Global Journal of Computational Intelligence Research.
ISSN 2249-0000 Volume 1, Number 1 (2011), pp. 83-88
© Research India Publications
http://www.ripublication.com/gjcir.htm

VoIP Application Integration Services: An Analysis

Anju Kumari and Mohan Mudgal

Research Scholars, Singhania University,
Pacheri Bari, Jhunjhunu, Rajasthan, India
E-mail: anju.chahal1@gmail.com, mohanmudgalckd@gmail.com

Abstract

Service providers and enterprises expect voice over Internet Protocol (VoIP) to enable third-party application development, allowing them to mix-and-match best-of-breed applications from multiple vendors. By taking advantage of applicable techniques employed for Web-based services, service broker functionality deployed in the network will provide a framework for specifying VoIP application interaction rules. However, each unique VoIP deployment will require development of a complex, customized, domain-specific set of interaction rules for the service brokers. The complexity of VoIP application interaction rule development in a multi-vendor environment will provide Lucent with an opportunity to sell application integration services to enterprises and service providers.

Introduction

One of the most compelling drivers of voice over Internet Protocol (VoIP) is the potential it offers for third-party application development. Cost savings alone may not be compelling enough to drive VoIP deployment beyond niche markets, but VoIP service architectures also enable development and deployment of best-of-breed applications from multiple vendors, encouraging a greater degree of innovation by more competitors and driving down total cost of ownership. Just as with Web-based services, however, integration of applications from multiple vendors will require a significant, complex, unique effort to properly integrate the applications into a seamless user experience. Consider, for example, a case in which a user has subscribed to service with two separate application servers. If one application believes that all calls should be diverted to voice mail and another application believes that all calls should be forwarded to the user’s wireless phone, which service gets its way and how is that determination managed? Will the applications be attempted serially, in which case the first one asked wins? Which should be asked first? Will the applications be attempted in parallel, in which case they may recommend different treatment? What are the criteria to determine which treatment to apply? The answer may be as simple as declaring that one application always takes precedence over another, or it may be a complex set of rules that relies on criteria like the time of day or what activity is indicated in the user's calendar.

Abbreviations, Acronyms, and Terms

IETF	: Internet Engineering Task Force
IP	: Internet Protocol
SIP	: Session Initiation Protocol
TCAP	: Transactional Capabilities Application Part
VoIP	: Voice over IP

Applying Web-Based Paradigms to VoIP

In order to understand how application integration can be accomplished for VoIP, it is helpful to examine the mechanisms used for Web-based services and their applicability to VoIP. There are, essentially, two models for managing content from multiple Web servers simultaneously: separate non-interacting sessions and custom-designed/configured software. They are discussed in the following subsections.

Separate Non-Interacting Sessions

Separate non-interacting sessions is a method of rendering content from multiple Web-based services in separate windows, frames, or channels. Typically, the service providing the content must be selected explicitly by the user; in the case of frames and channels, it may be implied by the user's explicit selection of a service in the parent window. To apply this method to VoIP, we would consider the phone as the client and the call as a session. This implies that, if this technique were to be used for VoIP, there would be only one application server per call and the user would have to specify which application server to use for each call. This approach, however, will not provide a service that meets subscribers' expectations. If a user is subscribed to abbreviated dialing and least-cost routing, the user will not want to have to explicitly choose between those services based upon the context of the call. The user will expect that the phone system will apply the appropriate routing service, based upon dialed digits. Users expect most terminating features, such as call waiting, call diversion, and distinctive alerting to be applied automatically, but with this model, there is no opportunity for the user to specify which service should handle any given call. It should be noted that the Internet Engineering Task Force (IETF) is defining a framework to facilitate user interfaces with multiple servers but that framework does not address application interaction management.

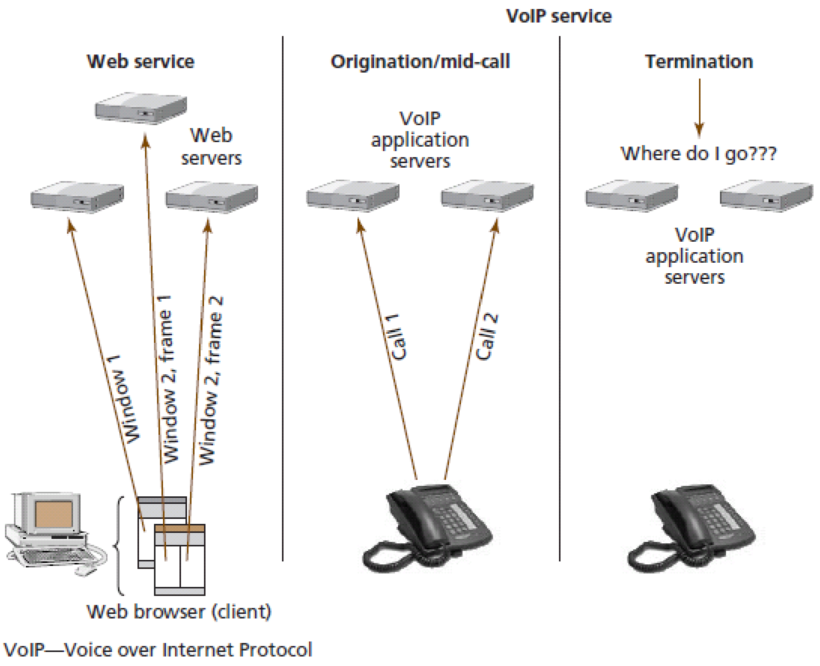

Figure 1: Separate non-interacting sessions.

Custom-Designed/Configured Software

The other method of rendering content from multiple Web-based services is to deploy intermediary software that embodies customized, domain-specific interaction rules. The software may reside within an endpoint, within an application server, or in an intermediary device, and it may be a custom-developed application or a customized configuration of an off the-shelf product. Standards bodies, industry consortiums, and product vendors have developed—and are continuing to develop—frameworks for specification and communication of interaction rules, but each unique set of Web services requires a unique, creative effort to determine its interaction rules. To apply this model to VoIP, the intermediary software, referred to as a service broker, must contain a set of rules to determine which service should be invoked under which circumstances and how interactions should be managed. Origination and mid-call services would flow through the service broker, who would determine which additional services should be engaged in the network and manage the interactions between them. Termination attempts would initially be delivered to the service broker, which would engage the appropriate network services, based on the interaction rules, to determine if, where, and how the call should be delivered.

Application Integration Services

A VoIP service broker may be custom-built to manage the interaction among a specific set of applications, or it may provide a framework in which to specify domain-specific interaction rules. For service brokers that provide a framework in which to specify interaction rules, each unique deployment requires a unique, creative effort to determine those rules. The rules for the telephony application interaction examples used above.

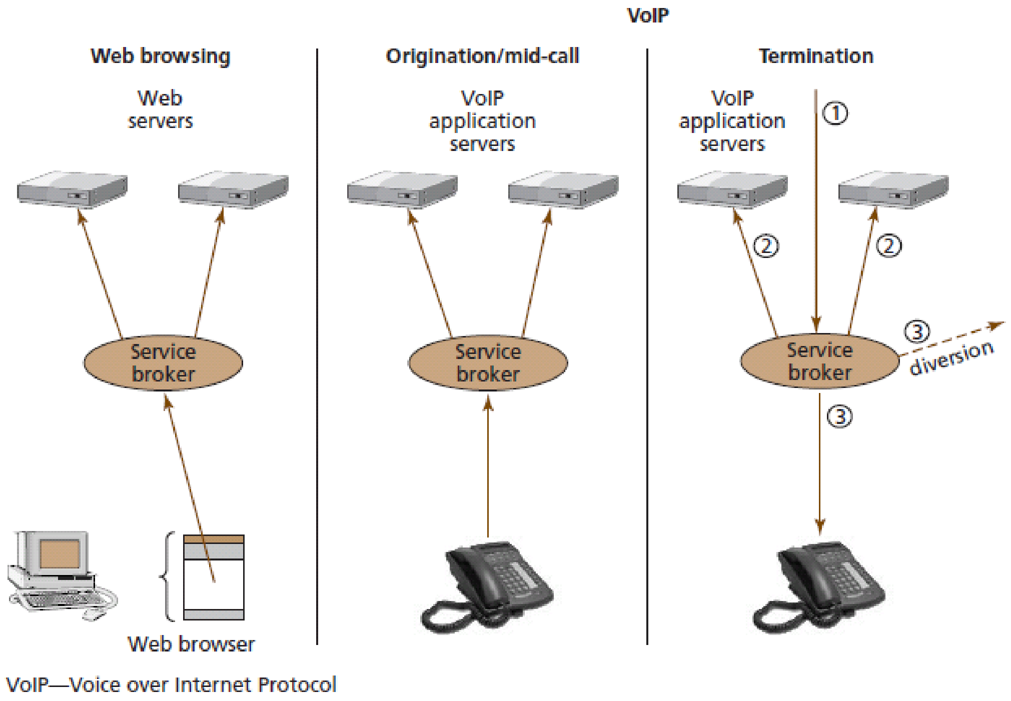

Figure 2: Custom-designed/configured software.

At least-cost routing, call waiting, call diversion, and distinctive alerting) would be trivial to develop, but as more application servers with more complex services are added to the network, the complexity of the rule set grows exponentially. There is additional complexity, as well, in the fact that the service broker is a logical concept that may be broken into multiple tiers, residing on different physical boxes, accessed via multiple protocols, and owned and controlled by different business entities. A service broker function may reside in a session controller/softswitch platform, or it may reside in a separate application server that brokers between other application servers.

Figure 3. Shows a fairly complex environment that is more representative of real-life deployments. In this example, some applications reside in the softswitch and some reside in application servers. This softswitch makes it possible to specify complex rules for determining the termination point, so it can act as a service broker by handing off calls to multiple call-based application servers via interfaces such as Session Initiation Protocol. This softswitch also supports a trigger-based interface via JAIN, Parlay, or Transactional Capabilities Application Part (TCAP), but it assumes that all triggers are being delivered to one application server. That application server must act as the service broker for all other trigger-based application servers. The fact

that, in an Internet Protocol Centrex environment, some application servers may be owned and operated by an enterprise, which implies that every enterprise served, may require unique engineering in the service provider's network. The expectation that every enterprise will have the same set of services available and that they will all be provided and managed by the service provider misses the point that enterprises want VoIP because it gives them the ability to rapidly deploy new services that integrate into their unique business operations environment.

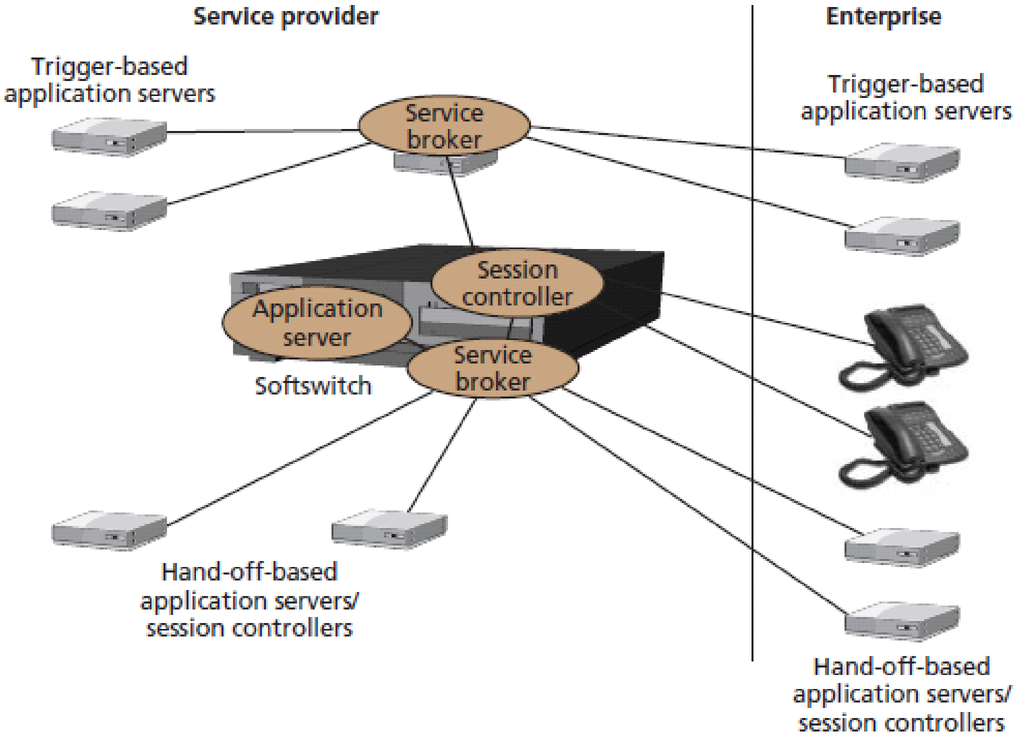
Service provider
Enterprise
Trigger-based application servers
Service broker
Trigger-based application servers
Session controller
Application server
Service broker
Softswitch
Hand-off-based application servers/ session controllers
Hand-off-based application servers/ session controllers

Figure 3: Complex deployment environment.

In all cases, however, it will require a unique solution design activity to identify the customer's functional requirements, ascertain the capabilities, interfaces, gaps, and overlaps within the target product set, and determine the most appropriate methods to manage interactions. The service offering could also take advantage of generic service broker frameworks currently being researched by Bell Labs.

Conclusion

Service providers and enterprises chose to invest in VoIP because they believed that it would facilitate fast-to-market, cheap-to-develop, simple-to-integrate, best-of-breed applications from multiple vendors. Most, however, are still mired in get-started issues like equipment installation/configuration and network design/trouble-shooting, and they do not yet recognize the complexity of the interaction management that will be required to realize the end goal of their investment. Service brokers deployed in the network will provide a framework for managing interaction between multiple application servers, but they will require service providers and enterprises to develop domain specific interaction rules for each unique deployment. As VoIP deployments mature, Lucent will have an opportunity to address the third-party application expectation gap by providing a professional services portfolio to determine the set of application interaction rules and realizing those rules by deploying and engineering service broker functionality in the network.

References

[1] International Telecommunication Union, Telecommunication Standardization Sector, “Packet-Based Multimedia Communications Systems,” ITU-T Rec. H.323, July 2003, <http://www.itu.int>.

[2] J. Rosenberg, “A Framework for Application Interaction in the Session Initiation Protocol (SIP),” IETF Internet Draft, Feb. 2002, <http://www.ietf.org/internet- rafts/draft-ietfsipping-app-interaction-framework-01.txt>.

[3] Session Initiation Protocol, <http://www.ietf.org/html.charters/sipcharter.html>.

[4] Kindberg T et al (2002) People, places, things: web presence for the real world. Mobile networks and applications. Kluwer, Dordrecht, pp 365–376