Report Number: WUCSE-2006-9

2006-01-01

Design of Routers for Diversified Networks

Jonathan Turner

2006-9

Design of Routers for Diversified Networks

Authors: Turner, Jonathan

Corresponding Author: jon.turner@wustl.edu

Web Page: http://www.arl.wustl.edu/~jst/

Design of Routers for Diversified Networks[*]

Jonathan Turner
Washington University
jon.turner@wustl.edu

1. Introduction

There is a growing recognition in the networking research community, that the protocols and services at the heart of the Internet have become so rigid and difficult to change, that they represent an impediment to the continuing evolution of the Internet. This poses a serious problem as the demands on the Internet continue to grow, making the limitations of current systems more and more evident. *Network diversification* [AN05] has been advanced as a tool for addressing the current impasse by creating an environment in which multiple diverse networks can co-exist within a common infrastructure. This makes it possible for new network architectures and services to be deployed on a global scale alongside incumbent network architectures, allowing them to compete on their own merits. (We use the term "diversification" in place of the more common "virtualization" since the latter term has been commonly used to describe systems that allow separate networks based on a *common protocol and service model* to operate over shared links. Network diversification, by contrast, enables networks with distinctly different protocols and service models to operate on a shared substrate comprising both links and routers.)

This report explores the implications of the diversified networking concept on the design of routers, and outlines a strategy for creating such systems using board level subsystems that are now being developed by a number of companies, in accordance with the emerging ATCA standards for telecommunications equipment. We then discuss how these systems could be used within a national networking research testbed, that is now in the planning stage.

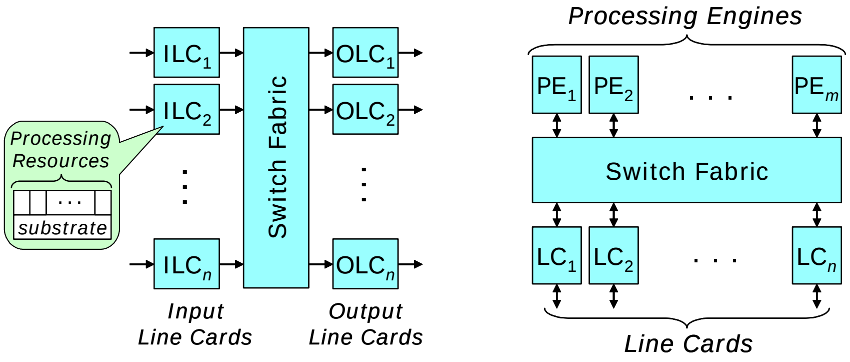
2. Design Issues for Diversified Routers

In a diversified network, a router hosts multiple *meta-routers*, belonging to distinct *meta-networks*. The diversified router provides resources that can be used by the different meta-routers and the underlying mechanisms to allow each meta-router to operate independently of the others, without interference. The objective is to allow each meta-router to function as though it were operating over a physically dedicated network, rather than on a shared substrate. The design of a diversified router is distinctly different from the design of conventional routers in that it requires the availability of generic processing resources that can be allocated flexibly to different meta-routers, and it requires mechanisms to allow the different meta-routers to co-exist without interference.

A conventional router consists of three major components, *line cards*, *switching fabric* and *control processor*. The line cards terminate the physical links and implement the specific protocol processing functions that define a particular network. On input, this typically includes performing a table lookup of some sort, to determine where the packet should be sent next and what special processing (if any) it should receive. On output, it typically includes scheduling the packet for transmission on the outgoing link, either using a simple FIFO queue or some more complex queueing subsystem. The switching fabric is responsible for transferring packets from the line cards where they arrive, to the line cards for their outgoing links. Switching fabrics are typically designed to be *nonblocking*, meaning that they should handle arbitrary traffic patterns, without subjecting packets to delays significantly larger than the intrinsic queueing delay implied by the rate of the output links. The control processor in a conventional router implements various control and administrative functions (such as executing routing protocols and updating tables in the line cards). These functions are generally implemented in software running on a general-purpose microprocessor.

Consideration of a conventional router architecture leads naturally to a diversified router architecture in which the line card is replaced by a *diversified line card* that consists of a *substrate* and generic processing resources that can be assigned to different meta-line cards (see Figure 1). The substrate is responsible for configuring the generic processing resources so that different meta-line cards can co-exist without interference. On receiving the packet from the physical link, the substrate first determines which meta-line card the packet should be sent to and delivers it

[*] This work supported by the National Science Foundation (CNS 0435173).



Diversified Line Card Architecture *Processing Pool Architecture*

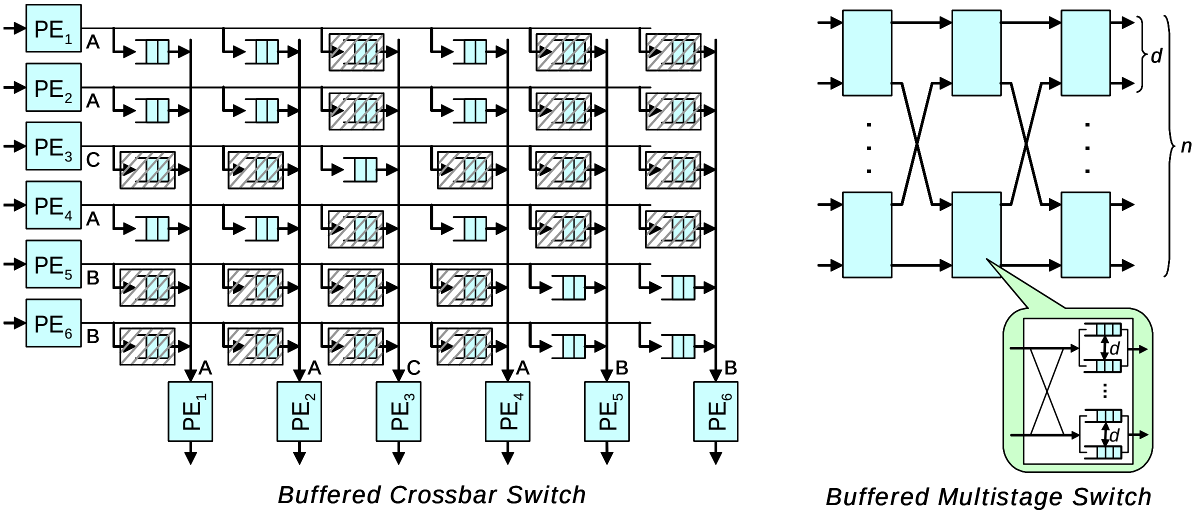
to that meta-line card. (The external packet format used by the diversified networking system must provide a way to allow the substrate to make this determination, but we are not concerned here with the specific mechanism used.) Meta-line cards can pass packets back to the substrate, which forwards them through the shared switch fabric, on input, or to the outgoing link, on output.

One issue with this architecture concerns how to provide generic processing resources at a line card, in a way that allows the resources to be shared by different meta-line cards. Conventional line cards are often implemented using Network Processors (NP), programmable devices that include high performance IO and multiple processor cores to enable high throughput packet processing. It seems natural to take such a device and divide its internal processing resources among multiple meta-line cards. For example, an NP with 16 processor cores could be used by up to 16 different meta-line cards, by simply assigning processor cores. Unfortunately, current NPs are not designed to be shared in this way. All processing cores share access to the same physical memory (there are no built-in mechanisms for memory protection), making it difficult to ensure that different meta-line cards don't interfere with one another. Also, each processor core has a fairly small program store. This is not a serious constraint in conventional applications, since processing can be pipelined across the different cores, allowing each to store only the program it needs for its part of the processing. However, a processor core implementing an entire meta-line card must store the programs to implement all the processing steps for that meta-line card. The underlying issue raised by this discussion is that efficient implementation of an architecture based on diversified line cards, requires components that support *fine-grained diversification*.

An alternative architecture for a diversified router separates the processing resources used by the meta-routers from the physical link interfaces. This allows a much more flexible allocation of processing resources and greatly reduces the need for fine-grained diversification. This architecture, which is also illustrated in Figure 1, provides a pool of *Processing Engines* (PE), that are accessed through the switch fabric. The line cards that terminate the physical links forward packets to PEs through the switch fabric, but do not do any processing that is specific to any particular meta-network. There may be different types of PEs, including some implemented using network processors, and others implemented using conventional microprocessors. The former are most appropriate for high throughput packet processing, the latter for control functions that require more complex software. A meta-router may be implemented using a single PE or multiple PEs. In the case of a single PE, packets will pass through the switch fabric twice, once on input, once on output. In a meta-router that uses multiple PEs to obtain higher performance, packets may have to be passed through the switch fabric a third time.

The primary drawback of the *processing pool architecture* is that it requires multiple passes through the switch fabric, increasing the delay that packets are subjected to and increasing the switch fabric capacity needed to support a given total IO bandwidth. The increase in delay is not a serious concern in wide area network contexts, since switch fabric delays are typically 10 μs or less. The increase in the switch fabric capacity does add to system cost, but since a well-designed switch fabric represents a relatively small part of the cost of a conventional router (typically 10-20%), we can double, or even triple the switch fabric capacity without a proportionally large increase in the overall system cost.

The great advantage of the processing pool architecture is that it largely eliminates the need for fine-grained diversification. In a large system with tens or hundreds of PEs (a 1 Tb/s router would require at least 200 PEs, using current technology), it's reasonable to assign entire PEs or groups of PEs to different meta-routers. Meta-routers that require less than one PE's worth of processing can still be accommodated by implementing them on a general



Buffered Crossbar Switch

Buffered Multistage Switch

Figure 2. Traffic Isolation in Switch Fabrics

purpose microprocessor, running a conventional operating system that supports a virtual machine environment. The processing pool architecture also makes it easy to adjust the system's processing capacity independently of the IO bandwidth, something that is more difficult to do with the diversified line card architecture.

In system contexts where the majority of traffic is associated with a single "default" meta-router, it makes sense to adopt a *hybrid architecture* in which line cards provide the processing resources for the default meta-router, while other meta-routers are implemented using PEs from a shared pool. In a system in which the default meta-router accounts for half of the total traffic, this can reduce the required switch fabric capacity by at least one-third, relative to a pure processing pool architecture.

In all three diversified router architectures, the switch fabric must be designed to be shared by different meta-routers. In particular, it should not be possible for any single meta-router to interfere with the flow of traffic from another meta-router. One way to ensure this is to constrain the traffic flows entering the switch fabric so as to eliminate the possibility of internal congestion. This is difficult to do in all cases. In particular, meta-routers consisting of multiple PEs should be allowed to use their "share" of the switch fabric capacity in a flexible fashion, without having to constrain the pair wise traffic flows among the PEs. However allowing this flexibility makes it possible for several PEs in a given meta-router to forward traffic to another PE at a rate that exceeds the bandwidth of the interface between the switch fabric and the destination PE.

There is a straightforward solution to this problem in the processing pool architecture. To simplify the discussion, we separate the handling of traffic between line cards and PEs from the traffic among PEs in a common meta-router. In the first case, we can treat the traffic as a set of point-to-point streams that are rate-limited when they enter the fabric. Rate-limiting these flows follows naturally from the fact that they are logical extensions of traffic flows on the external links. Because the external link flows must be rate limited to provide traffic isolation on the external links, the internal flows within the switch fabric can be configured to eliminate the possibility of congestion.

For PE-to-PE traffic, we cannot simply limit the traffic entering the switch, since its important to let PEs communicate freely with other PEs in their same group, without constraint. However, because entire PEs are allocated to meta-routers in the processing pool architecture, it's possible to obtain good traffic isolation in a straightforward way, for this case as well. We illustrate this in Figure 2 for two different switch fabric architectures. The first uses a buffered crossbar and divides the six PEs among three meta-routers, identified by the letters, *A, B,* and *C*. Each crosspoint has a configurable *enable* bit that allows the PE in its row to send to the PE in its column. If these bits are configured to allow only the traffic flows among the desired sets of PEs, each of the meta-routers can operate as though it has a dedicated crossbar of its own (in the diagram, the shaded boxes identify crosspoints that are disabled). The second architecture uses a more scalable three stage network, with buffered switch elements, similar to those used in large, conventional routers, such as Cisco's CRS-1 [CSCO]. Traffic entering the switch fabric from a PE belonging to one meta-router can be sent only to PEs in the same meta-router. This can be easily enforced at the switch fabric input. The switch elements in the first stage distribute traffic evenly across the switch elements in the second stage to balance the load. Each of the second stage switch elements implements *d* separate queues at each of its output links, where *d* is the number of output ports of the third stage switch elements (typically 32 or 64). This allows the second stage switches to isolate the traffic flows going to different outputs of the overall network, so that they cannot interfere with one another. Since PEs are assigned to specific meta-routers, this level of traffic isolation is sufficient to ensure that no meta-router can interfere with the traffic for another meta-router. Using current technology, this architecture can scale to several thousand ports of 10 Gb/s each, with as much as one megabyte of buffering available to each PE.

3. Implications of an Emerging Standard for Networking Research

The *Advanced Telecommunications Computing Architecture* (ATCA) is a rapidly developing set of standards designed to facilitate the development of carrier-class communications and computing systems [PCMG]. ATCA defines standard physical components and some standard patterns for how to use those components to construct high performance systems. It has attracted broad industry support and is expected to lead to the development of a range of inter-operable subsystems that will allow more cost effective and flexible development of new communication systems.

ATCA has important implications for the networking research community. Networking researchers interested in creating new network architectures and services have long had to content themselves with implementing experimental networks using commodity PCs. Commercial routers have been difficult to use in research contexts, because vendors have been unwilling to allow researchers to have access to the technical details needed to perform experiments and make changes. ATCA is creating an intermediate market for router subsystems that can be assembled into powerful, carrier-class communication systems. Subsystem vendors design their products to be highly flexible to enable their use by multiple system vendors. This is creating an unprecedented opportunity for the networking research community. We now have the tools to create high performance research systems that are built on a hardware platform that is directly comparable to the best commercial systems.

Figure 3 shows a standard 14 slot ATCA chassis with backplane, power distribution system and cooling fans. Such chasses are now available from several vendors. The backplane includes standard signals for clock distribution and low level system management. It also defines *fabric connections* that implement several interconnection topologies for high speed inter-board communication (differential signal pairs suitable for 2.5 Gb/s data rates). ATCA standardizes key aspects of the boards that are used with the chassis, including physical size, connector type and placement and the use of certain of the connector signals. It also defines standards for mezzanine cards that can be optionally used with an ATCA base card. In addition, it defines standards for optional *Rear Transition Modules* (RTM) which are small cards that are inserted into the back side of the chassis and are can be used for interconnecting multiple chasses in larger systems. These elements of the standard are also shown in Figure 3.

Figure 4 shows examples of ATCA subsystems that are now starting to appear. The first example (at left) is the Radisys 7010, a network processing blade that contains two IXP 2800 network processors [RA04]. Each NP has sixteen internal processor cores for high throughput data processing, plus an Xscale processor (typically running

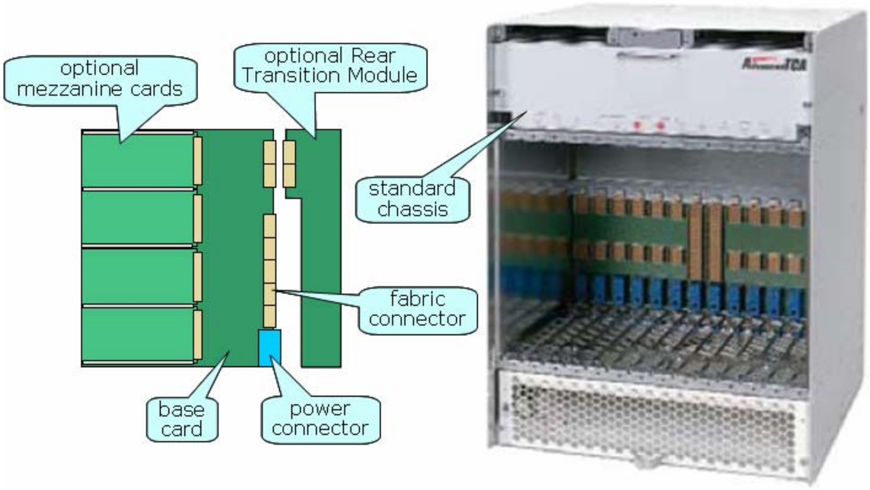


Figure 3. Selected ATCA Components

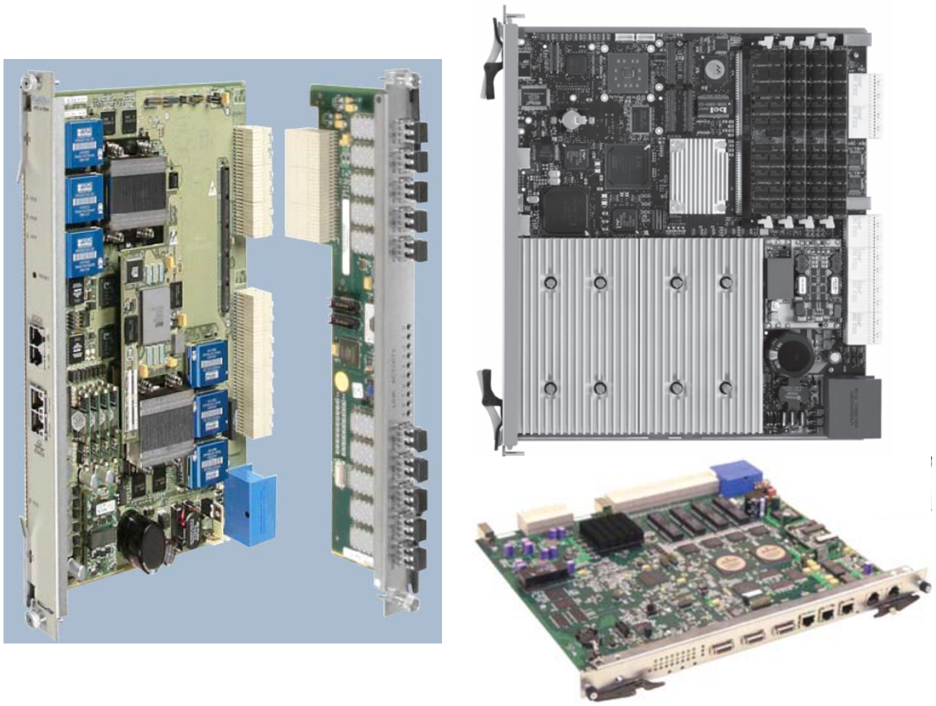


Figure 4. Sample ATCA Boards: Radisys 7010 network processor blade with RTM, Motorola 7120 Dual Xeon compute blade and Diversified Technologies Infiniband switch blade

Linux) for control. Each NP has three banks of RDRAM providing 750 MB of storage and three banks of QDR SRAM. The two NPs also share access to a dual-port TCAM that can be used for packet classification and other applications requiring associative lookup. The Radisys board supports RTMs that provide external IO connections. The figure shows an RTM with 10 fiber gigabit Ethernet interfaces. The upper right portion of the figure shows a compute blade from Motorola that includes two Xeon processors that implement a shared memory multiprocessor. The bottom right portion of the figure shows a switch blade from Diversified Technologies that includes an Infiniband switch with 10 Gb/s ports. In a typical application, two such switch blades would be used in a chassis to provide switching among twelve other cards. Other switch types are also available. In particular, switch boards that support 10 Gb Ethernet ports (with multi-spanning tree VLAN support) are expected within the next year from multiple vendors.

4. A Diversified Router for a National Network Testbed

An NSF-sponsored workshop early this year recommended that NSF support the development of a national testbed for research on new network architectures [WNSF]. The workshop report envisions a testbed that can support multiple networks sharing a common infrastructure and capable of supporting traffic from a large number of real users (tens to hundreds of thousands). Implementing such a testbed will require high performance diversified routers capable of hosting the meta-routers needed to implement multiple experimental networks. The developing ATCA market now makes it possible to assemble such a system from production-quality subsystems, allowing the networking research community to create a research platform that can deliver the kind of performance and reliability needed to attract the large numbers of real users necessary to credibly demonstrate and evaluate new network architectures and services.

To be successful, a diversified router needs to make it reasonably straightforward for networking researchers to implement meta-routers for new network architectures. One way to minimize the hurdles faced by researchers is to leverage existing models, such as PlanetLab [CH03] for implementing experimental networked systems. A user of a diversified network testbed could be assigned a virtual machine on a general purpose processor just as in PlanetLab today. For users who do not require high performance, such a virtual machine may provide all the resources necessary. Those users who require higher performance can also be assigned one or more Network Processor subsystems and the switching resources needed to connect these to one another and to the external links. For such

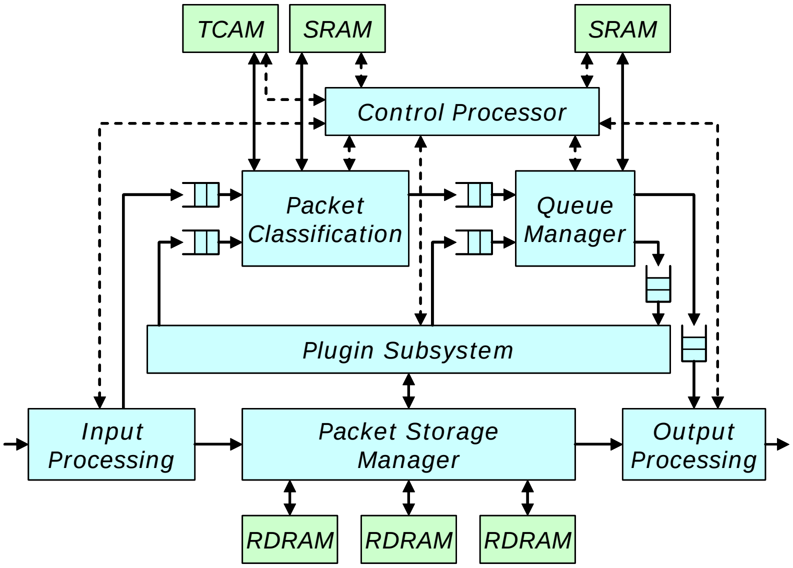


Figure 5. Generic Meta-router

users, the virtual machine can serve as a "staging area" from which to download software to the assigned network processors, configure them and monitor their operation.

Getting good performance from NPs requires taking full advantage of the parallelism of the underlying hardware and the architectural features provided to boost performance. This makes programming NPs considerably more challenging than programming a conventional processor. To enable networking researchers to use NPs effectively, it will be necessary to make them easier to use. There have been a number of efforts in recent years designed to make NP software development easier. The Shangri-La project is an ambitious effort that seeks to make programming NPs as easy as programming conventional workstations. In includes "(1) a domain-specific programming language for specifying packet processing applications, (2) a compiler that incorporates profile-guided techniques for mapping packet-processing applications onto complex packet-processing system architectures, and (3) a run-time system that dynamically adapts resource allocations to create systems that are robust against attacks and that optimize performance and power consumption for the current network conditions." [VI04] Shangri-La targets the IXP 2400 network processor, making it directly applicable to the proposed diversified router. Another approach has been taken in [SH03] which created a version of the popular Click toolkit for the IXP 1200, an earlier generation network processor. This is an attractive approach, since it allows researchers to work within the context of a basic conceptual model with which they are already familiar.

A third way to address this issue is to provide a generic router implementation, that can be easily adapted for different meta-routers. This approach is illustrated in Figure 5. In this design, the *Input Processing* block examines the headers of arriving packets, and passes the packet itself to the *Packet Storage Manager* (PSM), which stores the packet in off-chip DRAM, while passing a buffer pointer, a packet length and a *Lookup Vector* (LV), to the *Packet Classification* (PC) block, which uses the LV to perform a TCAM lookup. The result of the TCAM lookup is a queue identifier and a *Results Vector* (RV), which is passed through the *Queue Manager* (QM) without interpretation. The Packet Classification block uses an external SRAM to store statistics about packets received that matched specific filters. The *Output Processing* block uses the buffer pointer received from the QM to retrieve the packet from the PSM and uses the RV to select the outgoing meta-link the packet should be forwarded on and possibly make any needed changes to the packet header. The *Plugin Subsystem* provides an environment for software modules that implement special features. Filters, in the TCAM can direct packets to specific plugins, which can modify the packets (by directly accessing the memory where they are stored) and can re-insert packets into outgoing queues, or send them back through the PC block. The Control Processor (CP) can be implemented using the Xscale processor in the IXP 2400. It can add filters to the TCAM, monitor the collected statistics and configure the other elements of the architecture.

Notice that in this architecture, the PC block, the QM and PSM are completely generic. They have no dependence on the actual packet formats, so they can be used directly in a wide variety of different meta-routers. Indeed, much of the other blocks can also be made largely protocol independent, making it possible for researchers to implement new protocols by making relatively small changes to a few elements of the architecture. Those interested in using IP as a base protocol for their work will typically be able to accomplish their objectives through the addition of plugins and modifications to the software on the CP, allowing the rest of the infrastructure to be left unchanged. We believe that this approach can dramatically reduce the learning curve for networking researchers, allowing them to make effective use of high performance NPs without an excessive amount of effort.

5. Closing Remarks

This paper outlines a strategy for developing a diversified router that could form a central component of a national networking research testbed aimed at enabling the evaluation and experimental deployment of new network architectures and services. We have observed that the newly developed ATCA standard is creating a new market for high quality router subsystems and we have argued that this is leading to an unprecedented opportunity for the networking research community. It is now possible to assemble high quality, open, network research platforms that are directly comparable to the best commercial systems. This will allow the research community to demonstrate innovative new network architectures on a large scale and in a much more compelling way than is possible with PC-based routers. Because these systems can be built using programmable network processor boards, it is feasible for even small groups of networking researchers to develop novel meta-routers capable of supporting large traffic flows and large numbers of users.

References

[AN05] Anderson, Tom, Larry Peterson, Scott Shenker and Jonathan Turner. “Overcoming the Internet Impasse through Virtualization,” *Computer Magazine*, 4/05.

[CH03] Chun, Brent, David Culler, Timothy Roscoe, Andy Bavier, Larry Peterson, Mike Wawrzoniak, and Mic Bowman. “PlanetLab: An Overlay Testbed for Broad-Coverage Services,” *ACM Computer Communications Review*, 7/03.

[CSCO] Cisco Systems. “Next Generation Networks and the Cisco Carrier Routing System,” white paper, available at `http://www.cisco.com/warp/public/cc/pd/rt/12000/clc/prodlit/reqng_wp.pdf`, 2004.

[PCMG] PCI Industrial Computer Manufacturers Group. “AdvancedTCA Specifications for Next Generation Telecommunications Equipment ,” available at `http://www.picmg.org/newinitiative.stm`.

[RA04] Radisys Corporation. “Promentum™ ATCA-7010 Data Sheet,” product brief, available at `http://www.radisys.com/files/ATCA-7010_07-1283-01_0505_datasheet.pdf`.

[SH03] Shah, Niraj, William Plishker, Kurt Keutzer. “NP-Click: A Programming Model for the Intel IXP1200,” In the *Workshop on Network Processors & Applications – NP2*. Held in conjunction with *The 9th International Symposium on High-Performance Computer Architecture*, 2/03.

[VI04] Vin, Harrick, Jayaram Mudigonda, Jamie Jason, Erik J. Johnson, Roy Ju, Aaron Kunze, and Ruiqi Lian. “A Programming Environment for Packet-processing Systems: Design Considerations.” In the *Workshop on Network Processors & Applications - NP3*. Held in conjunction with *The 10th International Symposium on High-Performance Computer Architecture*, 2/04.

[WNSF] Report of NSF Workshop on Overcoming Barriers to Disruptive Innovation in Networking. Available at `www.arl.wustl.edu/noBarriers`, 5/05.