

Q-HTTP Frequently Asked questions

List of questions

| | | |
|----|--|----|
| 1 | What is quality at application level? | 2 |
| 2 | What is Q-HTTP and what is its purpose? | 2 |
| 3 | Is Q-HTTP a protocol to achieve measurements? Is it comparable to RTCP? | 2 |
| 4 | Is Q-HTTP a resource reservation protocol? Is it comparable to RSVP? | 2 |
| 5 | What is the meaning of QoS-ALERT message? | 3 |
| 6 | How does Q-HTTP perform measurement and how accurate is it? | 3 |
| 7 | How long takes Q-HTTP to react to network degradation? How long takes the Network to achieve a quality upgrade? | 4 |
| 8 | Is Q-HTTP able to measure the TCP throughput? | 5 |
| 9 | Who is granted to ask for quality upgrades/downgrades in Q-HTTP? Final user or ACP? | 5 |
| 10 | What is "QoS-level dictionary"? | 5 |
| 11 | Q-HTTP advantages versus static resources reservation. What is statistical multiplexing? | 6 |
| 12 | May an application modify the Q-HTTP quality thresholds required in the same real time communication? | 7 |
| 13 | What would be the best Q-HTTP server implementation? | 8 |
| 14 | What types of implementations are possible in Q-http? | 9 |
| 15 | Does Q-http break the Internet e2e principle? | 10 |
| 16 | Using Q-HTTP implies to change the interface between ACP and OAS? | 11 |
| 17 | Does Q-HTTP allow quality downgrades without risk? | 11 |
| 18 | What will happen in environments without Q-HTTP? Why is positive to ALU to standardize the Q-HTTP protocol? | 11 |
| 19 | How much will cost a quality service without Q-HTTP ? | 11 |
| 20 | Why does Q-HTTP allow a sustainable way of internet traffic growth? | 12 |
| 21 | What is the Money flow of the Business model? | 12 |
| 22 | How should/could the service be commercially offered to the ACP? | 13 |
| 23 | May this solution be relevant on the wireless side? | 15 |

1 What is quality at application level?

Error detection and correction is dealt with at the lower levels of the protocol stack; as a result, quality is perceived by application level as a set of four network parameters:

- **Latency:** the time a message takes to go from origin to destination. Usually, it is near to $RTT/2$ (Round trip time), assuming the networks are symmetrical.
- **Jitter:** or packet delay variation. There are some formulas to calculate Jitter, and in present context we will consider the statistical variance formula
- **Bandwidth:** to assure the quality, a protocol MUST assure the availability of bandwidth required by application.
- **Packet loss:** the percentage of packet loss is closed related to bandwidth and jitter. It affects bandwidth because a high packet loss implies sometimes retransmissions that consume extra bandwidth, other times the retransmissions are not achieved (for example in video streaming over UDP) and the information received is less than the bandwidth requirement. In terms of jitter, a packet loss sometimes is seen by the destination like a larger time between arrivals, causing a jitter growth.

2 What is Q-HTTP and what is its purpose?

Q-HTTP is an application level Client/Server protocol which pretends to continuously measure session quality for a given set of flows, end-to-end in real-time; and raise an alert if quality parameters are below a given threshold. The thresholds of each application are different, depending on the nature of each application. Q-HTTP does not describe either the actions carried out to deal with the alert or how to implement them.

Q-HTTP is session-independent from the application flow/s, in order to not impact them. To perform the measurements, two control flows are created in both directions (forward and reverse).

Note: see questions 4 and 13

3 Is Q-HTTP a protocol to achieve measurements? Is it comparable to RTCP?

Not at all. Q-http defines three phases with different purposes, and inside these phases the negotiated measurement procedure is used. Different measurement procedures can be used (even RTCP itself) inside Q-HTTP. In fact, Q-HTTP only defines how to transport SLA information and measurement results as well as providing some mechanisms for alerting.

4 Is Q-HTTP a resource reservation protocol? Is it comparable to RSVP?

Q-http does not ask for resources. Q-HTTP only alerts if one (or some) of SLA quality parameters are being violated. It depends on server (ACP) to do something with this information in order to upgrade the connection and return it back to a SLA-compliant state. The most logical implementation involves the OAS, which acts as an intermediate to ask for flow upgrades to the different ISPs (reserving resources, changing priority flows or making any other kind of actions over the network).

5 What is the meaning of QoS-ALERT message?

This is the message that Q-http generates when the measurements indicate that quality SLA is being violated. It is an informative message which indicates that the user's experience is being degraded and includes the details of the problem (bandwidth, jitter, packet loss measurements and the SLA). The QoS-ALERT message does not contain the actions to take, which depends on the agreements between all involved parties ACP, OAS and ISP.

6 How does Q-HTTP perform measurement and how accurate is it?

Q-http can use any measurement procedure. The measurement procedure "default" is designed to be simple and interoperable, even in low performance equipments (Mobile, STB, ...). The default procedure measures both directions (forward and reverse) using a control flow independent from application own flows in order to avoid disturbance. The measurement is achieved using a sliding time window containing the last samples.

The precision depends on the resources used in the control flow. For example, in our lab a 15kbps control flow provides a reaction time of 1 second when network parameters (latency, jitter or packet loss) degrades a 5%, and a precision of 10^{-3} in packet loss using a window of 50 seconds or 10^{-2} using a window of 5 seconds.

To achieve better accuracy or reaction time, control flow resources may be increased. A 150kbps allows a reaction time of 100ms and a precision of 10^{-3} for packet loss in 5 seconds window.

On startup the server sends to the client the control flow resources to be used and in consequence the reaction time and time window is set.

The Q-HTTP bandwidth consumption configuration is very flexible (from 1kbps to hundreds of kbps and can be different in downlink and uplink), allowing different sensitiveness. This allows adapt Q-HTTP to different environments (3G, LTE, wireline, etc).

7 How long takes Q-HTTP to react to network degradation? How long takes the Network to achieve a quality upgrade?

According to previous question, Q-HTTP can react in less than one second, but the complete scenario takes more time. First of all, Q-HTTP alerts ACP (one second), then ACP ask for Quality upgrade to OAS, which acts over network management elements. The last element of this chain are provisioning systems such as 5620 SAM and/or AWS. The performance on these elements is critical.

It is a good practice to set conservative thresholds, better than needed in order to react just before user experience is affected.

| Chain element | Description of time | value |
|---|--|--|
| Q-HTTP | Time to react | 100ms - 1sec (1) |
| ACP--->OAS | Time to invoke API | Around 200ms |
| OAS--->Management system | Time to invoke Management systems | Around 200ms |
| Management systems (5620 SAM, 5529APC) | Time to perform upgrade over elements | 5529APC:620ms 5620SAM:600ms |
| Degraded user experience | Time to upgrade | 0-1 second (2) |

(1) 100 ms for a sudden heavy degradation versus 1 second detection time for a slow degradation in the SLA.

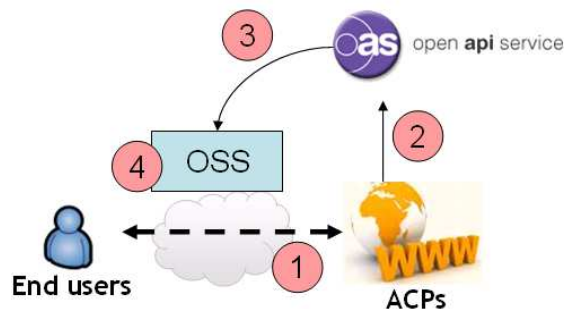
(2) There are 2 different cases:

- if the degradation is growing linearly, then the time of degraded user experience may be zero seconds (even when Q-HTTP reaction time could rise up to 1 second), because Q-HTTP thresholds are better than needed. It means that the user experience is not degraded yet when Q-HTTP alerts ACP. If the degradation does not grow quickly enough, the upgrade could be performed before user experience is affected.
- If the degradation happens instantaneously (i.e. network crash) then Q-HTTP does not need 1 second to react, but all the upgrade process could take up to 1 second in the worst case.

Conservative Threshold examples:

- Video packet loss tolerance is around 5%. Threshold of 3% could be set
- A maximum latency for gaming may be 100ms. Threshold of 90ms could be set

If a variable bit rate video streamer is integrated with Q-HTTP server, this second of bad user experience could be avoided reducing the video resolution during the time spent to upgrade.



8 Is Q-HTTP able to measure the TCP throughput?

TCP throughput is not considered a single parameter but a lineal combination of different net parameters. It depends on net bandwidth, net delay, and it depends on the stack TCP/IP implementation of the Operating System too.

$TCP\ Th = \text{Function}(RTT, \text{packet loss}, \text{stack implementation})$

Default measurement procedure of Q-HTTP is able to provide an estimated value of TCP throughput, and depending on this result, the chance to dynamically modify the input values (RTT and packet loss) is open, in order to get a constant TCP throughput.

9 Who is granted to ask for quality upgrades/downgrades in Q-HTTP? Final user or ACP?

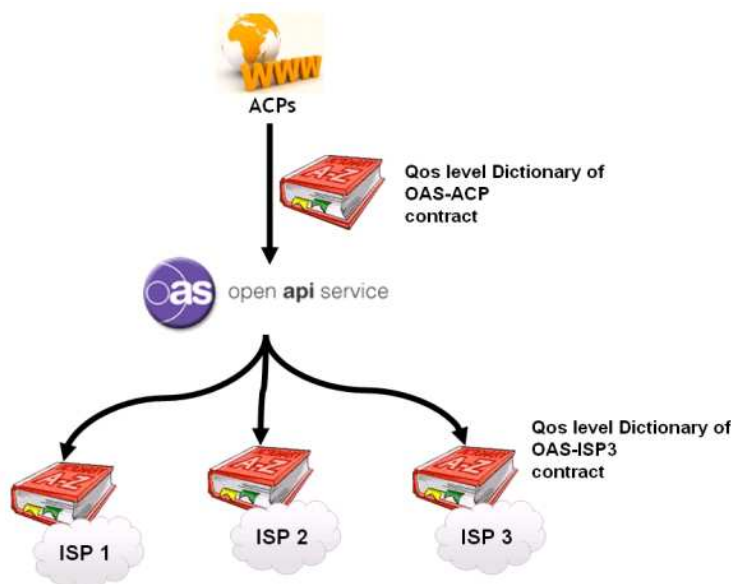
The contract is defined between OAS and ACP. The ACP pays for the upgrades and therefore the ACP is the only one who has this privilege.

10 What is "QoS-level dictionary"?

The QoS-level dictionary deals with the implementation of the different qos-levels used in Q-HTTP. However, the definition of the dictionary is out of scope of Q-HTTP. For example, the level 0 could be "best effort", level 1 could be "priority at access node", level 2 could be "priority at core" and level 3 could be "reservation". The meaning of each level is assigned by OAS, and should be present in the contract between OAS and ACP.

QoS-level dictionary is offer by the supra-entity (OAS) to the ACPs. The OAS should be able to offer a unique and consolidated dictionary, taking into account the different possible actions to take over each ISP network, depending on its technology base.

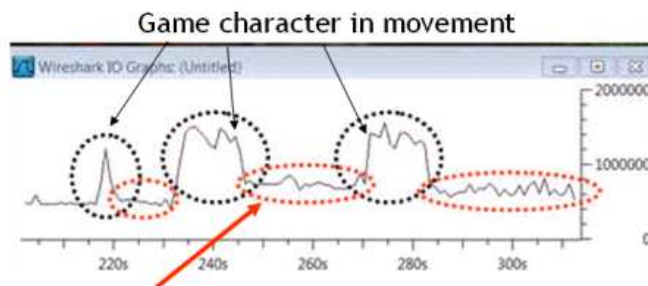
The set of available actions in each ISP constitutes the qos-level dictionary of that operator.



11 Q-HTTP advantages versus static resources reservation. What is statistical multiplexing?

The added value by Q-HTTP to current QoS management models (e.g. single measurement - or no measurement at all - and resource reservation at the beginning of the data exchange) is that enables dynamic measurement and resource allocation (nevertheless, static behavior is always an option if you set protocol options and dictionary so; thus, current model is actually a subset of Q-HTTP proposal).

This is a key feature for many applications which are highly dynamic in terms of bandwidth consumption: a connection may work perfectly: a connection may work perfectly right without extra effort (BE), in that case the resource reservation is an unnecessary waste. Due to the fact that the bandwidth used by a connection is not constant, statistical multiplexing offers great advantages, only if not doing the static reservation in the peak (allows two simultaneous Best Effort users instead of one full reserved user).

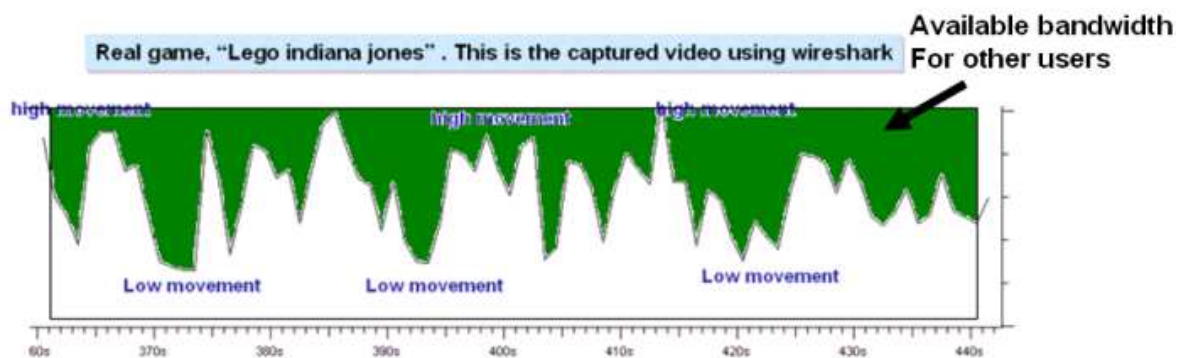


When SuperMario stops, downstream rate falls down dramatically

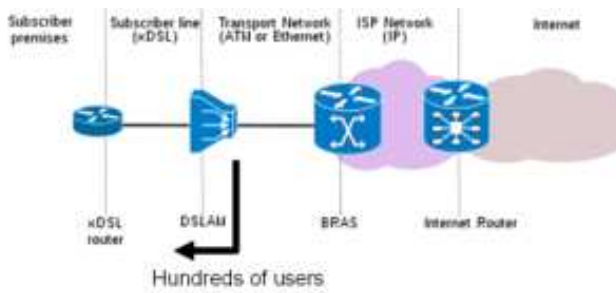
In the figure, when the character stops, the video stream rate falls down. This image is taken from a real experiment using "stream my game" tool (see www.streammygame.com). The flow rate is clearly changing with character movements.

Video streaming at constant rate has quality degradations when movement increases. This is because of the nature of codification mechanisms like MPEG-2 or MPEG-4. Only if the rate is variable then the quality can be constant. In videogames, streaming MPEG-4 is not possible due to the time needed to compress the video, but MPEG-2 or variations can be used.

This is the "streammygame" video capture rate evolution in time, during a real game (a real player). As you can see, the variation of flow rate is very high. The quality is always constant, thanks to the variable bitrate. The probability to be in a peak is very low, as well as the probability to be in a minimum. However it is very intuitive to appreciate that there is a lot of available bandwidth if we don't reserve the peak (the green zone)



Statistical multiplexation is possible in last mile, even though technologies like DSL or GPON do not share physical medium here: each DSL link runs over dedicated copper from a DSLAM port in the local office to a customer's home ; there are hundreds of users that can benefit from statistical multiplexing from the access device (DSLAM, ONT) to the core network all through the IP access layer of the network. Depending on the actual QoS enhancement implementation, core network and transit traffic may also benefit from classification made at the access layer, triggered by Q-HTTP alert.



Apart from stat mux at DSLAM output, we must consider Cablemodem access which has a shared architecture, also GPON (64 users) and also wireless access can benefit from stat mux at last mile.

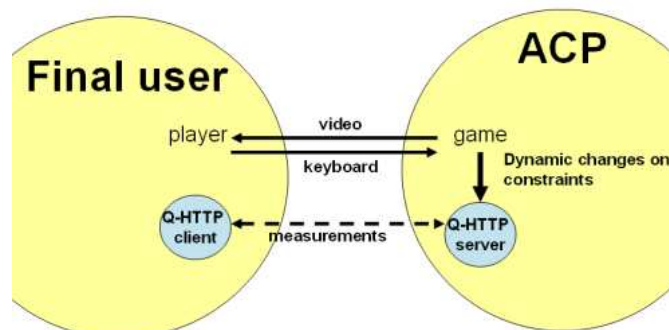
12 May an application modify the Q-HTTP quality thresholds required in the same real time communication?

Q-http is able to change dynamically network parameter thresholds. However, this possibility is not intended to be used to follow aggressive bandwidth changes (i.e. character movements. See question 11).

In addition, a change on thresholds does not imply an OAS request to change qos-level, simply it is possible to increase/decrease tolerance on quality measurements.

The main two use cases of dynamic thresholds changes are:

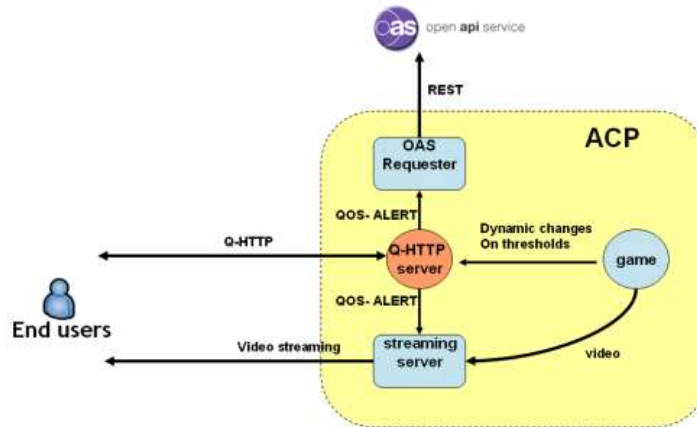
- When the application requires a constant TCP throughput and the latency has been degraded, then packet loss constraint can be changed in order to have the same TCP throughput, (see question 8).
- When the application has two or more phases with big differences in their requirements, for example, a menu and the arcade itself. In this case the application should communicate with Q-HTTP stack in order to change dynamically the constraints. Today this possibility is far, because games are not designed from the beginning to benefit from Q-HTTP, but it could be done in the future.



13 What would be the best Q-HTTP server implementation?

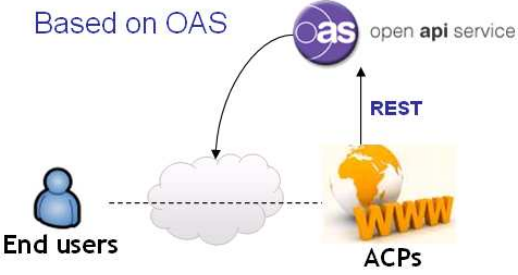
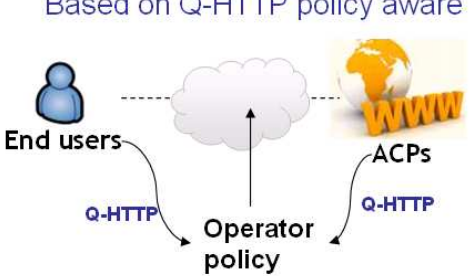
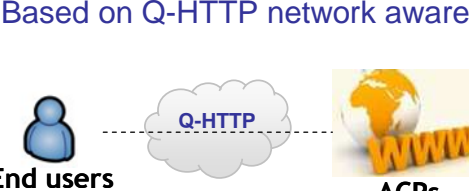
The best implementation of a Q-HTTP server should have:

- Interface with game: for dynamic changes on thresholds linked to phases (menu, video intro, arcade...)
- Interface/integrated with OAS: for request an upgrade/downgrade when needed.
- Integrated with video server: for increase/decrease resolution during the time awaiting for the upgrade confirmation



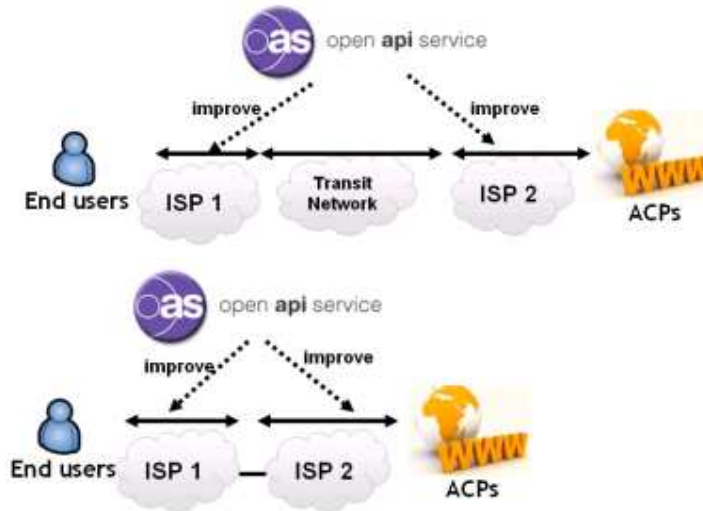
14 What types of implementations are possible in Q-http?

There are three identified types of implementations; however the first one (based in OAS) is the preferred and feasible one:

| | |
|---|--|
| <p>Based on OAS</p>  <p>End users</p> <p>ACPs</p> | <p>1. Based on OAS:</p> <p>The most logical implementation. OAS is the contract owner and manages the different QoS dictionaries/available-operations of the different ISPs, showing a unified set of levels to the ACP. ACPs can send Q-HTTP measurements to OAS in order to take the most intelligent decision</p> |
| <p>Based on Q-HTTP policy aware</p>  <p>End users</p> <p>ACPs</p> <p>Operator policy</p> | <p>2. Based on Q-HTTP policy Server</p> <p>This implementation is recommended for trials and proof of concept. In addition, it could be an option when OAS solution is not available or when an ISP wants to offer the service by itself. The disadvantage is the Management of Only one ISP qos-level dictionary, and the complexity of "roaming scenarios" in which different policy servers from different ISPs must communicate to share customer information and network status.</p> |
| <p>Based on Q-HTTP network aware</p>  <p>End users</p> <p>ACPs</p> | <p>3. Based on Q-HTTP network</p> <p>Futuristic implementation of a network Q-HTTP aware, with auto-tuning capabilities. It has the same disadvantages of implementation 2, but could be solved in the long term as Q-HTTP becomes a standard and "dictionaries" are agreed between most vendors/ISPs depending on each technology capabilities and network design.</p> |

In the OAS solution, when there are 2 ISPs and a transit network involved, the OAS could have the e2e measurements (client-server) using Q-HTTP and could act over both ISP's networks in order to reach the quality requirements. If improvements on the flow in each ISP solve the problem, the service could be offered. If the problem added by transit network can not be compensated by improvements in both ISPs, the service can not be offered. Anyway, certain QoS dictionary implementations may also involve also transit networks (requesting QoS markings to be honored in transit). This will leverage a wide business model based on QoS traffic exchange both at the edge of the internet (subscribers and ACPs) and in the core and transit carriers.

Simpler scenarios involve 2 ISPs without transit networks (connected through a neutral node, for example) in this case, the OAS could have full control over the policies applied in each ISP, and even could improve more one ISP than the other, in order to save money or resources



15 Does Q-http break the Internet e2e principle?

Regarding the Internet end2end principle:

"The end2end principle states that, whenever possible, communications protocol operations should be defined to occur at the end-points of a communications system, or as close as possible to the resource being controlled. According to the end-to-end principle, protocol features are only justified in the lower layers of a system if they are a performance optimization"

The "whenever possible" is the key. Clearly when a real-time service is offered, the mechanism to guarantee the quality constraints for the service could be something "diffserv or reservation on demand". Diffserv is standardized in IETF, and Q-http is an application level protocol to help to provide diffserv or reservation on demand. Therefore, Q-http goes not against IETF philosophy.

In addition, the Path Computation Element (PCE) working group of the IETF has complementary philosophy and could benefit from Q-HTTP. The PCE defines the procedures to calculate an optimal inter-domain (routing domain/layer/AS) path under Traffic Engineering constraints preferably in a (Generalized-) MPLS environment. TE information are retrieved from OSPF-TE or ISIS-TE databases. But, PCE procedures are entirely configurable and could be extended to economic objectives and/or QoS constraints. Q-HTTP measures in real-time the e2e path and these measurements may help PCE in the process to find a suitable path, or to reconfigure an inter-domain path if it is needed. PCE would be one of the functions of the Administrative Owner interfacing the ISPs, in our case "Open Api Service".

In a general scenario there will be two ISPs at the edge and one or more transit carriers in the middle. PCE defines the procedures to calculate an optimized intra-domain path based on economic and QoS constraints considerations. Q-HTTP may become the trigger to a set of changes in the network path between a client and a server, depending on which QoS level we are. First levels are intended to be the simplest and they would only involve edge ISPs optimizations; if this is not enough to meet service constraints, we may start thinking of a deeper optimization which may also involve transit carriers: first they should be able to honor edge QoS markings; second they might choose a different path through their available transit networks, based on detailed topology information - or cost, or whatever - further beyond current AS hops "length" information

Although there are many technical and non-technical issues to be addressed before implementing such a thing like a global (ISP-TRANSIT-ISP) capability to choose an optimal

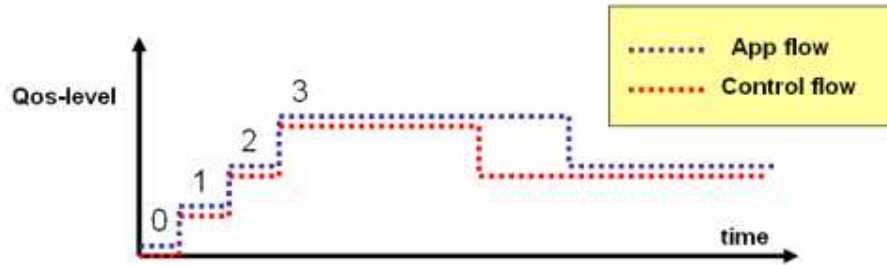
path, we believe that PCE fits into the global picture and makes it possible in a general scenario: QHTTP identifies the flow and the speakers and provide the e2e real-time measurements, PCE becomes a key part of the dynamic optimization process and value-added services on top of network infrastructure.

16 Using Q-HTTP implies to change the interface between ACP and OAS?

No, Q-HTTP only needs a light modification to include some extra information (for example, the QoS-Level required and the measurements).

17 Does Q-HTTP allow quality downgrades without risk?

Yes. In fact Q-HTTP measurements are achieved using a control flow separated from application flows, and this control flow can be downgraded to check an inferior qos-level . If measurements at that inferior level meet application requirements (with a security margin), then the application flow can be downgraded without risk



18 What will happen in environments without Q-HTTP? Why is positive to ALU to standardize the Q-HTTP protocol?

In environments with no standardized Q-http protocol, the OAS could offer only reservation, and this "binary dictionary" of OAS (reservation/ best effort) would limit the number of potential simultaneous users in the network, raising the costs for ISP, thus for ACP and logically for final users, putting on risk the massive feasibility of the business

If OAS offer a set of quality levels, uncountable proprietary methods for measurement of quality may appear and they will measure and alert the ACP, but depending on the implementation in some cases ACPs will know how to use the OAS type services, but in other cases they will not know how to use them, in that cases the OAS could be collapsed or blocked. ALU would need to develop and provide a set of "Q-HTTP like protocols" for different environments (for the browser, for the application, ...) instead of using the development of potential third parties.

Standardization is the better way to reach every ACP and every client.

19 How much will cost a quality service without Q-HTTP ?

The additional cost for ISPs without Q-HTTP is around 40% more, because the investments to support the reservation of generated traffic are higher. This increment is translated to every ACPs and final user.

Working without Q-HTTP forces the static reservation during the whole connection's life. Moreover, due to the lack of reliable information on the actual quality conditions, the only way to guarantee a good user experience is to perform a reservation of QoS end to end, while Q-HTTP only uses this option as its last option, it means, only when other options fail to achieve required quality. Those static peak reservations (when the application really does not need it) provoke a very high cost to the ISP, so the cost is very high to the ACP too, and consequently this extra cost may be passed on to customers; it makes the price of the service more expensive and less attractive.

20 Why does Q-HTTP allow a sustainable way of internet traffic growth?

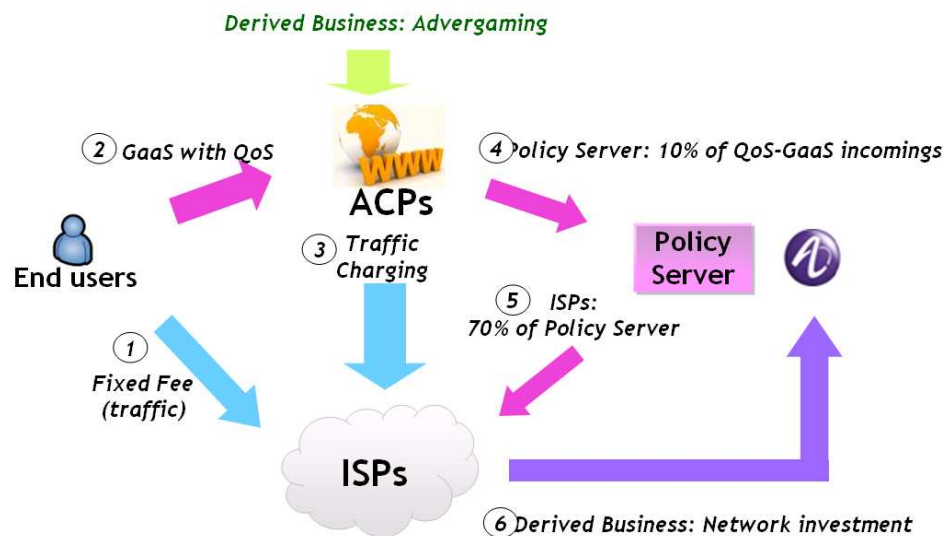
Today, monetize traffic growth represents a problem. With Q-http model of qos-levels, the ISP can take benefit from statistical multiplexing and as a consequence there is an important reduction in ISPs investment to support the traffic growth. With the 12% of revenues of this business is possible for ISPs to make the needed investments to support the new traffic. So the process will be auto sustainable.

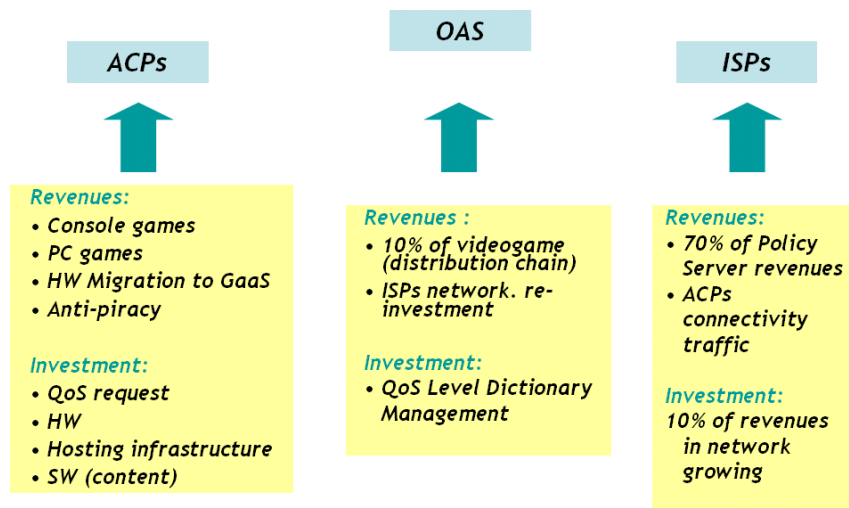
Q-HTTP allows to monetize the network with services that are used massively, charging for the resources used by the ACP in an intelligent way – charge based not on pure volume but on a value added scenario -: for best effort traffic the resources are cheaper, for reservation the charge is higher, in between these two, there is a set of QoS levels available depending on QoS dictionary . Operators will have two benefits, first they receive extra income for these services and they monetize the traffic growth providing an extra cost on each QoS level so that income does not only comes from gross traffic volume: there is a value-added in the same traffic volume for certain traffic classes and, thus, an extra income for the same traffic. The same idea may apply to pure transit carriers which must also engage in the roaming scenario.

This value-added charge model is an idea already put on the table by carriers and ISPs such as Telefonica, whose concern on traffic growth is high, because for current model (flat monthly charge, with or without traffic limits), income growth does not follow traffic growth.

In addition, operators like Telefónica want to offer “virtualized services on demand”, but in a sustainable way.

21 What is the Money flow of the Business model?





22 How should/could the service be commercially offered to the ACP?

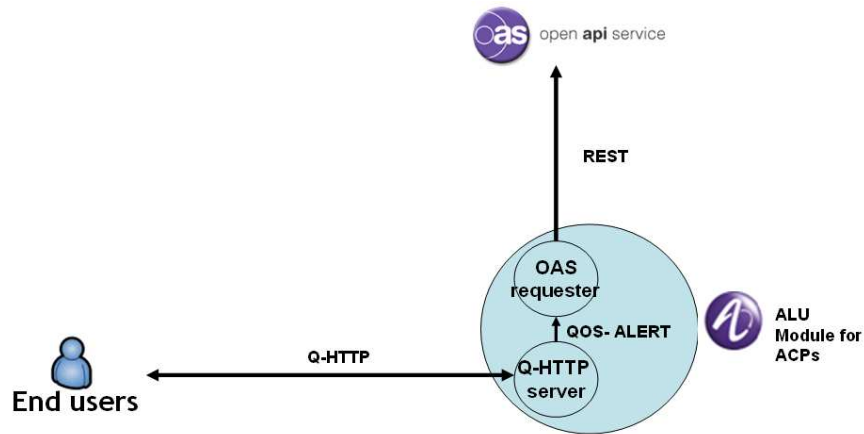
The Parameters to take into account in the price may be:

- Qos level Dictionary need to be clear in the service offered and simple for ACPs, for example 3 levels:
 - Best effort (1-BE),
 - prioritized traffic without full QoS guaranteed (2-PT),
 - resource reservation (3-RR).
 Other models are possible, but always with a reduced and clear number of levels.
- Time spent in each level used, during the session
- If user and ACP are both connected to same ISP: We'll consider the simplest case: both connected to same ISP. For example, we may assume price to be x1,5 (cost of negotiations with two networks)
- Timeframe of the day (network busy hour, not busy hour): Normally session will require a more restrictive service when network is congested, in BH, so this is implicit in the previous parameters

5 different commercial models are indicated here as examples:

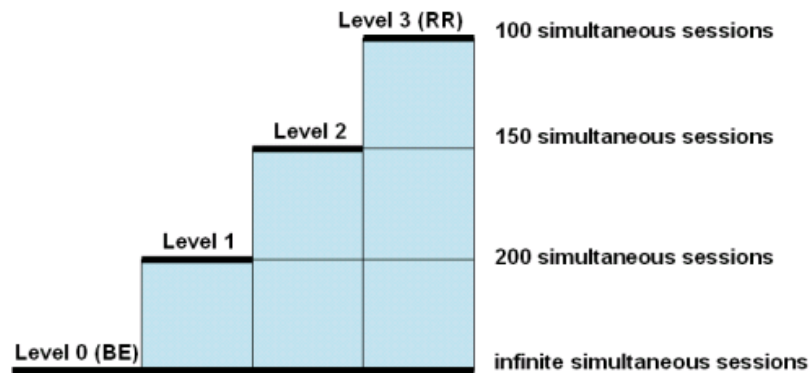
Option A) embedded transparent dictionary for guaranteed service

ALU (OAS) could offer a guaranteed service to ACPs, in which the Q-HTTP server is a piece of software delivered by ALU. This piece of software implements Q-HTTP protocol and invoke OAS API to take some actions when qos-alert is received. This commercial offer hides all complexity (management of levels, dictionary, etc) to the ACPs. The piece of software would be mandatory for ACPs who contract this guaranteed service.



Option B) level scale

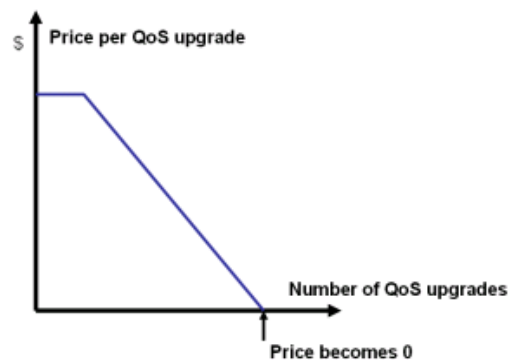
other interesting option is to charge a fix quantity of money for a given number of simultaneous users and to offer a different number of simultaneous users depending on the QoS level. In this model , ACP must be aware of the levels



Option C) payment per minutes spent in each QoS level (service for ACP)

In this option ACP is aware of qos levels. The charging is associated to:

- Tariff for BE (as now)
- Tariff for the usage of upgrades. Each alert and successful alert is registered. The payment to the OAS is incremented in the following way:
 - Price per minute consumed in each qos level: PT, RR
 - Volume discounts when nr. minutes increases (see figure)
 - RR more expensive than PT. PT more expensive that BE.



Option D) Guaranteed service and discount model in each session (downgrades)

In this option ACP is charged by number of sessions with resource reservation . The payment to OAS is decremented in the following way:

- Price per session with decremented service: PT, BE
- RR more expensive than PT. PT more expensive than BE.

Time spent in each level may be also a variable of this model

Option E) revenue sharing

ACP pays a percentage of the revenues per session for quality of service

- For all the sessions where QoS is applied
- It does not include Best Effort connectivity

23 May this solution be relevant on the wireless side?

Of course. The virtualization videogames and any type of applications are an opportunity in the mobile market.

For example, smart phones (preferable LTE) could connect to an ACP for playing virtualized games running on handheld consoles in the cloud (Nintendo DS, SONY PSP...) or even applications for other powerful platforms like PS3, XBOX...

This market represents an increment close to 30% in in the GaaS (game as a service) revenues.

END OF DOCUMENT