ALCATEL-LUCENT

Q4S: Quality for Services Architecture design proposal

Innovation & Research Group, Spain

February 2011

Version 1.0

Historial de Cambios

Fecha	Version	Author	Description
17/04/09	Version 1.0	Jose Javier Garcia Aranda Jacobo Pérez Lajo Luis Miguel Díaz Vizcaino	First version

Índice

1. ABSTRACT

Communications today are not longer limited to the choice of voice, data or video. The exploding variety of Internet applications offers a great variety of constraints for each one. Current solutions for voice and video services as offered by operators, trigger different mechanisms based on IP in order to assure quality. However, any other kind of services offered by content providers to subscribers (or p2p between subscribers) are excluded from NGN quality assurance mechanisms. This is why we think that Q4S proposal may fit, becoming an answer to trigger and monetize these functions when they are required (on-demand) and dynamically (during any service session timelife).

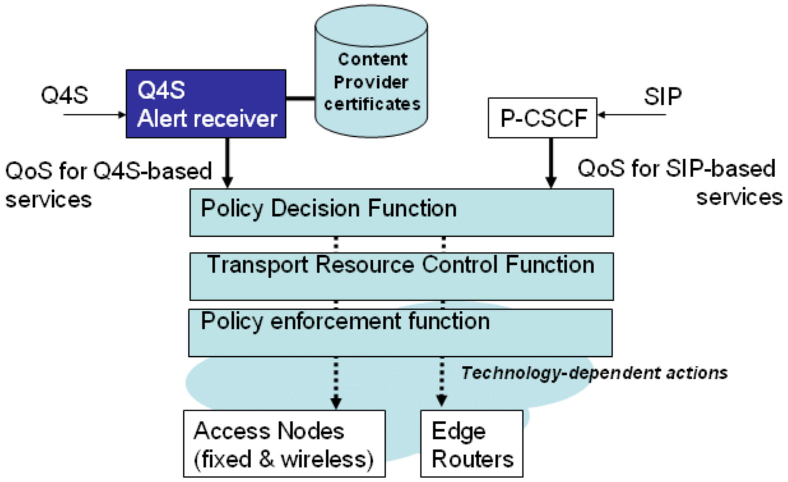

Q4S
Q4S Alert receiver
Content Provider certificates
SIP
P-CSCF
QoS for Q4S-based services
QoS for SIP-based services
Policy Decision Function
Transport Resource Control Function
Policy enforcement function
Technology-dependent actions
Access Nodes (fixed & wireless)
Edge Routers

Figure 1. Q4S may trigger QoS functions for third party content-providers

Nowadays, SIP-based operator services (voice, video) may trigger Quality assurance functions automatically, using NGN architecture available nodes , e.g. NGN-IMS services. For non-SIP services offered by operator or third party content provider, a QoS tracking protocol may be desirable in order to trigger the operator QoS available mechanisms when (and if) they are needed. This allows a more intelligent use of internet resources in terms of bandwidth, packet-loss, latency and jitter, thus generating new revenues for service providers which may support the traffic growth.

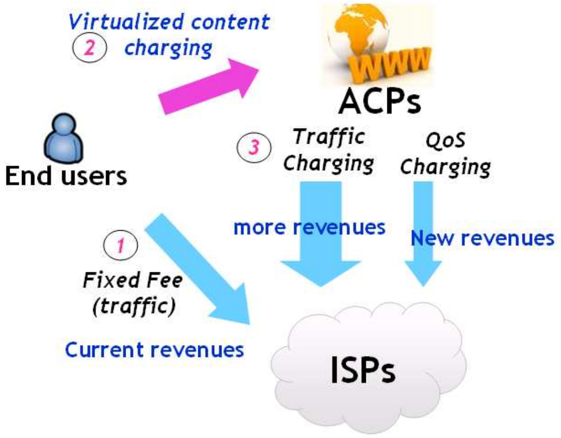


Figure 2. Q4S enables business model, in which users pays for experience to content providers

Q4S is a QoS tracking protocol which alerts ACPs when application requirements are being violated and may be used to trigger ISP's QoS provisioning service. Q4S pretends to achieve a very high efficiency in bandwidth usage (few kilobits/second), avoiding application disturbance.

2. OVERVIEW & PROBLEM STATEMENT

2.1. Quality of User Experiencie: AQoS & NQoS

The **QoE (Quality of User Experience)** involves "Application-based" QoS (**AQoS**) and "Network-based" QoS (**NQoS**). AQoS may include adaptive mechanisms (such as adaptive streaming, management of server side CPU assignment, reduce functionalities, etc) provided by applications to enhance the desired end-to-end performance, while NQoS includes QoS handling features, as provided by the network and networking devices (like access nodes, routers , switches).

The measure of performance from the network's perspective includes four fundamental network parameters: latency, Jitter, bandwidth and packet-loss. Depending on the application needs and these network parameter values, some "application-based QoS mechanisms" may be enough, e.g., compensate for the network latency by encoding video faster.

Each application has different needs (in terms of these four parameters) and also, these needs are different in each direction (forward and reverse traffic). For instance, the traffic of a virtualized videogame from the user to the server (the action controls) is quite low in bandwidth resources but not tolerant to packet-loss because when a user presses the key, the weapons must be launched or the spaceship must move quickly. However, traffic from the server to the user is high in bandwidth consumption but tolerates certain packet-loss.

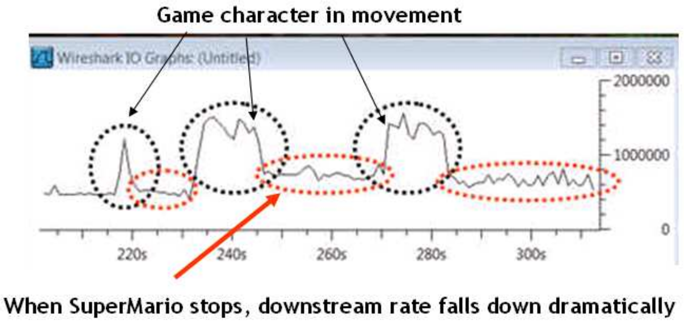


Figure 3. Traffic profile from server to user in a virtualized videogame

Even, within session lifetime application needs may evolve (e.g., it is not the same to navigate through a menu than play the arcade itself). In addition, network conditions may suffer degradation, and some AQoS and NQoS mechanisms should be triggered dynamically to keep the Quality of the user experience constant.

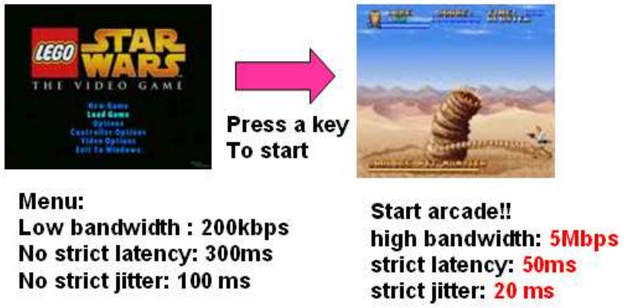


Figure 4. Application needs may evolve dynamically

2.2. QoS levels

The concept of **level** that Q4S is proposing is technology (and network) dependant: e.g., the meaning of each level may be a Diffserv value (DSCP marking) , may become a change on traffic mode at subscriber's access nodes, may result in a change to traffic e2e path, may benefit from RSVP, or any other mechanisms – or combination of several mechanisms -. The definition of each level is out of scope of Q4S. There is only one premise to consider: a higher level value provides a better QoS. The meaning is implementation dependant.

Let us consider the following examples of QoS levels dictionary that different ISPs could expose:

EXAMPLE DICTIONARY A

Level 0: best-effort

Level 1: Application-based QoS , more CPU at Server-side

Level 2: Network based QoS, alert is sent to ISP in order to change DSLAM traffic mode

Level 3: Network based QoS, alert is sent to ISP in order to change DSCP marking at edge Router to Assured Forwarding class 1

Level 4: Network based QoS, alert is sent to ISP in order to change DSCP marking at edge Router to Assured Forwarding class 2

Level 5: Network based QoS, alert is sent to ISP in order to change DSCP marking at edge Router to Assured Forwarding class 3

Level 6: Network based QoS, alert is sent to ISP in order to change DSCP marking at edge Router to Assured Forwarding class 4

Level 7: Network based QoS, alert is sent to ISP in order to change DSCP marking at edge Router to Expedited Forwarding class

EXAMPLE DICTIONARY B

Level 0: best-effort

Level 1: Application-based QoS , more CPU at Server-side

Level 2: Network based QoS, alert is sent to ISP in order to change PCRF policies

Level 3: Network based QoS, alert is sent to ISP in order to change DSCP marking at edge Router to Assured Forwarding class 2

Level 4: Network based QoS, alert is sent to ISP in order to trigger RSVP procedures

EXAMPLE DICTIONARY C

Level 0: best-effort

Level 1: Network based QoS, alert is sent to ISP in order to change DSCP marking at edge Router to Assured Forwarding class 2

Figure 5. Some QoS level dictionaries examples

Q4S measurements are carried out 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 lower level still meet application requirements (with a security margin), then the QoS level for the application flows can be downgraded without risk

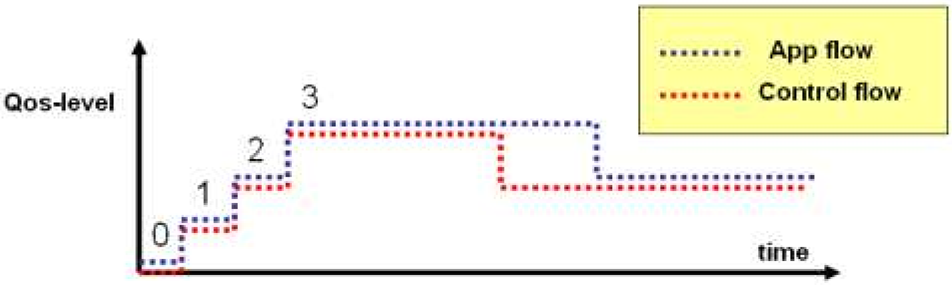


Figure 6. control flow and application flows evolution

2.3. NQoS current approach

It is important to note that although implementation of NQoS is technology dependant and out of scope of the protocol definition, currently there are valid solutions to deal with NQoS. Each one of them may be valid locally or globally and could become one of the Q4S level assignments.

The main NQoS implementation today is the DiffServ one. In order to provide application QoS, the network operator MUST be able to identify that particular application with QoS needs and MUST move those packets into a higher CoS (Class of Service). In non-congested situations that will have no impact at all, but when congestion happens, higher FECs will be safe, although a few more details have to be taken into account:

1. QoS MUST be **engineered**, that is, must be **capped** at a certain bandwidth to avoid lower FECs starvation. This is not only for fairness reasons but also for legal imposition (e.g. in Spain network operators MUST provide at least 10% of bandwidth contract to Internet subscribers).
2. While higher FECs treated better than lower ones under congestion, they may not be safe from intra-FEC congestion (congestion between traffic belonging to the same FEC). Since that higher FEC is limited to a certain bandwidth, **planification** must be put into play to avoid intra-FEC congestion. Intra-FEC congestion will make the service unavailable for each and every subscriber (that is being charged for that) using that specific application (assuming low-to-zero tolerance to packet loss which is the case for new application such as cloud gaming).
3. Thus, only a specific number of subscribers will be allowed to use that FEC, to ensure that in the maximum concurrent time-period the peak bandwidth is not exceeded. This will ensure that there is neither intra-FEC congestion nor lower-FEC starvation. The problem here is to determine the **number of subscribers** that are allowed to use the service.
4. To determine the maximum number of subscribers that are allowed to use the higher FEC, a specific bandwidth must be assigned to each subscriber. This bandwidth must be calculated taking into account the maximum bandwidth consumed by the subscriber and any other parameters that allow to model the service (period of time with that sustained bandwidth needs, maximum concurrency, etc.). Those calculations will lead to a "magic-number" that represents the **bandwidth per subscriber** that makes the CoS safe enough to avoid exceeding the CoS peak at any time (when peak is exceeded, everyone's QoE is destroyed).
5. This number just determines the maximum number of subscriber that are allowed to fulfil a **contract** to use this specific FEC. Once FEC is "full", nobody else must be allowed to join.

This approach has some drawbacks:

1. Since there is no **CaC** function (admission control), there is no way to oversubscribe the FEC with more subscribers, because the chance of creating intra-FEC congestion dramatically

increases. If a CaC function is put into play, there is no chance of intra-FEC congestion since "extra" subscribers will be denied access on connect if such a congestion will happen.

2. Panification of FECs assuming a **constant bandwidth** per subscriber (peak sustained bandwidth adjusted accordingly to optimize resources) but that may not be true in a number of modern applications. For instance, in cloud gaming, there are many different stages (Menus, configuration, equipment, city, battle…) with many different bandwidth requirements. With current approach, resources are not really well optimized if this information is not known in "real-time" and every susbscriber is assumed to be "close" to its peak almost all time contrary to actual needs. Current approaches can't do otherwise as real application requirements are not known (and there is no CaC function).
3. During network **disasters** (fiber cut, box shutdown) congestion is very likely to happen since planification only takes into account some specific most-probable scenarios. During network desasters, intra-congestion happens and the whole service is destroyed.

Q4S approach adds to the current situation a solution to those drawbacks:

1. Q4S carries **information** about application requirements in that specific moment in time (Q4S follows "per-stage" requirements) and about specific parameters that are being offered by the network (based on measurements). That info can be used to fine-tune planification.
2. Application may start as BE instead of on a higher FEC. If no congestion happens, there is no need to do anything. If congestion happens, Q4S (that is measuring if application requirements are fullfilled) will send an **alert** that will reach client, server AND NETWORK OPERATOR. At that moment, the Operator can decide to move that application (subscriber) to the higher FEC or not, depending on available network resources. Thus, Q4S can be use as a **CaC** function to protect higher CoS from intra-congestion and also enables operator to oversubscribe the higher CoS with more subscribers since only the exceeding subscribers will be rejected (and not charged) without affecting any other subscriber QoE at all.
3. Q4S carries **exact requirements per flow (and per stage)**. This way, resources usage optimization can be achieved at levels that estimation cannot.Bandwidth consumption per subscriber does not have to follow such a strict planification, instead, usage trends may be use to drive network growth.
4. Q4S can inform network operator when conditions in the Best-Effort FEC are good enough to carry the application (maybe because BE conditions have become better or because application requirements are lower), allowing to **optimize resources,** since the application will use high FECs only when it really needs to.

Q4S can add the dynamics to DiffServ by allowing network operators to know specific application (flow) requirements in a moment in time (stage) compared with offered parameters by the network itself. This will enable operator to move that flow(s) to higher FEC or not (CaC) and to take them back to BE when everything is fine enough. This way network resources are really optimized without the risk of oversubscription and intra-FEC congestion.

3. Q4S enables NGN Networks for new services

Given the coexistence of multiple QoS technologies and operator domains in the NGN, a key aspect of a universal solution is interworking across different technology and operator domains.

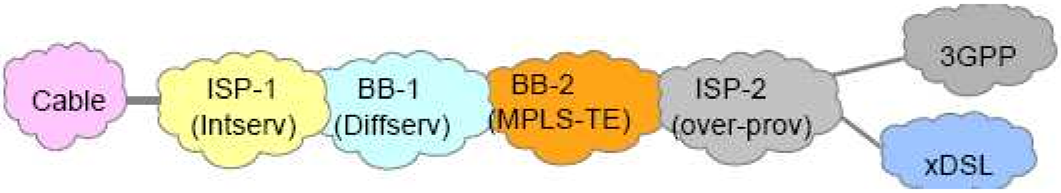


Figure 7. multiple technology and operator domains

NGN networks includes both fixed and wireless domains and allows unlimited choice of access possibilities, e.g. fixed (DSL, cable, GPON, …) , wireless (UMTS, WiFi, WIMAX, LTE, …).

In order to meet the end user's expectations, some kind of quality assurance must be achieved. This quality assurance mechanisms does not exist in the legacy environments. In addition, the real-time interactive applications require end-to-end quality assurance. End-to-end quality assurance relies on an end-to-end QoS tracking procedure, in order to trigger alerts and actions over the network when an application need them (on-demand and dynamically).

The NGN QoS mechanisms are technology dependent. The possible and available actions may be different depending on the underlying access technology (LTE, cable access, ADSL, GPON, etc.) Additionally, the possible actions to be taken on the core may be technology dependant.

In the IMS-based NGN architecture the CSCF (Call Session Control Function) entities perform the trigger of required quality assurance actions in order to provide a suitable operator voice/video service.

Anything out of SIP scope and operator service, cannot take benefit from this approach. The model for operator voice service is known and resources can be assured dynamically based on a Diffserv schema. Four basic NGN QoS-aware services are currently considered but not exposed to third party content providers, loosing a potential source of revenues.

As mentioned before, each application has different requirements. Not all data applications have the same constraints, and also the constraints in each direction (forward & reverse) may be different.

Based on these assumptions, Data applications could be split into several groups, defining a set of possible actions to take over the network.

The "de facto" standard for most QoS IP NGN solution is Diffserv. The principal drawback of this technology is the lack of QoS tracking & Admission control functions. Therefore complementary QoS tracking mechanisms to trigger Admision and provisioning control functions should be provided in order to open the QoS Network assurance capabilities to third party Application Content providers

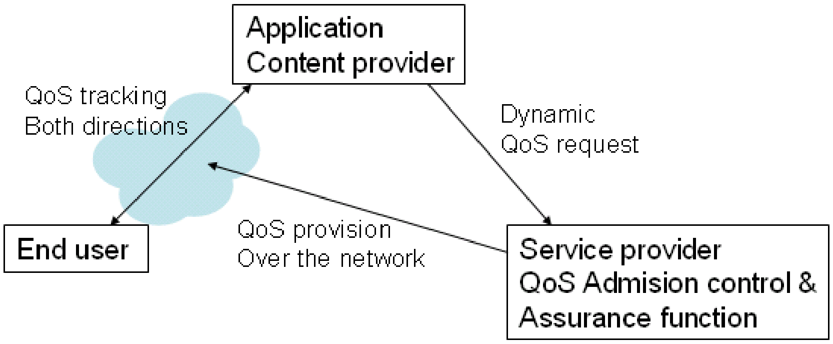


Figure 8. Operator expose Dynamic QoS as a service

The protocol to achieve the QoS tracking may be proprietary, made by each content provider. However a uniform, standarized solution has some benefits:

- Optimal in terms of availability (any content provider could use it) and interoperability (any browser may support it, allowing browser-to-browser scenarios)
- Provides a uniform way to request QoS services to any ISP. A standard protocol may trigger Network-based QoS requests towards ISP in a well-known way (in terms of when, how many times, and why). The same application is portable from one ISP to another.
- Provides a framework to reach a multi-ISP scenario in which an Administrative Owner wants to establish end-to-end QoS guarantees between to end points. The AO may be one of the ISPs or it may be a separate service provider (aligned with IPSphere tmforum initiative).

Q4S Protocol measures the perceived fundamental Network QoS parameters at any time of the session lifetime in order to raise an alert the application's needed constraints are being violated.

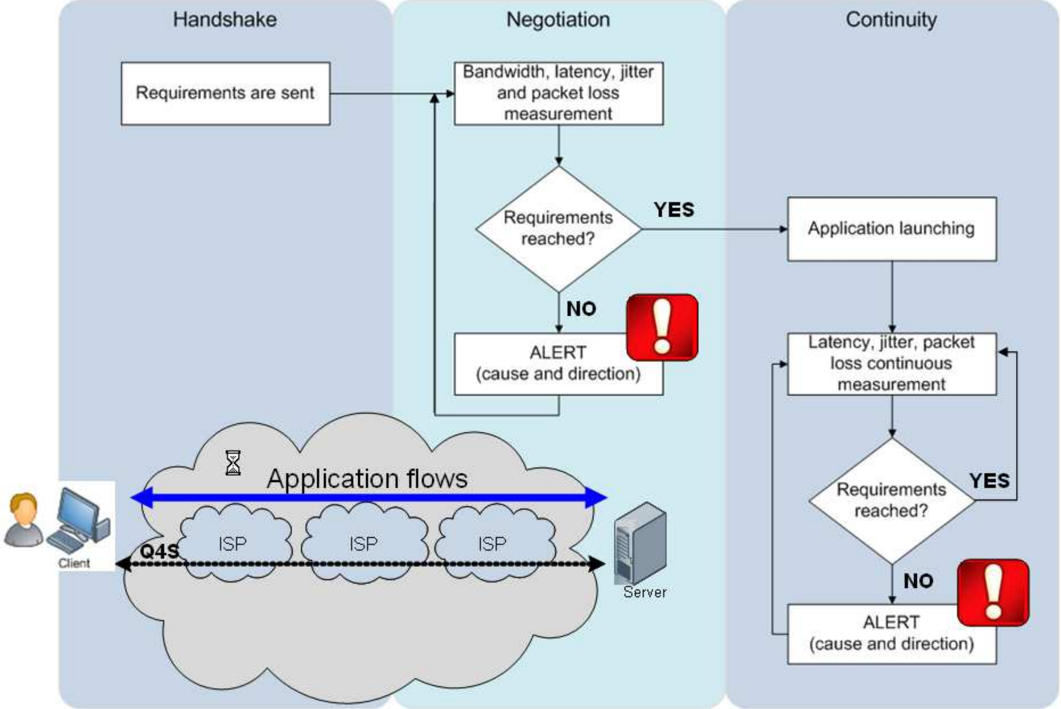


Figure 9. Q4S flowchart

The Q4S ALERTS include the following information for ISPs:

- Application constraints in both directions (forward and reverse) in terms of bandwidth, packet-loss, latency and jitter
- Measurement values in both directions
- Violated parameter and direction
- Current QoS level (the meaning of level is ISP dependant)
- Digital signature for admission control and charging purposes (ISP may charge Content provider based on a digital certificate)

Q4S is technology agnostic. ALERTs are raised **but triggering actions on the network is a decision of the ISP/Network operator**. The set of possible actions are technology and ISP dependant. It is out of scope of Q4S and is part of the network deployment. Q4S defines a set of QoS-levels, but the meaning of each level depends on what each ISP wants to expose as QoS services.

In addition, depending on the underlaying technology, the related actions over the network for each level may be different. Actions like invoke QMM in GPRS scenarios or PCRF in LTE or change profiles at DSLAM using NASS functions are ISP & technology dependant, but all of them can be triggered from a single event : the Q4S ALERTS.

In a NGN-IMS network the policy decision function is a standalone entity responsible for implementing the service-based local policy (SBLP) control for the IMS operator. These decisions are based on session and media-related information that is forwarded by the P-CSCF.

The overall picture may be complemented with a Q4S alert receiver in charge of trigger the same Network-Based QoS mechanisms as P-CSCF implements.

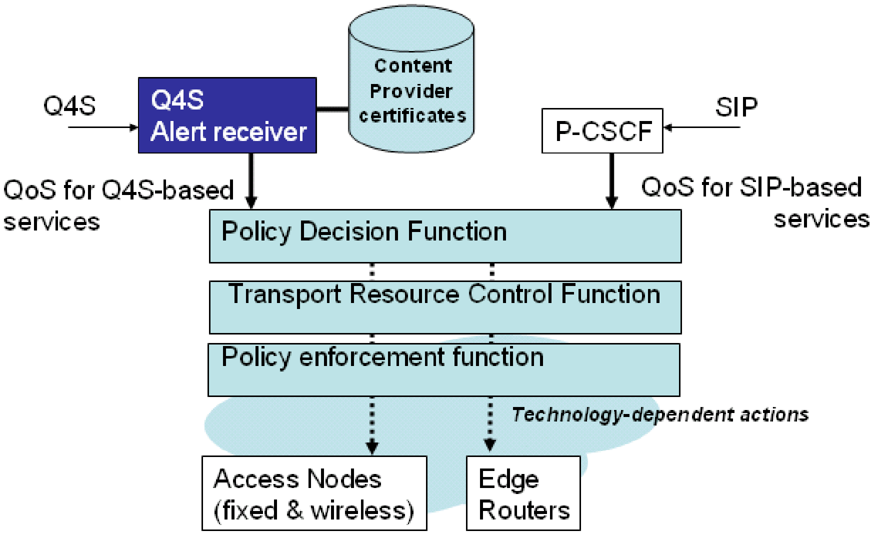
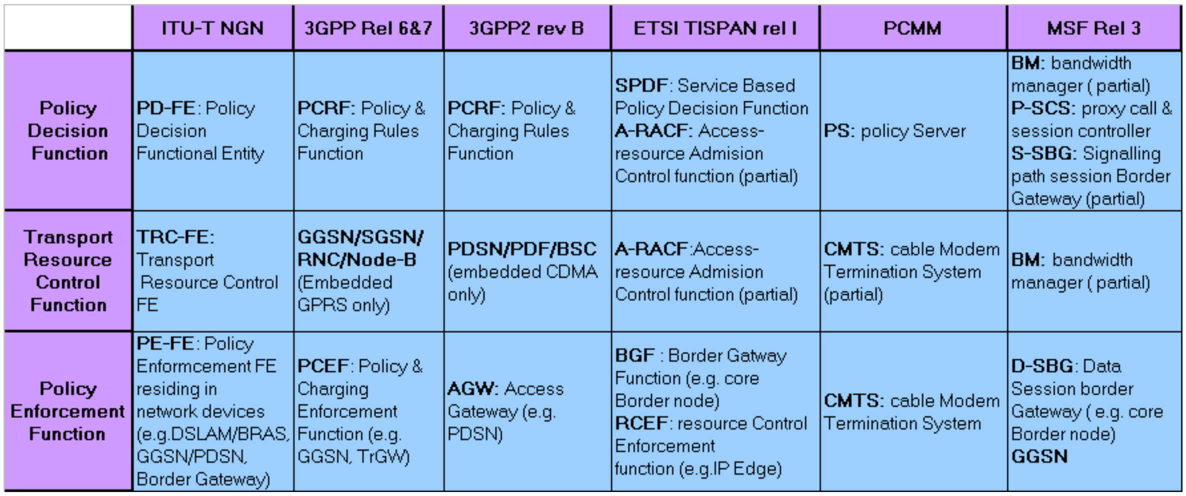

Q4S
Q4S Alert receiver
Content Provider certificates
P-CSCF
SIP
QoS for Q4S-based services
QoS for SIP-based services
Policy Decision Function
Transport Resource Control Function
Policy enforcement function
Technology-dependent actions
Access Nodes (fixed & wireless)
Edge Routers

Figure 10. Q4S may trigger QoS functions for third party content-providers

The following table represents different NGN networks and their nodes in charge of each functionality.



	ITU-T NGN	3GPP Rel 6&7	3GPP2 rev B	ETSI TISPAN rel I	PCMM	MSF Rel 3
Policy Decision Function	**PD-FE**: Policy Decision Functional Entity	**PCRF**: Policy & Charging Rules Function	**PCRF**: Policy & Charging Rules Function	**SPDF**: Service Based Policy Decision Function **A-RACF**: Access-resource Admision Control function (partial)	**PS**: policy Server	**BM**: bandwidth manager (partial) **P-SCS**: proxy call & session controller **S-SBG**: Signalling path session Border Gateway (partial)
Transport Resource Control Function	**TRC-FE**: Transport Resource Control FE	**GGSN/SGSN/ RNC/Node-B** (Embedded GPRS only)	**PDSN/PDF/BSC** (embedded CDMA only)	**A-RACF**:Access-resource Admision Control function (partial)	**CMTS**: cable Modem Termination System (partial)	**BM**: bandwidth manager (partial)
Policy Enforcement Function	**PE-FE**: Policy Enformcement FE residing in network devices (e.g.DSLAM/BRAS, GGSN/PDSN, Border Gateway)	**PCEF**: Policy & Charging Enforcement Function (e.g. GGSN, TrGW)	**AGW**: Access Gateway (e.g. PDSN)	**BGF** : Border Gatway Function (e.g. core Border node) **RCEF**: resource Control Enforcement function (e.g.IP Edge)	**CMTS**: cable Modem Termination System	**D-SBG**: Data Session border Gateway (e.g. core Border node) **GGSN**

4. Q4S BUSINESS MODEL APROACH

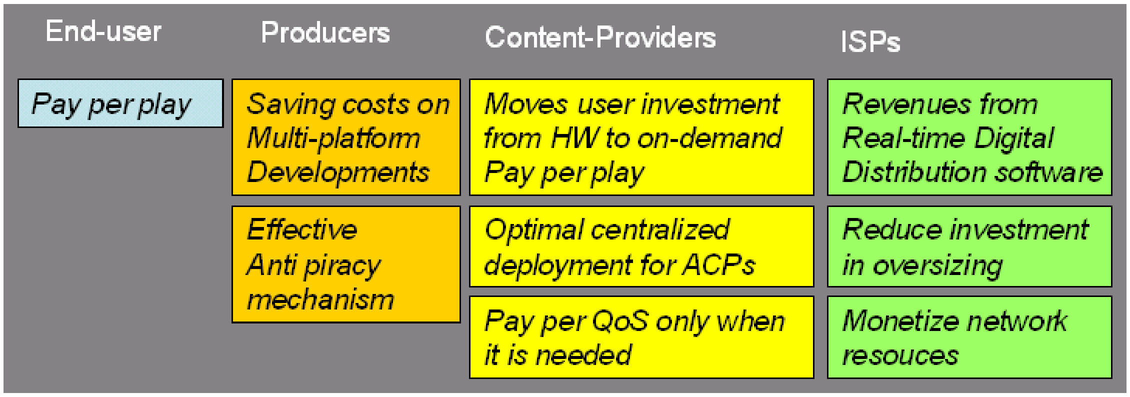
The number of potential applications which may take benefit from Q4S is huge. Examples of use cases enabled by Q4S are:

- Virtualized videogames
- Security use cases based on virtualization (banking applications)
- Premium VoIP and Conferencing over Internet
- E-Auctions
- Computer Supported Cooperative Work (CSCW)
- Financial Businesses
- Web TV HD portals
- …

Considering the video gaming industry is a quite challenging example because Q4S can be considered more than a digital distribution method: with Q4S not only the game is distributed, but also the hardware platform (console or PC)

The gaming industry moves around 50 billion Euros per year, thereof 35 billion are games and the rest, (15%) hardware consoles. In the value chain of gaming industry, the distribution layer takes around 10% and this represents the maximum size of the videogame market using Q4S distribution system, plus savings in consoles (15% of total), and anti-piracy benefits.

Detailed studies reveal a high ROI of 7 times for ISPs. These revenues for ISP are enough to support the traffic growth with a network investment of about 12% of the revenues.



End-user	Producers	Content-Providers	ISPs
Pay per play	*Saving costs on Multi-platform Developments*	*Moves user investment from HW to on-demand Pay per play*	*Revenues from Real-time Digital Distribution software*
	Effective Anti piracy mechanism	*Optimal centralized deployment for ACPs*	*Reduce investment in oversizing*
		Pay per QoS only when it is needed	*Monetize network resouces*

Figure 11. benefits of virtualization based on dynamic QoS

Following figure represent money flows between different entities:

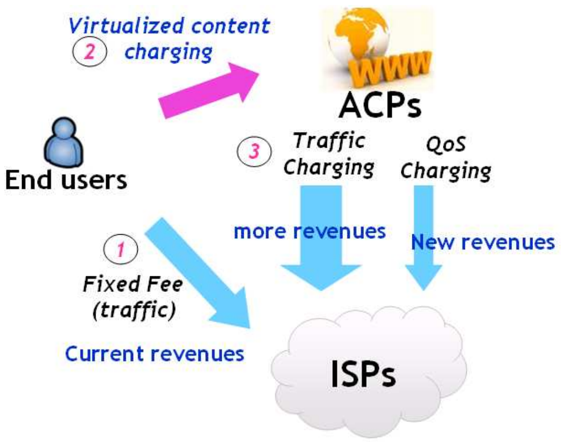


Figure 12. Q4S enables business model, in which users pays for experience to content providers

We may be also allowing for a broader market share:

- Users who already have one of the consoles, may want to try online games which are only available to other hardware platforms.
- Users who do not have any console, may become casual service users if they can play with a small investment in hardware . They could play any game, any console having none at home.
- The same game may be sold in different "packs": complete in DVD, partial DVD + certain online stages, only online, game online preview before official release, …

5. REFERENCES

Reference	Document name	Reference Number
	Bell labs technical Journal "Economic and Technical Propositions for Inter-Domain Services"	DOI: 10.1002/bltj
	QoS aspects of IPSphere	
	Internet Draft: "draft-aranda-dispatch-q4s-00"	
	Q4S charter proposal	
	ETSI ES 282 004 v3.4.1 NGN functional architecture; Network Attachment Sub-System (NASS)	
	3GPP standards "Policy and charging control architecture"	
	ITU-T standard "Resource and admission control functions in next generation networks"	
	PacketCable "Quality of Service Architecture"	
	ETSI " Resource and admission control subsystem (RACS)"	