♦ Web Communication Services and the *Packet*IN™ Application Hosting Environment

Yang Chen, Olivier B. Clarisse, Pascal Collet, Mark A. Hartman, Luciano Rodriguez, Lizette Velazquez, and Bruce A. Westergren

Large telecommunication customers are migrating their network infrastructure to support new converged services, while containing their operating costs. Deploying converged services on the networks today represents great opportunities to network service providers for new revenue generation. It brings big challenges as well, due to the requirements for a service platform with high capability to deal with the complexity of the network infrastructure, the difficulty of interoperability between different service platforms, and the diversity of signaling protocols and application programming interfaces (APIs). The Lucent Packet*IN™ application hosting environment (AHE) provides a solution that empowers network service providers to deliver a wide variety of enhanced services over the converged (packet and circuit, wireline and wireless) networks. It enables the creation and deployment of enhanced services on converged networks via the open service platform with interoperability, programmability, scalability, and wide protocol compliance. In particular, a new class of services is presented to demonstrate the transformation of telecommunication services that is enabled through Web presence. This article gives an overview of the* Packet*IN AHE with the focus on the customer values, the architecture, and enabling capability to deploy advanced applications and services. A new service portal, enterprise communication, is presented as an example of the innovation and implementation enabled by the service enabling environment. The enterprise communication provides Web access to presence information, instant messaging, third-party call management, and location. This convergence of features is enabled by the* Packet*IN AHE integration of the public switched telephone network (PSTN), session initiation protocol (SIP), and H.323 protocols through standard open APIs. The voice communication protocols are combined with a Web access interface to establish a new Internet presence, while leveraging existing switching products and reusing deployed communication networks and services.*

Bell Labs Technical Journal 7(1), 0–0 (2002) © 2002 Lucent Technologies, Inc. Published by Wiley Periodicals, Inc.
Published online in Wiley InterScience (www.interscience.wiley.com). • DOI: 10.1002/bltj.15

Introduction

The wide deployment of intelligent networks (IN) in the past two decades empowers the service/network providers offering numerous advanced, value-added, and revenue-generating communication services to the market. Some examples of the vastly desired IN services include toll-free telephone services (800 services), virtual private network (VPN), and calling card (pre-paid and post-pay) services. The growing demand and wide deployment of IN services contribute greatly to the extremely high growth level of the communication industry in the past twenty years. The trend of demanding advanced and revenue-generation services continues as the communication networks evolve from separate circuit switched and packet switched networks to a converged packet/circuit switched network. Many service providers worldwide are planning to offer services on the next-generation converged communication networks. Some front runner services include converged (data, fixed line and wireless) VPN, network contact center (which integrates the customer premise equipment-based call center), Web/Internet-based call center and customer relationship management, and various toll-free telephone type of services. The terminology of service/application used in this paper is referred to the communication capabilities received and used by end users and provided by service providers through commercial/business arrangements. The services and applications are used interchangeably.

Deploying next-generation services on the converged network represents great opportunities to network service providers. It brings big challenges as well to the technology and equipment vendors. It requires new service platforms with high capability to deal with the complexity of the network infrastructure, the difficulty of interoperability between different service platforms, and the diversity of signaling protocols and application programming interfaces (APIs). It also requires regulatory considerations by both service providers and regulatory bodies. Lucent's *Packet*IN™ application hosting environment (AHE) attempts to provide an answer with a next-generation converged services platform that enables carriers to deliver a wide variety of enhanced services for converged voice and data networks. It provides the infrastructure needed for the creation and deployment of new Internet protocol (IP) and multimedia services. The *Packet*IN AHE is an open-interface, programmable, and network agnostic application server for building, deploying, and managing next-generation communication services. It can be deployed in multi-vendor, multi-network (wireline, wireless, and data), and multi-protocol environments. The *Packet*IN AHE supports service development and deployment via various programming languages and APIs for rapid service deployment by independent software vendors, the service providers, or Lucent.

This article provides an overview of the *Packet*IN AHE with the focus on the system architecture and the enabling capability to deploy advanced services/applications, as well as the value to customers. A new service portal, enterprise communication, is presented as an example of the innovation and implementation enabled by the service enabling environment. The enterprise communication provides Web access to presence information, instant messaging, third-party call management, and location. This convergence of features is enabled by the *Packet*IN AHE with the integration of the public switched telephone network (PSTN), session initiation protocol (SIP) [14], and H.323 [4] protocols through standard open APIs. The voice communication protocols are combined with a Web access interface to establish a new Internet presence, while leveraging existing switching products and reusing deployed communication network and services.

*Packet*IN Overview

Figure 1 shows a simplified version of Lucent's *Packet*IN service platform architecture, which consists of the *Packet*IN AHE, the enhanced Service Authoring Environment (eSAE), the enhanced Services Manager (eSM), and the enhanced Services Directory (eSD). The *Packet*IN AHE is an open-interface, programmable, and network-agnostic application server that interfaces with Lucent Softswitch (LSS) or a third party's softswitch via Java* Telephony API (JTAPI) [8] based API or SIP for deploying next-generation

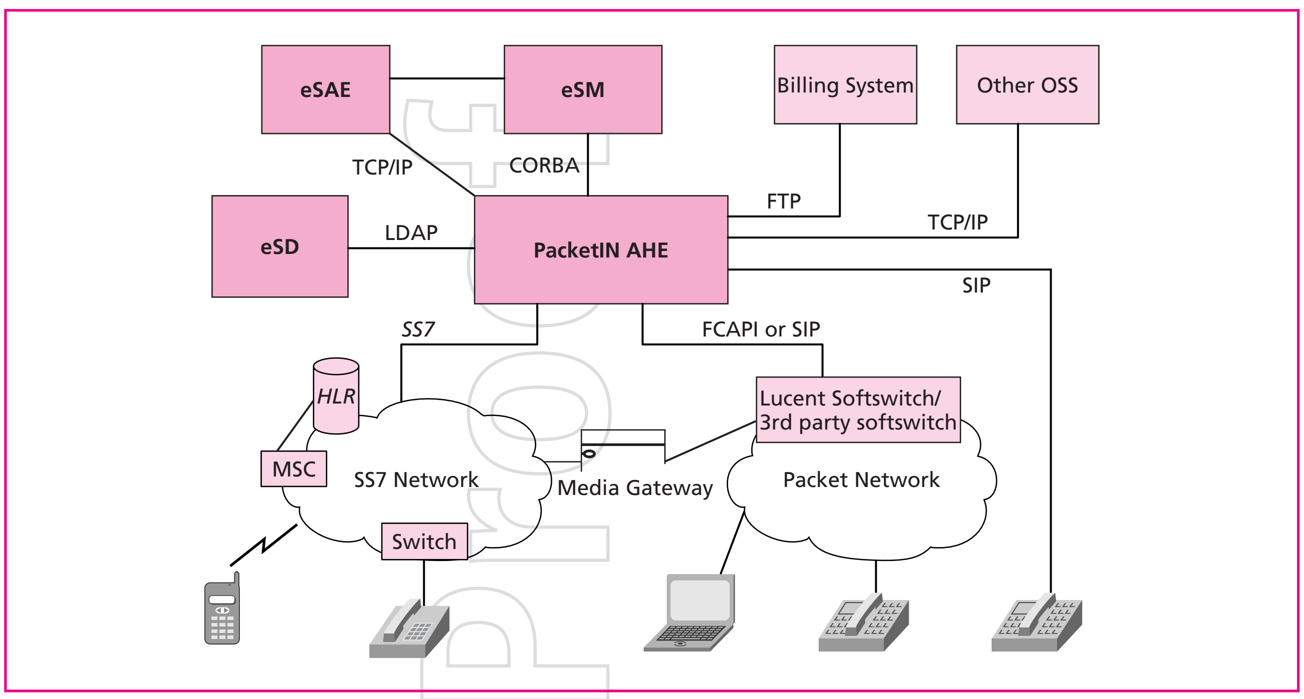

Figure 1.
Packet*IN*™ *AHE in the network.*

communication services. It communicates to the traditional voice network as well via Signaling System 7 (SS7) protocol to allow service providers to offer IN-based services and new services to their existing customers. The eSAE enables the creation of application and services using various programming languages and APIs. The services/applications can be provisioned on the AHE by the eSM via CORBA* over transmission control protocol/Internet protocol (TCP/IP) interface. Optionally, the services can be downloaded directly from the eSAE onto the AHE via TCP/IP. The eSD provides centralized storage of service and subscriber data for the services hosted on the AHE. Communication between the eSD and the AHE is via lightweight data access protocol (LDAP) [9] over TCP/IP. The AHE also communicates with various network elements for billing, network management, and fault management via TCP/IP based protocols. The various *Packet*IN functions and components enable the *Packet*IN platform to support service development and deployment in multi-vendor, multi-network (wireline, wireless, and data), and multi-protocol environments.

The following sections will be devoted to the *Packet*IN AHE with more detailed discussion on the AHE architecture, the functions of AHE components, and the key benefits.

*Packet*IN AHE Architecture

The *Packet*IN application hosting environment leverages the distributed architecture of next-generation IP infrastructure and provides a high availability platform for hosting next-generation services. **Figure 2** shows a high-level architecture of the *Packet*IN AHE with the major components of the AHE software.

The *Packet*IN AHE executes on a commercially available hardware platform (for example, Sun* platform), running the Solaris* operating system. The *Packet*IN AHE middleware consists of the platform core, the service logic execution environment (SLEE), Web server, SIP server, and various interfaces to operation support systems. The platform core provides capabilities used by the entire system such as initialization, process control, interprocess communication, database access, fault detection, measurement and

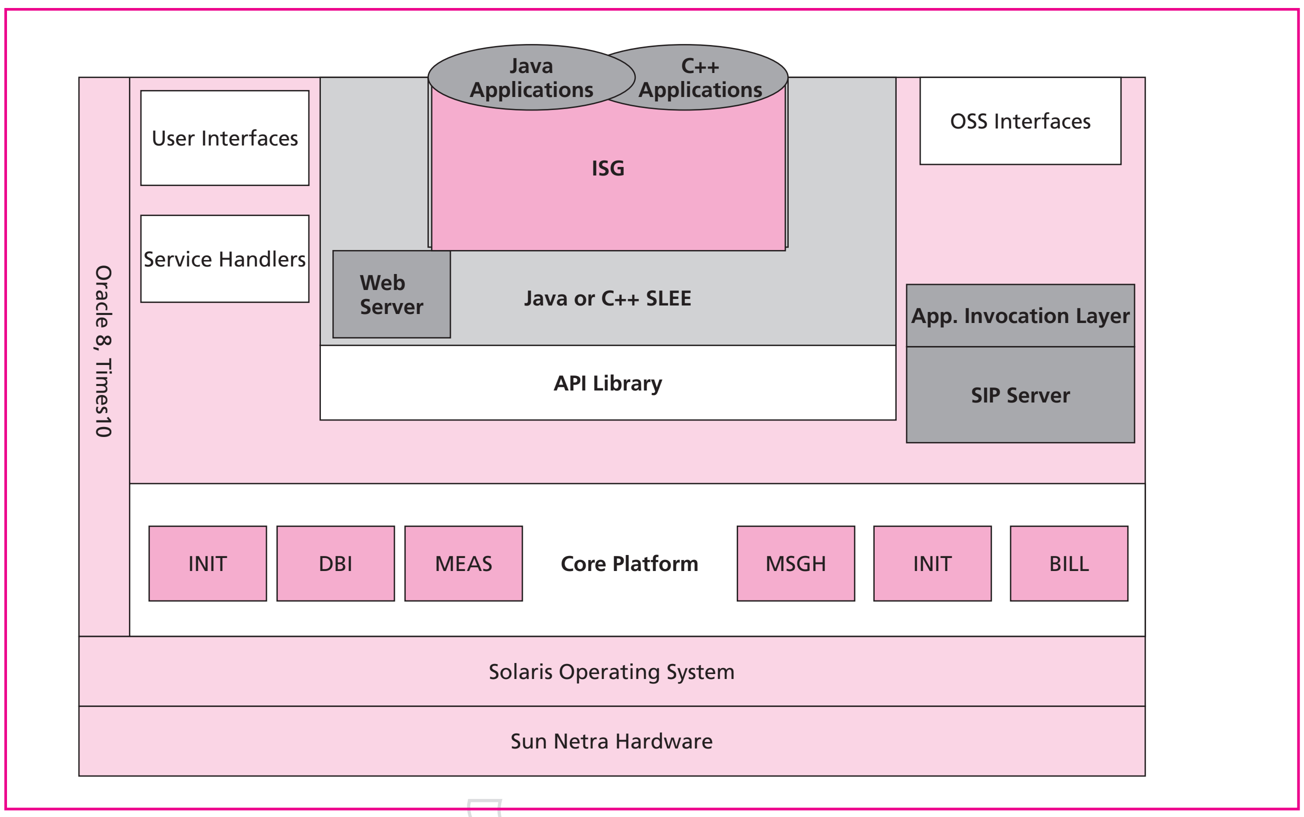

Figure 2.
Packet***IN™ application hosting environment software architecture.***

billing logging, and software updates. The SLEE is a crucial portion of the AHE that provides an execution environment and APIs for the applications hosted on the AHE. The SLEE enables the applications making use of the platform core capabilities, such as message handling/routing, application management, measurement, and billing. An API library provides interfaces between the SLEE and the platform core capabilities. The intelligent services gateway (ISG), a next-generation service gateway function on the AHE, provides a secure, standard interface between various applications and network resources/data required by the applications. The ISG enables service providers and third-party software developers to develop applications accessing network capabilities via the open service access (OSA) [1, 2, 11] and Parlay [12] open standardized APIs.

In the subsequent subsections, the AHE platform core and the SLEE are discussed in more detail. The ISG will be addressed in the next section.

AHE Platform Core

The AHE platform core consists of the various subsystems/modules that provide the fundamental capabilities required by the system itself, the SLEE, and the hosted applications. The main functions include system initialization, interprocess communication, database access, fault detection, measurement, and billing logging.

The system initialization function is provided by an initiation (INIT) module for initializing and synchronizing UNIX processes of the AHE. It starts and restarts processes, monitors the sanity of these processes, counts and thresholds process errors, and updates processes when requested by platform field update. INIT classifies processes with respect to their startup method and their effect on call processing. The interprocess communication is handled by the message handler (MSGH) subsystem, which supports queues for interprocess communication message routing. MSGH also supports a priority field so that the queue can be treated either as a priority queue or as a first in/first out queue at any given time.

The AHE is equipped with a relational database management system and an in-memory database for storage of system and services data. The AHE platform core provides database interface to allow all platform processes to access the database with structured query language (SQL) commands to perform insert, update, delete, and select functions. The database interface allows the processes to continue call processing while updating data in the database. The SLEE accesses the database via the database independent APIs (Oracle* database connectivity/Java database connectivity [ODBC/JDBC*]) API that enables various applications to access data stored in relational or non-relational databases by using SQL as the standard database access language. The AHE supports the concept of public tables that several applications (within the SLEE[s]) can access data from the same table or set of tables.

To meet service providers' billing needs, the AHE's billing subsystem (BILL) provides three major billing functions: the collection and storage of billing records generated by the applications, the creation of tracer records, and the transfer of billing information from the AHE to external billing systems in standard format for processing.

There are several measurements on the AHE for system performance, CPU and disk utilization, and TCP/IP traffic data, as well as services measurements. Each measurement is pegged and stored by the appropriate subsystem. The measurement subsystem (MEAS) on the AHE requests and collects the measurement reports on-demand at time intervals configured by the MEAS, instructs the appropriate subsystem to print measurement reports, and allows or inhibits measurement collection and reporting.

The service package manager (SPMAN) module is designed to manage the services/applications hosted on the AHE platform. The SPMAN is responsible to start, stop, or restart the applications, and monitor the status of the applications that could be non-exist, out-of-service or in-service. The SPMAN also provides measurement collection on call attempts.

AHE Service Logic Execution Environment

The SLEE provides the runtime environment for applications to be executed on the AHE platform. The applications (written in Java or C++) are built using a service creation environment (either Lucent's eSAE or other third-party tools) and are loaded on the AHE for execution. While some applications require only the Java virtual machine and standard Java libraries, many applications will take advantage of the hosting platform capabilities to enable value-added telephony functions. Time-to-market can be significantly reduced if application development can focus on the key, unique features of the services, while the AHE provides the commonly used platform functionality. The fundamental function of the SLEE is to provide applications on the AHE, an execution environment with access to a variety of platform capabilities required by the applications at runtime.

The platform capabilities that the applications can access through the SLEE include process management (INIT), message handling (MSGH), billing (BILL), measurements (MEAS), and application management (SPMAN). In addition, the SLEE may provide additional capabilities to the application, such as call processing support, security, and database access management. The SLEE consists of multiple functionally group packages. For example, the INIT package provides an execution interface to the INIT process on the platform, which initializes the application and brings it to the "in service" state. It also responds to heartbeat requests from INIT process. The operations, administration and maintenance package registers the SLEE with MSGH for sending/receiving messages, and provides a way of communicating input/output messages to the platform. The application manager determines the service keys to the deployed applications and registers them with SPMAN. It also performs service control by providing capability to suspend and resume service. Similar to the JAIN* [6] SLEE, the *Packet*IN SLEE provides a set of API interfaces to support application execution. The *Packet*IN SLEE is not currently compliant with the JAIN specification, but utilizes similar concepts, and may evolve to compliance in future releases as the JAIN standards evolve.

In summary, the SLEE provides the application with a rich set of APIs, exposing platform capabilities needed by applications to make AHE a robust platform for a wide range of telecommunications applications.

Intelligent Services Gateway

In a very competitive global telecommunication market, the network operators are facing the challenges and requirements to open their networks for third-party application providers to create new value-added services and make them available in the mass market. In the meantime, network reliability and integrity must be sustained for network operators and service providers. The AHE is designed to meet these challenges by building the ISG on top of it to provide a secure, standard interface between the applications "inside' or "outside" of the network, and the network resources and data within a telecommunications network. These applications may be provided by the network operator, or service provider, or by third-party application service providers. The ISG supports and is based on the Parlay, OSA, and presence and availability management (PAM) [13] standards. The ISG acts as the "front door" to network operator resources, allowing application developers to implement telephony features in their applications. Acting as a gateway, the ISG exposes a set of abstracted APIs that provide the glue between applications and features provided by the underlying network resources. These exposed APIs are implemented using CORBA, while the ISG supports a variety of standard and proprietary protocols for interfacing to elements in the services and control layers of the network. **Figure 3** illustrates a high-level ISG/AHE in the network.

ISG enables application interaction with network-based services such as call control, location, and user interaction, allowing those applications to become

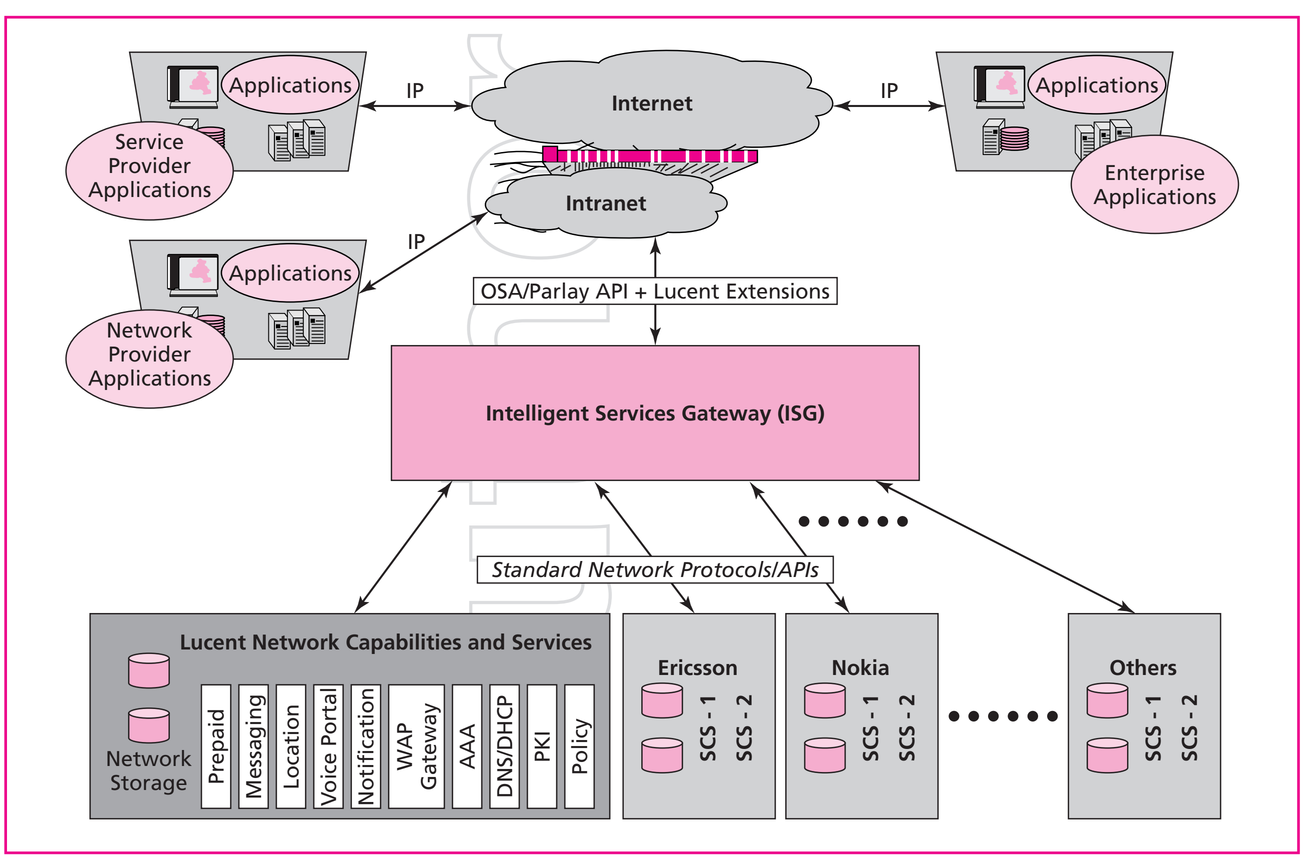

Figure 3.
ISG in the network.

"users" of the network. It also provides the opportunity for network-based services to interact with applications, such as information services, to enhance the end-user services. ISG provides secure access to the network services and supports policy-based mechanisms for users to control access to user information, such as location, and to personalize their services. APIs allow application developers to create applications, independent of network and transport technologies, which may reside outside the network, and are portable across various wireless and fixed networks.

ISG provides a detailed API, enabling the creation of new differentiated service combinations through the development of services that control the interaction of existing and new service delivery platforms within the network operators' domain. This ensures that the utilization of existing, evolved, and new service infrastructure is maximized. The APIs hide the complexities of the underlying network so that developers have an abstracted view of the network capabilities and protocols. The ISG translates the APIs to network service interfaces such as American National Standards Institute-41 (ANSI-41), Intelligent Network Application Part (INAP), Transaction Capabilities Application Part, (TCAP), Mobile Application Part (MAP), Capabilities Application Part (CAP), SIP, Internet Messaging Application Protocol version 4 (IMAP4), LSS API, and short message peer-to-peer protocol (SMPP). These protocols may be standard and/or proprietary, depending on the underlying network element(s) and functionality required.

ISG consists of a framework service and a set of service capability servers. The framework service implements interfaces that are common across all services, for instance, authentication, discovery, load management, and fault management. Service capability servers implement various service capability features (SCFs). These SCFs initially include Parlay/OSA SCFs, that is, location, call control, user interaction, user status, messaging, as well as non-Parlay/OSA-based SCFs, including directory access and PAM. We use presence and availability to refer to a set of applications that allow members in a working group to show presence (online or offline, talking or idle), or to show availability for communication (sending instant messages, text, voice, or chat) among all members of the intelligent network. The ISG also provides Lucent value-added extensions to allow access to non-standardized network resources. This includes access to prepaid and postpaid functionality defined as a mechanism to charge end-user access to applications to a prepaid account. APIs provided include access to query, top-up, and decrement of an end-user's prepaid balance.

AHE Key Benefits and Enabling Applications

The AHE possesses a set of desired capabilities for an application and services hosting platform in the next-generation telecommunication networks. The AHE provides an *open* service platform with standards-based APIs and programming languages enabling the service providers with the vendor and network *interoperability* and the extensive use of independent software vendor and application service provider developed applications. The AHE supports a wide range of Internet/PSTN (public switched telephony network) protocols such as the SIP, JTAPI-based Lucent Softswitch API, LDAP, Parlay/Open Service Access, SS7 [3], VoiceXML* [16], Wireless Markup Language (WML) [18], and Extensible Markup Language (XML) [19]. The AHE is built on commercially available hardware and software with carrier grade *high availability and reliability*. The platform is *scaleable* by design such that the service providers can select the right version of the AHE based on their current demand and grow with the future needs. The AHE can simultaneously run multiple service applications on the platform. The AHE is a next-generation services platform for service providers and carriers to offer a wide of variety of advanced telecommunication services on the converged voice and data networks.

The AHE allows service providers to take full advantage of the service innovations available from the Internet. Service providers can create innovative services using the Internet technologies and deploy them on the service platform. The AHE provides an evolutionary platform that builds on the strength of Lucent's intelligent network (IN). The service providers can enhance their current IN service offerings, protect the existing IN investment, and enter the

Internet arena by creating next-generation converged services. The AHE supports the latest Internet protocols, Internet programming languages, and open-network APIs to allow service providers to create next-generation converged services that are truly access- and network-independent.

The *Packet*IN AHE provides capabilities for development and deployment of advanced services. The following are just a few examples of the kind of applications that the Lucent *Packet*IN AHE can support: IP virtual private network, IP advanced routing services, SIP-based call management services, presence and availability management, IP prepaid, location services, VoiceXML-based applications, and unified messaging/communication. The following section will be dedicated to service examples in relative detail built using these capabilities.

Building Converged Services

To validate the effectiveness of the service construction platform, application and services are built leveraging available platform assets. An example of services that can be implemented on the AHE is the enterprise communication portal (ECP).

Enterprise Communication Portal

The ECP is used to demonstrate rapid creation of added value e-services portable to a variety of Lucent switching and network systems. **Figure 4** illustrates a sample service page provided by the ECP with a user *contact list* including several examples of

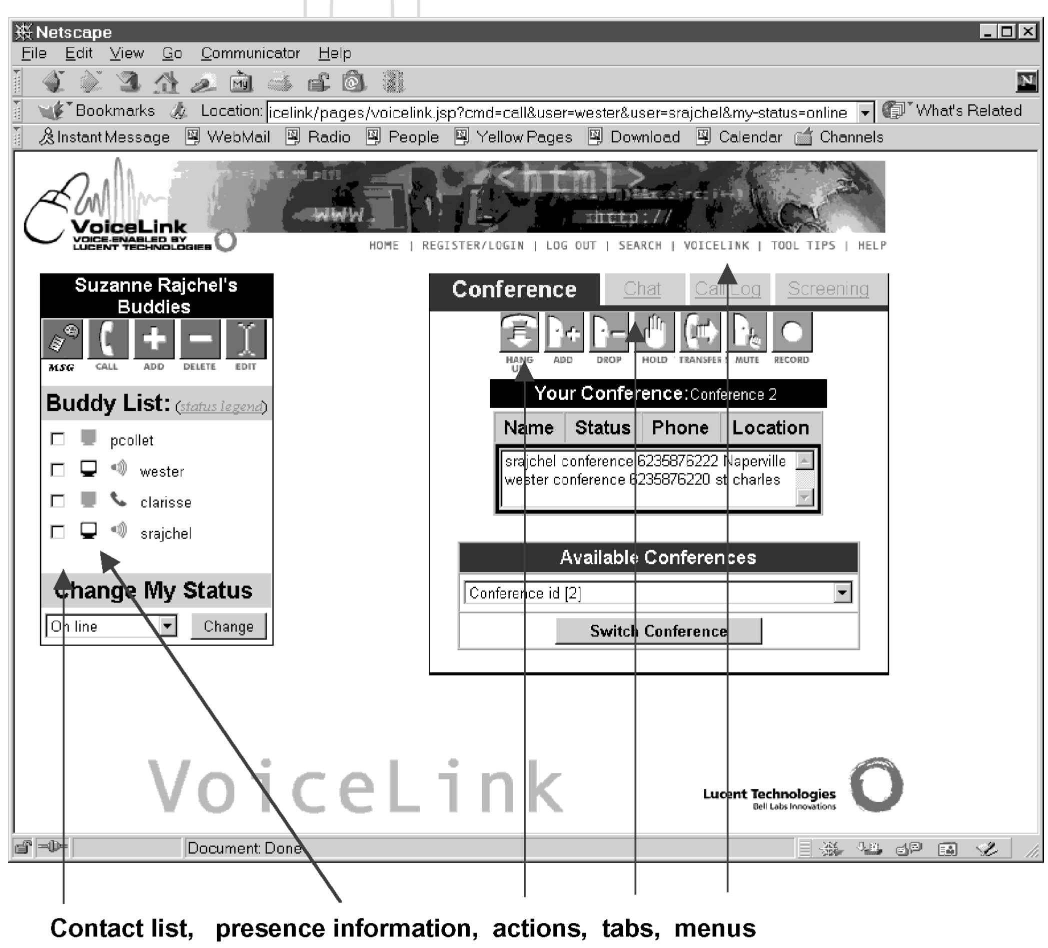

Figure 4.
Sample enterprise communication portal (ECP) Web page.

presence_information display. The *action* section displays the name of the service that is being currently used, and the actions enabled for this service (in this case a conference call with the list of participants). Other services are accessible as *tabs* in the action section or as part of a *menu* provided on the Web page.

Overview

Enterprise users of the ECP can create contact lists associated with work groups or projects. New contact persons can be added to the list via searches in an enterprise directory database, or selected from previous calls (that is, collected in the *CallLog*), or from instant messages received during normal business activities. Presence information is collected from the network (for example, phone, computer, or personal digital assistant activity) and used to update the status of each participant in the current contact list. Each participant, registered and using the ECP, can quickly identify available contact resources and call them directly via the *Contact List* graphical user interface. Depending on the feature set available in the telephony network, a number of participants can then be added or dropped from the call, creating a dynamic working group conference managed easily via the *Conference Call* interface. Call placed and received are logged per user and can be exported from the *CallLog* to personal computer (PC) applications to quickly create reports, follow up on sales or generate bills for services.

These are a few examples of enhanced productivity features for enterprise communication that can be offered in the ECP, and created and deployed using the *Packet*IN AHE.

Technological Assets

To realize ECP services, there are number of technologies that must be available. The AHE provides a substantial set of assets, as well as a foundation for incorporation of new assets. **Table I** provides a list of technology for the realization and delivery of ECP services.

The *Packet*IN AHE enables the service creation, implementation, and delivery of services combining these new assets as exemplified by the ECP. For example, in the ECP application, Parlay provides the programming interface between Web objects and the call engine, H.323 and SIP deliver call control messages to IP end-points, and media gateway control protocol [10] provides an interface to control media communication gateways. The AHE's SLEE and Web server assets provide the foundation for developing ECP server-based applications and dynamic presentation capabilities. In the ECP implementation, Tomcat [15] is used to provide the Web object framework supporting the service logic; the AHE also provides WebLogic [17]. Jabber [5] provides the implementation framework for instant messaging (IM) services in the ECP system. The ECP combines these technologies to give user access to enhanced communication services for the enterprise. Users of the portal can then view and manage all their services via a simple browser using Internet protocols.

Services List

The ECP application is a Web-based business communication portal that extends regular portal services like *Contact List* and *Presence Information* with telephony services. The main services offered by ECP are:

- *Contact list management*. Provides a convenient way to manage frequently-used contacts, and connect with them directly without time consuming lookups or sequential dial attempts.
- *Click2Dial*. Enables productivity-enhancing call management using a PC Web browser as a communication dashboard (for example, instant conferencing, and virtual attendant features.) The user can initiate conference call from a *contact list* or from the call history in one click.
- *Presence information*. Indicates to a caller the *availability* and *best method* of communication (for example, phone, chat, or e-mail) for target contact. Presence information saves time and enables more calls to be completed on first attempt.
- *CallLog*. Maintains a call log used to capture call activity. A Web interface provided with CallLog enables users to place an call from a list of previously received calls. The CallLog provides record-keeping for individuals, and corporate tracking allows users to annotate calls and use PC edit functions to generate reports.

Table I. Technology for the realization and delivery of enterprise communication portal (ECP) services.

Technology	Sample Protocol and Function	Platform Support
Open application program interfaces	Parlay/OSA/PAM	AHE, ECP
Open standard communication protocols	H.323, SIP, MGCP	ISG
Telecommunication standards	UMTS, 3GPP, 3GPP2	—
Personal computer, network appliance, personal digital assistants with thin client	Micro-browser, WAP, WML	—
Web application server providing user session support, information repository, and sharing	Tomcat, WebLogic	AHE
Location and presence information services	PAM, monitoring of multiple devices activity, usage sensing, and location information provided by the converged network	ISG, AHE
Web-based information	Data storage, information retrieval, data sharing, resource location, maps and direction, reservation systems	AHE
Instant Messaging service	Jabber, Yahoo!†, AOL IM	AHE, ECP

3GPP—Third Generation Partnership Project
AHE—Application hosting environment
AOL—America Online*
ECP—Enterprise communication portal
IM—Instant messaging
ISG—Intelligent service gateway
MGCP—Media gateway control protocol
OSA—Open service access
PAM—Presence and availability management
SIP—Session initiation protocol
UMTS—Universal mobile telecommunication system
WAP—Wireless application protocol
WML—Wireless Markup Language

*America Online is a registered trademark of Quantum Computer Services, Inc.
†Yahoo is a registered trademark of Yahoo! Inc.

- *Web-based call feature activation*. Allows the user to manage its phone services through a Web interface. For example, it enables the user to establish and activate or deactivate call forwarding rules.
- *Enterprise directory interactions*. Allows users to query the enterprise directory. The results are hyperlinks ready to initiate calls or e-mail, and can be used as input into a *contact list*.
- *Instant messaging*. Allows users to exchange text-based messages in real time and independently of any other services being used at the time (for example, one can initiate an instant messaging session in parallel to a conference voice call to exchange confidential information pertinent only to the two parties involved).

Service Architecture

ECP applications require a different kind of architecture dedicated to the creation of a new class of Web-enabled converged services (for example, caller-ID linked to personal Web page). This framework is developed on top of the *Packet*IN AHE using the AHE-defined programming interfaces. **Figure 5** shows a high-level view of this architecture.

The service architecture comprises several components that provide database connectivity, presence and instant messaging server, and call control engine interactions, as well as support for different endpoints. Standard protocols are used to communicate with end-user devices (for example, hypertext markup language [HTML], H.323, and SIP) while open APIs (for example, Java servlet pages [JSP], Parlay, LDAP, and SQL)

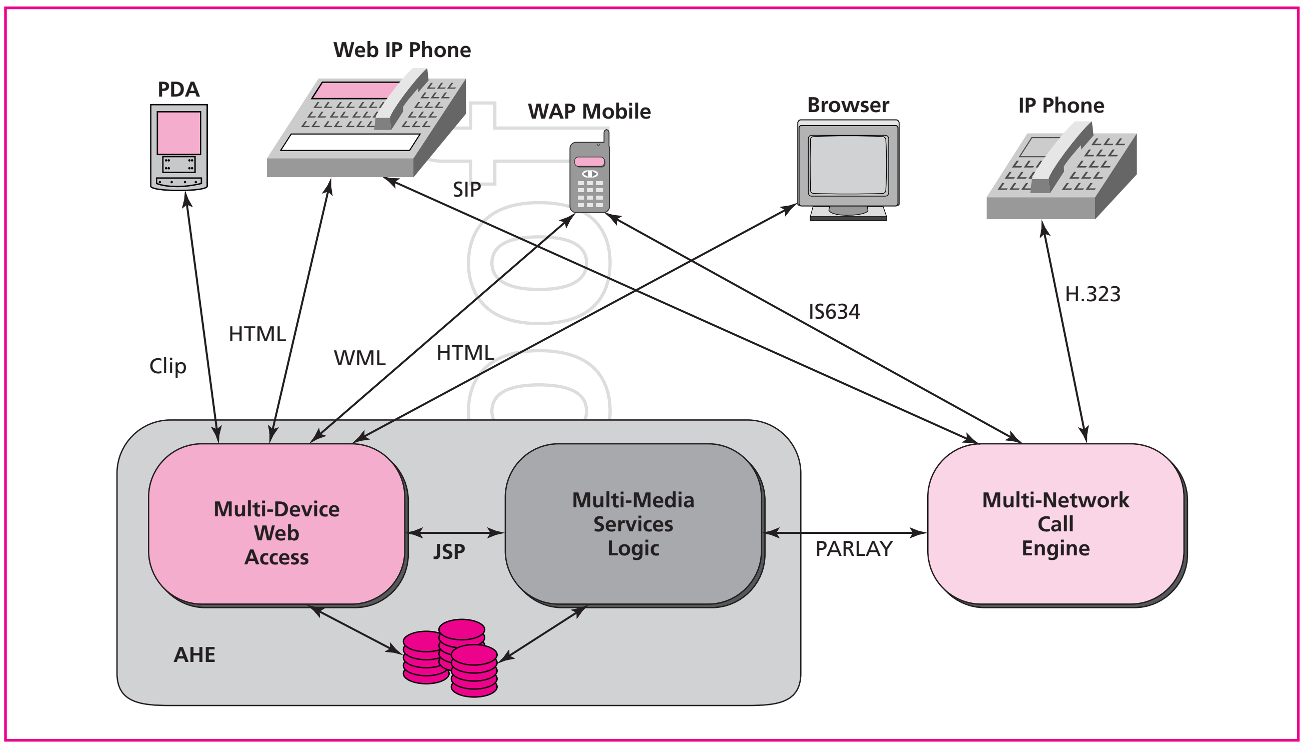

Figure 5.
Enterprise communication portal (ECP) service architecture.

are used for intersystem communication and control. This architecture enables the service logic to provide a service representation to the user using Web display, while driving a call engine to control telephony services using standard voice protocols independently.

Applications running on top of this architecture can operate entirely browser-based (no custom client software required), to allow anywhere, anytime access to mission-critical information from a variety of devices from laptop, to handheld device, to portable phone.

As **Figure 6** shows, there are two main areas in the service architecture as defined in the AHE. The Web access is provided by the Web server and the services logic by the SLEE in the AHE.

- *Web access*. A Web server handles requests from various types of devices and responds with an appropriate answer (for example, HTML for regular browser, WML for wireless application protocol [WAP*] devices).
- *Service logic*. For example, a Java Web server supports the user sessions through a set of Java

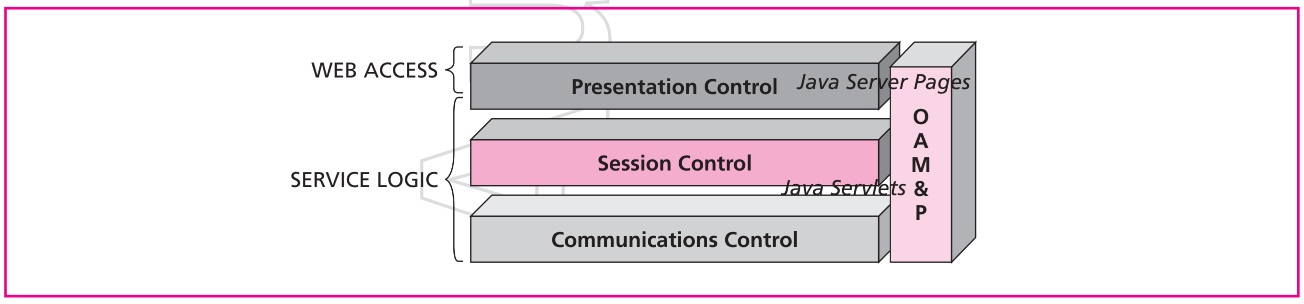

Figure 6.
Enterprise communication portal (ECP) functional layers.

servlets [7], accessing several databases for regular information manipulation, and passing over control to the call engine for the telephony services.

Service Framework

The ECP is built using three functional layers. The *Web access* and the *service logic* components shown previously are decomposed into functional layers in Figure 6. Other supporting functional layers (for example, databases, measurement, billing) are also provided by the AHE.

The following list describes the main functional layers of ECP and their mapping into the AHE programmable services platform:

- *Presentation layer*. The presentation layer is specific to the application being built for ECP. This layer manages the user interface and the rendering of the features provided by the lower service layer. Most of the presentation layer is composed of JSP or HTML pages. This layer is responsible for handling user interactions and for updating presence information on the user. The presentation layer is supported by the SLEE by providing the dynamic generation of HTML pages.
- *Session layer*. This is a set of core components that provides general services such as *logging*, *database connectors*, *administration*, *intercomponent messaging services*, and *user data representation*. These components are shared by the applications, and they are the basic blocks to build converged services. The session layer is supported by the SLEE and utilizes the application-specific components, namely *database connectors* for storing state information. The specific interfacing to the AHE databases is supported using JDBC. The SLEE also provides the infrastructure to support interprocess communication through its message handling facilities.
- *Communication layer*. The communication layer components provide the interface to interwork and manage the various resources used for access to network resources. The communication control layer spans multiple communication boundaries: telephony, presence, location, and instant messaging. Each of these services interacts with a network resource via Parlay, providing an open API for most of the communication media. The interface between the service layer is provided through a *Parlay connector*. This is an abstraction above the Parlay API allowing the functions for management of calls. The basic functionality implements call *creation*, *adding*, and *dropping* of endpoints. Additionally, this connector provides a multithreaded platform for handling requests to support the use of simultaneous user Web devices. This layer is implemented in the SLEE and is supported by the Parlay SCFs for performing third-party call control, access to location-based services and the presence capabilities provided through the PAM services.
- *Measurement*. This layer is provided by the *Packet*IN application server measurement system and provides usage reports on portal activities. There is also support for internal monitoring of application and server activity to provide tuning of the infrastructure.
- *Billing*. This layer provides access to the AHE billing subsystem. The basic billing primitives offered by the AHE allow the ECP to offer customized billing services to enterprise clients.

Web Application Server

A Java Web application server, provided by the AHE, is used to enable Web access to ECP from the enterprise PC clients providing:

- Separation of Web contents, service-oriented code, and template data (that is, JavaBean*, Enterprise JavaBeans*).
- Management of user session state on the server between requests (that is, servlet session).

This new generation of Web server provides the capabilities of serving highly dynamic pages to a large amount of users. The dynamic nature of converged services takes full advantage of Web application servers through JAVA servlets and associate technologies, such as remote backend databases via JDBC to provide personalized services, and cluster Web servers to provide high reliability that our customers are expecting from Lucent products.

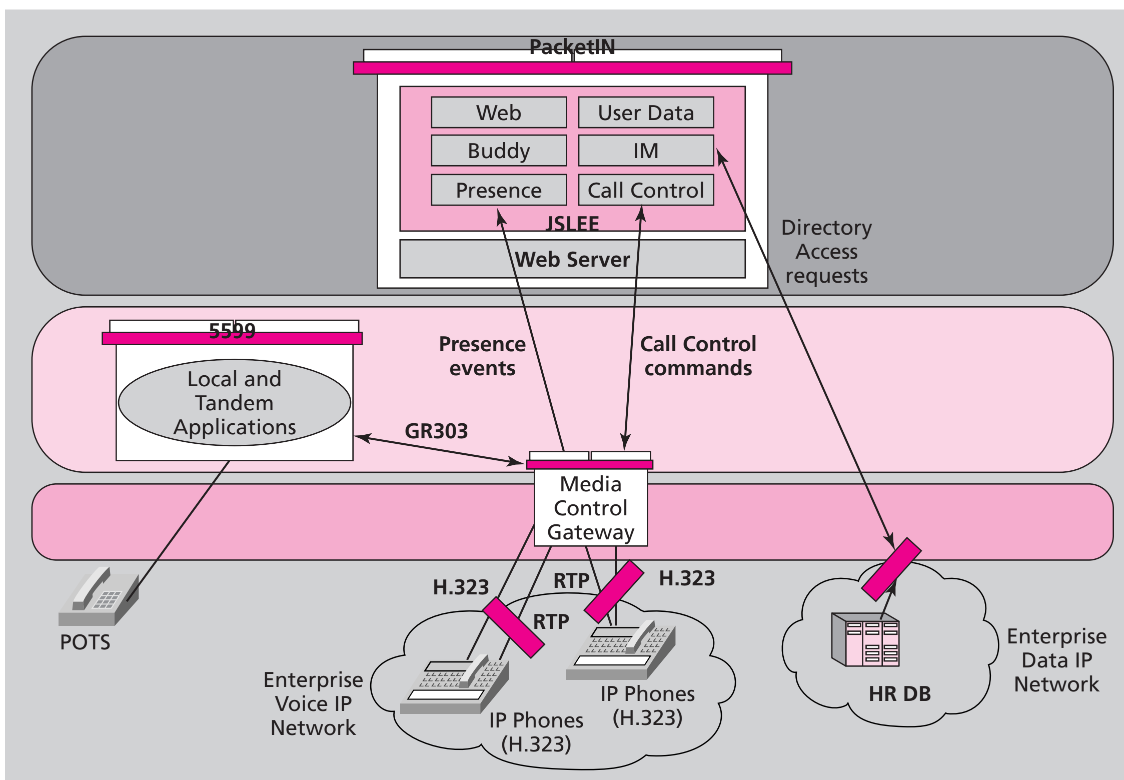

Figure 7.
Application with a call engine.

Call Engines

The call engine is external to the ECP service architecture. The ECP gives control to the call engine in the *communication control* layer via the open API provided by the *Packet*IN AHE. ECP service objects call the Parlay API to create calls, and add or drop parties on behalf of the user when such services are initiated from the ECP service session. In the ECP sample application illustrated in **Figure 7**, a media control gateway is used for the call engine. ECP can also work in cooperation with other platforms that support an open API interface, such as the Lucent Softswitch.

External Resources

The ECP services also rely on a variety of resources accessed via the *Packet*IN AHE and supported by the converged network. See **Table II**.

The accessibility of these resources via a single-application hosting environment via open API and programming languages greatly facilitates the

Table II. Resources accessed via the *Packet*IN™ AHE.

Resources	Platform Support
Media servers and gateways	ISG
Presence and Instant Messaging servers	AHE
SQL and LDAP Databases	AHE
Location servers	Network
Call engines	Network
Information servers	Network
Billing systems	AHE
Registration servers	AHE, Network

AHE—Application hosting environment
ISG—Intelligent service gateway
SQL—Structured query language
LDAP—Lightweight directory access protocol

Panel 1. Abbreviations, Acronyms, and Terms

AHE—application hosting environment
API—application programming interface
BILL—billing
CORBA—Common Object Request Broker Architecture
ECP—enterprise communication portal
eSAE—enhanced service authoring environment
eSD—enhanced services directory
eSM—enhanced service manager
HTML—hypertext markup language
IN—intelligent network
INIT—initiation
IP—Internet protocol
ISG—intelligent service gateway
JDBC—Java database connectivity
JTAPI—Java Telephony API
JSP—Java servlet pages
LSS—Lucent Softswitch
MEAS—measurements subsystem
MSGH—message handler
ODBC—Oracle database connectivity
OSA—open service access
PAM—presence and availability management
PC—personal computer
PSTN—public switched telephone network
SCF—service capability feature
SIP—session initiation protocol
SLEE—service logic execution environment
SMPP—short message peer-to-peer protocol
SPMAN—service package manager
SQL—structured query language
SS7—Signaling System 7
TCP/IP—transmission control protocol/Internet protocol
VPN—virtual private network
WAP—wireless application protocol
WML—Wireless Markup Language
XML—Extensible Markup Language

innovation and creation of new potential revenue-generating services. In particular, the availability of registration, presence and location information, business data, call control, management, and billing from a single unified framework enables more rapid creation and deployment of new converged services as illustrated by the ECP example.

Conclusion

The Lucent *Packet*IN service platform, in which *Packet*IN AHE plays a critical role, offers opportunities to develop and deploy advanced services rapidly and cost-effectively on the next-generation converged networks. The enhanced programmability supported by the open platform, open API support, scalability, and reliability capabilities of the *Packet*IN AHE make the *Packet*IN platform a powerful vehicle for service providers to offer advanced services and generate revenue in multi-vendor, multi-network, and multi-protocol environments of next-generation communication networks. The ECP provides a significant breakthrough in the use of Web-centric programming for providing a new paradigm for offering telecommunication services. At the core is a layered approach providing user-level abstraction to open up the packet network to a new generation of developer for providing enhanced and converged services. The use of open APIs provides the opportunity to utilize this technology across multi-vendor platforms, and enables new levels of programmability.

Acknowledgments

The authors would like to thank Mike Buckler, Jim Frasca, Jeff Cortley, Ajitkumar Rudran, Wilbur Bright, Jan Warpinski, Leslie Franklin, Rajesh Shah, Paresh Shah, Rhonda Blades, Jack Kozik, Doug Varney, and the AHE development team for their contributions and support for the AHE work and the paper. We would also like to acknowledge Trent Nanke, David Ayers, Douglas Dowden, and Peter Lessek for supporting this work. This work would not have been possible without our partnership with New Business Opportunities, AGCS, LSS, and eServices. We are thankful to all our partners—Bill Jenkins, Suzanne Rajchel, John Gibson, David Saar, Tushar Shah, Mavies Bates, Eric Beyler, and Eric Edberg. The authors are also grateful to the reviewers for their insightful and valuable comments on the original form of this article.

*Trademarks

CORBA is a registered trademark of Object Management Group.

Enterprise JavaBeans, JAIN, Java, JavaBeans, and JDBC are trademarks of Sun Microsystems, Inc.

Oracle is a registered trademark of Oracle, Inc.

Solaris and Sun are registered trademarks of Sun Microsystems, Inc.

VoiceXML is a trademark of VoiceXML Forum.

WAP is a trademark of the Wireless Application Protocol Forum Ltd.

References

1. 3GPP TS 29.198 v3.2.0 OSA API – Part 1 (Release 1999).
2. 3GPP TS 29.998 v3.2.0 OSA API – Part 2 (Release 1999).
3. ETSI ETS 300 287-1: ISDN; SS7; Transaction Capabilities (TC) version 2; Part 1: Protocol specification.
4. H.323, Framework and wire-protocol for multiplexed call signaling transport, <http://www.itu.int>.
5. Jabber, Open Source XML-Based Interoperable Instant Messaging and Presence Management Platform, <http://www.jabber.org>.
6. JAIN API's, <http://java.sun.com/products/jain/>.
7. Java Server Pages, <http://java.sun.com/products/jsp/index.html>.
8. JTAPI, Java Telephony Application Programming Interface, <http://java.sun.com/products/jtapi/>.
9. Lightweight Database Access Protocol, Internet-Draft RFC 1777 at <http://www.ietf.org/ietf>.
10. Media Gateway Control Protocol (MGCP), <http://www.ietf.org>.
11. Open Service Access (OSA), <http://www.3gpp.org>.
12. Parlay APIs 2.1, <http://www.parlay.org>.
13. Presence and Availability Management Forum (PAM), <http://www.pamforum.org>.
14. Session Initiation Protocol, Internet-Draft: 2543, <http://www.ietf.org/ietf/1id-abstracts.txt>.
15. Tomcat, Open Source Implementation Java Servlet and Java Server Pages Engine, <http://jakarta.apache.org>.
16. VoiceXML Forum, <http://www.voicexml.org>.
17. WebLogic, BEA Systems Commercial Implementation Java Servlet and Java Server Pages Engine, <http://www.bea.com>.
18. WML, <http://www.wapforum.org>.
19. XML, <http.//www.w3.org/XML>.

(Manuscript approved April 2002)

*YANG CHEN is a member of technical staff in the Packet Intelligent Network (*Packet*IN™) Group of Mobility Solutions at Lucent in Cary, North Carolina. His work in the telecommunications field includes intelligent network (IN) service planning, IN network dimensioning, Internet call management and call center services, location-based voice portal services, and* Packet*IN application hosting environment. His current work focuses on session initiation protocol (SIP) service platform and applications. He has conducted research and authored numerous papers on these topics, as well as on multilevel optimization theory/applications, network planning, and transportation planning in various international journals and books. Dr. Chen received a B.S. degree in mathematics from Shandong Normal University in Jinan, China, an M.S. in operations research from Academy of Sciences of China in Beijing, and a Ph.D. in computer science and operations research from University of Montreal in Canada.*

OLIVIER B. CLARISSE is a distinguished member of technical staff in the UMTS Trailblazers and Core Development Department of Lucent's Wireless Network Group in Naperville, Illinois, where he is currently working on UMTS voice services and wireless/Internet convergence. His recent work includes the Internet and broadband enhanced telephony project with emphasis on Internet technologies and data-enhanced telephony services. Previous works include service creation, multimedia clients and services, embedded devices, client mobility, advanced packet-based services, and visual programming. His related interests include the 3GPP, SIP, and Styx® families of protocols, efficient software distribution architectures, pseudo-open source Inferno® system, and the open source Plan9™ and Linux systems. Dr. Clarisse co-chaired the Dynamic Object Workshop in Object World East in 1996, and was president of the Association of Lisp Users between 1995 and 1998. Dr. Clarisse holds an electrical engineering diploma from the École Supérieure d' Électricité in Gif-sur-Yvette, France, and a Ph.D. in electrical engineering from the Illinois Institute of Technology in Chicago.

PASCAL COLLET, a member of technical staff in the UMTS Trial Department of Lucent's Mobility Solutions Group in Naperville, Illinois. He is currently working on the integration of terminals with Lucent's UMTS solution. Dr. Collet's areas of interest include *packet-based multimedia communication platforms, distributed architectures, and embedded systems. In 1999, he received the Bell Labs President's Silver Award for his work on the 7R/E Call Feature Server Breakthrough Team. Dr. Collet holds an M.S. in electrical and computer engineering from IRESTE in Nantes, and a Ph.D. in computer engineering from the University of Nantes in France.*

MARK A. HARTMAN is a technical manager in Mobility's Intelligent Networking Group at Cary, North Carolina, where he is responsible for the architecture of the Packet*IN product line. Mr. Hartman received a B.S. degree in computer science and mathematics from Northwest Missouri State University in Maryville, and an M.S. degree in computer science from the University of Illinois at Urbana-Champaign.*

LUCIANO RODRIGUEZ is a member of technical staff in the Platform Definition Development (former SPEED) Department of Lucent, Wireless Mobility Solution in Naperville, Illinois. As part of the Internet and broadband enhanced telephony project, he developed new services to exploit the platform capabilities. Currently, he is working on the distributed SS7 project in the SPEED platform. His areas of interest include software process improvement using business technologies such as behavior and domain engineering. He holds an electronic engineering degree from the USM Technical University in Valparaiso, Chile, and an M.S. degree in mathematics from the University of Puerto Rico.

LIZETTE VELAZQUEZ is technical manager in the Platform Definition and Development Department at Lucent Technologies in Naperville, Illinois. She currently leads the Advanced Development Environment Solutions Group and is responsible for providing current engineering services, SW build administration, and enhanced support services specializing in the management and engineering of computing environments, laboratories, and network resources. She is also a member of the Object Review Consultancy Associates (ORCA). Dr. Velazquez holds a B.S. in mathematics from the University of Puerto Rico in Rio Piedras, an M.S. in computer science from Rensselaer Polytechnic Institute in Troy, New York, and a Ph.D. in computer science from the University of Iowa in Iowa City.

BRUCE A. WESTERGREN is a member of technical staff in the Platform Definition and Development Department of Lucent's Wireless Networks Group in Naperville, Illinois, where he is currently working on the SPEED project developing communications software for the connection management framework. His areas of interest include distributed software computing, converged voice and Internet programmable architectures, and packet-based communications technologies. Mr. Westergren holds a B.S. degree from the University of Illinois at Urbana-Champaign and an M.S. degree from the DePaul University in Chicago, Illinois. Both degrees are in computer science. ◆