Converging the Evolution of Router Architectures and IP Networks

András Császár, TrafficLab, Ericsson Research
Gábor Enyedi and Gábor Rétvári, Budapest University of Technology and Economics
Markus Hidell and Peter Sjödin, KTH, the Royal Institute of Technology

Abstract

Although IP is widely recognized as the platform for next-generation converged networks, unfortunately, it is heavily burdened by its heritage of almost 30 years. Nowadays, network operators must devote significant resources to perform essential tasks, such as traffic engineering, policy enforcement, and security. In this article, we argue that one of the principal reasons for this is the way control and forwarding planes are interspersed in IP networks today. We review the architectural developments that led to the current situation, and we reason that centralization of network control functionality can constitute a solution to the pressing problems of the contemporary Internet.

Over the past decades, the Internet, and the accompanying Internet Protocol (IP) suite, have evolved by a tremendous progression. In the beginning, the Advanced Research Projects Agency network (ARPANET) was a closed academic network serving as a playground for network research. In our day, this rudimentary network has become a critical communications infrastructure supporting significant economic, educational, and social activities. Today, we deem email and the World Wide Web as elemental parts of our everyday lives, and new killer applications continue to emerge. But rarely do we ponder on the large-scale evolutionary progress that made the Internet what it is today.

One factor that played a key role in this continuous progress is the evolution of the Internet as a global network. The elementary network architecture of the ARPANET, basically a set of packet-forwarding computers interconnected by long-haul links, has grown into a complex network of autonomously operated domains interconnected by a sophisticated inter-domain control infrastructure. This diversification of responsibility enabled service providers to shape their networks according to their own taste, relatively isolated one from another, but due to the end-to-end service model, it also allowed the Internet to continue to function as a whole.

A second enabler of the unparalleled growth of the Internet is the evolution of IP-router architectures. Over the years, the IP router developed from simplistic packet-manipulating software implemented on a general purpose computer to sophisticated network equipment that fully utilizes the capabilities of specialized hardware. It integrates an endless suite of functionality — ranging from raw packet forwarding through traffic shaping, packet queuing, access control, and network address translation (NAT) with connection tracking — all the way to distributed network protocols. Recently, it was proposed to modularize these interspersed functions and organize them into administratively and physically distinct modules, yielding what is called the *distributed router*. Such distributed routers are expected to improve scalability of IP routers, open up new markets for device vendors, and foster rapid innovation in the area.

We see the co-evolution of the *IP-network architecture* and the *IP-router architecture* as a crucial constituent in the unrivaled success of IP and because the Internet is constantly facing new challenges as new applications and content are introduced, we believe that this evolutionary progress will persist in the foreseeable future. Therefore, instead of aiming at a clean-slate redesign of the Internet in response to these challenges (see e.g., the FIND project [1]), we believe that it is more natural to track the continuous evolution of IP-network and -router architectures and consider where current developments are heading.

The most important goal of this article is to present an overview of the ongoing efforts of network systems engineers to advance the design of IP routers and shape the Internet architecture to keep pace with ever-growing expectations. We point out the shortcomings of contemporary network and router architectures and argue that at some point, the architectural developments must converge. In particular, we reason that this point could be the *unification of all the network control and management functionality* in one central administrative unit. Centralized control would ease network management and the deployment of new services; it would allow operators to extract better performance from their network infrastructure; and it would improve the overall security, scalability, and price-efficiency of IP networks as well. We describe state-of-the-art IP-network architectures and also devote a section to router architectures and the forwarding and control element separation (ForCES) framework of IETF, a recent development in this field. We introduce the IP-network architecture based on the centralized control model, discuss how this model solves a large part of the problems of contemporary IP networks, and give a side by side comparison. Finally, we conclude the article.

0890-8044/07/$20.00 © 2007 IEEE

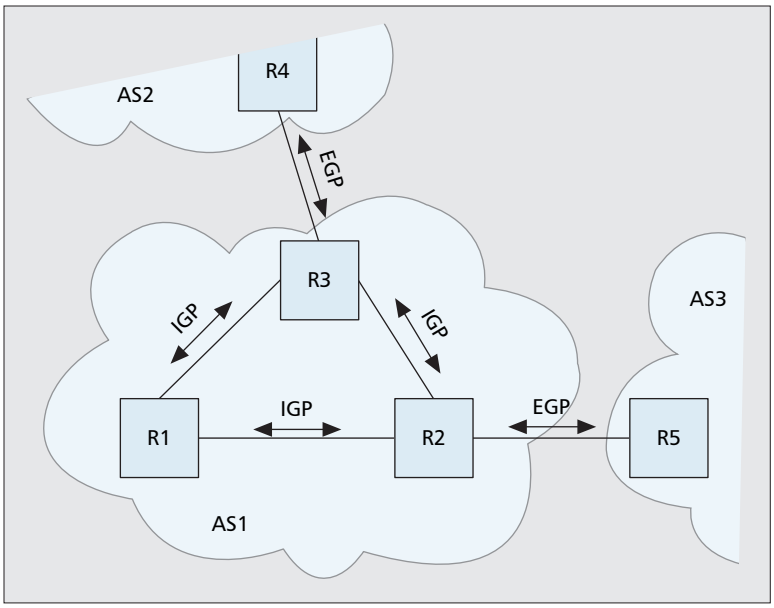


Figure 1. *The state-of-the-art hierarchical IP network architecture with three ASs. Inside an AS, an IGP provides distributed control, while inter-AS control is managed by an exterior gateway protocol (EGP) like BGP.*

The Evolution of the IP Network Architecture

The Internet has its roots in the ARPANET, the world's first operational packet-switching network funded by the Advanced Research Projects Agency principally for research purposes. The ARPANET consisted of a small but steadily growing number of end-hosts interconnected by store-and-forward switching computers known as interface message processors (IMP) that later evolved into IP routers. These IMP were connected by low-speed long-haul leased lines and point-to-point satellite links into a peer network. Over the decades this rudimentary network architecture has evolved in a sophisticated manner into a two-level hierarchy [2]. Hosts, subnets, and their interconnects are organized into independent domains called autonomous systems (AS), each one operated under the authority of a single administrative entity. These AS are glued together by a complex, policy-based inter-domain routing mechanism, implemented by the Border Gateway Protocol (BGP) [3]. A simple model of a state-of-the-art transit AS is shown in Fig. 1.

Today's Internet architecture is inherently distributed. On the one hand, it is distributed in the way packets are forwarded in the network: each router hands a packet on to the neighboring IP router it sees as most suitable to ensure that the packet arrives at its destination. Coordinating the forwarding decisions to achieve global reachability and a consistent loop-free routing is the routing control functionality that is again distributed in nature. This means that forwarding tables emerge through a complex interaction of the routers instead of being computed by a central management entity. Within an AS, this interaction is performed by means of an Interior Gateway Protocol (IGP) and involves the automatic discovery of the network topology based on which each router computes the best next hop toward each destination address prefix. Usually, this next hop is the one that lies on the shortest path towards the destination.

Due to its distributed nature, the IP network architecture is extremely tolerant of failures, and it is unique in its capability to coordinate a large number of heterogeneous networks. However, distributed operation comes with its own price. Most notably, the monitoring and management of the network is difficult and prone to errors. Furthermore, network performance optimization usually requires central overview of the network state and coordinated corrective action by the routers, which hardly fits into a distributed architecture.

The Evolution of the IP Router Architecture

In this section, we discuss the architectural development of the most important elements that make up the Internet: the evolution of IP routers [4].

The Internet stems from a time when links were the bottlenecks in the network. Routers were built according to a regular, single-storage von Neumann computer architecture with packet processing in software, in a way very similar to PC-based routers today. Figure 2a illustrates such a first generation router with a processor, a central memory bank, line cards, and a shared bus interconnect.

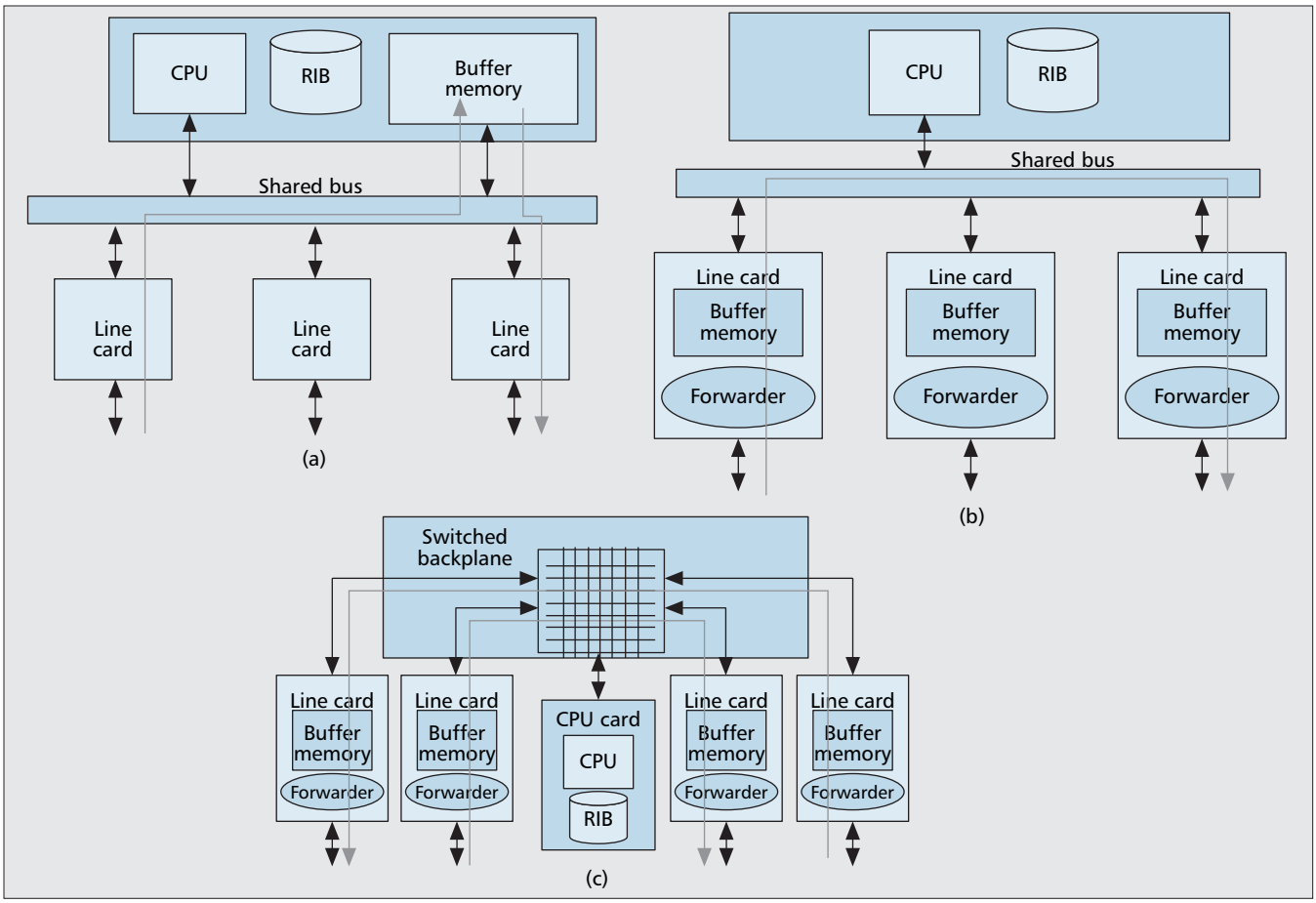
As the amount of traffic increased and links became faster, the shared memory bus and the centralized packet processing in software limited performance. In second generation routers, as shown in Fig. 2b, buffer memory and packet-processing logic are distributed onto each line card, which helps to increase capacity by sharing the packet-processing load over multiple units.

With local memory on each line card, packets are not required to traverse the shared bus to reach the buffer memory, which saves bus bandwidth. But the shared bus can only be used for one transfer at a time, so as a way to further increase capacity, the shared bus can be replaced with a switching fabric — an interconnection with parallel data paths — allowing multiple transfers to take place at the same time. This type of third-generation router is depicted in Fig. 2c. Many of today's high-end routers can be characterized as variants of the third-generation architecture, with buffer memory and packet processing logic on the line cards, and a switching fabric with parallel data paths.

We have seen that during the course of the development of the Internet, IP routers have developed from rather ordinary, single-CPU systems to highly specialized multiprocessing systems. Curiously, this dramatic architectural development has taken place almost exclusively for the data plane, while the control plane has remained virtually the same: software running on a commodity, general-purpose CPU. The few modifications that have been made to the control plane are mainly confined to hardware upgrades to deal with the growth of the network. In particular, more processing power has been added to tackle larger shortest-path tree-computation instances in intra-domain routing, while for inter-domain routing, storing more peering information and more entries in the forwarding tables has required more memory at the routers.

However, this relative changelessness of the control plane should not be perceived as an indication that the requirements on the control plane remained unchanged. On the contrary, as the requirements for more advanced network services increase, functionality supporting new protocols and services is added to the router control software.

To summarize, we conclude that the demands on the control and data planes of a router are vastly different. For control, on the one hand, we need software that supports flexibility and extensibility while always maintaining backwards compatibility without jeopardizing availability. On the other hand, for data-plane packet forwarding, we need high-speed packet-switching hardware that can handle the data feeds from high-



■ Figure 2. *The evolution of IP router architectures: a) first-generation IP router; b) second-generation IP router; c) third-generation IP router.*

capacity optical links. Because forwarding and control, while interrelated, perform functions that are largely independent of each other, we conclude that it is time to rethink the traditional, monolithic architecture of routers, where routing control and packet forwarding are intertwined into a single complex system and move IP-router architectures in the direction of separating control and forwarding functions.

ForCES and Distributed Routers

The IETF ForCES Working Group takes a first step to separate the control and forwarding plane functionality as ForCES specifies a standardized interface between the two planes within a router [5]. The ForCES architecture (illustrated in Fig. 3) is defined in terms of exchange of information between control elements (CEs) and forwarding elements (FEs). A group of CEs and FEs together form a network element (NE) that appears as an integrated piece of network equipment to external entities, quite similar to a router in the traditional sense. In such an architecture, a CE is typically based on a general purpose CPU executing control software (routing protocols, management, etc), while an FE can be based on a variety of hardware platforms performing packet-forwarding functions (classification, metering, forwarding, etc).

The ForCES protocol executes over the CE-FE interface and is responsible for the association establishment between CE and FE, as well as for steady state communication. In steady-state, an FE is continuously updated with configuration information from the CE, queried for information by the CE, or spontaneously sending asynchronous event notifications to the CE. According to ForCES, an FE is represented as a set of logical functional blocks (LFB) connected together to form a data path along which a packet travels. By configuring these individual LFBs, a CE can decide how an FE should process packets.

The ForCES framework can be seen as a first important step toward the visionary concept of a *distributed router*, where forwarding and control are separated inside routers both physically and administratively, with a standardized communications interface between the two (Fig. 4). This modularization delivers several potential benefits. Separate components allow component vendors to specialize in one component without having to become experts in all components. A standard protocol allows interoperability between the components of different vendors that translates into increased design flexibility and rapid innovation in both the control and forwarding planes. Scalability is also provided by this architecture in that additional forwarding or control capacity can be added to or removed from the NE without disrupting the operation of the rest of the NE.

Centralized Control Architectures in IP

The distributed router architecture holds much promise for network operators and device vendors. For the former, it brings better device scalability and price-efficiency. For the latter, it has the potential to open up new markets and business models. However, from a pure systems engineering standpoint, an IP network equipped with distributed routers still remains plagued by the shortcomings that today's IP networks suffer from, including the complex and error-prone network management and difficulties in optimizing performance. In this section, we argue that the vast majority of these problems originate in one very important design concept inherent in the IP protocol suite, namely, the distributed implementa-

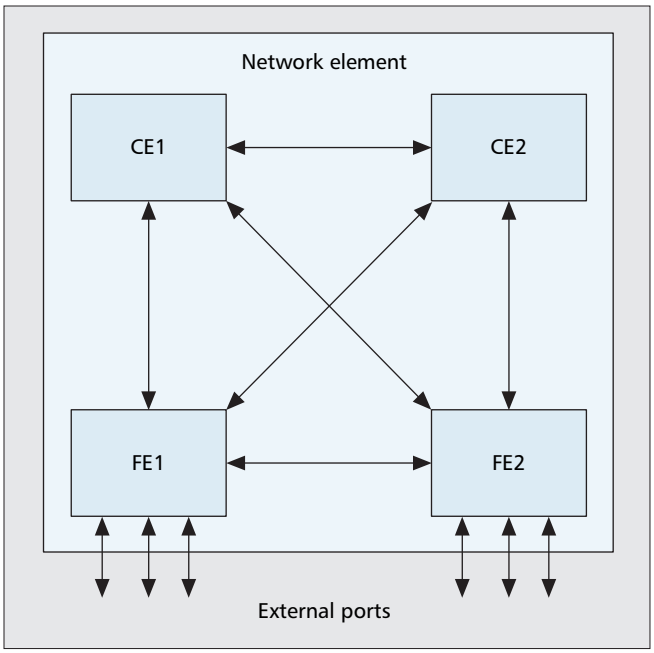


Figure 3. *The ForCES framework.*

tion of the control functionality. We further argue that overcoming this problem requires *converging the evolution of the IP network and router architectures*, and we discuss an advanced architecture that, in our view, represents the next logical step in this process. In this architecture, control functionality is decoupled from ordinary network devices and unified at a central administrative unit.

Initiatives for centralizing network control continue to appear in the last decade in diverse areas of networking. Central network management applications are extensively used to remotely monitor and administer network devices via standardized protocols, such as the Simple Network Management Protocol (SNMP). In the area of routing control, BGP route reflectors were standardized to enable routers in an AS to negotiate inter-domain routes through a central element instead of being required to establish a full mesh of BGP sessions among themselves. This idea is improved upon in [6], where *all* the control functionality related to inter-domain routing is centralized at a single module called a routing control platform. In the area of differentiated services over IP, the concept of bandwidth brokers [7] was introduced. A bandwidth broker is a central node that tracks available resources in a network to make admission control decisions on behalf of the edge nodes. In telecommunication networks and wherever required, centralized AAA servers perform authentication, authorization, and accounting. Other examples in this area are centralized operational and business support systems (OSS and BSS). It seems that currently only one important piece is missing from the puzzle: bringing together all the widely scattered control functionality into one dedicated central control module.

An IP network following such a centralized control approach is depicted in Fig. 5. This architecture retains the concept of IP-distributed routers as NE consisting of FEs and CEs. The main difference is that the control mechanisms in each NE have been reduced to the minimum functionality that is hard to centralize due to the local scope, including:

- Neighbor discovery and keepalive/failure detection by means of a simple Hello protocol
- A rapid fail-over mechanism for alternative forwarding tables in case of a failure
- *Slow path* for forwarding packets that require special handling, such as (packet data unit) PDU or difficult-to-NAT protocol messages, and so on.

All remaining control functions are stripped from the NE and delegated to a central module that we call centralized control platform (CCP). This functionality can include:

- Discovering and tracking the topology of the network
- Computing and downloading forwarding tables to the routers
- Running the inter-domain routing protocol
- Running the signaling protocol and manipulating the quality of service (QoS) infrastructure (packet filters, queues, etc.)
- Central administration and network management
- Admission control, AAA, policy enforcement
- Interoperation with legacy network protocols; remote network management applications; AAA, BSS, and OSS systems; and so on.

A Comparison of Central and Distributed Control

As our networks are steadily getting larger and more complex, *maintenance and administration* becomes more and more of a challenge. This is especially so in distributed architectures, where network state information and administration databases must be maintained in an inherently heterogeneous and multivendor network environment. This is further complicated by standardization and commercial support of management information bases that are always lagging behind. Therefore, a typical IP management software is complex and costly, requiring a significant amount of management in itself. However, in a centralized architecture, where all the relevant information is readily available at the CCP, network management boils down to executing simple queries to the management information bases at the CCP and bringing management decisions into effect is as easy as modifying entries in the administration database. Note that network management did not just magically disappear; it is delegated to the control information

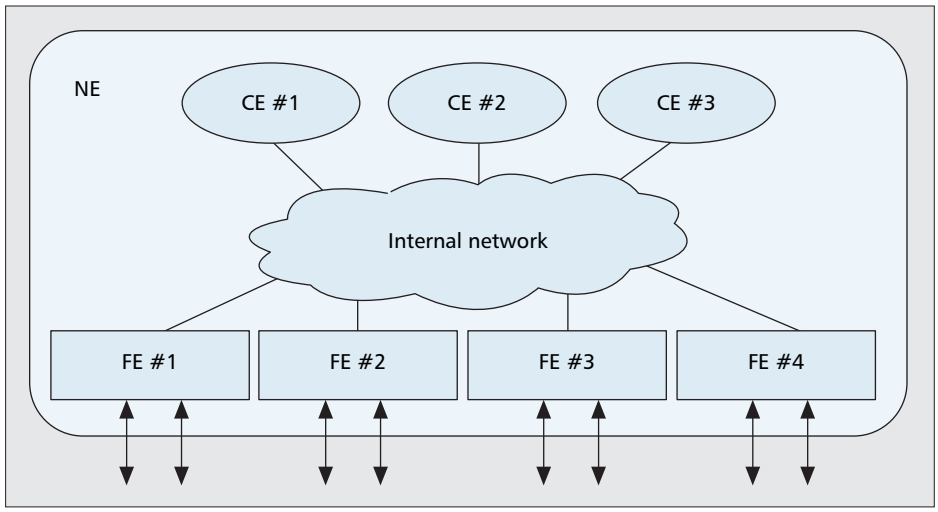


Figure 4. *Overview of a distributed router.*

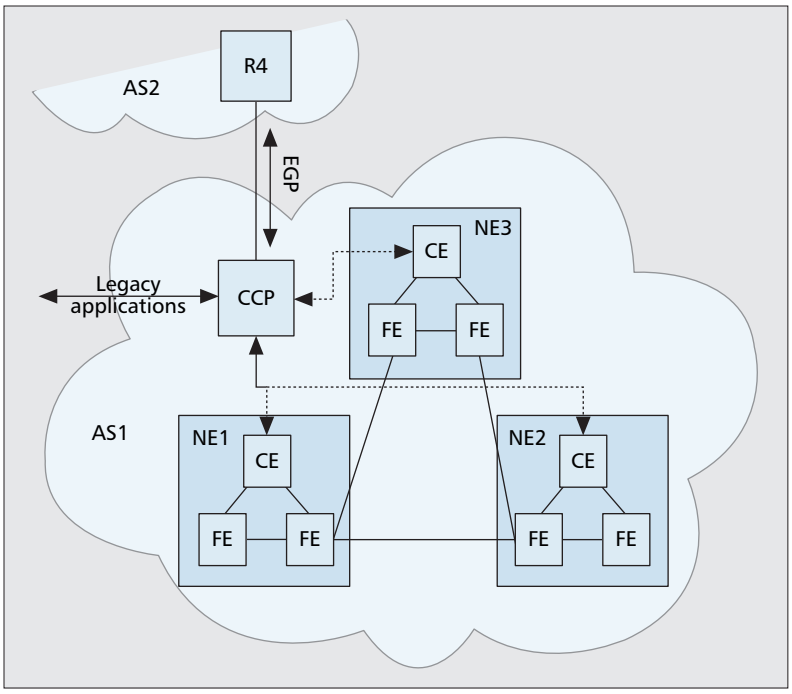


Figure 5. *Model of an AS consisting of three NEs and a CCP module providing centralized network-global control functions. NEs consist of a CE module for local control and several FEs responsible for data forwarding. CE-FE interaction inside an NE is managed by the ForCES protocol.*

exchange protocol operating between the CCP and the NE, a mechanism that is an intrinsic component of the network architecture itself.

Another important question is that of *scalability and price-efficiency*. Scalability in this context does not connote how the global network copes with the exponentially increasing amount of traffic or the growing number of users. We know that the Internet scales extremely well. Instead, by scalability, we mean how network devices themselves can be scaled to cope with the increasing burden, and what capital and operational expenditures (CAPEX and OPEX) are involved in this process. This is important as a more economical network architecture could be one of the few reasons to consider a new architecture at all [2]. In this regard, the separation of control functionality from forwarding functionality yields important benefits. Under the restrictions of the monolithic design of today's IP routers, upgrading a router involves substantial expenditure, especially as we enter into the reign of top-notch IP routers. In contrast, in an architecture where forwarding modules and control modules can be scaled separately, dimensioning of the required resources is much easier, which yields smoother and less expensive upgrade paths.

A notable difference between the distributed control model and the centralized model lies in the differing *organization of the routing functionality*. Conventionally, a laborious interaction of the IGP and Exterior Gateway Protocol (EGP) protocol instances at distant network elements assures comprehensive intra- and interdomain forwarding. In the centralized model, however, the situation is quite different. On the one hand, the IGP functionality in the centralized model is divided between the NE and the CCP: while neighbor-discovery runs at the NE, topology discovery, and computation of the forwarding tables occur at the CCP. On the other hand, the EGP functionality is completely integrated into the CCP, which uses BGP to exchange interdomain routing information with the neighboring ASs [6]. Similar compatibility interfaces could manage the interaction between the CCP and legacy network applications, such as AAA systems or OSS and BSS modules.

In order to maximize the profitability of the valuable network infrastructure, it is essential to optimize the path that traffic takes as it traverses the network. This process is usually referred to as network performance optimization, or traffic engineering (TE). Unfortunately, performing TE under distributed control is not easy. First, monitoring the network is difficult. Moreover, because distributed routing protocols usually implement some type of shortest-path routing, the range of applicable paths is rather limited, and load-balancing schemes, for example, equal-cost-multipath, (ECMP) do not help much either here. In contrast, central control is much better suited for TE. First, all relevant network state information is readily available at the CCP, which is therefore a plausible choice to host TE functionality. In addition, a centralized control architecture is not confined to shortest-path routing, but instead is free to adopt a much wider range of sophisticated path assignment and load-balancing strategies by encoding them into the forwarding tables. With central control, it is possible to realize the optimal traffic allocation in an IP network (the one yielded by the minimum cost multicommodity-flow linear program), which is in sharp contrast to the shortest-path model where attaining the optimum can be shown to be *theoretically* impossible in many cases.

Occasionally, network device outages, link cuts, and other failures inevitably appear in an operational network, and the network must be able to cope. The traditional approach to IP *resilience* is global response: the router incident to the failed component floods an advertisement throughout the network, so that routers several hops away from the failed component can adjust their forwarding tables appropriately. Although global response is quite reliable, it is intrinsically slow. Additionally, this approach is not suitable for centralized control, because it makes it impossible to react to failures without the intervention of the CCP, which makes the process fragile and even more cumbersome. To cut the all-important response time, the IETF initiated the IP fast reroute (IPFRR) framework to introduce local-response to IP networks similar to multiprotocol label switching (MPLS) fast reroute [8]. In IPFRR, routers adjacent to the failure perform local measures to mitigate it, such as installing a precomputed alternative *backwarding* table or redirecting incoming packets to pre-established tunnels that circumvent the failure. With the advent of IPFRR, handling of failures has become possible without the immediate involvement of the control functionality, and this means that IPFRR is readily applicable with central control.

Providing QoS by implementing per-flow admission control has been an important challenge for quite some time. Unfortunately, the distributed hop-by-hop admission control scheme implemented by the *de facto* IP signaling protocol suite, the resource reservation protocol (RSVP), does not scale very well to large networks, as each router must maintain per-flow state information. In contrast, centralized admission control and resource management (similar to the way bandwidth brokers operate) seems a better fit for providing QoS in IP networks and at the same time, it also opens up the possibility of integrating admission control with AAA and network-wide policy enforcement. Again, the CCP proves a plausible choice to host all these functions.

Networks operate under rapidly and persistently varying conditions, and it must be guaranteed that packet forwarding remains intact through the disruptions. In today's Internet, it

Control	Centralized	Distributed
Management	Easy to manage, deployment of new features and software updates are straightforward	Distributed management is hard and error-prone, new features are hard to deploy
Scalability	Highly scalable — generally, FEs are the bottlenecks, but FEs scale well	Less scalable — forwarding and control are entangled
Price efficiency	FEs are the bottlenecks, but FEs are simple and, as such, "cheap"	Price/capacity figure of monolithic routers rises exponentially
Forwarding	Standalone forwarding tables at the routers — routing decisions for different destinations are decoupled one from another	Shortest path routing — implies an inherent coupling between forwarding paths toward different destinations
TE algorithm	More complex routing algorithms can be implemented, since all computation occurs at one place	Only simpler algorithms, designed from scratch for distributed operations
Path assignment	More freedom to assign paths	Only shortest path(s) can be employed
Load sharing	Arbitrary traffic splitting ratios can be chosen	Only ECMP can be used
Resilience	IPFRR	Global response or IPFRR
QoS	Bandwidth broker solutions	Hop-by-hop signaling protocols (e.g., RSVP or NSIS)
Consistency	Forwarding tables might be inconsistent during an update, but this can be avoided with clever configuration	Unavoidable inconsistencies and micro-loops during routing table recomputations
Robustness	Backups for assuring fault tolerance	Robust by nature
Security	No service disruptions as long as the CCP runs, though, a single CCP might pose a target to central attack vectors	Compromising a router could render all its protocol functionality unavailable
Comp. requirements	All computation at one place — CCP might be a bottleneck	Computations are essentially distributed, which may or may not cause replication
Standardization status	With possible exception of ForCES, not standardized	Standardized, tested out and deployed in large quantities

Table 1. *A comparison of distributed and centralized control architectures.*

is the clever design of distributed-routing protocols that assures that forwarding tables at routers, apart from temporary transients, are *consistent* and loop-free. Although this consistency of forwarding tables is inherently guaranteed in a central control architecture (because the forwarding tables are computed in one place), it seems that centralized control has difficulty matching the intrinsic *robustness* of a distributed architecture. This is because the CCP poses a single point of failure. However, under the authority of the local CE and endowed with their default forwarding tables, routers can continue normal forwarding operations even if central control fails temporarily. Additionally, at least in routing control, there is no state that a back-up CCP must share with the primary CCP, which makes it possible to invoke cold backups: as soon as the primary CCP becomes unavailable, a back-up CCP kicks in; it quickly associates with the routers, learns the network topology, and takes control. This *graceful restart* mechanism ensures that a centralized control architecture can be as reliable and fault-tolerant as the distributed architecture of today's Internet.

An extensive comparison of control architectures must not ignore the differing *computational requirements* involved in setting up the forwarding tables. In the case of a distributed IGP, routers must solve the all-pairs shortest path problem cooperatively to set up consistent routing tables. Curiously, this computation is paralleled quite efficiently: each router computes the shortest-path tree routed at itself, so computational load is well balanced among the routers, and no replication of computations occurs. On the other hand, distributed operations in the case of BGP results in a lot of unnecessary replication of computations at the routers. In contrast, centralization opens up the possibility of deploying sophisticated TE algorithms at the CCP that are computationally intensive but heavily optimized for the specific task. It must be noted, however, that centralization of computations might render the CCP a bottleneck.

Another important issue is that of *network security*. One might think that a distributed control architecture is intrinsically well equipped against attacks. However, the truth is that compromising just a single router might induce a chain reaction rendering the entire distributed control service unreachable and induce a wave of compromises spreading along the lengthy chains of trust. Additionally, these chains of trust are rather management-intensive, and this will become even worse with the advent of ForCES, where CEs and FEs must authenticate each other one by one. In contrast, central control alleviates most of these shortcomings, but again, one must not ignore attack vectors that could pose threats to the CCP as a single point-of-failure.

Table 1 summarizes the discussions in this section.

Open Issues

Introducing central control in IP networks raises a number of important questions. First and foremost, a means for arranging the CCP-NE communications must be worked out, preferably on the basis of the ForCES protocol. Next, it must be clarified how differently architected islands of IP networks, either being complete ASs or IP subdomains inside an AS, can be operated side-by-side. For some pointers on the nondisruptive deployment of centralized control, see [6]. Additionally, it is also questionable how the CCP interoperates with legacy network applications, designed for a distributed environment but already working in a centralized fashion, such as BSS, OSS, and AAA systems or remote network management applications.

Another issue is how to implement load sharing between multiple CCPs if for improving survivability or decreasing processing load, an operator wishes to use multiple CCPs in parallel. In [9], the authors propose a method to run multiple BGP modules in parallel, but it is questionable exactly how this method could be extended to other routing and control protocols.

Finally, we mention a perplexing problem concerning the organization of the control and the data plane: if control communication takes place directly over the data network (in-band signaling), then we have an interesting chicken and egg problem. Namely, an NE can be remotely configured no sooner than it becomes reachable from the CCP, but reachability is hard to assure without first configuring the NE [10]. A solution would be to provide routers with some default forwarding table to assure minimal reachability. Alternatively, establishing a dedicated control network (out-of-band signaling) would be a viable solution as well, as is the case with the signaling system no. 7 (SS7/C7) architecture.

Conclusion

In this article, we have reviewed the most important architectural developments of the past 25 years that led to the unrivaled success of the Internet we are witnessing today: the evolution of IP networks from the ARPANET to today's hierarchically structured Internet and the evolution of IP-router design from an essentially monolithic architecture to the modern concept of distributed routers with separated control and forwarding functionality. We argue that this co-evolution of network and router architectures must converge at some point, and we have identified centralization of the control functionality as the next logical step in this process. As [6] phrases it: "IP routers should be lookup-and-forward switches, forwarding packets as rapidly as possible without being concerned about path selection. A separate entity should be responsible for computing the best [. . .] paths on behalf of all the routers in a domain and disseminating the results to the routers." We have compared distributed and central control architectures from the different aspects of systems engineering, and conclude that central control brings important benefits, such as improved manageability and security, more economical operations, and better performance.

References

[1] Future Internet Network Design — FIND program, http://find.isi.edu
[2] M. Handley, "Why the Internet Only Just Works," *BT Tech. J.*, vol. 24, July 2006.
[3] S. H. Y. Rekhter and T. Li, "A Border Gateway Protocol 4 (BGP-4)," IETF RFC 4271, Jan. 2006.
[4] J. Aweya, "On the Design of IP Routers Part 1: Router Architectures," *J. Sys. Architectures*, vol. 46, Apr. 2000, pp. 483–511.
[5] Y. Lang *et al.*, "Forwarding and Control Element Separation (ForCES) Framework," IETF RFC 3746, Apr. 2004.
[6] N. Feamster *et al.*, "The Case for Separating Routing from Routers," *ACM SIGCOMM Wksp. Future Directions in Network Architecture*, Portland, OR, Sept. 2004.
[7] Z.-L. Zhang, "Decoupling QoS Control from Core Routers: A Novel Bandwidth Broker Architecture for Scalable Support of Guaranteed Services," *SIGCOMM*, 2000, pp. 71–83.
[8] A. Raj and O. C. Ibe, "A Survey of IP and Multiprotocol Label Switching Fast Reroute Schemes," *Comp. Networks*, vol. 51, no. 8, 2007, pp. 1882–1907.
[9] M. Hidell, Decentralized Modular Router Architectures, Ph.D. thesis, Royal Inst. Tech. (KTH), Sept. 2006.
[10] A. Greenberg *et al.*, "A Clean Slate 4D Approach to Network Control and Management," *SIGCOMM Comp. Commun. Rev.*, vol. 35, no. 5, 2005, pp. 41–54.

Biographies

ANDRAS CSASZAR (Andras.Csaszar@ericsson.com) graduated in computer science from the Budapest University of Technology and Economics (BME), Hungary, in 2001. He is currently finalizing his Ph.D. thesis work at the same university. Since 2001 he has been working at TrafficLab in Ericsson Research Hungary. His research interests include IP inter- and intradomain routing, next-generation networking technologies, network resilience, and resource reservation.

GABOR ENYEDI (enyedi@tmit.bme.hu) received an M.Sc. degree in computer science from BME in 2006. Since then he has been a Ph.D. student at BME and a member of the High Speed Networks Laboratory in the Department of Telecommunication and Media Informatics. His research interests include traffic engineering in circuit-switched networks and IP fast reroute.

GABOR RÉTVARI (retvari@tmit.bme.hu) received his M.Sc. and Ph.D. degrees in electrical engineering from BME in 1999 and 2007, respectively. He is now a research assistant at the High Speed Networks Laboratory, Department of Telecommunications and Media Informatics, BME. His research interests include QoS routing, traffic engineering, the networking applications of computational geometry, and the mathematical theory of network flows.

MARKUS HIDELL (mahidell@kth.se) received his M.Sc. degree in electrical engineering and his Ph.D. degree in telecommunications from the Royal Institute of Technology (KTH), Stockholm, Sweden, in 1992 and 2006, respectively. He is now an assistant professor (acting) in communication systems at the Telecommunication Systems Laboratory, School of Information and Communication Technology, KTH. His current research interests include switch and router architectures, protocols, and network architectures.

PETER SJÖDIN (psj@kth.se) received his M.Sc. and Ph.D. degrees in computer science from Uppsala University, Sweden, in 1982 and 1992, respectively. He is now an associate professor in communication networks at Telecommunication Systems Laboratory, School of Information and Communication Technology, KTH. His current research interests include network system architectures, protocols and network architectures, and IP-optical integration.