



Information Society
Technologies



Title: Deliverable D5.2 Implementation and documentation of the Advanced Services APIs and improvements in the applications for PLC networks	Document Version: 1.2
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------

Project Number: IST-2001-37613	Project Acronym: 6POWER	Project Title: IPv6, QoS & Power Line Integration
------------------------------------------	-----------------------------------	-------------------------------------------------------------

Contractual Delivery Date: 31/07/2003	Actual Delivery Date: 15/09/2003	Deliverable Type* - Security**: R – PU
-------------------------------------------------	--------------------------------------------	--------------------------------------------------

* Type: P - Prototype, R - Report, D - Demonstrator, O - Other

** Security Class: PU - Public, PP - Restricted to other programme participants (including the Commission), RE - Restricted to a group defined by the consortium (including the Commission), CO - Confidential, only for members of the consortium (including the Commission)

Responsible and Editor/Author: Marco Stahlmann	Organization: MCL	Contributing WP: WP5
----------------------------------------------------------	-----------------------------	--------------------------------

Authors (organizations): Emilio García (ASSA), Pedro M. Ruiz (ASSA), Juan Antonio Sánchez (ASSA), Jordi Palet (Consulintel), Antonio Gómez-Skarmeta (UMU), Daniel Martínez (UMU).

Abstract: The document describes the interfaces of the applications for the usage of advanced services provided by the QoS supporting PLC network and lists the improvements for existing applications resulting by the usage of PLC networks.

Keywords: Adaptive Applications, IPv6, PLC, QoS.

Revision History

The following table describes the main changes done in the document since created.

Revision	Date	Description	Author (Organization)
v1.0	28/07/2003	Document creation	Marco Stahlmann (MCL)
v1.1	06/09/2003	Document review	Jordi Palet (Consulintel)
v1.2	15/09/2003	Final review	Jordi Palet (consulintel)

Executive Summary

The document describes the interfaces of the applications for the usage of advanced services provided by the QoS supporting PLC network and lists the improvements for existing applications resulting by the usage of PLC networks.

The document lists the advanced applications with their currently defined QoS profiles and describes the procedure of call setups at the example of the videophone for SIP based applications and with Gnomemeeting for H.323 based applications.

Table of Contents

1.	<i>Advanced Services for PLC</i>	6
1.1	Guaranteed Bandwidth	6
1.2	Guaranteed Latency	6
1.3	Monitoring the Network Parameters	6
2.	<i>Advanced Applications for PLC Networks</i>	7
2.1	Videophone	7
2.2	VoIP	7
2.2.1	SIP Based Applications.....	7
2.2.2	H.323 Based Applications.....	7
2.3	Legacy Applications	8
3.	<i>QoS Mapping</i>	9
3.1	Description of the Parts	9
3.1.1	QoS Broker.....	9
3.1.2	Client Using SIP Signaling	10
3.2	General Interaction	11
3.2.1	Interaction of Applications Using H.323 for Signaling	13
4.	<i>Summary and Conclusions</i>	15
5.	<i>References</i>	16

Table of Figures

Figure 2-1:	QoS Predefined Profiles for Videophone	7
Figure 2-2:	QoS Predefined Profiles for SIPphone	7
Figure 2-3:	QoS Predefined Profiles for H323phone	8
Figure 3-1:	Usage of Parameter b	10
Figure 3-2:	Videophone GUI	11
Figure 3-3:	QoS Predefined Profiles.....	11
Figure 3-4:	QoS Broker / Videophone Interaction.....	12
Figure 3-5:	QoS Broker / Gnomemeeting Interaction	14

1. ADVANCED SERVICES FOR PLC

1.1 Guaranteed Bandwidth

CPEs can be assigned a minimum bandwidth, whereas no fixed rate can be reserved but only a percentage of the currently available bandwidth. This percentage is actually a portion of time and a portion of the spectrum frequency, which is assigned to the CPE. When some CPEs don't use their reserved bandwidth all other CPEs can take profit of it. This percentage will actually yield a variable bandwidth depending on the quality of the power line seen by the CPE in both paths (Upstream and Downstream). If more bandwidth is desired for a CPE, the network operator has to increase the CPE's physical BW percentage. Bandwidth percentage can be individually set for the Upstream and Downstream.

1.2 Guaranteed Latency

CPEs can be assigned a maximum latency in the Upstream. In the Downstream, the maximum latency depends on the number of active users and at this moment is limited to 50 milliseconds approx. Packets are guaranteed to see a latency lower than the maximum set for the CPE provided the data packets find the driver queues empty. Latency is assured if the data flow throughput doesn't exceed the QoS contract. This throughput is the actual data rate achievable with the present line quality and bandwidth percentage. We must also distinguish between access latency and active latency. Access latency is the latency seen when the CPE's state is IDLE. This is greater than the usual latency (active latency) which is the latency seen by ACTIVE CPEs. Access latency can also be limited with the CPE_ACCESS_TIME but it is a trade-off between initial access time and overall bandwidth efficiency since more slots for polls have to be inserted and thus reducing the overall bandwidth.

1.3 Monitoring the Network Parameters

In the PLC network it is not possible to reserve a fixed transmission rate but only a percentage of the currently available bandwidth, and so the QoS broker has to constantly check the network status, the link quality and with it, the overall available bandwidth is decreased. If it detects a decrease in link quality it then has to verify if the requested bandwidth requirements for all CPEs are still met and reset the reserved bandwidth in all observed PLC components to the new values. In some cases when the link quality drops so drastic so that not all bandwidth requirements can be achieved the QoS broker has to drop calls according to some specified rules or it has to signal the applications to start a bandwidth renegotiation with applications supporting this, otherwise the call has to be dropped.

2. ADVANCED APPLICATIONS FOR PLC NETWORKS

2.1 Videophone

The Videophone application is a video telephone tool derived from the Isabel CSCW application. It supports various audio and video codecs and is currently capable of supporting three different levels of quality, which are listed in the table below.

Profile	Audio Codec	Video				BW (kbps)	SDP Field
		Codec	Size	Fps	Quality		
LOW	GSM	MJPEG	SQCIF	15	66	350	b=CT:350
MEDIUM	G.711a-law	MJPEG	QCIF	16	70	700	b=CT:700
HIGH	PCM	MJPEG	CIF	16	60	1800	b=CT:1800

Figure 2-1: QoS Predefined Profiles for Videophone

2.2 VoIP

There are currently two widespread standards for VoIP signaling namely SIP and H.323, which will also be supported within the PLC network so that the customers have the less restraints for the use of VoIP.

2.2.1 SIP Based Applications

The SIP VoIP phone developed for 6POWER is based on the Vovida Open Communication Application Library (Vocal) project, which provides tools for VoIP development. The supported quality levels will be the ones listed in the table below.

Profile	Audio Codec	BW (kbps)	SDP Field
LOW	G.723	28	b=CT:28
MEDIUM	G.723	45	b=CT:45
HIGH	G.711-uLaw	70	b=CT:70

Figure 2-2: QoS Predefined Profiles for SIPphone

2.2.2 H.323 Based Applications

The H.323 VoIP phone developed within the 6POWER project is based on the GPL licensed Gnomemeeting tool enhanced for the support of QoS. The supported quality levels will be the ones listed in the table below.

Profile	Audio Codec	BW (kbps)	SDP Field
LOW	LPC-10	5	b=CT:5
MEDIUM	GSM-06.10	20	b=CT:20
HIGH	G.711-uLaw	70	b=CT:70

Figure 2-3: QoS Predefined Profiles for H323phone

2.3 Legacy Applications

For legacy applications, which do not provide QoS support but would also benefit from QoS there are two ways to support QoS:

- Applications which can be run with startup parameters to set maximum bandwidth usage. A wrapper application will take care of the signaling with the QoS broker and start the application with the corresponding parameters.
- Applications using a fixed port number. Can be filtered inside the equipment and given a predefined global QoS.

3. QoS MAPPING

Whatever signaling between the applications and the broker is chosen, it effectively establishes a bind from some IP traffic to some specific QoS treatment, that is, a QoS mapping. But if a standard protocol were used for doing the signaling, the QoS broker would be generic; it would work with any application that uses that signaling protocol. Moreover, those applications would have to undertake minor changes to obtain full-benefit from the broker capabilities.

The solution for 6POWER will use the Session Initiation Protocol (SIP) to do the signaling. SIP is a standard protocol, thus using it makes the broker be generic: It may manage QoS requirements from any final application that uses SIP with the simple extensions we are proposing. The selected application will be Videophone, which has been enhanced to take advantage of the QoS capabilities from the embedded SIP support.

Using SIP, the QoS broker can be considered a layer-3 entity capable of taking QoS requests from generic applications and managing them hiding layer-2 details, so that final users only have to deal with their final applications.

3.1 Description of the Parts

3.1.1 QoS Broker

The QoS broker has a SIP module that implements a simple API for SIP, together with a set of functions to offer SIP services to the other modules. A structure is defined to hold a SIP message, along with functions to read a message from a data buffer into such a structure and vice versa. The functions available to work on SIP messages include message parsing, header access and modification, QoS reading (from the SDP body), SIP call cancellation and message forwarding.

The main high-level function of the SIP module is to forward SIP messages. The necessary steps for doing this depend on whether the message being forwarded is a SIP request or a SIP response.

When forwarding a SIP request, the steps are:

- Insert an appropriate “Via” header into the message.
- Get the host name / address of the next hop from the request line.
- Send the message to the next hop.

If the message is a SIP response, the steps are:

- Remove the topmost “Via” header.
- If there are no more “Via” headers left, this message is for this host and must not be forwarded.
- Otherwise, get the host name / address of the next hop from the next “Via” header.
- Send the message to the next hop.

Another important function in the SIP module is to cancel a SIP call. This is needed when the QoS requirements stated in a SIP INVITE are not allowed by the policies. In this case, the original SIP INVITE is handed to a function in the SIP module, which does the following:

- Changes the request line from “INVITE ...” to “SIP/2.0 603 Declined”.
- Adds a tag to the “To” header.
- Drops the SDP body.
- Inserts a “Via” header.
- Forwards the message.

3.1.2 Client Using SIP Signaling

The client can be set-up to connect directly to any other peer, or send all the session establishment messages through a gateway. In order to take advantage of the QoS benefits mentioned before, the later will be selected as operation mode. This way, the QoS broker embedded in the gateway will intercept and interpret all the SIP messages used to negotiate the conference parameters, and thus the appropriate actions will be taken.

To make this possible, SIP messages generated by the client (or rather, their SDP contents) must be adequately defined. The client uses and extends the SDP definition, by using the *b* field (described in Section 3.1.1) to signal the bandwidth that is to be consumed in the session being requested. An example of the usage of this parameter is depicted in Figure 3-1.

```
v=0
o=flateric 1055752557 1055752557 IN IP4 192.168.0.78
s=Videofono
i=Test session
u=http://www.agora-2000.com
b=CT:700
(...media description...)
```

Figure 3-1: Usage of Parameter *b*

The parameter is fixed according to user preferences, expressed using the Graphical User Interface of e.g. the Videophone (see Figure 3-2).

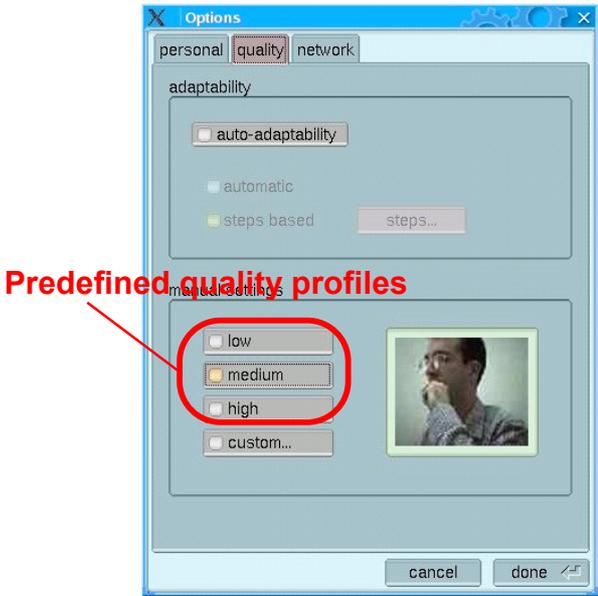


Figure 3-2: Videophone GUI

The procedure is as follows: Before initiating the call, the user selects the desired level of quality (*low*, *medium* or *high*). Each pre-defined level has a correspondence with some audio/video settings, and will consume a known bandwidth (approximately), which will be inserted in the SDP (using *b* field). When performing the invitation to start a conference, the appropriate SIP INVITE message, containing this SDP, is sent to the SIP Gateway, which will use the *b* field as described in Section 3.1.1.

Figure 3-3 details the characteristics of the sample set of quality levels at the example of Videophone, although these are configurable:

Profile	Audio Codec	Video				BW (kbps)	SDP Field
		Codec	Size	Fps	Quality		
LOW	GSM	MJPEG	SQCIF	15	66	350	b=CT:350
MEDIUM	G.711a-law	MJPEG	QCIF	16	70	700	b=CT:700
HIGH	PCM	MJPEG	CIF	16	60	1800	b=CT:1800

Figure 3-3: QoS Predefined Profiles

The *b* field is also present in the SDP sent by the called in its SIP OK message, so the backwards path is also considered in this approach.

3.2 General Interaction

The QoS Broker and the client usually interact in a peer-to-peer fashion. Figure 3-4 depicts a sample scenario with a Videophone as a client where all the components described are present, as well as some network elements that form the necessary PLC network.

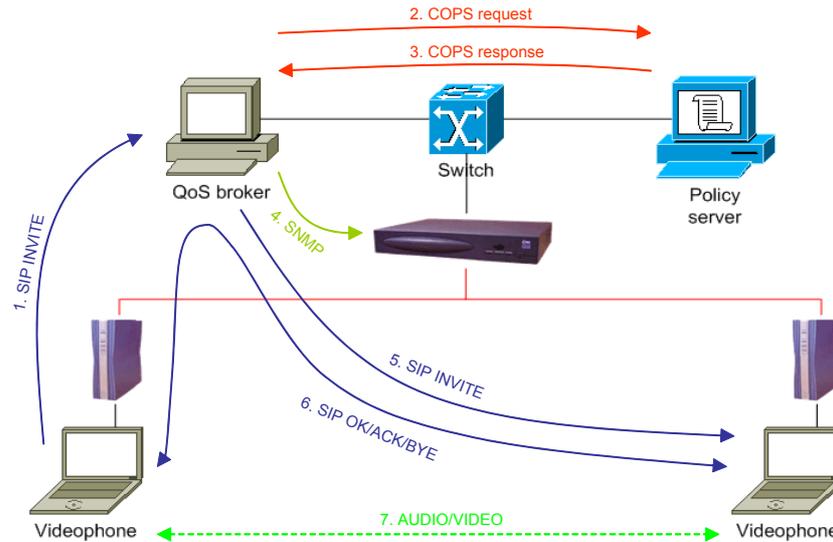


Figure 3-4: QoS Broker / Videophone Interaction

When a Videophone intends to start a conference, it prepares a SIP INVITE message that will contain the SDP defining its media capabilities, as described in Section 3.1.1. The QoS Broker is to be configured in Videophone as the element that will proxy all SIP messages, so it can intercept, process and forward them, transparently to both peers.

The main QoS Broker code listens for SIP messages and acts accordingly, that is, serves as a SIP proxy while managing QoS requests for the SIP calls. The QoS Broker is capable of managing multiple simultaneous SIP calls. To do this, it makes use of a call table, which stores the following information for each call:

- Original SIP INVITE message
- COPS handle for this call
- COPS signalled data

When the SIP INVITE message is sent, it is received **(1)** by the Broker – meaning that a new SIP call is to be established – a new entry is created for it in the call table. QoS requirements are extracted from the SIP message, which is then copied to this entry in the call table. The COPS handle (a COPS transaction identifier) is calculated from the SIP message, the signalled data is built and then a COPS request is issued to the policy server **(2)**. The response to this request will be an asynchronous event, so after doing this, the broker goes back to listening for SIP messages.

It is expected that the QoS will be stated in an SDP (Session Description Protocol) body within the SIP INVITE message. The SDP header that carries this information has the form “b=<bandwidth modifier>:<bandwidth value>”. The QoS Broker recognizes three custom bandwidth modifiers: “X-UBR” (unspecified bit rate), “X-VBR” (variable bit rate) and “X-CBR” (constant bit rate), although there is no real support for these traffic classes at the PLC layer yet (this is why Videophone always uses the standard modifier “CT”). The other field is the bandwidth value in Kbps (kilobits per second). If an SDP body is not found within the SIP message, or if the “b=”

header is not found in the SDP body, the QoS requirement is assumed to be UBR/0, meaning that no special QoS treatment is required.

When the policy server decision is received **(3)**, the QoS Broker must check whether the QoS request from the SIP node was accepted or rejected. In the first case, the QoS requirements are extracted from the call table, and set on the PLC nodes via SNMP **(4)**. This done, the original SIP INVITE message (which was also saved in the call table entry) is forwarded **(5)**. Otherwise, a canceling message, namely a “603 Declined”, is sent back to the caller and the entry in the call table is freed.

Any type of SIP message other than INVITE or BYE is forwarded without taking further action **(6)**. This includes the ones that are necessary to initiate the audio/video transmission **(7)**, which will use the resources that have been reserved in the network as a result of the previous negotiation.

Eventually, one of the peers will end the conference, issuing a BYE message to the other peer, through the gateway. Its side of the conversation is immediately dropped.

When the QoS Broker receives a SIP BYE message, it is time to free the resources that were allocated when the correspondent INVITE was processed. To do this, the COPS handle is calculated from the BYE message; since it will be the same handle that was calculated from the INVITE message, it is used to search the correct entry in the call table. The QoS requirements are taken again from this entry in the call table, and then the QoS configuration in the PLC nodes is updated. After this, the BYE message is forwarded and the entry in the call table is freed. This message reaches then the second peer, and the conference ends completely.

3.2.1 Interaction of Applications Using H.323 for Signaling

The QoS Broker and the client can in this case not interact in a peer-to-peer fashion since the QoS broker only understands SIP. Therefore the previously described scenario for the videophone has to be extended for H.323 signaling.

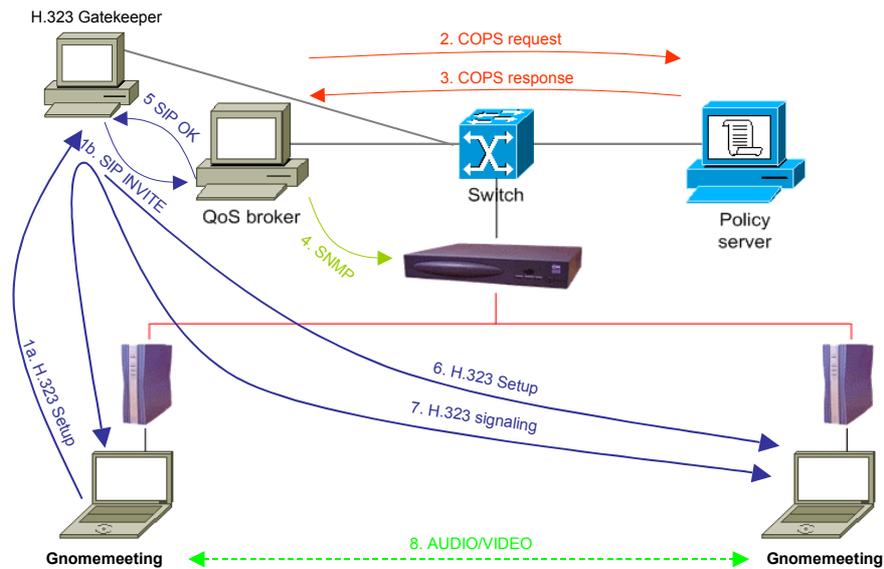


Figure 3-5: QoS Broker / Gnomemeeting Interaction

As with SIP signaling it is possible within H.323 to route all signaling messages via a Proxy, which in this case is called a Gatekeeper. When a client wants to initiate an audio session the gatekeeper intercepts the call setup from the client and extracts the submitted QoS parameters and adapts them to the format supported by the QoS broker. With this information it then contacts the QoS broker and the procedure will be processed as described above. At the end the QoS broker will tell the gatekeeper if the call is accepted and the gatekeeper will resume the paused connection setup.

4. SUMMARY AND CONCLUSIONS

Through the introduction of advanced applications interacting with the QoS broker and the policy server it is now possible to reserve a certain quality for each CPE, which can be maintained even in the presence of applications that do not provide support of QoS. Even legacy applications can benefit from these services if a prioritization by port number or startup parameters is possible.

By extending the currently available applications to adaptive applications the bandwidth broker will be able to support as many users as the network can support as it can allow more concurrent users since then it will be possible to renegotiate communication parameters so that all users will get their requested QoS. Or if the network quality should decrease the users will at least be able to continue their service with a slight deterioration of their perceived quality.

5. REFERENCES

Gnomemeeting: <http://www.gnomemeeting.org>

Voida: <http://www.voida.org>