A Study of Advanced Switching Fabric Design of a "Router"

Alok Sahu* Dr. Bharat Mishra* Gauri Shankar Kushwaha

Abstract—Routers perform the "traffic directing" functions on the internet. The rapid growth of the Internet has created different challenges for routers in backbone, enterprise, and access networks. Future routers must not only at high speed, but also one is important non-trivial issue such as switching fabrics with a wide range of packet format and routing protocol. Head of Line Blocking (HoLB) is a problem that occurs in any system where congestion exists at the output port. HoLB occurs when multiple packets, destined for multiple destinations, all share one queue. Packets destined for a specific location must wait until all packets ahead of it are processed before being passed through the switch fabric. An example of this is when several multiple lane highways are merged into a one lane highway. In this paper we suggest for a different approach, with the non-trivial issue Advanced Switching Fabric Design(HoLB) of a router in order to enable them to focus solely on managing a computer network.

Index Terms—Switching Fabric Card, Route Processor, Clock and Scheduler Card, Media Access Control, Virtual Output Queues, Multicast & Unicast Process

1 INTRODUCTION

Router is a platform for routing which is used for sharing internet access through sharing networks within local area. These routers have designed with the potential to transfer signals from a single point to the multiple exact destinations. It is essential to get a router for sharing your application and internet within your LAN.

These devices were different from most previous packet networks in two ways. First, they connected dissimilar kinds of networks, such as serial lines and local area networks. Second, they were connectionless devices, which had no role in assuring that traffic was delivered reliably, leaving that entirely to the host. The idea was explored in more detail, with the intention to produce a prototype system, as part of two contemporaneous programs. One was the initial DARPA-initiated program, which created the TCP/IP architecture in use today. Modern high-speed routers are highly specialized computers with extra hardware added to speed both common routing functions, such as packet forwarding, and specialized functions such as IPSec encryption. [9]

In light of explosive growth of the Internet and numerous enterprise networks in recent years, there is strong interest in the industry to design and build a new class of routers able to offer access to hundreds (or even thousands) of ports on a single router. Such a router can increase its capacity as the need arises. Many routers commercially available cannot scale well since they employ switching fabrics like crossbars or shared busses to interconnect key components. A next-generation must provide extensive scalability, able to connect a large number of ports. Hardware required for link aggregation to such a high data rate will be rather complex and expensive. Separately, a few scalable switching fabrics with direct interconnection styles have been developed [4, 8]. We have considered novel switching fabrics for scalable routers recently. Such a fabric comprises small routing units (RU) interconnected by connecting components (CC) in accordance with grid structures, where a CC is composed of a multistage interconnect. [9]

2. Components of a router

A generic router has four components: input ports, output ports, a switching fabric, and a routing processor. An input port is the point of attachment for a physical link and is the point of entry for incoming packets. Ports are instantiated on line cards, which typically support 4, 8, or 16 ports. The switching fabric interconnects input ports with output ports. We classify a router as input-queued or output queued depending on the relative speed of the input ports and the switching fabric. If the switching fabric has a bandwidth greater than the sum of the bandwidths of the input ports, then packets are queued only at the outputs, and the router is called an output-queued router. Otherwise, queues may build up at the inputs, and the router is called an input-queued router. An output port stores packets and schedules them for service on an output link.

Finally, the routing processor participates in routing Protocols and creates a forwarding table that is used in packet forwarding. An input port provides several functions. First, it carries out data link layer encapsulation and decapsulation. Second, it may also have the intelligence to look up an incoming packet's destination address in its forwarding table to determine its destination port (this is also called route lookup). The algorithm for route lookup can be implemented using custom hardware, or each line card may be equipped with a general-purpose processor. Third, in order to provide QoS guarantees, a port may need to classify packets into predefined service classes. Fourth, a port may need to run data link-level protocols such as SLIP and PPP, or network-level protocols such as PPTP. Once the route lookup is done the packet needs to be sent to the output port using the switching fabric. If the router is input queued, several input ports must share the fabric: the final function of an input port is to participate in arbitration protocols to share this common resource. The switching fabric can be implemented using many different techniques. [1]

The most common switch fabric technologies in use today are busses, crossbars, and shared memories. The simplest switch

fabric is a bus that links all the input and output ports. However, this is limited in capacity by the capacitance of the bus and the arbitration overhead for sharing this single critical resource. Unlike a bus, a crossbar provides multiple simultaneous data paths through the fabric. A crossbar can be thought of as 2N busses linked by N*N crosses points: if a cross point is on, data on an input bus is made available to an output bus, and else it is not. However, a scheduler must turn on and off cross points for each set of packets transferred in parallel across the crossbar. Thus, the scheduler limits the speed of a crossbar fabric. In a shared-memory router, incoming packets are stored in a shared memory and only pointers to packets are switched. This increases switching capacity. However, the speed of the switch is limited to the speed at which we can access memory. Unfortunately, unlike memory size, which doubles every 18 months, memory access times decline only around 5% every year. This is an intrinsic limitation with shared-memory switch fabrics. Output ports store packets before they are transmitted on the output link. They can implement sophisticated scheduling algorithms to support priorities and guarantees. Like input ports, output ports also need to support data link layer encapsulation and de-capsulation, and a variety of higher-level protocols. The routing processor computes the forwarding table, implements routing protocols, and runs the software to configure and manage the router. It also handles any packet whose destination address cannot be found in the forwarding table in the line card. [2][Figure: 1]

3. Switching fabrics Design

Internet Router is a multi-gigabit crossbar switch fabric that is optimized to provide high capacity switching at gigabit rates. This architecture allows multiple line cards to transmit and receive data simultaneously. The CSC is responsible for selecting which line cards transmit and which line cards receive data during any given fabric cycle.The switch fabric provides a physical path for Initial fabric downloader from the Route Processor (RP) to the line cards on power up, Express Forwarding updates, Statistics from the line cards, Traffic switching. [19]The switch fabric is an NxN non-blocking crossbar switch fabric where N stands for the maximum number of LCs that can be supported in the chassis (this includes the GRP). This allows each slot to simultaneously send and receive traffic over the fabric.[Figure: 2]

The CSC accepts transmission requests from line cards, issues grants to access the fabric, and provides a reference clock to all the cards in the system to synchronize data transfer across the crossbar. Only one CSC is active at any time. The CSC can be removed and replaced, without disrupting normal system operations, only if a second (redundant) CSC is installed in the system. One CSC must be present and operational at all times to maintain normal system operations. A second CSC provides data path, scheduler, and reference clock redundancy. The interfaces between the line cards and the switch fabric are monitored constantly. If the system detects a Loss of Synchronization (LoS), it automatically activates the data paths of the redundant CSC, and data flows across the redundant path. The switch to the redundant CSC usually occurs in the order of seconds (the actual switch time depends on your configuration and its scale), during which time there can be a loss of data on some/all LCs. [21,13]

An optional set of three SFCs can be installed in the router at any time to provide additional switch fabric capacity to the router. This configuration is called full bandwidth. The SFC cards increase the data handling capacity of the router. Any one or all of the SFCs can be removed and replaced at any time without system operations being disrupted or the router being powered down. For the length of time that any SFC is not functional, its data carrying capacity is lost to the router as a potential data path for the router's data handling and switching functions. [6,17]

When a packet comes in an interface, a lookup is performed the lookup determines the output LC, interface, and appropriate Media Access Control (MAC) layer re-write information. Before the packet is sent to the output LC through the fabric, the packet is chopped into Cells. A request is then made to the clock scheduler for permission to transmit a Cell to the given output LC. One cell is transmitted every fabric clock cycle by E0 LCs and every four fabric clock cycles by E1 and higher LCs. The output LC then re-assembles these Cells into a packet, uses the MAC rewrite information sent with the packet to perform the MAC layer rewrite, and queues the packet for transmission on the appropriate interface.[21]. If a packet arrives on an interface on an LC and is supposed to go out another interface (or on the same interface in case of sub-interfaces) on the same LC, it is still segmented into Cells and sent over the fabric back to itself.But a problem is there Head of Line Blocking (HoLB) problem that occurs in any system where congestion exists at the output port.[Figure:3]

4. Switching Fabric Design Approach

The job of moving the packets to particular ports is performed by switching fabrics. Switching can be approached in number of ways:

4.1 Switching via Memory: The simplest, easiest routers, with switching between output and input ports being done under direct control of CPU (router processor). Whenever a packet arrives at input port routing processor will come to know about it via interrupt. It then copies the incoming packets from input buffer to processor memory. Processor then extracts the destination address look up from appropriate forwarding table and copies the packet to output port's buffer. In modern routers the lookup for destination address and the storing (switching) of the packet into the appropriate memory location is performed by processors input line cards.

4.2 Switching via Bus: Input port transfers packet directly to the output port over a shared bus, without intervention by the routing processor. As the bus is shared only one packet is transferred at a time over the bus. If the bus is busy

the incoming packets has to wait in queue. Bandwidth of router is limited by shared bus as every packet must cross the single bus.

4.3 Switching via Interconnection Networks: In cross-bar switching networks input and output ports are connected by horizontal and vertical buses. If we have N input ports and N output ports it requires 2N buses to connect them. To transfer a packet from the input port to corresponding output port, the packet travels along the horizontal bus until it intersects with vertical bus which leads to destination port. If vertical is free the packet is transferred. But if vertical bus is busy because of some other input line must be transferring packets to same destination port. The packets are blocked and queued in same input port.

5. Figures and Tables

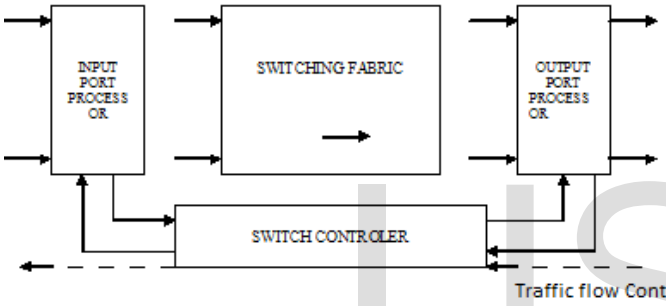


Figure 1: Basic Router Structure

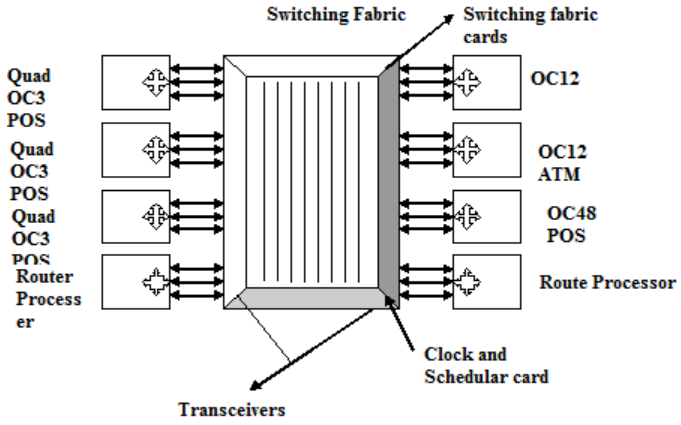
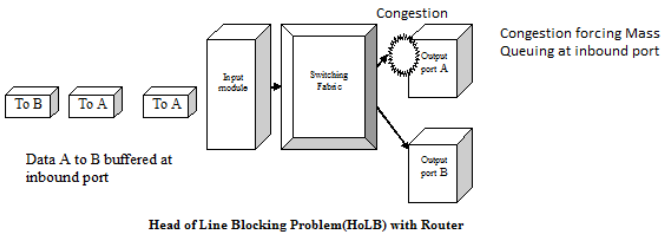


Figure 2: Advanced Switching Fabric Design



Head of Line Blocking Problem(HoLB) with Router

Figure 3: Head of Line Blocking Problem(HoLB) with Router

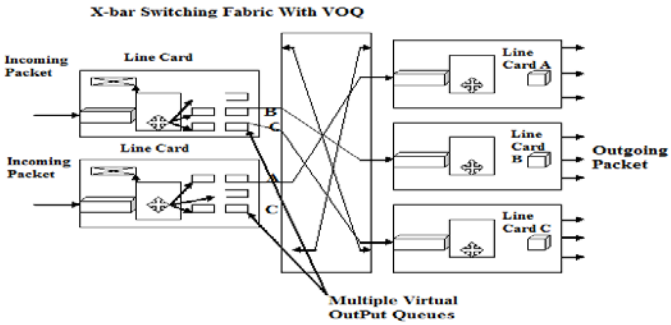


Figure 4: X-bar Switching Fabric with VOQ(Multiple)

5. Soluotions/Suggestion

Switch fabric design includes innovative approaches resulting in a highly efficient system. The switch fabric uses the following key components to provide a highly efficient carrier class and scalable design: Virtual output queues per line card to eliminate head of line blocking, An efficient scheduling algorithm in place of the traditional round robin approach to improve fabric efficiency, Hardware-based replication for multicast traffic; supports partial fulfillment to provide a highly efficient platform for multicast traffic, Pipelining to improve switch fabric performance [4]. Internet Router uses a unique multi-queue implementation to eliminate Head of Line Blocking. As packets arrive into the line card, they are arranged into one of multiple output queues categorized by slot, port, and Class of Service (CoS). These queues are referred to as virtual output queues (VOQs).

In the figure: 4, Virtual Output Queue (A) represents line card A, VOQ B represents line card B, and so on. Each packet is sorted and placed in the proper VOQ. The sorting and placement in the VOQ are based on the forwarding information contained in the Express Forwarding (EF) table.

The following figure shows how the VOQ approach avoids the HoLB problem. As the figure indicates, packet placement minimizes the HoLB problem. Even if a series of packets is being sent to one line card, the other packets in the different VOQs can be sent across the switching fabric, avoiding the classic HoLB problem.

The switching fabric is also designed for next-generation applications, which use IP multicast. The switching fabric overcomes the traditional problems associated with IP multicast by: Using special hardware that performs intensive replication of IP packets on a distributed basis (in the fabric and line card), Dedicating separate queues (VOQs) for multicast traffic, and so that other unicast traffic is not impacted, Allowing for the creation of partial multicast segments. [16] An interface can send both multicast and unicast requests to the switch fabric. When a multicast request is sent, it specifies all destina-

tions for the data and the priority of the request. The CSC handles multicast and unicast requests together, giving precedence to the highest priority request, whether unicast or multicast.

When a multicast request is received, a request is sent to the Clock Scheduler Card. Once a grant is received from the CSC, the packet is then forwarded to the switch fabric. The switch fabric makes copies of the packet and sends the copies to all destination line cards simultaneously (during the same cell clock cycle). Each receiving line card makes additional copies of the packet if it must be sent to several ports. In order to reduce blocking, the switching fabric supports partial allocation for multicast transmissions. This means that the switching fabric performs the multicast operation for all available cards. If a destination card is receiving a packet from another source, the multicast process is continued in subsequent allocation cycles.[5]

These new enhancements avoid the bandwidth-wasting obstacles inherent in first generation crossbar switching fabrics, and enable Router to deliver a switching fabric that achieves a very high level of switching efficiency without sacrificing reliability. In other hand the switching fabric supports full-duplex operation, supplemented by advanced pipelining techniques. Pipelining allows the switch fabric to start allocating switch resources for future cycles before it has completed transmission of data for previous cycles. By eliminating dead time (wasted clock cycles), pipelining dramatically improves the overall efficiency of the switch fabric. Pipelining enables high performance in the switching fabric, allowing it to reach its theoretical maximum throughput.[7][Figure :4]

6.CONCLUSION

IP routers are in the midst of great change, due to technology push with the demanding higher bandwidth, greater reliability, lower cost, greater flexibility, and ease of configuration and In order to resolve the fundamental tussle of non-trivial issue (Advanced Switching Fabric). We propose that customized route computation and different path selection mechanism to coexist, and evolve overtime.

While these advances have solved some difficult problems important issues still remain unresolved. We believe that understanding the stability of a network router is a critical issue, will be a challenge for router designers in years to come.

7. ACKNOWLEDGMENT

I take this opportunity to express my profound gratitude and deep regards to my supervisor Dr. Bharat Mishra, Reader, M.G.C.G.V Chitrakoot, Satna M.P for his exemplary guidance, monitoring and constant encouragement throughout the course of this research papaer. The blessing, help and guidance given by him time to time shall carry me a long way in the journey of life on which I am about to embark. I also take this opportunity to express a deep sense of gratitude to M.G.C.G.V Chitrakoot Satna M.P for their cordial support, valuable information and guidance, which helped me in completing this task through various stages.

8. REFERENCES

[1] D. S. Alexander, M. Shaw, S. M. Nettles and J. Smith, Active Bridging, Proceedings of ACM SIGCOMM'97, Cannes, September 1997.

[2] S. Keshav, An Engineering Approach to Computer Networking, Addison-Wesley, 1997.

[3] J. Anderson and S. Abraham, .Performance-Based Constraints for Multi-dimensional Networks,. IEEE Trans. on Parallel and Distributed Systems, vol. 11, pp. 21-35, Jan. 2000.

[4] W. Dally, .Scalable Switching Fabrics for Internet Routers,. White Paper for Avici Systems Inc., URL . http://www.avici.com/technology/whitepapers/TSRfabric- WhitePaper.pdf, 2002.

[5] D. Dias and J. Jump, .Packet Switching Interconnection Networks for Modular Systems,. Computers, vol. 14, pp. 43- 53, Dec. 1981.

[6] M. Galles, .Spider: A High-Speed Network Interconnect,. IEEE Micro, vol. 17, pp. 34-39, Jan./Feb. 1997.

[7] R. Jain, The Art of Computer Systems Performance Analysis. John Wiley & Sons, New York, 1991, Ch. 31.5, pp. 540-544.

[8] Pluris, Inc., .Tech Briefs: Multistage Routing,. URL . http://www.pluris.com/terabit/ techbriefs/, 2001.

[9] N. Tzeng and M. Mandviwalla, .Cost-Effective Switching Fabrics with Distributed Control for Scalable Routers,. Proc. 22nd IEEE Int.l Conf. on Distributed Computing Systems, July 2002, pp.65-73.

[10] K. Yun, .A Terabit Multi-Service Switch,. IEEE Micro, vol. 21, pp. 58-70, Jan./Feb. 2001.

[11] S. M. Ballew, Managing IP Networks with Cisco Routers, O'Reilly 1997.

[12] S. Floyd and V. Jacobson, Random Early Detection Gateways for Congestion Avoidance, IEEE/ACM Transactions on Networking, August 1993.

[13] P. Goyal, Packet Scheduling Algorithms for Integrated Service Networks, PhD Thesis, UC Texas Austin, 1997.

[14] P. Goyal, H. Vin, and H. Chen, Start-time Fair Queueing: A Scheduling Algorithm for Integrated Services Packet Switching Networks, Proceedings of ACM SIGCOMM '96, August 1996.

[15] S. Lin and N. McKeown, A simulation of IP Switching, Proceedings of ACM SIGCOMM'97, Cannes,September 1997.

[16] C. Labovitz, G. R. Malan and F. Jahanian, Internet Routing Instability, Proceedings of ACM SIGCOMM'97,Cannes, September 1997.

[17] Y. Oie, T. Suda, M. Murata, D. Kolson, and H. Miyahara, Survey of Switching Techniques in High-Speed Networks and Their Performance, Proceedings of IEEE INFOCOM>90, June 1990, pp. 1242-1251.

[18] I. Castineyra, N. Chiappa, and M. Steenstrup. The Nimrod Routing Architecture. RFC 1992, 1996.

[19] Nick Feamster, Hari Balakrishnan, Jennifer Rexford, Aman Shaikh, and Jacobus van der Merwe. The Case for Separating Routing from Routers. In Proc. Future Directions in Network Architecture, August 2004.

[20] T. Roscoe, S. Hand, R. Isaacs, R. Mortier, and P. Jardetzky. Predicate Routing: Enabling Controlled Networking. In Proc. Workshop on Hot Topics in Networking, 2002.

[21] E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol Label Switching Architecture. RFC 3031, January 2001.

[22] J. Turner, T. Anderson, L. Peterson, and S. Shenker. Virtualizing the Net: A strategy for network desi_cation. http://www.arl.wustl.edu/.jst/talks/hotI-9-04.pdf. March 2005.

[23] Davies, Shanks, Heart, Barker, Despres, Detwiler, and Riml, "Report of Subgroup 1 on Communication System", INWG Note #1 May 2011.