulate MPEG-DASH streams by two means. First, the Wifi AP can send the traffic in bursts so that the terminal can save energy while its radio is sleeping during bursts (this functionality is implemented by using containers). Second, the Wifi AP can use Fountain encoding to encode redundancy into traffic, so that the terminal can save energy in lousy networks.

Partner: **TelHoc**

TelHoc measures the power consumption of the wireless AP using the Yoctopuce module and in the mobile terminal using measurement chipsets provided in certain Android phones. For the wireless AP this is measured directly from the power source, while in the mobile phone it is measured power the embedded chipset whose data is provided to the linux kernel for use in the demonstration application.

Partner: **Ericsson / University of Oulu**

The energy consumption of the camera node is measured using a Monsoon power monitor.

5.3.1.2 Parameters Measured

EXFO UI is used for visualizing the measured results of the scenario. Also, live visualization for the power consumption is visible from the VTT encoding server. Monsoon power monitor is used for visualizing the power consumption of the integrated camera node of University of Oulu and Ericsson.

5.3.1.3 Measured Results

The figure below shows an example run of a power measurement that compares the orchestrated (i.e. virtualized) version of the camera node to the non-orchestrated one (described Data Path 2 for video surveillance camera). It is worth mentioning that this particular measurement did not use VTT's encoder, but the power consumption would be the same since the camera node is the source of video stream.

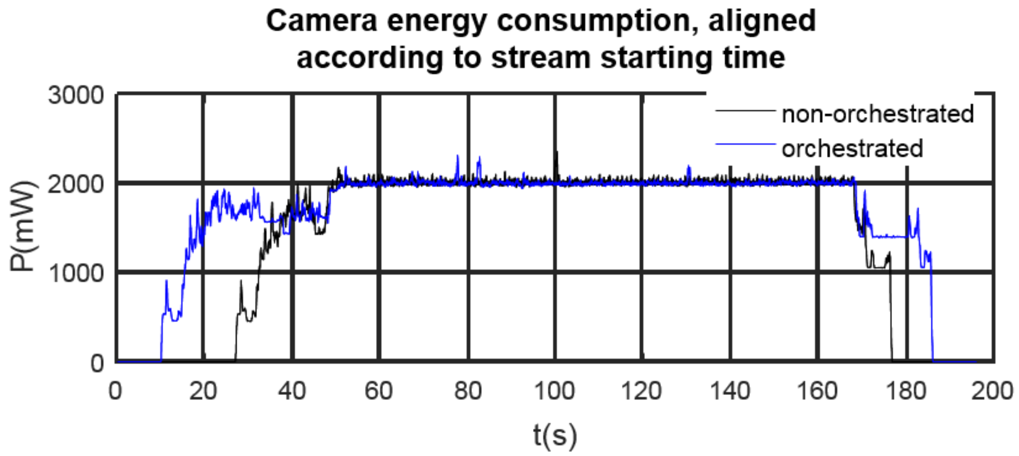


Figure 30: Camera Energy Consumption

The measurement shows that the orchestrated camera node (blue line) requires more time to boot and power-off, so in total it consumes more energy. The virtualization delays the start-up of the camera by roughly 20 seconds and powering off by 15 second delay, which naturally increases energy consumption (40 % for boot and 74 % for power-off). However, the relative overhead of boot-up/shut-down decreases when the recording time increases.

The overhead of the virtualization for the camera node for the actual video streaming part (from 50 to 170 seconds on the x-axis) was negligible. During streaming, the virtualized version consumed even a slightly less energy (0 – 2 percent). This could be accounted by the margin of error (standard deviation was a bit high, 250 mW), but we are still investigating the root cause for this.

5.4 Data Path 4 for Content popularity

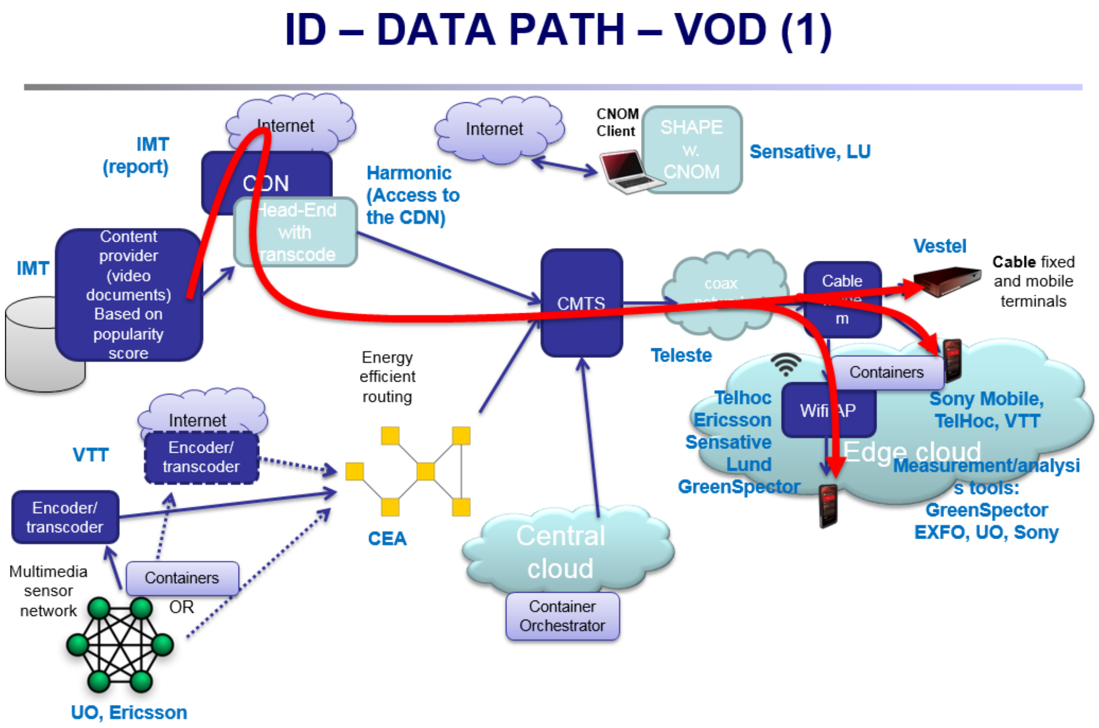


Figure 31*: Data Path –Video on Demand*

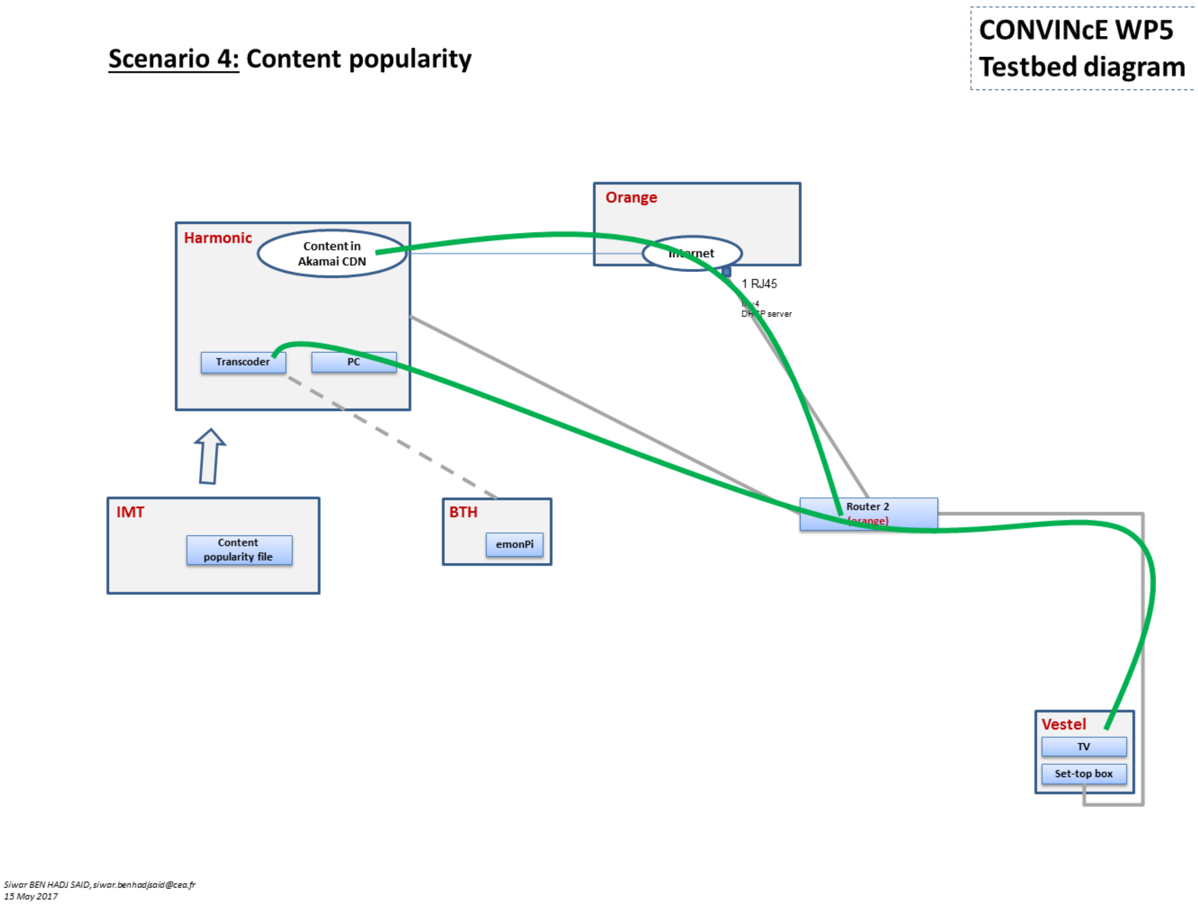


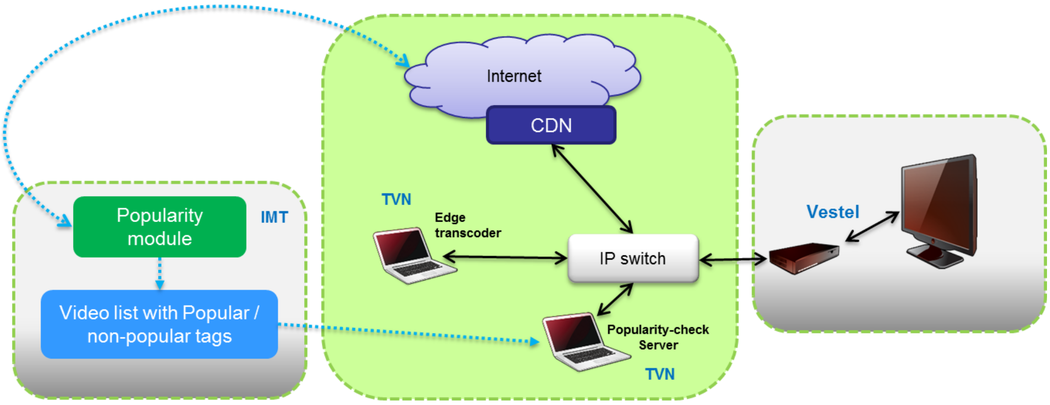
Figure 32: Test-Bed content popularity

5.4.1.1 Description

Partners: **Telecom SudParis, Harmonic, Vestel**

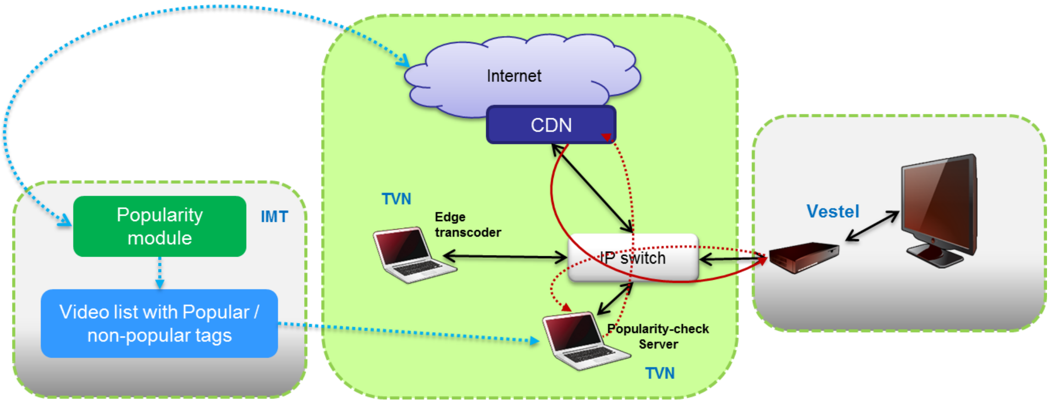
Ten videos, some popular and others non-popular, have been selected from the output of the popularity video streams module, which sorts the videos based on their popularity. These videos are decided to cache in the CDN (popular contents) or in another server (non-popular contents).

Background task:



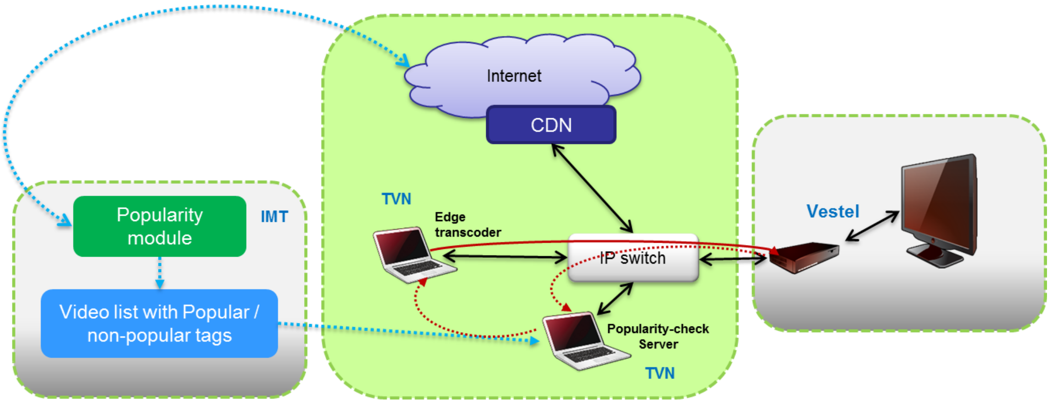
Content popularity is scanned over the Internet and the popularity video list is updated in the popularity server

Popular contents:



If a request for a popular content is sent by the STB, the content is asked to the CDN by the popularity server (red dotted arrows). Then, the CDN sends the popular content to the STB (red plain arrow).

Non-popular contents:



When the STB requests a non-popular content, the content is not requested to the CDN but the popularity server asks the transcoder (red dotted arrows) who provides the content (red plain arrow).

For non-popular contents, two configurations were analyzed: First one – called "brute force approach" - is achieved with a basic transcoder which fully decodes the content and then re-encode it. Second one is a smarter approach making use of metadata avoiding a full decode-encode operation and thus saving energy. It is fully described in deliverable D2.2.2, section 5.1.3.

5.4.1.2 *Parameters Measured*

Consumption is measured by monitoring the current and the voltage of the battery of the laptop used for transcoding, when this laptop is not connected to the mains. This is done by reading every second Linux registers giving the current and voltage values.

We tested an external tool to measure the consumption on mains when using the external power supply of the laptop, but it led to unreliable/unreproducible results and we decided to forget this method.

For a better accuracy, 12 transcoders are launched simultaneously. Consumption is measured for the two configurations and also when the laptop is "doing nothing" (no transcoding). Power consumption for one transcoder is given by:

$$\boxed{P_{1\ \text{transcode}} = (P_{12\ \text{transcode}} - \mathrm{P}_{\text{idle}})/12}$$

5.4.1.3 *Measured Results*

Configuration	Power consumption 1 transcoder	Gain
Brute force	1.05 W	
Smart with metadata	0.64 W	0.41 W (39%)

5.5 Data Path 5 for Optimizing crowded WiFi

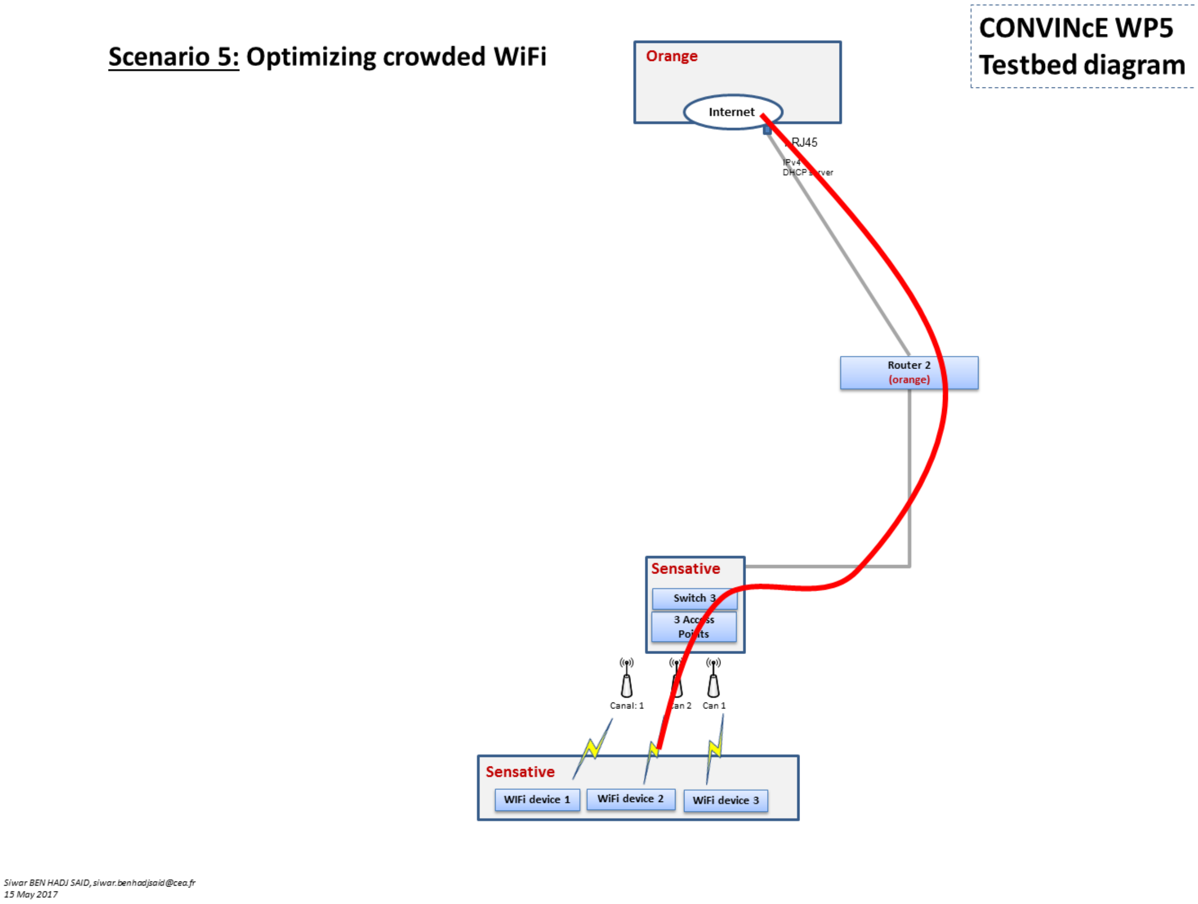


Figure 33: Test-bed Optimizing crowded WiFi

5.5.1.1 Description

Partner: **Sensative**

The overall goal with the WiFi CNOM system in Convince has been defined to lower the use of cellular connectivity for mobile devices by enabling an increased transfer from the cellular network to the far more energy efficient WiFi based data communication.

This should be obtained by increasing the availability of the WiFi-environment by utilizing optimized WiFi router configurations.

The obtained degree of optimization of the energy consumption has been evaluated through the increased availability and capacity of WiFi communication, rather than the actual energy consumption itself. The fundamentally interesting measurement would be the amount of data traffic that can be off-loaded from the cellular networks. This value would only be possible to achieve in a real-life emulating test set-up, which was above the possible scope for Sensative's short engagement in the Convince project. It is in Sensative's and Lund University's plan, to continue this work.

Sensative arrived late (2016) in the Convince project, replacing Terranet, and the time constraints only admitted the development of a rather basic CNOM algorithm and a limited test set-up.

The developed CNOM algorithm was per design held simple, in order to enable verification of the principles of the method, as well as the CNOM control system as such.)

5.5.1.2 Parameters Measured

The following WiFi parameters are currently measured and are used by the CNOM algorithm:

- Each channels Channel Load
- Measurement time frame length
- Network Busy time (per measurement time frame)
- Router transmit time (per measurement time frame)
- Router receive time (per measurement time frame)

The above parameters are used by the CNOM algorithm to decide the optimal WiFi configuration for each router, connected to the CNOM system.

5.5.1.3 Measured Results

The measured parameters are used by the CNOM algorithm to define a set of WiFi router configuration parameters, and the measured parameters themselves, are of now particular interest.

The fundamentally interesting measurement is instead the amount of data traffic that can be off-loaded from the cellular networks. The tests have demonstrated the general value of a WiFi CNOM control system and have verified the the technical solution with WiFi routers controlled by the CNOM control system, and the CNOM algorithm.

The real-life energy saving that is possible to obtain, would only be possible to demonstrate in a more complex test set-up, including local cellular base station, which was above the possible scope for our short engagement in the Convince project, see above.

5.6 End-to-End Energy and QoS/QoE

Partner: **VTT and BTH**

5.6.1.1 Description

The system collects the energy usage from devices by using the Open Energy Monitor (OEM) and other systems that report energy usage. Hence, it will act as an independent reference.

VTT:

In this demonstration, the original video content is encoded with different quality, resolution and frame rate settings and the power consumption is measured. The video is streamed to terminal from which the power is also measured. The baseline for measurements is coding with AVC with 24 fps at full HD resolution with 43 dB as an average PSNR value. Based on these measurements, the best trade-off and calculations on the overall effect and power saving situations are done. More detailed and systematic measurements on effect of the quality to power consumption are given in deliverables D2.1.2 and D4.1.2.

5.6.1.2 Parameters Measured

Power [W].

VTT:

- Power consumption and energy of the video processing in the VTT’s head end and mobile terminal including the CPU (four general purpose Core i7 processor cores and on-chip GPU) but excluding the external GPU.
- Amount of data transferred aka bitrate between the head end and the terminal.

5.6.1.3 Measured Results

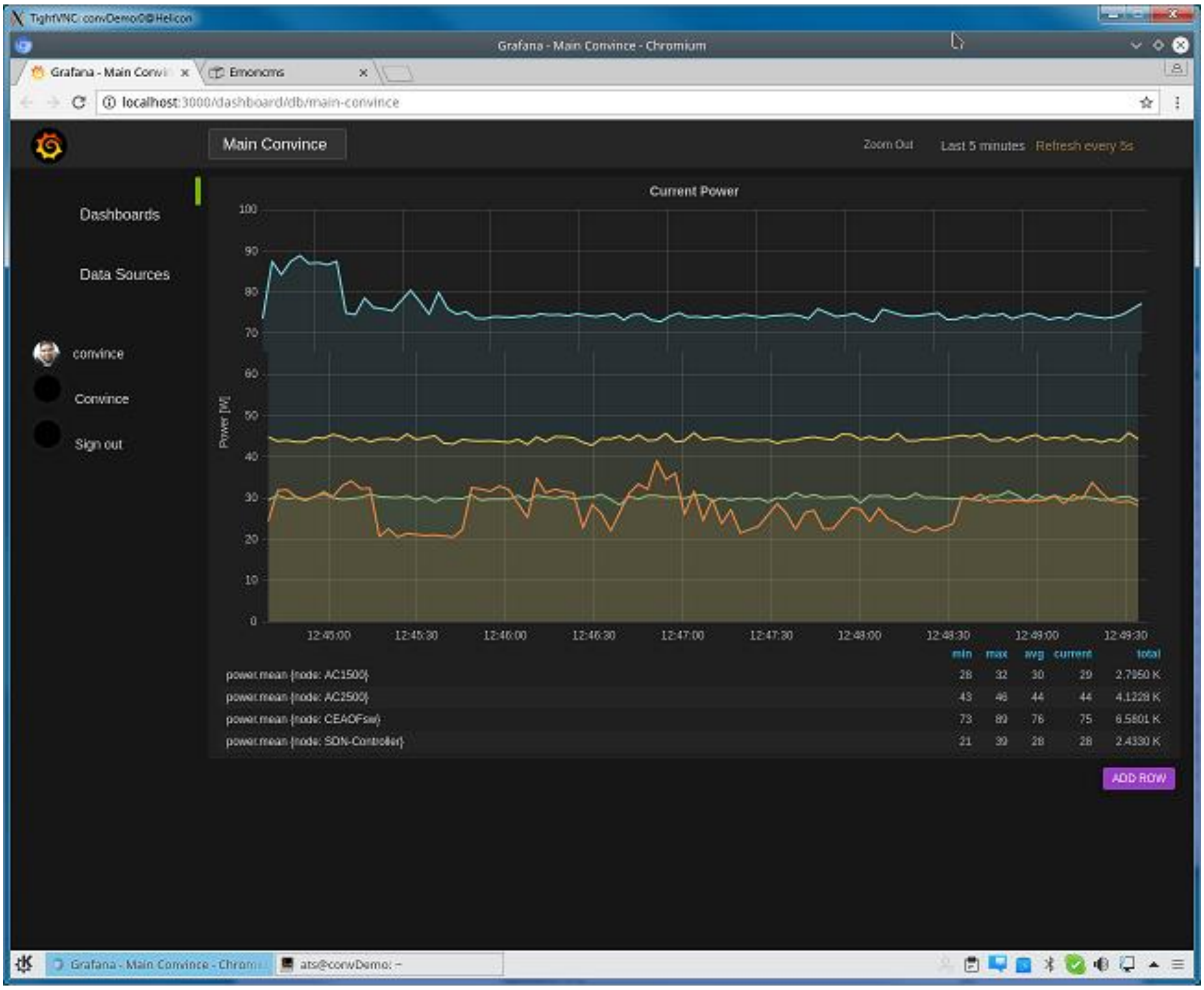
Real-time graphs showing the power usage by the different systems participating in an end-to-end video transmission.

VTT:

Error! Reference source not found. shows the power consumption curves for the head end encoding server when altering lower resolution, quality and frame rate for HEVC video. The results show that all of these selections decrease the required power in the head end. In particular:

AVC to HEVC => head-end power increases from 57,4 W to 140W (+144%), but bit rate decreases from 10 Mb/s to 4,4 Mbps

- Decrease quality 4 dB => HEVC power decreases from 140W to 102W (-28%) and bitrate from 4,4 Mb/s to 1,3 Mb/s
- Decrease frame rate to half (12 fps) from its original (24 fps) => HEVC power decreases from 140W to 88,4W (-37%) and bitrate from 4,4 Mb/s to 3,1 Mb/s
- Decrease resolution from full HD (1080p) to HD (720p) => HEVC power decreases from 140W to 84,8W (-39%) and bitrate from 4,4 Mb/s to 3,1 Mb/s



	min	max	avg	current	total
power.mean {node: AC1500}	28	32	30	29	2.7950 K
power.mean {node: AC2500}	43	46	44	44	4.1228 K
power.mean {node: CEAOFsw}	73	89	76	75	6.5801 K
power.mean {node: SDN-Controller}	21	39	28	28	2.4330 K

Figure 34: Sample of Real-Time power values

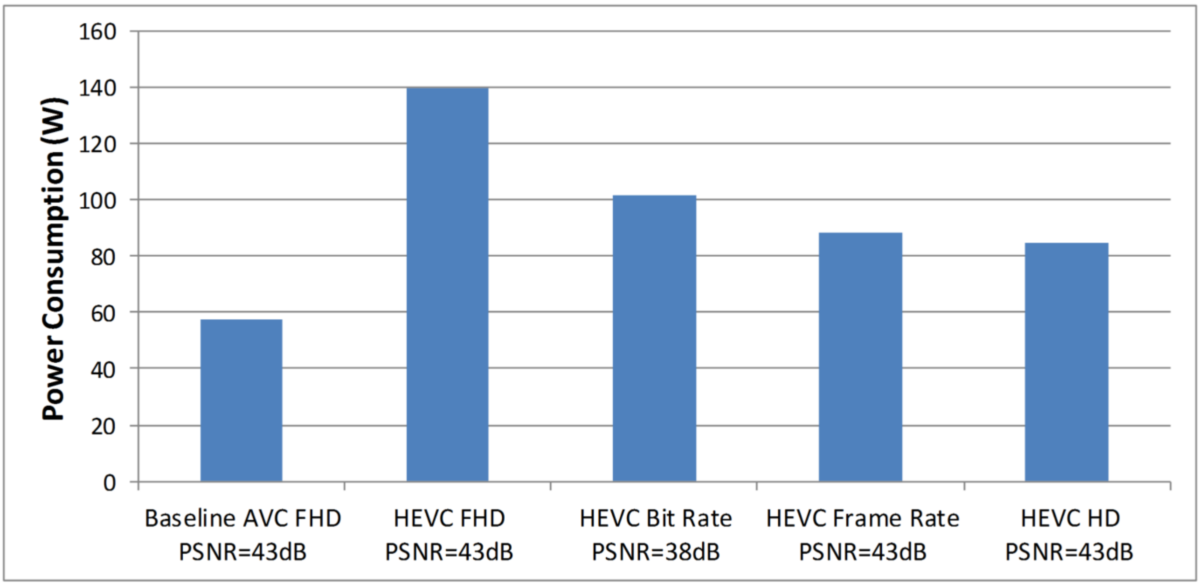


Figure 35: Power consumption figures for measuring end-to-end energy for VTT encoding server.

The figure below shows the results of video processing energy measurements in the terminal side with popular DivX Player for H.264/AVC (used as baseline in D4.1.2) and GPAC Osmo for H.265/HEVC. Energy is used as a metric since streaming times were significantly longer for AVC due to preloading. We can make the following observations from the results:

- The energy consumed by AVC streaming with the D4.1.2 baseline player DivX can be reduced by 81.5% if the configured/optimized player GPAC Osmo is used with HEVC streaming with the same quality settings.
- Switching from full HD 1080P to HD 720P drops the energy of playback by 25.6% in the terminal.
- Similarly, dropping the PNSR by 4 dB reduces the energy consumption by 25.4%.
- Halving the frame rate to 12 FPS drop the energy consumption by 32.3%.

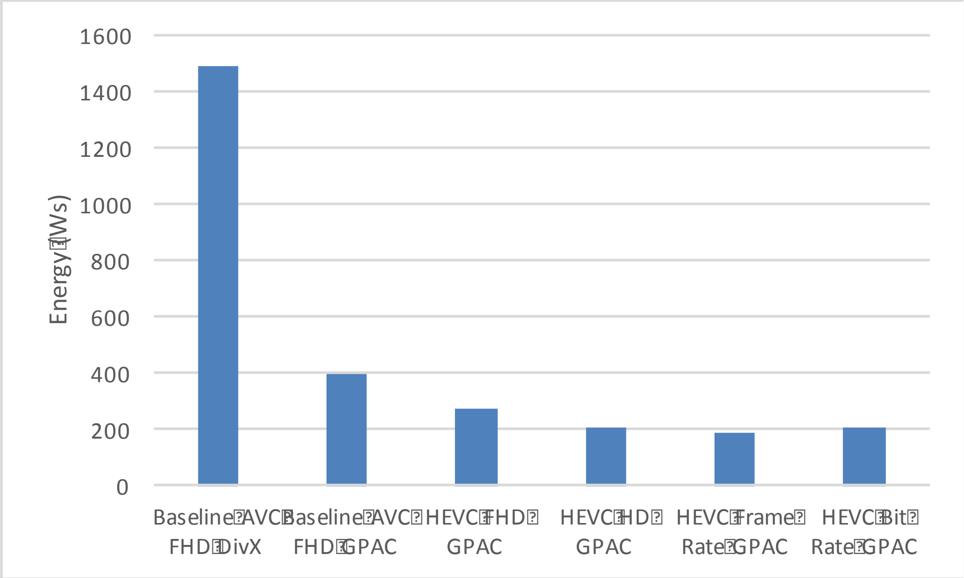


Figure 36: Energy consumption figures for measuring end-to-end energy for VTT mobile terminal

The figure below shows the amount of video data transferred from the head end to terminals in the tests. Our remarks are:

- The amount of video data for AVC is 56% higher than for HEVC with the same quality. This induces significant power savings in the network between the head end and terminal.
- Switching from full HD 1080P to HD 720P reduces the amount of transferred data by 31%.
- The same 31% savings can also be obtained from halving the frame rate.
- The saving obtained with dropping the PNSR by 4 dB is as high as 72%.

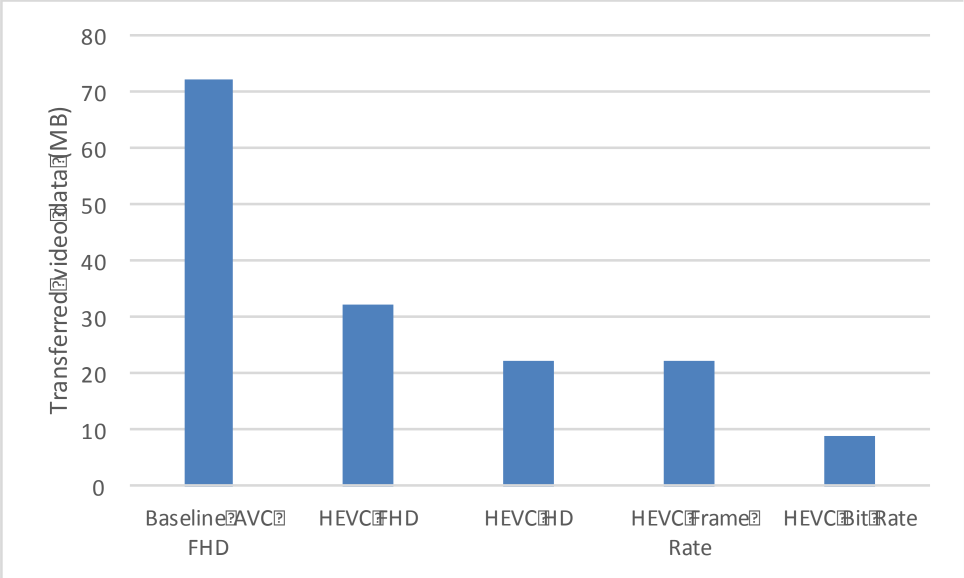


Figure 37: Amount of video data transferred from the head end to the terminal.

The CONVINcE solution is to use high efficiency H.265/HEVC coding instead of current H.264/AVC. This increases the energy consumption of the head end but decreases it in the network and terminals since the amount of data transferred from the head end to terminals decrease by 56% and decoding for H.265/HEVC is able to exploit power-savvy parallel execution capabilities of modern processors better than H.264/AVC. These end-to-end energy studies reveal that in the case of these settings, the additional energy consumption in the head end can be compensated in terminals if at least 4 runs per encoded video is played back in the high-performance mobile terminals used in these tests and if advanced GPAC Osmo terminal with optimized configuration/compilation and execution parameters is used instead of popular DivX baseline player (assuming SW encoding and decoding).

5.6.2 QoE-and-Energy-Optimal Video Streaming

5.6.2.1 Description

This section shows a set of QoE estimations, obtained in December 2016 by asking 40 users of the BTH Eduroam WiFi network about their opinion scores (OS) of Youtube videos with different resolutions (240p, 360p, 480p, and 720p). These QoE estimations are then combined with (an extrapolation of) the end-to-end energy measurement results presented in Section 5.4.1.3. This way, we can illustrate the trade-off between QoE, described by the mean opinion score (MOS) of all 40 users, and energy as a function of the video resolution R. This trade-off is captured by the parameter Quality of Experience per Joule, which expresses the gain on the MOS scale per extra Joule, and is again a function of R:

$$QoEJ(R) = (MOS(R) - 1) / E(R)$$

The higher the QoEJ, the better the quality-energy tradeoff.

The function $E(R)$ is extrapolated from the end-to-end energy consumption results presented in Figures 32 and 33, focusing on HEVC FHD (R = 1080p) and HD (R = 720p), respectively. We obtained two models:

1. Linear extrapolation: $E_{\text{lin}}(R) \approx 40\text{J} + R/\text{p} \times 0.22\text{J}$
2. Exponential extrapolation: $E_{\text{exp}}(R) \approx \exp(0.0009\, R/\text{p}) \times 102\text{J}$

While the linear extrapolation is a kind-of default choice, the exponential extrapolation inherently assumes a higher idle consumption (102J instead of 40J).

5.6.2.2 Parameters measured

$MOS(R)$ (for each resolution R; scale: 1 = bad; 2 = poor; 3 = fair; 4 = good; 5 = excellent)

Note: The confidence intervals (± 0.2…0.25 on the MOS scale) are omitted for sake of brevity.

5.6.2.3 Measured results

Table 7 illustrates the measured MOS (serving as QoE estimations) and QoEJ measures obtained through the linear and exponential models derived from the data in Section 5.4.1.3 and specified in Section 5.4.2.1. The optimal values ("sweet spots") are typeset in bold.

Resolution R	MOS(R)	$QoEJ_{\text{lin}}(R)$	$QoEJ_{\text{exp}}(R)$
240p	4.05	**0.033/J**	**0.024/J**
360p	**4.18**	0.026/J	0.022/J
480p	3.48	0.017/J	0.015/J
720p	3.00	0.010/J	0.010/J

Table 7: QoE estimations and QoEJ measures for different resolutions

It may be a bit surprising to see that the QoE “sweet spot” is obtained at a resolution of 360p. Indeed, for 480p and 720p, increasing amounts of jerkiness and freezes were disturbing the users. The latter disturbances are due to heavy traffic conditions on the BTH Eduroam WiFi network that is shared by many students – a rather typical case for a quasi-public wireless network. As soon as disturbances due to resource overload show up, the QoE affected quite significantly.

However, taking the rise of the end-to-end energy with growing resolution into account, the QoEJ “sweet spot” is now found at 240p, i.e. the higher resolutions (360p and onwards) lead to a suboptimal quality-energy tradeoff. These insights go hand-in-hand with the indications on the **end-to-end energy savings by reducing resource consumption** (through reduced resolution, SNR or frame rate) that were presented in Section 5.4.1.3. Moreover, the QoEJ decreases even faster as network-induced disturbances appear for 480p and 720p, which means that **conditions of resource over-utilization imply suboptimal quality-energy tradeoffs**.