# The PacketCable Architecture

*Ed Miller, Terayon, Inc.*
*Flemming Andreasen, Cisco Systems*
*Glenn Russell, CableLabs®*

## ABSTRACT

PacketCable™ defines a multimedia system architecture that overlays a high-speed cable modem access network, with the goal of enabling a wide variety of Internet-Protocol-based multimedia services, such as voice over IP, unified messaging, PBX extension, videoconferencing, and online gaming. Currently, the architecture provides a complete solution for delivering VoIP services. In this article we present the core capabilities necessary to implement such services in a scalable fashion. Next, we describe the major functional components that comprise the PacketCable architecture and illustrate how they work together to form an integrated IP multimedia-enabled system architecture, which is presently focused on voice over IP. Finally, we present some possible next steps in the evolution of PacketCable.

## INTRODUCTION

The emergence of the Internet Protocol (IP) [1] as the standard transport mechanism for data networks has enabled a revolution in communications services and applications. This online revolution is evidenced by the explosive growth of a wide variety of Internet applications, such as e-mail, chat groups, music, video, and the World Wide Web. New classes of IP-based information appliances are also emerging, including multimedia personal computers, set-top boxes, and voice and video IP phones.

For the past decade, circuit-switched voice traffic has been growing at a linear rate while packet-switched data traffic has been growing exponentially. Today, the amount of data traffic already exceeds that of voice traffic, and if current trends continue, the mix of these traffic types will soon be skewed to the point where traditional voice traffic becomes a fraction of the total traffic. Maintaining two separate network infrastructures for voice and data traffic will become increasingly economically inefficient, thereby forcing service providers to consolidate. In addition, as the Internet revolution has shown, IP-based networks enable a whole new class of advanced services that are appealing to subscribers.

One of these advanced services is voice over IP (VoIP), which is a special case of real-time multimedia over IP. VoIP technology first appeared around 1995, and was used for toll bypass services where an arbitrage opportunity existed for providing long distance calls. Today, the technology has matured to the point where VoIP can be used to provide voice services comparable in quality and performance to those provided by the public switched telephone network (PSTN).

However, providing a scalable VoIP service and supporting next-generation multimedia services requires a number of components, which must all work together. PacketCable is a project conducted by Cable Television Laboratories, Inc. (CableLabs®) and its cable operator member companies, which seeks to define an end-to-end architecture and an associated set of interface specifications for providing VoIP, video, and other real-time multimedia services over two-way hybrid fiber-coax (HFC) cable access systems. The initial phase of PacketCable focuses on VoIP, specifically primary and secondary line residential IP telephony; however, the long-term goal encompasses a larger suite of packet-based capabilities, such as videoconferencing, unified messaging, and online gaming.

Taken as a whole, PacketCable defines a highly integrated system, and includes mechanisms for device control, session signaling, dynamic quality of service (QoS) management, client provisioning, PSTN interfacing, network management, billing and settlements, and security. To the authors' knowledge, this work represents the most complete and advanced set of interoperable VoIP specifications available anywhere in the world today.

In the remainder of this article we present an overview of the PacketCable architecture. We then describe the functional components that form the PacketCable architecture and illustrate how they work together to form an integrated VoIP system. Finally, we will present some of the next steps in the evolution of PacketCable.

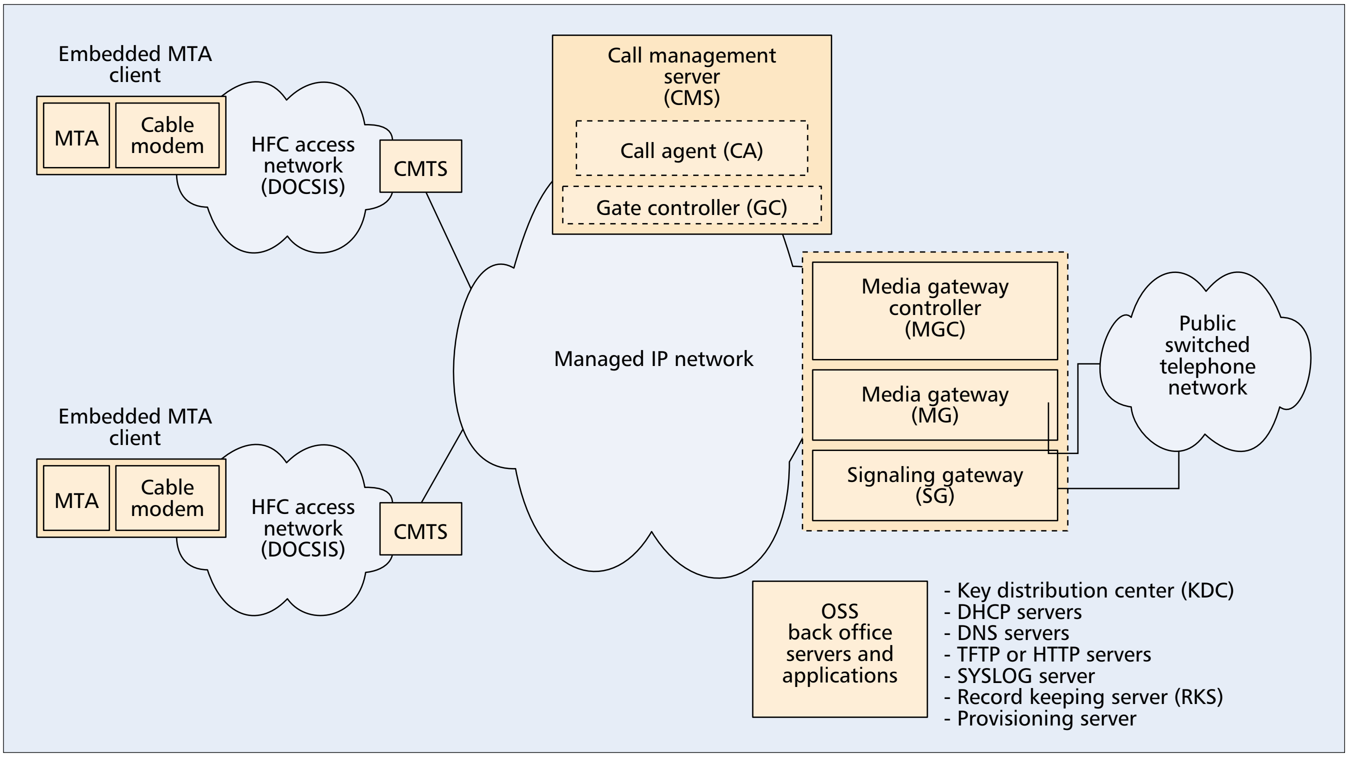



**Figure 1.** *The PacketCable reference architecture.*

## PACKETCABLE ARCHITECTURE

The PacketCable architecture [2, 3] utilizes the services of three underlying networks: the HFC access network, the managed IP network, and the PSTN, as illustrated in Fig. 1. The cable modem termination system (CMTS) uses the Data Over Cable Service Interface Specification (DOCSIS™) 1.1 protocol [4] to manage transmission on the shared HFC access network to and from the cable modems. The multimedia terminal adapter (MTA), which provides the services on top of IP, may be either a separate device or embedded within the cable modem as shown here. The CMTS also provides connectivity between the HFC access network and the managed IP network, which serves several functions. It provides interconnection between the basic PacketCable functional components responsible for signaling, media, provisioning, and QoS establishment. In addition, the managed IP network provides long-haul IP connectivity between other managed IP and DOCSIS HFC networks or the gateways to the PSTN.

PacketCable uses what is known as a softswitch architecture for VoIP, where many of the functions traditionally resident in a class 5 central office switch have been implemented in a decomposed and distributed architecture. Instead of running on special-purpose hardware, most of the software runs on general-purpose servers, which leads to a low-cost, highly flexible architecture.

Subscribers served by the PacketCable architecture will have an MTA at their premises which performs various signaling and protocol conversion functions to permit voice communications over the network. The MTA provides codecs and all signaling and encapsulation functions required for media transport and call signaling.

The MTA contains a subscriber-side interface to client devices (e.g., a telephone) and a network-side signaling interface to other PacketCable network elements via the DOCSIS 1.1 high-speed data network.

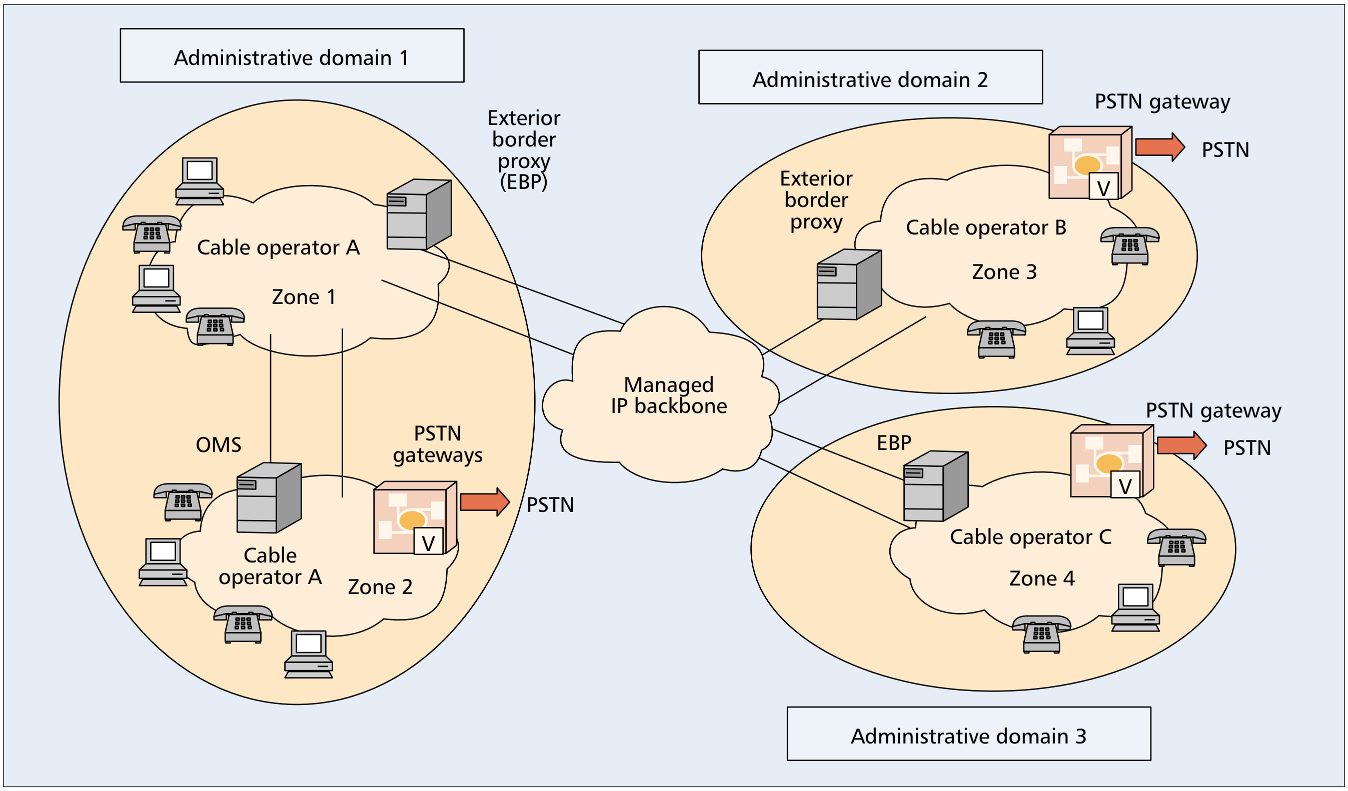
The service provider in turn has a set of servers that communicate with the MTAs and with other servers to handle the actual call setup and feature invocation, to manage and allocate network resources (bandwidth), and to generate call detail records for usage. Also, the service provider has supporting infrastructure systems for security, provisioning, name resolution, and other operational support services.

PacketCable service providers may interconnect their networks to form a national or even international footprint. Well-defined signaling, security, and settlement mechanisms allow the service providers to interconnect using a variety of different business models. In instances where connectivity with the PSTN is desired, such connections may be accomplished through conventional means of interconnect using PSTN gateways.

### FUNCTIONAL COMPONENTS

The PacketCable architecture is defined in terms of functional components and protocol interfaces, rather than physical device characteristics. This allows manufacturers flexibility in their product implementations, as functional components which may be physically combined to create differentiated product offerings. The following are the major functional components within the PacketCable Architecture:

• Call management server (CMS): The CMS




■ **Figure 2.** *Zones and administrative domains.*

consists of a call agent (CA) component and a gate controller (GC) component. The CA contains the call state for a call and controls the MTA. The CA instructs the MTA to look for certain events (e.g., off-hook), play certain signals (e.g., dialtone), and set up media connections to other devices, such as another MTA. The CA component also communicates with other CMSs and media gateway controllers. The GC component performs QoS admission control. It communicates with the CMTS to either admit or deny requests for QoS generated by the MTA.

- Media gateway controller (MGC): Interconnection to the PSTN involves a PSTN gateway, which consists of an MGC, a media gateway (MG), and a signaling gateway (SG). The MGC controls the MG similar to how the CMS controls the MTA. Additionally, the MGC interfaces to the SG for common channel signaling (CSS) such as Signaling System 7 ISDN User Part (SS7 ISUP).
- Media gateway (MG): The MG terminates the PSTN bearers. The MG is controlled by the MGC and exchanges packetized media with other devices on the PacketCable network (e.g., an MTA).
- Signaling gateway (SG): The SG provides the call control signaling interface to the PSTN for CCS. The SG communicates with the MGC, which is the termination point for the CCS signaling.
- Operational support systems (OSS): The OSS contains supporting infrastructure functions such as provisioning systems, record keeping servers (RKS) for billing, key distribution center (KDC) for security, and domain name system (DNS) servers for name resolution.
- Exterior border proxy (EBP): The EBP is a Session Initiation Protocol (SIP) proxy that provides a secure signaling interface between PacketCable domain networks.

### SYSTEM INTERFACES

PacketCable networks are organized into zones and domains as illustrated in Fig. 2.

A PacketCable zone consists of the set of MTAs managed by a single functional CMS, which may be a single server or a cluster of servers operating as a single logical entity.

A PacketCable domain is made up of one or more PacketCable zones that are operated and managed by a single administrative entity (e.g., a single cable operator). The interface between zones within a domain (intradomain) is similar to the interface between zones in different domains (interdomain), with one exception: each domain initiates and receives initial communication through an EBP. The EBP's primary role is to route calls and facilitate secure communication between domains.

In the following, we describe the communication interfaces between the various functional components, providing more detail on the architecture and its major components to support single-zone, intradomain, and interdomain communication.

***Endpoint Signaling*** — In recent years, two distinct VoIP signaling models have emerged. The first is based on an intelligent endpoint model where the MTA contains the call state and logic associated with setting up and controlling calls as well as providing various value-added features. SIP [5], for example, is based on the intelligent endpoint model. In the second model, often referred to as the *decomposed gateway*, the functionality is split between two functions: an intelligent CA or MGC,[1] which controls a simple MG through a control protocol such as the MGC Protocol (MGCP) [6]. PacketCable 1.x is based on the decomposed gateway architecture; however, it is still completely compatible with the intelligent endpoint model.

The CA part of the CMS interacts with the MTA via the Network-Based Call Signaling (NCS) Protocol [7]. NCS is a profile of MGCP that only includes the parts of MGCP necessary for providing residential voice services. In addition to this, NCS contains extensions necessary to support the PacketCable QoS and security mechanisms. NCS is a simple User Datagram Protocol (UDP)-based protocol for managing *endpoints* (e.g., telephones) and *connections* between endpoints. Endpoints can have events detected on them, such as off-hook, and signals can be generated on them (e.g., dial tone), all of which is controlled by the CMS. The events and signals are defined in *packages*, which allows for modularity and extensibility. The CMS also instructs the MTA to create, modify, and delete connections, through which endpoints exchange packetized media, typically a Real-Time Transport Protocol (RTP) audio stream [8]. CMSs and MTAs address each other through the use of domain names, which facilitates reliability and failover. Also, the MTA informs the CMS when it comes in service, goes out of service, and if it experiences any temporary service problems. Finally, NCS includes auditing commands that allow the CMS to learn the endpoints' capabilities as well as protocol-related state; the actual call state, however, only exists in the CMS.

***Intra- and Interdomain Signaling*** — CMS-to-CMS communication[2] is based on SIP with extensions from the Distributed Call Signaling (DCS) protocol framework [9] for intelligent endpoints, a possible future direction for architecture expansion. These extensions include:

- Integration of resource management and call signaling to ensure proper availability of QoS resources prior to alerting the called party
- Passing of accounting information to enable generation of call detail records
- Passing of calling and called party identity information and the ability of either to request anonymity
- Support for operator services, including busy line verification and emergency interrupt
- Support for lawfully authorized electronic surveillance

[1] *These are essentially two different names for the same function.*

[2] *The term CMS tends to be used loosely and generally also includes MGCs, as is the case here.*

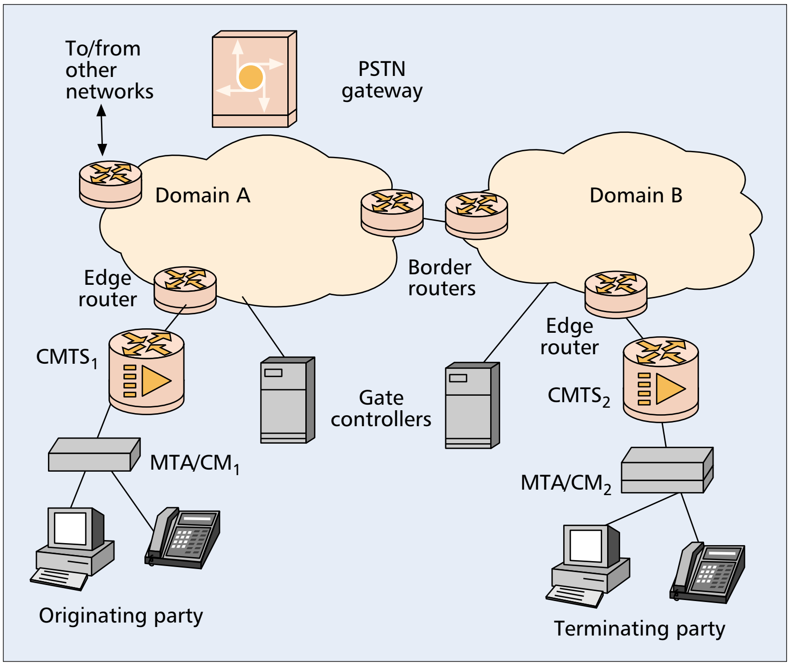



**Figure 3.** *PacketCable QoS architecture.*

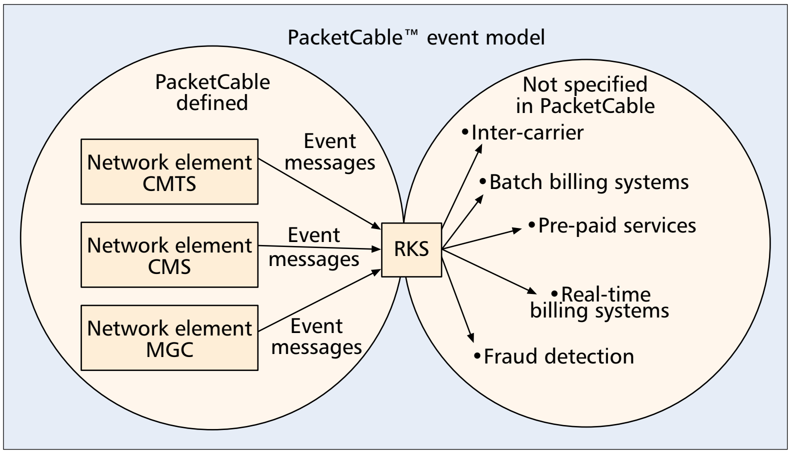
***Quality of Service*** — PacketCable is designed to run over a QoS-enabled managed IP network (i.e., not the public Internet) in order to ensure high quality and reliability. Both the DOCSIS 1.1 cable modem access network and the backbone itself support these features in a scalable manner based on the Internet Engineering Task Force (IETF) integrated services (Int-serv) [10] and differentiated services (Diff-serv) [11] models.

An IP-based data network will, by default, deliver a best effort service. Packets may experience variable queuing delays and may be dropped at random, resulting in unpredictable end-to-end throughput and delay. Such characteristics are not acceptable to real-time multimedia services that place upper bounds on acceptable packet loss, latency, and jitter. QoS is the mechanism that allows such requirements to be met.

PacketCable has adopted a segmented QoS model (Fig. 3) partitioning resource management into distinct access network and backbone network segments. Segmented resource management is beneficial for two reasons:

- It allows for different bandwidth provisioning and signaling mechanisms for the originator's network, the far end network, and the backbone network.
- It allows for resource-constrained segments to maintain per-flow reservations and carefully manage resource usage. At the same time, segments that have sufficient resources may employ a coarse-grained management approach to avoid keeping per-flow state, thereby enhancing scalability.

**Access Network QoS** — The access network consists of the network between the MTA and the CMTS. Access network QoS is based on the IntServ model where QoS is reserved and scheduled for individual flows. Fundamental to pro-




■ **Figure 4.** *The PacketCable event message architecture.*

viding access network QoS is the use of DOCSIS 1.1 between the CM and the CMTS on the shared HFC access network. DOCSIS 1.1 contains extensive QoS mechanisms that allow network resources to be reserved for individual packet flows.

Embedded MTAs may use DOCSIS 1.1 signaling directly when they wish to reserve network resources; the CMTS then originates Resource Reservation Protocol (RSVP) [12] messages to the far end. Alternatively, MTAs can use end-to-end RSVP signaling, which in turn will use the QoS services of the underlying DOCSIS 1.1 HFC access network. Non-embedded MTAs, which by definition cannot perform DOCSIS 1.1 signaling, must use RSVP signaling. In either case, the CMTS receives the resource reservation request and then performs an admission control decision to determine whether or not the QoS request will be honored.

In addition to normal single-phase resource reservation, the QoS model for the access network supports a two-phase reservation model for integration with call signaling. In the first phase, network resources are reserved. However, it is not until the second phase, where the network resources are committed, that the resources can be used. This allows calls to only be established when resources are available, yet enables billing only for answered calls while preventing theft of service. Several other mechanisms to prevent theft of service are in place as well.

**Backbone Network QoS** — The backbone network consists of the packet network between the CMTSs. QoS across the backbone network is based on the Diff-serv model. The edge router (ER) function, which may reside in the CMTS, sits at the edge of the DiffServ region. The ER provides per-flow RSVP admission control, DiffServ code point (DSCP) marking, and traffic policing as needed. Border routers (BRs) in turn reside at service provider boundaries within the DiffServ region, where they perform DSCP remarking and traffic policing as needed. Within the DiffServ region, admission control can be based on provisioning, per-flow, and aggregated RSVP signaling.

**Admission Control** — In a pure IntServ model, the admission control decision may be based on resource availability alone. However, since QoS is a value-added service that provides preferential treatment to some packet flows at the expense of others, the service provider should be able to control the admission control decision based on other parameters as well. PacketCable provides this ability via policy-based admission control based on the IETF Resource Allocation Protocol (RAP) framework [13]. The RAP framework defines two important functions: the policy decision point (PDP), which makes policy decisions, and the policy enforcement point (PEP), where the policy is enforced. The PacketCable PDP function is the GC, which uses the IETF Common Open Policy Server (COPS) protocol [14] to communicate with the PEP function, which is the CMTS.

In order for the CMTS to admit a resource reservation, it must first be authorized by the GC. The GC in turn communicates with the CMS to determine the allowable parameters for the resource reservation (e.g., bandwidth, source and destination addresses). Also included is subscriber identification that enables generation of QoS usage event messages, which can be used for subsequent billing. In order to reduce the call setup delay, the CMS instructs the GC to provision the CMTS with its policy during call setup. When the resource reservation arrives, the CMTS can then instantly perform the policy-based admission control as opposed to sending a request to the GC and waiting for a response.

***Event Recording and Billing*** — PacketCable uses an event-based approach to capturing information to be used for billing and session accounting. Each network element is responsible for generating event messages for the portion of the communication pertaining to them (e.g., call signaling or QoS). An event message is a data record containing information about usage and activities. An event-based format is necessary to accommodate the distributed architecture where complete session state no longer resides in one or two network elements, but is instead spread out among many (e.g., CMS and CMTS).

A single event message may contain a complete set of data regarding usage or only part of the total usage information. When correlated by the RKS, information contained in multiple event messages provides a complete record of the service. This complete record of the service is often referred to as a call detail record (CDR). Event messages or CDRs may be sent to one or more back office applications such as a billing system, a fraud detection system, or a prepaid services processor.

The structure of the event message data record is designed to be flexible and extensible in order to carry information about network usage for a wide variety of services beyond basic VoIP. Figure 4 depicts the PacketCable event message architecture. By standardizing the transport, syntax, and collection of appropriate event message attributes from a distributed set of network elements, this architecture provides a single reference point to interface to existing billing, settlement, reconciliation, and other systems.

***PSTN Interconnection*** — The interface to the PSTN is an important part of the PacketCable architecture. In addition to providing ingress and egress connections for voice traffic exchange with the PSTN, the capabilities within the gateway allow the network operator to utilize certain services within the PSTN, thereby alleviating the need for duplication within the IP network. Examples of these include operator services (411, 0+ dialing, etc.), emergency services (E911), and database services (local number portability, 800/888/900 translations, etc.).

The interconnection requirements for the PacketCable architecture can be met utilizing a relatively small number of interfaces such as SS7 ISUP signaling and standard bearer trunks for normal calls. However, some special types of calls utilize MF trunks, in particular:

- Operator services trunks:
  - Cellular local exchange carrier (CLEC) subscriber access through a LEC access tandem
  - Operator access to equal access end office/access tandem for busy line verification and barge-in
- E911 trunks for access to an E911 tandem

PacketCable was one of the early adopters of the decomposed gateway architecture for PSTN interconnection. There were several factors behind the decision to decompose the gateway. These reasons include a desire to facilitate SS7 point code conservation and to localize SS7 management requirements. It was also envisioned that a decomposed gateway will ultimately lead to products with lower cost and higher functionality by allowing vendors to specialize in their areas of expertise, either hardware interfaces or software systems.

As described earlier, the PacketCable PSTN gateway architecture defines three functional components: the MG, the SS7 signaling gateway (SG), and the MGC.

The MG provides the interfaces between the PSTN bearer channels and the packet network. Media received from the PSTN is packetized and possibly transcoded before it is forwarded to the packet network in the form of RTP packets, and vice versa.

The SS7 SG provides an interface between the PacketCable network and the SS7 network. The SG terminates the lower layers of the SS7 stack (MTP1, MTP2, MTP3, and SCCP) and transports the SS7 application protocols (ISUP and TCAP) to other components in the architecture using the PacketCable-defined Internet Signaling Transport Protocol (ISTP) [15]. In terminating the MTP and SCCP layers, the SG performs the management functions required by the SS7 network and performs an address mapping function between SS7 addresses and IP addresses.

The MGC accepts call signaling information (ISUP) from the SG, converts it to the CMS-to-CMS communication protocol (SIP with DCS extensions), and sends it to the relevant CMS, and vice versa. The MGC also controls the MG, instructing it to set up media streams to other entities (e.g., an MTA). All call state associated with the gateway is maintained in the MGC. The MGC interacts with the MG via the Trunking Gateway Control Protocol (TGCP) [16], which is also a profile of MGCP but targeted for trunking gateways. The TGCP and NCS profiles are nearly identical, the primary difference being the packages supported.

Although PacketCable has standardized the above defined decomposed gateway architecture, it is still possible to combine two or more of these functional components to create different product combinations. For example, a small deployment may be best served by an integrated product in which all of the gateway components reside in a single physical unit.

***Security*** — The last component we will examine is security, which spans all interfaces in the PacketCable architecture. It provides confidentiality for all media packets and all sensitive signaling communication across the network. Furthermore, it ensures that unauthorized message modification, insertion, deletion, and replays anywhere in the network are easily detectable and do not affect proper network operation. The mechanisms employed depend on the particular interface, but the majority of signaling interfaces are secured using IPSec [17], whereas media is secured by encrypting the payload and optionally adding a message authentication code. Depending on the particular interface, key management in turn may use Kerberos with public key initial authentication extension (PKINIT), Internet key exchange (IKE) with either preshared keys or X.509 certificates, or simply randomly generated keys exchanged within secured signaling messages. The choice of key management method depends on the expected number of connections for the interface as well as whether the two sides of the connection can be assumed to know each other's identity in advance.

## NEXT STEPS

Initially designed to support the time-to-market business considerations of North American cable operators, the PacketCable architecture will continue to evolve to meet new business requirements and to accommodate advances resulting from the maturing of IP-based technology. The scope of the PacketCable architecture has already been broadened with its adoption by the International Telecommunication Union — Telecommunication Standardization Sector (ITU-T), where it is the basis for a set of international standards known as IPCablecom, which may or may not use DOCSIS 1.1 signaling in the access network. As new service requirements are identified, the architecture will also be extended to define additional capabilities to support these services. Architectural extensions will add incremental functionality that complements the capabilities defined in the current architecture. Below, we discuss some of these future directions.

### MULTIMEDIA SERVICE ENHANCEMENT

The PacketCable architecture has been demonstrated to support a wide range of mono-media features similar to those offered by standard residential telephony service offerings. The next architectural challenge is to define additional capabilities for establishing sessions that use

*Initially designed to support the time-to-market business considerations of North American cable operators, the PacketCable architecture will continue to evolve to meet new business requirements and to accommodate advances resulting from the maturing of IP-based technology.*

*To facilitate an intelligent endpoint model, PacketCable defines support for a standalone MTA — a client device that implements the necessary multimedia signaling and codecs but is not directly embedded in a DOCSIS cable modem.*

richer media, such as real-time interactive video sessions. Applications such as multimedia telephony, interactive gaming, and media streaming will impose new requirements on the architecture and lead to protocol extensions necessary to provide such services. For multimedia telephony, the signaling must be extended to support the establishment of interactive video sessions using multimedia codecs and the QoS infrastructure already in place.

### INTELLIGENT ENDPOINTS

To date, the PacketCable architecture has defined functionality for simple subscriber devices that have been designed to support basic residential voice services. As multimedia services are required, the architecture must evolve to support greater intelligence and functionality in the endpoint device. Rather than incorporating the MTA functionality directly in the cable modem, the signaling and codec support for advanced services, such as video telephony, may be implemented on a personal computer or a personal handheld device. Such devices are expected to be physically separate from the cable modem.

To facilitate an intelligent endpoint model, PacketCable defines support for a standalone MTA — a client device that implements the necessary multimedia signaling and codecs but is not directly embedded in a DOCSIS cable modem. The standalone MTA is physically connected to the cable modem via a home network using established protocols such as Ethernet, USB, or IEEE 1394. A standalone MTA must fully implement the PacketCable call signaling and QoS protocols. QoS establishment on the home network segment between the standalone MTA and the cable modem will also be required. Additional provisioning and security extensions will be required to enable customers to purchase and install their own endpoint devices.

## CONCLUSION

We present the PacketCable architecture and its major constituent components, which, to the authors' knowledge, represent the most complete and advanced set of interoperable VoIP specifications available anywhere in the world today. We describe the underlying multimedia QoS architecture and the current VoIP services and protocols provided on top by the use of the NCS, TGCP, and SIP protocols. We also describe the event messaging model to enable billing, and provide an overview of the complete security infrastructure. Finally, we look at the future evolution of PacketCable, which we expect to focus on multimedia and intelligent endpoints.

### ACKNOWLEDGMENTS

We wish to both thank and acknowledge the numerous individuals and organizations that have contributed to the PacketCable project.

## REFERENCES

[1] J. Postel, Ed., "Internet Protocol — DARPA Internet Program Protocol Specification," RFC 791, USC/Info. Sci. Inst., Sept. 1981.
[2] PacketCable 1.0 Architecture Framework Technical Report, PKT-TR-ARCH-VI01-991201, Dec. 1, 1999, CableLabs, Ihttp://www.PacketCable.com
[3] PacketCable 1.2 Architecture Technical Report, PKT-TR-ARCH1.2-V01-001229, Dec. 29, 2000, CableLabs, http://www.packetcable.com
[4] Data-Over-Cable Service Interface Specifications, Radio Frequency Interface Specification, SP-RFIv1.1-I06-001215, CableLabs, Dec. 15, 2000, http://www.cable-modem.com
[5] Handley *et al.*, "SIP: Session Initiation Protocol," RFC 2543, Mar. 1999.
[6] M. Arango *et al.*, "Media Gateway Control Protocol (MGCP) Version 1.0," RFC 2705, Oct., 1999.
[7] PacketCable Network-Based Call Signaling Protocol Specification, PKT-SP-EC-MGCP-I02-991201, Dec. 1, 1999.
[8] H. Schulzrinne *et al.*, "RTP Profile for Audio and Video Conferences with Minimal Control", RFC 1890, Jan. 1996.
[9] PacketCable Distributed Call Signaling Specification, pkt-sp-dcs-D03-000428, Apr. 28, 2000 (ftp://ftp.cablelabs.com/pub/ietfdocs/dcsdraft2.pdf).
[10] R. Brade, D. Clark, and S. Shenker, "Integrated Services in the Internet Architecture: An Overview," RFC 1633.
[11] S. Blake *et al.*, "An Architecture for Differentiated Services," RFC 2475, Dec. 1998.
[12] R. Braden *et al.*, "Resource ReSerVation Protocol (RSVP) — Version 1 Functional Specification," RFC 2205, Sept. 1997.
[13] R. Yavatkar *et al.*, "A Framework for Policy-Based Admission Control," RFC 2753, Jan. 2000.
[14] J. Boyle *et al.*, "The COPS (Common Open Policy Service) Protocol," RFC 2748, Jan., 2000.
[15] PacketCable Internet Signaling Transport Protocol (ISTP) Specification, PKT-SP-ISTP-I01-991201, Dec. 1, 1999, CableLabs, http://www.PacketCable.com
[16] PacketCable PSTN Gateway Call Signaling Protocol Specification, pkt-sp-tgcp-I01-991201, Dec. 1, 1999, http://www.packetcable.com
[17] S. Kent and R. Atkinson, "Security Architecture for the Internet Protocol," RFC 2401, Nov. 1998.

## BIOGRAPHIES

FLEMMING ANDREASEN is a technical leader with Cisco Systems and has been involved in VoIP since 1997. He has helped develop several VoIP-related specifications and standards, and has given numerous industry presentations on the topic. Prior to joining Cisco, he consulted with Telcordia Technologies, Lucent Technologies, and NCR. He holds an M.S. degree in computer science from the University of Copenhagen.

EDWARD MILLER (emiller@terayon.com) is vice president of Voice Solutions for Terayon, Inc., where he is responsible for defining product and solution strategies to meet the needs of Terayon's VoIP customers. Prior to joining Terayon, he was director of the PacketCable project for Cable Television Laboratories, Inc. (CableLabs), where he played a key role in the development of the PacketCable architecture. He has 16 years of experience in the telecommunications and software industries. He holds a B.S.E.E. degree from the University of Pittsburgh and an M.S.E.E. degree from Carnegie-Mellon University.

GLENN RUSSELL (g.russell@cablelabs.com) is director of Multimedia Architectures for the PacketCable project at CableLabs, where he has overall responsibility for the development of the interface specifications that define the PacketCable architecture. Prior to joining CableLabs, he spent several years in the teleconferencing industry where he led efforts to develop and promote standards for voice, video, and data conferencing services. He attended the University of Colorado at Boulder where he earned his B.A. in economics and completed graduate studies in the Interdisciplinary Telecommunications Program (ITP).