# Evaluating the Quality of Service for a Satellite Based Content Delivery Network

Helmut Hlavacs, Guido Aschenbrenner, Ewald Hotop, Aadarsh Baijal, and Ashish Garg

***Abstract*—In the past years content delivery networks (CDNs) have become increasingly popular as they bypass the contended Internet and deliver content to their paying customers with higher quality of service (QoS). Satellite based CDNs additionally exploit the advantage of satellite multicast and broadcast, being able to deliver the same content to an arbitrary number of points of presence (PoPs) in its footprint. Although many future CDNs will use satellites, little is known about the QoS as perceived by end users of such a network and how to measure it.**

**In this paper we present a methodology for evaluating the QoS of a satellite based CDN for the European IST project CODIS. Designed as a professional CDN transporting MPEG-2 and MPEG-4 based multimedia presentations for content providers, broadcasters, and ISPs, the network will use the satellite STENTOR, containing an onboard MPEG-TS multiplexer. The methodology is centered around the QoS perceived by network end users.**

***Index Terms*—content delivery network, end user, measurement framework, quality of service, satellite.**

## I. Introduction

Content delivery networks (CDNs) have become popular in the last year for transporting data for institutions and companies relying on fast data transport with a guaranteed level of quality of service (QoS). A CDN essentially overlays existing networks which make up the Internet and which may use any combination of satellites and terrestrial networks. The basic idea for enhancing the observed QoS is to bring content to the network edges close to the end users, thus reducing access latencies and error probabilities [18]. As satellites intrinsically provide point-to-multipoint (multicast) communication within their footprints, content may be easily distributed to many points of presence (PoPs) of content providers with no additional overhead, a point especially interesting for distributing multimedia content like streaming media presentations and static web pages. Many future CDNs are thus likely to use satellites as means for transporting their content [11].

Customers paying for CDN services expect to experience a QoS being significantly better than the one delivered by the Internet, the QoS either denoting the ability of the CDN to move around content or, more importantly, as experienced by network end users accessing the CDN content.

It is therefore important to have means for measuring this experienced QoS, a difficult task when considering the variety of the used network technologies (including network and application protocols) and the transported content. For example, for a streaming application, other QoS parameters are important than compared to a web browser.

In this paper a framework for measuring the QoS of a satellite based content delivery network called CODIS (Content Delivery Improved by Satellite) is described. The framework is the result of an extensive literature survey and provides a basic methodology for evaluating the *end-user centered* QoS of such a CDN. It consists of the general network setup, the hardware and software to be used for measurement, the parameters to be measured together with an interpretation of how end users would subjectively judge them, and an experimental design.

## II. The CODIS Network

CODIS[1] is an IST project supported by the European commission. The CODIS consortium, consisting of Alcatel Space, the French space agency CNES, the broadcasting research institutions Telediffusion de France (TDF) and Institut für Rundfunktechnik (IRT), the broadcasting test equipment manufacturer Rohde & Schwarz, the content management system provider Activia, and the Institute for Computer Science and Business Informatics of the University of Vienna, will setup, run and test a satellite based CDN using the satellite STENTOR constructed by Alcatel Space. STENTOR will be launched approximately in September 2002 by CNES and be positioned at GEO 11° West. It contains an on-board MPEG-2-TS multiplexer, enabling the reception of

This work was supported by the Commission of the European Union through the Information Society Technologies (IST) project CODIS, grant number IST-2000-30066.

H. Hlavacs is assistant professor at the Institute for Computer Science and Business Informatics at the University of Vienna. (e-mail: helmut.hlavacs@univie.ac.at)

G. Aschenbrenner is research assistant at the Institute for Computer Science and Business Informatics at the University of Vienna. (e-mail: guido@ani.univie.ac.at)

E. Hotop is research assistant at the Institute for Computer Science and Business Informatics at the University of Vienna. (e-mail: hotop@ani.univie.ac.at).

Aadarsh Baijal is a third year student of Mathematics and Computing at the Indian Institute of Technology, Delhi.(e-mail: aadarsh-baijal@lycos.com )

Ashish Garg is a third year student of Mathematics and Computing at the Indian Institute of Technology, Delhi.(e-mail: ashish_iit@yahoo.com )

data sent by multiple sources, which is multiplexed and broadcasted back. Uplinks are limited by approx. 9 Mbit/s, the downlink bandwidth is limited by 38 Mbit/s.

The goal of the CODIS trials is to demonstrate the usefulness and QoS of such a CDN, which may be used, for example by public broadcasters to bring their pre-recorded content near the end users, or which may directly send their content live from anywhere using STENTOR as a CDN entry point.

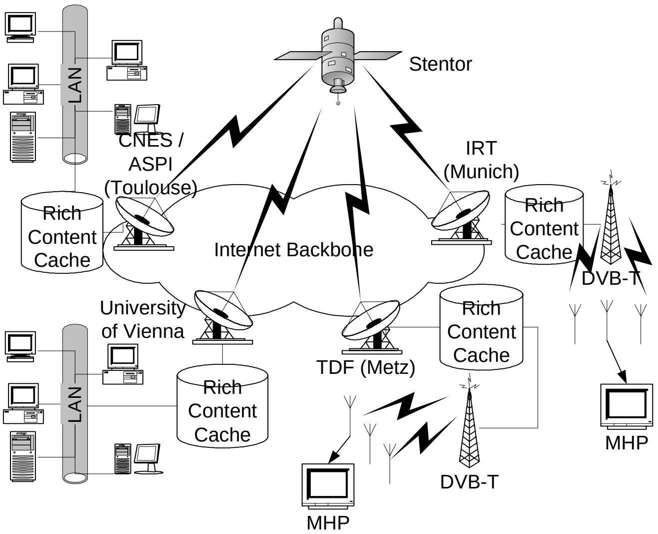


Fig. 1. CODIS CDN network.

Fig. 1 shows the structure of the CODIS CDN. Satellite stations will be placed at Toulouse, Metz, Munich and Vienna, each being able to send and receive data from STENTOR, with the exception of Vienna, which will run in receive mode only. At each site, a remotely operated cache stores content at the network edges. Content may be transported in push mode, meaning that it is transported before being requested by an end user, or in pull mode, meaning that an explicit end user request initiates the transfer from a remote cache to the cache near the end user. Due to the large latencies experienced when using GEO satellites, rather than using TCP (large latencies may impede the TCP slowstart mechanism), the management system uses reliable multicast based on UDP and retransmission of lost packets.

At the receiver, content may be stored in an edge cache or directly streamed to the end user. As CODIS focuses on public broadcasting, in our concept, the last mile to the end user is given by a terrestrial digital video broadcasting (DVB-T) infrastructure, the back channel being based on local ISPs using systems like cable modems, ISDN or 56k modems.

As prototypical content protocols we focus on the MPEG-2 based DVB standard [29] and the emerging multimedia standard MPEG-4 [26].[2] Whereas the DVB content will be transported over an MPEG-2 transport stream, the MPEG-4 content may be transported over MPEG-2-TS private sections, multiprotocol encapsulation, or streamed over IP. As the Activia management system is mainly based on IP, we focus on MPEG-4 over IP, using standard software like the RealPlayer[3] enhanced with the Envivio[4] MPEG-4 plugin, the Darwin[5] streaming server or the MPEG4IP[6] tools. Furthermore, the DVB content will be presented on standard multimedia home platforms (MHPs),[7] enhanced by the capability to play MPEG-4. One such terminal is given by SAMBITS, a prototypical MHP implementation by IRT.[8]

One of the main advantages of such a hybrid solution (DVB and MPEG-4) is to be able to mix broadcasted with on-demand content. Consider for example the broadcast of ordinary television news via DVB, which is enhanced by a sign language translator for handicapped people. Whereas the sign language presentation would consume too much of the broadcast link budget to send it too all viewers, it can be stored on edge devices near the DVB-T stations and transported only on-demand to the DVB-T sender near particular end users.

The above described example scenario demands that the presentation of both contents be highly synchronized. Also, end users requesting the signal expect not to wait too long for the start of the MPEG-4 presentation, as they do not want to lose too much of the news information. Finally, the sign language video quality must be high enough to easily identify the shown information, and dependable enough to guarantee a presentation not being interrupted by stalls. The latter is often experienced when streaming high quality multimedia content over the Internet.

The above example shows that seen from the end user, such a CDN exhibits numerous QoS parameters which, when measured, should lie within some bounds in order to guarantee end user satisfaction. In this paper a general QoS framework is described which is used for measuring QoS of the CODIS CDN. The framework consists of several classes of parameters to be measured, some representing technical metrics, others focusing directly on the end user perceived QoS. The framework contains the parameters to be measured, a list of hardware and software tools for obtaining them, and bounds for some parameters describing how to interpret the end user experienced QoS, i.e, how end users would subjectively rate the QoS of the network.

## III. The Measurement Framework

As a starting point we use the CCITT Recommendation X.140 [12], comprising a general framework for user-oriented QoS measures in data networks. The basic parameters are shown in TABLE I. Here, a *block* is a *unit of user information* that is transferred over the network [30]. This can be a web page, a video frame or a simple file.

[1] http//www.codis-satellite.com
[2] http://mpeg.telecomitalialab.com
[3] http://www.realnetworks.com
[4] http://www.envivio.com
[5] http://developer.apple.com/darwin/projects/streaming
[6] http://mpeg4ip.sourceforge.net
[7] http://www.mhp.org
[8] http://www.irt.de/sambits

TABLE I
GENERAL QOS PARAMETERS FOR COMMUNICATION VIA PUBLIC DATA NETWORKS (CCITT REC. X.140).

| Criterion Function | Speed | Accuracy | Dependability |
|---|---|---|---|
| **Access** | Access Delay | Incorrect Access Prob. | Access Denial Prob. |
| **User Information Transfer** | Block Transfer Delay<br>Block Transfer Rate | Block Error Prob.<br>Extra User Inf. Del. Prob.<br>Block Misdelivery Prob. | Block Loss Prob. |
| **Disengage-ment** | Disengagement Time | Disengagement Failure Prob. | |

There are two types of parameters. The *primary* parameters describe the QoS during normal hours of service operation, the *secondary* parameters describe the frequency and duration of service outages.

The following list contains an explanation of the parameter categories, which will be used for CODIS:

- *Access Delay.* This is the time elapsed between an access request and successful access. This parameter is generalized to *Response Time* as the time between manually issuing a request to the system, until the request is satisfied.
- *Block Transfer Delay.* This is the latency of a block sent over the network.
- *Block Transfer Rate.* This is the throughput experienced when transferring a block.
- *Block Error Probability.* This is the probability for bit errors or bit losses occurring in a transferred block.
- *Block Loss Probability.* This is the probability that a block is lost during transfer.
- *Disengagement Time.* This is the elapsed time between the attempt to close a connection until the connection is actually closed.

These categories define the basic parameter space that will be used for CODIS. If direct interaction with the system and the user is necessary in order to start or change data delivery, the *Access* parameters will be used. When evaluating the transport of user information data, the *User Information* categories will be used, where either single blocks, multiple asynchronous or streams of blocks may be delivered. For measuring the stopping of presentations, the *Disengagement* categories will be used.

In the CODIS CDN there will be two categories of users. The first one is given by the institutions in between the content providers and the end users, i.e., network providers, ISPs, broadcasters, content management providers etc. They will be interested into the capability of CODIS for transporting content from its point of origin to the network edges. The primary QoS metrics important here are described in Sections IV and V. The second user class is given by the network end users accessing and watching multimedia content live or on demand. Metrics describing the QoS observed by them are described in Sections V, VI, and VII.

## IV. IP NETWORK ANALYSIS

One important aspect of CODIS is to transport content via IP from one cache to another (see Fig. 1). Thus, the first aspect of our methodology deals with measuring the QoS of the CODIS IP network and how to compare this QoS to the one of the standard Internet, here with respect to the Block Transfer Delay, Rate, and Loss entries of TABLE I. Functioning as a starting point for our methodology, the IETF IP Performance Metrics (IPPM)[9] group has issued several documents and drafts defining performance metrics, all being based on one initial framework [24]. From there we use the following network parameters: A *path* is a sequence of $N+1$ hosts $h_0,\ldots,h_N$, being interconnected by $N$ links $l_1,\ldots,l_N$. The value $N$ is called *hop count*. Now assume that a packet $p$ of size $s$ is sent from host $h_0$ over a link with bandwidth $b_1$ to the subsequent host $h_1$. The packet is sent at time $t_0$ and received at time $t_1$. Furthermore assume that the packet is put into a queue at the sender and must wait for time $q_1$ until the line is free. Also, when $d_1$ denotes the physical length of link $l_1$ and $c$ denotes the speed of electrical signals in the link, $d_1/c$ is called *propagation time*. Finally, the time for handling packets at the receiver is denoted by $f_1$ and the *link latency* is called $d_1/c+f_1$. Then the *sending time* of $p$ is given by

$$T_1(s)=t_1-t_0=\frac{s}{b_1}+\frac{d_1}{c}+f_1+q_1. \tag{1}$$

Following from (1), when traversing over $N$ hops the total sending time for packet $p$ is given by [9], [17]

$$T_N(s)=t_N-t_0=\sum_{i=1}^{N}\frac{s}{b_i}+\frac{d_i}{c}+f_i+q_i\ . \tag{2}$$

The only nondeterministic parameters in (2) are the queuing delays $q_i$ which depend on the router load, itself often changing drastically over time. For some metrics this run-time aspect is irrelevant and is thus omitted. The end-to-end *network latency* is the sending time of an empty packet over a lightly-loaded network

$$L_N=t_N-t_0=\sum_{i=1}^{N}\frac{d_i}{c}+f_i\ , \tag{3}$$

which is the minimum time a packet will need when sent from $h_0$ to $h_N$. Additionally, the roundtrip time (*RTT*) is the network latency from $h_0$ to $h_N$ plus the network latency from $h_N$ to $h_0$. Note that due to the possibility of different routes in both directions, the *RTT* may differ from $2\times L_N$ significantly.

As CODIS is meant for moving large amounts of multimedia data, the ability for doing so is measured by the bulk transfer capacity (*BTC*) [20]. Assume that $s$ bytes are to

[9] http://www.ietf.org/html.charters/ippm-charter.html

be sent over $N$ hops, then

$$BTC = s / T_N(s)\,. \tag{4}$$

The *BTC* is severely influenced by the *bottleneck bandwidth* $\hat{b} = \min\{b_1,\ldots,b_N\}$ in the following sense. Assume $q_i = 0$. As there is a maximum size for packets for each path in the Internet called maximum transfer unit (*MTU*), and keeping in mind the necessary TCP/IP overhead, for transferring $s$ bytes, the number of sent packets is $k > s / MTU$. When denoting the sending time for $k$ packets of size $MTU$ over $N$ hops by $T_N^k(MTU)$, then the following inequalities can be derived:

$$L_N + \frac{k\,MTU}{\hat{b}} \le T_N^k(MTU) \le L_N + (k + N + 1)\frac{MTU}{\hat{b}}\,, \tag{5}$$

which immediately give bounds for the *BTC*. The lower bound is derived by setting $b_i = \infty$ (with the exception of $\hat{b}$), the upper by setting all $b_i = \hat{b}$.

Another parameters influencing the BTC is the *packet loss rate* $r_N$, denoting the percentage of packets lost when sent from $h_0$ to $h_N$. As each lost packet must be retransmitted, each packet loss increases the sending time and thus decreases the *BTC*.

As a result, in our methodology, the parameters to be measured for each pair of sites are given in TABLE II.

TABLE II
CODIS IP NETWORK PARAMETERS.

| Parameter | Symbol | Priority |
|---|---|---|
| Hop count | $N$ | Mandatory |
| Network latency | $L_N$ | Mandatory |
| Bottleneck bandwidth | $\hat{b}$ | Mandatory |
| Packet loss rate | $r_N$ | Mandatory |
| Block transfer capacity | *BTC* | Mandatory |
| Link latencies | $d_i / c + f_i$ | Optional |
| Link bandwidths | $b_i$ | Optional |

One tool being able to measure these parameters is pchar.[10] Pchar is a tool used to characterize the bandwidth, latency, and loss of links along an end-to-end path through the Internet. It is based on the algorithms of the pathchar[11] utility. It works by sending out a series of probes with varying values of the time to live (TTL) field and varying packet sizes. The TTL determines how many links a packet traverses before it expires If a router receives a packet that has expired, it drops the packet and sends an ICMP error packet back to the sender. For each probe, it measures the time until the error packet is received. By performing statistical analysis of these measurements, it infers the latency and bandwidth of *each link* $l_i$ in the path and the probability that a packet is lost.

Assume that the latency and bandwidth for links $l_1,\ldots,l_k$ are known. Then pchar measures the round trip times for varying packet sizes for the path $h_0,\ldots,h_{k+1}$, taking the minimum round trip time for each size (to account for zero queuing delay) yields a scatter which is very close to a straight line. Therefore, fitting a simple linear regression results in an estimate of latency and bandwidth for this new path, shown in Fig. 1. Then, the link latency and the bandwidth (inverse) of link $l_{k+1}$ are estimated by subtracting the sum of the old values from the measured values of the new path [9].

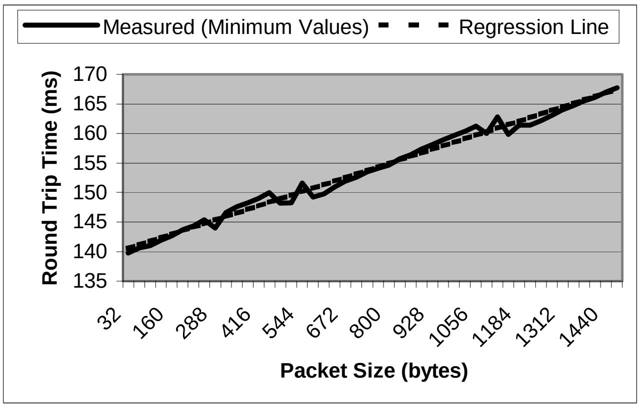


Fig. 2. Response time measurements and estimates for a specific path.

The drawbacks of pchar include the facts that pchar relies on the complete knowledge of all intermediary links and that the path in both directions is the same. Unfortunately, some Internet routers are configured not to respond with ICMP packets to zero-TTL-packets, leaving black spots in the path. Additionally, routes may change frequently in the Internet and often different paths are observed for packets from one host to another and back. Other methods for estimating the bottleneck bandwidth can be found, for instance, in [17], [23].

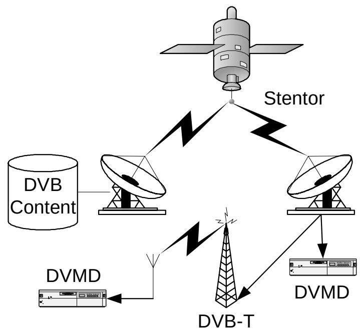


Fig. 3. MPEG-2 over MPEG-2 Transport Stream Network Scenario

## V. MPEG2-TS Analysis

We will check at least a subset of parameters recommended in [10], [29]. As shown in Fig. 3 MPEG-2 transport streams will be measured at two points. Right after the satellite receiver the Rhode & Schwarz DVMD will measure the validity of the satellite downlink. Additionally, at the end user, the DVMD will check the transport stream received at the CODIS end user.

[10] http://www.employees.org/~bmah/Software/pchar/
[11] http://www.caida.org/tools/utilities/others/pathchar/

### A. TS related parameters: Hierarchy

The tests are grouped in three priorities according to the importance for monitoring purposes. The *first* group comprises a basic set of parameters that are necessary to ensure the Transport Stream (TS) *can be decoded*. Once first priority errors are detected it makes no sense to scan for second or third priority errors.

The *second* group comprises additional parameters which are *recommended* for continuous monitoring. Many of the tests in this group are only meaningful, if the content is not scrambled.

The *third* group lists *optional* additional parameters which could be of interest for certain applications. Most of the tests in this group refer to further tables with service information. Tables checked are the Network Information Table (NIT), Service Description Table (SDT), Event Information Table (EIT), Running Status Table (RST), and the Time and Date Table (TDT).

### B. Evaluation by combination of parameters

[10] proposes combinations of parameters described above, which can approximate the probability for a certain percentage of time and location that a service is available in a certain area with a defined quality. The *Service_Availability_Error*, *Service_Degradation_Error* and *Service_Impairments_Error* all work the same – they are the maximum of the number of occurrences of selected errors within a given time interval ΔT. Table III lists the types of errors used.

TABLE III
METRICS FOR ESTIMATING AVAILABILITY OF SERVICE.

| Metric | Error Types |
|---|---|
| Service_Availability_Error | TS_sync_loss, PAT_error, PMT_error |
| Service_Degradation_Error | CRC_error, PCR_error, NIT_error, SDT_error |
| Service_Impairments_Error | Continuity_count_error, Transport_error |

When running the test over a period of time (e.g. 5 hours), the percentage of time for which the parameter exceeds a predefined threshold can be computed for each metric – these are the so-called ratios.

The *Service_Availability_Error* (and *_Ratio*) can identify severe distortions and interruptions of the service, while the *Service_Degradation_Error* (and *_Ratio*) indicates severe degradations and strong impairments of the service. Finally, the *Service_Impairments_Error* (and *_Ratio*) reacts on first signs of service degradation and infrequent impairments of the service.

[10] proposes the following criteria for quality of reception: Very good reception quality (no visible or audible impairments for several minutes) is reached if:

- *Service_Availability_Error* = 1 for 100% of the time
- *Service_Degradation_Error* = 1 for 100% of the time
- *Service_Impairments_Error* <= 2 for 95% of the time

Very bad reception conditions are defined as:

- *Service_Availability_Error* >= 2 for 75% of the time
- *Service_Degradation_Error* >= 2 for 95% of the time
- *Service_Impairments_Error* >= 3 for 95% of the time

For CODIS, this information will be recorded and used as QoS measurement results.

## VI. RESPONSE TIME

In this methodology we use the term access to denote all direct interactions between users and the system (see Section III). The quality of service parameter is the user perceived *response time* $T_R$.

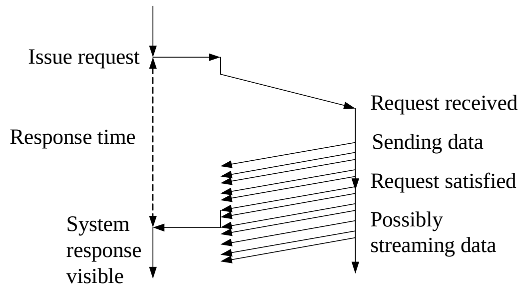


Fig. 4. Definition of response time.

Fig. 4 shows the definition of response time, being the time between issuing a request to the system until the result is visible (or audible) to the user. Response time is influenced by the time it takes to transfer the request to the remote server, the time the remote server needs for satisfying the request, and the time it takes to transfer and present the request to the end user.

From intuition it is clear that longer response times decrease user satisfaction. It is, however, generally not easy to quantify the user satisfaction as a function of response time. In order to be able to interpret the response times as being acceptable or not, several limits have been proposed in the literature, which will be discussed briefly.

Response time limits derived in the scientific literature, denoting what users subjectively would rate “good”, include **4 seconds** [25], [28], **5 seconds** [4], **8 seconds** [40], **10 seconds** [21], and **11 seconds** [7]. Zona Research later extended its 8 second rule to a mapping of latency to expected exit rates, where users would abort online transactions (Table IV). Zona also states that 20% of users exiting are lost and will not revisit the Web site again.

More advanced research states that the user perceived QoS is not only a function of the response time, but also depends on the user’s expectations [5] and the online time already spent [3]. In [6], Web response times have been rated for different scenarios using a scale *low*, *medium*, and *high*. In scenario 1, no progress of current downloads was visible. In scenario 3, downloads were *incremental*, and downloaded Web page components were immediately visible (Table V).

TABLE IV
EXIT RATES DEPENDING ON RESPONSE TIME.

| | |
|---|---|
| <7 seconds | 7% |
| 8 seconds | 30% |
| >12 seconds | 70% |

TABLE V
USER SATISFACTION DEPENDING ON RESPONSE TIME.

| Rating | Scenario 1 | Scenario 3 |
|---|---|---|
| High | 0-5 seconds | 0-39 seconds |
| Average | >5 seconds | >39 seconds |
| Low | >11 seconds | >56 seconds |

A more general subjective rating by 30 individuals of latencies is shown in Fig. 5. It can be seen that the low-rating coincides with several results from other studies.

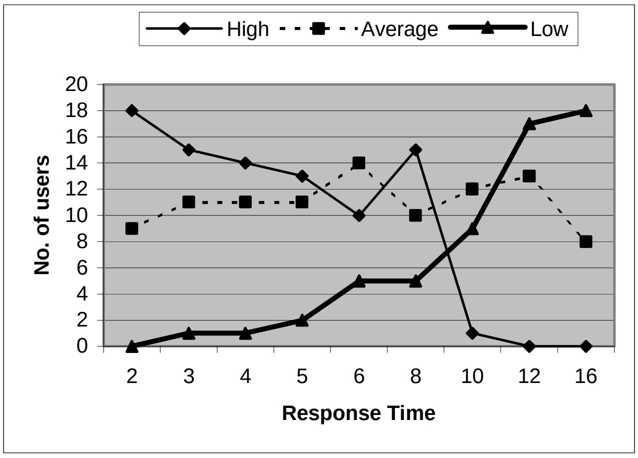


Fig. 5. Subjective rating of response time.

Two other metrics for direct interaction will be considered for CODIS: *access denial probability* $p_a$ and *disengagement time* $T_D$. For access denial probability, we have no mapping to user satisfaction. However, this may be mapped to business loss, when considering that each denied customer stops the transaction with probability $p_r$ or tries again with probability $1-p_r$. Disengagement time will be interpreted like response time.

TABLE VI
CODIS USER TERMINAL ACCESS METRICS

| Parameter | Symbol | Priority |
|---|---|---|
| Response Time | $T_R$ | Mandatory |
| Access denial probability | $p_a$ | Mandatory |
| Disengagement time | $T_D$ | Optional |

Fig. 6. shows the instrumentation of the SAMBITS terminal and the used MPEG-4 player for measuring the system response time as experienced by end users. The time points at which direct user interactions are issued as well as the timepoints at which the results are visible will be recorded. Their difference then yields the desired response time. Response time will then be interpreted according to Table IV, Table V, and Fig. 5, resulting (i) in the probability that the response time is high, medium, or low (using Table V), (ii) the percentage of lost users (using Table IV), (iii) and the percentage of users rating the response time high, medium, or low (using Fig. 5).

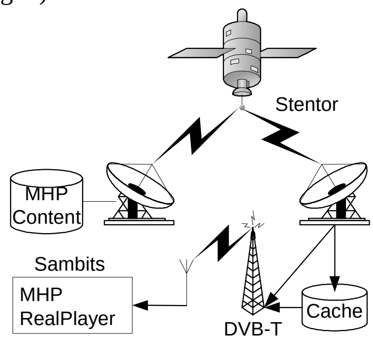


Fig. 6. MPEG-2 / 4 over SAMBITS Scenario.

## VII. VIDEO PRESENTATION QUALITY

In CODIS videos will be based on MPEG-2 (pure DVB) and MPEG-4 (IP based). Two sources for reducing the presentation quality of digital videos exist: (i) picture artifacts (impairments) and reduced frame rates stemming from reducing the video data rate in the encoding process, and (ii) picture artifacts and frame losses due to lost or delayed packets.

The ability for a CDN to transport multiple video streams is directly influenced by the video bitrate, thus it is imperative to relate the CODIS transport capacity (number of videos that can be transported concurrently) to the expected end user perceived quality.

Usually, the quality of digital video transmission are measured in terms like block distortion, blurring, edge business, mosquito noise, etc [1], [15]. As we are interested in the viewer's opinion of video quality, the most obvious technique to rate the quality of digital video is conducting subjective tests. This means that the sequences of interest are viewed by test- persons, who are asked for their quality-rating of the sequences [14]. Conducting such subjective tests is a very complex and time-consuming task [31], thus we will rely on measuring impairments by using objective techniques that can be mapped to subjective ratings.

### *A. Metrics Based on Pixel Differences*

The simplest kind of objective measures is based on the pixel value difference of the original picture sequence $I$ and the degraded picture sequence $\tilde{I}$

$$e(x,y,t) = I(x,y,t) - \tilde{I}(x,y,t),$$

where $x$ and $y$ are the pixel coordinates in a frame and $t$ is the frame index. Here it is assumed that pictures are of size $X \times Y$ and there are $T$ frames in both sequences.

The metrics then compute an overall sum indicating the overall picture quality. Popular measures are the Mean Squared Error (*MSE*), the Root Mean Squared Error (*RMSE*) and the Peak Signal to Noise Ratio (*PSNR*).

The *MSE* is the mean of the squared differences between the grey-level values of pixels in two pictures or sequences $I$ and $\tilde{I}$:

$$MSE = \frac{1}{TXY} \sum_t \sum_x \sum_y [e(x, y, t)]^2 . \tag{6}$$

The *RMSE* thus defines the average difference per pixel:

$$RMSE = \sqrt{MSE} \tag{7}$$

The *PSNR* is defined in decibels as follows:

$$PSNR = 20 \log_{10} \frac{m^2}{MSE} , \tag{8}$$

where $m$ is the maximum value that a pixel can take [21]. Yet another metric based on the *MSE* is called *DIST* [16]. However, [16] report that the *PSNR* correlates better to subjective ratings than *DIST*. Yet another metric is given in [35]. Subjective ratings of a set of video presentations are modeled by a linear estimator based on objective measurements $m_i$ from the presented video, the $m_i$ being functions of the spatial information (*SI*) of the original and degraded frames, and the temporal information (*TI*) of the picture sequences. For calculating the *SI*, the pictures are first filtered by a vertical Sobel filter

$$\begin{aligned} SI_v(x, y, t) \quad = \quad & I(x+1, y-1, t) - I(x-1, y-1, t) + \\ & 2I(x+1, y, t) - 2I(x-1.y.t) + \\ & I(x+1, y+1, t) - I(x-1, y+1, t) \end{aligned} \tag{9}$$

and a horizontal Sobel filter [2]

$$\begin{aligned} SI_h(x, y, t) \quad = \quad & I(x-1, y+1, t) - I(x-1, y-1, t) + \\ & 2I(x, y+1, t) - 2I(x, y-1, t) + \\ & I(x+1, y+1, t) - I(x+1, y-1, t) \end{aligned} . \tag{10}$$

(9) and (10) are then combined into one single frame

$$SI_r(x, y, t) = \sqrt{SI_v^2 + SI_h^2} \tag{11}$$

containing only the contours of the former content. The *TI* is yielded simply by taking the pixel-by-pixel differences of successive frames: $TI(x, y, t) = I(x, y, t) - I(x, y, t-1)$.

In [37] the authors have conducted extensive standardized video tests and have compared subjective ratings to several objective metrics from [2], mostly also based on *SI* and *TI*. The objective metrics explaining most of the subjective data were

- Negative Sobel (Negsob) $M_{Negsob}$. Here all positive values of $SI_r$ are set to zero, the Negsob is then the mean of the negative and zero values.
- Positive Sobel (Possob) $M_{Possob}$. Like Negsob, only that negative values of $SI_r$ are set to zero.
- Measure 714 ($M_{714}$) or the family of measures of lost motion [2], and
- Measure 711 ($M_{711}$) or measures of added motion [2].

The paper also contains the regression coefficients for various linear models based on the selected objective measures, for example

$$\hat{s}_1 = 0.224 \cdot M_{Negsob} - 8.662 \cdot M_{711} - 7.547 \cdot M_{714} . \tag{12}$$

Although computing these metrics is very easy and fast, they have only limited applicability, because they do not measure the quality perceived by human observers, but only consider pixel-per-pixel differences. Not considering the type of degradation though can result in misleading ratings. Furthermore, these metrics require that all pictures of the degraded sequence must be mapped to their originals (yielding *double ended metrics*), which may be difficult to be achieved.

### *B. Metrics Using a System Model*

These models use a priori knowledge about the system under test, the compression algorithm in use, as well as the accompanying artifacts [13]. That means for example they measure block distortion [1] that occurs in MPEG compressed video. A measure for blockiness usually is deducted from edge detection filters. A basic version of edge detection can be found in [2]. Also the DVQ-W metric [38] developed by Rohde & Schwarz in cooperation with the IFN (Institut für Nachrichtentechnik) of the TU Braunschweig makes use of block detection. One possible way is to obtain a measure of blockiness is to sum the squared differences of horizontal neighbor pixels for each column $i$ :

$$S(i) = \sum_{j=1}^{Y} \left( I(i+1, j) - I(i, j) \right)^2 , \; i = 1, \ldots, X-1 , \tag{13}$$

yielding characteristic spikes at the block borders. An example for such spikes (at multiples of 8) can be seen in Fig. 7, being smaller for low compression rates and larger for high compression. An overall measure then can be calculated by relating the mean of the $S(i)$ taken at block borders to those inside the blocks .

An important advantage of this approach is given by the fact that only the received pictures are needed (*single ended* metrics), no mapping to their originals is required.

### *C. Metrics Incorporating the Human Visual System*

Even more sophisticated metrics take into account the Human Visual System (HVS). That means that distortions are weighted depending on the detectability by a human observer, depending thus on a model of the HVS [19], [21], [27], [33], [34], [36], [39]. However, all HVS based metrics found in the literature cannot be reproduced as important data (e.g. weights) have not been specified. Furthermore, statistical analysis has shown that all HVS based metrics proposed for a particular test have not correlated better with subjective ratings than the much simpler *PSNR* [32].

### *D. Measurement Scenarios*

#### *1) Digital Video Broadcast*

The scenario for measuring the quality of the DVB based presentations can be seen in Fig. 8. The picture quality analysis is carried out by the Rhode & Schwarz digital video quality analyzer DVQ, itself using an algorithm based on (13). The DVQ analyses the blockiness of the MPEG-2 presentation and results in quality estimates called DVQ-W between 0 and 100. The interpretation of this value can be seen in Table VII.

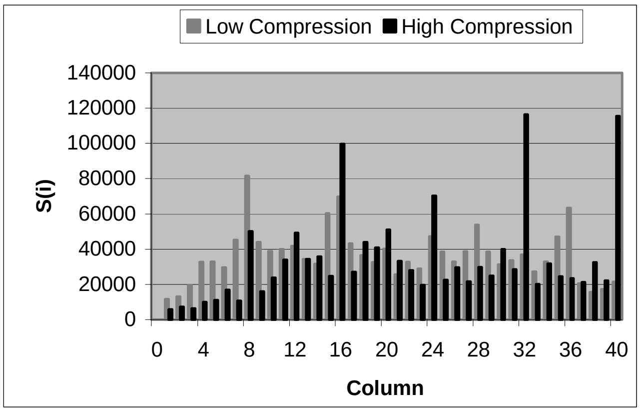


Fig. 7. (13) for two images with different blocking artifacts.

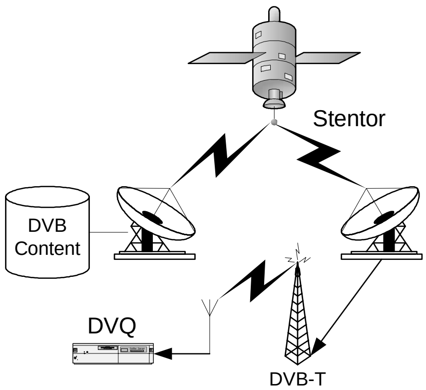


Fig. 8. Measuring DVB presentation quality.

TABLE VII
DVQ-W INTERPRETATION.

| Interval | Bounds | Interpretation |
|---|---|---|
| 1 | 0—20 | Bad |
| 2 | 20—40 | Poor |
| 3 | 40—60 | Medium |
| 4 | 60—80 | Good |
| 5 | 80—100 | Excellent |

For different contents and encoded in different bitrates, the following will then be computed:

- Probability for content being in one of the five intervals.
- Mean, standard deviation, ECDF and histogram of DVQ-W values.
- Autocorrelation function for DVQ-W values.
- Mean time that is spent in an interval before leaving it.
- Probabilities for transition from interval $i$ to interval $j$.

### 2) *Streaming Applications*

The setup for measuring the presentation quality of streaming applications is shown in Fig. 9. Media files are streamed over an IP network and presented in a player. The presentation can be recorded frame by frame using frame grabber software like Camtasia[12] or Hypercam[13] and stored in a Windows AVI file. Problems that might arise here are due to the additional CPU power and memory bandwidth necessary for grabbing and storing the pictures. Preliminary tests have shown that capturing with 25 fps is possible only for small frames, thus it might be necessary to use double processor machines or capture subframes only. Other possibilities include to change the source code of the open source MPEG-4 player MPEG4IP to save all frames to disk. Additionally, the players will be modified in a way that they log away the current stream bandwidth, framerate, packet loss rates, and the timepoints of user interactions. From there we may also derive response times according to Section 5. For single ended picture quality metrics like DVQ (see Section VII.B) we only need the recorded AVI, for double ended metrics (see Section VII.A) we additionally must identify the original frames found in the original media file.

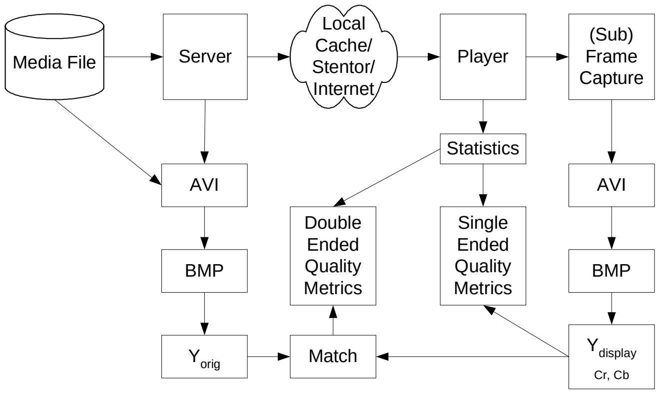


Fig. 9. Video Capturing and Extraction of QoS Metrics.

## VIII. EXPERIMENTAL DESIGN

The basis for the CODIS experiments will be the Sambits terminal together with enhanced MHP applications using both DVB and streamed MPEG-4. In order to obtain a complete impression about the CODIS QoS we will design a set experiments which will be defined by different values of the following parameters:

- Application. Here we define a subset of the Sambits applications and obtain QoS measures for all of them.
- Location. Here the source and receiver of content (geographical location) yield different pairs, for which the transport will be measured.
- Content. Here content with different type (still image, talking head, high motion, high spatial activity, …) will be chosen to be transmitted.
- Bitrates. The chosen content will be encoded with different bitrates.
- Number of receivers. Here, different numbers of clients will be defined to access both DVB and MPEG-4 in parallel.

Combining the above parameters will yield settings for the single experiments which will be measured accordingly.

[12] http://www.camtasia.com/
[13] http://www.hyperionics.com/

## IX. Preliminary Results

We have conducted preliminary experiments for the following reasons:

- To verify our own measurement and analysis software.
- To learn details about the used hard- and software provided by our partners.
- To compare different metrics and choose those that will be used for CODIS.
- To choose parameters for the content that will be sent and measured.

As the CODIS network is not set up yet, our experiments were based on IP networks and the Rohde & Schwarz equipment we have.

### *A. IP Network Measurements*

We evaluated the Internet performance from the University of Vienna to Toulouse, Metz and Munich in May 2002 using the tool pchar. We were able to measure characteristics such as packet loss percentage, latency and bandwidth successfully for the routes to Toulouse and Munich and with limited success for Metz. These measurements give us a good idea of the general performance of the internet for these routes as well as some idea of the bottlenecks. We took the measurements on a Dell notebook, with a Pentium III processor. The operating system was SuSE Linux (Pro 7.3) and we used pchar (version 1.4) and ping.

We executed pchar with the standard settings of packet sizes and repetitions (i.e. 46 different packet sizes - 32 to 1500 bytes with an increment of 32 - and 32 repetitions for each packet size which makes 1472 packets sent to each hop) and we took the measurements of pchar five times for both, the server in Toulouse and Munich.

We faced some problems while taking these measurements because of which we had to resort to different means of collecting the data. The 24$^{th}$ hop to Toulouse was unidentifiable (the IP address is unknown) and so pchar could not work beyond that. To work around this problem, we collected the round trip time and packet loss data using ping for each of the hops beyond the 24$^{th}$ and then used pathchar's algorithm to calculate the latency and bandwidth. The flaw remaining was that the estimated bandwidth for the hop from the 23$^{rd}$ to the 25$^{th}$ is actually two bandwidths (23-24 and 24-25) and the value calculated will be less than both of these bandwidths. To Munich, all hops were identified with no problem but pchar did not manage to send packets to the 23$^{rd}$ hop (zit-108.irt.de). We worked around this problem in the same way as above, but there was a missing or skipped hop again which did not respond to ping either. For the server in Metz, all hops after the 15$^{th}$ were unidentifiable and even traceroute did not manage to go beyond the 15$^{th}$ hop. For this server, we could only use ping to the destination and try and get the overall latency and bandwidth estimate.

We used ping the same number of times as pchar sent packets, i.e., 46 different packet sizes and 32 repetitions per size. The results can be seen in Table VIII and Table IX.

Table VIII
Mandatory Network Parameters.

| Location | Site | Hop count | Latency (ms) | Bottleneck Bandwidth (Mbps) | Packet Loss (%) |
|---|---|---|---|---|---|
| Munich | www.irt.de | 24 | 15.77 | 1.086 | 0 |
| Toulouse | www.codis-satellite.com | 26 | 69.45 | 0.984 | 0 |
| Metz | www.tdf.fr | -- | 13.39 | 0.482 | 5.45 |

Table IX
Latencies for all Hops.

| www.irt.de | | www.codis-satellite.com | |
|---|---|---|---|
| Hops | Latency (ms) | Hops | Latency (ms) |
| 0…1 | 1.473 | 0…1 | 1.454 |
| 1…6 | 0.989 | 1…6 | 1.649 |
| 6...7 | 3.736 | 6…8 | 3.447 |
| 7...8 | 0.146 | 8…9 | 2.995 |
| 8…10 | 1.848 | 9…10 | 5.328 |
| 10…11 | 3.320 | 10…12 | 0.497 |
| 11…16 | 0.449 | 12…13 | 36.09 |
| 16…17 | 1.517 | 13…17 | 0.161 |
| 17…18 | 0.093 | 17…18 | 9.497 |
| 18…21 | 1.846 | 18…22 | 0.039 |
| 21…24 | 0.352 | 22…26 | 9.283 |

For Munich the bottleneck lies between ifrt.Munich. alter.net and webby.irt.de. However one hop between these two did not respond to ping as well as pchar and so the exact bottleneck bandwidth could not be measured. Assuming that the two links have the same bandwidth, the bottleneck bandwidth is 1.086 Mbps. If they do not, then this is an upper bound.

For Toulouse the bottleneck lies between pos1-0-622M. cr1.ATL1.gblx.net and pos5-0-0-155M.ar1.TPA1.gblx.net which is the 23$^{rd}$ hop. This hop also gives the maximum round trip time (more than the total latency) probably because the processing speed of this router is low and the time to return the ICMP error is greater that the time it takes to forward a packet.

For Metz the individual hop data was not collected as the hops after the 15$^{th}$ were not known. Therefore the bandwidth value is misleading as the bottleneck bandwidth, for which we only have a lower bound.

### *B. Presentation Quality of DVB*

As a preliminary test a 20 seconds long MPEG-2 sequence with high motion was tested using the Rhode & Schwarz DVQ (see Section VII.D.1). The content was encoded at the data rates 3 Mb/s, 4 Mb/s, 6 Mb/s, and 9 Mb/s. As can be seen in Fig. 10, the 3 Mb/s version regularly shows significant quality drops, periodically dropping below the critical limit 20. The quality of the others remain at least acceptable, the 6 and 9 Mb/s versions showing a stable curve without significant drops.

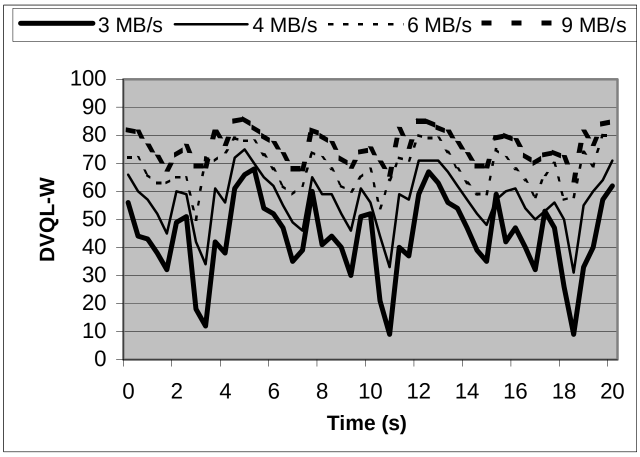


Fig. 10. DVQ-W values for different DVB bitrates.

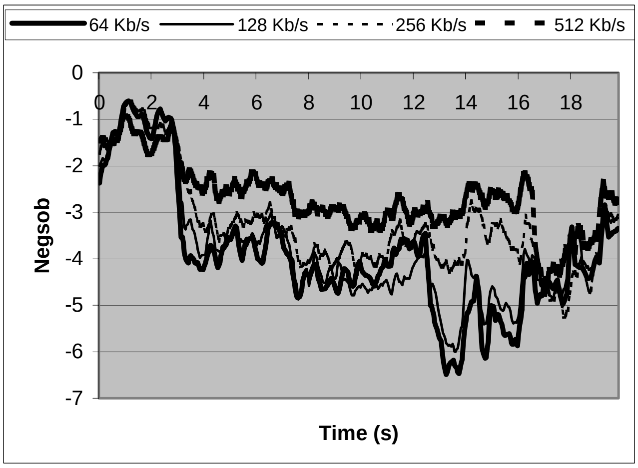


Fig. 11. Negsob for the first 20 seconds.

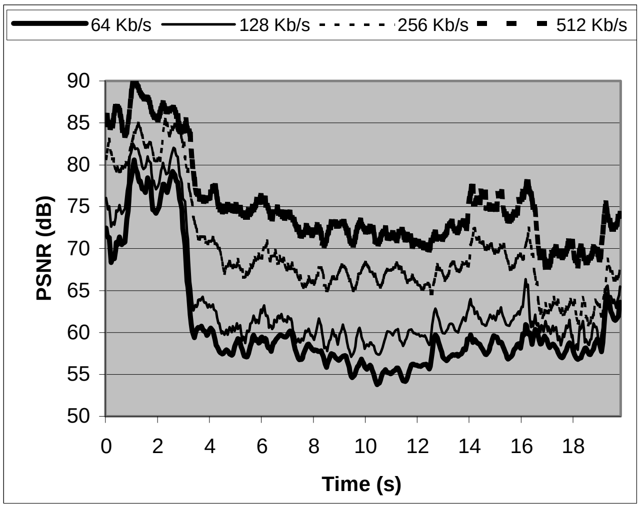


Fig. 12. PSNR for the first 20 seconds.

## C. Presentation Quality of Streaming Media

We took one minute from a popular computer animated movie ("Monsters's Inc.") showing high motion with many scene changes. This was encoded using the RealProducer Plus version 8 for all possible target audiences. Using the RealProducer's simulation tool we set the maximum bandwidth to the values 512 Kb/s, 256 Kb/s, 128 Kb/s, and 64 Kb/s. The presentations were then recorded using the tool Hypercam together with the Camtasia lossless encoder. After identifying the offset of the recorded subframes, for each recorded frame, the original frame from the movie was identified. The following measures were then calculated for each frame the Negsob $N$, the Possob $P$, and the *PSNR*. Additionally, the framerate for each second was estimated by comparing the recorded frames with their successors. The framerate was defined to be the number of different frames recorded in this particular second. Two successive frames being identical would not contribute to the framerate, thus lowering it accordingly.

As can be seen, at the scene start, the Negsob (Fig. 11) temporarily judges the 512 Kb/s stream to produce the worst quality. Furthermore, the Negsob temporarily judges the 128 Kb/s to be worse than the 64 Kb/s stream. On the other hand, both *PSNR* (Fig. 12) and Possob (not shown) produce consistent results judging high bitrate streams to produce better quality than the lower ones. However, sometimes the 64 Kb/s and the 128 Kb/s streams are judged equivalently.

The explanation for this can be seen in Fig. 13 showing the estimated framerates. As these have been computed from the received frames only (where some frames may be missing due to the unreliable frame grabbing method), the actual value might be a little higher. The 64 Kb/s stream has a significantly lower framerate than the 128 Kb/s. Thus, in order to save bandwidth, in this case the RealProducer chooses to lower the framerate only instead of further lowering the picture quality.

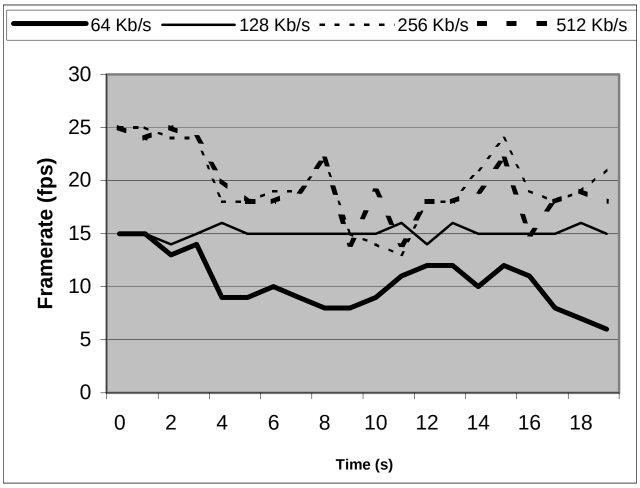


Fig. 13. Framerate lower bound for the first 20 seconds.

## X. Conclusion

In this paper the CODIS methodology for measuring the quality of service of a satellite based content delivery network has been described. The CODIS network transports either pure DVB content or IP data encapsulated in the MPEG-2 transport stream. Exploiting the favorable multicast properties intrinsic to satellite data communication, multimedia content may be moved easily from content providers to the network edges.

The measurement methodology consists of several layers. The first layer focuses on the capacity of the network to transport both IP data and pure DVB content over MPEG-2-TS to the network edges. For measuring the IP-network QoS, the relevant parameters and standards are given. Here, the consortium will use both freely available tools as well as instrument the content management system. The same is done for the DVB based content, where the consortium will rely on the equipment of Rohde & Schwarz.

The second layer focuses on the end user experienced QoS. Here we distinguish between the presentation quality and the response time. For both categories, metrics are specified that represent what is actually perceived and how users would subjectively judge such an observation.

## XI. Future Work

Our future work will include a further specification of the proposed experimental setup, a description of the necessary steps to obtain the measurements, and how to post process and interpret the results. Furthermore, we will define the contents to be sent, the exact QoS parameters to be measured and also aim at relating measures like *PSNR*, Possob and *TI* to subjective ratings.

## References


[1] *Digital Transport of Video Teleconferencing/Video Telephony Signals–Performance Terms, Definitions, and Examples*, ANSI Standard T1.801.2-1996, 1996.

[2] *Digital Transport of One—Way Video Signals–Parameters for Objective Performance Assessment*, ANSI Standard T1.801.03-1996, 1996.

[3] N. Bhatti, A. Bouch, A. Kuchinsky, "Integrating User-Perceived Quality into Web Server Design," in *Proc. WWW'00*, Amsterdam, 2000.

[4] N. Bhatti, R. Friedrich, "Web Server Support for Tiered Services", Technical Report HPL-1999-160, HP Laboratories Palo Alto, 1999.

[5] A. Bouch, M.A. Sasse, "It ain't what you charge it's the way you do it: a user perspective of network QoS and pricing," in *Proceedings of IM'99*, Boston MA, 1999.

[6] A. Bouch, A. Kuchinsky, N. Bhatti, "Quality is in the Eye of the Beholder: Meeting User's Requirements for Internet Quality of Service" Technical Report HPL-2000-4, HP Laboratories Palo Alto, January 2000.

[7] A. Bouch, M. A. Sasse, H. Demeer, "Of packets and people: A User-centered Approach to Quality of Service," in *Proceedings of IWQoS'00*, 2000.

[8] C. J. van den Branden Lambrecht, "Color Moving Pictures Quality Metric," in *Proceedings of ICIP 96*, Lausanne, Switzerland, 1996

[9] A. B. Downey, "Using pathchar to estimate Internet link characteristics", in *SIGCOMM'99*, 1999.

[10] *Digital Video Broadcasting (DVB): Measurement Guidelines for DVB Systems.* ETSI TR 101 290 V1.2.1 (2001-05), 2001.

[11] Frost & Sullivan, "World Caching, Streaming & Multicasting Via Satellites Markets", Report No. 7920-66, Available: http://www.frost.com

[12] *General quality of service parameters for communication via public data networks*, ITU-T Recommendation X.140, ITU, 1992.

[13] *Methodology for the Subjective Assessment of the Quality of Television Pictures*, ITU Study Group 12, Contribution 7, 1997.

[14] *Methodology for the Subjective Assessment of the Quality of Television Pictures*, ITU Recommendation ITU-R BT.500-10, 2000.

[15] *Principles of a reference impairment system for video*, ITU-T Recommendation P.930, ITU, 1996.

[16] C. Kuhmünch, C. Schremmer, "Empirical Evaluation of Layered Video Coding Schemes", in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, 2001.

[17] K. Lai, M. Baker, "Measuring Link Bandwidths Using a Deterministic Model of Packet Delay", in *Proceedings of ACM SIGCOMM* 2000, August 2000.

[18] M. Liste, "Content Delivery Networks (CDNs)—A Reference Guide," *Cisco World Magazine*, white paper, March 2001.

[19] J. Lubin, J. Fibush, Sarnoff, "JND Vision Model," ANSI T1A1 Working Group Document #97-612, T1 Standards Committee, 1997.

[20] M. Mathis, M. Allman, "A Framework for Defining Empirical Bulk Transfer Capacity Metrics", RFC 3148, IETF, July 2001.

[21] M. J. Nadenau, S. Winkler, D. Alleysson, M. Kunt, "Human Vision Models for Perceptually Optimized Image Processing–A Review," submitted to Proceedings of the IEEE, Sep 2000.

[22] J. Nielsen, *Usability Engineering*, Morgan Kaufman, San Francisco 1994.

[23] V. Paxson, "Measurement and Analysis of End-to-End Internet Dynamics," Ph.D. Thesis, Computer Science Division, University of Berkeley, 1997.

[24] V. Paxson, G. Almes, J. Mahdavi, M. Mathis, "Framework for IP Performance Metrics," RFC 2330, IETF, May 1998.

[25] Peakstone, "Peakstone eAssurance Concept Guide", 2001. Available: http://www.peakstone.com.

[26] F. Pereira, "Overview of the MPEG-4 Standard", ISO/IEC JTC1/SC29/WG11 N4668, March 2002.

[27] A. C. F. Pessoa, "Video Quality Assessment Using Objective Parameters Based On Image Segmentation.," ITU Study Group 12, Question: 11/12.

[28] R. Rajamony, M. Elnozahy, "Measuring Client-Perceived Response Times on the WWW," in *3rd USENIX Symposium on Internet Technologies and Systems (USITS)*, San Francisco, 2001.

[29] U. Reimers, *Digital Video Broadcasting (DVB). The International Standard for Digital Television*. Springer-Verlag Berlin Heidelberg, 2000.

[30] N.B. Seitz, S. Wolf, S. Voran, R. Bloomfield, "User-Oriented Measures of Telecommunication Quality," *IEEE Communications Magazine*, January 1994, pp. 56-66.

[31] Tektronix, "A Guide to Maintaining Video Quality of Service for Digital Television Programs", 2000.

[32] Video Quality Experts Group, "Final Report from the Video Quality Experts Group on the Validation of Objective Models of Video Quality Assessment," March 2000.

[33] Zhou Wang, "Objective Image/Video Quality Measurement – A Literature Survey," Available: http://www.ece.utexas.edu/~bevans/courses/ee381k/projects/fall98/zhou/literatureSurvey.pdf

[34] A. B. Watson, "Toward a perceptual video quality metric," *Human Vision, Visual Processing and Digital Display*, Vol. VIII, No. 3299, 1998, pp. 139 – 147.

[35] A. A. Webster, C. T. Jones, M. H. Pinson, S. D. Voran, S. Wolf, "An objective video quality assessment system based on human perception," in *SPIE Human Vision, Visual Processing and Digital Display TV*, vol. 1913, pp. 15-26, (San Jose, CA), Febuary 1993.

[36] S. Winkler, "Quality metric design: A closer look," in *Proc. SPIE Human Vision and Electronic Imaging Conference*, vol. 3959, pp. 37-44, San Jose, California, January 22-28, 2000.

[37] S. Wolf, M. H. Pinson, A. A. Webster, G. W. Cermak, E. P. Tweedy, "Objective and subjective measures of MPEG video quality," in *Proc. 139th SMPTE Technical Conference*, New York, NY, Nov. 1997.

[38] A. Wörner, "DVQ–An objective picture quality measurement not requiring a reference (NR)," Committee T1–Telecommunications Committee Group T1A1.1, T1A1.1/2000-044, Charleston, 2000.

[39] Feng Xiao, "*Final Project for EE392J"*, Engineering and Science faculty at Stanford University, Winter 2000.

[40] Zona Research Inc., "The Economic Impacts of Unacceptable Web-Site Download Speeds", April 1999.