A Mechanism to Enable Differentiated Services QoS in HIPERLAN/2

Konstantinos Oikonomou[1], Ioannis Tenidis[1], and Ioannis Stavrakakis[2]

[1]Development Programmes Department, INTRACOM S.A.,

{okon@intracom.gr, iten@intracom.gr}

[2]Department of Informatics, University of Athens,

{istavrak@di.uoa.gr}

ABSTRACT

HIPERLAN (HIgh PErformance Radio Local Area Network) type 2 platform (H/2) is a Wireless LAN and it considered as an in-door network where users would demand improved services in terms of data rates and quality. DiffServ (Differentiated Services) is a currently evolving approach to providing QoS over IP networks. Therefore, DiffServ QoS parameters of the IP layer need to be mapped to the H/2 specific QoS parameters. The DLC layer of H/2 has special features that can be used to support QoS. An effective *mapping* between DiffServ QoS and H/2 parameters is expected to increase the *performance of the system* while maintaining the QoS constraints. Consequently, a mechanism is needed in order to allow each application to request and reserve certain resources over H/2. So far there is no standard mechanism and the main focus of this paper is to propose one.

I. Introduction

H/2 is an ETSI wireless LAN standard designed to enable wireless access to different networks such as IP, UMTS (Universal Mobile Telecommunications System) and ATM.

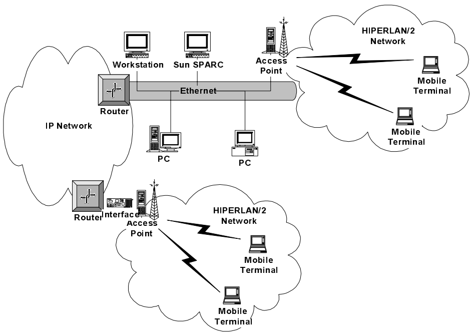

Figure 1. Possible Locations of a H/2 system.

H/2 will allow user mobility inside a building maintaining certain level of Quality of Service. Figure 1 shows possible locations of the H/2 system in an IP network.

So far there is no specification regarding the adaptation of IP to H/2. The current work explores this adaptation and furthermore provides a mechanism for co-operation with the DiffServ QoS architecture.

H/2 DLC should be able to provide different service to traffic associated with different applications. Standard H/2 DLC does not support any special QoS guarantees and it is left to the individual designer to support enhanced QoS features. Two of its functions are (i) to accept/reject a new connection based on the connection's needs and the available resources (this is similar to what is called Call Admission Control – CAC) and (ii) to schedule the data transmission in an efficient manner.

Scheduling can be efficient if and only if the DLC can properly interpret the specific QoS constraints of different flows associated with the different connections. Therefore, the QoS parameters at the IP layer should be properly mapped to the DLC parameters. This is the main role of the *Convergence Layer*: mapping of the DiffServ QoS parameters to the H/2 specific DLC ones.

In section II the basic features of the DiffServ model, as it is currently proposed, are presented. In section III an overview of the DLC architecture is presented. The main focus is to identify and explain the DLC QoS features that are important to the mapping. The mechanism to provide DiffServ QoS is presented in section IV. This section is the core of this paper and the information presented there is our proposal for H/2 with DiffServ QoS enabled.

II. Differentiated Services

Differentiated Services, or DiffServ, is based on flow aggregation; i.e. reservations are made for a set of related flows (e.g. for all flows between subnets). Furthermore, these reservations are rather static since no dynamic reservations for a single flow are possible. Therefore, one reservation may exist for several possible consecutive flows. [3] and [4] is the primary source for this section.

IP packets are marked differently by the user (either at an end system or a router) or by the service provider. This marking is done in the ToS (Type of Service) byte of the IPv4 packet header or at the Traffic Class field of the IPv6 packet header. The router reserves resources, in particular bandwidth, for different priority classes. This concept enables a service provider to offer different classes of QoS at different costs to its customer.

The main difference between DiffServ and the classical approaches used to provide QoS over networks (ATM or even Integrated Services for IP networks) is that the former is based on the *Per-Hop Behavior* in each router while the latter are connection oriented and reserve resources in each particular router/switch. Different IP packets can be forwarded through different routes to the destination point (DiffServ is routing-independent). There is no need in DiffServ to define a certain route which IP packets originating from a particular source and having a particular priority, should follow. DiffServ depends on the ability of each individual router to forward packets according to their priorities, as they are specified in the ToS byte of the IP header.

Three kinds of service can be distinguished:

- **Premium Service** is destined for applications that have delay constraints. These applications can typically tolerate only small delays and delay jitter. The DiffServ model guarantees that every premium packet will be forwarded as soon as possible in the intermediate routers, though losses might occur.

- **Assured Service** is destined for applications that have loss constraints. Loss at the network layer does not necessarily imply loss at the transport layer. For this reason, loss at the IP layer may imply extra delay if the network layer protocol is TCP, due to retransmissions. On the contrary, if the network layer protocol is UDP it implies packet loss for the application.

- **Best Effort Service** is destined for the rest of the applications. These applications are considered to tolerate packet losses or time delays (e.g. file transfer or pre-recorded video).

An important element of the DiffServ QoS architecture is the so-called *Bandwidth Manager* (BM). BM is a special agent that supervises a separate domain of the network. BMs communicate with each other in order to request/reserve resources. In our approach we will use an entity called LNC (Layer Network Coordinator) [5] instead of BM. An LNC includes more functionality than a BM and is able to handle and network management related issues.

III. H/2 DLC Architecture

The H/2 platform consists on the following layers (see Figure 2) [1], [2]:

- *Physical layer*: An OFDM-based Physical Layer capable in providing up to 54 Mbps transfer rate for both the uplink and downlink.

- *DLC layer*: Its main function is to transfer the data to the physical layer in an efficient manner. This can be achieved by prioritizing the use of the shared medium. This layer uses retransmission mechanisms, priorities for different flows of data, error control mechanisms, maintenance of the wireless link (status of each separate connection, accept/reject new calls for resources) etc.

- *Convergence layer (CL)*: We are interested in IPv4 as the higher layer (IPv6 is supported as well). Consequently, a conversion of the IP packets into DLC packets has to be performed. CL is divided into two sublayers: the Service Specific Convergence Sublayer (which is attached to the higher layers) and the Common Part (which is attached to the DLC layer). The former performs the QoS mapping for each particular technology (e.g. IP, ATM, and Ethernet) while the latter takes care of segmentation and re-assembly among other functionalities.

- *Higher layers*: The higher layers can be IP, ATM or even Ethernet. For each particular case there exists a certain Service Specific Convergence Sublayer. Here IP is considered as the higher layer.

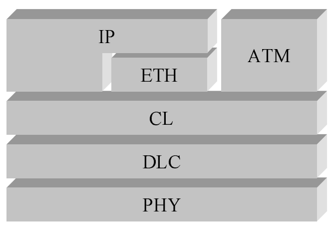

Figure 2. General Structure of H/2.

A. DLC Layer

An important feature of H/2 is the connection-oriented nature of the DLC layer. In particular, the DLC layer establishes connections between the AP (Access Point) and each MT (Mobile Terminal) of interest. The AP is the base station that supervises a certain area and the MT is a mobile terminal that depends on this AP at a given time. A MT can have one or more simultaneous connections.

A. Traffic Management at DLC

DLC has mechanisms to manage the wireless link, communicate with the upper layer, transmit or retransmit data etc. Three of them – listed below -are QoS related. Note that these mechanisms are not part of the standard and thus there is propriety in implementing new features/interfaces in order to provide H/2 with DiffServ QoS capabilities:

- **Wireless Connection Admission Control** (WCAC) is responsible for accepting or rejecting DLC connections. The decision of acceptance or rejection of the connections depends on the QoS requirements, traffic characteristics and radio link parameters of the requested connection.

- **Wireless Congestion Control** (WCC) mechanism is included in the DLC to manage congestion, which might occur at the AP when the number of arrivals at the AP is high. Whenever congestion occurs, it is WCC's responsibility to detect it and take all the appropriate actions to it.

- **Scheduler** is the third mechanism that is involved in the traffic management of H/2. It is responsible for preserving the traffic characteristics of the DLC connections already accepted. The allocation of the capacity will be done using a TDMA/TDD MAC protocol.

The WCAC, WCC and scheduler are not part of the H/2 standard. These three special entities, together with the aforementioned mapping, will be the focus of our future work. Thus a mechanism is needed, capable to provide info to DLC about the QoS demands of a certain data flow. This mechanism is part of the role of the Convergence Layer described next.

B. Convergence Layer

The Convergence Layer (CL) is divided into two sublayers: the Common Part (CP) sublayer, which is located just above the DLC, and the Service Specific Convergence Sublayer (SSCS), which is located at the uppermost part of the CL. This can be seen in Figure 3.

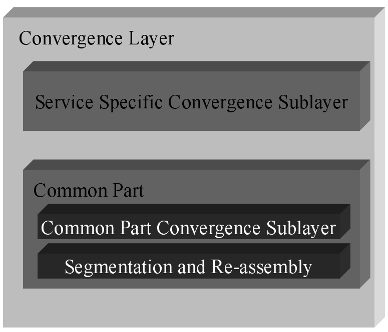

Figure 3. General Structure of CL.

Two main functions of the Common Part are the segmentation and reassembly processes. The Common Part is completely transparent regarding the QoS parameters originating from the upper layers. On the contrary, the SSCS maps the IP QoS parameters into the DLC specific QoS parameters.

The role of the SSCS is to adapt the IP layer to the H/2 specific DLC layer. We call this particular sublayer *IPSSCS*. Its main function is to perform the necessary QoS mapping, keep important information and handle all necessary communications when a service request is placed.

C. The Data and QoS flow in H/2

One of DLC's roles is to receive control information from the upper layers regarding QoS parameters. These parameters are mapped into DLC specific parameters by the IPSSCS. Then, at the DLC layer the three mechanisms presented earlier (WCAC, WCC, scheduler) deal with the traffic management issues. It is rather important to use an efficient mapping. The information regarding the QoS constraints will be used to accept/reject a new call (WCAC), to avoid congestion (WCC) and to send the data across the wireless medium (scheduler) in an efficient manner.

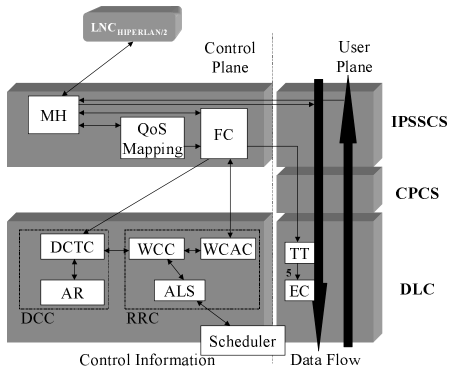

Figure 4. Dependencies between all entities in H/2 for the AP.

The important entities that appear in Figure 4 are described next. Note that these are the entities needed to be added/enhanced, in order for the mechanism presented later to be supported.

- **Error Control** (EC) is the error control mechanism that should be provided for services that are sensitive to packet losses.

- **DLC Connection Control** (DCC) is the part of DLC that contains mechanisms responsible for setting up a new connection, maintaining, re-negotiating and closing a DLC user connection at the DLC layer.

- **Adaptive Link Statistics** (ALS) is a mechanism that keeps track of the traffic for each particular connection.

- **DLC Communication Traffic Contracts** (DCTC) is the entity where all the information about traffic contracts is kept.

- **Address Resolution** (AR) is responsible for the mapping of the DLC address to the upper IP layer specific addresses.

- **Traffic Table** (TT) informs the scheduler about the destination of each packet.

- **Radio Resource Control** (RRC) is the set of entities responsible for resource control.

Figure 5 associated with a MT. Note that a MT is simpler than an AP. H/2 has a centralized architecture and the AP is responsible for most of the processes. In our case AP is designed to be responsible for the QoS mapping as well.

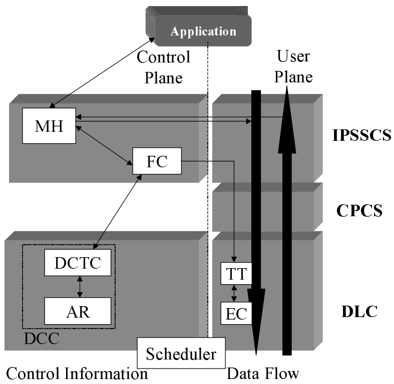

Figure 5. Dependencies between all entities in H/2 for the MT.

A. IPSSCS and its Special Entities

Figure 5, shows the communication between an application and MT-IPSSCS. This communication is necessary when the application opens a session with another host.

From Figure 4 it can easily be seen that there is a kind of communication between IPSSCS and the $LNC_{HIPERLAN/2}$. $LNC_{HIPERLAN/2}$ is a special entity that coordinates the H/2 network and its resources. For example when a user requests resources, the request will be addressed to the corresponding $LNC_{HIPERLAN/2}$. The latter's role is to respond according to the availability or not of the requested resources.

IPSSCS contains many special entities in order to communicate with $LNC_{HIPERLAN/2}$ and achieve good system performance.

- **Flow Control** (FC) is the part of IPSSCS that handles all requests originating from $LNC_{HIPERLAN/2}$. The information about the connections and their specific QoS parameters should be placed in a table inside FC. FC is not responsible for performing any controlling the actual data.

- **QoS Mapping** is the entity where mapping between DiffServ QoS parameters to H/2 specific QoS parameters is performed and vice versa.

- **Message Handler** (MH) is another entity of interest, responsible for the exchange of the necessary messages between the MT-IPSSCS and AP-IPSSCS. In order to support the QoS Mapping and the associated requests. MH also forwards the received messages to the appropriate entity.

IV. The Proposed Mechanism

An application running on a MT may request resources in order to open a session with another host (either another MT or a host located somewhere in the rest of the network - see Figure 1). The application itself is responsible for providing the characteristics of the resources needed to satisfy the demands for this particular session. These resource parameters will be forwarded to MT-IPSSCS (see Figure 5) and in particular to MH. MH is responsible for forwarding this particular request to AP-IPSSCS through the User Plane (a dedicated connection will be provided by DCTC).

The request arrives to AP-IPSSCS (Figure 4) and MH is responsible for forwarding the request to $LNC_{HIPERLAN/2}$. It should be noted that the QoS characteristics, as they are defined by the application, do not necessarily correspond to DiffServ-like classes of services. They might be specific for this particular H/2 DLC design. For example, in case the scheduler consists of two queues (high and low priority respectively), the requested resources might be expressed in terms of the queue's priority (high or low) and the average data rate required.

If they are expressed in DiffServ-like parameters then the request is forwarded immediately, otherwise, the QoS parameters are mapped to DiffServ compliant parameters. This is an advantage of this mechanism since it can be used irrespectively of the underlying QoS model. Consequently, since H/2 QoS design is not standard, all different vendors can use the mechanism.

When the request arrives at the $LNC_{HIPERLAN/2}$ it is the latter's responsibility to forward the request to the appropriate entity. In the already described scenario (which we will call *1st scenario*) there are actually two different cases. In the *first case* an application running on the MT opens a session with another MT (located under the supervision of the same $LNC_{HIPERLAN/2}$). In the *second case* the application opens a session with a host in the rest of the network.

Both cases look similar to each other but we need to consider that in the first case only one LNC (in particular $LNC_{HIPERLAN/2}$) coordinates all operations and handles the assignment of the system resources. Thus, proprietary QoS levels of services can be given to applications running “locally” (under the same $LNC_{HIPERLAN/2}$, which is probably the coordinator of a single building etc.) according to the requirements for “local” use. For the second case there are not any such “local” requirements. $LNC_{HIPERLAN/2}$ communicates with LNC_{DS} (a higher level LNC that is responsible for resource requests for the rest of the DiffServ network). In order to provide this kind of diversity in the offered service both cases are treated differently.

First case: $LNC_{HIPERLAN/2}$ makes a resource request to the AP. MH of AP-IPSSCS receives the request, it informs FC and then DLC is requested resources. The reply is forwarded back to $LNC_{HIPERLAN/2}$. Note that if the reply is positive then the requested resources are also reserved. The positive reply is forwarded back and the application running on the MT, that initiated the request, is allowed to begin the session.

Second case: $LNC_{HIPERLAN/2}$ asks AP for the specified resources and if the reply is positive then the request is forwarded to LNC_{DS}. If the reply of LNC_{DS} is positive then the session can start. If the reply is negative then the reserved resource in AP should be released.

The 2^{nd} *scenario* corresponds to the case where a host located in the rest of the network requests resources in order to open a session with a MT. In this scenario the request arrives at $LNC_{HIPERLAN/2}$ from LNC_{DS}. $LNC_{HIPERLAN/2}$ asks AP for resources and the reply is forwarded back to the LNC_{DS}.

A. Request AP for Resources

From the previous two scenarios it is concluded that the most important operation is the request for resources when it is applied to the AP. All other operations are message passing from one entity to another and they will not be further discussed here.

Figure 6 corresponds to the exchange of messages that should take place at the IPSSCS layer of AP. The request is generated by the $LNC_{HIPERLAN/2}$.

As soon as the request for resources arrives, IPSSCS maps the DiffServ QoS parameters to H/2 specific QoS. Then these H/2 specific QoS requirements are forwarded to the DLC and an answer is received whether it is possible to meet the needs of such a request or not. If the answer is negative then it is forwarded to the $LNC_{HIPERLAN/2}$. If DLC accepts the request then a special table keeps all relevant information that should be updated. Note that as long as the request can be satisfied, then the resources are also reserved. A positive or negative message is returned to $LNC_{HIPERLAN/2}$.

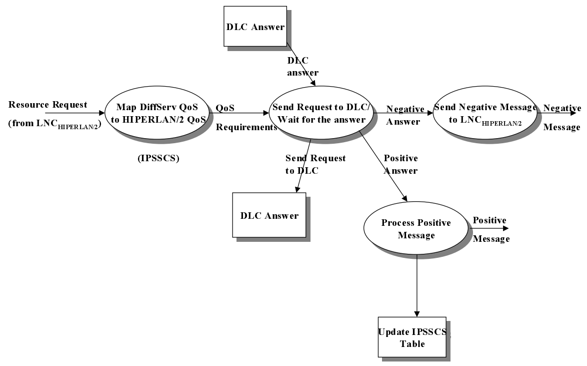

Figure 6. Resource Request Originating from the $LNC_{HIPERLAN/2}$.

QoS Mapping is one of the functionalities that IPSSCS should perform.

The exact messages exchanged between $LNC_{HIPERLAN/2}$, AP-IPSSCS and AP-DLC are presented in detail in Figure 7. A Finite State Machine (FSM) is depicted in the figure and a detailed description is followed.

It is important to give an example of the form of the resource requests. A resource request message consists of the source and destination entity in addition to the specific parameters that define the requested QoS parameters.

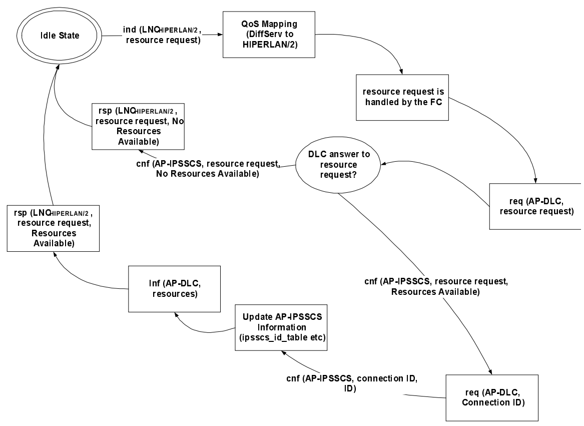

Figure 7. AP-IPSSCS: The AP is requested for resources from $LNC_{HIPERLAN/2}$.

The FSM shown in Figure 7 is described in detail next. The double-line ellipse corresponds to the **starting point**. Each ellipse corresponds to a particular **state** where a certain question should be asked in order for an event to occur, and consequently, move to the next state. Each arrow corresponds to an **event** while each rectangular corresponds to **action**s that should take place.

Starting Point: *Idle State?*

In this state nothing is happening. This is also the end point.

Event: *ind (LNC$_{HIPERLAN/2}$, resource request)*

LNC$_{HIPERLAN/2}$ sends a request for resources to the AP-IPSSCS.

Action: *QoS Mapping (DiffServ to HIPERLAN/2)*

Mapping of QoS parameters.

Action: *resource request is handled by the FC*

FC is notified that it should ask DLC for resources and if this request is to be satisfied, FC will be aware about the particular connection ID that is needed when data should be forwarded.

Action: *req (AP-DLC, resource request)*

The resource request is forwarded to the AP-DLC. (This request is applied to the WCAC entity).

State: *DLC answer to resource request?*

AP-DLC is processing the request and as a result a negative or positive answer is returned. If the answer is positive, the resources are also reserved and DLC is able to return the ID of the connection.

Event: *cnf (AP-IPSSCS, resource request, No Resources Available)*

There are no resources available.

Action: *rsp (LNC$_{HIEPRLAN/2}$, resource request, No Resources Available)*

The LNC$_{HIPERLAN/2}$ should be informed.

Event: *cnf (AP-IPSSCS, resource request, Resources Available)*

There are resources available. These resources are reserved and the only thing that remains is the parameters of the particular DLC connection.

Action: *req (AP-DLC, Connection ID)*

A request is applied to AP-DLC in order to get identification (ID) for the connection. (This request is applied to the DCTC entity).

Event: *cnf (AP-IPSSCS, connection ID, Con. ID)*

Information about the characteristics of the connection arrived.

Action: *Update AP-IPSSCS Information (ipsscs_id_table)*

Update information table (called *ipsscs_id_table*) with resource characteristics for this particular connection.

Action: *inf (AP-IPSSCS, resources)*

Specific DLC entities should be updated (TT).

The FSM in Figure 7 is described next as an MSC (Message Sequence Chart), where all special entities and the interactions between them can be clearly seen. Refer to Figure 4 and Figure 5 for the interfaces between different special entities.

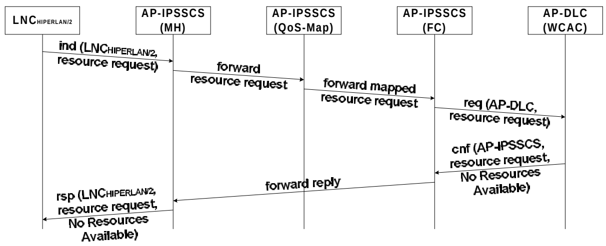

Figure 8. LNC$_{HIPERLAN/2}$ requests resources. There are no resources left in the system to satisfy the request.

Figure 8 depicts the case where there are no resources left in the system and as a result the request cannot be satisfied. Figure 9 corresponds to the case where the system does have the resources to satisfy the request.

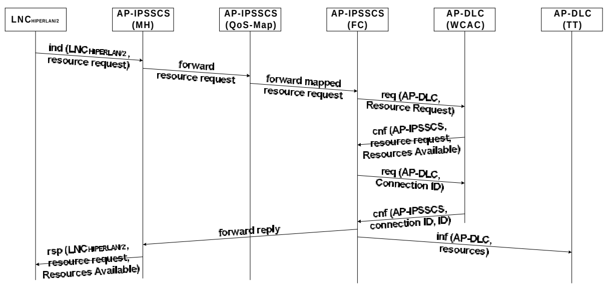

Figure 9. LNC$_{HIPERLAN/2}$ requests resources. The request can be satisfied.

V. Conclusions

The proposed mechanism is a first attempt to provide H/2 with a mechanism capable of providing DiffServ QoS. Mapping between DiffServ QoS and H/2 specific QoS parameters is necessary but this is realistic only if a mechanism like the one presented in this paper is implemented inside H/2.

The important advantage of the mechanism is that it can be used by all different H/2 systems that provide their own specific QoS capabilities.

Another advantage is that IPSSCS can be included transparently to an existing H/2 DLC.

A disadvantage is that DLC needs special features to be added or enhanced and thus any existing standard DLC implementation should be modified.

VI. Future Work

The proposed mechanism gives us flexibility the to study different mappings between DiffServ and H/2 specific QoS parameters. H/2 is located at the edge of the IP network and it is important for each user that wants to access the network, to receive certain and well preserved QoS. H/2 is a Wireless LAN and the nature of the wireless medium might have a great impact on the performance of the overall system.

The focus of our future work will be mainly on the *performance of the system* when a certain *mapping* algorithm is used. Different *scheduling policies* as well as an improved *WCAC* will be investigated.

Performance results will be obtained through *queuing analysis*. *Simulation results* as well as *experimentation* since the proposed mechanism can be used in a real H/2 system, allowing the evaluation of the performance of the proposed scheduler, WCAC and mapping in a *real system*.

VII. ACKNOWLEDGEMENTS

This work has been conducted under the framework of the HARMONICS IST program (IST-1999-11719). The project partners, Lucent Technologies, Portugal Telecom Inovacao, Corning, T-Nova - Deutsche Telekom, KPN Research, Mason, IMEC - University of Ghent, University of Limerick and INTRACOM are gratefully acknowledged for their inputs, as well as the European Commission partly funding the HARMONICS project by the IST Programme.

REFERENCES

[1] ETSI TS 101 493-2 v1.1.1 (2000-04): "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Packet Based Convergence Layer; Part 1:Common Part".

[2] ETR0230002 V0.2.0 (1999-04):"Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; System Overview".

[3] K. Nichols, V. Jacobson, and L. Zhang, "A Two-bit Differentiated Services Architecture for the Internet", ftp://ftp.ee.lbl.gov/papers/dsarch.pdf, November 1997.

[4] S. Blake, D. Blake, E. Davies, Z. Wang and W. Weiss, "An Architecture for differentiated Services, " RFC 2475, December 1998.

[5] M. Lapierre et al, "The TINA Book, A co-operative solution for a competitive world," Prentice Hall, 1999.