IP Telephony
Contact Centers
Mobility
Services

WHITE PAPER

Enterprising with SIP — A Technology Overview

Version 2

October 2005

This White Paper explores the business benefits, operational models, and enabling technologies for the multi-site contact center. Although the dominant 1990's multi-site contact center technology model served enterprises well, the assumptions behind that model are now obsolete. The new realities drive a new multi-site paradigm that is flat, consolidated, and extended.

This new architecture enables new models for running contact centers that dramatically reduce costs and improve responsiveness to marketplace changes.

Table of Contents

Section 1: Executive Overview

The vision that Avaya has for converged voice and data communications is to create a network environment that supports maximum enterprise efficiency. Session Initiation Protocol (SIP) [18] is a key technology for this evolution towards a converged communication architecture. SIP as an enabler of converged communication applications has been able to gain industry-wide acceptance by placing control of the communication across distributed networked entities and by providing a means of integration at various levels in the protocol stack. Such integration is now possible in a multi-vendor environment with multiple applications across a wide range of industry sectors. Converged communications that is driven by ubiquitous IP network connectivity defines the next stage of evolution in enterprise communications.

SIP is an application layer, peer-to-peer communication protocol that facilitates openness, simplicity, flexibility, and reuse in an "All IP" Architecture. This paper offers a look at how Avaya views SIP in enterprise evolution. It also describes SIP technology, elaborates on the role of SIP in realizing converged communications, and describes details of the SIP protocol and associated technologies.

Section 2: Evolution to Converged Communication

Avaya sees the evolution of IT infrastructures in three phases as shown in Figure 1. Enterprises will evolve portions of their infrastructures from one phase to the next according to their business needs and will often be in more than one of these phases at the same time.

In the ***traditional*** phase, enterprises have separate infrastructures for voice and data networks, with time division multiplexing (TDM) for voice and IP for data. In the ***converged networks*** phase, enterprises build out their IP networks to leverage a common infrastructure for both voice and data. This enhances the IP network to meet enterprise-class criteria: improving quality of service (QoS) and increasing the reliability of real-time, mission-critical business and communication applications.

As enterprises become more distributed and business performance needs dictate enhanced user capabilities, ***converged communications*** applications will be deployed. The essence of converged communications is modularity: network components and applications that can be used over a wide variety of systems. As solutions become more modular, their services can be deployed in a greater number of configurations and more easily integrated into multi-vendor environments, which can lead to increased network flexibility and cost efficiency. Avaya is taking the lead in modularization of its software and systems into an open communication architecture to help organizations smoothly transition to converged communications for a more adaptive enterprise.

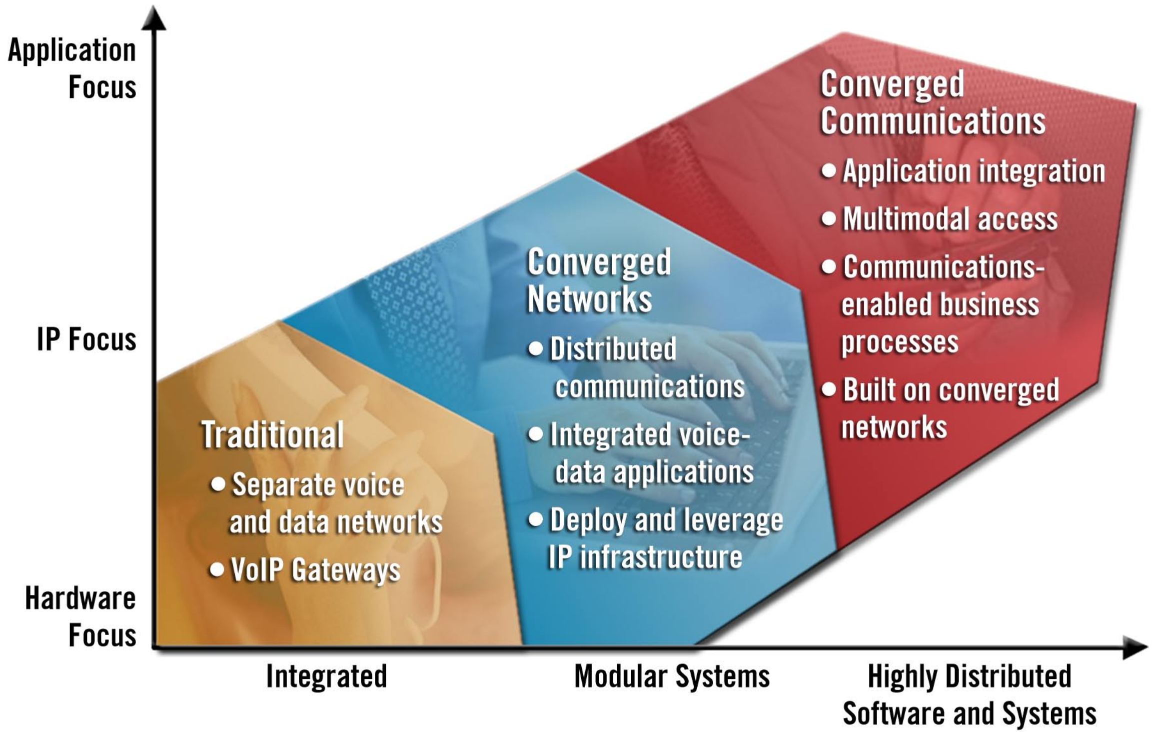

Figure 1 Evolution to Converged Communications

As enterprises become more virtualized and the needs for ubiquitous end user capabilities become a critical business imperative, enterprises will begin to evolve to a ***Converged Communications*** phase. In order to leverage their existing investments in various technologies, businesses will naturally evolve parts of their infrastructure from one phase to the next. As such, they will often discover themselves in more than one of these phases at the same time. For example, the majority of enterprises today are transitioning between Traditional and Converged Networks, with a few visionary enterprises starting to transition to Converged Communications. Due to this gradual migration, it is essential that an enterprise's solutions are evolutionary enough to accommodate existing infrastructures and investments, while at the same time provide a new foundation for deployment of new applications and services.

Section 3: Avaya Converged Communication Evolution with SIP

Session Initiation Protocol (SIP) is a simple protocol that facilitates peer-to-peer communication sessions. Users (or, in general, any addressable entities) in a SIP framework are identified by Universal Resource Identifiers (URI). Each such Internet-style address (for example, sip: johndoe@avaya.com) maps into one or more ***Contacts***, each of which typically represents a device or service at which the corresponding user may be reached. Examples are phones, desktop multimedia clients, instant message accounts, email accounts and so on. The SIP framework is responsible for routing a request for a peer-to- peer session addressed to a given URL to one or more appropriate contacts for that URL. The framework may utilize information about the preferences, presence and location of the user identified by the URL, to determine the most appropriate contacts. The protocol also provides mechanisms to specify the type of session that is requested as well as means to change session parameters.

In current paradigms, solutions are intricately tied; for example, users call each other using a set of voice application related protocols, and users send Instant Messages (IM) using another set of protocols. The situation is further complicated when multiple vendors employ proprietary protocols (or protocol extensions). In addition to the complexity of Enterprise CIO operations forcing communication into specific silos, this reduces end user productivity. A SIP-based solution changes that paradigm by allowing users to indicate the fact that they would like to communicate to another user, but using a user's preferred mode. SIP-enabled Avaya converged communication makes that easier for enterprises as described in "*Evolving to Converged Communications with SIP*." The Avaya Converged Communication Solution is geared towards helping Enterprise CIOs to benefit from the impact of SIP based converged communication in the broad areas of Voice over IP (VoIP), Unified Communication, Customer Relationship Management (CRM) solutions. It lays out the foundation for migrating to an enterprise SIP network.

Though SIP is gaining popularity within the communication industry as a "telephony protocol," in reality, SIP is applied to a wide variety of communication sessions such as voice, video, instant messaging, presence, conferencing and real-time collaboration. SIP has become the industry-wide standard mechanism to achieve real-time interactive converged communication services. SIP provides a practical method of service integration across multiple networks—such as Enterprise/Service provider network and TDM/IP network—using multiple modes in a media independent way. In this way, SIP changes the focus of communication from the modality of the communication to a user level representation of a session. SIP is also a vital component of next-generation enterprise and carrier wireless networks.

With these properties, SIP is an important enabler of converged communications, where the emphasis is on ubiquitous end user capabilities and end user control. SIP is to converged communication what HTTP is to information exchange for the World Wide Web (WWW)—it makes the communication infrastructure transparent to the end users and enables ready access to many modes of communication. Through its use of the URI, SIP enables a communication request to be handled in the same manner as an HTTP request, creating a natural solution for integrating communication services and enterprise applications.

Consider Mr. John Doe's work life, as shown in Figure 2. His typical day involves use of several phones: office phone, cell phone and home phone for his virtual office. He uses multiple voice messaging systems, email and an Instant Messaging service from a respected ISP, as well as corporate email with a remote access capability, and a fax machine to received legal documents with original signatures.

On a good day, all of John's contacts, devices and services work together, but John must provide context to each communication session and manage it separately. John also has to deal with multiple billing systems and voucher appropriately through various expense management systems. But what if it all worked together in a seamless system, with a multi-modal device interacting with the multiple applications that John uses? What if John had access to a communication system that allowed communication session requests to be routed based on John's presence and personal preferences? SIP enables this graceful application integration to help John handle communication complexity in his day-to-day life.

Figure 2 A schematic representation of John Doe's work environments

Now consider an IP phone call that can be generalized to most of the participants in a communication session and can be broken into functional steps as in Figure 3. Such a functional breakdown accommodates a distributed communication model that can be represented in the form of a Uniform Resource Identifier (URI) and the devices can be represented as contacts. Like the web model, various Application Service Providers can add value-added services and applications, increasing the effectiveness of communication. SIP has an impact on peer-to-peer communications analogous to the impact of HTTP on information exchange over the Internet.

HTTP facilitates a client server environment where a standards-based client can request information from a web server using a simple text-based protocol. This has led to the prevalence of web servers, offering various kinds of services, all easily accessible over the Internet. A simple language such as HTML enables a flexible and simple means of creating content on the Web. Similarly, SIP allows communication agents to request interactive communication services from peer-connected entities over an IP network. Thus it has the potential to disaggregate today's monolithic communication applications into standard components that can be re-combined into powerful new distributed communications services.

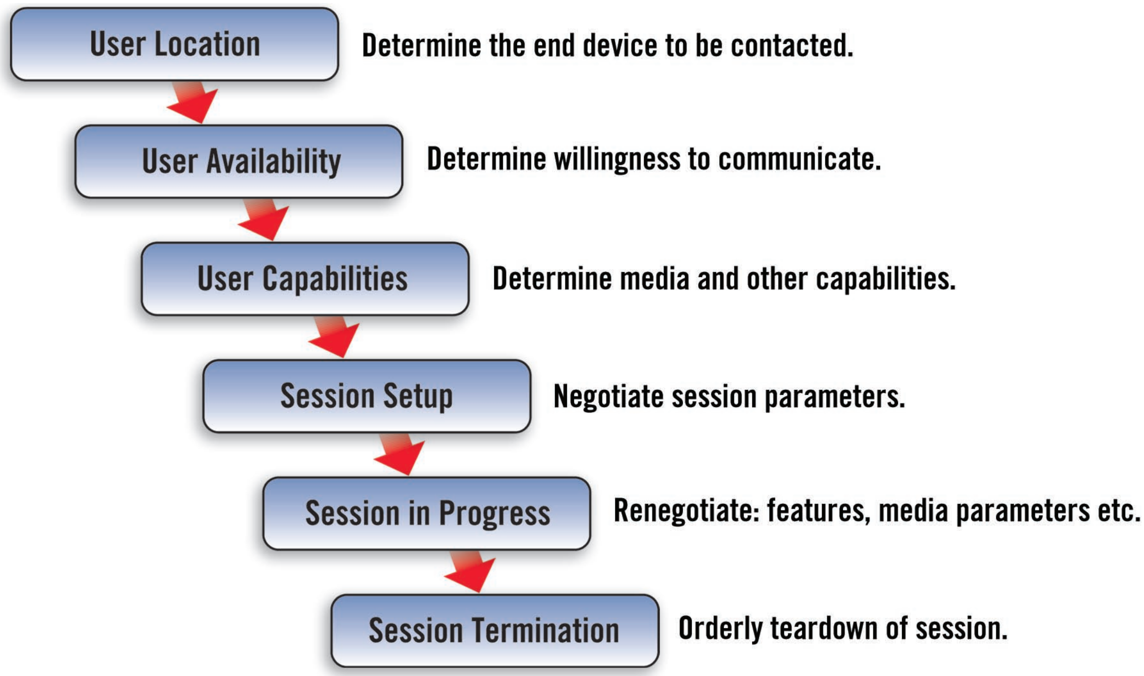

Figure 3 A generic communication session

Section 4: What is SIP?

SIP is an application layer Internet protocol for establishing, manipulating and tearing down communication sessions. It is important to realize that SIP is more than a telephone call setup protocol. The protocol is designed to be extensible; extensions of note include instant messaging via text channels and a publish/subscribe mechanism for events such as presence and availability information. SIP has been developed in the Internet Engineering Task Force (IETF) by common participation from various vendors including Avaya. The IETF community took Internet standards like HTTP as a model and as a result a text-based request/response peer-to-peer model is at the heart of the SIP protocol.

SIP is designed to leverage a broad spectrum of existing protocols for various applications including IP telephony (see Figure 4). In addition, it is extensible and thus can accommodate new protocols and mechanisms as they are designed and accepted in the Internet community. As a result, SIP is adaptable to many scenarios where traditional telephony protocols play. However, it is not specifically designed for feature equivalence with PSTN capabilities. Its applicability lies in an application and service driven network that goes well beyond telephony. In particular, it enables a host of negotiations relevant to any communication session (whether voice over IP or not) including capability exchange, request routing and rerouting, forking, retransmission, and so on. However SIP is not designed to be a one-stop shop for protocol needs. SIP is used in combination with other network protocols as well as application-layer technologies to provide end-to-end functionality. Some of the protocols and technologies that SIP leverages are: Session Description Protocol (SDP) [17], Resource Reservation Protocol (RSVP), Transport Protocols like RTP [16], SMTP, UDP/TCP, DHCP, Voice XML [13], XML [14], HTTP, WSDL [12], UDDI [15], Simple Object Access Protocol (SOAP) [11], etc. SIP provides true Web Telephony Integration (WTI) and integrates a Web-centric framework to incorporate more applications and thus allow a true application development environment to be coupled with enterprise telephony communication systems. The flexibility of SIP application deployment allows applications to be deployed on the network where it makes sense.[6]

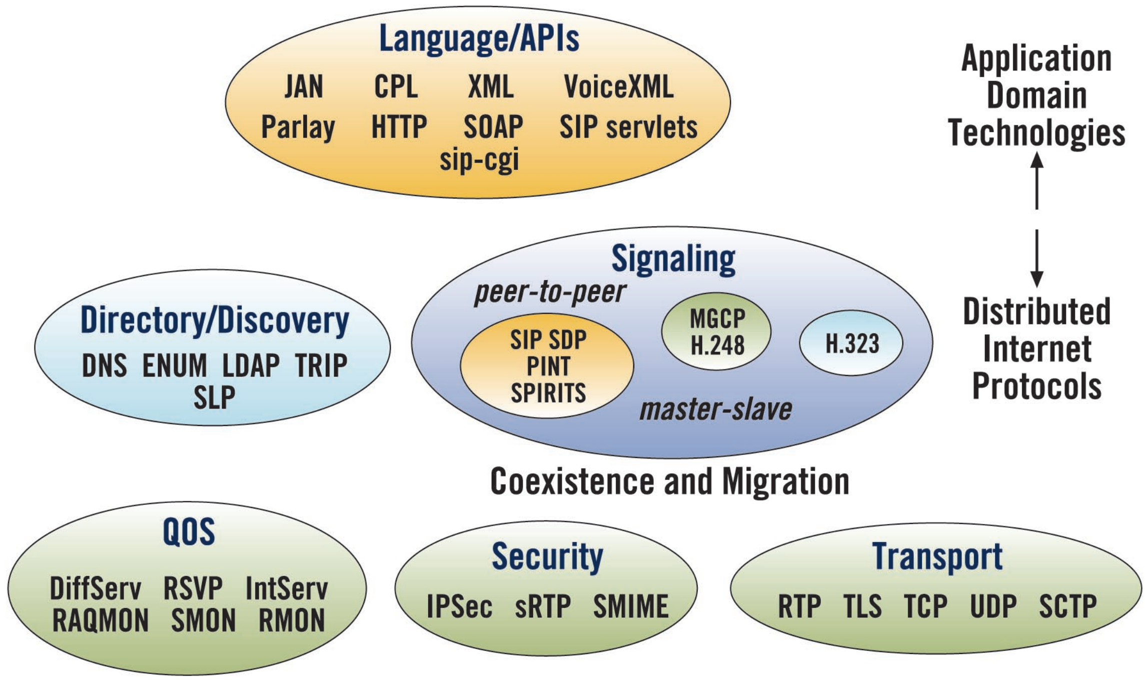

Figure 4 The relationship of SIP to other protocols

In keeping with the IETF philosophy of defining simple protocols with powerful functionality, the basic SIP protocol follows a peer-to-peer based architecture containing a small set of different methods or types of messages.

SIP clients invoke methods (as in HTTP) on the server, and the SIP responses generated by the servers that receive requests are HTTP-like. The methods defined in the basic SIP specification are:

- ***INVITE:*** to invite a user to a communication session.
- ***ACK:*** to acknowledge the final response of an INVITE request.
- ***BYE:*** terminates the session between two users in a call.
- ***OPTIONS:*** to query a server about its capabilities, but does not set up a call.
- ***CANCEL:*** to cancel a request pending with a server. This is useful, for example when a proxy server has forked a request and some of the many endpoints that the request was sent to respond, and the request-initiating client wants to cancel the search for the others.
- ***REGISTER:*** to register an address with a SIP server that will subsequently route requests to the registered entity. A user agent may send a REGISTER request to a preconfigured server address, by normal SIP routing to the SIP server for the entity's domain, or to a well-known multicast address ("all SIP servers").

SIP responses generated by the servers that receive requests are HTTP-like, again. These are grouped as below:

1. ***1xx:*** provisional, e.g., “180 ringing”

2. ***2xx:*** success, e.g., “200 OK”

3. ***3xx:*** redirection, e.g., “302 moved temporarily”

4. ***4xx:*** client error, e.g., “404 not found”

5. ***5xx:*** server failure, e.g., “501 not implemented”

6. ***6xx:*** global failure, e.g., “604 does not exist anywhere”

Session Processing

At a basic level, four of the SIP methods are necessary to setup, change or terminate a session. These are INVITE, ACK, BYE and CANCEL.

Session Routing

SIP specifies the routes of a session by mechanisms such as via, Record Route (RR), forking, etc. Sessions can also be routed based on application rules and profiles like presence.

Presence

Presence information is propagated using a subscription-based event notification model supported by messages like SUBSCRIBE and NOTIFY.

Capabilities and Preferences

An end device can get information about an end-point’s capabilities in two ways. First the call-agent responds to an INVITE (which might offer several different connection options) with a message that accepts one of the options or rejects the call if it does not support any of the options. Second, the caller can explicitly query the agent to be called using a typical message. An OPTIONS message asks an endpoint to specify its capabilities and preferences.

4.1 SIP Architectural Components

SIP protocol architecture is comprised of two major architectural elements: SIP User Agents (UA) and various Network Servers as outlined in Figure 7. The two classes are usually further divided into several types of entities.

1. ***User Agents (UA):*** User Agents are composed of a User Agent Client (UAC) responsible for issuing SIP Requests and a User Agent Server (UAS) responsible for responding to various requests. End devices like IP Phones, Voicemail servers, etc. act like User Agents in a SIP network. User Agents communicate with other User Agents directly or via an intermediate server.

2. ***Registrar:*** Registrars are used as a repository to record SIP URLs and associated contacts. Most of the User Agents REGISTER with the registrar and the registrar stores the registration information in a repository-type database, directory, or location service via a non-SIP protocol.

3. ***Proxy Servers:*** Proxy Servers perform application level routing of SIP requests and responses. A proxy server can be call and transaction stateful or stateless, depending upon whether or not it remains in the session processing path for the entire duration of the session. Though stateless proxies tend to achieve higher scalability, stateful proxies are required for billing purposes. Furthermore, SIP differs from other signaling protocols in that it allows a session request to fork, so that a session request can trigger a server to send out many requests to different destinations. As a result,

SIP proxies can be forking or non-forking in nature. The ability to accommodate forking either at once or in sequence further enriches the Proxy Servers' role in the SIP architecture. This feature supports a number of advanced telephony services, such as call forwarding to voice mail and automatic call distribution (ACD). It also supports user location one number portability as demonstrated by Extension to Cellular applications.

4. ***Home Proxies and the Location Service:*** In SIP networks, proxies have various roles. One of the roles of a "home" or "user feature" proxy is to provide a Location Service for local subscribers. Messages in SIP are usually sent to a user's Address-of-Record (AOR), or public address, such as sip: *joe@company.com*. It is the job of a user's home proxy to route that request to one or more *contact addresses* for the user. For example, Joe may be currently registered at one or more of the following addresses otherwise known as Fully Qualified Domain Names (FQDN):

   - His Avaya SIP Softphone, running on his PC, capable of receiving SIP instant messages or voice calls. The contact address may be something like sip: joe@pc1.usae.avaya.com or sip: 23200@135.8.45.63.
   - Avaya SIP Phone or a traditional Digital Phone connected to an Avaya Communication Manager. This contact may look like sip: 2200@acm1.usae.avaya.com .
   - His cell phone, reachable via SIP using a "tel" URI, such as tel: 17325551234, which can be routed to an Avaya Communication Manager for delivery over the PSTN.

The Location Service is an abstract concept in SIP and it is up to the proxy to provide this service in an implementation dependent way. Some of the mechanisms that can be used are given below.

- ***Simple Database or Directory:*** This is one of the simplest methods. The registrar and proxy both have access to a simple database, LDAP directory, or some other data store. When a SIP user agent registers one or more contacts, these are written into the data store by the registrar. Usually, the contacts are registered with a ***q-value***, which is simply a way to prioritize the order in which contacts are tried. The proxy then retrieves this list, sorts it by q-value, and attempts to locate the user at one of the contacts. The contacts may be tried sequentially or in parallel (known as ***forking***). The proxy will typically forward back the first 200 OK messages when one contact accepts the request, or it may forward back other final responses such as "486 Busy." If the public address is not found, the proxy will send back a 404 Not Found response. If the address is known, but no contacts are currently registered, it will send back a "480 Temporarily Unavailable" response.

- ***Application Logic:*** A proxy may support a "plug-in" module approach that lets other applications write more complicated logic. It may then map incoming requests to the application configured to handle it. For example, an application module may check a user's calendar, or execute some other application-defined logic to determine the best way to contact the user. Other examples include a CRM application that may map a request to *sip: tech_support@company.com* to the next available representative, or to a representative with a certain skill set.

- ***CPL Scripting:*** Call Processing Language (CPL) [19] is a standard scripting language defined by the IETF that describes how to process inbound and outbound requests. The CPL language is protocol neutral and has a language binding for SIP. It is designed to be a simple and safe mechanism suitable for executing user-defined scripts. The SIP proxy is usually separated from CPL interpreter and application logic. The application logic can examine fields in the inbound request such as the To, From, Subject, Priority, etc.; perform location service lookups; and perform routing based on those fields. It also supports time of day routing, such as "send calls between 9am and 5pm to my desk phone, but others to my cell phone." Since CPL is a standard, any application can generate CPL and a compliant proxy should understand it. Some other mechanisms, such as SOAP [11] or HTTPS maybe used between the application that generates the script and the proxy that must execute it. This mechanism is used to "install" or "register" scripts with the proxy and to specify such things as which request-URIs should cause the script to be executed. This allows for system-wide scripts, group scripts or individual user scripts to be located.

A proxy and registrar may use any combination of the above mechanisms to service inbound requests to its authoritative domain.

5. ***Redirect Servers:*** Some SIP servers provide routing by responding with 3xx class (redirection) responses. Redirect servers do not forward requests to the next server, but direct the User Agent (or a previous proxy that supports route recursion) to contact another server.

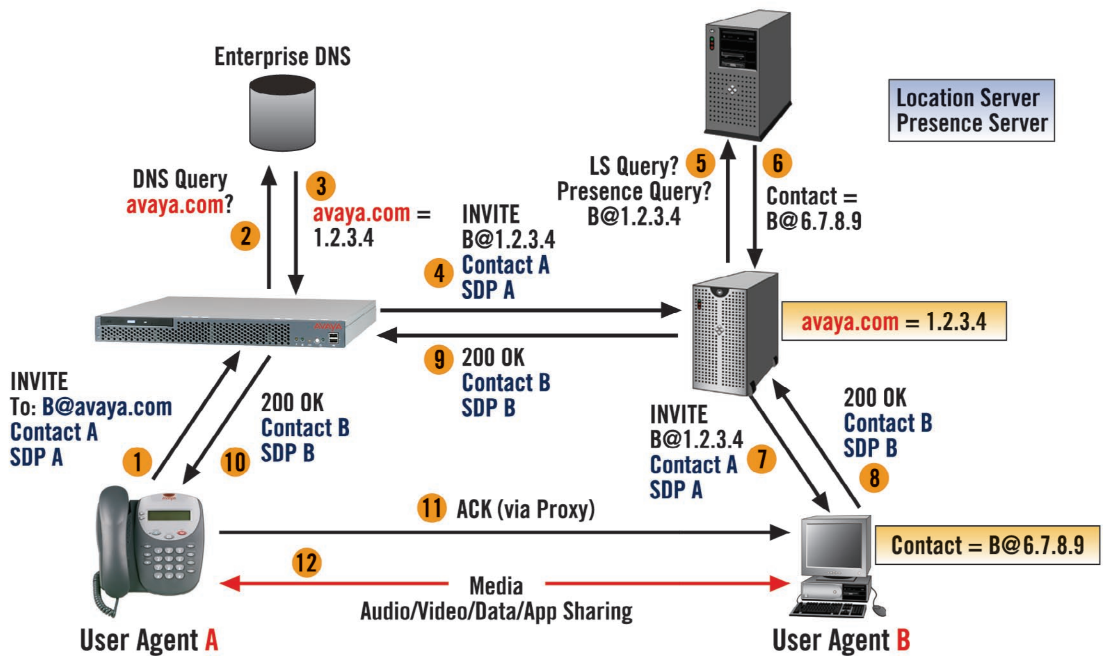

Figure 5 A basic SIP network

A SIP User Agent (UA), Registrar and SIP Proxy Server, along with Location Services and Domain Name Service (DNS), constitute an elemental SIP network that is capable of establishing basic communication like Point to Point Audio, Instant Message, video, etc. as portrayed in Figure 5.

However, a value-added business communication system requires more targeted features. To support needs such as Boss-Secretary relationship, voicemail coverage rules and policies, time of day routing, group calling and uniform dial plan, requires additional intelligence in the network as well as at the endpoints. To provide the functionality of presence and advanced features, a flexible and open SIP network can easily add application and infrastructure servers with appropriate software upgrades. To provide maximum business continuity and enable a successful migration to SIP within an existing enterprise network, CIOs should leverage existing features servers and communication policy servers. Please refer to the Avaya White Paper titled, “Evolving to Converged Communication with SIP” for a discussion on migration to a SIP network. This discussion focuses on investment protection and ease of adding new features to existing business communication features without requiring an expensive forklift.

6. ***Presence Servers:*** A Presence Server accepts presence intelligence, stores it, and distributes it. The presence service has two distinct sets of clients. One set of clients, called PRESENTITIES (for example Producers of information), provides presence information to be stored and distributed. The other set of clients, called WATCHERS (for example Consumers of information), receives presence information from the service. These two sets of clients can be combined in an implementation, but treated separately for the purpose of definition in the model based on functionality.

7. ***Gateways:*** Gateways provide Protocol translation and interoperability with non-SIP systems like H.323, MGCP, ISDN, etc.

8. ***Back-to-Back User Agent:*** The Back-To-Back User Agent (B2BUA) processes requests as a SIP User Agent Server (UAS). It also acts as a SIP User Agent Client (UAC) that determines how the request should be answered and how to initiate outbound calls. Unlike a SIP proxy server, the B2BUAmaintains complete session state and participates in all session requests. Features and applications can be easily built using B2BUAs.

9. ***Feature Servers and Various Application Servers:*** Various other Feature Servers and Application Servers work in the SIP framework that use technologies like Web Server, CPL [19], XML [14], Voice XML [13],Text to Speech (TTS), and RTSP Streaming Media Server.

10. ***Developing SIP Solutions and Applications:*** Many of the SIP entities already mentioned are functional SIP entities and cannot be deployed cost effectively in a practical deployment without bundling functionalities within a server. This reduces operational expenses by:

    - Minimizing the total number of entities that need to be managed within an enterprise network

    - Minimizing possible point of network vulnerabilities as a good design practice

The Avaya SIP Enablement Services (SES), part of the Avaya SIP Enablement Services portfolio, includes the following functionalities (these SIP services are also integrated with Avaya Communication Manager to provide new features as well as support for existing Avaya Communication Manager voice features):

- SIP Registrar

- SIP Proxy

- Presence Server

- SIP Instant Message Gateway

- Support for SIP based Instant Messaging and “click to conference”[12]

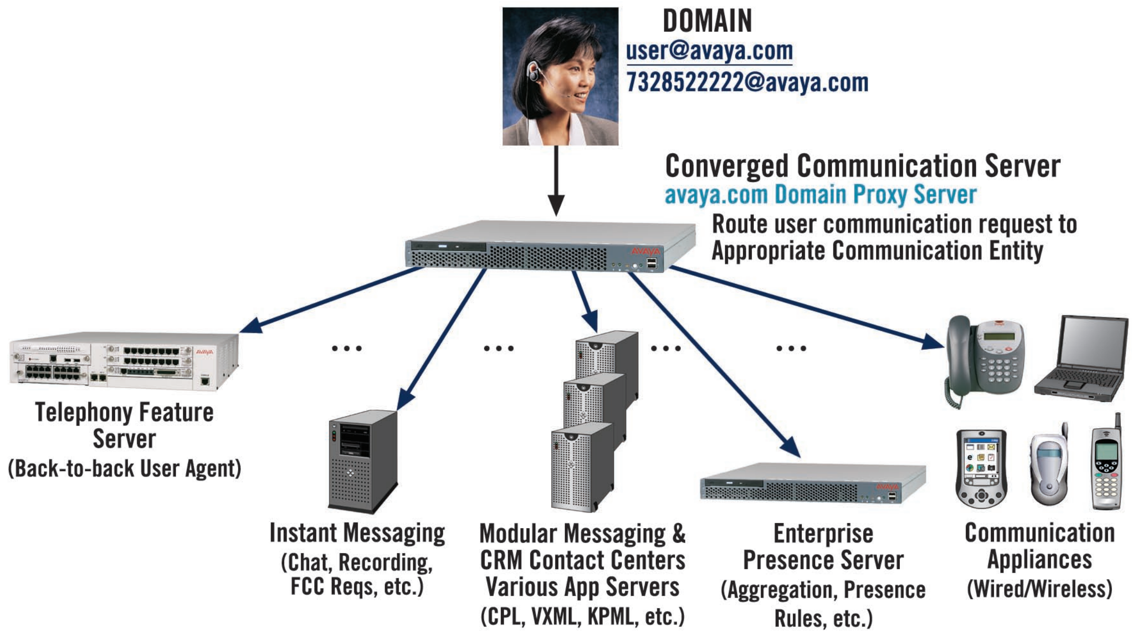

Figure 6 Adding new features to existing business communication features

Avaya Converged Communication Server as depicted in Figure 6 creates a communication value chain that allows Enterprises to add new capabilities within their real-time network such as Instant Messaging and Presence. Such a solution also integrates tightly with Avaya Communication Manager, which acts as a Telephony Feature Server within the solution and provides presence of desktop phones to a designated presence server within the SIP network. Such integration provides business class telephony otherwise not provided by SIP and also accommodate for migration of the embedded base to a SIP network without loss of features and functions usually provided by telephony systems.

Some of the SIP Enablement Services geared towards business communication efficiency are outlined in "*Evolving to Converged Communications with SIP.*" Future application development in a SIP-enabled Avaya converged communication environment can be approached in any of the following ways:

1. **Using standards based SIP, Instant Messaging and Presence protocols**
   Avaya SES supports all required IETF RFCs and Internet Drafts defining protocol interfaces for these areas. Developers can write SIP user agents or "back to back user agent" application servers in any language or platform that supports SIP. SIP stacks, user agent toolkits and server development frameworks are available in multiple languages and platforms. Java developers can use tools standardized by the Java Community Process (JCP) such as JAIN-SIP, JAIN-SIP Lite or Java Servlets.

2. **Call Processing Language Scripting**
   Avaya SES will support execution of CPL scripts by the Avaya SIP Proxy. CPL supports customized routing of SIP requests based on fields of a SIP request (To, From, Domain, etc.), direction (inbound or outbound), time of day, and language. It also supports contact lookups from the location service. Applications can present web pages or graphical user interfaces that let users or system administrators define request handling rules and preferences and generate standard CPL scripts that can be installed and dynamically executed by the proxy when requests arrive. A web-based mechanism will be supported to allow authorized applications to install scripts and specify the conditions and triggers that should result in script execution.

3. **Proxy Plug-in Modules**
   The Avaya SES SIP Proxy supports a module plug-in capability similar to that supported by the Apache Web server. This allows the Location Service, the key component that handles routing decisions, to be customized by application developers. Modules written in either C/C++ or Java can be configured and executed by the proxy during request handling. The interface definitions are provided as abstract classes that can be independently developed and compiled into either Linux shared libraries or Java class files.

These entities are actually functions that may be bundled into one or few physical servers depending on the network configuration, size and scalability needs. These bundled servers will also be distributed logically as well as geographically to provide services across a large enterprise.

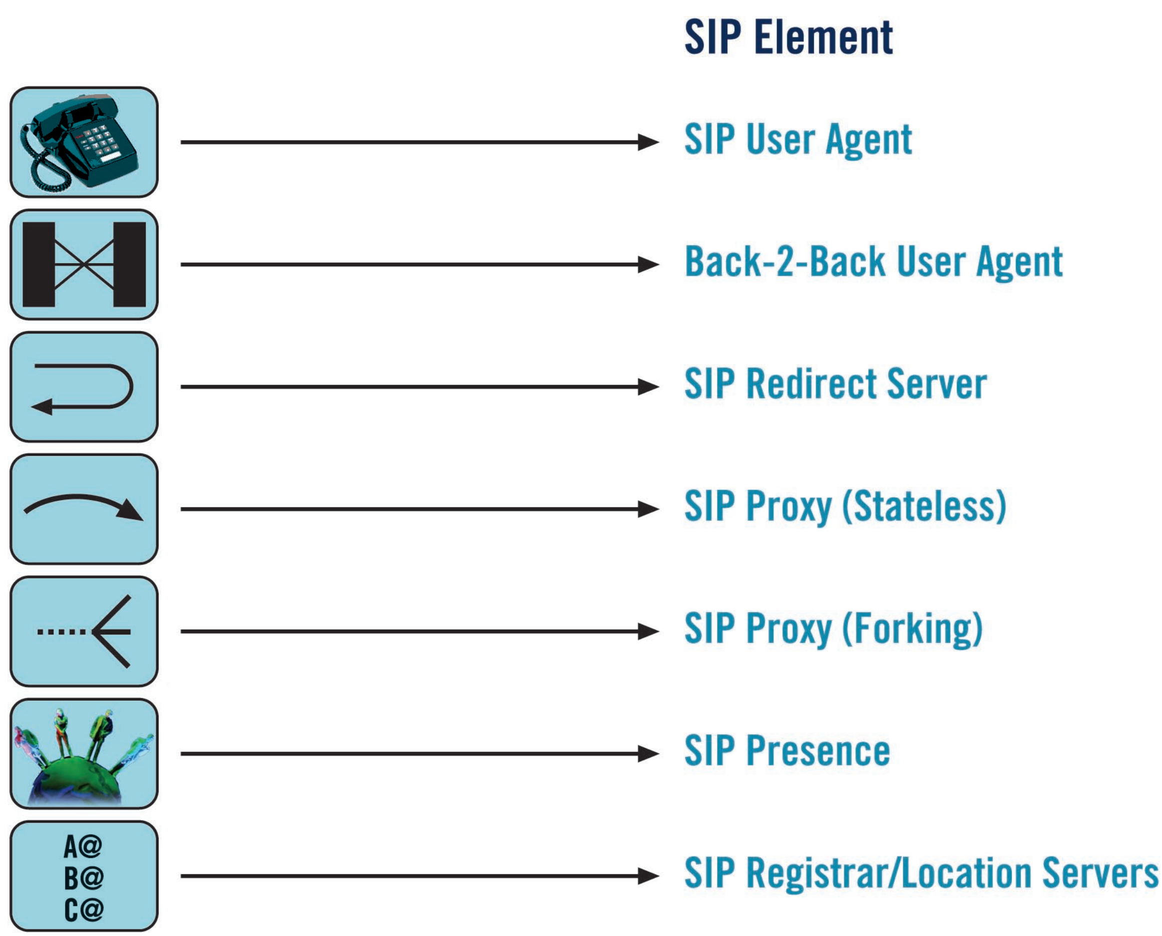

Figure 7 SIP network entities

Section 5: Basic SIP Operations

A few example scenarios that reflect how basic SIP communication works in an Avaya VoIP environment are described in this section. A SIP client realized in the form of an IP phone within a network, like Avaya 4600 Series Phone or Soft Communication Client, sets up a session by issuing an INVITE request. This request contains header fields used to convey information about the session:

- To and From contain the callee's and caller's address, respectively.
- Subject header field identifies the subject of the call.
- Call-ID header field contains a unique call identifier.
- CSeq header field contains a sequence number.
- Contact header field lists addresses where a user can be contacted. It is placed in responses from are direct server, for example.
- Require header field is used for negotiation of protocol features, providing extensibility.
- Content-Length and Content-Type header fields are used to convey information about the body of the message.
- Body contains a description of the session that is to be established.

```
INVITE sip:John@avaya.com SIP/2.0
Via: SIP/2.0/UDP homepc.earthlink.net:5060
To: John <sip:John@avaya.com>
From: Michael <sip:Michael@avaya.com>
Call-ID: 12345678@officephone.avaya.com
Cseq: 1 INVITE
Subject: Business call                          SIP
Contact: sip:Michael@avaya.com
Content-Type: application/sdp
Content-Length: 158
V=0
O=Michael 2890844526 2890844526 IN IP4 officephone.avaya.com
S=Phone Call
C=IN IP4 198.1.102.103
T=0.0                                           SDP
M=audio 49170 RTP/AVP 0
A=rtpmap:0 PCMU/8000
```

Figure 8 Example of an INVITE method in SIP with SDP for audio

5.1 Point-to-Point SIP Call with the Proxy

Figure 9 below shows how two users—sip: Michael@avaya.com and sip: John@bigcompany.com—using two SIP User Agents, would establish a point-to-point session through a proxy server. An example of such User Agents could be Avaya 4602 SIP Phone, SIP Soft Phone, and IP Agent contact center soft phone. In this case the proxy server works to connect two UAs.

1. The user Michael at avaya.com (UAC) initiates a session by inviting user John. An INVITE request is generated and sent to John. As mentioned earlier, the INVITE message contains Session Description Protocol (SDP) parameters that informs the receiving UA about the type of media the caller can accept and where it wishes the media data to be sent (See Figure 9).

2. A DNS SRV record lookup for SIP services at this stage resolves to John's proxy server,"proxy.bigcompany.com." An INVITE request is generated and sent to the server.

3. The server processes the invitation and looks up Michael's contacts in the Registrar.

4. The Registrar returns the host "host@officephone.bigcompany.com," at which the user John is currently located.

5. The proxy server generates and sends an INVITE request to the hosthost@officephone.bigcompany.com.

6. The UAS at "host@officephone.bigcompany.com" asks John whether he wants to talk or not. This may take the form of ringing the phone, blinking a LED, or displaying text in the phone's screen.

7. The acceptance is returned to the proxy server.

8. The proxy server sends the acceptance to the original caller Michael.

9. The acceptance is confirmed by an ACK.

10. After the conversation when John hangs up the Phone, John's UAC sends a BYE to Michael (not shown in Figure 9).

11. Michael's UAS responds to that BYE to end the session (not shown in Figure 9).[16]

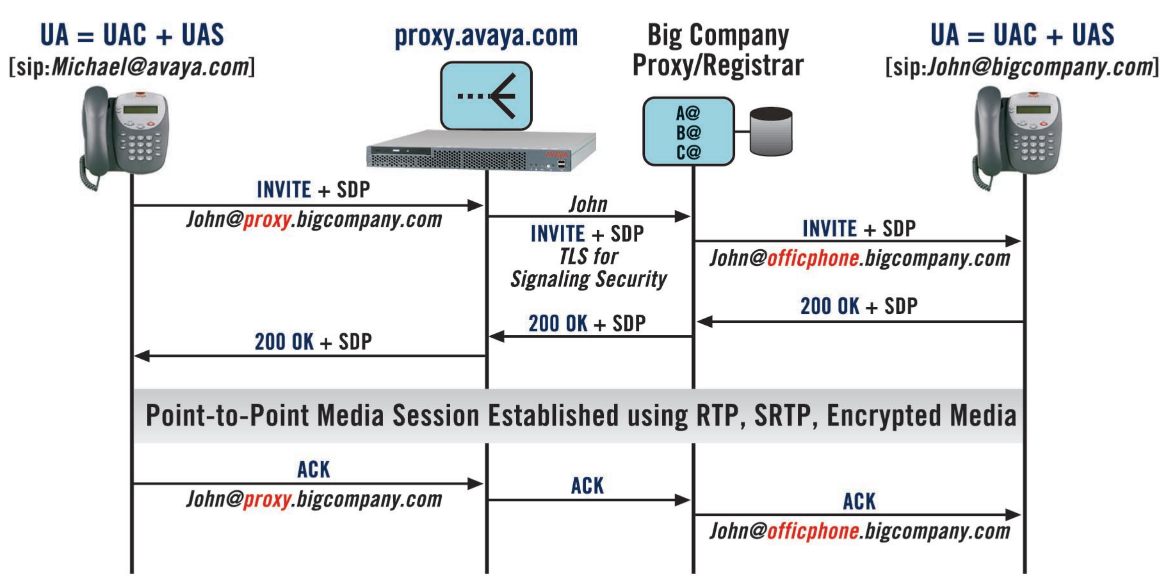

Figure 9 Example of a point-to-point call using a SIP proxy server

5.2 SIP Call to a Digital Phone via SIP-Enabled IP PBX

A critical aspect of converged communications deployment is the need to establish communication to non-SIP entities. Examples include SIP to PSTN Gateway, phone calls to a digital or analog phone, or a SIP Instant Message to a proprietary IM protocol in the public Internet. In such an environment, an intermediary is required to terminate SIP connectivity and map SIP requests to the appropriate protocol to complete a connection. In such an environment, a UA is actually hosted in the intermediary that terminates the SIP connection. The following scenario is an example of a point-to-point SIP call to a non-SIP entity, which is very similar to calls described in section 4.1. However, in this scenario the IP-PBX that controls digital phones or IP-PBX that acts as a SIP to H.323 gateway are examples of intermediaries acting as a SIP UA on behalf of non-SIP entities.

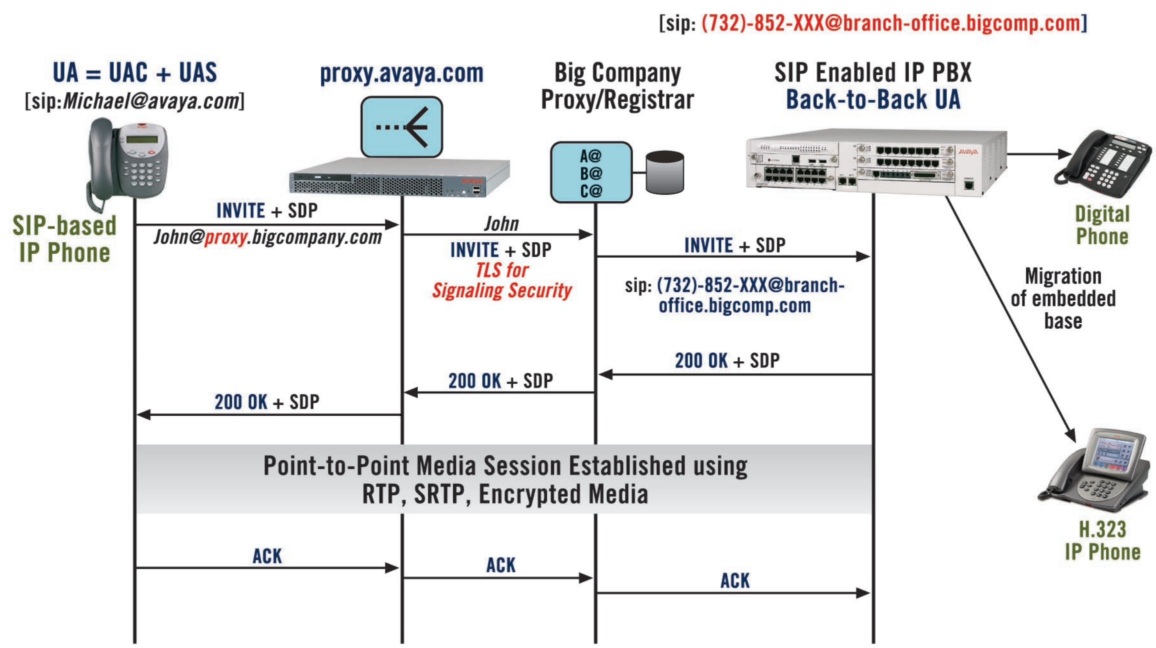

Figure 10 Example of a point-to-point call to a digital phone via SIP enabled IP PBX

5.3 Presence Enabled Point-to-Point SIP Call

Presence is a vital aspect of SIP technology. Presence is the notion that the current state of an entity, particularly its communications state, can be exposed and represented in a standardized, sharable way. Entities so represented need not be human or singular. For example a device status or a User status might be captured as a Presence Status (for example "Phone Status = Off-Hook" or "User Status = Online").Presence for composite entities like groups or shared documents can be similarly represented. Starting with a simple definition of "Online/Offline" status, Presence Status has been extended to include other context information around the status such as disposition (out-to-lunch, away-from-the-computer) and activity status (on the phone, idle, etc.).

SIP presence and availability build on the SIP event notification mechanism and on the registrar and other servers. Scripts can be set up at the server to route calls based on inspection of the INVITE message. A presence server uses SIP SUBSCRIBE/NOTIFY with a presence event package to gather a User Agent's presence status and send responses to a watcher interested in the presence status of a specific entity(presentity). Figure 11 shows a functional scenario on how a presence server determines presence status and distributes the status to a watcher upon appropriate authentication and authorization. For example, two users can see each other's presence status in their appropriate buddy lists by using the presence server and thus communicating effectively resulting in higher productivity.

An enterprise user in a typical day operates in multiple presence domains as depicted in Figure 11.However, presence events from multiple presence domains can be aggregated to represent one virtual presence across the enterprise network. The creation and maintenance of an enterprise-wide buddy list with appropriate access control list (ACL) and presence distribution policy can be used effectively to enhance end user productivity.18

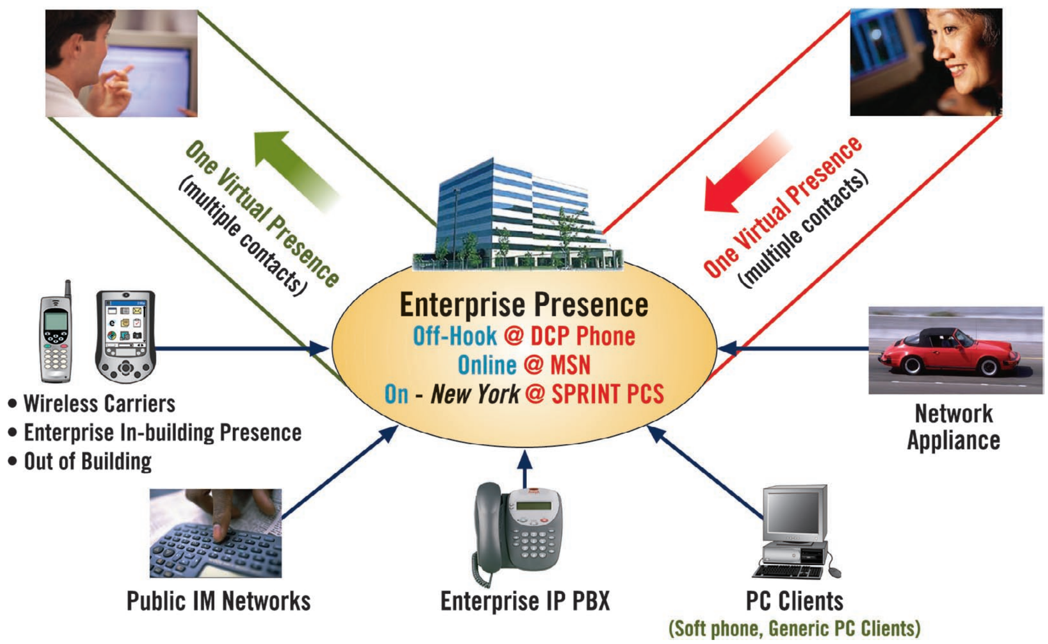

Figure 11 Enterprise presence and buddy list

SIP systems may support presence as a capability using SIP extensions. The SIP Instant Messaging and Presence Leveraging Extensions (SIMPLE) WG at IETF has advanced specifications toward standards status.

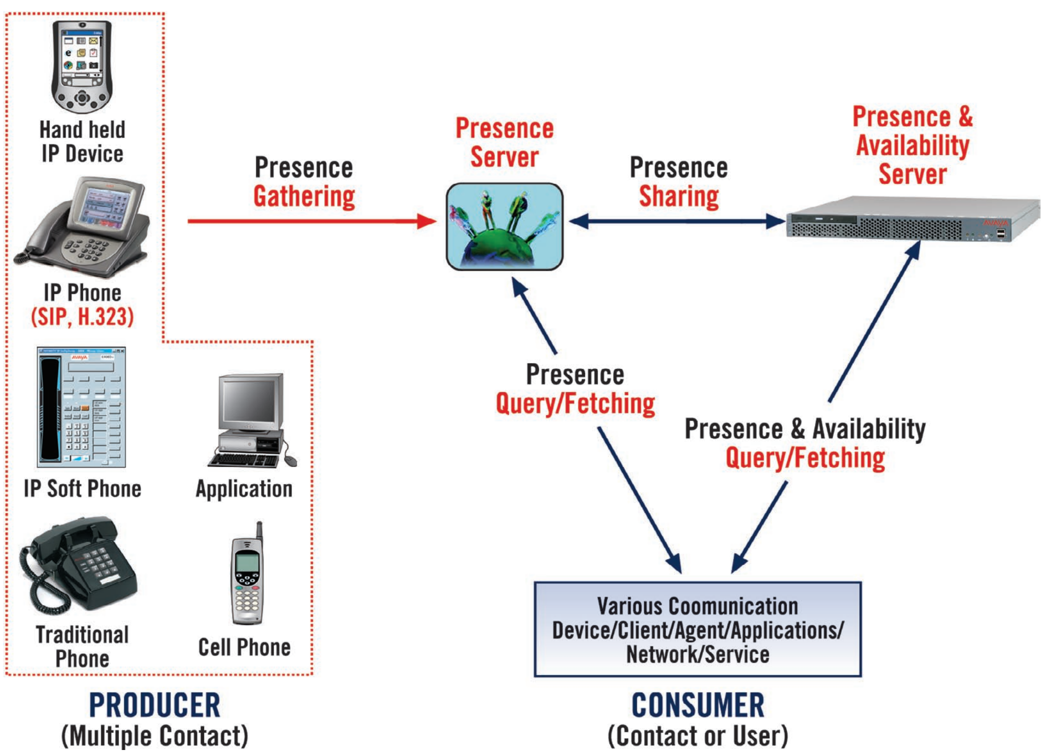

Figure 12 SIP presence server solution

The presence server scenario follows the same basic procedure as the proxy server and the redirect server do. Figure 13 shows how two users, sip: Michael@avaya.com and sip: John@bigcompany.com, would establish a point-to-point session by leveraging a presence server in the network and routing a call to a client where Michael is present. After a successful invitation based on the presence of the UAC, two users are able to have a conversation through RTP. Presence-driven calling can reduce voicemail tagging, achieving a higher call completion rate and increasing the company's bottom line.

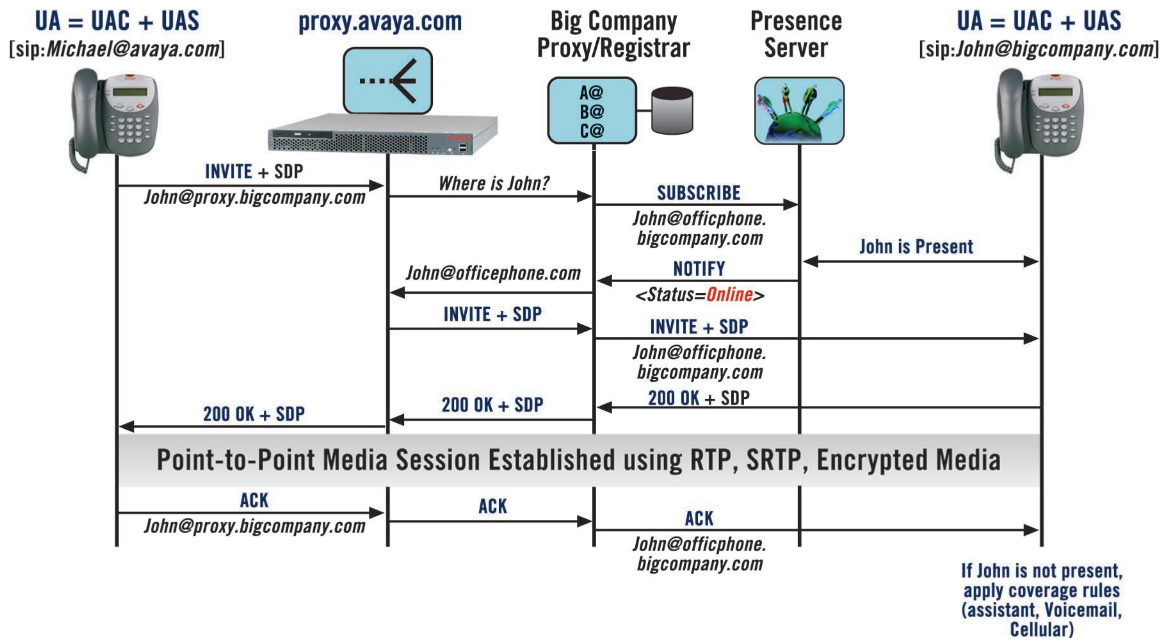

Figure 13 Example of using a SIP presence server

5.4 Point-to-Point SIP Instant Messaging

SIP is not only about telephony and voice calling features; it also supports other types of communication sessions such as Instant Messaging. Instant Messaging is becoming an active tool in the Internet for millions of consumers and business users. Consider an IP phone that can send a note through the IrDA port as an Instant Message and can receive a quick response. SIP provides a framework to add these types of productivity tools to a day-to-day communication environment. Figure 14 shows how two users, sip: Michael@avaya.com and sip: John@bigcompany.com, would establish a point-to-point Instant Message session through a proxy server. In this case the proxy server works to connect two UAs for IM.

1. The user Michael at host "***avaya.com***" initiates an IM to John.
2. A MESSAGE request is generated and sent to the proxy server, "***proxy.bigcompany.com***."
3. The server accepts the request and looks up Michael's contacts in the Registrar.
4. The Registrar returns the host, "***officePC.bigcompany.com***," where the user, John, is located.
5. The proxy server forwards the MESSAGE request to the host "***officePC.bigcompany.com***."
6. The UAS at "officePC.bigcompany.com" provides John a choice to accept the invitation in the form of ringing the phone, blinking a LED, or displaying text in the phone screen.
7. The acceptance is returned to the proxy server.
8. The proxy server sends the acceptance to the original caller Michael.

[2] Results are highly dependant on individual operating environments. Different implementation methodologies, assumptions, processes, and objectives may contribute to lower or higher results.

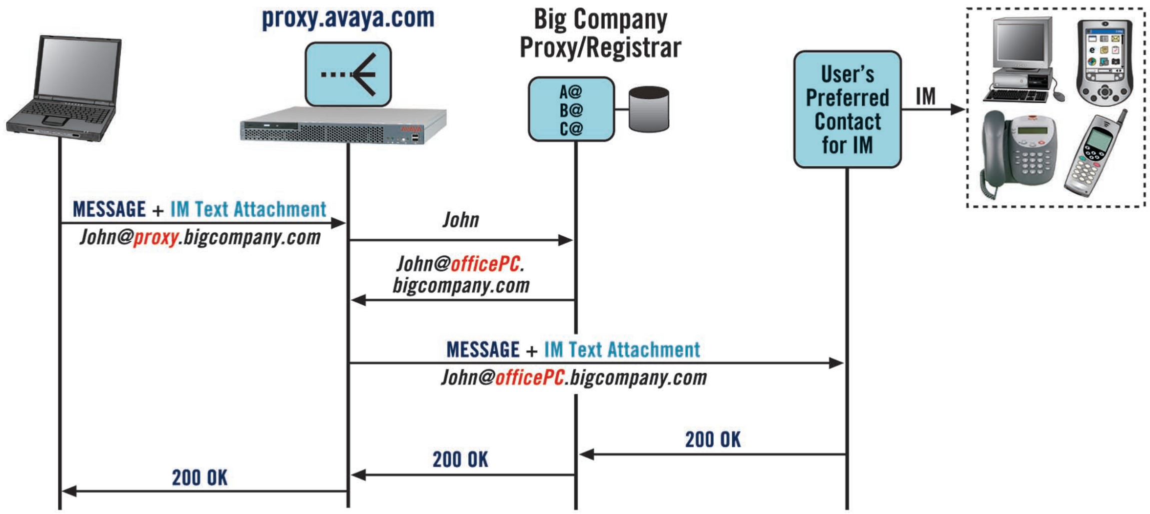

Figure 14 Example of point-to-point Instant Messaging

The example scenarios presented in this section show basic SIP operations in next generation VoIP environments. More detailed call flows and services examples can be found from the vast Internet based SIP documentation and resources provided by IETF, SIP Forum, and SIP Center.

Section 6: SIP and ENUM: Phone Number as a VoIP Address

One of the key issues of an evolutionary approach to converged communications is the way that new IP-based services deal with the most familiar telephony address: the phone number. E.164 numbers(roughly speaking, phone numbers including country and area code) are globally unique, language independent identifiers for a resource on Public Telecommunication Networks [E164]. These numbers are already used to identify multiple resource types, including ordinary phones, fax machines, pagers, data modems, email clients, text terminals for the hearing impaired, etc.

The ENUM working group within IETF [ENUM] has defined a DNS-based architecture and protocol[RFC2916] by which an E.164 number can be expressed as a Fully Qualified Domain Name in a specific Internet Infrastructure domain defined for this purpose (e164.arpa). The result of the ENUM query is a series of DNS NAPTR resource records [RFC2915] that can be used to contact one or more resources (for example, URIs) associated with that number.

In particular, an entity routing a SIP request directed to a global phone number can make use of ENUMDNS records to look up a SIP URI(s) registered for that number. This publicly routable SIP address can be used in the usual way to forward the request. Such a mechanism provides a standardized alternative to proprietary routing algorithms and databases. This presupposes the collection of these records into a centralized or hierarchical service. In many ways, this resembles the administration of number portability in the PSTN, in that there will typically be authorities for this information outside the service provider or enterprise. Avaya Converged Communication Architecture is capable of accommodating ENUM to be integrated as part of an Enterprise Communication Solution.

Section 7: A Comparison of SIP and H.323

H.323 is actually an umbrella specification that describes elements and interfaces (for example, a system),and specifies the use of various protocols:

- H.225.0 for session establishment
- H.245 for media signaling
- RTP for media transport
- H.235 for security signaling
- H.450.x or H.460.x for additional services

SIP (RFC 3261) is a specification that describes elements and interfaces, and this "base" specification also defines the protocol for session establishment. SDP is used for media signaling; RTP for media transport and several other specifications, such as published RFCs for completed work or Internet Drafts for work in progress.

SIP and H.323 have architecturally different models while they appear functionally similar at the elementary level. However there is a fundamental difference between SIP and H.323. H323 uses a telephony-based model, while SIP is an Internet-based model in its core design and re-uses many Internet components used by other popular Internet applications such as Email and the World Wide Web.H.323 defines system elements gatekeepers and endpoints, where an endpoint can include a gateway or multipoint control unit (MCU). At a minimum, the gatekeeper serves as a registration point to map an alias address like telephone numbers, URLs and text handles, to a transport address (for example, an IP address). Endpoints register with the gatekeeper. Depending on the implementation, the gatekeeper can expand to take on additional responsibilities, even functioning as a full-fledged feature server within anH.323 network such as an IP-PBX.

The gatekeeper and endpoints are functional elements that can be physically combined in interesting ways. For example, a gatekeeper, MCU, and gateway can be combined to make an IP-PBX. H.323 allows feature control logic to be centralized or distributed, including on a per-call basis. Conferences may be created using a centralized bridge (for example, audio mixer or video switcher) or in a distributed manner, allowing each endpoint to mix or switch. SIP also supports gateways and conference bridges. The defined elements are functional elements that can be physically combined in a variety of ways. Conferences may be created in a distributed or centralized manner.

The immediately obvious differences are in syntax and coding. The H.323 family of protocols defines syntax using ASN.1, and the protocol is encoded using Packed Encoding Rules (PER), a kind of binary coding. SIPs and SDPs syntax and encoding are defined using Augmented Backus Naur Form, meaning the bits on the wire are text. The different syntaxes reflect a bit of a philosophical difference between SIP and H.323. SIP defines a variety of "headers," elements that convey specific information. Messages are defined to include certain headers, so the approach is modular. H.323 explicitly defines the bits of information in each message. Either protocol can be extended or enhanced to carry private information or to add new standardized elements without changing the base protocol.

Many of the messages are semantically similar, as shown in the following table. However, some of the messages in one protocol do not have semantic counterparts in the other.

A Comparison of SIP and H.323 Messages		
H.323 Message	**SIP Message**	**Meaning**
Setup	INVITE	Request to establish a session (for example, place a call)
Alerting	180 Ringing	The called party is ringing
Connected	200 OK	The called party answered
Release Complete	BYE	The party ended the session
Call Proceeding	100 Trying	Request to establish acknowledged, working on next step
Progress	183 Session Progress	Convey information about the progress of the call
RRQ	REGISTER	Register an alias and IP address

Table 1: A Comparison of SIP and H.323 Messages

Though many procedures are similar, there are some significant differences between SIP and H.323. H.323has unique messages for admission request (permission to set up a call) and call initiation, although the content of the messages is nearly identical. The result is signaling diagram as follows:

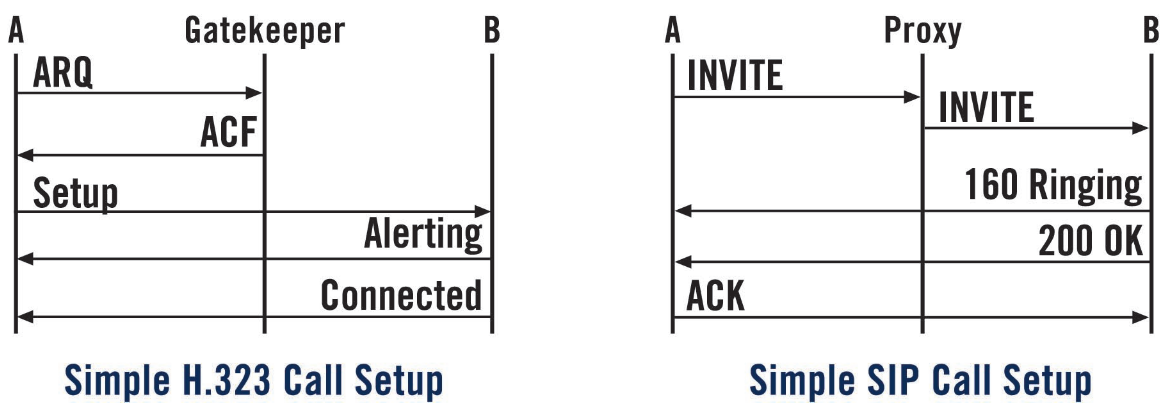

Figure 15 A comparison of SIP and H.323 message flows

SIP has been the focus of strong standardization activity and is addressing a number of applications, including presence or instant messaging. Avaya Converged Communication Architecture accommodates for SIP and H.323 interoperability and provides servers and gateways to accommodate for multi-protocol interoperability, which provides enterprise communication architects the choice of protocol and investment protection without sacrificing communication features and functionality. Such interoperability also helps ensure migration of the embedded base.

Section 8: A Comparison of SIP and H.248/MEGACO

Although H.248/Megaco was created as a joint project between IETF [4] and ITU [5], it has a different name in each organization. It is known as H.248 in the ITU and Megaco protocol or RFC 3015 in the IETF. H.248/Megacois called a "gateway control" protocol. H.248/Megaco is a master/slave protocol that allows a relatively intelligent Media Gateway Controller (MGC) to control a relatively unintelligent Media Gateway (MG). Because of the types of devices targeted for control by H.248/Megaco and the low level of its control structure,H.248/Megaco is generally viewed as complementary to SIP. While an MGC will use H.248/Megaco to communicate with a number of MGs, a protocol such as SIP would be used for one MGC to communicate with another MGC. From a SIP perspective, the combination of MGC and MGs are treated together as a SIP Gateway.

H.248/Megaco allows a MGC to instruct a MG to make connections within the MG, to play a signal on a termination (for example, to apply ringing tone or flash a trunk), and to report events associated with a termination (for example, that a DTMF digit was detected, or that a line went off-hook). This low level of control does not exist in SIP, which operates at a more functional, peer-to-peer level.

The text-encoded version of H.248/Megaco is described in a manner similar to SIP, and SDP can be passed directly between SIP and H.248/Megaco without the need for intermediate processing. The signaling diagram in Figure 16 is an example of the manner in which SIP and H.248/Megaco might interact. Some example steps involving communication between SIP and H.248/Megaco entities are as follows. For this example, assume that the MG provides connectivity to a number of POTS telephones.

- When a user takes a POTS phone off-hook, the MG notifies the MGC, which then instructs the MG to play dial tone, collect digits and compare to a digit map, and watch for the POTS phone going back on-hook.
- Once the MG has collected all the required digits (for example, some other phone's extension), the MG notifies the MGC.
- The MGC instructs the MG to add 2 terminations—the POTS phone and the dialed party—to a "context," and asks the MG to select from a provided set of audio coders, expressed in SDP.
- The MG replies with the selected coder and RTP ports, expressed in SDP.
- At this point, the MGC issues the SIP INVITE method.
- When the MGC receives media information from the called SIP phone, expressed in SDP, the MGC updates the MG so that the MG knows which coder to use and where to send the audio stream.
- When the SIP phone answers, the MGC instructs the MG to receive and send audio.

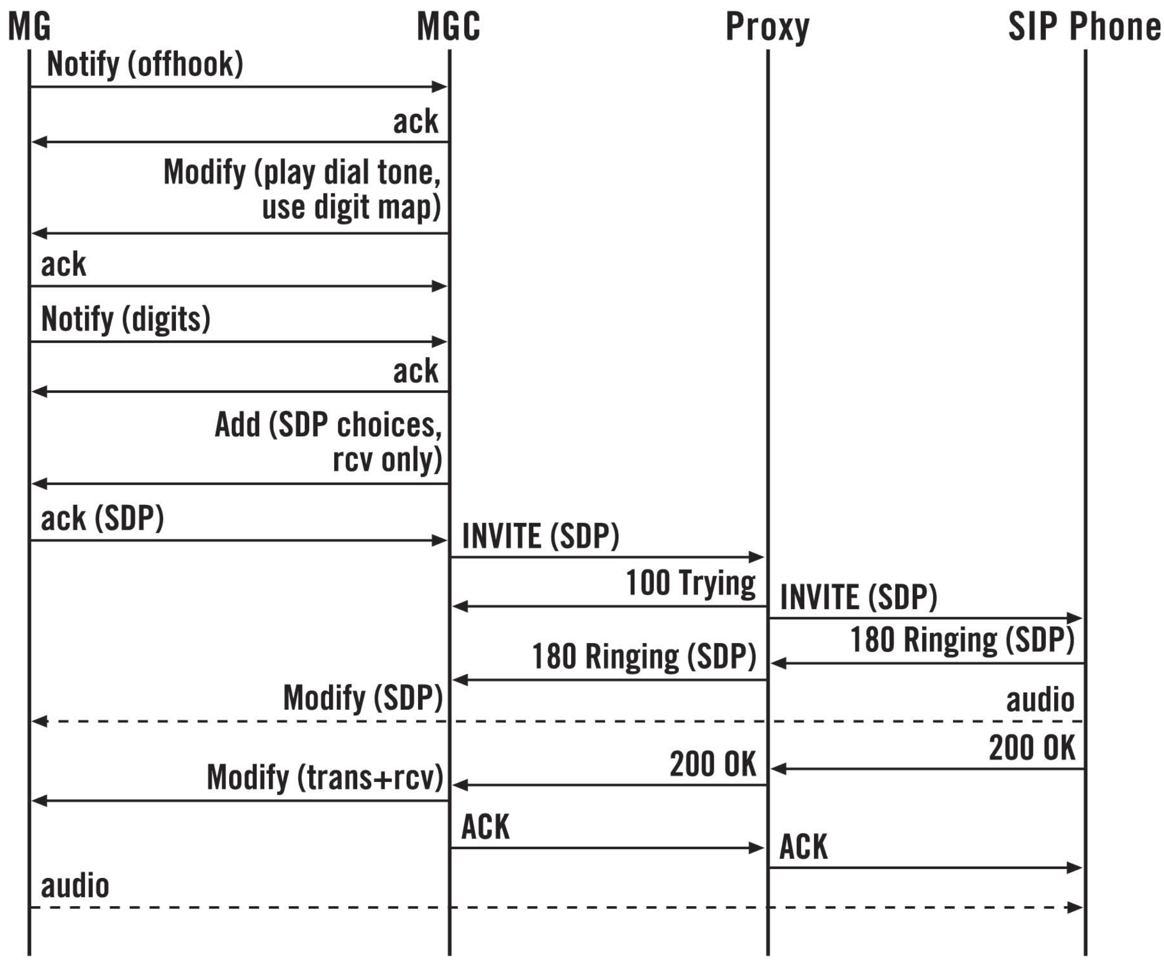

Figure 16 Basic interoperability between SIP and H.248

One possible application of H.248/Megaco is to allow an MGC to control a set of relatively unintelligent IP telephones. In this scenario, the MGC controls all aspects of the IP telephone, such as instructing the IP telephone to light specific lamps, produce a specific display, or report button pushes. The result is that the IP telephone closely mimics the operation of "circuit" telephones such as are commonly found on traditional PBXs. Avaya Converged Communication Architecture accommodates for SIP and H.248 interoperability.

Section 9: Practically Deploying Enterprise SIP Networks

A variety of deployment challenges arise in reaching the goal of a SIP-based communications infrastructure. Typically, the migration to SIP starts with a large installed base of existing enterprise users and enterprise operators that have been accustomed to certain telephony quality and features. There is always some inherent simplicity that comes with the advance in underlying fundamental technologies. Faster processors, smaller footprints, better algorithms, new software technology, etc. benefit all solutions including Voice over IP. Choosing the protocol in planning is only a small piece of the puzzle but numerous other issues need to be accounted for to deploy SIP practically within an Enterprise.

9.1 Leveraging Existing Network Assets

When technologies such as SIP emerge as end-to-end protocols, it allows many vendors to build communication products that provide a minimal set of capabilities. However, the fundamentals of enterprise business practice are very complex. They need to be mapped into a next generation converged communication architecture that leverages the functionality of existing enterprise assets. Migration, Investment Protection, Return on Investment (ROI) and Total Cost of Ownership (TCO)—with increased productivity for end users—have to be accounted for within the converged architecture and phased deployment plan to address enterprise business processes complexity.

9.2 Integrating Existing Business Functions as SIP Feature Servers

Feature servers are entities that enable business communication logic. In particular, they must resolve issues that are not addressed at the SIP protocol level but are necessary aspects of a total communications solution. Examples of these features are time-of-day routing, enterprise dialing plans, boss-assistant interactions, hotlines, emergency calls, and class of service restrictions. Many of these features are already deployed in today's enterprise communications systems and reflect critical practices developed over time, and that may differ from enterprise to enterprise. Effective deployment of an enterprise SIP network requires that these attributes be migrated and enhanced where necessary. Feature servers are critical to leveraging existing solution components in a SIP framework.

9.3 Security

End-to-end technology like SIP relies heavily on an underlying distributed Internet network for end-to-end security that includes authentication, authorization, and privacy. Security should be one of the key deployment considerations and needs to be factored into the deployment plan. Some of the security elements that need to be considered are:

- Signaling Channel Encryption using standards such as Transport Layer Security (TLS) [20]
- Associated Certificate Authorities (CA) for authentication and strong encryption algorithms
- Advanced Encryption Standard (AES) driven media encryption for confidentiality in all IP devices

9.4 Quality of Service

End-to-end technology like SIP also relies heavily on an underlying distributed Internet network for end-to-end Quality of Service (QoS). Standard QoS technologies to be factored in are IEEE 802.1p/Q, Differentiated Service (DiffSrv), and IP Precedence. More advanced application monitoring solutions such as Real-Time Application QoS Monitoring (RAQMON) capabilities in all IP devices also need to be considered.

9.5 Service Provider Interaction with Enterprise Proxies

A truly ***Free Enterprise*** is based on the notion that it's not the service providers who control the features or application, but rather the enterprise users and operators who decide which features and services they need. They need complete flexibility to match applications with business demands. Though an enterprise can deploy intra-enterprise SIP solutions without requiring SIP services in the wide area, some enterprises might optionally choose to run such operations. Special caution should be taken to achieve bordering functionalities between enterprises and the service provider network edge by addressing NAT and firewall issues as discussed in the next section. Unlike the PSTN network, a core SIP proxy server at the service provider network does not perform the role of a Class 5 switch. Instead, proxies assume a protocol architecture in which signaling intelligence is distributed at the edge. Features and applications in this architecture remain aggressively independent from signaling and message transport, and are hosted within the enterprise network in the form of Feature Servers, IP PBXs and Application Servers. Technologies such as SIP—and IP-based communication technologies in general—are increasingly using service providers for network access, while the application intelligence is actually at the edge. SIP pushes this boundary further, empowering the network edge to create and add applications with extreme ease.

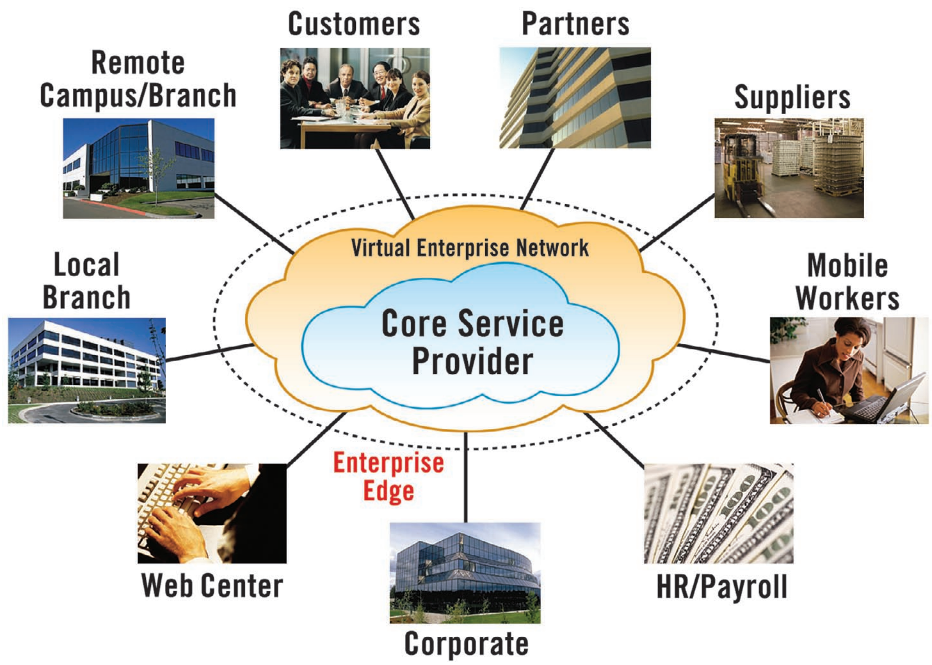

Figure 17 Distributed logical enterprise network

This notion of Free Enterprise is further advantaged by ever growing application domain technologies like XML, VoiceXML, Web Services, CPL, and Servlets, making the success of an enterprise-driven converged communication a reality. Monolithic applications have been functionally decomposed into a set of flexible commodities, services, applications and appliances. SIP integrates these entities in a flexible and open environment while keeping the end user at the heart of the communication architecture. Finally the true Free Enterprise is here.

9.6 NAT/Firewall Traversal Issues

A significant set of issues in the path of deployment of SIP has to do with managing the boundaries between the enterprise and the public network. In particular, the following issues arise:

- Enterprises have very rigorous security requirements
- Firewalls/NATs
- Regulation requirements (logging/recording, etc.)

One of the deployment issues most often discussed is dealing with firewalls and NATs. The NAT is a mapping between Internal network addresses and external public network addresses. It allows reuse of globally unique IP addresses that are in limited supply until IPv6 becomes available. Enterprises can use "any" addresses within their private networks provided packets to the public Internet use globally unique addresses. A side benefit of this is the privacy of internal addresses. However, NAT issues for SIP involve translations on transport addresses embedded within "via headers." Firewalls restrict connections to outbound only; hence the same problem that plagues NATs also plagues firewalls. The associated firewall function must be able to parse SIP messages, extract the IP addresses and port numbers from the SDP, and open up "pinholes" for these sessions. The firewall must close these holes when a bye or session timer expires. UDP traffic will not traverse firewalls. Therefore, RTP over TCP or TLS [20] should be used instead. In the worst case, the RTP would need to be carried over a TLS connection on port 443. The solution that Avaya has is to provide a SIP enabled NAPT/firewall ALG that makes modifications to SIP headers and SDP fields.

Section 10: SIP in Avaya Converged Communication Topology

As outlined in the three-phase evolution to converged communication, most enterprises have a mixture of traditional and converged networks topologies. These topologies are primarily based on a mix of PBX based switching systems catering to both IP and PSTN based endpoints. Communications applications are resident on various application servers. The converged communications topology is characterized by a distributed architecture with various software components. As outlined in *The Evolution to Converged Communications, Avaya Communication Architecture,* and *Evolving to Converged Communications with SIP,* Avaya provides a migration path for customers wanting to move from existing topologies to a converged communication topology, using SIP as a bridge. The goal is to provide services such as Instant Messaging, Presence and Wi-Fi integration of dual-mode cellular phones for existing endpoints and new intelligent endpoints, while protecting investments and migrating at the customer's pace. Investment protection is even greater for an existing Avaya-installed customer base, since the telephony applications from Avaya can be easily migrated to the SIP/IP world while providing additional SIP Enablement Services as outlined in *Evolving to Converged Communications with SIP*.

Section 11: Avaya SIP Leadership, Standards Compliance, and Interoperability

Multi-vendor interoperability and compliance to IETF standards are keys to the success of SIP as a technology. Avaya has been an active contributor to the Internet Engineering Task Force's SIP, SIPPING and SIMPLE Work Groups. Avaya is also a key member of the SIP Forum and various other SIP initiatives within the industry and a regular participant in SIP interoperability events, such as SIPiT. Avaya is firmly committed to SIP, leading and sponsoring many SIP events to promote incorporation of SIP as a multi-vendor open protocol, within various industry sectors, as a key enabler of converged communication. Avaya will continue its leadership to promote open standards and industry-wide breakthroughs to move ahead in the space of SIP-enabled converged communication.

Section 12: Conclusion

SIP is a technology for real time, converged, peer-to-peer communications. It supports dynamic new communications methods such as IM, presence, and multimedia applications in a flexible, distributed environment. SIP enables seamless convergence of communication and ease of end-to-end service integration based on open technologies. SIP technology builds on IP telephony momentum and provides the ability to quickly and seamlessly integrate with numerous rich Internet-based applications and real-time collaboration applications. It offers a powerful means for enterprises to enhance the efficiency of eBusiness-enabled IT environments, and serves a common point of integration across vendors, service providers and application providers. SIP provides value to end-users by:

- Providing choice of "intelligence" at the endpoint, regardless of device or network
- Leveraging the Internet development community to create richer, converged communication applications via open, standardized interfaces
- Accelerating convergence of the Internet and PSTN networks into the next-generation, presence enabled communication network
- Accounts for web-based service creation and application development, which enables new features and capabilities to be quickly added in response to competitors or changing market dynamics

SIP enables dynamic communications applications in an open, industry standard distributed environment and reduces total cost of ownership by driving the exploitation of private IP networks. SIP increases communication efficiency by integrating real-time voice, presence, instant messaging, voice messaging, location, and mobility applications. It provides an enterprise bridge to new, hosted applications as well as to SIP carriers. It makes available new applications that complement existing features and reduces operational cost significantly by enabling convergence of communication.

Section 13: Reference

1. **The Evolution to Converged Communications**
   http://www.avaya.com/gcm/master-usa/en-us/resource/assets/whitepapers/lb1845.pdf

2. **Avaya Communication Architecture**
   http://www.avaya.com/gcm/master-usa/en-us/resource/assets/whitepapers/lb1846.pdf

3. **Evolving to Converged Communication with Session Initiation Protocol (SIP)**
   http://www.avaya.com/gcm/master-usa/en-us/resource/assets/whitepapers/lb2337.pdf

4. **Internet Engineering Task Force**
   http://www.ietf.org

5. **International Telecommunication Union (ITU)**
   http://www.itu.int/home/index.html

6. **IETF ENUM WG**
   http://www.ietf.org/html.charters/enum-charter.html

7. **Falstrom, P., "E.164 number and DNS," RFC 2916, September 2000.**

8. **ITU-T/CCITT Recommendation E.164, "Numbering Plan for the ISDN Era," August 1991.**

9. **Mealling, M. and R. Daniel, "The Naming Authority Pointer (NAPTR) DNS Resource Record," RFC 2915, September 2000.**

10. **Berners-Lee, T., Fielding, R. and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax," RFC 2396, August 1998.**

11. **Simple Object Access Protocol (SOAP)**
    http://www.w3.org/TR/SOAP

12. **Web Services Definition Language (WSDL)**
    http://www.w3.org/TR/wsdl

13. **Voice Extensible Markup Language (VoiceXML)**
    http://www.w3.org/TR/voicexml20/

14. **Extensible Markup Language (XML)**
    http://www.w3.org/XML/

15. **Universal Description, Discovery and Integration (UDDI)**
    http://www.uddi.org/about.html

16. **A Transport Protocol for Real Time Applications (RTP)**
http://www.ietf.org/rfc/rfc3550.txt

17. **Session Description Protocol (SDP)**
http://www.ietf.org/rfc/rfc2327.txt

18. **Session Initiation Protocol (SIP)**
http://www.ietf.org/html.charters/sip-charter.html

19. **Call Processing Language (CPL) Framework and Requirements**
http://www.ietf.org/rfc/rfc2824.txt

20. **Transport Layer Security (TLS)**
http://www.ietf.org/html.charters/tls-charter.html

Learn More

For more information on how Avaya can take your enterprise from where it is to where it needs to be, contact your Avaya Client Executive or Authorized Avaya BusinessPartner, or visit us at www.avaya.com

About Avaya

Avaya enables businesses to achieve superior results by designing, building and managing their communications infrastructure and solutions. For over one million businesses worldwide, including more than 90 percent of the FORTUNE 500®, Avaya's embedded solutions help businesses enhance value, improve productivity and create competitive advantage by allowing people to be more productive and create more intelligent processes that satisfy customers.

For businesses large and small, Avaya is a world leader in secure, reliable IP telephony systems, communications applications and full life-cycle services. Driving the convergence of embedded voice and data communications with business applications, Avaya is distinguished by its combination of comprehensive, world-class products and services. Avaya helps customers across the globe leverage existing and new networks to achieve superior business results.

AVAYA

COMMUNICATIONS
AT THE HEART OF BUSINESS

avaya.com

Printed in the U.S.A.
10/05 • EF-LB2343-01