

ARQUITECTURA PARA EL APROVISIONAMIENTO DE SERVICIOS DE VALOR AGREGADO SOPORTADA EN ESB Y SLEE

ARCHITECTURE FOR VALUE ADDED SERVICES PROVISIONING SUPPORTED IN ESB AND SLEE

Gerardo Rojas Sierra

Universidad del Cauca, Popayán (Colombia) • gerardorojas@unicauca.edu.co

Francisco Martínez

Universidad del Cauca, Popayán (Colombia) • fomarti@unicauca.edu.co

Oscar Mauricio Caicedo R.

Universidad del Cauca, Popayán (Colombia) • omcaicedo@unicauca.edu.co

Laura González

Universidad de la República Oriental del Uruguay, Montevideo (Uruguay) • lauragon@fing.edu.uy

Resumen

Actualmente, los requerimientos del mercado en las telecomunicaciones son cada vez más exigentes; los clientes solicitan respuestas rápidas a sus necesidades y facilidad a la hora de comprar y/o personalizar sus productos o servicios; igualmente, los operadores de telecomunicaciones deben atender estas necesidades con el propósito de poder mantenerse en el mercado. Para contribuir a la solución de esta problemática, se requiere definir mecanismos para el aprovisionamiento de Servicios de Valor Agregado (VAS)¹ que se adapten a estas nuevas dinámicas del sector. En este sentido, el presente artículo realiza una propuesta de arquitectura de integración de VAS con servicios IT basada en ESB² y SLEE³.

El SLEE es utilizado para ejecutar la lógica de los VAS en una Red de Próxima Generación (NGN: Next Generation Network) y cumplir con los requerimientos de alta disponibilidad, alto tráfico y baja latencia del dominio de telecomunicaciones. ESB y SOA⁴, por su parte, son empleados para facilitar la

¹ VAS: Value Added Services

² ESB: Enterprise Service Bus

³ SLEE: Service Logic Execution Environment

⁴ SOA: Service Oriented Architecture

integración de la plataforma de telecomunicaciones con los sistemas TI permitiendo así la personalización, venta y tarificación de los VAS con el objetivo de explotar la infraestructura como negocio y brindar valor agregado al cliente final. De esta manera, la arquitectura propuesta separa claramente la infraestructura del operador de telecomunicaciones utilizada para la ejecución de los VAS y la relacionada con su operación y mantenimiento gracias al débil acoplamiento que propone SOA y la excelente integración que ofrece ESB.

Palabras clave: arquitectura de integración, ESB, SDP, SLEE, SOA, NGN

Abstract

Currently, telecommunications market requirements are becoming more demanding; customers require quick responses and efficient procedures when buying and / or customize their products or services and telecommunications operators (Telco) must address these requirements to be competitive in the market. The problem solution consists in to define appropriate mechanisms for provision of Value Added Services (VAS). According to this context, this article presents an architecture proposal to integrate VAS with IT services based on ESB and SLEE.

The SLEE is used to execute the logic of the VAS in a Next Generation Network (NGN) and to meet the requirements for high availability, high traffic and low latency of the telecommunications domain. ESB and SOA are used for the integration of telecommunications platform and IT systems, which improve the customization, sale and pricing processes of VAS. The proposed architecture offers a weak coupling between the telecommunications infrastructure operator used to deploy the VAS and related operation and maintenance processes.

Keywords: integration architecture, ESB, SDP, SLEE, SOA, NGN

Introducción

Actualmente, los clientes requieren de Servicios de Valor Agregado (*Value Added Services*) y para proporcionarlos, los operadores están adoptando nuevas tecnologías de telecomunicaciones, y conceptos derivados del mundo de las tecnologías de la información (Graham, 2009) como NGN (*Next Generation Networks*) y SDP (*Service Delivery Platforms*). En este sentido, las NGN proponen la separación de capas en aras de facilitar la creación de servicios: i) Capa de acceso, encargada de la conexión hacia los usuarios finales. ii) Capa de transporte o núcleo MPLS (*Multi Protocol Label Switching*) sobre la cual se ejecutan varios protocolos como H-248, H-323, SIP (*Session Initiation Protocol*), MGCP (*Media Gateway Control Protocol*) y SS7 (*Signaling System No 7*), entre otros. iii) Capa de control, que permite el acceso a los servicios, los recursos de red y el establecimiento de sesiones. iv) Capa de aplicación, en la cual se despliegan los servicios (Imen et al., 2006). Por su parte, las SDP soportadas con SOA

(*Service Oriented Architecture*) son definidas como soluciones que proporcionan una base unificada para el intercambio óptimo de los servicios entre usuarios, operadores, proveedores de servicio y proveedores de contenido (Johnston et al., 2007) e igualmente, habilitan el rápido despliegue, creación eficiente, ejecución, orquestación y gestión de VAS sobre las NGN (Hanhua et al., 2008). Los beneficios de SOA se traducen en un bajo acoplamiento, interfaces bien definidas, granularidad del servicio y localización independiente, generalmente usando un esquema “*find-bind-execute*” para la comunicación entre proveedores de servicios (Chen et al., 2008); (Ferenc et al., 2006).

EMCALI (Empresas Municipales de Cali), en su negocio de telecomunicaciones no ha sido ajena a esta dinámica sectorial y, por tanto, sus planes de inversión han apuntado en esta línea, realizando gradualmente una migración de la PSTN (*Public Switched Telephone Network*) hacia los servicios de NGN, implementando una capa de aplicación compuesta por las plataformas

ZXUP10 y ZXUMS⁵. De otro lado, la empresa cambió la plataforma IT (BSS/OSS, Sistema Recursos Humanos, Recursos Físicos y Financieros, entre otros) en busca de mejorar sus procesos de negocio. Sin embargo, actualmente la integración entre la plataforma de telecomunicaciones y el BSS/OSS, se realiza mediante el sistema tradicional punto a punto haciendo uso de API's definidas para cada actividad requerida.

Tomando como base el contexto anterior, el presente trabajo propone una arquitectura de integración basada en ESB y SLEE, que permite el aprovisionamiento de VAS para las empresas de telecomunicaciones que cuenten con NGN. El desarrollo del trabajo de investigación, es abordado a través de las siguientes secciones: la sección 2 presenta los principales trabajos relacionados, la sección 3 relaciona las posibles arquitecturas de integración, la sección 4 muestra la propuesta de arquitectura, la sección 5 desarrolla el caso de estudio, la sección 6 comparte los resultados obtenidos durante la fase de experimentación y la sección 7 plantea algunas conclusiones relevantes.

Trabajos relacionados

A continuación se presentan los principales trabajos relacionados con la definición, diseño e implementación de AITE.

“Deployment of Contextual Corporate Telco Services Based on Protocol Adaptation in the NGN Environment” (Cadenas et al., 2010) plantea la implementación de un servicio contextual ofrecido por un operador de telecomunicaciones convergente. La inteligencia del servicio es definida para el enrutamiento y desvío de llamadas, y proporcionada por un sistema de orquestación que utiliza las capacidades del nivel de control de una NGN. El contexto del usuario se almacena en un elemento de red central, con el fin de utilizar la información a través de diferentes servicios. La aplicación demuestra que los servicios contextuales de valor agregado se pueden construir eficientemente con el uso de protocolos y productos disponibles actualmente. En este trabajo, los autores no abordan el aprovisionamiento integral del servicio

y, por tanto, los aspectos de interoperabilidad son dejados de lado. AITE justamente se enfoca en este último aspecto.

“Research of Enterprise Application Integration Based-on ESB” (Jieming et al., 2010) presenta a ESB como una gran infraestructura de software para la integración de aplicaciones, basada en SOA y que soluciona algunas de las desventajas de la topología *hub-and-spoke* (intermediario de sistemas): i) ausencia de control regional sobre dominios locales de integración, ii) inexistencia de procesos de negocio entre departamentos o unidades de negocio, y iii) una limitada integración en las fronteras físicas de segmentos de red y servidores de seguridad. Adicionalmente, los autores también describen los atributos del ESB y detallan sus características principales: i) transparencia de localización, ii) construcción de mensajes, iii) conversión de protocolos de transporte, iv) transformación de mensajes, v) gestión y monitoreo, y vi) seguridad. Aunque los autores presentan a ESB como una excelente solución para la integración de sistemas heterogéneos, no plantean su integración con sistemas de telecomunicaciones ni desarrollan un caso de estudio que valide su propuesta.

“Conceptual Framework for Services Creation/Development Environment in Telecom Domain” (Kryvinska et al., 2008) presenta una definición básica de NGN, de su importancia y necesidad para la convergencia de voz, datos e Internet. También, analiza y compara diferentes tecnologías para el desarrollo y creación de servicios como OSA/Parlay (Unmehopa et al., 2006), Servicios Web (Booth, 2007) y SIP Servlets (Chrighton, 2007). Adicionalmente, en este trabajo, los autores proponen una metodología para la creación de servicios de telecomunicaciones basada en fundamentos matemáticos como modelos jerárquicos y subsistemas paralelos. Sin embargo, la metodología no considera conceptos de interoperabilidad con sistemas IT que son requeridos para un completo aprovisionamiento de VAS.

Arquitecturas de integración

Hub & Spoke: la arquitectura *Hub/Spoke* utiliza un intermediario central (*Hub*) y adaptadores (*Spoke*) los cuales conectan las aplicaciones con el *Hub*. *Spoke* conecta las aplicaciones y convierte el formato de datos de la aplicación a un formato en el cual el

⁵ Son plataformas que proporcionan VAS como mensajería unificada, número portable y soporta estándar parlay.

Hub lo pueda entender y viceversa. El *Hub* manipula todos los mensajes y cuidadosamente transforma y traslada el contenido de los mensajes entrantes hacia un formato que el destinatario pueda entender. Los adaptadores toman datos de la aplicación origen y publican mensajes al intermediador, quien a su vez, hace la transformación, traducción y enrutamiento; y pasa los mensajes a un adaptador de suscripción que envía a la aplicación destino.

BUS: la arquitectura BUS, utiliza una mensajería central (BUS) para la propagación de mensajes. Las aplicaciones los publican usando adaptadores y pueden fluir a aplicaciones suscritas. Las aplicaciones tendrán adaptadores, que pueden tomar mensajes del bus y transformarlos hacia el formato requerido de la aplicación. La diferencia clave entre el bus y *hub/spoke*, es que para la arquitectura bus el mensaje en

la máquina de integración es transformado, ruteado y distribuido en los adaptadores de aplicación y se requiere un adaptador para correr en la misma plataforma de las aplicaciones originales.

ESB: Bus de servicios empresariales es una infraestructura que facilita la arquitectura orientada al servicio. Este suministra *API's* para el desarrollo de servicios y facilita su interacción. Técnicamente, ESB es una columna vertebral de mensajería la cual contiene protocolos de conversión, transformación de formato y ruteo (Chapell, 2004).

La tabla 1 muestra una comparación de las posibles arquitecturas de integración. Como se puede observar ESB muestra los mayores beneficios para la integración, dada la alta escalabilidad, arquitectura orientada al servicio y esfuerzo moderado.

Tabla 1. Comparación de las posibles arquitecturas de integración

| Parámetros de Evaluación | Arquitectura Hub | Arquitectura BUS | |
|--------------------------|--|---|---|
| | | Bus propietario | ESB |
| Esfuerzo de instalación | Menor esfuerzo de instalación comparado con las soluciones de arquitectura BUS | Esfuerzo Moderado | Esfuerzo Moderado |
| Administración | Fácil mantenimiento y administración, por tener un hub central. | Administración puede ser compleja dependiendo de los sistemas integrados. | Administración puede ser compleja dependiendo de los sistemas integrados. |
| Costo | Alto | Alto | Bajo costo porque no usa formatos propietarios. |
| Escalabilidad | Alta si se usan arquitecturas federadas limitado por el hardware y los host. | Alta | Alta |
| Estandares | Usa estandares y formatos internos propietarios. | Usa estandares y formatos internos propietarios. | Basado en estandares. |
| SOA. | Puede ser implementado como orientado al servicio. | Puede ser implementado como orientado al servicio. | Orientado al servicio. |

AITE (Arquitectura de Integración Telecomunicaciones - ESB)

La arquitectura AITE incorpora un ESB como mediador entre el servidor JSLEE y las posibles tecnologías de los sistemas de información, tanto del operador de telecomunicaciones como de terceros; con el fin de obtener una solución completa de integración que cumpla con los siguientes requisitos: i) Adaptación a las necesidades de los proyectos de integración, ya que es capaz de soportar la evolución de los protocolos, las tecnologías de interfaz y el modelado de procesos. ii)

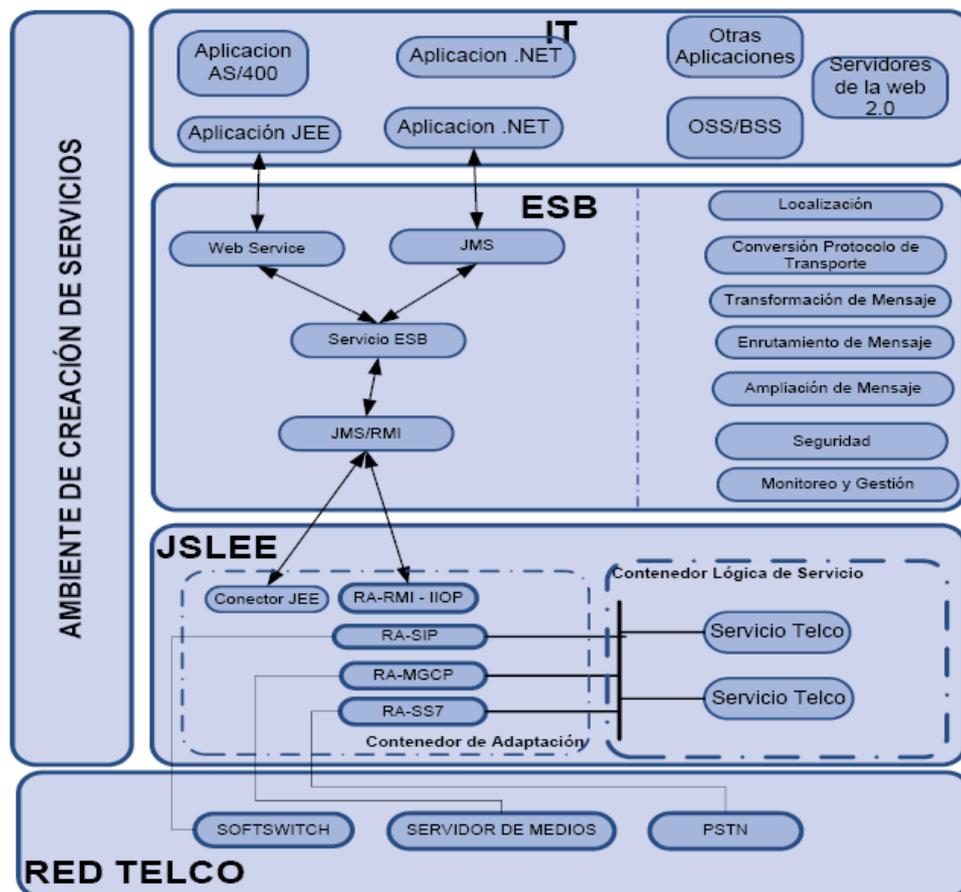
Facilidad en la integración de aplicaciones pertenecientes a cualquier área organizacional de la empresa, utilizando un enfoque unificado e infraestructura común. iii) Simplicidad de diseño y escasas barreras de entrada. iv) Capacidad de reaccionar y responder a las necesidades cambiantes del negocio y el mercado (Figura 1).

La característica más importante de AITE radica en el respeto a la autonomía de los servicios de telecomunicaciones, los cuales continúan funcionando en los servidores JSLEE, cumpliendo así: con la escalabilidad, confiabilidad, disponibilidad, alto

tráfico y baja latencia. Igualmente, el ESB incorpora una integración altamente distribuida ofreciendo una clara separación de la lógica de negocio y la lógica de integración entre los servidores JSLEE y los servidores

de aplicación IT. De esta manera, la incorporación de un nuevo servidor SLEE o IT a la arquitectura, se convierte en un proceso flexible, rápido y menos complejo respecto a un esquema punto a punto.

Figura 1. Arquitectura de integración Telco - IT con ESB



Por otro lado, la inclusión de ESB, brinda a AITE, un conjunto de características relevantes para el proceso de integración de servicios:

- Integración basada en estándares: AITE facilita la integración de aplicaciones creadas con .NET, COM, C#, C/C++ y otros lenguajes de programación específicos que soportan Servicios Web.
- En cuanto a manipulación de datos y procesos de negocio, AITE utiliza estándares basados en XML (*eXtensible Markup Language*) como WSDL (*Web Service Description Language*) y BPEL (*Business Process Execution Language*).
- Transformación de datos distribuidos: i) Transacciones locales y distribuidas. ii) Transformaciones agregadas, filtradas y/o traducciones de metadatos. iii) Enrutamiento tradicional, enrutamiento basado en el contenido del mensaje y/o encabezados, y enrutamiento externo controlado y/o coordinado por un BPM (*Business Process Manager*).
- Orientado a eventos: aplicaciones y servicios son tratados como los puntos finales de servicios abstractos, que fácilmente pueden responder a eventos asíncronos. En consecuencia, las implementaciones no necesitan entender cómo el evento (mensaje) será dirigido al servicio destino.
- Flujo de procesos: definición de secuencias simples de pasos finitos para orquestar procesos de negocio sofisticados, controlados por metadatos o mediante el uso de BPEL4WS (*BPEL for Web Services*).
- Seguridad y confiabilidad: posibilidad de incorporar *firewalls* entre las conexiones de sus nodos.

A continuación se describe cada una de las capas de AITE, definidas para soportar las características ya estipuladas:

Red Telco: esta capa de la arquitectura representa todos los recursos de red del operador de telecomunicaciones, tales como: *softswitch*, servidores de medios, servidores de presencia, redes inteligentes legadas, etc.

JSLEE: la capa de JSLEE es utilizada para ejecutar de manera autónoma los diferentes servicios del operador de telecomunicaciones, cumpliendo con los requisitos propios de éste: baja latencia, alto rendimiento, alta disponibilidad, alto tráfico y orientación a eventos. El SLEE cuenta con dos contenedores: el primero, denominado contenedor de adaptación, encargado de almacenar los adaptadores de recursos con los cuales el SLEE se comunica con los recursos de red del operador de telecomunicaciones y con el ESB. El segundo, contenedor de lógica de aplicación cuya finalidad es almacenar todos los elementos u objetos que configuran la lógica de servicio.

ESB: la importancia clave de adoptar el enfoque SOA en ESB, es la clara separación entre la definición del servicio y el mecanismo para localizarlo e invocarlo. Así, lo crucial para la integración IT/Telco es definir administrativamente un flujo de procesos de negocio compuesto, conectando los servicios a través de itinerarios de mensajes. El itinerario representa un conjunto de operaciones de enrutamiento de mensajes discretos. En este sentido, ESB puede ser configurado en tres tipos de arquitectura: i) MBS (*Message Bus Architecture*), ii) *Protocol Switch*, y iii) *Gateway*. AITE usa una configuración MBS, principalmente por el enfoque orientado a servicios de las aplicaciones construidas tanto del lado IT y Telco y la naturaleza asíncrona y orientada a eventos de las aplicaciones de telecomunicaciones.

IT: esta capa contiene las diferentes aplicaciones almacenadas en servidores específicos como los BSS/OSS del operador, servidores de terceros o servidores de empresas que prestan servicios de *outsourcing*, mediante las cuales el operador subcontrata actividades comerciales u operativas. Por lo tanto, en esta capa están alojados los servidores de aplicación del mundo IT sobre tecnologías como .NET, COM, C#, C, C++,

integradas con el ESB haciendo uso de estándares como SOAP y JMS/MOM.

Ambiente de creación de servicios: es el entorno donde residen todos los recursos y herramientas que el desarrollador del servicio requiere. Específicamente, el entorno de desarrollo Eclipse (ECLIPSE, 2011) permite incorporar diferentes “*plug-in*” para facilitar la programación de las aplicaciones y servicios en este contexto.

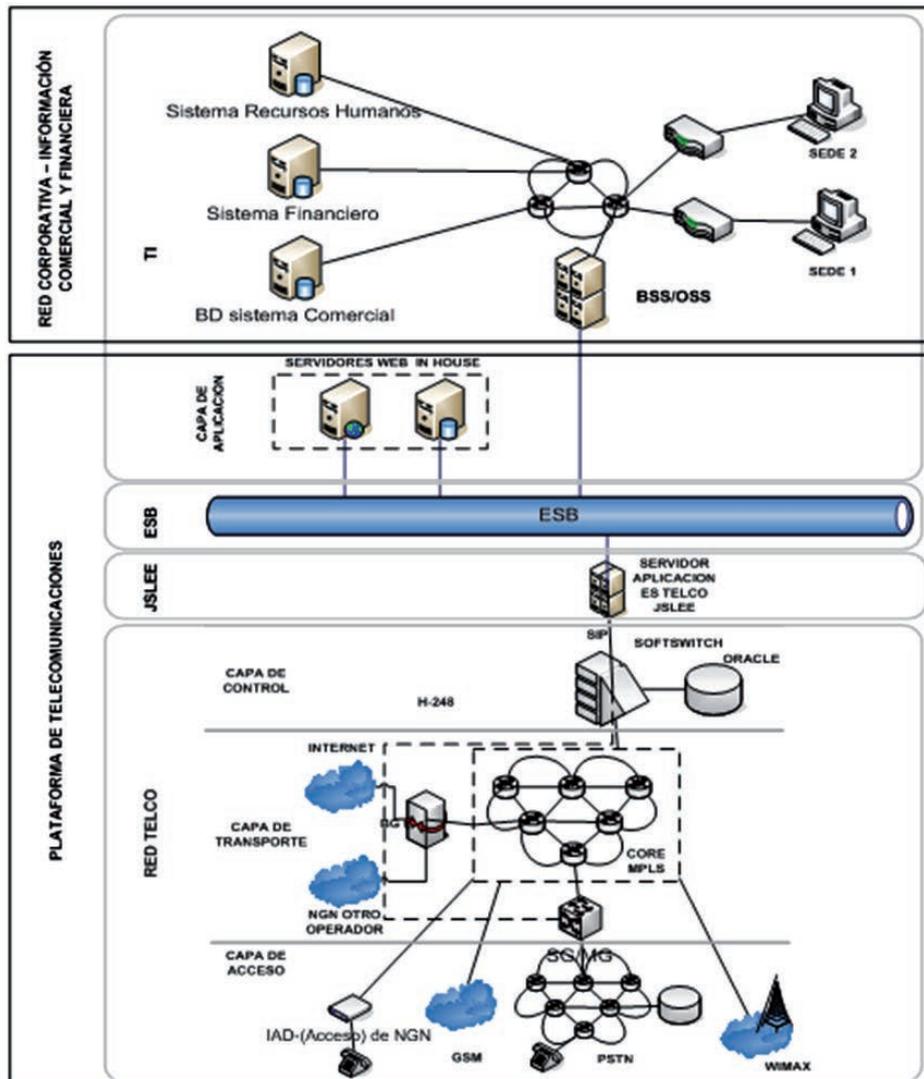
Caso de estudio

Inicialmente, se construyó un servicio de valor agregado denominado CRBT: (*Color Ring Back Tone*), el cual fue desplegado en un servidor *Rhino SLEE*; la evaluación del mismo se encuentra descrita completamente en (Rojas et al., 2011). Para el caso de AITE, se evalúa la comercialización y personalización del CRBT. La figura 2 muestra la implementación de AITE en la NGN y los sistemas IT de EMCALI para el caso de estudio propuesto.

Definición del servicio CRBT: es un servicio de valor agregado de telecomunicaciones que basa su funcionamiento en las siguientes premisas: cuando un número X llama al número B (suscriptor del servicio), el número X escucha una canción mientras el número B contesta; una vez el número B contesta la llamada, la canción deja de sonar y la comunicación entre el número X y el número B es establecida. La canción puede ser personalizada por el usuario B.

Comercialización y personalización del CRBT: la adquisición y personalización del servicio CRBT se realiza a través de la aplicación de comercialización almacenada en un servidor JEE de la empresa *outsourcing* contratada para tal fin. Una vez es aceptada la transacción de compra o personalización, el sistema de la empresa *outsourcing* almacena los datos y envía un mensaje al ESB, el cual a su vez direcciona dos mensajes: uno para actualizar el OSS/BSS del Operador de Telecomunicaciones con información relacionada al usuario y posterior generación de la cuenta de cobro; y otro enviado al servidor *Rhino SLEE* para la creación del perfil necesario en la ejecución del servicio CRBT. Cuando el usuario desee cambiar la personalización del servicio CRBT, se realiza el mismo procedimiento anterior.

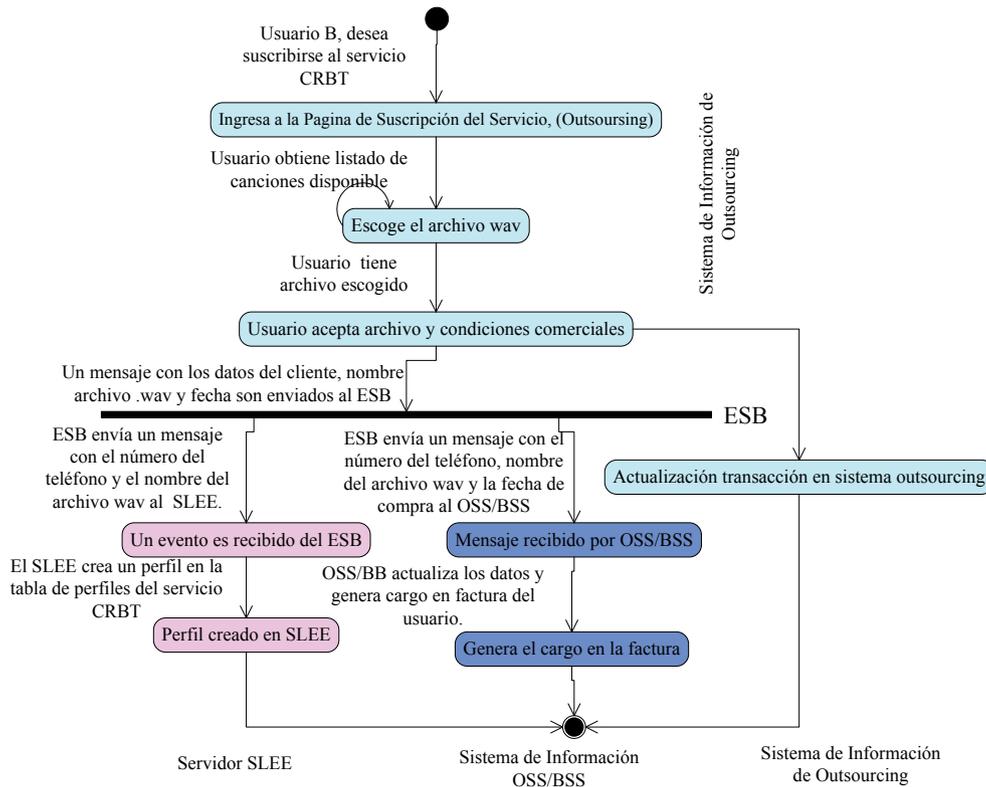
Figura 2. Implementación de AITE en la NGN de EMCALI



A continuación se describe el proceso de compra y venta del servicio CRBT (ver figura 3):

1. El usuario B, cuenta con el servicio de telefonía y desea obtener el servicio CRBT.
2. El usuario B ingresa a la página del agente comercial, empresa *outsourcing* contratada por el operador de telecomunicaciones.
3. El usuario B obtiene un listado de las canciones disponibles.
4. El usuario B escoge la canción de su preferencia, revisa las condiciones comerciales y si está de acuerdo acepta la transacción.
5. Un mensaje con los datos de teléfono, fecha de transacción, nombre de archivo media (.wav) es enviado al ESB.
6. El ESB realiza dos tareas:
 - a. Envía el número del teléfono y nombre del archivo media (.wav) al servidor SLEE, en el cual un servicio de personalización está corriendo; el SLEE crea el registro en la tabla de perfiles definida para el funcionamiento del servicio CRBT.
 - b. El ESB envía el número del teléfono, nombre de archivo y fecha de compra al sistema OSS/BSS de la empresa de telecomunicaciones, la cual a su vez genera un cargo que será facturado.
7. El sistema *outsourcing* almacena los datos de la transacción que utilizará para el respectivo cobro de la comisión de venta a la empresa de telecomunicaciones.

Figura 3. Diagrama de comercialización del servicio CRBT



La arquitectura presentada en la figura 4 es una instanciación de AITE y soporta el caso de estudio, a través de los siguientes componentes:

Red Telco: esta capa está constituida por una NGN con *softswitch* en la capa de control, servidores de medios *Mobicents* y *Convedia* de la Empresa EMCALI telecomunicaciones. Igualmente, para el uso de recursos de red en el caso de estudio se utilizaron: i) Adaptador de recursos SIP de *Open Cloud* (OCSIP) con el fin de comunicar el servidor de aplicaciones *Rhino SLEE* y el *Softswitch*. ii) Adaptador de recursos MGCP de *Mobicents* encargado de comunicar el servidor *Rhino SLEE* y el servidor de medios.

- El *softswitch* utilizado fue el ZXSS10 y proporciona los siguientes servicios (ZTE, 2007): i) control de conexión con una pasarela de medios y/o terminales IP, ii) selección de procesos que se pueden aplicar a una llamada, iii) enrutamiento para una llamada dentro de la red basada en la señalización y la información de base de datos de clientes, iv) transferencia del control de la llamada

a otro elemento de la red, y v) funciones de apoyo a la gestión, como el aprovisionamiento, fallos y facturación.

- El almacenamiento y ejecución de los archivos de audio requeridos para el aprovisionamiento del CRBT fue implementado en un servidor *Mobicents mms-standalone-2.0.0.GA*.

JSLEE: para la implementación de esta capa se utilizó el servidor *Rhino SLEE* versión 2.1 de la empresa *Open Cloud*, definiendo en su interior dos contenedores: lógica de aplicación y de adaptación.

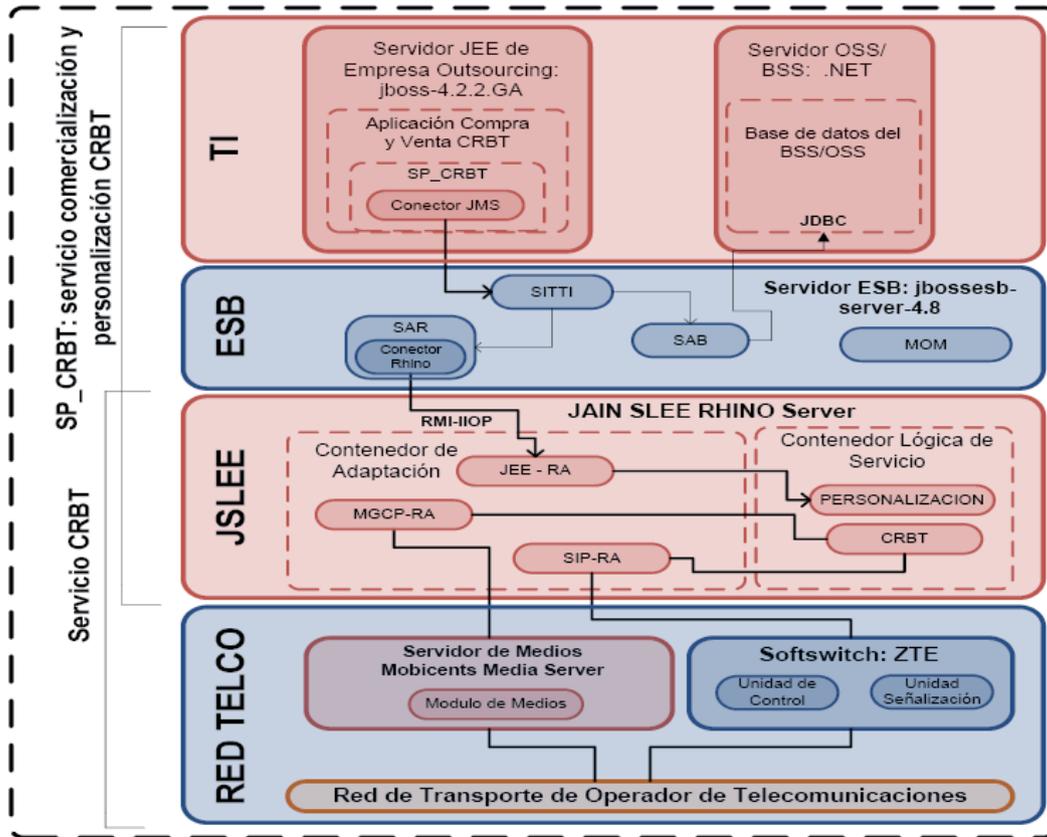
Contenedor de Lógica de Aplicación

- **CRBT:** es el servicio de prueba utilizado en el caso de estudio, construido a partir de dos SBB: i) *SoftswitchUserAgent* que realiza la comunicación con el *softswitch* a través del protocolo SIP. ii) *ColorringBackToneSBB*, encargado de enviar mensajes MGCP al servidor de medios, con el fin de dar instrucciones de apertura o cierre de conexiones, o inicio/detención de la reproducción de archivos de audio.

- **Personalización:** este servicio es el encargado de recibir los parámetros del servidor ESB mediante

el conector RMI-IIOP y actualizar la tabla de perfiles del servicio CRBT.

Figura 4. AITE Caso de estudio



Contenedor de adaptación

- **RA-SIP:** adaptador SIP de la empresa *Open Cloud* (OCSIP) que permite la comunicación entre el servidor *Rhino SLEE* y el *Softswitch ZXSS10*.
- **RA-MGCP:** adaptador MGCP *open source* de Mobicents (MGCPRA) encargado de la comunicación entre el servidor *Rhino SLEE* y el servidor de medios *Mobicents mms-standalone-2.0.0.GA*.
- **RA-RMI:** adaptador de Integración JEE de la empresa *Open Cloud* (*rhino-j2ee-connector*) que facilita la comunicación entre el servidor *Rhino SLEE* y el servidor *jbossesb-server-4.8*.

ESB: esta capa se implementó con *jbossesb-server-4.8*, donde se ejecutan los servicios de integración para los componentes de IT y *Rhino SLEE*.

- El conector *Rhino* facilita la integración entre el servidor *jbossesb-server-4.8* y el servidor *Rhino*

SLEE; a través de este conector un mensaje es enviado desde el ESB hacia *Rhino*, en el cual, el servicio de personalización está escuchando los mensajes provenientes del ESB. El servicio de personalización se encarga de actualizar la tabla de perfiles del servicio CRBT con los parámetros recibidos en el mensaje.

- El conector JMS, permite que los mensajes de la aplicación de compra del servicio CRBT, ubicada en el servidor *jboss-4.2.2.GA*, lleguen al *jbossesb-server-4.8*, para que a su vez, el servicio ubicado en el ESB, dirija los mensajes al servidor *Rhino* y al servidor OSS/BSS.
- El servicio MOM, recibe los mensajes provenientes de la aplicación de compra del servicio CRBT, ubicada en el servidor *jboss-4.2.2.GA*, y a su vez envía dos mensajes: el primero, a través de un conector JMS hacia el servidor OSS/BSS encargado de facturar el servicio; y el segundo, hacia el servidor *Rhino SLEE* a través del conector *rhino-j2ee-connector*.

IT: en la capa de servidores de IT se utilizó:

- Un servidor JEE jboss-4.2.2.GA para ejecutar la aplicación de comercialización o personalización del servicio CRBT (perteneciente a la empresa *outsourcing*).
- Un servidor de BSS/OSS, encargado de administrar la información técnica, comercial y operativa del operador de telecomunicaciones.

Pruebas y resultados

Además de las pruebas funcionales del servicio de comercialización y personalización del CRBT, se tomaron medidas de latencia con respecto al tiempo que tarda en llegar el mensaje de comercialización o personalización al *Rhino-SLEE*. Estas pruebas se realizaron bajo el contexto IT dadas las características del mismo. La infraestructura utilizada para las pruebas consta del siguiente hardware y software: Servidor *RHINO SLEE 2.1* de la empresa *Open Cloud*: Desplegado en un PC HP Intel® Core 2 Duo 2 GHz, 2 GB RAM, OS *Debian lenny 5*, Máquina virtual de java 1.6.18 con 1.5 GB de *heap memory*. Servidor ESB *jbossesb-server-4.8*: Desplegado en un laptop Gateway® M-1631U AMD Turion X2 TL-60 2.0 GHz, 4 GB RAM, OS *Ubuntu 10.04*, Máquina virtual de Java 1.6.0_18. Servidor *Outsourcing jboss-5.0.1.GA*: Desplegado en un Laptop HP® Pavilion dv4 Intel® Core 2 Duo 2.13 GHz centrino® 2, 4 GB RAM, OS *Windows Vista™ Business*, Máquina virtual de Java 1.6.0_18.

La figura 5 muestra los resultados de la prueba. Como se puede observar, el tiempo del envío del mensaje de la aplicación de personalización y la actualización del perfil en el servidor *Rhino SLEE* es de 700 milisegundos para 100 mensajes por segundo, cifra aceptable dentro los tiempos de respuesta definidos por (Dyson et al., 2004), quienes consideran que respuestas entre 3 a 20 segundos es adecuada para un Sistema Web. Este retardo también se ubica dentro del rango aceptable definido por (Nielsen, 2000) quien considera como aceptable un retardo menor a un segundo.

Finalmente, es importante recalcar que AITE no interfiere en el funcionamiento del servicio CRBT, dado que cuando se ejecuta el servicio CRBT, el servidor *Rhino SLEE* recibe y entrega mensajes

SIP hacia el *softswitch ZXSS10*, y envía y recibe mensajes MGCP al servidor de medios *Mobicents mms-standalone-2.0.0.GA*. Estas pruebas se realizan bajo el contexto Telco, dado que es allí donde se ejecuta el servicio CRBT.

La figura 6 muestra el tiempo de respuesta del servicio del servicio CRBT en el servidor *Rhino SLEE*. Como se puede observar, el tiempo de respuesta del servicio en el servidor *Rhino SLEE* es de 20 milisegundos para 50 llamadas por segundo, lo que demuestra que el servidor *jbossesb-server-4.8* (ESB) no interfiere en el desempeño del servicio como tal.

Figura 5. Tiempo de respuesta del servicio comercialización y personalización del CRBT (contexto IT)

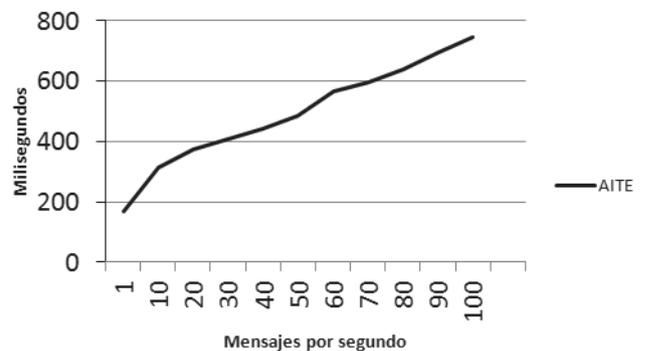
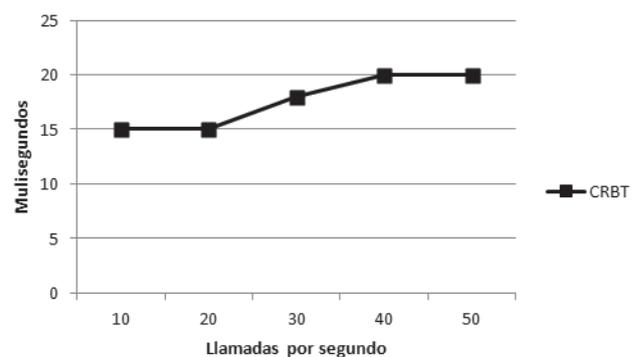


Figura 6. Tiempo de respuesta del servicio CRBT (contexto Telco)



Conclusiones

- AITE facilita la integración con diferentes sistemas de información gracias a sus funcionalidades como: conversión de protocolo de transporte, enrutamiento inteligente, transformación de mensaje y transformación de datos.
- Las empresas de telecomunicaciones que cuentan con plataforma NGN, tienen grandes oportunidades

- de negocio en un mercado poco explotado como el colombiano, y la utilización de tecnologías basadas en estándares con protocolos abiertos permiten explotar con rapidez los atributos de una NGN y no quedar amarrados a las soluciones propietarias.
- Contar con una NGN no es suficiente para desarrollar y desplegar servicios de valor agregado, hace falta definir cuál es el objetivo estratégico empresarial: i) ser una empresa que suministre servicios tradicionales, como la voz, datos y acceso a internet o ii) trabajar en conjunto con terceros para agregar valor a los servicios de los clientes. El segundo punto no es la entrega de su objeto, sino la creación de sinergias a través de tecnologías y arquitecturas con débil acoplamiento, reúso de componentes y definición de nuevos modelos de negocio.
 - La propuesta de arquitectura como AITE, bajo el concepto SOA, coloca a las empresas de telecomunicaciones a analizar juiciosamente los procesos de negocio orientados a mejorar los servicios hacia el cliente e incrementar los ingresos.
 - La modularidad de la arquitectura brinda un rápido desarrollo, lo cual redundará en menores tiempos de salida al mercado y menores costos de implementación.
 - AITE plantea una interoperabilidad altamente distribuida y de bajo acoplamiento respetando la funcionalidad de cada componente tanto de IT como de Telecomunicaciones, lo cual permite que los servicios de telecomunicaciones sean explotados más fácilmente.
 - La rápida y sencilla incorporación de nuevos sistemas de información en la arquitectura AITE, facilita las alianzas estratégicas en los nuevos modelos de negocio entre operadores de telecomunicaciones y/o negocios con terceras partes.
 - AITE respeta la independencia tanto de los servidores de IT como los de telecomunicaciones, adicionando un retardo en el envío del mensaje que no afecta la calidad del servicio VAS de telecomunicaciones.

Referencias

- Booth, D. (2007). *Web Services Description Language (WSDL) 2.0*. Recuperado el 27 de febrero de 2011 de <http://www.w3.org/TR/2007/REC-wsdl20-primer-20070626/>
- Cadenas, A., Sánchez, A., Carro, B. (2010). Deployment of contextual corporate telco services base on protocol adaptation in the NGN environment. *Communications Magazine IEEE*, (8)4, 34-40.
- Chapell, David. (2004). *Enterprise Service Bus*. USA: O'Reilly.
- Chen, R., Shen, V., Wrobel, T., Lin, C. (2008). Applying SOA and Web 2.0 to Telecom: legacy and IMS next-generation architectures. *International Conference on e-Business Engineering*, Xian, China.
- Chrighton, C., Long, D.T., Page, D.C. (2007). JAIN SLEE vs SIP Servlet which is the best choice for an IMS application server. *Telecommunications Networks and Applications Conference*. Australasian.
- Dyson, P., Longshaw, A. (2004). *Architecting Enterprise Solutions: Patterns for High-Capability Internet-based Systems*. Chichester, UK: John Wiley & Sons.
- Eclipse. (2011). Recuperado el 3 de marzo de 2011 de <http://www.eclipse.org/>.
- Ferenc, T., Ryan, A., Cherki, S., Muller, P. A., Jakobson, S. (2006). *Service oriented architectures for convergent service delivery platforms*. Recuperado el 14 de febrero de 2012 de <http://archive.eurescom.eu/~pub/deliverables/documents/P1600-series/P1652/D1/P1652-D1.pdf>
- Graham, F. (2009). *Working with third party services: An action plan for network providers*. Recuperado el 10 de febrero de 2010 de <http://www.heavyreading.com>
- Hanhua, L., Zheng, Y., Sun, Y. (2008). The next generation SDP architecture: based on SOA and integrated with IMS. *Second International Symposium on Intelligent Information Technology Application*, Shanghai, China.
- Imen, G., Bertin, E., Descrevel, J.P., Crespi, N. (2006). Service definition for next generation networks. *International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL'06)*, Morne, Mauritius.
- Johnston, A., Gabrielson, J., Christopoulos, C., Huysmans, M., Olsson, U. (2007). *Evolution of service delivery platforms*. Recuperado el 12 de febrero de 2010 de http://www.ericsson.com/res/thecompany/docs/publications/ericsson_review/2007/4_sdp_web.pdf
- Jieming, W., Tao, X. (2010). Research of Enterprise Application Integration Based-on ESB. *Second International Conference on Advanced Computer Control*, Beijing, China.

- Krivinska. N., Strauss. C., Auer. L., Zinterhof. P. (2008). Conceptual Framework for Services Creation/Development Environment in Telecom Domain. *10th International Conference on Information Integration and Web-based Applications & Services (iiWAS2008)*. Linz, Australia.
- Nielsen. J. (2000). *Designing Web Usability: The Practice of Simplicity*. New Riders Publishing, Indianapolis.
- Rojas. G., Estrada. F., Caicedo. J., Caicedo. O. M. (2011). Technical Criteria for Value-Added Services Creation, Execution and Deployment, on Next Generation Network. *The Seventh Advanced International Conference on Telecommunications*. St Maarten, The Netherlands Antilles.
- Unmehopa. M., Vemuri. K., Bennett. A. (2006). *Parlay/OSA: From Standards to Reality*. Chichester, UK: John Willey & Sons.
- ZTE. (2007). ZXSS10 SS1b Softswitch control equipment technical manual.

Sobre los autores

Gerardo Rojas Sierra

Ingeniero electricista. Especialista en Sistemas Gerenciales, Especialista en Telemática. Estudiante de Maestría de la Facultad de Ingeniería Electrónica y Telecomunicaciones. Universidad del Cauca, Grupo de Ingeniería Telemática, Cauca, Colombia. Profesional I Empresas Municipales de Cali EMCALI EICE ESP.
gerardorojas@unicauca.edu.co

Francisco Martínez

Ingeniero en Electrónica y Telecomunicaciones. Magíster en Ingeniería, área Telemática. Docente del Departamento de Telemática de la Facultad de Ingeniería Electrónica y Telecomunicaciones de la Universidad del Cauca. Integrante del grupo de Ingeniería Telemática, Grupo de Interés en Desarrollo de aplicaciones móviles e inalámbricas W@PColombia y Grupo de Ingeniería Telemática. Cauca, Colombia.
fomarti@unicauca.edu.co

Oscar Mauricio Caicedo R.

Ingeniero en Electrónica y Telecomunicaciones. Especialista en Redes y Servicios Telemáticos. Magíster en Ingeniería, área Telemática. Docente del Departamento de Telemática de la Facultad de Ingeniería Electrónica y Telecomunicaciones de la Universidad del Cauca. Integrantes del Grupo de Ingeniería Telemática, Grupo de Interés en Desarrollo de aplicaciones móviles e inalámbricas W@PColombia, Grupo de Ingeniería Telemática, Cauca, Colombia.
omcaicedo@unicauca.edu.co

Laura González

Ingeniera en Computación. Docente del Instituto de Computación de la Facultad de Ingeniería de la Universidad de la República, Montevideo, Uruguay. Integrante del grupo de Laboratorio de Integración (LIInS) de dicho instituto.
lauragon@fing.edu.uy

Los puntos de vista expresados en este artículo no reflejan necesariamente la opinión de la Asociación Colombiana de Facultades de Ingeniería.