Internet Engineering Task Force Internet Draft

Philippe Gentric, Philips Electronics

May 2003 expires November 2003

draft-gentric-mmusic-stream-switching-req-00.txt

Requirements and Use Cases for Stream Switching

STATUS OF THIS MEMO

This document is an Internet-Draft and is in full conformance with all provisions of <u>Section 10 of RFC2026</u>.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress".

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/lid-abstracts.txt

To view the list Internet-Draft Shadow Directories, see http://www.ietf.org/shadow.html.

Abstract

Stream switching is a technique used to change the data rate of a media being streamed, typically for the purpose of adaptation to the effectively available bandwidth of the network. This memo lists the use cases and the requirements for stream switching.

1. Introduction

Stream switching is a technique used to change the data rate of a media being streamed, typically for the purpose of adaptation to the effectively available bandwidth of the network.

The aim is that a real time streaming system can switch from stream to stream in order to vary the data rate. This requires that the same content is encoded as multiple streams at various bit rates.

This memo lists a number of use cases in section 2 and requirements in <u>section 3</u>.

<u>1.1</u> Typical usage context

The typical scenario is video distributed on demand, also known as "Video On Demand" (VOD). The situation is depicted in figure 1. This is the domain of RTSP [RTSP] servers. HTTP is typically used for the service/application i.e. provides the entry point, usually a RTSP URL. The media can be pre-recorded on file or can be a "live" source in which case the RTSP/RTP server acts as a relay.

		* * * * * * * * * * * * * * * *	* *		* *	* * * * * * * * * * * * *	* *
		*	*	HTTP	*		*
		* HTTP Server	*	<>	*	HTTP Client	*
		*	*		*		*
		* * * * * * * * * * * * * * * * *	* *		* *	* * * * * * * * * * * * *	**
		* * * * * * * * * * * * * * * * * * *	т т Т		ч- т.	* * * * * * * * * * * * * * *	ىلە بلە ،
		~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~	^ ^		^ ^		^ ^
		*	*	RTSP	*		*
		* RTSP Server	*	<>	*	RTSP Client	*
		*	*		*		*
		* * * * * * * * * * * * * * * *	* *		* *	* * * * * * * * * * * * *	**
		* * * * * * * * * * * * * * * *	* *		* *	* * * * * * * * * * * * *	**
		*	*	RTP on UDP UC	*		*
		* RTP Sender	*	>	*	RTP Receiver	• *
		*	*		*		*
media		*	*	RTCP SR	*		*
on	>	*	*	>	*		*
file		*	*		*		*

[page 2]

Internet	Draft	Stream Swi	tching Requirements	March 2003	
or	*	*	RTCP feedback	*	*
live	*	*	<	*	*
	* * * * *	* * * * * * * * * * * *		********	* * * * *

Figure 1: video on demand

<u>1.2</u> Generalities

The rationale of stream switching is based on the following premises:

. With the emergence of streaming on wide band networks the lack of congestion control tools for streaming has been creating an increasing level of concern among users and operators. Obviously it is highly desirable that these tools should provide a "generalized" stream switching framework i.e. should not depend on a given codec technology or particular network configuration.

. With the emergence of streaming on wireless networks where bandwidth fluctuations are the rule the need for such tools is becoming vital, which is acknowledged by some dedicated fora activity [<u>3GPP-alt-attr</u>] [<u>3GPP-BWS</u>].

. While codec based schemes (scalable coding schemes, fine grained scalable video etc) have been promoted for years in various standardization bodies they did not succeed to pass the next stage i.e. to enter the fora closer to product specifications for the following reasons:

. In the mean time the classical "constant bit rate" paradigm of codecs has led to (ever improving) compression efficiencies ... at constant bit rates.

. For a media distribution service (by opposition to a conferencing service where the situation is significantly different) there are only two paradigms:

. On demand (point to point). In that case since the distribution is point to point between the media server and the client, the key argument will be the coding efficiency i.e. the perceptualquality versus bit-rate ratio, which favors switching between hyper-optimized constant bit rate streams.

. Live (broadcast). In that case changing the rate of one unique encoder based on reports from a number of receivers is difficult to imagine. On the other hand simultaneously encoding the same

[page 3]

program at various bit rates is easy to deploy either with a point-to-point relay (which makes it equivalent to the previous case) or using some type of multicasting.

1.3 Vocabulary

We define a "program" as a set of "tracks", for example a movie is composed of an audio and a video track.

We define a "stream" as an encoded instance of a track, for example the video track of a movie may be encoded at 50kb/s, 150 kb/s and 400 kb/s using respectively H263 baseline SQCIF 7.5 fps, MPEG-4 SP@L3 QCIF 15 fps and MPEG-4 ASP@L3 CIF 30 fps, the audio track may be encoded at 5 kb/s, 20 kb/s, 48 kb/s and 80 kb/s using respectively AMR, AMR WB, AAC mono and AAC stereo.

We define one "flavor" of a program as a given set of streams (a pair for a movie, usually consisting in audio and video), for example 400 kb/s video and 80 kb/s AAC is the high quality flavor in the example above for which we have 12 different flavors (but some flavors may not always make sense).

We define a "switch-set" as the set of all the streams for a track or a program. A switch-set can be organized either as ordered first by track or first by flavor. Obviously switch-sets are prepared during the content production or deployment phase.

We define "down-switch" as a switch toward a smaller rate.

We define "up-switch" as a switch toward a higher rate.

We define the "effective rate" as the data rate that the network can sustain at a given moment, this is the smallest data rate among all the links in the path from sender to receiver, usually the last hop. The situation were several links would "compete" for this position makes modeling more complex but does not affect the overall rationale.

2. Use Cases

2.1 Home VOD service

A service provider has deployed a VOD service (RTSP+RTP) as part of some wired home Internet access (DSL, cable).

[page 4]

The service is designed to sustain N concurrent users watching N different movies (i.e. has a bandwidth of N*BR where BR is the nominal bit rate for the movies). However the same network is also used for Web browsing.

At some point of time there is a peak in traffic (TCP and/or streaming) and the last hop between a given head-end, as a result users experience congestion.

2.1.1 Without stream switching

The router queues overflow, all TCP traffic falls back to share whatever is left of the bandwidth by the streaming service.

TCP users are unhappy because of the small bandwidth.

Video users may be slightly unhappy because TCP causes a constant packet loss rate of several percents, because TCP sessions are constantly probing by increasing their rate, which causes video playback to be slightly affected, but the video users who have good (error resilient) decoders are almost not affected at all.

2.1.2 With stream switching

In this case the detection of packet losses causes all or most of the streaming systems to switch down to lower rates, leaving more bandwidth for the TCP traffic.

TCP users are happy because of the relatively high remaining bandwidth.

Video users are moderately unhappy because the lower video rates cause lower objective quality, also both types of traffic (TCP and RTP) still causes a constant packet loss rate of several percents, because there are constantly TCP and RTSP/RTP sessions probing by increasing their rate, which causes video playback to be slightly affected (but again this is decoder dependant).

2.2 GPRS wireless VOD service

The video service is a news service for GPRS handsets based on the availability of 5 GPRS slots (roughly 50 kb/s) for download. This bandwidth is divided in 5 kb/s for AMR speech and whatever is left for video which can be: (1) no video (2) 25 kb/s video "slide show" (3) 45 kb/s video at 10 fps.

[page 5]

The policy implemented in the radio system is that each authenticated user (or non-authenticated emergency user) must at least have 1 slot (i.e. voice has precedence other data).

Another policy implemented in the radio system is that all IP traffic is handled the same way and the system is tuned for minimum error rate, specifically the low level layers will try the maximum number of attempts for each radio packet while increasing the redundancy, before giving up.

2.2.1 Static user

The user is static in a very busy cell i.e. there are many people popping in and out of the cell (either physically moving or making short calls). They are causing rapid fluctuations in the effective GPRS bandwidth for streaming.

2.2.1.1 Without stream switching

When there are too many calls in progress in the cell the video user gets only 1 slot and the video session causes congestion (in the base station router). Catastrophic degradation follows with player freeze, difficulties in reconnecting etc...

2.2.1.2 With stream switching

When the bandwidth folds to 1 slot the decoder immediately detects it (e.g. by measuring the jitter). The feedback causes the source to switch off the video allowing the system to recover without having caused congestion.

If the bandwidth folds to 3 or 4 slots the system will have the opportunity to switch to the 25 kb/s "slide show" alternative.

2.2.2 Mobile user, fully transparent hand over.

The user is moving from cell to cell, some are very busy some are empty. We assume hand over from cell to cell is fully transparent.

This case is similar to the previous one.

2.2.3 Mobile user, non-transparent hand over.

In this case a hand over may cause the player to receive nothing during a certain time. We assume that the network does not loose any data i.e. that the data accumulated in the network during the hand over will eventually reach the player. The lack of data may

[page 6]

Internet Draft Stream Switching Requirements March 2003

cause an underflow depending on how the player de-jittering buffer have been configured. This type of problem is the primary reason why large de-jittering buffers (many seconds of data) are required for this type of networks.

2.2.4 Mobile user, radio problems

In this case the motion of the users causes radio problems (obstacles between the antenna(s) and the phone).

Assuming per configuration (see above) the radio network is "almost" lossless the congestion effect is multiplied i.e. when the radio bandwidth decreases packets will pile up instead of being discarded.

Apart from that (which will make the problem worse) this case is similar to 2.2.1. Switching both streams (audio and video) off if the effective bandwidth becomes really small may be an interesting behavior.

2.3 Feedback from congested routing device

An interesting use case to consider is the case when by some mean that we don't need to specify here the routing device experiencing congestion has a way to signal it to the RTP source (RTSP server in our case).

This case is similar to 2.2.1. with the advantage that the reaction will be faster.

2.4 GPRS video with server initiated video stream switching

The video configuration for GPRS as in 2.2 is documented by the 3GPP Packet Switched Streaming specification.

In the case the whole bit rate range is covered by a single video codec with the same configuration (MPEG-4 video Simple Profile Level 0). This means that the scenario depicted in 2.2 is possible as soon as the server receives feedback about the network conditions.

RTCP feedback or extended RTCP feedback can be used for this purpose. Direct feedback from the congested node as described in 2.3 is another possibility.

No other signaling is needed assuming that the video packets are

[page 7]

Internet Draft Stream Switching Requirements March 2003

sent as belonging to the same single RTP session.

This configuration is called "client-transparent", for that reason i.e. client implementations are expected to be robust to bit rate changes, including a complete cut off for many seconds (however some time-outs may occur)

2.5 GPRS audio with codec change

The audio service is a music on demand service based on the availability of 5 GPRS slots (roughly 50 kb/s) for download. The available bandwidth is expected to vary down to 5 kb/s. The switch set prepared for the service is as follows:

- (1) 5 kb/s AMR
- (2) 12 kb/s AMR WB
- (3) 20 kb/s AMR WB
- (4) 30 kb/s AAC mono
- (5) 40 kb/s AAC stereo
- (6) 50 kb/s AAC stereo.

The specific problem in that case, when compared to the previous one, is that there are several different decoders and/or decoder configuration involved (specifically there are 4 of them: AMR, AMR WB, AAC mono and AAC stereo which must be processed as different codecs). Therefore the server MUST signal a switch to the client since feeding AAC into an AMR decoder (or vice versa) may crash it.

This configuration is called "non-client-transparent".

[Note: although a Requirement section should not hint at the solution, the next paragraph will, in order to explore some additional requirements with respect to synchronization] The obvious thing to do is that the switch set should be set up within a single RTSP session between client and server using a different RTP session for each stream. This means that each stream will at least have a different Payload Type and in addition may be transported toward a different UDP port. This insures that the receiver can "perceive" that the server switched simply because one RTP session will not receive any packet

[page 8]

Internet Draft Stream Switching Requirements March 2003

anymore while another one will start to receive some packets. One key question is how the client will be able to seamlessly synchronize. RTP time stamps will be used but since they have a different random offset for each RTP session additional information is required.

Notes:

1) The way this is handled in RTSP is to use RTP-info in responses to PLAY commands in order to convey the mapping between the Normal Play Time (media time) and the RTP time stamps .

2) The way this is handled in RTP is to send RTCP sender reports containing the mapping between the sender wall clock and the RTP time stamps. Doing this has two drawbacks, firstly such packets may be lost, secondly the timing may be late.

3. Requirements

Requirements listed here are characterized by a number (e.g. "R23") a description (a sentence), "Utility" (Always, config specific, player specific, server specific, rare) and "Importance" (Critical, high, medium, low).

3.1 Out of scope requirements

This memo does not address the requirement for a way to convey the description of the switch set(s) (it would typically need a memo of its own).

This memo does not address requirements affecting the rate control algorithm itself. i.e. it is considered here as given that the rate control must provide a suitable target for the switch in terms of bit rate (for more on rate control algorithm for streaming see [TFRC]). In particular the rate control system must follow the following rules:

. For down-switches the target rate should be substantially lower than the effective bandwidth in order for the streaming system to "recover" i.e. to compensate the negative effects of running an excessive data rate for the amount of time Tsw needed to detect the problem then compute a target and execute the switch. The time it takes to "recover" i.e. to flush routing buffers and replenish the receiver buffers increases with Tsw and is in first order proportional to the difference between the (new) rate and the effective bandwidth.

[page 9]

Internet Draft Stream Switching Requirements March 2003

. Subsequently up-switches will be performed in order to "explore" and find the ceiling i.e. the effective bandwidth, this exploration MUST follow extremely strict rules in order to avoid congestion explosions.

In a similar fashion this memo does not address "application policy" issues such as:

. For some applications a complete switch off may be better perceived (or easier to bill).

. For some applications media type may create preferences; for example a music service will first reduce the video to a slide show while a news service would first switch from high quality stereo audio to low bit rate mono audio.

This memo also does not cover the multicast cases (i.e. simulcast) for which switching is performed by routers.

3.2 Minimal receiver perturbation: Seamless switching requirements

Seamless stream switching is obtained when the switch is performed in such a fashion that media playback is minimally disturbed from a player point of view.

This requirement divides in several key issues as follows.

<u>3.1.2</u>. Preventing gaps in media

Gaps in the media can have 2 causes, packet losses and discarded packets.

3.1.2.1 Packet losses

Losses (whatever the cause) create gaps. We assume here that retransmission and FEC are out of scope in as much as the solution should work without them. We will assume in the next section that losses occur due to routing buffer overflow, which is due to sending data at a rate that is higher than the link bandwidth.

R1: Prevent packet losses Utility: Always Importance: high

[page 10]

<u>**3.1.2.2</u>** Discarded packets</u>

The receiver may be unable to process incoming data for two reasons:

3.1.2.2.1 Random Access Point Required

Some decoders (typically video decoders) may need a Random Access Point (usually in video this is an "Intra" frame) in order to start decoding; a stream switching system that would switch "anywhere" in a stream would cause such receivers to discard data until such a RAP is found.

However, thanks to recent video compression technologies "well implemented" video decoders can restart decoding "anywhere". This is a by-product of implementing error concealment and resilience techniques. Note that for some extremely resilient implementations the capability to minimize the visible artifact when jumping "anywhere" is surprisingly good, while for more naive implementations the result can be awful.

R2: Switch on RAP Utility: player and config specific Importance: medium

3.1.2.2.2 Synchronization information unavailable

For audio and video playback accurate relative synchronization (a.k.a. "lip-sync") is a key requirement. In some application one may even prefer to switch the audio off rather than playing out of sync. The name "lip-sync" indicates the type of content for which this is an extremely critical feature: video displaying people talking (unfortunately videos not displaying people caught in the action of talking are rather the exception than the rule!) The issue at stake is that for RTP streaming in the context of a RTSP sessions the receiver expects to receive the required lipsync information in response to a PLAY command thanks to the RTPinfo field.

Quote from <u>RFC2326 section 12.33</u>:

"A mapping from RTP time stamps to NTP time stamps (wall clock) is available via RTCP. However, this information is not sufficient to generate a mapping from RTP time stamps to NPT. Furthermore, in order to ensure that this information is available at the necessary time (immediately at startup or after a seek), and that it is delivered reliably, this mapping is

[page 11]

placed in the RTSP control channel."

R3: Send sync info after switch Utility: Client non-transparent Importance: Critical

3.1.2. Preventing pauses in playback

Playback pauses are caused by buffer underflows: the receiver simply does not have data to decode and must therefore wait for some. There can be a number of causes (as follows) but all causes share the same precondition: the sender is pushing packets corresponding to a data rate higher than some hop (very often the last one) in the path to the client can sustain, the obvious strategy then is to perform a down-switch.

3.1.2.1 There was no down-switch

The buffers in the network will eventually saturate in high data rate packets while these packets take longer to arrive to the decoder than it takes time to decode them.

R4: Switch down to compensate bandwidth decrease Utility: Always Importance: Critical

The next issue is obviously to make sure that the switch-down signal arrives at the sender.

R5: Make sure the switch-down signal is not lost Utility: Always Importance: Critical

3.1.2.2 The down-switch occurred too late

If the switch occurs too late the result is similar: the routing buffers are still full of high rate packets which takes a long time to flush (this may depend on the router discard policy, if the router has the policy to discard the oldest data first this is not true, but then this policy will create gaps, ... see above)

R6: Switch down as soon as possible when congestion detected Utility: Always Importance: Critical

Note that the urgency to switch is roughly increasing with the

[page 12]

difference between the (old) rate and the effective rate, which leads to another requirement:

R7: Down-switch to the smallest bit rate available when congestion detected Utility: Always Importance: high

This last requirement can be interpreted by considering that the smallest bit rate is zero i.e. suppress one media, an example is a video news service where video is cut off while audio remains.

3.1.3 Preventing visible quality changes

Media quality is directly a function of the data rate. The obvious requirement is therefore to always use the highest possible data rate (which is in exact opposition with the previous item!):

R7bis: Down-switch to the highest bit rate available (but below the effective rate) Utility: Always Importance: high

A way to solve the contradiction is to defer to the rate control algorithm the responsibility to compute a low target rate so as to cause a fast recovery but to prepare for an up-switch just below the estimated available bandwidth as soon as the network conditions show signs of recovery. This effectively eliminates R7 and R7bis.

<u>**3.2</u>** Minimize network perturbation</u>

It is yet another key Requirement to minimally disturb the network.

3.2.1 Avoid accumulating data in network (routing) devices

Since the amount of storage in routing device is limited streaming traffic should behave and avoid using too much of this storage too often. It is also obvious that since data accumulates in routers in case of congestion, this requirement is exactly similar to the one above (R5) i.e. the key parameter is to switch down as soon as possible when congestion is detected.

3.2.2 Avoid sending redundant data

[page 13]

There are several possible reason why the sender may send redundant data. Obviously sending more data when the system is experiencing congestion is a very bad idea, on the other hand it is less important when switching up.

3.2.2.1 RTP session using retransmission or FEC

Obviously RTP sessions using some type of retransmission scheme or some type of adaptive FEC (Forward Error Correction) scheme will cause additional traffic in case losses are detected, which may worsen congestion.

R8: Avoid retransmission and addaptive FEC Utility: Always Importance: High

3.2.2.2 Back track to RAP

As mentioned above decoders may need a RAP to start decoding. The hypothesis explored above was that the decoder would discard data until a RAP is received, the reverse solution consist in having the sender back track the stream until a RAP is found and start sending the new stream at this point. This solution can be extremely costly in case the stream has few RAPs and the previous one is many seconds away.

R9: Avoid back tracking to RAP for down-switch Utility: video and configuration specific Importance: variable

3.2.2.3 Packetization overlap

Two streams encoding the same media at different rates may have packetization overlap. This is typical for audio in VOD where each packet contains as many frames as possible, i.e. up to the path MTU or some safe smaller value (in order to reduce the packet header overhead). In this case the time stamps of packets from streams at different rates coincides very rarely. This means that up to 1 packet equivalent of redundant media will be sent at the switch, which is not a lot of data except for very low bit rates (e.g. 4 kb/s audio packetized in 1500 octet datagrams have a packet rate of one packet every 3 seconds, an additional packet represents then a 30% peak rate increase!)

R10: Avoid packetization overlap Utility: Audio

[page 14]

Importance: Low

3.3 Minimize receiver resource usage

It is a requirement to minimize the amount of resources necessary to implement stream switching in the players. This is especially true for mobile clients. This requirement however is pretty weak due to the comparatively vast amount of resources required for media decoding.

R11: Avoid large receiver resource requirements Utility: Embedded players Importance: Low

3.4 Minimize sender resource usage

It is a requirement to minimize the amount of resources necessary to implement stream switching in the servers. This is only true for high volume servers, but it is extremely important in that case. Indeed high volume VOD servers are dedicated machines optimized for thousands of concurrent sessions. Cost effectiveness then depends on the ability of the implementers to produce more concurrent sessions for the same hardware configuration which resolves ultimately in the ability to switch context, which in turn depends critically on the amount of memory and CPU cycles each context individual cycle requires (see also section 1. of [TFRC]).

R12: Avoid increasing sender resource requirements Utility: High volume servers Importance: Critical

3.5 Minimize receiver security risk

The key risk for the receiver is to be the victim of an unexpected switch or a switch that it does not support.

<u>3.5.1</u> Switch with change in decoder configuration

Changes in decoder configuration are in general either not covered or explicitly excluded by compression standards. For example in MPEG-4 video it is explicitly forbidden to change the screen size in the middle of a stream (e.g. by sending a VO-VOL update), more generally nobody would expect a given decoder to detect that the content it is receiving has changed in nature (say from AMR to AAC!).

[page 15]

Internet Draft Stream Switching Requirements March 2003

R13: No unsignaled or unprepared switches involving decoder configuration changes Utility: client non transparent Importance: Critical

<u>**3.5.2</u>** Switch without change in decoder configuration</u>

This is the case when nothing changes but the bit rate (many codecs support this, but unfortunately usually over a restricted bit rate range).

In theory no signaling is required.

In practice there is an extremely high risk that some part of most existing implementations relies on the assumption that streaming is performed at a constant (average) rate, however adding explicit signaling would obviously not solve this backward compatibility issue either...

R14: Avoid unsignaled switches even if decoder configuration does not changes Utility: client transparent Importance: low to very low

<u>3.6</u> Minimize network security risk

The key security issue for the network is directly related to congestion avoidance, as such stream switching will be a benefit (when comparing with streaming without stream switching!) providing that it uses the correct rate control algorithm. In case the congestion problem is not handled correctly by the rate control system a nice safe feature would be that servers can be authoritatively limited in their output bandwidth.

R15: Use proven rate control algorithms Utility: Always Importance: Critical

R16: Allow servers to deny an up-switch Utility: Always Importance: Critical

<u>3.6</u> Minimize sender security risk

The key security issue for the sender is DOS in various forms, for which the defenses are simple:

[page 16]

R17: Allow servers to deny a switch Utility: Always Importance: high

R18: recommend that servers implement safe limits (max switch rate etc) Utility: Always Importance: high

3.7 Backward compatibility

Another key requirement is maximal backward compatibility with the relevant IETF standards: RTSP, RTP/RTCP, SDP

R19: Backward compatibility Utility: Always Importance: critical

3.7 Forward compatibility

Another requirement is maximal forward compatibility with the relevant future IETF standards for example RTSPv2 and SDPNG.

R20: Forward compatibility Utility: Always Importance: high

3.8 Table of Requirements

* * * * * *	* * * * * * * * * * * * *	* * * * * * * * * * * * * *	* * * * * * * * * * * * * * * * * * * *
* R#	Utility	Importance	Description *
*****	**********	*****	
R1 +	Always	High	Prevent packet losses
R2 +	Video	Medium	Switch on RAP
R3	Client	Critical	Send sync info after switch
	Non		
	Transparent		
R4	Always	Critical	Switch down to compensate
			bandwidth decrease
R5	Always	Critical	Make sure the switch down
			signal is not lost
R6	Always	Critical	Switch down as soon as possible

[page 17]

T			L
+ R8	Always	High	Avoid RTX and adaptive FEC
+ R9	Video	Variable	Avoid back tracking to RAP
R10	Audio	Low	Avoid packetization overlap
R11 	Embedded Clients	Low	Avoid large receiver resource requirements
R12 	Large VOD Servers	Critical	Avoid large sender resource requirements
R13 	Client Non Transparent	Critical	No unsignaled or unprepared switches involving decoder configuration changes
R14 	Client Transparent 	Very Low	<pre> No unsignaled switches even if decoder configuration does not change</pre>
R15	Always 	Critical	Use proven rate control algorithms
R16	Always 	Critical	Allow servers to deny an up-switch
R17	Always 	Critical	Allow servers to deny any switch (DOS resistance)
R18 	Always 	High	Servers should implement safe limits
R19	Always	Critical	Backward compatible
+			· _ · · · · · · · · · · · · · · · · · ·

<u>4</u>. Security considerations

See the security specific requirements in the above section.

5. References

[RTP]	htt	p://	/www.	ietf	.org/	/rfc/	/RFC1889.txt
-------	-----	------	-------	------	-------	-------	--------------

[RTSP] <u>http://www.ietf.org/rfc/RFC2326.txt</u>

[page 18]

[TFRC] http://www.ietf.org/rfc/RFC3448.txt

[3GPP-alt-attr] http://www.3gpp.org/ftp/tsg_sa/WG4_CODEC/TSGS4_22/Docs/S4-<u>020407.zip</u>

[3GPP-BWS]

http://www.3gpp.org/ftp/tsg_sa/WG4_CODEC/TSGS4_25/Docs/S4-<u>030024.zip</u>

6. Authors' Addresse

Philippe Gentric Philips MP4Net 51 rue Carnot 92156 Suresnes France e-mail: philippe.gentric@philips.com