Guest Editorial
Active and Programmable Networks

DRIVEN BY advances in underlying technologies, increasing acceptance of computing and middleware paradigms in telecommunication networks, and a need to accelerate network innovation, network researchers are exploring new ways for network routers, switches, and base stations to be dynamically programmed by network applications, users, operators, and third parties. Network elements such as program controlled-switches have been programmable for some time but not in the dynamic manner proposed by the current work. Active and programmable networks seek to exploit advanced software techniques and technologies in order to make network infrastructure more flexible, thereby allowing users and service providers to customize network elements to meet their own specific needs. Customizing routing, signaling, resource allocation, and accelerating information processing in this manner raises a number of significant security, reliability and performance issues.

Results from this field of research have the potential for broad impact on customers, service providers, and equipment vendors across a range of telecommunication sectors, including broadband, mobile, and IP networks. Competition among existing and future Internet service providers (ISPs) may hinge on the speed at which one service provider can respond to new market demands over another. The introduction of new services is a challenging task requiring new tools for service creation, including new network programming platforms and supporting technologies. Future programmable networks are likely to be based on active networking and open signaling techniques. Both of these proposals squarely address the same problem: how to "open up" the network and accelerate its programmability in a controlled and secure manner for the deployment of new architectures, services, and protocols.

The separation of communication hardware (switching fabrics, forwarding engines) from control software is fundamental to making the network more programmable. Such a separation is difficult to realize today. The reason for this is that switches and routers are vertically integrated—akin to mainframes of the 1970s. Typically, service providers do not have access to switch/router control environments (e.g., the router's operating system), algorithms (e.g., routing protocols), or states (e.g., flow states). This makes the deployment of new network services, which may be many orders of magnitude more flexible than proprietary control systems, impossible due to the closed nature of network nodes.

The work on open signaling (OPENSIG) exemplifies the development of new network programming environments that explicitly recognize service creation, deployment, and management in the network infrastructure. Here, there is a clear distinction between the transport and control, and the objective is to make the control plane fully programmable. The work on active networks grows from the idea of allowing the traffic itself to program the network's behavior—an idea that is potentially applicable in various dimensions of programmability, for example in-band versus out-of-band or per-packet, per-flow, or per-network granularity. In its most general form, executable programs are embedded within data packets (called "capsules") and executed by network elements as they traverse the network.

Programmable networks have evoked substantial interest in the networking community, standards bodies and industry. Standards initiatives include the IEEE 1520 Programmable Interfaces for Networks, Multiservice Switching Forum, and IETF GSMP and FORCES initiatives on application programming interfaces(APIs) for routers. A growing number of international forums have arisen for presentation of advances in active and programmable networks, including OPENSIG, IEEE Conference on Open Architectures and Network Programming (OPENARCH), and IFIP International Working Conference on Active Networks (IWAN).

For this issue on active and programmable networks, we received well over 40 papers, of which 14 papers were selected for publication. The papers selected for this issue identify new directions and challenges spanning active networking and open signaling, and include: hardware design considerations, Internet programmability, network resource management, node resource management, network programming models and systems, and service provider perspectives.

One fundamental question raised by active networks is how much processing can practically be applied to each packet. Because transmission bandwidth has been growing faster than processing bandwidth (i.e., Moore's Law is losing), active switch/router designs must rely on other means of scaling processing power. The paper by Wolf and Turner explores the consequences of this observation for the design of hardware for processing packets at wire speeds of 2.4 Gb/s and higher. In addition to a design, the authors present benchmarking results characterizing the type and quantity of processing required for some typical network operations.

Two papers deal with programmability in the specific context of the Internet. The flexible intra-AS routing environment, described in the paper by Partridge *et al.*, supports the customization of routing through the specification of sophisticated routing metrics. The Concast service, described by Calvert *et al.*, is an IP-compatible "inverse multicast" framework that can be customized by applications, which specify the semantics of merging operations to be carried out by the network.

Resource management is both a challenge and a potential application of active and programmable networks. Considering resource allocation at the network level, Chandra *et al.* argue the need for extensibility of end-to-end resource sharing services

Publisher Item Identifier S 0733-8716(01)02000-5.

for various applications, and describe their control architecture for supporting such extensions. Kiwior and Zabele present simulation results illustrating the benefits of strategic placement of active functionality. Isaacs and Leslie present a set of control-plane mechanisms that support the specification and implementation of resource-assured virtual private networks.

At the level of individual nodes, a number of researchers have investigated the problem of providing robust and efficient platforms for the processing of active packets. In the DARPA active networks program, the node operating system provides a layer of abstraction between the underlying hardware and the execution environments, which export the programming interfaces on which services are developed. Peterson *et al.* describe this interface and give an overview of some implementations currently in progress. The paper by Tullman *et al.* presents Janos, a third, Java-oriented implementation of the NodeOS. The paper by Yau and Chen describes the CROSS system, which integrates QoS-aware scheduling for multiple types of resources including CPU time, transmission bandwidth, and state-store capacity.

The topic of network programming systems and models (i.e., composition paradigms and programmable building blocks) has been a subject of great debate in the research community. The following three papers present a range of different approaches to network programming. The Genesis Kernel, described in the paper by Kounavis *et al.*, automates the creation, deployment, and management of network architectures. The authors describe the design, implementation and evaluation of a "spawning network." The second paper by Ott *et al.* is motivated by the need to allocate computational resources in the network in an efficient manner. The authors propose the Journey model, which provides computation as an integrated network service to meet service needs. The third paper by da Silva *et al.* presents the NetScript system, which takes a dataflow approach to the construction of new services. The authors discuss their experiences using the NetScript programming language and demonstrate the utility of their system by dynamically extending an IP router to support value-added firewall protection services.

Another key question about active and programmable network technology is how service providers can make money from it. Two papers consider the utility of active networks from an economic standpoint. "Active Routing," by Maxemchuk and Low, considers the benefits of enabling network users to customize routing through capsules that carry routing policies as well as data. "Carrier-Scale Programmable Networks," by Safaei *et al.*, considers the advantages of offering a more general form of computation in the network. Both papers highlight assumptions under which active and programmable networking can be profitable for service providers.

The Guest Editors would like to thank all of the authors who submitted papers to this special issue on active and programmable networks and the reviewers for their time, energy, and comments. It has been a great pleasure to put together this issue and we hope you enjoy it.

KENNETH L. CALVERT, *Guest Editor*
University of Kentucky
Lexington, KY 40506

ANDREW T. CAMPBELL, *Guest Editor*
Columbia University
New York, NY 10027

AUREL A. LAZAR, *Guest Editor*
Columbia University
New York, NY 10027

DAVID WETHERALL, *Guest Editor*
University of Washington
Seattle, WA 98195

RAJ YAVATKAR, *Guest Editor*
Intel Corporation
Hillsboro, OR 97124

Kenneth L. Calvert (S'90–M'90) received the Ph.D. degree from the University of Texas, Austin, TX in 1991.

He is an Associate Professor with the Department of Computer Science, University of Kentucky, Lexington, KY. He has been active in the Active Networking community, among other roles as the Editor of the "*Architectural Framework for Active Networks*" document and as Technical Program Chair of the Second IEEE Conference on Open Architectures and Network Programming (OPENARCH '99). His current research interests include active networks, secure multicast, and abstract models of network topology. From 1979 to 1984, he was with Bell Telephone Laboratories, Holmdel, NJ, and from 1991 to 1998, he was with the College of Computing, Georgia Institute of Technology, Atlanta, GA.

Andrew T. Campbell (M'96) received the Ph.D. degree in computer science in 1996.

He is an Assistant Professor with the Department of Electrical Engineering, Columbia University, New York, and a Member of the COMET Group with the Center for Telecommunications Research, Columbia University, New York. His areas of interest encompass programmable networks, mobile networking and QOS research. He is currently the Co-Chair for the IEEE Conference on Open Architectures and Network Programming (OPENARCH 01) and has been active in OPENSIG.

Dr. Campbell received the NSF CAREER Award for his research in programmable mobile networking in 1999.

Aurel A. Lazar (S'77–M'80–SM'90–F'93) has been a Professor of Electrical Engineering with Columbia University, New York since 1988. His current theoretical research interests are on networking games and pricing. His experimental work focuses on building open programmable networks. This work led to the establishment of the IEEE standards working group on Programming Interfaces for Networks. He was instrumental in establishing the OPENSIG international working group with the goal of exploring network programmability and next generation signaling technology. He was the Program Chair of the Fall and Spring OPENSIG'96 workshops and of the First IEEE Conference on Open Architectures and Network Programming (OPENARCH'98). Currently on leave from Columbia University, he is Chairman and CEO of Xbind, Inc., a high technology start up located in Manhattan's Silicon Alley.

David Wetherall (S'88–M'89) received the Ph.D. degree from the Laboratory for Computer Science, Massachusetts Institute of Technology (MIT), Cambridge, MA in 1998.

He is a Member of the Faculty of Computer Science and Engineering with the University of Washington (UW), Seattle, WA. In addition to his role at UW, Dr. Wetherall is a Founder and Consultant at Asta Networks. He has conducted computer systems research for ten years and authored papers on topics ranging from distributed systems, to internetworking, to programming languages. Dr. Wetherall's thesis research helped to pioneer the field of active networks, in which flexible network infrastructures are used to enable rapid service innovation. While still in Western Australia, Dr. Wetherall worked for QPSX Communications, the company that led the development of the IEEE802.6 standard for high-speed metropolitan area networks.

Raj Yavatkar received the Ph.D. degree in computer science from Purdue University, West Lafayette, IN in 1989.

He is currently the Director of Internet Exchange Architecture with Intel's Network Communications Group, Hillsboro, OR, where he leads Intel's efforts to establish a new architecture for programmable network processors. Previously, Raj led the research and development activities in the areas of programmable networks, policy-based network management, and end2end quality of service (QoS). He initiated and led the definition of a framework for policy-based networking that resulted in both the development of IETF standards (e.g., COPS) and the technology that Intel licensed to HP OpenView and others. At Intel, Raj has played a major role in defining Intel's Internet QoS roadmap and previously led the development of RSVP and a policy-based network management framework. Raj was also a principal contributor to the design of the packet scheduling framework and Generic QoS API for Windows 2000 OS from Microsoft.

Dr. Yavatkar has numerous publications and serves on the technical program committees of leading conferences and workshops such as the ACM Sigcomm, Global Internet, IEEE Infocom, IEEE Multimedia, IEEE HPNC (High-Performance Networking), and ACM Multimedia. He also co-authored the book *Inside the Internet's Resource Reservation Protocol (RSVP)* (Wiley, 1999). Until recently, he was the US Editor for the journal, Computer Communications. He also serves on the editorial board of two journals: ACM/Springer Verlag's Multimedia Systems and Kluwer Academic Press's Multimedia Tools and Applications. At the Internet Engineering Task Force (IETF). He plays significant role in various working groups such as ISSLL, Diff-Serv, Policy, and RAP.