A building-block approach to media gateway development

By Joshua Adelson, James Rafferty, and Vivek Pachaury

The current telecom market downturn has caused equipment vendors to scrutinize their technology investments and sharpen their market focus. Vendors who once pitched VoIP gateways are now tailoring their offerings to specific markets such as voice over cable, wireless voice over packet, or IP Centrex, to name a few.

The need to support these specific network environments has increased the diversity of functional requirements for media gateways (MGs). In addition to market-based specialization, other trends that continue to challenge gateway developers include scalability from low to high port densities, high-availability requirements, and ongoing evolution of packet call control protocols.

To meet these demands of the marketplace, few system vendors can afford to build all the hardware and software elements from scratch. A building block approach – using commercially available hardware and software elements – allows gateway system vendors to take advantage of already-developed core technology elements and focus their energies on the particular needs of their target markets. This article will review the general functional requirements of MGs, and then present hardware and software strategies that can reduce development cost, risk, and time-to-market without sacrificing functionality or flexibility.

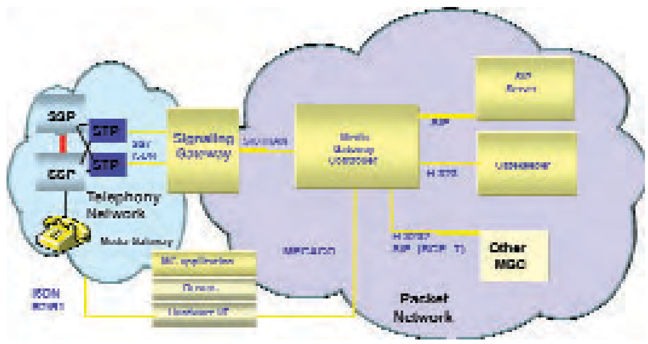


Figure 1

Media gateway: A brief history

One of the strategic steps taken to realize next-generation communication solutions based on the voice over packet was the clean separation of the logically distinct functions of call control (switching) and media (bearer). It was realized early on that a monolithic gateway network element that straddles the legacy PSTN network and the upcoming packet network would have issues of scalability, single point of failure, performance, and others. It would also tend to discourage competition, as the complexity of this network element creates a prohibitive entry barrier.

Therefore the TIPHON group of ETSI proposed the "decomposed" architecture initially in 1999 to enable competition and vendors to focus on their core offerings. It also allowed the possibility of best-of-breed networks with open published interfaces among the gateway components allowing network elements to interface with each other.

The result of the decomposition of the monolithic gateway is depicted in the softswitch network model and elements in Figure 1. The signaling gateway (SG), MG, and the media gateway controller (MGC or softswitch/call agent) are the three logical elements that form the decomposed gateway in the next generation voice over packet architecture.

The device control protocols MGCP and MEGACO have been defined to control media related operations between the MGC and the MG and the SIGTRAN suite of standards has been defined to manage signaling between the SG and the MGC.

Media gateway functional requirements

Before examining development strategies, it's useful to review the functional requirements for an MG. We'll discuss generic requirements common to all MGs, as well as specific requirements for particular markets or applications.

At its core, an MG is simply a translation device that converts media traffic from circuit to packet and vice-versa. On the circuit side is some type of telephone switch such as a Class 5 switch, while on the packet side there is usually a router. Therefore, the gateway must contain standard circuit interfaces and signaling, standard packet interfaces, and transcoding from PCM to a packet format such as G.711 or G.729. In addition, the MG must contain a signaling interface to the MGC.

Within each of these basic functions, there are numerous variations depending on existing network elements, the type of service being provisioned and regional factors, to name a few. On the circuit side, connection to a central office switch will typically require T1/E1 interfaces with Primary Rate ISDN or channel associated signaling (CAS). Higher-density trunking gateways may require T3 interfaces. Many switches in Asia, Latin America, and the Middle East use tonal R2 signaling with many variants from one country to the next. Last-mile environments such as voice over cable and wireless local loops usually require loop signaling such as GR-303 in North America and V5.2 in the rest of the world. Emerging applications such as IP Centrex may require very specialized translation of legacy switch functions across the IP network to enable, for example, transparent call-forwarding between POTS and IP phones.

On the packet side, the picture is somewhat brighter since IP networks, unlike circuit-switched networks, don't have a century of country and vendor-specific variations. Here Ethernet is the physical interface and RTP/RTCP is the standard packetization method. There are a handful of ITU-T standard voice coding methods such as G.711, G.723.1, or G.729A that are now widely supported. Control interfaces from the MGC or call agent are more of an evolving area, but by and large the industry has coalesced around two fairly similar approaches: MGCP and MEGACO/H.248.

And of course an overriding variable across all systems is scale. Even excluding very small enterprise and residential gateways, carrier-level gateways can range from around 100 channels to tens of thousands of channels, depending on where they are in the network.

For gateway vendors, it would be nice to carve out a narrow subset of all these requirements and focus on a particular market segment. In practice, however, this has proved difficult to do. For example many networks are international; therefore, a single operator may require multiple circuit-switched signaling variants. Similarly, points of presence vary in size so that an operator will require different scale gateways, but will need them to interact with common switching elements and operations, administration, and management interfaces.

Role of software in the media gateway

Within this next-generation voice over packet architecture, the role of software becomes critical and complements the accompanying hardware to fulfill all requirements.

The various open interfaces defined to allow the seamless and full interoperability between network elements i.e. Sigtran, MEGACO, SIP, and H.323 are all realized through software control. The rules of the syntax, messaging, transport, etc. are all processed through software.

Further, the semantics of these protocols, their interworking, and their processing is also managed mostly through software at the network elements. As an example the Nodal Interworking Function (NIF) at the SG allows the SS7 ISUP messages to be carried between the legacy MTP3 and the Sigtran M3UA. The call control at the MGC/softswitch allows the Signaling termination and processing needed for completion of the call.

In the same vein, there are several parts to the manner in which software helps fulfill the required functionality of the MG.

Let's consider the case of a MEGACO/H.248 based media gateway. With the 3G and DSL domains adopting MEGACO as their protocol of choice and with its inherent design advantages over earlier protocols like MGCP that allow it to manage multimedia requirements and ATM networks, the MEGACO protocol is fast gaining momentum to help implement next generation network solutions.

First, the software starts at the edge of the media gateway by helping to implement the basic syntax of the MEGACO/H.248 protocol, enabling an MG to respond to commands issued by an MGC. This protocol defines the syntax of the message set, its structure, and the method of transport.

Second, the software in the MG implements the semantics of the MEGACO protocol. This constitutes the processing of the various messages received from the MGC and its handling. The resulting actions involve interactions with the platform and the hardware that comprise the MG, plus the generation of responses to the MGC. The related software element is referred to as the MG application.

Finally, there is a substantial amount of software in the MG whose function is to implement peripheral aspects like configuration, manageability, trace, and other such modules required for the realization of the full network deployable MG solution.

Further, where carrier class features such as high availability, redundancy, and failover are required by the service provider, the combination of software and hardware in the MG need to work in harmony to ensure uninterrupted service availability.

Strategies for MG software implementation

The strategy to implement the software becomes crucial in such a scenario where substantial part of the MEGACO protocol related action, interactions with the MGC, and the profile of the MG itself depend upon the manner in which the software component is developed. Some of the key challenges and choices faced by the developer to realize an effective MG solution include:

- Scale in features: The base protocols for MGCP and MEGACO are easily extensible with respect to features. They utilize the concept of a package, which extends functionality of the MG without changing the base protocol. However, this flexibility places additional demands upon the underlying software architecture of the MG. It is imperative that the core software modules be designed in a modular fashion so that it is not difficult to add new features in the form of packages. These functional extensions should be possible without requiring any major changes to the other core software components in the MG.

  Simultaneously, the MG may also need to manage a heterogeneous set of terminations (e.g. POTS, ATM, IP) with each requiring unique feature support (e.g. different services). The software should be able to model these features both from the protocol and hardware operation perspectives. The model should also be scalable for multiple sets of terminations.
- Easy customization: Scaling is an important part of customization. Other important customization requirements are the ease of OS (operating system) porting, hooks for incorporation of application-specific features (e.g.: resource

selection policy) and transport layer processing (e.g. IP, UDP, TCP, SCTP, MTP3B). These features must be an integral part of the software components of the MG.

- Migration from MGCP to MEGACO or vice-versa: The MGC-MG interface has gone through various levels of maturity before reaching the state it is in today. The road to maturity has seen many milestones, the two most important ones being the development and definition of the MGCP and MEGACO protocol interfaces. There are major organizations that endorse one protocol or another, and as a result both protocols have found market niches. While MEGACO has an edge over MGCP in terms of it being more flexible and seems more future-safe (due to its handling of multimedia and IP/ATM choices), MGCP deployments are prevalent in emerging market niches such as voice over cable. Therefore it becomes important for OEMs/ ISPs to develop generic gateway software components that can ensure smooth migration from one protocol to another.
- Scale in density: MGs can have different profiles – a residential gateway on one end to a trunk office gateway on the other end of the spectrum, with a whole array of media gateways in the middle i.e. for small businesses, enterprises and for access networks. In telecommunication terms this means scaling the MG from 2-3 terminations to a few thousand terminations or even more. The MG software should have the provision to easily scale the density of the modeled components – virtual (protocol defined) and real (hardware defined). This is important for the effective utilization of the current resources on the platform.
- Seamless fit to hardware: Among all the logical elements of the gateway (i.e. MGC, SG, and MG), the impact of corollaries to Moore's law are most applicable on the MG where there is typically a substantial investment in underlying hardware in addition to the software. The complex and performance-effective hardware components of new generation MGs need sensitive software. The software components should be flexible enough to offer a lightweight abstraction layer between the software and underlying hardware components such as voice over packet gateway cards.
- Mapping of the control protocol to underlying hardware operations: Almost all protocol messages directed to an MG request operations that will be executed by using the combination of the MG application software and underlying hardware elements such as digital signal processors (DSPs) and packetization engines. It is important for the software modules to ensure that the protocol messages are processed to meet both protocol and hardware requirements. For example, this may entail mapping the operations to enable the setting of parameters on the hardware or the modification of the properties related to media streams.
- Distribute on chassis/board: An important feature of higher end MGs is to offer extensibility in the form of a multi-chassis and multi-slot distributed architecture. Consequently, the software also needs to be extensible in a similar fashion. For example, the software model needs to uniquely identify the chassis/board and the individual elements on each board. This allows the software to transition from managing a single board (or even a Single IC) solution on up to more complex multiple board and chassis MG systems.

Hardware strategies

A useful way to view hardware is as the packaging for the software. That is, the software defines the functionality of the system and then you choose hardware to achieve desired attributes such as system capacity, physical size, how the system will be serviced in the field, high-availability strategies, and manufacturing cost. If you accept the notion of hardware as packaging for software, then it follows that two elements will weigh heavily in your hardware strategy:

1. The degree to which the hardware has been or can be integrated with the key software elements
2. The degree to which it gives you flexibility in the above attributes: scale, size, HA strategy, serviceability, and cost

Using these criteria, open systems and CompactPCI are an attractive approach for hardware. CompactPCI is a relatively mature standard; this has resulted in a competitive market for DSP resources, network interfaces, switches, and single-board computers, as well as other system elements like racks and power supplies.

For MGs, much of the core functionality is typically contained in a VoIP resource board such as Brooktrout Technology's TR2020. These products, now in their second or third CompactPCI generations, offer some or all of the following features:

- DSP operations – voice compression, G-series, and GSM transcoding, echo cancellation, voice activity detection, and silence compression
- Packetization – using RTP/RTCP as well as T.38 UDPTL for fax
- Onboard telephony – multiple T1/E1 or T3 interfaces with signaling firmware for ISDN, R2, V5.2 and other variants
- Advanced features – for example VQmon voice quality monitoring
- Onboard microprocessors – such as PowerPC chips, to perform signaling, packetization, and advanced functions on the board, reducing host system loading, and board-to-host transfers
- Flexible programming interfaces under popular operating systems such as Linux, Solaris, and Windows

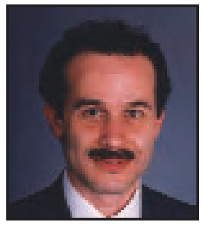
At the system level, there has been a good deal of promising work among vendors to ensure that their respective technology building blocks will work together in a system. For example, Brooktrout Technology and Hughes Software Systems have built proof-of-concept MG prototypes using Brooktrout's TR2020 and Hughes' MEGACO MG Toolkit.

For all its advantages, CompactPCI also has its limits. But a major attraction of CompactPCI is that it exists not alone but within a continuum of relatively compatible architectures, with PCI on the low end, and PICMG 2.x and 3.0 extending the high end. Here is where the developer can really obtain flexibility in scale. CompactPCI's sweet spot is in the range of several hundred to several thousand ports. PCI ranges from a single T1/E1 up to several hundred ports. PICMG 2.x refers to switched fabric backplane standards such as 2.16, which don't directly increase the scale of the system but do improve the HA characteristics of the system such that the high end of the CompactPCI range is more tenable. PICMG 3.0, or AdvancedTCA, now being specified, incorporates the switched fabric characteristics of PICMG 2.x while ultimately adding considerably to board real estate and power management, thereby enabling systems that will scale to tens of thousands of ports.

The key point here is that, through this continuum of standards and form factors, it's possible to scale from very small to very large systems while substantially preserving one's software investment. This is made possible by using open systems building blocks that have consistent firmware features and APIs across a variety of scales and form factors. Open Systems building blocks come in a highly scalable range (see Table 1).

Conclusion

The MG is a key component in the realization of next-generation networks and the deployment of value added services. The MGs are present in the network in many forms like residential gateway, IADs, IP PBX's, access gateways, soft phones, trunking gateways, etc. Therefore, MG requirements continue to have high complexity and frequent changes. Fortunately, open system software and hardware building blocks can provide a foundation and evolve with the changing standards, allowing system developers to focus on their unique target applications.



***Joshua Adelson** is Director of Market Development for Brooktrout Technology where responsible for marketing programs and business development for the voice over packet infrastructure market.*

***James Rafferty** serves as the Senior Product Manager for IP Telephony at Brooktrout Technology. Rafferty is also active in the IP Telephony standards process through his work with the MEGACO working group of the Internet Engineering Task Force and related contributions to ITU-T Study Group 16.*

***Vivek Pachaury** is currently Product Line Manager for the Voice over Packet Division of Hughes Software Systems.*

	PCI	CompactPCI	PICMG 2.x	PICMG 3.0
Board Height	~2U	6U	6U	8U
Economic Scale	Tens to hundreds	Hundreds to thousands	Hundreds to thousands	Thousands to tens of thousands
HA Strategy	Redundant system	Redundant board w/hot swap	Redundant board w/fail over	Redundant board w/fail over
FRU	Board or whole system "pizza box"	Board	Board	Board
Market Maturity	High	High	Moderate	Emerging

Table 1

For more information, please contact:

Brooktrout Inc.
250 First Avenue
Needham MA 02494
Tel: 781-449-4100
Fax: 781-449-9009
E-mail: info@brooktrout.com
Web site: www.brooktrout.com

Hughes Software Systems
Bangalore, Prestige Opal, 146, Infantry Road
Bangalore 56001
Tel: 91-80-228 6390
Fax: 91-80-228 6393
E-mail: info@hssworld.com
Web site: www.hssword.com