

DQSA and the Internet

Graham Campbell

campbell@dqsa.net

Ph. 1-323-205-5041

Abstract: DQSA (Distributed Queue Switch Architecture) provides an efficient and economical means of switching packets by enabling users to share a common communications channel, over any distance and at any speed. This paper describes how DQSA can be utilized to provide both packet and fixed-bandwidth services for the Internet using only the underlying synchronous infrastructure – routers are bypassed.

1.0 Introduction

1.1 Computers Beget Packets

The arrival of computers meant that for the first time entities, albeit not human, could transmit information in milliseconds. POTS (Plain Old Telephone System) facilities were the only option for establishing connectivity between computer-based devices but the almost century-old experience that a circuit, once established, would be in use for a much longer time than it took to establish the connection was stood on its head. Using POTS as is for computer communications was not economically viable.

Packet switching was proposed as a solution to this low utilization by several researchers [1] and what could be described as the first router, the IMP (Internet Message Processor) was developed for what became known as the Arpanet, the progenitor of the Internet. Information was segmented into packets with each packet containing a destination. The packets were then transmitted to a “packet switch” (router) where each packet in turn was routed via the port that would move it towards the final destination. The practice of breaking up of information into packets, including voice, has become almost universal. These networks are asynchronous in that the time taken for information to travel from A to B could vary over time.

However, despite a serious look at utilizing what can be described as “pure packet switching” in the form of ATM (Asynchronous Transfer Method), these packet switches today rely on what can be described as the 21st century equivalent of Alexander Graham Bell's manually switched synchronous network wherein the original copper and switchboard operators are replaced by fiber and computer controlled optical

switches. These are synchronous in the sense that once the connection has been established the time for information to travel from A to B never varies.

Thus the Internet today can be thought of as consisting of two networks: an asynchronous network sitting astride the circuit-switched infrastructure. Figure 1 shows a circuit-switched network with the addition of IP routers located in both the carrier and customer premises, the latter make for increased utilization of the underlying switched network.

However, an asynchronous network is subject to congestion and the discarding of packets, and is not conducive to providing the equivalent of fixed-bandwidth services.

1.2 A Better Way to Move Packets

The “infinite” buffer that queueing theory calls for to avoid the discarding of packets is not achievable, and as pointed out later in this paper, such a buffer would introduce more problems, so what is called for is a fresh approach to moving packets.

There are actually two “switching” mechanisms that can be utilized to deliver a packet to a destination and both emerged at roughly the same time. The system described above uses routers to deliver a single packet to a designated destination. The second method employs a single channel of sufficient capacity to carry packets destined for multiple recipients, all connected to that same channel. The intended recipient copies only those packets so addressed. The first deployment of what can be called a distributed “shared” switch was deployed by Abramson [3] at the University of Hawaii to allow island campuses access to the new Arpanet. Two wireless channels, one inbound and one outbound, were utilized. All packets were broadcast to all recipients on the outbound channel; a fairly primitive MAC (medium access control), Aloha, was developed by Abramson that enabled individual nodes to access the inbound channel.

Soon the sharing concept was tried on copper, which led to the development of Ethernet by Metcalfe and Boggs [4]. For distances up to 2.5 Km technology supported a channel with 10 Mbps capacity and even though Ethernet was not efficient, it became the dominant method of switching packets over a limited geographic area. However Ethernet does not scale and as speeds increased to 100 Mbps and then 1 Gbps and higher the use of Ethernet switches (routers) became standard and shared Ethernet usage fell sharply.

Thus the new architecture this paper proposes is really an old architecture, i.e., utilize a pair of common channels to deliver and collect packets from multiple

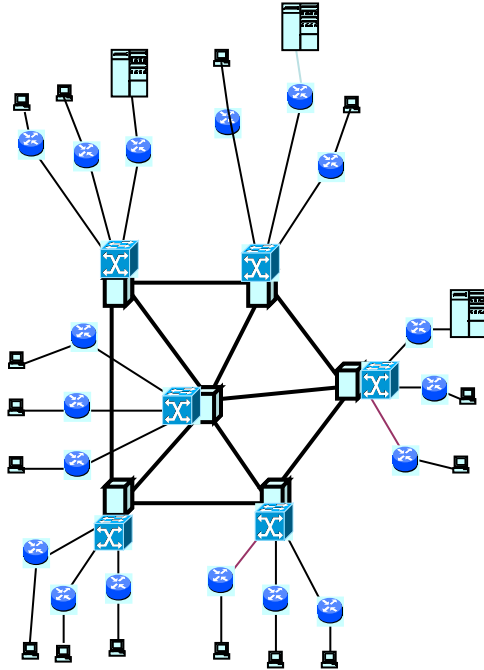


Figure 1 – Packet Network Superimposed on Switched Infrastructure

stations. New technology now makes it feasible for this method of switching to be utilized over distances of thousands of Km at multi Gbps rates as opposed to the 2.5 Km limit at 10 Mbps of the original Ethernet.

The shared approach can be likened to Fedex and their hub-and-spoke system. Prior to Fedex a map depicting the movement of packages around the USA would look very much like a map of the Internet today, i.e., a multiplicity of nodes, each connected to adjacent nodes, and with algorithms developed over scores of years that dictated a “minimum-distance -route” between any two points. Fred Smith, founder of Fedex, had an idea that was counter-intuitive in that he proposed that that all parcels be sent to a central hub via spokes where the parcels would be efficiently sorted for outward shipment on these same spokes. [5] In other words it was claimed that rather than sending a parcel directly from Omaha to Dallas it would be more efficient to send a parcel from Omaha to Dallas via Memphis. If accomplished overnight by using aircraft it would still satisfy delivery requirements for virtually all shippers. Smith recognized that making the overall system more efficient overcame the fact that for an individual parcel more energy and time was required to move that parcel over two legs. Hub-and-spoke proved so successful that it is now in general use.

Why not “hub-and-spoke” for packets? The increase in energy and time to route a packet over two legs rather than one is almost too small to be measured. Obviously for hub-and-spoke to be effective it is necessary that only a single packet arrive at any given time at the hub, or at any intermediate point, where two or more circuits join. Hub-and-spoke is common in LANs but typically confined to a single hop.

1.3 Hub-and-Spoke for Packet Networks

The requirement that only a single packet arrive at any given time at any junction point does eliminate internal queueing and the need for a buffer. However, queues are a fact of life in networks thus rather than queues being eliminated, they are moved elsewhere. In DQSA the queues are moved to the source nodes so that rather than hundreds of packets arriving from disparate sources at single queue, there are now hundreds of smaller queues, each with a packet at the head of a much shorter queue awaiting the green light. The equivalent of an “infinite” buffer is achieved by the simple expedient of refusing admittance when the queue length indicates too long a wait before transmission. This mechanism could be used by TCP to achieve the flow control now predicated on packets being dropped and thus not acknowledging. And no congestion.

It is not claimed that even if the requirement of perfect queueing at the edge of the network is achieved that this would satisfy all Internet traffic. But what is suggested is that hub-and-spoke would satisfy those segments of traffic that are at present subject to aggregation at a common point. VPNs (Virtual Private Networks) are a case in point where the intent is, using software control, to restrain all traffic to a closed group of nodes, indeed often all traffic is destined to a common point, typically HQ or some central data base. In addition to VPNs there are natural aggregation points in a typical packet network that occur because of the desire to take advantage of some feature of the aforementioned underlying synchronous network. VPNs are utilized to further the discussion on how DQSA/hub-and-spoke can be implemented.

2.0 DQSA and Hub-and-Spoke

2.1 An Alternate Packet Switching Method

The underlying concept behind DQSA (Distributed Queue Switch Architecture) is that the shared switching described above, now also referred to as hub-and-spoke, will prove satisfactory in most VPNs and also where aggregation of packets occurs. The “spokes” are conventional circuits that can be shared by hundreds or even thousands. Packet traffic is

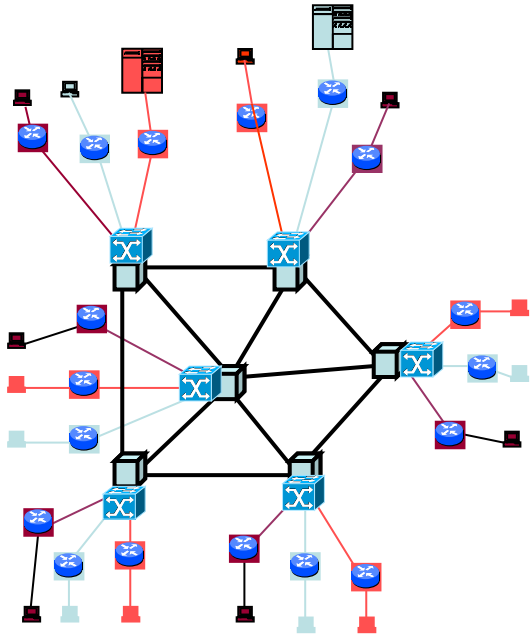


Figure 2 – Packet Network with VPNs Color-coded

sporadic by nature and a common channel, in addition to being simpler to implement, achieves better utilization by taking advantage of statistical averaging. However, a requisite for implementing DQSA is a near-perfect access method; this is assumed for now but such methods are identified in Section 3. This section describes how a typical VPN can be implemented utilizing only the underlying circuit-switched infrastructure.

Figure 2 is the network of Figure 1 color coded to show five VPNs, with computers representing the

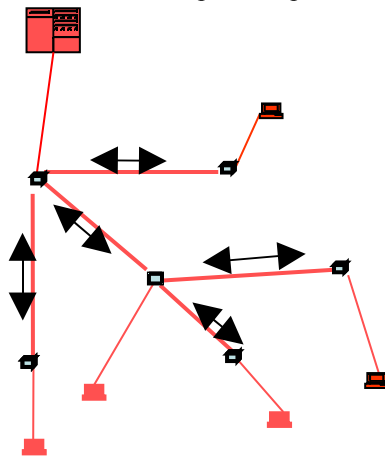


Figure 4 – VPN Configured as Physical Private Network

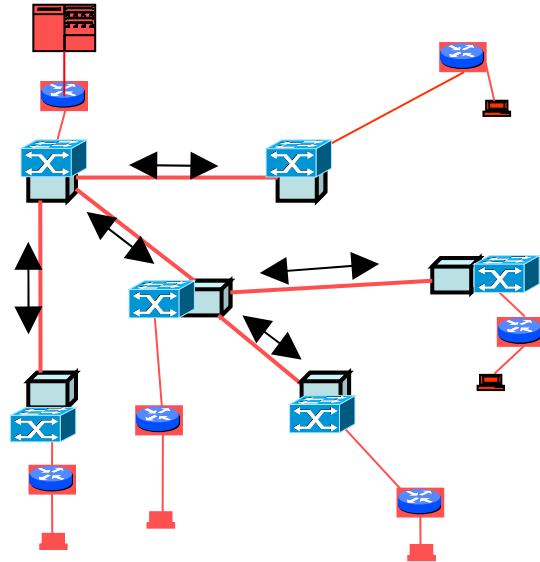


Figure 3 – Single VPN has Tree-and-branch (hub-and-spoke) Topology

central site of two of the VPNs and the equivalent color terminals representing the respective branch offices. Assume that the red VPN serves a series of branch offices of a corporation. Packet network topology is normally a mesh so that successive packets in a single message could travel via different paths to the same destination node. In general, however, when shortest-path routing is used the links traversed by packets in a typical VPN form a tree-and-branch topology, as illustrated in Figure 3, with the root node usually at the headquarters of the organization. The topology can also be treated as a hub-and-spoke with sub-spokes. DQSA allows for such a topology.

2.2 Are Routers Necessary to Support VPN traffic?

The question is: “could equivalent packet services be provided that would satisfy VPN requirements using only the switched infrastructure?” The answer is in the affirmative and is shown by defining the requisites for such a system. A router has two functions: (1) the routing function that steers packets from an input port to the desired output port, and (2) a buffer function that queues packets when two or more packets simultaneously arrive destined for the same output port. We need only demonstrate that a packet can travel from source to destination without relying on either of these functions at intermediate points.

2.3 Eliminating the Router

The need for a router function at each node of the VPN is eliminated by ensuring that each router be “hard-

wired” so that packets arriving on input lines be directed to a specific output port. When the root node is not the final destination, the packet is simply turned around and broadcast. At this point we would have a very expensive VPN with costly routers acting as simple buffers. But if we then assume that packets are perfectly scheduled, i.e, only one packet at a time will ever arrive at the router at a junction point then the router can be removed and the two, or more, circuits joined using “or” logic. This is demonstrated in Figure 4 with the removal of the routers, the remaining boxes represent standard switching office hardware where two circuits are joined. Figures 5(a), (b) and (c) illustrate the steps that allow removal of the router. The key to implementation, of course, is to schedule the packets, and section 3 introduces the access methods that satisfy that need by having queueing occur at the edge of the network.

2.4 Bandwidth Requirements

The packets in the VPNs will in the main be following “shortest path” routing resulting in packets flowing from hundreds or thousands of branch nodes to their respective root nodes and HQ. The traffic flow is actually similar to the pattern if that VPN had been implemented as a physical private network. There will in fact be less traffic since there will be no packet discard resulting in retransmission. In addition the efficient access method described in section 3 results in close to full utilization of the circuits thus reducing capacity requirements.

A DQSA private network could be implemented with minimal or no modifications to the existing plant; the routers are simply bypassed. It must also be remembered that a physical DQSA network can itself support multiple virtual DQSA networks. A single

physical DQSA network, operated say by one of the national carriers, could support multiple virtual networks, that are assigned to specific companies or bands of users.

In general the inbound traffic in our sample networks is destined for the HQ node. When outbound, a message is “broadcast” which appears to “waste” bandwidth but consider that typically the HQ transmits only a single message at a time to the root node of a conventional VPN we can ask, does it “waste” bandwidth that multiple stations “copy” this message before discarding it. A plus for DQSA in this environment is that often data transmitted from HQ is common to most or all branches; with DQSA broadcast is the default mode and so all stations receive all messages as a matter of course, addressing mechanisms ensuring only authorized stations actually see the messages.

In the environment under discussion, a typical VPN, the case can be made that there would be a reduction in required bandwidth for a DQSA PN over a conventional VPN. However it is acknowledged that research is necessary to determine the actual bandwidth requirements for each type of network under different degrees of loading.

3.0 A MAC That Satisfies the Requirements of DQSA.

We have demonstrated that the requirements of a VPN could be satisfied by connecting the remote nodes to a common physical channel and ensuring that a node only transmitted at such a time that there would be a clear path from the node to a hub. Two access methods enable the implementation of a DQSA network: DQRAP (Distributed Queue Random Access Protocol)

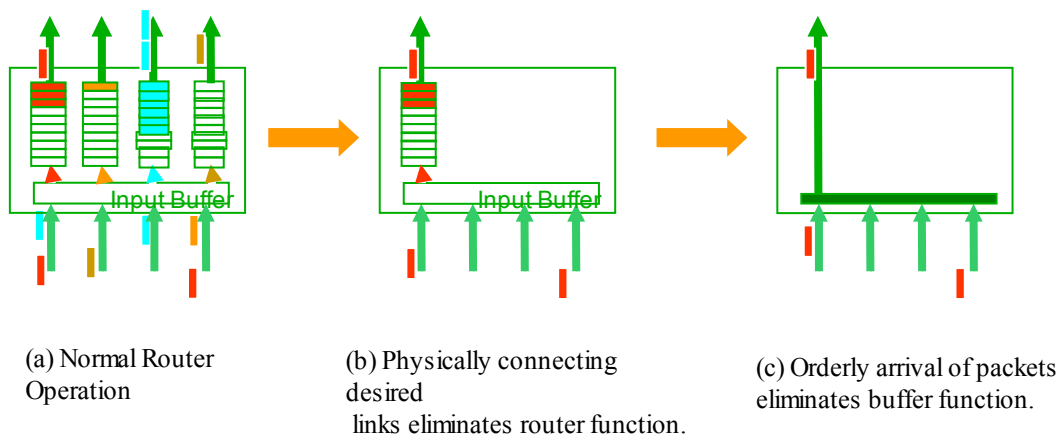


Fig. 5 – Steps in Eliminating Need for Router

[8], and XDQRAP (Extended DQRAP) [6]. DQRAP supports the transport of fixed length packets such as ATM cells while XDQRAP supports the transport of variable length packets. Both were developed at the Illinois Institute of Technology and possess close-to-ideal access characteristics in that they enable a communications channel to be efficiently shared by an arbitrary number of users distributed over arbitrary distances operating at any bit rate.

3.1 Overview of XDQRAP-based DQSA

DQSA based on XDQRAP divides the inbound and outbound channels into fixed size slots where each slot contains room for data (64, 128, etc., bytes) plus space for either two or three requests, resulting in an overall bandwidth utilization of upwards of 90%. This feature also facilitates “pipe-lining” so that service can be provided over arbitrary distances. Control is distributed among the nodes, no central controller is required but since in most instances a central hub is utilized it is convenient to assign some responsibilities to that hub. We next list some of the major features of XDQRAP that make it ideal for implementing DQSA.

3.2 DQSA and Packets

The ability to switch variable length packets is the key to supporting Internet traffic. With DQSA variable length packets are segmented for transmission resulting in possible fragmentation in the last segment, but there is no further overhead. The basic XDQRAP algorithm ensures both full utilization (less the above mentioned fragmentation) and minimal delay for the transmission of individual packets.

3.3 DQSA Priorities

True priorities are implemented as in a computer operating system, i.e., separate dispatch queues are maintained at nodes so that a packet is not transmitted from a node till all packets with higher priority waiting at other nodes have been transmitted [9]. Preemption is supported so that for instance a low-priority “jumbo Ethernet” packet can be interrupted in mid-transmission, again with no further overhead.

3.4 DQSA and Fixed-Bandwidth

The bandwidth in DQSA is divided into fixed-size segments and groups of contiguous segments are allocated to each packet but many applications, such as packet video, would be better served with the equivalent of a TDM channel. DQSA supports this feature; a node requests that a segment be allocated on a recurring basis resulting in an isochronous (TDM) channel of the desired bandwidth. This feature is of true significance since it means that DQSA can satisfy

with equal facility both packet and fixed-bandwidth requirements. This is described in detail by Wu and Campbell in [10].

3.5 Distributed Control

The word “Distributed” in DQSA indicates a truly desirable feature and XDQRAP delivers. Each node maintains the “state” of the network, i.e., the length of the transmission queue (those waiting to transmit) and the length of the collision resolution queue (the number of groups waiting to resolve collisions), each updated at the arrival of every segment. No central control is required but since the hub is a common point, it could be utilized. A potentially valuable feature is that each node can determine the length of time for a packet to reach the destination before entering it in the queue. See Wu and Campbell [6].

4.0 Observations

Observations are in two sections: one addresses the unique benefits of a DQSA hub-and-spoke network while the other discusses potential concerns.

4.1 Benefits

4.1.1 Control: Control is distributed to all the nodes, no central controller is required. The complexity of the DQSA NIC at each node is somewhat less than that of an Ethernet card.

4.1.2 Economics: A DQSA spoke with hundreds of nodes can be implemented by using the same physical circuits employed to connect the nodes to the packet network, using NICs that are comparable in cost to an Ethernet NIC. The routers are simply bypassed leading to a considerable decrease in network capital and operating costs.

4.1.3 Flexibility: DQSA operates at Level 2, above the physical layer, thus a DQSA spoke could consist of a copper link followed by a circuit on a fiber, or even include a wireless link. All that is required is that the delay across the junction point not vary with time. Furthermore both fixed-bandwidth and packet services can be supported on the same “spoke”.

4.1.4 Virtual DQSA: A DQSA hub-and-spoke could coincide with a physical network as described in Section 2. But just as VPNs are deployed on a physical packet network so multiple virtual DQSA spokes could be implemented on a single physical spoke. Carriers can establish hundreds or thousands of DQSA-based hub-and-spokes, physical or virtual, to satisfy communication requirements.

4.1.5 Utilization: There is no congestion in a DQSA network thus networks can be designed for average loading of 90%. The surges over 100% that cause chaos in conventional routers just mean that the

distributed queues get longer, temporarily. No lost packets except those due to line error. If only a single node has packets to send, that node can utilize 100% of available capacity, when a second node desires to transmit, the available capacity is split, automatically without any central control input, evenly between the two stations. And so on for an arbitrary number of stations. Priorities can be utilized to negate this inherent fairness.

4.1.6 Scaling: DQSA operates at any speed, can be deployed on parallel circuits, and will operate on an asymmetric “hub-and-spoke”. The advent of 100 Gbps and higher rates means that DQSA is in a good position to satisfy the increasing demand for digital video.

4.1.7 Integration of Switching: The Internet today is divided into a wireless component (shared routing of packets in cell phone towers) and a terrestrial component (switched routing of packets) on copper and fiber. This division is eliminated in that cell towers can themselves be placed on “spokes” in the manner demonstrated for VPNs in Section 2. The problem of cell tower backhaul is also reduced.

4.2 Concerns

4.2.1 Bandwidth Requirements: On the surface it appears that a DQSA network substitutes bandwidth for hardware; given that bandwidth is decreasing in cost faster than hardware this is a good trade-off. But in many instances there would be a savings in bandwidth, i.e., when the “spoke” is directly connected to multiple nodes and operates at say 90% capacity (possible with DQSA), then it is obvious that overall less bandwidth is being utilized to satisfy requirements than using conventional packet switching. But it is acknowledged that further research is in order to gain the true picture.

4.2.2 Broadcast vs. Unicast: Broadcast is the default for all wireless systems and so where security is a concern encryption is used so the same can be done in a DQSA network. If it was desired to not physically transmit a packet to a non-destination then in DQSA it is relatively straightforward to ensure that at junction points a packet is transmitted only on the link leading to that destination – much simpler than buffer and switch in a conventional router.

4.2.3 Reliability: DQSA operates at the Level 2 (Link/MAC) layer thus reliability is dependent on the underlying physical circuits. Where necessary as with conventional networks this may call for redundant physical circuits. However the absence of routers, the source of many if not the majority of network problems, suggests that a DQSA network, will be

more reliable. The NICs at the edge of the network, can be made safe from interfering with the network in the same manner as has been done with millions of Ethernet NICs.

4.2.4 Compatibility with TCP/IP: Problems first identified in the theoretical work that led to packet switching, i.e., congestion and buffer overflow, necessitating over-provisioning in packet networks, are still with us but the fact that packets are discarded is utilized by TCP to “throttle” traffic on individual streams and is a major reason for the successful operation of the Internet. The availability of low-cost memory has led to a dramatic increase in the size of the buffers but this in turn has introduced another problem i.e., packets tend to “hang around” past their previous “sell by” times and so upsetting the TCP/IP algorithms and so introducing increased latency and jitter. Doc Searls, Senior Editor of Linux Journal, turned over his EOF column to guest columnist Dave Täht who described the problem, in particular the research carried out by Jim Gettys who introduced the term “bufferbloat” to describe this phenomenon. [2]. Obviously even achieving an “infinite” buffer would not be a cure-all but it makes clear that any solution must be compatible with TCP/IP. DQSA achieves this with a minor modification, TCP simply uses the length of the queue, described in Section 3.5, instead of non-acknowledgment, of as the determinant as to when to submit a packet for transmission. There will be no retransmission, excepting due to line-error, thus improving efficiency.

5.0 Conclusions

From the 1870s to the 1970s communications was dominated by synchronous switched networks. For the past forty years what can be called hybrid networking has dominated. This paper describes DQSA, a switching mechanism that, in the main, utilizing already existing infrastructure, will provide flexible, high-utilization, congestion-free communications that satisfies virtually all requirements for the transport of voice, video and data.

References

- [1] <http://www.historycentral.com/Technology/1stTran sPhoneConvers.html>
- [2] Doc Searls “Whatever Sinks Your Boat”, Linux Journal June 2011, Issue 206, p80.
- [3] N. Abramson, “The ALOHA system -- Another alternative for computer communication,” in AFIPS Conf. Proc. Fall Joint Comp. Conf. 1970, pp. 281-285.
- [4] Metcalfe, Robert M. and Boggs, David R. (July 1976). "Ethernet: Distributed Packet Switching for

- Local Computer Networks". *Communications of the ACM* **19** (5): 395–405.doi:10.1145/360248.360253
- [5] Foust, Dean "Frederick W. Smith: No Overnight Success". *Business Week*, September 20, 2004.
- [6] C.T. Wu and G. Campbell, "Extended DQRAP (XDQRAP): A Cable TV Protocol Functioning as a Distributed Switch", *Proceedings of 1st International Workshop on Community Networking*, July 1994, San Francisco. *Computer Communication Review*, Vol 23, No. 4, Oct 1993, pp. 270-278.
- [7] H.J. Lin and G. Campbell "Using DQRAP (Distributed Queueing Random Access Protocol) for Local Wireless Communications." *Proceedings of Wireless '93*, July 14, 1993, pp. 625-635.
- [8] W. Xu and G. Campbell "DQRAP - A Distributed Queueing Random Access Protocol for a Broadcast Channel", presented at SIGCOMM '93, San Francisco, September 14, 1993. *Computer Communication Review*, Vol 23, No. 4, Oct 1993, pp. 270-278.
- [9] H. J. Lin and G. Campbell, "PDQRAP - Prioritized Distributed Queueing Random Access Protocol", *Proc. of 19th Conference on Local Computer Networks*, Oct. 1994, pp 82 - 91.
- [10] C. T. Wu and G. Campbell "CBR Channels on a DQRAP-based HFC Network", *SPIE '95 (Photonics East)*, Philadelphia, PA Oct. 1995.