Dialogic
Making Innovation Thrive™

White Paper

Making It Simple: Bringing Communications to Web-Based Applications

Making It Simple: Bringing Communications to Web-Based Applications

White Paper

Executive Summary

In today's telecommunications market, the service delivery infrastructure of service providers is not always able to keep pace with customer demand. The existing infrastructure for most service providers was built on networks designed for voice-centric, circuit switched services, and not designed to deliver today's content-rich, integrated data services. Current infrastructure limitations create a service bottleneck that limits subscriber growth, increases churn, lowers ARPU (Average Revenue per User), and curtails service-based competitive differentiation.

To be competitive, service providers are looking for new ways to do all of the following:

- Bring new data-rich services to market faster and better than competitors
- Combine IT and telecommunication capabilities
- Enhance existing voice applications, including conference services
- Find new ways to provide third party access to service provider assets
- Increase the quantity and variety of new services
- Make the service creation and delivery process more efficient
- Leverage, extend, and monetize the existing network infrastructure
- Support a flexible infrastructure that can be used to adopt new technologies

One way service providers can address these goals is to change their approach from a product-offering paradigm to a service-offering paradigm, in which the web acts as a service platform. This white paper explores how service providers can use Web 2.0 concepts to service-enable their networks, and thus enhance their position in the new, service-oriented telecommunications market.

Making It Simple: Bringing Communications to Web-Based Applications

White Paper

Table of Contents

Introduction

Innovations such as Apple's iStore and cloud computing have led to an environment where horizontal service platforms are bringing proprietary technology to a wider community. The convergence of IT and the network is upon us — and as new companies emerge with Internet-service-based business models, like those of Amazon and Google, ARPU (Average Revenue per User) is declining across traditional lines of business. A challenge for traditional communications service providers is to move quickly towards this new service-oriented environment, while continuing to support existing technology.

A horizontal service platform creates a shift from a product-offering paradigm to a service-offering paradigm based on convergent communications. If traditional communications service providers do not begin to transform their organizations, business models, processes, and service delivery infrastructure, they risk being reduced to a network utility, while their more nimble competitors reap the profits from value-added services.

This white paper explores how communications service providers, often accustomed to taking months or years to develop and deploy new services, can compete with Web 2.0 companies, which can often develop and deploy new services in days and weeks. It also explores how service providers can create innovation through an external development community, and how they can take advantage of new Web 2.0 concepts to innovate with new service ideas.

Current Market Trends

Although communications service providers do not normally have the same background and approach as true Web 2.0 companies, in practice, they are beginning to take cues from online Web 2.0 players such as Google and YouTube. Traditional operators stand to gain by adopting new business models and emulating Web 2.0 ideas, including converting users into content and application developers, and driving new revenues from user-generated content and social applications.

The following Web 2.0 characteristics have become important to communications service providers:

- Support for the development community through service offerings based on familiar standards such as SIP, Web Services, and Service Oriented Architecture (SOA) technologies.
- Fast and easy service integration through the exposure of standardized service capabilities, such as media enablers with encoding and transcoding capabilities.
- Increased capacity for code reuse by building media and service capabilities once and then allowing multiple applications and third parties to reuse these capabilities. With the capacity, new applications do not need to recreate capabilities that already exist in other applications.
- Consistency, achieved by exposing standard media and service capabilities.
- Efficient content management and distribution.

Emergence of Web 2.0

The Web 2.0 concept has roots with a brainstorming session between Tim O'Reilly and MediaLive International [O'Reilly], and it was refined at the first O'Reilly Media Web 2.0 conference in 2004. Web 2.0 highlights the massive expansion of web connectivity and exponential increase in modes of communication, social interactions, and activities. It focuses on the interactivity between users and their traditional service providers on Service Delivery Platforms (SDPs) — platforms that encourage service innovation by providing standards-based interfaces to the development community. Web 2.0 tools encompass blogs, wikis, podcasts, social networking, and open APIs.

In a Web 2.0 environment, both content providers and service providers can quickly create applications by linking together multiple chunks of reusable, pre-coded capabilities that service providers expose. This process is referred to as "mashing," and it allows for a more rapid and simpler roll-out of new products and services than was previously possible.

Web 2.0 solutions also enable applications to be developed independently of the underlying network technology. This addresses the need to support disparate communication networks, such as High Speed Packet Access (HSPA), Code Division Multiple Access (CDMA), Global System for Mobile Communications (GSM), Wireline, and IP Multimedia Subsystem (IMS), with the same applications and interworking.

Convergence between Service Providers and Content Providers

One effect of Web 2.0 is that the roles of service providers and content providers are converging. Traditionally, service providers provided network connectivity, and sometimes built a limited number of service offerings such as call waiting, caller ID, and SMS directly into the network layer. Content providers, on the other hand, generally focused on the application or services layer by developing new communications and services that ran on existing networks.

The emergence of SDPs is changing these roles. Most service providers are now trying to retain and grow their revenue by leveraging their existing networks and offering content through higher-value services such as voice communications, multimedia, and location-based services. Investing in SDPs is one way for service providers to achieve their goals. SDPs can allow third parties to build and provide content, thus taking advantage of the relationship the service provider already has with the customer, and the service provider's distribution channels, billing capabilities, and device-management capabilities.

Evolving Business Models

A typical business model for the communications service provider has primarily focused on offering voice and connectivity services to consumers and has targeted services to enterprise customers. Many service providers also offer wholesale services to partners who address niche markets (usually targeting the enterprise) that cannot be addressed by their own sales force.

This approach has not proven to be a largely successful in the Web 2.0 world. To compete with Web 2.0-based providers, traditional communications service providers are looking to innovate and offer more exciting services. One strategy for accomplishing this is to embrace third party development, because that is where most innovation lies. Communication service providers can also leverage the SDP's ability to enable services, bundle services, and provide content.

Enabling Services

In traditional business models, most telecom service providers offered services, such as voicemail and hosted PBXs, in which complex functionality was hidden behind a "black-box" interface. Today, service providers can use Web 2.0 principles to offer services as sets of discrete components with standard interfaces. Examples of Web 2.0 telecom services include components that implement IVR, conferencing, and the playing of announcements. Users can combine and/or modify these services to create their own customized applications.

Enabling and Bundling Services

Service providers can also benefit from Web 2.0 principles by bundling their existing telecom assets with new, innovative capabilities. Thus, they can receive part of the subscription, transaction, and advertising revenue associated with these services, instead of monetizing only the data traffic that these services generate.

Providing Content

Providing and enabling the sharing of value-added content is a new business model for traditional service providers, and the concept of an SDP can help service providers manage and distribute this content. For example, service providers can use portals to manage and filter user-generated content and applications.

Characteristics of a Service Developer Platform

Although there is no universal definition of an SDP, most SDP deployments share the characteristics described in Table 1:

Trait	Description
Network access	The mechanism between the service provider network and its subscribers, and the SDP. This usually consists of access networks such as IMS, LTE, or WiMAX, and the value added interfaces that they expose, such as SS7, RTSP, WIN, AIN, and other existing interfaces. The SDP aggregates requests coming from and going to a network supported by the service provider.
Exposure layer	Provides secure and managed access to services and applications by managing and enforcing third-party Service Level Agreements (SLAs) and exposing the underlying services for external consumption. The exposure layer enables service and application developers (internal as well as third party) to access the service provider's network capabilities and services in a controlled, secure, and automated fashion.
Exposed set of service enablers	In most SDPs, the service provider exposes a set of service enablers that encapsulate reusable components and include services such as encoding and transcoding, IVR, network address book, rules engines, presence and location aggregators, messaging (SMS, MMS), and more.
Service creation	Handles "traffic" and includes the eTOM (enhanced Telecom Operations Map) model for fulfillment, assurance, and billing.
Service mediation	Binds together the loosely-coupled service components of the SDP, and leverages SDP architecture to provide facilities for connecting service components in a controlled fashion.
Common services	Services that span the breadth of the other functional areas of the SDP such as service exposure, service enablers, and service creation.
OSS and BSS integration	Operations Support Systems (OSS) and Business Support Systems (BSS) are leveraged and exposed services, just as network elements are in a traditional SDP view. Integration points are realized as service endpoints that are exposed and managed via a central node.

Table 1. SDP Characteristics

Figure 1 shows an SDP that interfaces with the mobile network and with various types of IP networks:

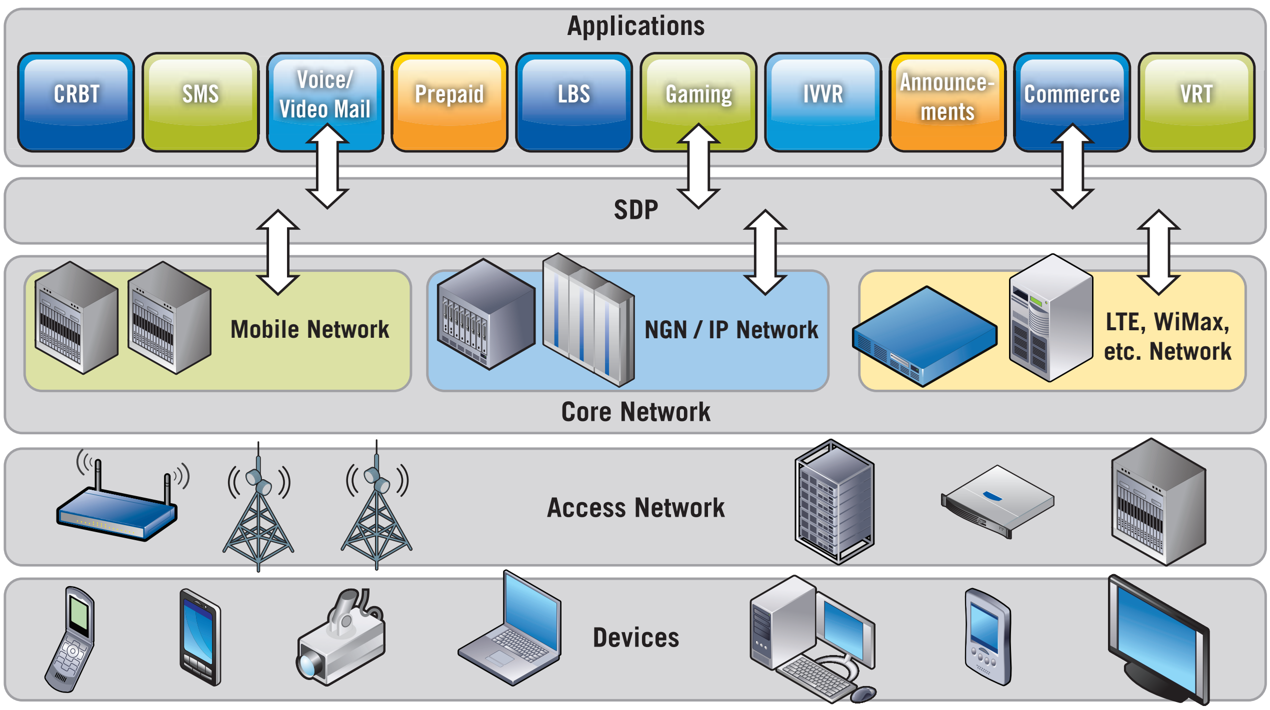

Figure 1. SDP Example

Service-Oriented Architecture

Views on how to implement an SDP are typically supplier-dependent. One way to implement an SDP is to use a Service-Oriented Architecture (SOA) framework, in which applications are constructed from a set of self-contained building blocks called "services". Each service typically encapsulates the code for implementing one type of capability and is abstracted out into a simple programming paradigm. With an SOA framework, content developers can combine, or "mash up," multiple services to create advanced services.

Services created within SOA frameworks exhibit common characteristics, no matter where they sit and what they do. At a high level, such services can be described as building blocks that:

- Provide functionality to systems, services, and subscribers
- Are discoverable, manageable, measurable, and versionable
- Consist of a contract, interface, and implementation

Table 2 describes some specific characteristics of SOA services:

Characteristic	Description
Abstraction	Hides as much of the underlying service logic as possible.
Autonomy	Services have control over the logic they encapsulate.
Composability	Services must be capable of being coordinated and assembled into composite services and applications.
Contract	Adherence to a communications agreement, as defined collectively by one or more description documents.
Discoverability	Services have capabilities that are outwardly descriptive, so that they can be found and accessed via available discovery mechanisms.
Loose coupling with other reusable components	Relationship that minimizes dependencies between reusable components, but requires that the components maintain an awareness of each other so they can interoperate.
Reusability	Services should be designed for reuse so that content providers can easily create applications with them.
Technology	To the extent possible, services should not be tied to any particular underlying technology, delivery channel, or physical location. Services should be written to accomplish their function, regardless of the protocol used to invoke them, their physical location, or the type of hardware or operating system on which they run.

Table 2: Specific Characteristics of SOA Services

SOA-Based Considerations

To compete with Web 2.0 developers, service developers should take into account the following additional SOA-based considerations when building the next generation service delivery architecture:

Fast Service Creation by Composition and Reuse

The next generation of network services and applications require fast service creation because of the shortened marketing, provisioning, and production cycles.

To achieve this goal, existing assets must be reused in an efficient manner, and the underlying architecture must be reactive and adaptable to the ever changing business environment. Because services are accessed via different channels, multi-channel access possibilities should also be considered during service creation.

Access to Network and IT Resources

Access to network and IT resources implies openness at multiple points and levels. Openness is closely related to security. In this context, access has to be considered also through organizational units and their frontiers. Subsidiaries can be service providers or service consumers for each other. Moreover, main organizational units can benefit from other units' services as well as other subsidiaries' services. This leads to the question of the visibility of the services across frontiers, and brings forth related issues such as security, performance, and governance.

Service Control

Proliferation of services implies new practices and methods, so content developers can know what the available services are, how they can access the services, and what the responsibilities and policies are in terms of development, administration, and reuse. This is mainly a governance issue.

Service Quality

Services and reused assets are offered to different organization units and users. These exchanges should take SLAs into account, and the quality of service should be satisfying for all parties. In an SOA model, SLA management is a governance issue, because the customer is an actor with defined rights and responsibilities.

Investment Protection

SOA approaches are closely related to investment protection, because they try to leverage existing technology. Service providers should consider the best way to benefit from and reuse existing infrastructure investments when they adapt new architectures and services.

Lower Risks

Lowering risks requires a progressive approach to change that balances the cost of new developments and the efficient reuse of existing systems. An SOA model creates a suitable environment for managing this type of risk.

Service Enablers

A service enabler is an SDP component that consists of reusable components and functions for providing standard interfaces. Service enablers are intended to speed up service development.

Within a typical SDP, service enablers provide access to a variety of enabling services, including:

- Media control
- Network address books
- Encoding
- Transcoding
- Rules engines
- Presence and location aggregators
- Federated identity management
- Notification engines

Service enablers are usually made available to both the service provider and third party applications through a standardized exposure mechanism, such as an API or signaling protocol.

The concept of service enablers as reusable components and functions is well understood in the traditional “IT” sense, and it continues to evolve in the service provider and telecom domains. Standard IT-related functions can be exposed easily, and the technology now exists to expose network-specific functionality using traditional network technology as well as standard IT technology, such as Java and Web Service APIs. This brings Web 2.0 and SOA into the traditional telecom world.

Enablers have common characteristics and, at a high level, can be described as building blocks that:

- Provide functionality to systems, services, and subscribers
- Are discoverable, manageable, measurable, and versionable
- Consist of a contract, interface, and implementation

Service enablers exist at several levels, or "classes." Traditional enabler functionality defined within most SDPs includes presence, location, and messaging functions. The following additional enabler classes can facilitate the construction of a wide range of converged communications applications:

- Media server control
- Transcoding and rendering
- Device type and capabilities
- User interaction
- Intelligent networking

Media Server Control

Media servers are central to the delivery of IP-based telephony applications. A media server typically performs the following tasks:

- Voice and video conferencing, including mixing multiple audio streams and switching between active video streams
- Playing and recording audio
- Dual Tone Multi Frequency (DTMF) capture and recognition
- Interactive Voice Response (IVF), including VoiceXML parsing and browsing
- Providing announcements to the network

Because an application server typically controls the call flow logic of an application, it also needs to control media server functionality. Although there are several existing standards for controlling media server capabilities, such as Media Server Control Markup Language (MSCML) and Media Server Control (MediaCtrl), most media servers provide their own set of APIs across all of these different technologies. It is expected that service providers will make these controls reusable, so that third parties can control certain media functionalities for their particular applications.

The following types of controls should be reusable:

- Interactive media, including functionality such as playing prompts and recording user input; and handling DTMF, Text to Speech (TTS), Automatic Speech Recognition (ASR), and other media-based processing
- Conferencing, including creating, modifying, and destroying a conference; joining participants to a conference; removing participants from a conference; playing music to a conference; and notifying existing conference participants about newly joined participants
- Blind and bridge transfer
- Play announcements
- Mute/unmute

A new standard called JSR 309: Media Server Control API is currently under development. JSR 309 is a protocol-agnostic service API that abstracts media server control and can be used as a portable interface for creating applications with IVR, conferencing, and speech recognition capabilities. JSR 309 can be exposed as a service enabler within an SDP.

Transcoding and Rendering Capabilities

Because mobile devices support varying standards and offer a wide variety of capabilities, it is expected that third party application providers know the type of device being used by a subscriber, and that the application providers do either of the following:

- Properly transcode and deliver the correct content to the device based on the device description.
- Make the device description available to the service provider, who can properly transcode based on that information.

User Interaction

User interaction functionality includes collecting and making decisions based on user input. This capability can allow user input to be collected via mechanisms such as IVR or keyboard press, and then provided to a third party.

Intelligent Networking

Because third parties might want to provide SS7 services and access SS7-based capabilities, an SDP needs access to gateways in order to deliver of services into and out of the legacy telecom network. SS7 capabilities can be provided by these gateways and exposed as enablers through the SDP.

Role of the Dialogic® IP Media Server in an SOA Framework

The Dialogic® IP Media Server can supply the media processing resources that are required for building rich, interactive services within an SDP that uses an SOA framework. The Dialogic IP Media Server is a software-based, carrier-grade IP media server that can provide many of the service enablers mentioned within this white paper such as interactive media, conferencing, and announcements. Developers can access the enablers provided by the Dialogic IP Media Server through numerous mechanisms, including NETANN, MSCML, and RESTful interfaces. The Dialogic IP Media Server allows a vast array of developers (such as telephony and web developers) to build innovative services easily and quickly.

For more information about the Dialogic IP Media Server, see the Dialogic® IP Media Server Datasheet on the Dialogic website.

Conclusion

With the shift toward Web 2.0 and the convergence of roles between service providers and content developers, service providers are trying to retain their existing network infrastructure and grow their revenue by creating rich-media services that run on both IP and legacy networks. By using services and service enablers on an SOA-based SDP, service providers and their third-party partners can be poised to rapidly create and deploy new services across multiple networks.

Service providers can use the Dialogic® IP Media Server within an SOA-based SDP to provide media-rich services, such as conferencing and multimedia transcoding. These services can help service providers generate new revenue streams and stay competitive in the Web 2.0 world.

References

[O'Reilly] "What is Web 2.0: Design Patterns and Business Models for the Next Generation of Software," September 30, 2005. http://oreilly.com/web2/archive/what-is-web-20.html

Acronyms

AIN	Advanced Intelligent Network
ARPU	Average Revenue per User
ASR	Automatic Speech Recognition
BSS	Business Support Systems
CDMA	Code Division Multiple Access
DTMF	Dual-Tone Multi-Frequency
eTOM	Enhanced Telecom Operations Map
GSM	Global System for Mobile Communications
HSPA	High Speed Packet Access
IETF	Internet Engineering Task Force
IMS	IP Multimedia Subsystem
IPMS	Dialogic® IP Media Server
IVF	Interactive Voice Response
IVR	Interactive Voice Response
LTE	Long Term Evolution
MediaCtrl	Media Server Control
MMS	Multimedia Messaging Service
MSCML	Media Server Control Markup Language
NETANN	Network Announcement
OSS	Operations Support Systems
PBX	Private Branch Exchange
RTSP	Real-Time Streaming Protocol
SMS	Short Message Service
SDP	Service Delivery Platform

Acronyms *(continued)*

SIP	Session Initiation Protocol
SLA	Service Level Agreement
SOA	Service-Oriented Architecture
SS7	Signaling System 7
TTS	Text-to-Speech
WiMax	Worldwide Interoperability for Microwave Access
WIN	Wireless Information Networks

For More Information

Along with the O'Reilly website cited as a reference above, the following articles present information about Web 2.0:

"Web 2.0," Paul Graham, November 2005 at http://www.paulgraham.com/web20.html

"Web 2.0 Reference Center," ProgrammableWeb at http://www.programmableweb.com/reference

"Web Squared: Web 2.0 Five Years On," Tim O'Reilly and John Battelle, 2009 at http://assets.en.oreilly.com/1/event/28/web2009_websquared-whitepaper.pdf

Dialogic

Making Innovation Thrive™

www.dialogic.com

Dialogic Corporation
9800 Cavendish Blvd., 5th floor
Montreal, Quebec
CANADA H4M 2V9

01/10 11588-01