PalC NETWORKS

Egress QoS in BCM Katana Chipset

1. Introduction

This document describes the project undertaken by PalC Networks for developing QoS feature for Katana chipset

1.1. QoS Introduction

The key to providing QoS in the new service architectures is the ability to differentiate traffic and to provide differentiated service levels based on the types of traffic. For example, for real-time applications such as voice over IP (VoIP), the amount of available bandwidth and end-to-end delay is crucial compared to fax and e-mail transmissions, which are quite insensitive to bandwidth and delay issues.

To provide QoS from a network, the following must be satisfied:

- User/application requirements should be known to the network; and
- The network should have appropriate mechanisms for providing service levels that approximate these requirements.

1.2. IP QoS

The standard IP architecture was never designed to deliver on either of the two; it is based on a “best-effort” model where all network traffic is equally important and everyone receives service based on availability, without guarantees.

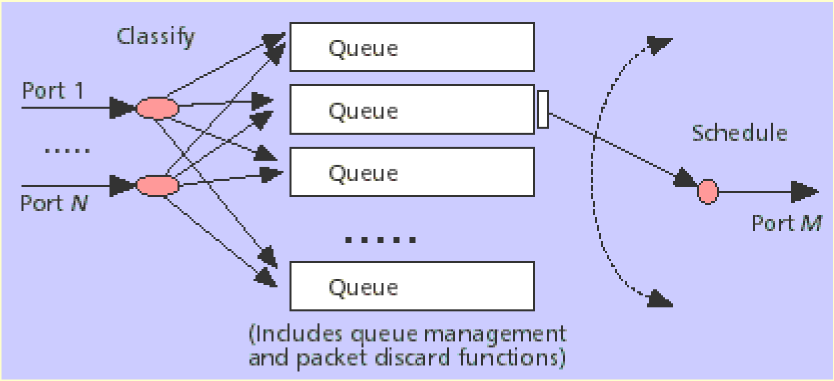
In the absence of QoS mechanisms, the industry traditionally has opted for over-provisioning bandwidth. While bandwidth over-provisioning continues, industry experts agree that QoS mechanisms are needed to address the needs of converging networks.

In general, end-to-end QoS in Internet is built from the concatenation of edge-to-edge QoS from each domain through which traffic passes, and ultimately depends on the QoS characteristics of the individual hops along any given route. The solution can be broken into three parts:

- per-hop QoS,
- traffic engineering, and
- signaling/provisioning.

QoS mechanisms do not generate more bandwidth. They manipulate router/switch queues so that when congestion occurs, priority “VIP” traffic is serviced quickly, while less important traffic experiences delays and drops. The network applies packet-filtering criteria to identify and prioritize VIP traffic using either provisioned or signaled QoS: Provisioning assumes the network nodes are configured ahead of time, while signaling assumes that filtering criteria are communicated in real time, upon demand. Classification is based on class of service (CoS) or QoS criteria. QoS deals with individual flows; CoS, generally considered a subset of QoS, deals with aggregate classes of traffic. After being classified, packets are serviced by a predefined queuing discipline that determines their final service level.

Because not all QoS mechanisms are the same, selecting the right QoS mechanism for the network could affect results significantly.

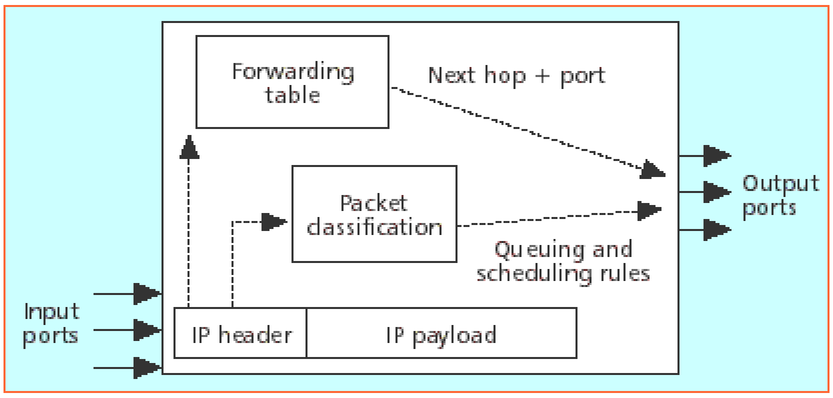


A variety of standard QoS approaches are common. Some router/switches are capable of setting filters to classify traffic and map it to specific queues. Some of the more popular disciplines include:

- Priority Queuing (PQ),
- Class-Based Queuing (CBQ),
- Weighted Fair Queuing (WFQ), and
- Random Early Detection (RED).

In all these approaches, the entire QoS process (classification and queuing) is provisioned within a single node and requires no cooperation from others. However, the process of filtering packets based on multiple attributes causes

- High overhead,
- Does not scale well, and
- Hard to create consistent multi-hop QoS by preconfiguring individual routers in a routing insensitive manner.



An important issue in the Internet, and consequently in every network connected to it, is support for multimedia applications (video, voice). These applications have specific requirements in terms of delay and bandwidth which challenge the original design goals of IP's best effort service model, and call for alternate service models and traffic management schemes that can offer the required quality of service (QoS). To this end, two QoS architectures have emerged in the IETF:

- **Integrated services architecture (IntServ)**, which provides end-to-end QoS on a per-flow basis; features soft states and end-to-end signaling.
- **Differentiated services architecture (DiffServ)**, which supports QoS for traffic aggregates; features class of flows and code points contained in the IP header's differentiated services field.

Both proposals suggest solutions to overcome the QoS limitations in the current best-effort IP service architecture. Each system has, however, its own advantages and disadvantages, and its own role to perform in an appropriate segment of an IP network.

We now review these two proposals on how such QoS enabling schemes could be utilized to enhance the best effort service model of IP architecture

1.2.1. Int Serv

IntServ was defined in IETF RFC 1633, which proposed the resource reservation protocol (RSVP) as a working protocol for signaling in the IntServ architecture. This protocol assumes that resources are reserved for every flow requiring QoS at every router hop in the path between receiver and transmitter using end-to-end signaling.

The IntServ model for IP QoS architecture defines three classes of service based on applications'delay requirements (from highest performance to lowest):

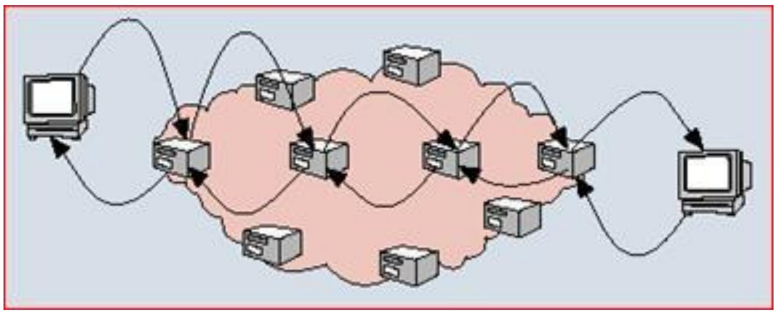
- **Guaranteed-service class** - with bandwidth, bounded delay, and no-loss guarantees;
- **Controlled-load service class** - approximating best-effort service in a lightly loaded network, which provides for a form of statistical delay service agreement (nominal delay) that will not be violated more often than in an unloaded network;
- **Best-effort service class** - similar to that which the Internet currently offers, which is further partitioned into three categories:
    - interactive burst (e.g., Web),
    - interactive bulk (e.g., FTP) and
    - asynchronous (e.g., e-mail)

The main point is that the guaranteed service and controlled load classes are based on quantitative service requirements, and both require signaling and admission control in network nodes. These services can be provided either per-flow or per-flow-aggregate, depending on flow concentration at different points in the network. Although the IntServ architecture need not be tied to any particular signaling protocol, Resource Reservation Protocol (RSVP) described below, is often regarded as the signaling protocol in IntServ. Best-effort service, on the other hand, does not require signaling.

RSVP

Using a method similar to the switched virtual circuit (SVC) of asynchronous transfer mode (ATM) networks, IntServ uses RSVP between senders and receivers for per-flow signaling (Fig.5). RSVP messages traverse the network to request and reserve resources. Routers along the path, including core routers, must maintain soft states for RSVP flows. (A soft state is a temporary state governed by the periodic expiration of resource reservations, so that no explicit path tear down request is required. Soft states are refreshed by periodic RSVP messages.)

RSVP is a set-up protocol providing a receiver-based, guaranteed, end-to-end QoS pipe. A reserved pipe is created in the following manner: First, PATH messages flow from the sender downstream to discover the data path. An RESV message, originating from a receiver and traveling in the reverse direction of the PATH messages, attempts to set local Integrated Services standard (IntServ) reservation for the flow. Each node along the path may either admit or reject the reservation subject to capacity or policy admission controls.



The major advantage of IntServ is that it provides service classes, which closely match the different application types described earlier and their requirements. For example, the guaranteed service class is particularly well suited to the support of critical, intolerant applications. On the other hand, critical, tolerant applications and some adaptive applications can generally be efficiently supported by controlled load services. Other adaptive and elastic applications are accommodated in the best-effort service class.

A major characteristic of IntServ is that it leaves the existing best-effort service class mostly unchanged (except for a further subdivision of the class), so it does not involve any change to existing applications. This is an important property since IntServ is then capable of providing this class of service as efficiently as the current Internet. IntServ also leaves the forwarding mechanism in the network unchanged. This allows for an incremental deployment of the architecture, while allowing end systems that have not been upgraded to support IntServ to be able to receive data from any IntServ class (with, of course, a possible loss of guarantee).

IntServ provides a very interesting set of service classes that, although maybe not ideal, represent an excellent approximation of the kind of services required in a global telecommunication platform since it does not discriminate against any applications.

Although IntServ is a straightforward QoS model, end-to-end service guarantees cannot be supported unless all nodes along the route support IntServ. This is obviously so because any best-effort node along any route can treat packets in such a way that the end-to-end service agreements are violated. In the case of end-end implementation of IntServ QoS model, it is recognized by the industry that the support of per-flow guarantees in the core of the Internet will pose severe scalability problems. Therefore, scalability is a key architectural concern for IntServ, since it requires end-to-end signaling and must maintain a per-flow soft state at every router along the path. Other concerns are, how to authorize and prioritize reservation requests, and what happens when signaling is not deployed end-to-end. Because of these issues, it is generally accepted that IntServ is a better candidate for enterprise networks (i.e., for access networks), where user flows can be managed at the desktop user level, than for large service provider backbones. A hybrid model (RSVP-DS) that uses RSVP at the edges and DiffServ in the backbone has been proposed and seems to be winning consensus as a backbone service architecture concept.

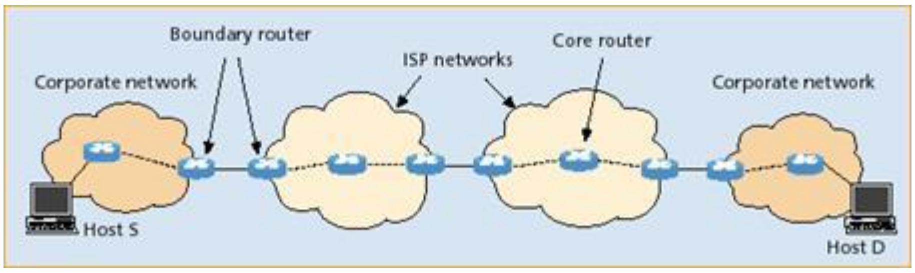
Finally, although the subclassing of best-effort service, although already a significant improvement on the flat best-effort service currently provided in the Internet, finer-grained subclassing of the best-effort service class may be desirable in a commercial network.

1.2.2. Diff Serv (Differentiated Services Architecture)

Diff-Serv is the product of an IETF working group that has defined a more scalable way to apply IP QoS. It has particular relevance to service provider networks. *Diff-Serv minimizes signaling and concentrates on aggregated flows and per hop behaviour (PHB) applied to a network-wide set of traffic classes.* Arriving flows are classified according to pre-determined rules, which aggregate many application flows into a limited and manageable set (perhaps 2 to 8) of class flows.

Traffic entering the network domain at the edge router is first classified for consistent treatment at each transit router inside the network. Treatment will usually be applied by separating traffic into different queues according to the class of traffic, so that high-priority packets can be assigned the appropriate priority level at an output port.

DiffServ approach separates the classification and queuing functions. Packets carry self-evident priority marking in the Type-of-Service byte inside packet headers. (ToS byte is part of the legacy IP architecture.) IP Precedence (IPP) defines eight priority levels. DiffServ, its emerging replacement, reclaims the entire ToS byte to define up to a total of 256 levels. The priority value is interpreted as an index into a Per-Hop Behavior (PHB) that defines the way a single network node should treat a packet marked with this value, so that it will provide consistent multi-hop service. In many cases, PHBs are implemented using some of the queuing disciplines mentioned earlier. This method allows for an efficient index classification that is considered to be highly scalable and best suited for backbone use. However, translating PHBs into end-to-end QoS is not a trivial task. Moreover, inter-domain environments may require a concept known as “bandwidth brokerage,” which is still in the early research stage.



DiffServ outlines an initial architectural philosophy that serves as a framework for inter-provider agreements and makes it possible to extend QoS beyond a single network domain. *The DiffServ framework is more scalable than IntServ because it handles flow aggregates and minimizes signaling, thus avoiding the complexity of per-flow soft states at each node. Diff-Serv will likely be applied most commonly in enterprise backbones and in service provider networks*.

There will be domains where IntServ and DiffServ coexist, so there is a need to interwork them at boundaries. This interworking will require a set of rules governing the aggregation of individual flows into class flows suitable for transport through a Diff-Serv domain. Several draft interworking schemes have been submitted to the IETF.

The DiffServ architecture is an elegant way to provide much needed service discrimination within a commercial network. Customers willing to pay more will see their applications receive better service than those paying less. This scheme exhibits an "auto-funding" property: "popular" traffic classes generate more revenues, which can be used to increase their provisioning.

A traffic class is a predefined aggregate of traffic. Compared with the aggregate of flows described earlier, traffic classes in DiffServ are accessible without signaling, which means they are readily available to applications without any setup delay. Consequently, traffic classes can provide *qualitative* or *relative* services to applications that cannot express their requirements quantitatively. This conforms to the original design philosophy of the Internet. An example of qualitative service is "traffic offered at service level A will be delivered with low latency," while a relative service could be "traffic offered at service level A will be delivered with higher probability than traffic offered at service level B." *Quantitative* services can also be provided by DiffServ. A quantitative service might be "90 percent of in-profile traffic offered at service level C will be delivered."

Since the provisioning of traffic classes is left to the provider's discretion, this provisioning can, and in the near future will, be performed statically and manually. Hence, existing management tools and protocols can be used to that end. However, this does not rule out the possibility of more automatic procedures for provisioning.

The only functionality actually imposed by DiffServ in interior routers is packet classification. This classification is simplified from that in RSVP because it is based on a single IP header field containing the DS codepoint, rather than multiple fields from different headers. This has the potential of allowing functions performed on every packet, such as traffic policing or shaping, to be done at the boundaries of domains, so forwarding is the main operation performed within the provider network.

Another advantage of DiffServ is that the classification of the traffic, and the subsequent selection of a DS codepoint for the packets, need not be performed in the end systems. Indeed, any router in the stub network where the host resides, or the ingress router at the boundary between the stub and provider networks, can be configured to classify (on a per-flow basis), mark, and shape the traffic from the hosts. Such routers are the only points where per-flow classification may occur, which does not pose any problem because they are at the edge of the Internet, where flow concentration is low. The potential noninvolvement of end systems, and the use of existing and widespread management tools and protocols allows swift and incremental deployment of the DiffServ architecture.

1.3. Requirement

The requirement is to bring egress QoS functionality in Katana based chipset. The QoS features to be supported in mentioned in section 4.

2. NOS Platform Architecture

The below diagram represents the high-level overview of the NOS architecture.

3. Features to be supported

- Classification based on Layer 4 source, destination port number
- Support E-LSP (EXP-inferred-PSC LSP) in RFC3270 to use the MPLS-EXP field to determine both PSC and drop preference
- Supported on layer 2/layer 3 interfaces and CIR, PIR, CBS, PBS should be configured on both ingress port and egress port (only on customer port)
- Should be supported Diffserv PHBs and mapping of different QoS classes to different PHBs
- RFC 2474 DiffServ Precedence; RFC 2598 DiffServ Expedited Forwarding (EF);
- RFC 2597 DiffServ Assured Forwarding (AF)

4. Approach

5. GLOSSARY

BFD	Bidirectional Forwarding Detection
CE	Customer Edge
H&S	Hub & Spoke
IDU	Indoor Unit
MNGT	Management
ODU	Outdoor Unit
PE	Provider Edge
PoC1	Point of Concentration 1st level (aggregation level next to the core network)
PoC2	Point of Concentration 2nd level (intermediate aggregation level)
PoC3	Point of Concentration 3rd level (first aggregation point after last mile/access)
SDN	Software Defined Networks
DCSG	Disaggregated Cell Site Gateways