

UNIVERSIDAD NACIONAL DE INGENIERÍA

FACULTAD DE INGENIERÍA INDUSTRIAL Y DE SISTEMAS



**“INTEGRACIÓN DE APLICACIONES EN UNA EMPRESA DE
TELECOMUNICACIONES”**

INFORME DE SUFICIENCIA

**PARA OBTAR EL TÍTULO PROFESIONAL DE:
INGENIERO DE SISTEMAS**

SOTOMAYOR INGA, WALTER LUIS

LIMA – PERU

2013

DEDICATORIA:

A mis padres: Por sus consejos, por el ejemplo de perseverancia y la motivación constante que me han permitido salir adelante.

A mis hermanos: Por estar conmigo y apoyarme siempre.

A Yeiny: Por su comprensión y por ayudarme a ser mejor persona cada día.

INDICE

DESCRIPTORES TEMÁTICOS	1
RESUMEN EJECUTIVO	2
INTRODUCCIÓN	4
CAPÍTULO I	6
PENSAMIENTO ESTRATÉGICO	6
1.1. DIAGNÓSTICO FUNCIONAL	6
1.1.1. <i>Organización</i>	6
1.1.2. <i>Productos y Servicios</i>	8
1.1.3. <i>Proveedores</i>	10
1.1.4. <i>Clientes</i>	10
1.2. DIAGNÓSTICO ESTRATÉGICO	11
1.2.1. <i>Misión</i>	11
1.2.2. <i>Visión</i>	11
1.2.3. <i>Valores</i>	11
1.2.4. <i>Análisis Externo</i>	12
1.2.5. <i>Análisis Interno</i>	13
1.2.6. <i>Objetivos</i>	13
CAPÍTULO II	14
MARCO TEÓRICO Y METODOLÓGICO	14
2.1. ARQUITECTURA EMPRESARIAL	14
2.2. ARQUITECTURAS DE INTEGRACIÓN.....	16
2.2.1. <i>Punto a punto</i>	16

2.2.2. <i>Arquitectura Hub y Spoke</i>	17
2.2.3. <i>Arquitectura integrada por Bus de Servicios</i>	19
2.3. ARQUITECTURA ORIENTADA A SERVICIOS	20
2.3.1. <i>Capas en una Arquitectura Orientada a Servicios</i>	22
CAPÍTULO III	31
PROCESO DE TOMA DE DECISIONES	31
3.1. SITUACIÓN ACTUAL	31
3.2. IDENTIFICACIÓN DEL PROBLEMA	35
3.3. PLANTEAMIENTO DE ALTERNATIVAS DE SOLUCIÓN	36
3.3.1. <i>Alternativa 1: Integración punto a punto</i>	36
3.3.2. <i>Alternativa 2: Integración por bus de servicios</i>	38
3.4. CRITERIOS DE SELECCIÓN	39
3.4.1. <i>Criterio 1: Costo de la implementación</i>	40
3.4.2. <i>Criterio 2: Tiempo de respuesta</i>	41
3.4.3. <i>Criterio 3: Productividad</i>	42
3.4.4. <i>Criterio 4: Interoperabilidad</i>	45
3.4.5. <i>Criterio 5: Reutilización</i>	46
3.4.6. <i>Criterio 6: Costo de mantenimiento</i>	47
3.4.7. <i>Criterio 7: Know How del equipo de Sistemas</i>	49
3.5. SELECCIÓN DE UNA ALTERNATIVA DE SOLUCIÓN	49
3.5.1. <i>Matriz Multicriterio</i>	50
3.6. PLAN DE ACCIÓN	51
3.6.1. <i>Requerimiento Funcional de ejemplo</i>	52
3.6.2. <i>Caso de Uso</i>	53
3.6.3. <i>Modelo de Análisis</i>	59
3.6.4. <i>Diseño del Servicio</i>	60
3.6.5. <i>Diagrama de Componentes</i>	61
CAPÍTULO IV	62
ANÁLISIS DE RESULTADOS	62
4.1. COSTO DE PRODUCCIÓN	62

4.2. COSTO DE MANTENIMIENTO	64
CONCLUSIONES Y RECOMENDACIONES	67
CONCLUSIONES	67
RECOMENDACIONES	69
GLOSARIO DE TÉRMINOS	69
BIBLIOGRAFÍA	71
ANEXO 1 - IMPLEMENTACIÓN DEL SERVICIO WEB	72
ANEXO 2 - EJECUCIÓN DE UN SERVICIO WEB	82

ÍNDICE DE FIGURAS

Figura 1. Organigrama Telecom.....	7
Figura 2. Relación entre Arquitecturas.....	15
Figura 3. Arquitectura Punto a Punto.....	16
Figura 4. Integración en una Arquitectura Punto a Punto.....	17
Figura 5. Arquitectura Hub y Spoke.....	18
Figura 6. Arquitectura por Bus de Servicios.....	20
Figura 7. Capas de una Arquitectura Orientada a Servicios.....	23
Figura 8. Capa de Componentes de Servicios.....	25
Figura 9. Capa de Servicios.....	26
Figura 10. Capa de Negocio.....	27
Figura 11. Capa de Integración.....	28
Figura 12. Diagrama Conceptual de aplicaciones de la empresa Telecom..	33
Figura 13. Integración punto a punto en Telecom.....	34
Figura 14. Definición del problema.....	35
Figura 15. Alternativa 1: Integración punto a punto.....	37
Figura 16. Alternativa 2: Integración por bus de servicios.....	38
Figura 17. Capas en una Integración Punto a Punto.....	411
Figura 18. Capas en una integración por Bus de Servicios.....	4242
Figura 19. Adaptadores en una integración Punto a Punto.....	433
Figura 20. Adaptador en una integración por Bus de Servicios.....	44
Figura 21. Enterprise Service Bus.....	46
Figura 22. Diagrama de Casos de Uso.....	53
Figura 23. Modelo de Análisis.....	59

Figura 24. Diagrama de Secuencia.....	60
Figura 25. Diagrama de Clases.....	60
Figura 26. Diagrama de Componentes	61
Figura 27. Proyecto en Eclipse.....	72
Figura 28. Proyecto OSB en Eclipse.....	80
Figura 29. Transformación en OSB.....	80
Figura 30. Servicio Web desplegado 1.....	82
Figura 31. Servicio Web desplegado 2.....	82
Figura 32. Prueba del Servicio Web 1.....	833
Figura 33. Prueba del Servicio Web 2.....	833

ÍNDICE DE CUADROS

Tabla 1. Número de Interfaces.....	400
Tabla 2. Comparación de costos de alternativas.....	400
Tabla 3. Tabla de Know How.	48
Tabla 4. Resultados de la Matriz de Multicriterio.....	500
Tabla 5. Resultado Costo de Producción.	633
Tabla 6. Resultado Costo de Mantenimiento.	65
Tabla 7. Resultado Final.	65

DESCRIPTORES TEMÁTICOS

Arquitectura orientada a servicios (SOA).

Enterprise Service Bus.

Conexión Punto a Punto.

Arquitectura Empresarial.

Integración de aplicaciones.

Servicios Web.

Interfaz de Programación de Aplicaciones.

RESUMEN EJECUTIVO

Telecom es una empresa que brinda servicios de telecomunicaciones y sustenta su posicionamiento en la rapidez de su red de negocios y en su servicio de clase mundial.

Pensando en la mejor manera de colaborar con el éxito de sus clientes es que Telecom decide implementar un nuevo portal empresarial (Web Portal). La visión del portal empresarial, a largo plazo, es contar con una plataforma única que les brinde soporte a los clientes buscando darles una experiencia única y relevante.

La iniciativa de contar con un nuevo portal empresarial está enmarcada en el lanzamiento de la banda 3G por parte de Telecom. Este lanzamiento es una decisión estratégica que ha impulsado a replantear aspectos relevantes de su operación, entre los cuales se encuentran las herramientas de tecnología que le dan soporte.

Es en este contexto que surge el problema de integración de este nuevo portal empresarial con el resto de aplicaciones de Telecom, que ya vienen funcionando y que dan soporte al día a día de la operación de la empresa.

La arquitectura empresarial que maneja Telecom es anticuada y se basa en conexiones punto a punto entre aplicaciones, lo que hace muy difícil el mantenimiento de las mismas. La ventaja principal de este enfoque es el

rendimiento, este es un punto importante pues como se mencionó en líneas arriba la rapidez es uno de los pilares de la estrategia de Telecom.

Por otra parte tenemos un nuevo paradigma de arquitectura empresarial, la denominada Arquitectura Orientada a Servicios (SOA), este enfoque es muy adecuado para integrar aplicaciones empresariales pues es muy flexible al cambio y muy fácil de mantener.

La solución a este problema se detalla en los siguientes capítulos de este informe y consiste en la adopción del nuevo enfoque orientado a servicios, no solo por parte del área de sistemas sino por toda la empresa ya que esta implementación no se podría realizar sin el apoyo de la alta gerencia de la empresa.

INTRODUCCIÓN

Las demandas empresariales siguen cambiando a un ritmo frenético. Para mantener el ritmo, la organización de TI, la infraestructura y los procesos deben evolucionar rápidamente y adaptarse a esas demandas cambiantes del negocio.

Una arquitectura orientada a servicios (SOA) ayuda a las organizaciones de TI a ser más ágiles y sensibles. SOA es un conjunto de principios y metodologías para el diseño y desarrollo de software en forma de servicios interoperables. Estos servicios tienen bien definidas las funcionalidades de negocio que se construyen como componentes de software (piezas discretas de código y / o estructuras de datos) que pueden ser reutilizados para diferentes propósitos.

La naturaleza de las aplicaciones está cambiando. Tanto los proveedores de paquetes de aplicaciones, proveedores de herramientas, proveedores de software, proveedores de servicios cloud, todos se están moviendo hacia SOA para una rápida innovación. Esto significa que ahora es el momento de adoptar SOA para tomar ventaja de esta innovación al tiempo que aumenta la agilidad general.

Entre los beneficios principales de adoptar SOA tenemos:

- Mejorar la interoperabilidad - Simplifica la integración y hacer más flexibles los sistemas de TI a través del uso de estándares basados en los servicios compartidos.
- Fomentar la reutilización - Reducir costos y aumentar la calidad de las aplicaciones de la reutilización de servicios compartidos en lugar de construir la funcionalidad redundante en aplicaciones
- Impulsar la inversión legado: Conservar y utilizar los sistemas de legado mediante la encapsulación de la lógica, la inversión y el capital intelectual de las aplicaciones.

En el presente trabajo vamos a comparar dos alternativas de solución a un problema común en las empresas, la integración de aplicaciones, y se va demostrar que una de las mejores alternativas que encontramos en el mercado es la Arquitectura Orienta a Servicios.

Finalmente se resalta la estructura del presente documento. El primer capítulo está dedicado al diagnóstico de la empresa, se describe su organización, productos, servicios, proveedores, clientes y se realiza un análisis estratégico. En el segundo capítulo se detalla el marco teórico que servirá para explicar la solución planteada. En el capítulo 3 se describe el proceso de toma de decisiones, primero se identifica el problema, luego se plantean alternativas de solución y al final se elige una alternativa. En el cuarto capítulo se analiza los resultados económicos obtenidos. Finalmente se detalla las conclusiones, recomendaciones y anexos.

CAPÍTULO I

PENSAMIENTO ESTRATÉGICO

1.1. DIAGNÓSTICO FUNCIONAL

1.1.1. Organización

Telecom del Perú S.A. es una empresa de telecomunicaciones subsidiaria de NII Holdings Inc., que inició sus operaciones en el Perú en el año 1998 como prestadores de servicios públicos de telecomunicaciones.

Es uno de los proveedores líderes a nivel mundial en servicios integrados de comunicación inalámbrica, diseñada para cubrir las necesidades de sus clientes.

A través del servicio de Conexión Directa Internacional actualmente brinda cobertura a diversos países del mundo como son México, Brasil, Argentina, Chile, Estados Unidos y Canadá, teniendo un aproximado de 3 440 000 suscriptores netos.

La tecnología IDEN desarrollada por Motorola, le da el valor diferencial que nos distingue en el mercado de las telecomunicaciones ya que nos permite integrar en un mismo equipo 4 tipos de servicio:

Conexión Directa Telecom.

Conexión directa internacional.
 Interconexión telefónica.
 Servicio de mensajería Telecom.
 Telecom Online.

El Organigrama es el siguiente:

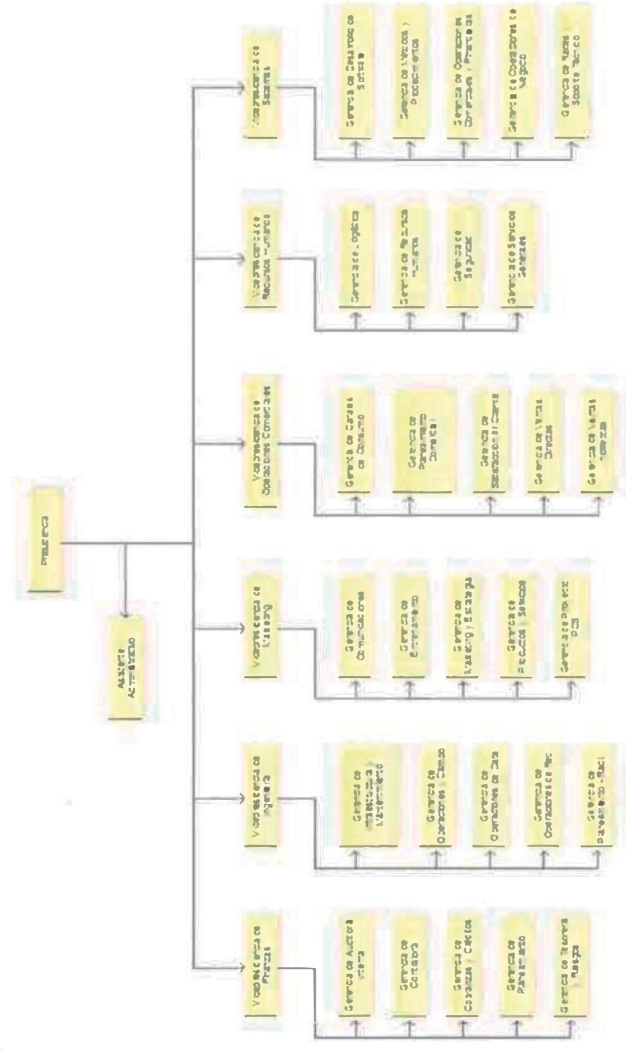


Figura 1. Organigrama Telecom.
 Fuente: Telecom.

- Finanzas: Facilita el crecimiento rentable de la organización mediante un servicio rápido y riguroso al cliente externo e interno.
- Ingeniería: Brinda a los clientes Telecom una red de alta calidad y disponibilidad.
- Marketing: Interpreta el plan de negocios de la empresa (ingresos, ventas, rentabilidad, fidelización de clientes, etc.) en términos de diseño de producto.
- Operaciones Comerciales: Se encarga de Brindar a los clientes soluciones de negocios acorde con sus necesidades que les permitan mejorar su productividad respaldado por un servicio de post venta de clase mundial.
- RRHH y Administración: Dota a Telecom de los recursos necesarios para garantizar su éxito. Estos pueden ser humanos, materiales o de servicios y deben tener la calidad y oportunidad debida.
- Sistemas: Provee soluciones tecnológicas, rentables, pro activas para soportar las necesidades del negocio, contribuyendo así con el éxito de la organización.

1.1.2. Productos y Servicios

Entre los principales productos de Telecom tenemos:

- Equipos: Telecom ofrece equipos de telefonía celular con tecnología iden (radio) y 3g.
- Internet: Modem USB, que permite la conexión a internet desde cualquier lugar donde exista cobertura Telecom.
- Chip: Módulo de seguridad insertado en un equipamiento móvil y que incluye datos utilizados para identificar al usuario y proporcionar seguridad para la transmisión de voz y datos.
- Accesorios para teléfonos móviles.

Entre los principales servicios tenemos:

- **Conexión Directa:** Este es un servicio que permite la comunicación directa entre dos dispositivos móviles solo presionando un botón, la tecnología que se usa para este fin es la denominada IDEN, que es una tecnología desarrollada por Motorola.
- **Telefonía móvil:** Servicio de interconexión telefónica a través de dispositivos móviles.
- **Servicios de valor agregado:** Entre los cuales tenemos el servicio de mensajería, Localizador Satelital, MODEM Inalámbrico, Bloqueo de Correo no Solicitado, Telecom Backup, iAlarm.
- **Servicio de banda ancha:** Ofrece acceso a Internet, Telefonía Fija y Enlace de Datos a una gran velocidad de transmisión, utilizando la tecnología WIMAX (Worldwide Interoperability for Microwave Access), una tecnología que permite transmitir información de manera inalámbrica y en forma eficiente.
- **Mensajería Telecom:** Servicio que permite el envío de mensajes cortos, pueden ser del tipo texto (SMS) o con algún contenido multimedia (MMS).
- **Soluciones de negocio:** Son un conjunto de aplicaciones que desarrolla Telecom orientadas a optimizar determinados procesos de negocios, buscando incrementar la productividad del trabajo, el ahorro de tiempo y un mejor control a nivel gerencial.

Algunas de estas soluciones son las siguientes:

- Gestión de ventas
- Gestión móvil de reparto
- Despacho aduanero
- Alerta plus

1.1.3. Proveedores

Telecom tiene proveedores principalmente de Tecnología, en resumen podemos agruparlos así:

Equipos:

- Motorola
- Alcatel – Lucent
- Nokia

Software y Soluciones de Negocio:

- Accenture
- EMC
- LHS
- BCTS CONSULTING
- Hewlett Packard

Auditoría:

- Deloitte

1.1.4. Clientes

Desde que Telecom inicio sus operaciones en el Perú, enfocó sus productos a grandes y medianas empresas, tratando de identificarse como una empresa que brinda soluciones a nivel empresarial. Entre los principales clientes tenemos:

- América Express S.A.
- Cementos Lima.
- GMD S.A.

- LAN PERU S.A.
- Municipalidad Metropolitana de Lima.
- Pesquera Diamante S.A.

Desde principios del 2009 Telecom impulsó una campaña publicitaria para cambiar el pensamiento que tiene la mayoría de personas, quienes asocian el radio Telecom con un dispositivo para trabajo, para el resto de actividades usan otro teléfono celular. Con esta campaña se busco que las personas decidan tener un radio Telecom y no necesiten comprarse otro dispositivo, la venta de equipos de 3ra generación es un paso importante para vender esta idea.

1.2. DIAGNÓSTICO ESTRATÉGICO

1.2.1. Misión

La misión de Telecom es hacer que sus clientes logren el éxito mediante soluciones de valor agregado en servicios de telecomunicación para grupos móviles de trabajo.

1.2.2. Visión

La visión de Telecom es ser líder y ser la empresa de mayor prestigio en servicios de telecomunicaciones, que satisfagan las necesidades de sus clientes con personas involucradas y comprometidas.

1.2.3. Valores

- Pasión por servir: Brindar una atención de primera y mantener una comunicación permanente con los clientes, utilizando un lenguaje sencillo y limpio.

- Pro actividad: Evaluar permanentemente los procesos existentes y necesidades futuras para una mejora continua.
- Flexibilidad: Tener la capacidad de adaptarse a las necesidades del cliente.
- Confiabilidad: Cumplir con las promesas que hacemos a nuestros clientes como base fundamental para competir. Prestar un servicio con exactitud y seriedad para generar confianza.
- Comunicación: Brindar una atención de primera y mantener una comunicación permanente con nuestros clientes, utilizando un lenguaje sencillo y limpio.
- Servicio de clase mundial: Empresa que compite con éxito en un ambiente global. Garantiza su diferenciación a través de ventajas competitivas, procesos y servicios innovadores.

1.2.4. Análisis Externo

Oportunidades

- En nuestro país y en muchos otros existe el fenómeno de la adquisición de equipos móviles.
- La etapa económica por la que atraviesa el país es muy buena para el negocio de las telecomunicaciones.
- La comunicación entre la gente es vital por ello el rubro de la telecomunicaciones logra ser una de las más rentables del mercado.

Amenazas

- La posibilidad de que exista un cuarto operador y posea un mejor servicio IDEN y 3G.

- La amenaza latente de una posible crisis, cual afecte las acciones de la empresa invertidas en la bolsa de valores.
- La implementación en nuestro país de la Portabilidad numérica por parte de OSIPTEL que promueve el posible cambio hacia otro operador del número móvil adquirido.

1.2.5. Análisis Interno

Fortalezas

- Telecom es la única empresa que en nuestro país ha implementado el uso de la tecnología IDEN, gracias a esta tecnología la gente puede estar interconectada con la facilidad que brinda el poder del botón.
- La implementación y el uso de la red 3G, innovando en un mejor costo en el uso del internet móvil.

Debilidades

- El alto costo de los equipos lo cual no los hace tan comerciales a comparación de sus competidores.
- La limitante de que la señal de Telecom abarca solo a los departamentos desde Tumbes hasta Tacna, Cuzco y una parte del centro del País.

1.2.6. Objetivos

- Reducción del Churn (Ratio que indica la cantidad de gente que se desvincula de la empresa).
- Captar a la contratación de los mejores profesionales del medio.
- Brindar un servicio simple, confiable y extraordinario a los clientes.
- Alcanzar los resultados del negocio según los objetivos planteados.

CAPÍTULO II

MARCO TEÓRICO Y METODOLÓGICO

2.1. ARQUITECTURA EMPRESARIAL

El mundo de los negocios es cada vez más complejo, dinámico y competitivo. En este entorno las empresas tienen la necesidad de contar con un soporte de las tecnologías de información que les permita sostener la operación diaria.

Los objetivos estratégicos, las personas, los procesos, las necesidades de información, las aplicaciones y la infraestructura de TI deben ser gestionados simultáneamente. Es en este contexto que nace la Arquitectura Empresarial.

Gartner define a la Arquitectura empresarial como “El proceso de trasladar la visión de negocio y la estrategia hacia un cambio empresarial creando, comunicando y mejorando los requerimientos claves, principios y modelos que describen el estado futuro de la empresa y permitiendo su evolución”.

El MIT Centro de investigación de Sistemas de Información (MIT CISR) tiene un concepto propio de lo que es una arquitectura empresarial: “Es la organización lógica de los procesos de negocio y la infraestructura de TI reflejando la integración y los requerimientos de estandarización del modelo de operación de la empresa. El modelo de operación es el estado deseado

de la integración y estandarización de los procesos de negocio para la entrega de productos y servicios a sus clientes”.

La arquitectura empresarial describe a la empresa como una estructura coherente, proporcionando un mapa de la empresa que ayuda a entender su estructura, como opera actualmente y como encontrar la manera de cumplir los objetivos estratégicos de la organización con un adecuada estructura de tecnologías de información.

En base al marco de referencia TOGAF una arquitectura empresarial debería contar con las siguientes arquitecturas:

- Arquitectura de negocio: Está alineada con la estrategia de la empresa representa los procesos de negocio y como estos se relacionan para satisfacer las necesidades de los clientes.
- Arquitectura de información: Se encarga del flujo de información de la empresa y como esta se procesa.
- Arquitectura de aplicación: Representa el conjunto de aplicaciones que procesan información y soportan los procesos de negocio en una empresa.
- Arquitectura tecnológica: Determina la tecnología a utilizar, el lenguaje de programación, la base de datos y otros componentes tecnológicos que se requieran.

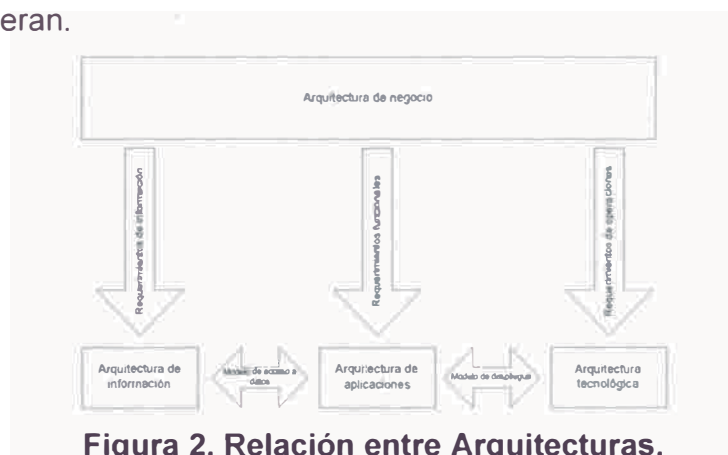


Figura 2. Relación entre Arquitecturas.

Fuente: Elaboración propia.

2.2. ARQUITECTURAS DE INTEGRACIÓN

Las arquitecturas de integración son el nivel más alto de abstracción en el proceso de llegar a una solución de integración empresarial. A continuación vamos a presentar las arquitecturas usadas hoy en día.

2.2.1. Punto a punto

El enfoque más simple para intercambiar mensajes de un sistema a otro es conectarlos directamente a través de una interfaz customizada, denominada comúnmente como API. Esto permite que dos sistemas se comuniquen usando un adaptador que se diseño y desarrollo para administrar transferencia de data entre ellas.

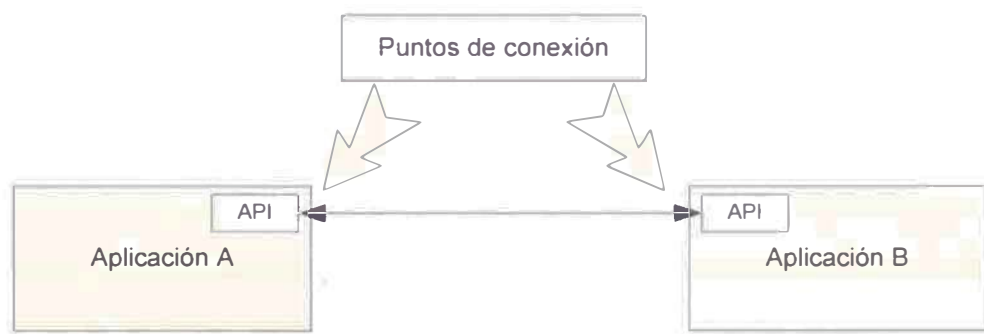


Figura 3. Arquitectura Punto a Punto.

Fuente: Elaboración propia.

En una arquitectura punto a punto la integración de sistemas requiere código customizado por cada interface. Cuando algún cambio ocurre en cualquiera de las aplicaciones, las interfaces (API) deben ser actualizadas y cambiadas.

Con solo dos aplicaciones, es relativamente simple mantener las interfaces de comunicación. Pero cuando más aplicaciones necesitan ser integradas más detalles necesitan ser mantenidos por cada conexión. En este

escenario el número de conexiones crece exponencialmente. La fórmula que describe esto es $n(n-1)/2$. La variable n es el número de aplicaciones que necesitan ser conectadas en una malla completa de conexiones punto a punto.

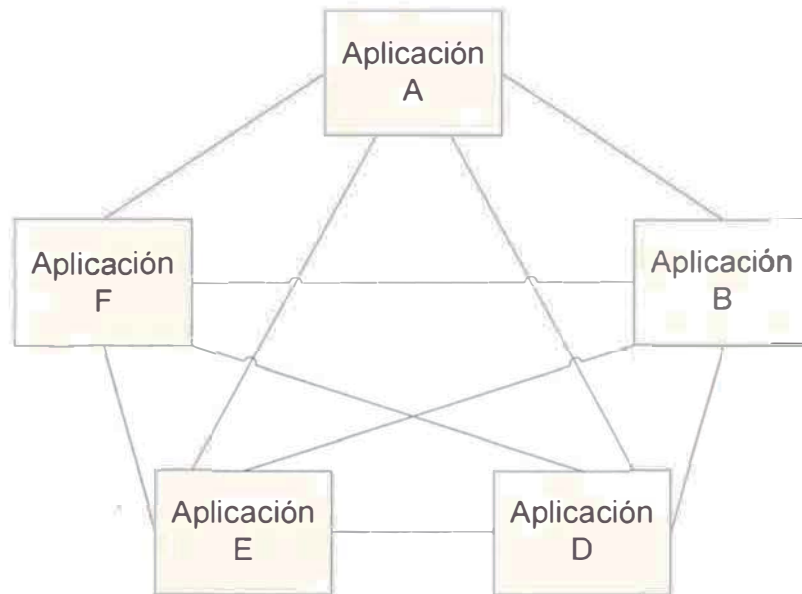


Figura 4. Integración en una Arquitectura Punto a Punto.

Fuente: Elaboración propia.

En la figura se puede apreciar el alto acoplamiento que tienen las aplicaciones. Un cambio en una interfaz afecta a la otra. Este enfoque puede llegar a ser muy complejo y difícil de mantener.

2.2.2. Arquitectura Hub y Spoke

En esta arquitectura un agente único centralizado referido como el "Hub" se ocupa del intercambio de mensajes, la transformación, y las rutas para los sistemas que deben integrarse. Estos sistemas son integrados al Hub a través de adaptadores llamados "Spokes". En la siguiente figura podemos apreciar esta arquitectura que se asemeja a una rueda con sus rayos.

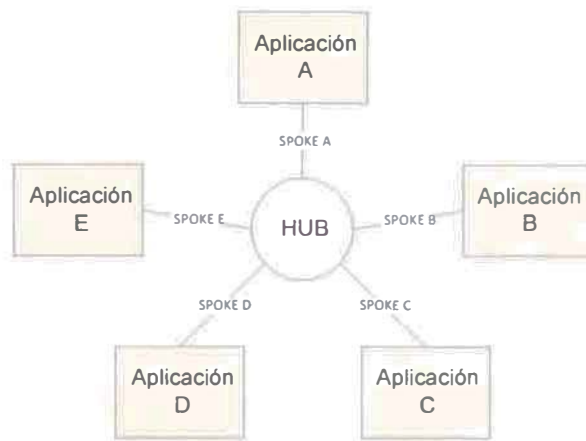


Figura 5. Arquitectura Hub y Spoke

Fuente: Elaboración propia.

Una típica integración por Hub tiene tantas conexiones como aplicaciones que requieren integrarse y contiene lo siguiente:

- Un modelo genérico de objetos de negocio, que lidia con conflictos, ambigüedades, y temas de compatibilidad entre aplicaciones.
- Un motor de transformación que mapea los objetos de negocio específicos de la aplicación a la forma genérica y viceversa durante el proceso de integración.
- Un motor de colaboración que ejecuta cualquier lógica de proceso de negocio que es parte del proceso de integración.

El uso de esta arquitectura es equivalente a un sistema de seres humanos que utiliza un corredor u otro tipo de intermediario. Digamos que estamos tratando de comprar una casa. Podríamos conducir por la ciudad en busca de señales colocadas en los jardines delanteros de las personas que indiquen su deseo de vender, luego nos pondríamos en contacto con cada propietario para preguntar los precios y condiciones. Cuando encontramos una casa que nos gusta, podríamos hacer la negociación directa con el vendedor y nos encargamos de todo nosotros. Lo que es una molestia y un peligro pues no sabemos todo sobre las leyes que tienen que ver con la

transferencia de los bienes inmuebles ni acerca de las búsquedas de título e informes de inspección.

En vez de esto, usamos un corredor, que hará todo este trabajo por nosotros. Este se encargara del inspector, los diversos organismos gubernamentales, los bancos, la gente de seguros y todo lo demás.

Tener este tipo de enfoque hace que la solución de arquitectura sea relativamente fácil de administrar.

2.2.3. Arquitectura integrada por Bus de Servicios

Un bus de servicio empresarial es una infraestructura que facilita la adopción de una arquitectura orientada a servicios. Provee de APIs los cuales pueden ser usados para desarrollar servicios y hacer que estos interactúen unos con otros. Técnicamente un ESB es un sistema de mensajería troncal el cual realiza conversiones de protocolo, transformación de formatos de mensaje, ruteo, acepta y envía mensajes de varios servicios y aplicaciones los cuales son enlazados al bus de servicios.

La grandeza de una solución de bus de servicio empresarial radica en que posibilita la comunicación entre aplicaciones sobre cualquier protocolo existente. Es una especie de pasarela encargada de traducir el lenguaje de una aplicación para que pueda ser leída por otra que la requiera.

Gracias al ESB los servicios no interactúan directamente sino a través de un conector.

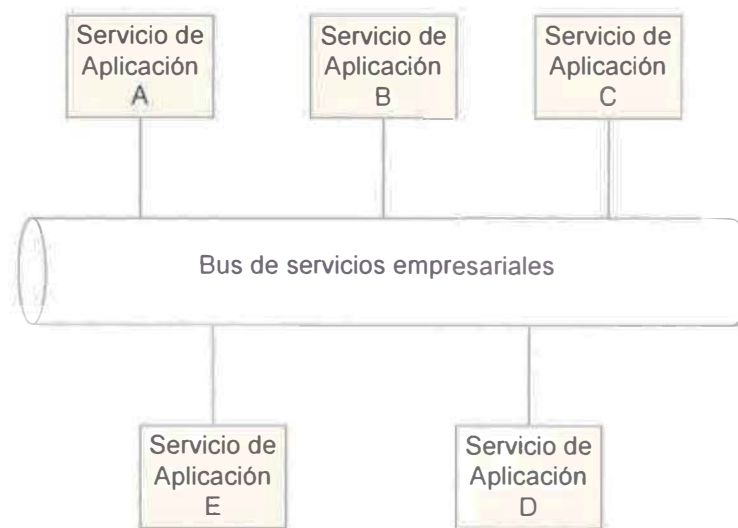


Figura 6. Arquitectura por Bus de Servicios

Fuente: Elaboración propia.

Cada aplicación expone un servicio, estos son expuestos en el bus para que sean consumidos por otras aplicaciones.

En resumen las funciones de un ESB son:

- Identificar los mensajes y las rutas entre los servicios
- Permitir el flujo de mensajes a través de diferentes protocolos de transporte (HTTP, FTP, SMTP)
- Transformar los formatos de los mensajes entre el solicitante y el servicio
- Proporcionar robustez y seguridad de las comunicaciones
- Proporcionar enrutamiento inteligente y ubicación independiente de la transformación

2.3. ARQUITECTURA ORIENTADA A SERVICIOS

En los últimos 25 años la arquitectura de software ha crecido como disciplina. Esto trae la necesidad de contar con patrones y estilos

arquitectónicos. Un estilo arquitectónico es la combinación de los rasgos distintivos en los cuales una arquitectura es realizada o expresada. Esto puede ser visualizado como un conjunto de componentes, conectores, composiciones y configuraciones. Sabemos que un sistema de software a menudo tiene un estilo arquitectónico predominante para su diseño. Se ha demostrado que la modularidad y el desacoplamiento facilitan la creación de sistemas complejos, tales como los requeridos por las aplicaciones de negocio actuales.

En años recientes, el desacoplamiento de la interface con la implementación se ha elevado de un nivel de programación a un nivel de arquitectura de bajo acoplamiento entre una interface de un servicio consumido por un cliente con su implementación por un proveedor de servicio. A este estilo de arquitectura se le conoce como Arquitectura Orientada a Servicios (SOA), donde en lugar de las implementaciones de componentes expuestos y conocidos, únicamente los servicios provistos se publican y los consumidores están aislados de los detalles de la implementación del proveedor.

Por lo tanto, SOA permite a las empresas y tecnologías de información converger a través de un acuerdo sobre un conjunto de servicio TI alineados a los negocios que soportan los procesos y objetivos de la organización. No solo proporcionan una funcionalidad flexible desacoplada que puede ser reusada, además proporciona los mecanismos para exteriorizar variaciones de calidad de servicio en especificaciones declarativas como WS-Policy y estándares relacionados.

Como un flexible y extensible framework de arquitectura, SOA tiene las siguientes características que lo definen:

- Reducción de costos: Proporcionando la oportunidad de consolidar la funcionalidad de aplicaciones redundantes y desacoplar funcionalidad de aplicaciones costosas y obsoletas mientras se aprovecha las inversiones existentes.
- Agilidad: Estructura las soluciones de negocio basadas en un conjunto de servicios de tal manera que se facilite la rápida reestructuración y reconfiguración de los procesos de negocio y las soluciones que los consumen.
- Incrementa la ventaja competitiva: Ofrece la oportunidad de entrar en nuevos mercados y aprovechar las capacidades de negocio existente en formas nuevas e innovadoras, utilizando un conjunto de servicios bajamente acoplados. Potencialmente aumenta la cuota del mercado y el valor del negocio al ofrecer nuevos y mejores servicios de negocio.
- Time to Market: Entregar soluciones alineadas al negocio permitiendo que el negocio pueda decidir sobre los factores claves de una solución y les permite soportar con rapidez y poner en práctica esta dirección.
- Consolidación: Reducir el número físico de los sistemas y permitir la consolidación de las plataformas en un programa de referencia SOA. Transición elegante de dependencia espagueti de aplicaciones legadas a un conjunto de sistemas más organizados e integrados.

2.3.1. Capas en una Arquitectura Orientada a Servicios

SOA tiene nueve capas que representan a nueve de los grupos claves de consideraciones y responsabilidades que surgen en el proceso de diseño y solución SOA. Además cada capa está diseñada para materializar y reforzar cada una de las diversas perspectivas del valor empresarial SOA.

Por cada capa es práctico postular 2 aspectos: lógico y físico. El aspecto lógico incluye todos los bloques de construcción arquitectónicos, las decisiones de diseño, opciones, etc; mientras que el aspecto físico de cada

capa incluye la realización de cada aspecto lógico con tecnología, normas y productos, que se determinarán teniendo en cuenta las diferentes decisiones arquitectónicas que son necesarias para hacer comprender y construir la arquitectura.

El siguiente diagrama muestra a la Arquitectura Orientada a Servicios como un conjunto de capas lógicas. Hay que tener en cuenta que una capa no solo depende de una capa inferior, por ejemplo, un consumidor puede acceder a la capa de procesos de negocio como un servicio o ir a la capa de servicios directamente. Por ejemplo una determinada solución de SOA puede excluir a la capa de procesos de negocio y la capa de consumidores tendría que interactuar directamente con el nivel de servicio. Esta solución no se vería beneficiada con la propuesta de valor asociado a la capa de procesos de negocio. En este sentido, esta arquitectura de referencia representa una parcial arquitectura de capas.

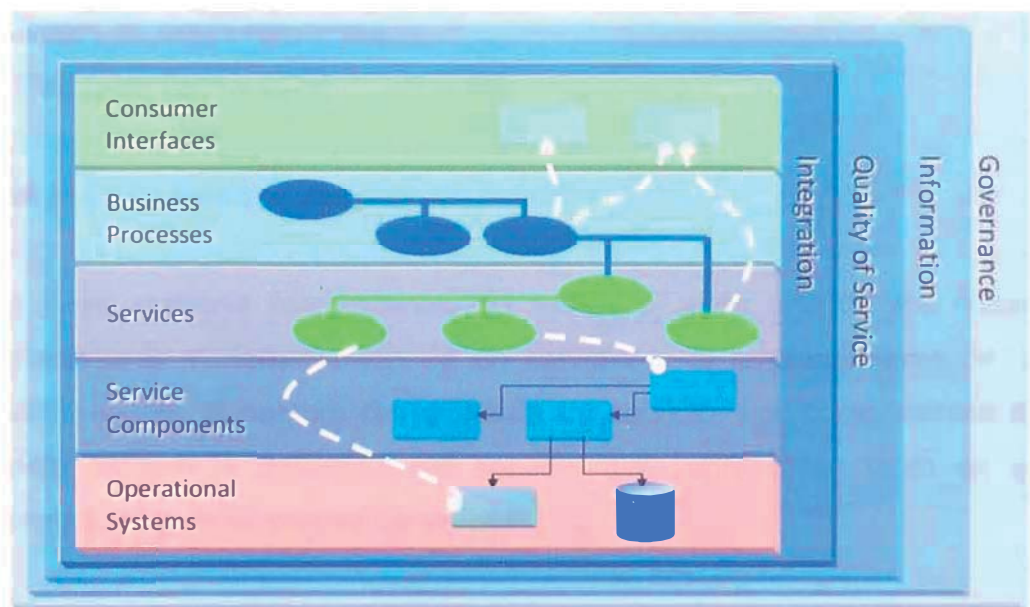


Figura 7. Capas de una Arquitectura Orientada a Servicios.
Fuente: SOA Reference Architecture.

Capa 1: Capa de Sistemas Operacionales

Esta capa captura la nueva y existente estructura de la organización, incluye los actores participantes necesarios para soportar una solución SOA:

- Toda la infraestructura para ejecutar SOA y sus componentes.
- Todos los servidores operativos.
- Todos los activos necesarios para apoyar la funcionalidad del servicio en SOA, incluyendo aplicaciones personalizadas o empaquetadas, nuevos servicios, servicios creados a través de orquestación, etc.

Un número de sistemas de software existentes son parte de esta capa. Estos sistemas incluyen:

- Aplicaciones personalizadas, incluyen aplicaciones J2EE y .NET.
- Otros servicios.
- Sistemas y aplicaciones legadas.
- Bases de datos existentes.
- Paquetes ERP y CRM.

Capa 2: Capa de componentes de servicio

Esta capa contiene componentes de software, cada uno de los cuales proporciona la implementación de un servicio, o el funcionamiento de un servicio, de ahí el nombre Componente de servicio. Los componentes de servicio reflejan la definición del servicio que representan tanto en su funcionalidad y en su calidad de servicio.

Revisemos el ejemplo en la figura siguiente, donde el servicio A se implementa usando una combinación del comportamiento del paquete X y la aplicación Y. La aplicación B, el consumidor, está acoplada solo a la descripción del servicio expuesto. El consumidor debe asumir que la

realización del servicio es fiel a su descripción publicada y es responsabilidad de los proveedores asegurarse que dicho cumplimiento se logre. Los detalles de la implementación, sin embargo, carecen de importancia para la aplicación B. A actúa como una fachada de la implementación del servicio. La aplicación B invoca con un contrato de servicio y las especificaciones definidas en la interfaz del servicio A.

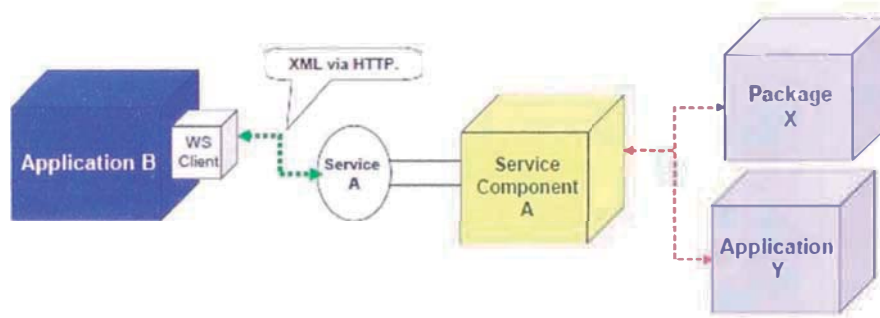


Figura 8. Capa de Componentes de Servicios.

Fuente: SOA Reference Architecture.

Capa 3: Capa de servicios.

Esta capa consiste de todos los servicio definidos en SOA. Esta capa puede ser considerada como un contenedor de descripción de servicios y de la implementación propia de los servicios.

La especificación da a los consumidores suficiente detalle para invocar las funciones de negocio expuestas por el proveedor de servicios. La especificación de servicio incluye:

- Una descripción de la funcionalidad ofrecida por el servicio. La información no es necesariamente un wsdl.
- Un documento de políticas.
- Descripción de la administración SOA.

- Adjuntos que categorizan o muestran dependencias entre servicios.

La capa de servicios incluye aspectos de integración, información, calidad del servicio y capa de gobierno.

La siguiente figura muestra una vista de la solución de cómo se va construir una arquitectura orientada a servicios utilizando un middleware de base y con servicios de infraestructura.

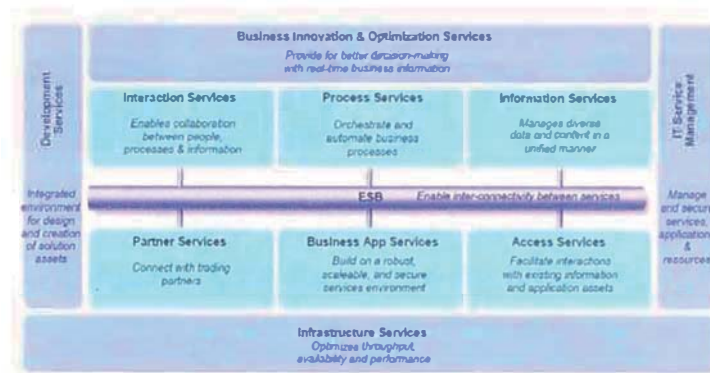


Figura 9. Capa de Servicios.

Fuente: SOA Reference Architecture.

Capa 4: Capa de procesos de Negocio

En un mundo orientado a servicios, las capacidades de negocio se realizan con los procesos de negocio o colecciones de los procesos de negocio. Estos procesos de negocio incluyen servicios de orquestación y composiciones, y la capacidad para insertar “la intervención humana” y apoyar a las operaciones de larga duración.

En particular, las composiciones y coreografías expuestas en la capa 3, se definen en esta capa. La evolución de los servicios de composición en los flujos, o coreografías de servicios integrados en un flujo actúan juntos para establecer una aplicación. Estas aplicaciones soportan específicos procesos de negocio. La siguiente figura muestra como un proceso de negocio P

puede ser implementado mediante los servicios A, B, C y D de la capa de servicios. El proceso P contiene la lógica de la secuencia en que los servicios deben ser invocados y ejecutados.

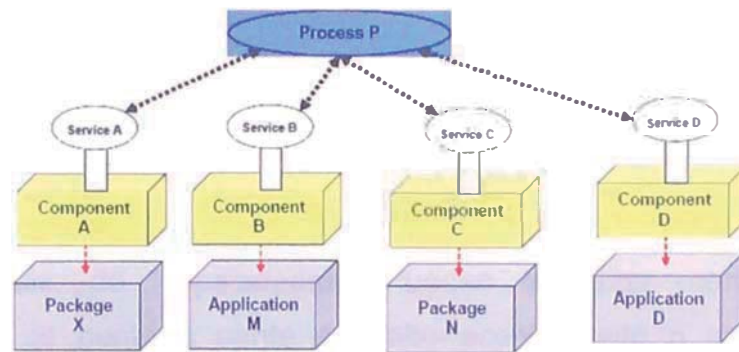


Figura 10. Capa de Negocio.

Fuente: SOA Reference Architecture.

La capa de procesos de negocio cubre el proceso de representación, composición de métodos, y construcción de bloques para el bajo acoplamiento de servicios como una secuencia de procesos alineados con los objetivos de negocio.

Capa 5: Capa Consumidora

SOA soporta un cliente independiente. Independizando el conjunto de canales de la funcionalidad que se consume. Esta capacidad permite a las organizaciones utilizar SOA para mejorar la calidad, la coherencia y la reutilización.

La capa consumidora de la arquitectura orientada a servicios proporciona la capacidad de crear rápidamente el fron-end de los procesos de negocio para responder en los cambios en el mercado. Esto permite al canal acceso

independiente a los procesos de negocio soportados por diversas aplicaciones y plataformas.

Capa 6: Capa de Integración

La capa de integración es un factor clave para una arquitectura SOA ya que proporciona la capacidad de transformar, enrutar y transportar los requerimientos del servicio a los proveedores correctos.

Esta capa permite la integración de servicio a través de la introducción de un conjunto fiable de capacidades. Pueden empezar con modestas capacidades de punto a punto con alto acoplamiento o cubrir todo el espectro de rutas, protocolos, y otros mecanismos de transformación a menudo descrito como ESB (Enterprise Service Bus).

La capa de integración tiene las siguientes características:

- Proporciona un nivel de integración entre el consumidor de la funcionalidad y su proveedor. Un consumidor de servicio interactúa con el proveedor de servicio vía la capa de integración.
- Consumidores y proveedores son desacoplados, su desacoplamiento permite la integración de sistemas heterogéneos en nuevas soluciones.

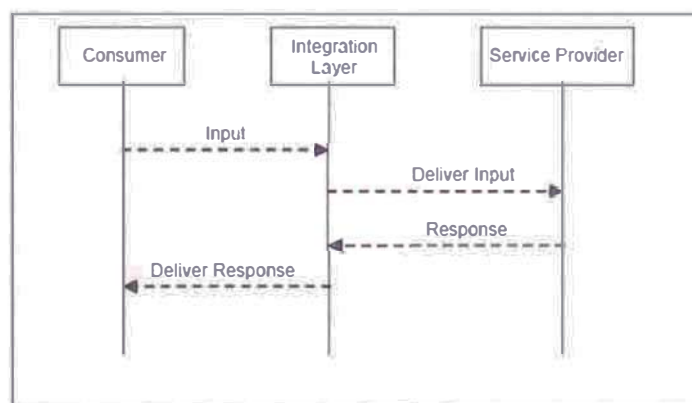


Figura 11. Capa de Integración.

Fuente: SOA Reference Architecture.

La capa de integración soporta la capacidad de reglas comunes, la habilidad para integrar el concepto de un ESB y una capacidad de registro para la virtualización del servicio.

Capa 7: Capa de calidad del servicio

Inherente a SOA son las capacidades que agravan las preocupaciones actuales de calidad de servicio en los sistemas informáticos: Virtualización, acoplamiento, uso generalizado de XML, composición de servicios, SLA, métricas de calidad de servicio para producir indicadores de negocio, etc. Estas características generan complicaciones para la calidad de servicio que claramente requieren atención dentro de cualquier solución SOA.

Esta capa establece requerimientos no funcionales relacionados a temas como:

- Confiabilidad.
- Disponibilidad.
- Capacidad de gestión
- Escalabilidad
- Seguridad

Por último, esto mejora el valor de negocio de SOA, permitiendo a las empresas monitorear los procesos de negocios contenidos en SOA con respecto a los KPI de negocio.

Capa 8: Capa de Arquitectura de Información

Esta capa incluye la arquitectura de información, inteligencia de negocios, consideraciones de meta-data, y asegura la inclusión de consideraciones fundamentales relativas a la arquitectura de datos y arquitecturas de información que también pueden ser utilizadas como la base para la

creación de la inteligencia empresarial a través de los Data Marts y Data Warehouse.

Capa 9: Capa de Gobierno

El gobierno de SOA garantiza que los servicios y soluciones de SOA dentro de una organización se adhieran a las políticas definidas, directrices y normas que se definen en función de los objetivos, estrategias y reglamentos que se aplican en la organización.

El objetivo de la capa de gobierno de SOA es asegurar la coherencia de la cartera de servicios. En esta capa se asegurará que todos los aspectos de la Arquitectura Orientada a Servicios son gestionados y gobernados tales como:

- Acuerdos de nivel de servicio.
- Políticas de administración de capacidad y rendimiento.
- Seguridad.

La información en esta consiste en:

- Lineamientos de gobierno SOA.
- Lineamientos de servicios y soluciones SOA.
- Mejores prácticas.
- Políticas.
- Estándares.
- Hojas de ruta de los servicios y de la solución SOA.
- Cumplimiento y comunicación de la documentación.

CAPÍTULO III

PROCESO DE TOMA DE DECISIONES

3.1. SITUACIÓN ACTUAL

En una empresa de telecomunicaciones existen muchas aplicaciones que soportan diversos procesos de negocio, es probable que cada vez que se crea un nuevo sistema se requiera de alguna funcionalidad que ya fue implementada en algún otro aplicativo, lo que lleva a desarrollar algún componente de integración para reutilizar código existente en alguna aplicación ya desarrollada.

En Telecom tenemos las siguientes aplicaciones principales:

- CRM-Portal: Portal con una interface web, usado por representantes de servicio y otros usuarios para ver y modificar información del cliente. CRM-Portal tiene muchos componentes que son aplicaciones back end que lo utilizan como interface para mostrar información.
- Ordenes: Aplicación que administra las órdenes de venta de equipos, datos y banda ancha, y ordenes de post venta, como cambio de plan, actualización de equipos y servicios adicionales.
- Administración de clientes: Es el modulo que administra la información de los clientes. Todos los usuarios pueden ver esta información pero solo los representantes de venta que son dueños del cliente pueden modificar esta información.

- **Acuerdos Comerciales:** Este módulo administra el workflow para obtener y aplicar descuentos a los clientes. Los representantes de venta aplican el descuento y luego este tiene que ser aprobado por su supervisor inmediato.
- **Catálogo de productos:** Este módulo administra los precios y promociones de los productos comerciales. Soporta varios productos para ser vendidos por el sistema de órdenes.
- **Caja:** Aplicación que procesa todos los pagos recibidos en las tiendas Telecom. Responsable de la administración de caja, es decir, abrir, cerrar y manejar las transacciones diarias de caja.
- **Créditos:** Sistema que brinda una interface para la evaluación crediticia de los clientes. La evaluación resulta generada por el sistema de acuerdo a unas variables como pagos y límite de crédito los cuales son usados como referencia para la aprobación de las ventas.
- **Retail:** Punto de venta para las tiendas Retail. Es un quiosco para comprar, cambiar y actualizar equipos.
- **Oracle Financial:** Es el ERP de la empresa, administra el sistema contable y financiero.
- **BPEL (Business Process Execution Language):** Solución de integración para interactuar con BSCS. Es un componente que sirve para la orquestación de servicios, maneja un lenguaje de ejecución de procesos.
- **BSCS (Business Support & Control System):** Sistema de facturación y atención al cliente que es ampliamente utilizado por empresas de telefonía. Este sistema se encarga de la activación de los equipos comunicándose con los elementos de red.
- **ATS:** Sistema de aprovisionamiento y facturación prepago. Permite la configuración de planes tarifarios.
- **ICC:** Sistema de prepago y control para el mercado de productos 3G. Maneja el control físico, nominal y monetario de las tarjetas prepago.

- Content Manager: sistema que almacena imágenes y documentos para luego ser entregados a los sistemas que lo requieran.

Para mostrar la situación actual en términos de arquitectura de aplicaciones vamos a utilizar el siguiente gráfico conceptual:

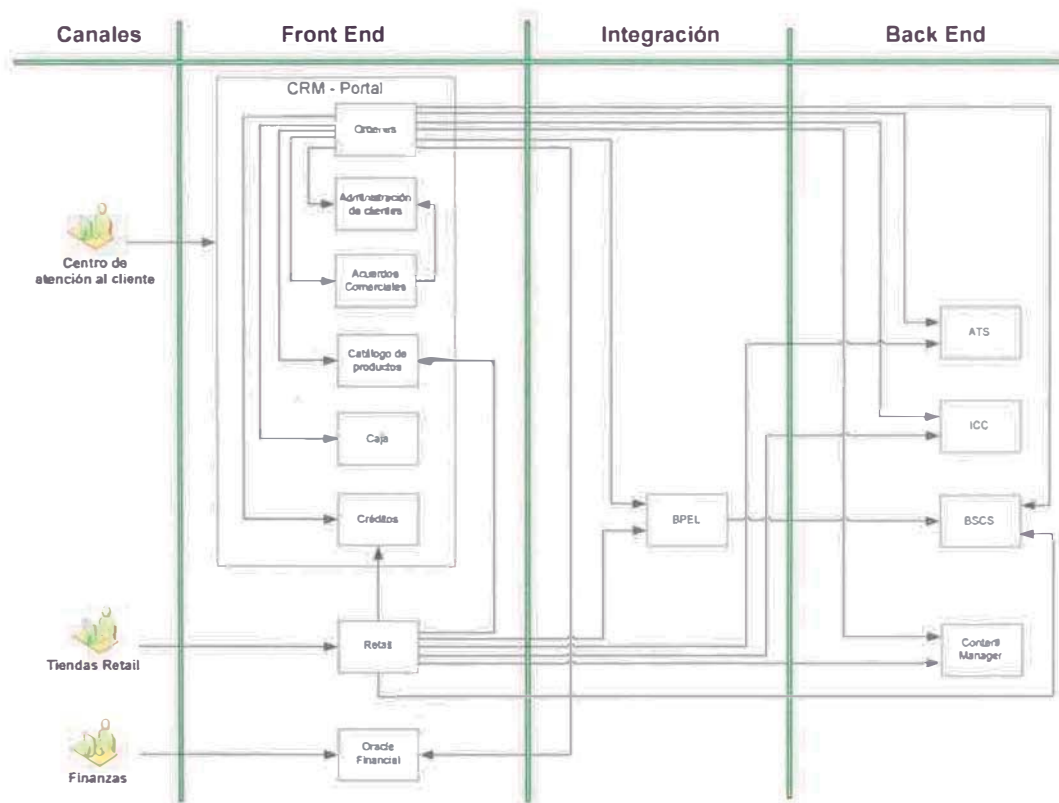


Figura 12. Diagrama Conceptual de aplicaciones de la empresa Telecom.

Fuente: Elaboración propia.

El único intento por integrar las aplicaciones es a través de BPEL, que desempeña labores de orquestación que soporta algunos procesos como es el de activación de servicios.

La arquitectura de integración con la que cuenta Telecom es la de Punto a Punto, cada aplicación implementa en el código alguna forma de

comunicarse con la otra. Para apreciar mejor este punto tenemos el siguiente gráfico:

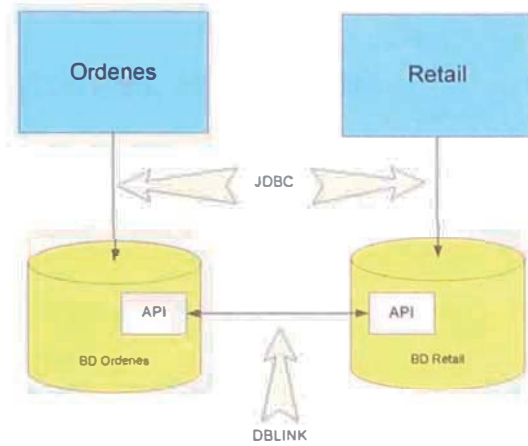


Figura 13. Integración punto a punto en Telecom.

Fuente: Elaboración propia.

En la figura observamos dos aplicaciones importantes de Telecom, el sistema de órdenes de venta y el sistema de finanzas. Ambas son aplicaciones desarrolladas en la tecnología java, y Oracle para la persistencia de los datos. Como mencionamos en el capítulo de Marco Teórico las arquitecturas Punto a Punto obligan a las aplicaciones a usar un adaptador que se diseña y desarrolla para administrar transferencia de data. En nuestro caso este adaptador es lo que se denomina comúnmente API de base de datos, que es una interfaz que nos oculta la complejidad de la conexión a otro sistema.

Los tipos de conexión que se tienen en Telecom son de una base de datos a otra base de datos a través de DBLINK, y de una base de datos a un servicio web a través del protocolo SOAP.

Al no contar con tantas capas en las aplicaciones la comunicación entre ellas es muy rápida pero ocasiona muchos problemas para el mantenimiento de las mismas.

3.2. IDENTIFICACIÓN DEL PROBLEMA

Como parte del lanzamiento de la banda 3G, Telecom impulsa la iniciativa de contar con un nuevo portal empresarial. La visión del portal empresarial en el largo plazo es contar con una plataforma única que le brinde soporte a los diferentes stakeholders de Telecom Perú (clientes, prospectos, empleados, periodistas, distribuidores, dealers entre otros) buscando darles una experiencia única y relevante. Es en este contexto se presenta la siguiente incógnita: ¿Cómo integrar este nuevo portal al resto de aplicaciones que se tiene en la empresa?

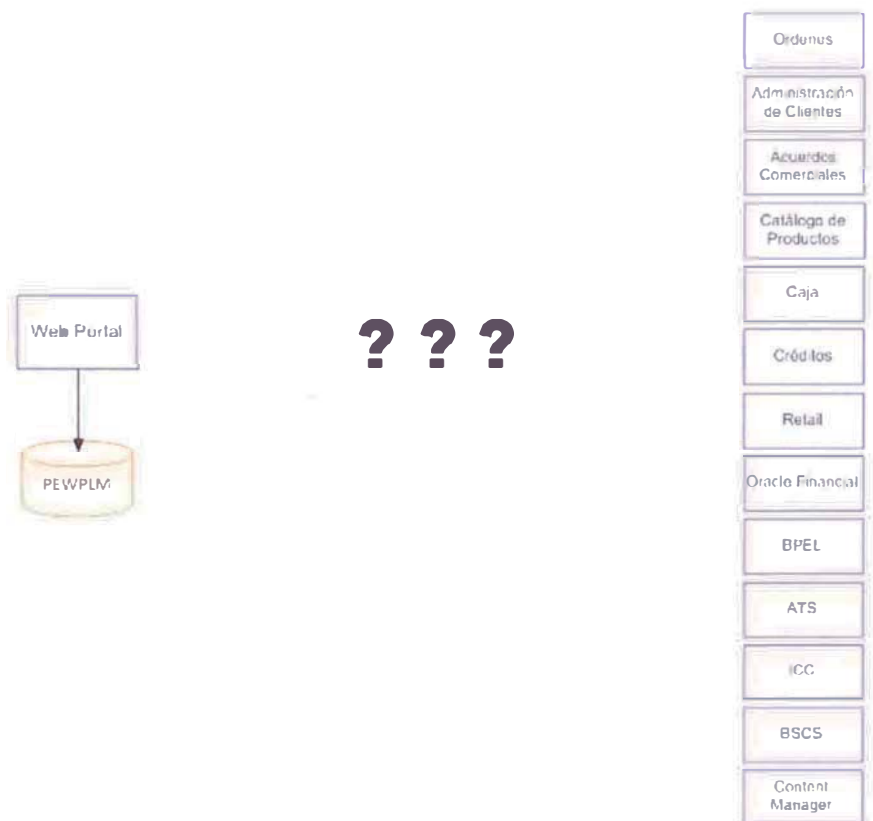


Figura 14. Definición del problema.

Fuente: Elaboración propia.

Por el lado izquierdo tenemos la aplicación Web Portal y por el lado derecho tenemos el resto de aplicaciones que de alguna manera Web Portal usara para cumplir con sus procesos. Las más importantes son Órdenes, ICC, ATS y BSCS.

La tecnología que se utilizará para construir el portal es JAVA para la presentación y Oracle para la persistencia de los datos. Telecom posee un soporte Premium de Oracle, esto le permite solicitar algún especialista para que le solucione problemas específicos que no pueden ser atendidos por su área de sistemas.

Todas estas aplicaciones de Telecom son muy parecidas en cuanto a estructura, todas manejan java o Pl/Sql para la capa de presentación y Oracle para la capa de lógica del negocio y persistencia de datos.

Se plantean 2 alternativas de solución:

- Integración Punto a Punto
- Integración a través de un bus de servicios.

3.3. PLANTEAMIENTO DE ALTERNATIVAS DE SOLUCIÓN

3.3.1. Alternativa 1: Integración punto a punto

Esta alternativa es la más fácil de implementar debido al uso intensivo de la tecnología Oracle en la empresa. De usar esta alternativa el enfoque de construcción de aplicaciones no cambiaría.

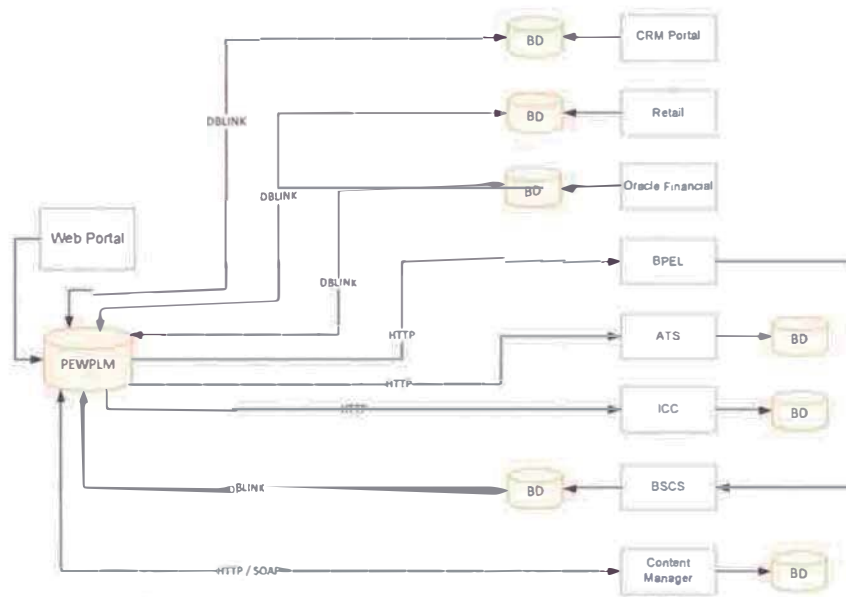


Figura 15. Alternativa 1: Integración punto a punto.

Fuente: Elaboración propia.

Para simplificar el gráfico se fusiono todas las aplicaciones desplegadas en el CRM Portal como una sola aplicación.

Ventajas:

- Fácil de implementar.
- Este tipo de arquitectura es ampliamente conocida por el equipo de desarrollo.
- Bajo costo.
- No requiere de adquisición de nuevas tecnología.
- La eficiencia de esta solución es muy buena, debido a que se utiliza la capa de base de datos para la integración.

Desventajas:

- Cada conexión puede ser relativamente sencilla, pero como el número de conexiones aumenta el entorno de las aplicaciones se hace más complejo.
- El mantenimiento de este tipo de arquitectura es muy costoso.
- Este tipo de enfoque tiene un alto grado de acoplamiento lo que la hace poco flexible cuando es necesario realizar cambios en la arquitectura.
- Muchos de los productos que se ofrecen en el mercado, como por ejemplo, ERP, SOA, Portales, etc., utilizan otros enfoques para la integración de aplicaciones.

3.3.2. Alternativa 2: Integración por bus de servicios

Esta alternativa plantea el uso de un bus de servicios para la integración de aplicaciones, hay que tener en cuenta que de usarse esta alternativa el paradigma de programación en Telecom tendría que cambiar a un enfoque orientado a servicios.

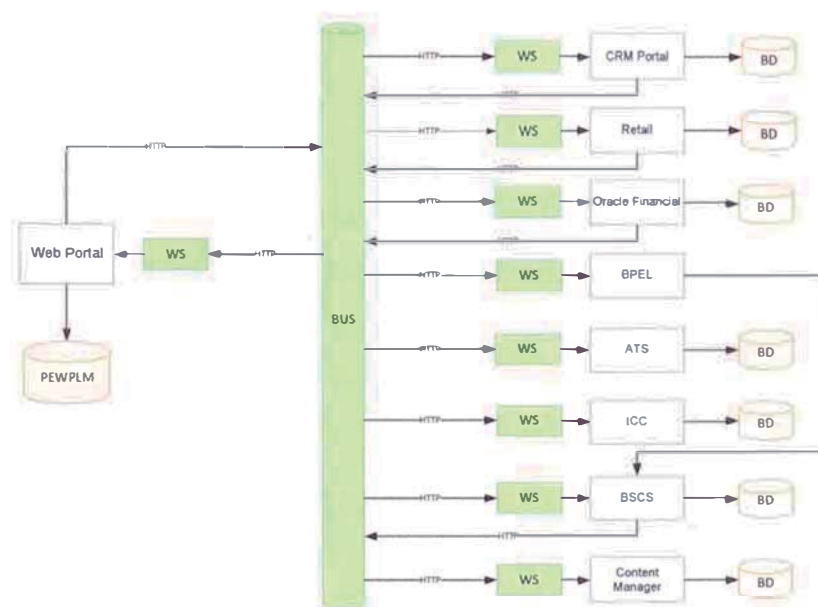


Figura 16. Alternativa 2: Integración por bus de servicios.

Fuente: Elaboración propia.

Ventajas:

- Es muy flexible a los cambios.
- Menos codificación, más configuración.
- Es fácil de mantener.
- Bajo acoplamiento entre las aplicaciones
- Nuevas aplicaciones son fácilmente adaptadas y configuradas para que utilicen el bus de servicios. Esto hace que el diseño sea mucho más simple.
- Implementar el bus de servicios es el primer intento para acercarse a una arquitectura orientada a servicios que es el nuevo paradigma para construir arquitecturas empresariales eficientes.
- A largo plazo esta arquitectura reduce costos, debido a la mantenibilidad, escalabilidad y bajo acoplamiento que logran las aplicaciones.

Desventajas:

- La comunicación entre las aplicaciones se hace más lenta al aumentar una capa de integración.
- El costo de implementar esta arquitectura es elevado.
- El equipo de desarrollo no está familiarizado con este enfoque.
- Al tener como único punto de conexión el Bus de servicios, una falla en este podría ocasionar que ningún sistema funcione correctamente.

3.4. CRITERIOS DE SELECCIÓN

Para determinar cuál es la arquitectura de integración que se debería implementar se va realizar un análisis en base a criterios de selección que nos darán una visión general y nos ayudara a escoger una alternativa.

3.4.1. Criterio 1: Costo de la implementación

Para determinar el costo, necesitamos conocer cuántos son los puntos de integración, según los requerimientos funcionales del sistema, Web Portal necesita 60 interfaces de integración divididos de la siguiente manera:

Aplicación	Número de interfaces
CRM Portal	25
Retail	5
Oracle Financial	3
ATS	7
ICC	7
BSCS	12
Content Manager	1
TOTAL	60

Tabla 1. Número de Interfaces.

De este cuadro podemos deducir que se necesitarán analistas por cada módulo y desarrolladores que implementarán los puntos de integración. La diferencia entre una alternativa y otra es que se necesitará nuevo hardware y más desarrolladores para que implementen los servicios y los configuren en el bus de servicios. Este costo adicional lo podemos apreciar en el siguiente gráfico:

Requerimientos	Integración punto a punto	Integración con bus
Analistas	\$20,000.00	\$20,000.00
Programadores	\$16,000.00	\$24,300.00
Soporte ESB		\$6,000.00
Software		\$43,700.00
Hardware		\$18,000.00
TOTAL	\$36,000.00	\$92,000.00

Tabla 2. Comparación de costos de alternativas.

Implementar una solución con un bus de servicios costaría \$76,000.00 más, en total tres veces más que una punto a punto.

La ponderación quedaría así:

Integración punto a punto: 1

Integración por bus de servicios: 3

3.4.2. Criterio 2: Tiempo de respuesta

Para analizar este criterio es necesario analizar el número de capas que se tiene por cada alternativa. A mayor número de capas mayor el tiempo de respuesta de una transacción. Para determinar cual pasa por más capas vamos a suponer una consulta del aplicativo CRM Portal al aplicativo Retail.

En el enfoque punto a punto la transacción tendría que pasar por: Capa de presentación de CRM Portal – Capa de negocio de CRM Portal – Capa de base datos CRM Portal – Capa de base de datos de Retail. En total 4 capas de aplicación para llegar a su objetivo.

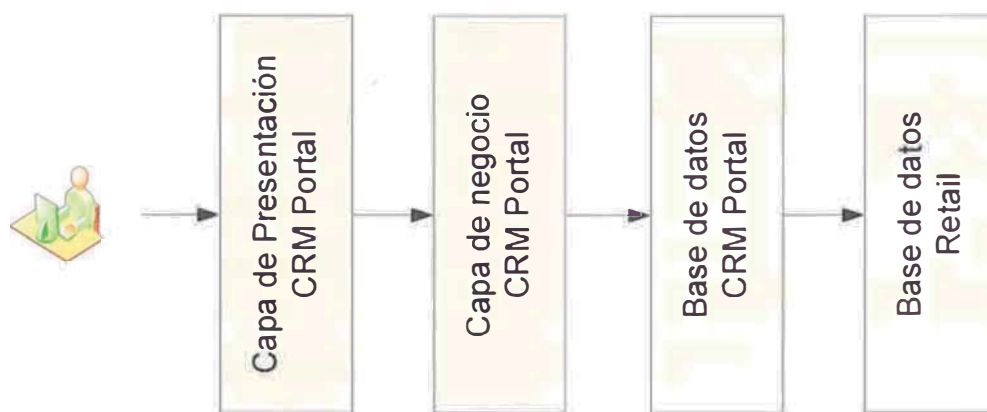


Figura 17. Capas en una Integración Punto a Punto.

Fuente: Elaboración propia.

En el enfoque SOA la transacción tendría que pasar por: Capa de presentación CRM Portal – Capa de negocio CRM Portal – Capa de bus de servicios – Capa de servicios Retail – Capa de base de datos de Retail. En total 5 capas de aplicación para llegar al objetivo.

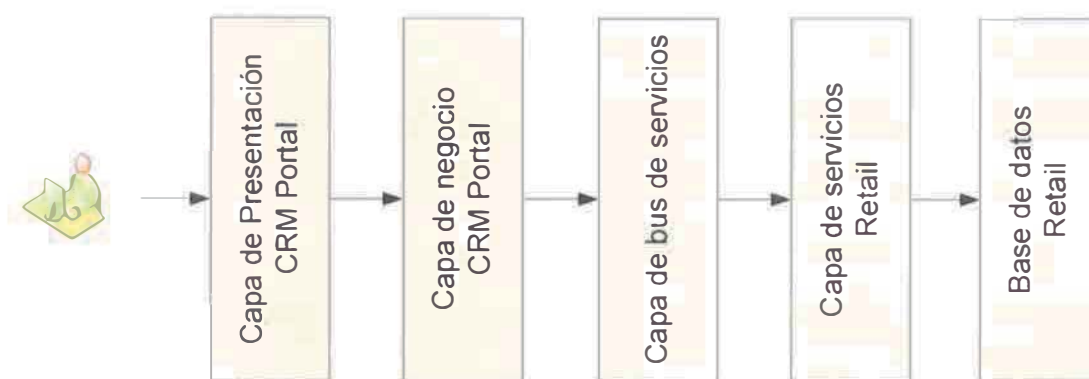


Figura 18. Capas en una integración por Bus de Servicios.

Fuente: Elaboración propia.

Cabe mencionar que el bus y los servicios se encuentran desplegados en un servidor de aplicaciones independiente lo que hace más lento el tránsito de datos por esta capa.

Se realizaron pruebas de concepto y se llegó a la conclusión de que la integración por bus de servicios es 25% más lenta que una integración punto a punto. La ponderación quedaría de la siguiente manera:

Integración punto a punto: 4

Integración por bus de servicios: 5

3.4.3. Criterio 3: Productividad

Podemos definir la productividad como la relación entre los resultados y el tiempo utilizado para obtenerlos.

Imaginemos que tenemos un proceso back End que queremos utilizar en las aplicaciones Front End.

En una integración punto a punto tenemos que construir un adaptador por cada conexión a la aplicación Back End.

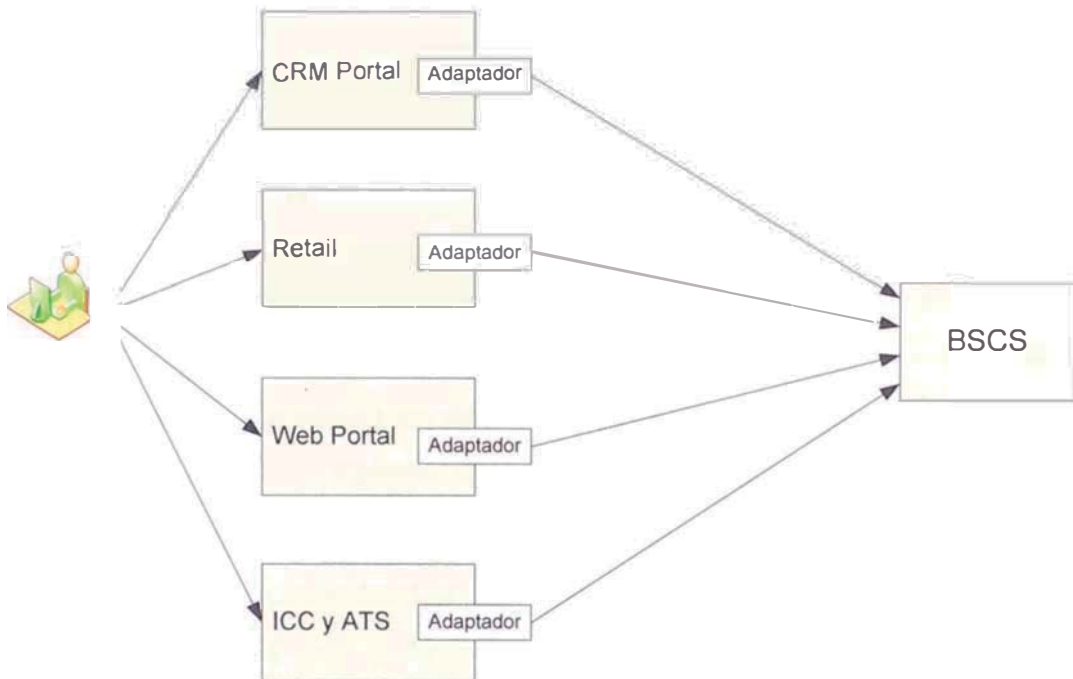


Figura 19. Adaptadores en una integración Punto a Punto.

Fuente: Elaboración propia.

Para el caso de una arquitectura integrada con un bus de servicios la tarea de conexión es más sencilla. Simplemente tendríamos que crear un servicio Web y exponerlo en el bus de servicios, aquí se configuraría un adaptador que sería usado por todas las aplicaciones que quisieran conectarse a este proceso.

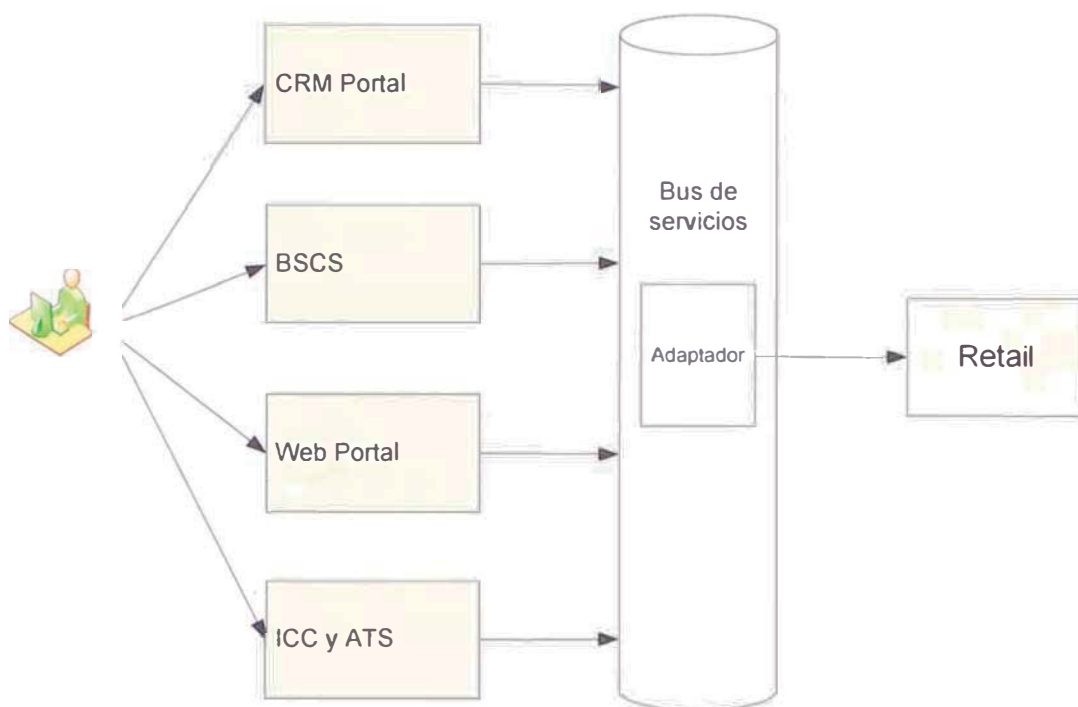


Figura 20. Adaptador en una integración por Bus de Servicios.

Fuente: Elaboración propia.

Después de la implementación de Web Portal Telecom contará con tres aplicaciones principales y la cantidad estimada de interfaces que se construirán en los próximos años será la siguiente:

CRM Portal	20
Retail	18
Web Portal	21
BSCS	17
ICC y ATS	22

Con el bus de servicios solo será necesario la creación de un adaptador por cada interfaz lo que hace un total de 98 adaptadores.

Con la integración punto a punto será necesario construir 4 adaptadores por cada tipo de aplicación lo que hace un total de 392.

Si definimos la productividad en el caso de desarrollo de aplicaciones en Telecom como:

$$\text{Productividad} = \text{Interfaces} / \text{Cantidad de desarrolladores}$$

Teniendo en cuenta que la cantidad de desarrolladores será la misma para cualquiera de las dos alternativas, la ponderación quedaría:

Integración punto a punto: 1

Integración por bus de servicios: 4

3.4.4. Criterio 4: Interoperabilidad

El Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) define interoperabilidad como la habilidad de dos o más sistemas o componentes para intercambiar información y utilizar la información intercambiada. Las aplicaciones de Telecom están desplegadas en diferentes plataformas (Unix, Linux, Windows), y construidas utilizando diferentes tecnologías (Java, .Net, Oracle), haciendo de este criterio uno de los más importantes.

La integración punto a punto difícilmente pueda brindar interoperabilidad, cada aplicación que requiere información de otra, necesita implementar el conector para comunicarse con ella. No hay reutilización y se hace difícil que las aplicaciones desarrolladas con diferentes tecnologías intercambien información.

La integración por bus de servicios hace mucho más fácil esta comunicación, el principal motivo es que el bus de servicio sirve como un ente conector entre las aplicaciones, este se encarga de conseguir los datos, transformarlos y utilizar diversos conectores para casi la totalidad de tecnologías que se encuentran disponibles en el mercado.

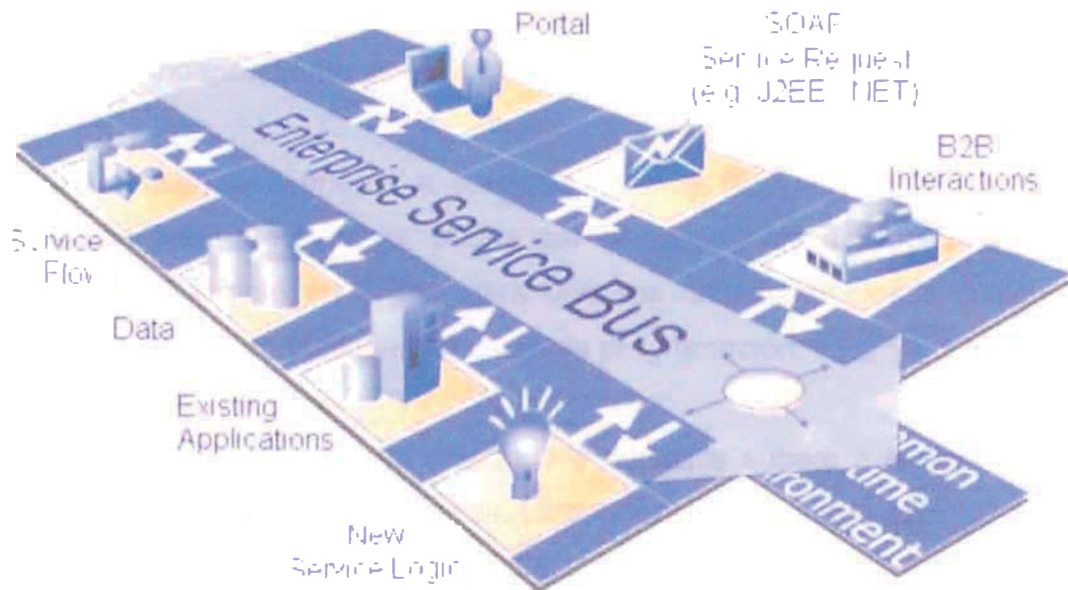


Figura 21. Enterprise Service Bus.

Fuente: eAppWare.

La ponderación quedaría así:

Integración punto a punto: 0

Integración por bus de servicio: 1

3.4.5. Criterio 5: Reutilización

Para analizar este criterio tomemos el proceso de activación de un equipo de telefonía celular. Este es un proceso que puede ser usado desde diferentes canales de venta, activamos un equipo cuando vamos al centro de atención al cliente, cuando vamos a una tienda Retail o en este último caso desde el Web Portal. El proceso que debería ser implementado una sola vez y luego ser reutilizado por el resto de aplicaciones.

En una arquitectura punto a punto no tenemos las herramientas para cumplir con este objetivo pues cada aplicación implementa su proceso de activación de manera independiente pues no tienen un mecanismo que lo centralice.

En una arquitectura orientada a servicios es muy fácil implementarlo. Solo basta con crear un servicio, podría ser el servicio de activación, y este debería configurarse en el bus para que sea consumido desde cualquier aplicación escrita en cualquier lenguaje de programación.

En conclusión la arquitectura orientada a servicios supera a la de punto a punto en la reutilización de servicios. La ponderación quedaría así:

Integración punto a punto: 0

Integración por bus de servicios: 1

3.4.6. Criterio 6: Costo de mantenimiento

Este criterio tiene mucha relevancia pues Telecom gasta mucho dinero en el mantenimiento de sus aplicaciones, tiene un área de soporte a la producción muy numerosa y grupo de desarrolladores dedicados única y exclusivamente a la solución de incidencias que ocurren en los aplicativos que están desplegados en ambientes productivos.

Una integración punto a punto es muy fácil de implementar pero muy costosa de mantener, pues se crean muchísimas conexiones con adaptadores exclusivos. Este problema no ocurre con una arquitectura orientada a servicios, en donde el mantenimiento de la capa de integración es muy sencillo y muy configurable.

La medición de este tipo de criterio es muy complicada, lo que se utilizó es el juicio de los expertos para llegar a una ponderación final:

Integración punto a punto: 3

Integración por bus de servicios: 1

3.4.7. Criterio 7: Know How del equipo de Sistemas

Este es un tema muy importante, pues el conocimiento de las tecnologías es fundamental para implementar un proyecto de desarrollo de software.

Se realizó una investigación dentro de la empresa para conocer cuál es el conocimiento que tienen los profesionales, sobretodo programadores, con las tecnologías relacionadas con una arquitectura orientada a servicios.

	Personas que tienen conocimiento las tecnologías de integración
Integración punto a punto	34
Integración por bus de servicios	8

Tabla 3. Tabla de Know How.

En total son 34 personas quienes conforman el equipo de desarrollo de software de Telecom. En base a estos resultados podemos sacar la siguiente ponderación.

Integración punto a punto: 4

Integración por bus de servicios: 1

3.5. SELECCIÓN DE UNA ALTERNATIVA DE SOLUCIÓN

Para la decisión final, se utilizará una Matriz Multicriterio que es una herramienta que ayuda a la toma de decisiones en base a factores cualitativos o a múltiples factores no homogéneos que intervienen en un suceso.

Es importante mencionar que la gerencia de sistemas encargo esta tarea al grupo de Arquitectos de Telecom.

Todos los pesos asignados fueron definidos en base al criterio de los arquitectos expertos, estos armaron la siguiente matriz en base a los criterios ya descritos.

3.5.1. Matriz Multicriterio

	Costo de implementación (P=-2)		Tiempo de respuesta (P=1)		Productividad (P=1)		Interoperabilidad (P=1)		Reutilización (P=1)		Costo de Mantenimiento (P=-2)		Know How del equipo de sistemas (P=2)		Total
	V	VxP	V	VxP	V	VxP	V	VxP	V	VxP	V	VxP	V	VxP	
Integración punto a punto	1	-2	4	4	1	1	0	0	0	0	4	-8	4	8	3
Integración por bus de servicios	3	-6	5	5	4	4	1	1	1	1	1	-1	1	2	6

Tabla 4. Resultados de la Matriz de Multicriterio.

Podemos apreciar claramente que una de las alternativas gana con un contundente 6 -- 3.

Por lo tanto la alternativa elegida es la número 2:

Integración por bus de servicios

3.6. PLAN DE ACCIÓN

El plan de acción permite visualizar todas las actividades necesarias para desarrollar la solución seleccionada.

Para determinar cuáles son las actividades que se deben realizar se tomó como referencia el framework para proyectos de desarrollo de software que maneja Telecom:

1. Definir y analizar el desarrollo de la solución.
 - 1.1 Determinar el alcance.
 - 1.2 Determinar los requerimientos.
 - 1.3 Definir los casos de uso.
 - 1.4 Definir la arquitectura.
2. Diseñar la solución.
 - 2.1 Diseñar los servicios.
 - 2.2 Diseñar la integración de los servicios.
3. Construir la solución.
 - 3.1 Construir los servicios.
 - 3.2 Construir los casos de prueba.
 - 3.3 Ejecutar pruebas unitarias.
4. Probar la solución
 - 4.1 Ejecutar las pruebas del sistema.
 - 4.2 Revisar la solución con el cliente.
5. Desplegar la solución.
 - 5.1 Establecer el entorno de despliegue.
 - 5.2 Instalación en ambiente de producción.

El desarrollo fue dividido en dos partes bien diferenciadas:

- Front End: Desarrollo de la interface usuario, se utilizo el Web Center de Oracle para implementar el portal WEB.
- Back End: Desarrollo de los procesos internos a través de servicios web, se utilizo el Oracle Service Bus para la integración de servicios web.

A modo de resumen se detalla todo el proceso de desarrollo de un requerimiento en la parte Back End, por lo que se presentara las interfaces de los servicios más no interfaces usuario.

3.6.1. Requerimiento Funcional de ejemplo

Permitir al cliente que consulte su detalle de llamadas de Interconexión Telefónica

Se requiere que un cliente de Telecom pueda consultar el detalle de sus llamadas de interconexión telefónica para el ciclo de facturación actual o los últimos 4 ciclos.

Sobre la autenticación:

Para poder utilizar esta funcionalidad del Web Portal, es necesario que el usuario sea un cliente de Telecom y que haya ingresado su clave de registro.

Sobre la consulta:

Por defecto se le debe mostrar al cliente un resumen de consumo por día del ciclo actual y así como también el detalle de llamadas por cada día. Los campos a mostrar son los mismos que se muestran actualmente en la web de Telecom.

El cliente también debe tener la posibilidad de seleccionar un ciclo de facturación anterior, pudiendo seleccionar cualquier de los últimos 4 ciclos, los cuales están especificados con fecha de inicio y fecha de fin de ciclo.

Una vez que el cliente haya ingresado a la consulta y antes de mostrar los registros del ciclo actual por defecto, se debe generar un incidente del tipo consulta en CRM portal con la referencia del número del cliente de Telecom y el tipo de consulta. Este incidente se generará una única vez por el uso de la opción.

3.6.2. Caso de Uso

Consulta de detalle de llamadas de interconexión telefónica

- Diagrama

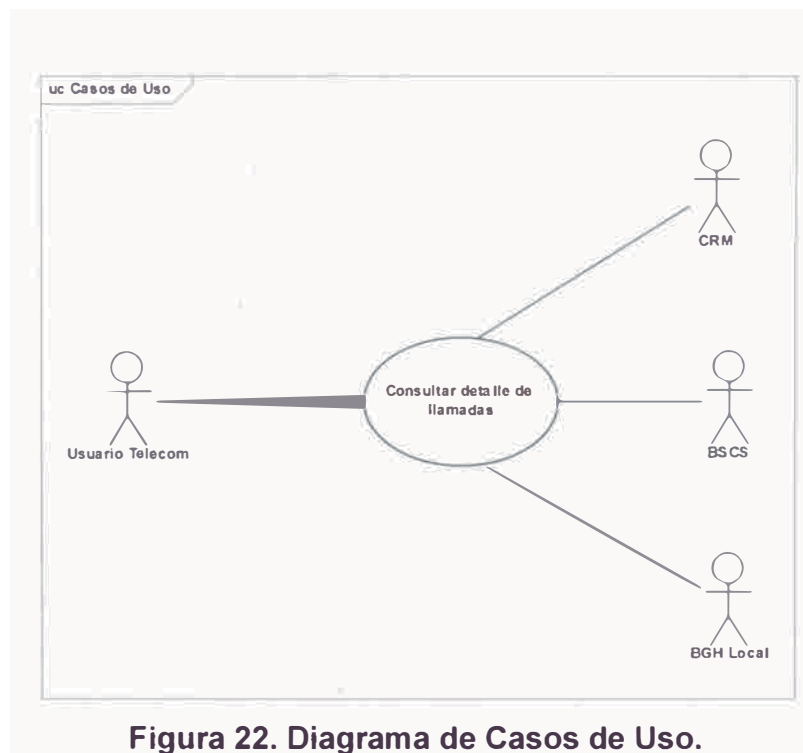


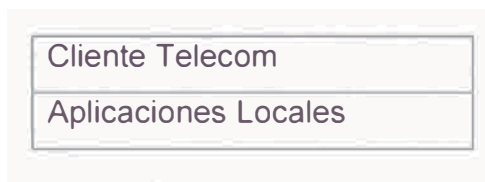
Figura 22. Diagrama de Casos de Uso.

Fuente: Telecom.

- Resumen

El cliente de Telecom visualiza en pantalla en consumo, del periodo actual, el detalle de las llamadas telefónicas realizadas. Además, el cliente de Telecom tiene la opción de hacer la consulta de sus llamadas por mes (últimos 4 meses).

- Actores



- Post-Condiciones

1. El cliente de Telecom visualiza el detalle de sus llamadas en el período actual o en el periodo que seleccione
2. Se crea un incidente de consulta de detalle de llamadas

- Flujo de Básico

1. El cliente de Telecom ingresa a la aplicación Web Portal.
2. La aplicación Web Portal muestra las siguientes opciones:
 - a. Cliente
 - b. No cliente
3. Si el cliente de Telecom selecciona la opción de "Cliente"
 - a. La aplicación Web Portal solicita el número.
 - b. El cliente de Telecom digita su número.
 - c. La aplicación Web Portal valida el número ingresado
4. La aplicación Web Portal muestra el menú principal

5. El cliente de Telecom selecciona la opción de “Consulta de detalle de llamadas de interconexión telefónica”
 6. La aplicación Web Portal solicita el ingreso de su clave de usuario.
 7. El cliente de Telecom ingresa su clave de usuario
 8. La aplicación Web Portal crea el incidente del tipo pedido de detalle de llamadas
 9. La aplicación Web Portal muestra en pantalla el resultado de la consulta de detalle de llamadas para el ciclo actual. El listado de llamadas realizadas se encuentra agrupado por día (mostrándose un resumen por día) y además se muestra el resumen por ciclo.
 10. El cliente de Telecom presiona el botón “Imprimir”.
 11. La aplicación Web Portal imprime el detalle de la consulta
 12. El caso de uso termina
- Flujos Alternativo

AF1

Si en el paso 2 el cliente de Telecom selecciona la opción de “No cliente”

1. La aplicación Web Portal muestra el menú principal
2. El cliente de Telecom selecciona la opción de “Consulta de detalle de llamadas de interconexión telefónica”
3. La aplicación Web Portal muestra mensaje informativo “Esta opción solo está disponible para clientes de Telecom”
4. La aplicación Web Portal retorna al menú principal

5. El flujo alternativo AF1 termina

AF2

Si en el punto 3.c el cliente de Telecom ingresa un número de teléfono:

1. La aplicación Web Portal muestra un mensaje de error indicando que el número no es válido
2. El flujo retorna al punto 2 del flujo básico

AF3

Si en el punto 5 del flujo básico, el cliente no posee una clave de acceso:

1. Web Portal sugiere ir a la opción de registro de usuarios
2. El cliente selecciona la opción sugerida
3. Web Portal muestra la pantalla de "Registro de usuarios" (UC011)
4. El flujo alternativo AF3 termina

AF4

Si en el punto 7 del flujo básico, el cliente de Telecom no recuerda su clave de acceso:

1. Web Portal sugiere ir a la opción de recuperación de clave de usuario
2. El cliente selecciona la opción sugerida
3. Web Portal muestra la pantalla de "Recuperación de clave de usuario" (UC012)
4. El flujo alternativo AF4 termina

AF5

En el punto 7 del flujo básico, si la clave del usuario de Telecom es errónea:

1. Web Portal muestra un mensaje de error y pide ingresar la clave nuevamente con un máximo de 3 intentos. En el caso de que se ingrese erróneamente más de 3 veces el flujo continua en el punto 4 del flujo básico (menú principal)
2. En el caso de que el cliente ingrese correctamente su clave, el flujo continúa en el punto 8 del flujo básico

AF6

En el punto 7 del flujo básico, si el cliente de Telecom no se encuentra registrado en Web Portal:

1. Web Portal registra un incidente del tipo consulta describiendo la intención del usuario de poder consultar su detalle de llamadas y además le da las opciones de registrarse o de recordar su clave según sea el caso.

AF7

Si finalizando el punto 9 del flujo básico el cliente selecciona ver detalle de un día:

1. Web Portal muestra el detalle de llamadas del día seleccionado por el cliente
2. El flujo alternativo AF5 termina

AF8

Si finalizando el punto 9 del flujo básico el cliente selecciona otro ciclo de consulta diferente al actual:

1. Web Portal muestra la opción de poder seleccionar un ciclo diferente de consulta de llamadas, pudiendo el cliente elegir entre los últimos 4 ciclos, los cuales se muestran indicados por fecha de inicio y fecha de fin de ciclo.
2. El cliente selecciona algún ciclo anterior al actual
3. Web Portal muestra en pantalla el resultado de la consulta de detalle de llamadas para el ciclo seleccionado agrupadas por día
4. El flujo continúa en el punto 10 del flujo básico
5. El flujo alternativo AF7 termina

- Asunciones

1. El número con el que se ingresa a la aplicación tiene un plan Postpago.

- Criterio de Aceptación

1. El cliente de Telecom puede visualizar en pantalla los datos de detalle de llamadas de interconexión telefónica del periodo seleccionado.
2. Se visualiza en el CRM Portal el incidente creado con la consulta de detalle de llamadas realizada por el cliente

- Riesgos

El tiempo de respuesta de la consulta de tráfico aumenta cuando el equipo desde se consulta está fuera de la red de Telecom.

3.6.3. Modelo de Análisis

Con este modelo se va a describir la realización del caso de uso. El siguiente diagrama muestra las clases de análisis y se clasifican en 3 tipos: Interface, Control y entidad.

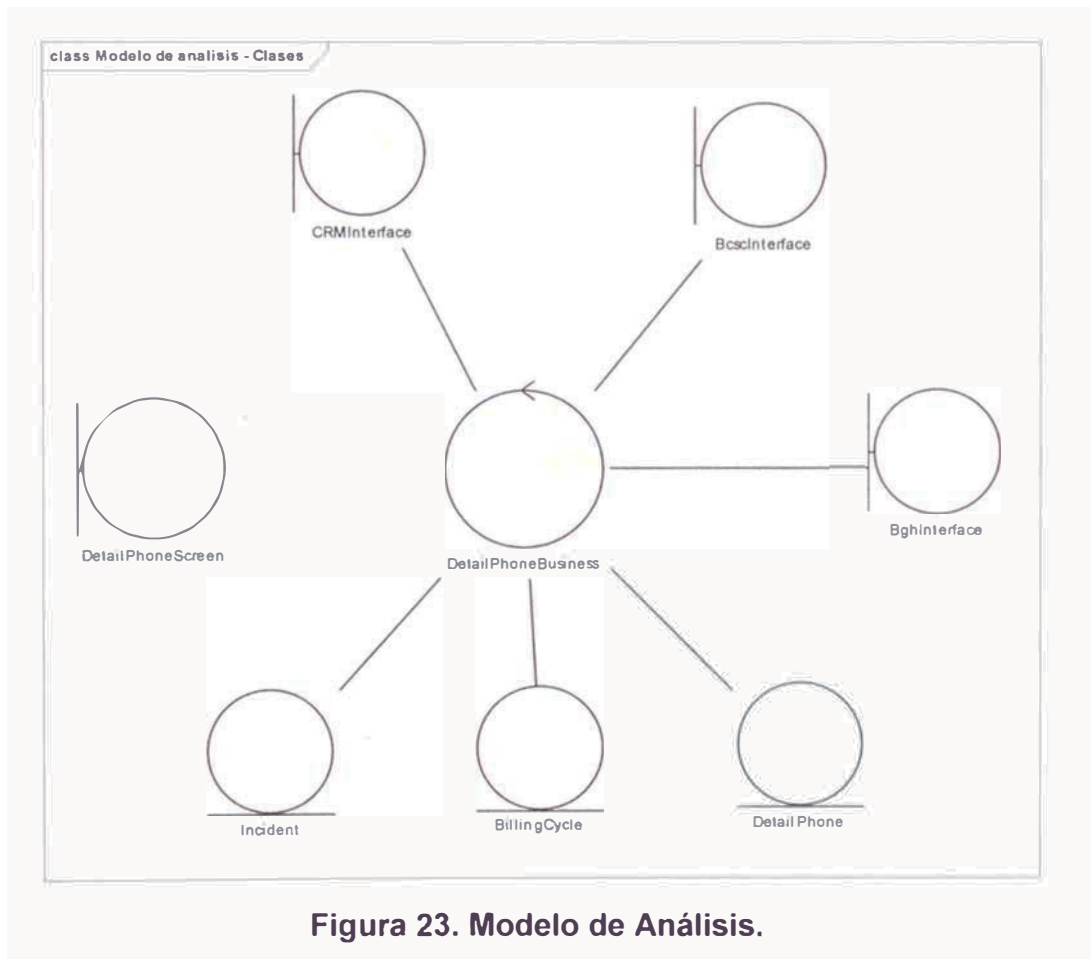


Figura 23. Modelo de Análisis.

Fuente: Telecom.

El siguiente diagrama muestra el flujo cronológico de eventos:

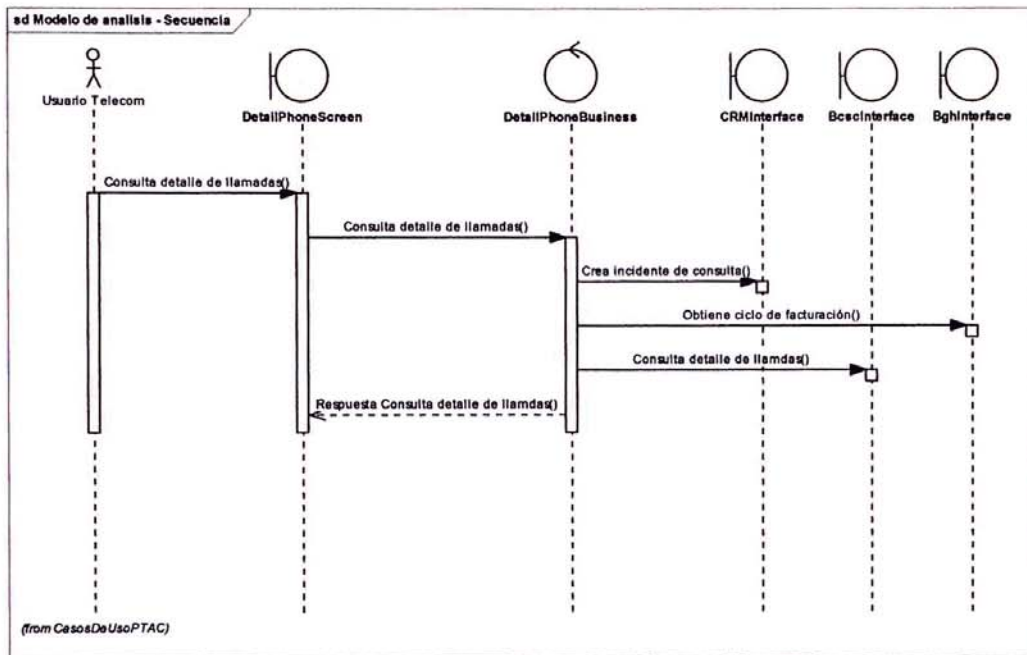


Figura 24. Diagrama de Secuencia.

Fuente: Telecom.

3.6.4. Diseño del Servicio

A continuación se muestra el diagrama de clases del diseño del servicio Detalle de llamadas.

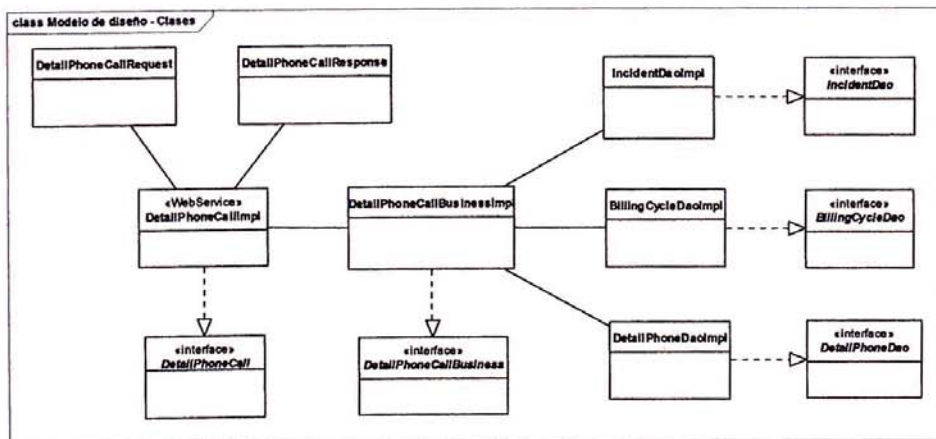


Figura 25. Diagrama de Clases.

Fuente: Telecom.

3.6.5. Diagrama de Componentes

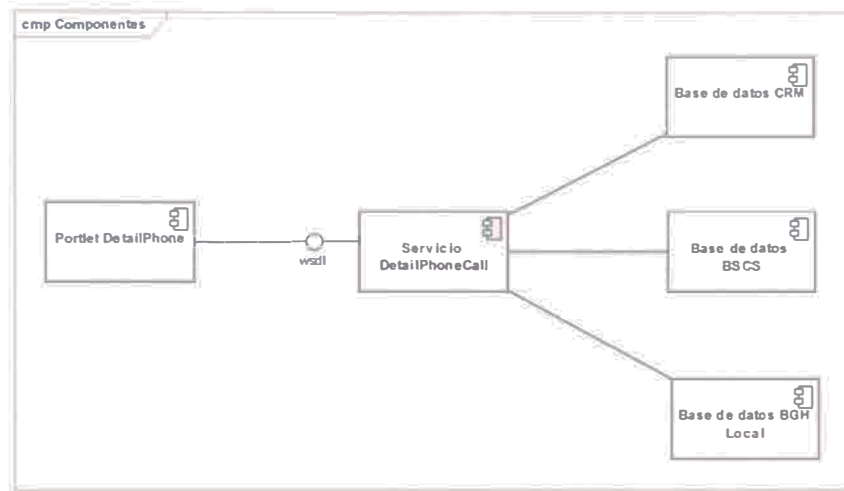


Figura 26. Diagrama de Componentes

Fuente: Telecom.

Para ver detalles de la implementación revisar el Anexo 1.

CAPÍTULO IV

ANÁLISIS DE RESULTADOS

Los siguientes datos han sido obtenidos en base al resultado del uso de los Servicios Web en los años siguientes a su implementación. En el Anexo 2 se puede apreciar los Servicios Web desplegados en un entorno productivo.

Para este análisis se utilizará tres variables que nos ayudarán a medir el beneficio que se generó con la solución planteada.

4.1. COSTO DE PRODUCCIÓN

Para el calcular el costo de producción se tomaron en cuenta los siguientes datos.

Horas invertidas en el desarrollo de interfaces con bus de servicios (en horas):

Complejo	303
Moderado	58
Simple	32

Horas invertidas en el desarrollo de interfaces con integración punto a punto (en horas):

Complejo	522
Moderado	86
Simple	43

Costo por hora de un programador: \$8

Resultados:

	Año 1			Año 2			Año 3		
	Complejo	Moderado	Simple	Complejo	Moderado	Simple	Complejo	Moderado	Simple
# Interfaces PP	2	22	36	4	29	43	8	36	47
Esfuerzo PP(\$)	\$8,352	\$15,136	\$12,384	\$16,704	\$19,952	\$14,792	\$33,408	\$24,768	\$16,168
# Interfaces Bus	2	22	36	3	23	31	6	26	34
Esfuerzo bus (\$)	\$4,848	\$10,208	\$9,216	\$7,272	\$10,672	\$7,936	\$14,544	\$12,064	\$8,704

Tabla 5. Resultado Costo de Producción.

Total PP: \$161,664

Total Bus: \$85,464

4.2. COSTO DE MANTENIMIENTO

Para el calcular el costo de mantenimiento se tomaron en cuenta los siguientes datos:

Horas invertidas en el mantenimiento de interfaces con bus de servicios (en horas):

Complejo	50
Moderado	10
Simple	5

Horas invertidas en el mantenimiento de interfaces con integración punto a punto (en horas):

Complejo	150
Moderado	25
Simple	15

Costo por hora de un programador: \$8

	Año 1			Año 2			Año 3		
	Complejo	Moderado	Simple	Complejo	Moderado	Simple	Complejo	Moderado	Simple
# Interfaces PP	2	22	36	4	29	43	8	36	47
Costo mant. PP(\$)	\$2,400	\$4,400	\$4,320	\$4,800	\$5,800	\$5,160	\$9,600	\$7,200	\$5,640
# Interfaces PP	2	22	36	3	23	31	6	26	34
Costo mant. Bus (\$)	\$800	\$1,760	\$1,440	\$1,200	\$1,840	\$1,240	\$2,400	\$2,080	\$1,360

Tabla 6. Resultado Costo de Mantenimiento.

Total PP: \$49,320

Total Bus: \$14,120

Al final del análisis se obtienen los siguientes costos para cada alternativa:

	Costo Producción	Costo Mantenimiento	Costo Implementación	COSTO TOTAL
Punto a Punto	\$161,664	\$49,320	\$36,000	\$246,984
BUS	\$85,464	\$14,120	\$92,000	\$191,584

Tabla 7. Resultado Final.

El ahorro total, en un periodo de tres años, con la implementación del bus de servicios es: \$55,400.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

- Uno de los principales errores al momento de implementar una arquitectura orientada a servicios es pensar que todo gira alrededor de los servicios web, esto nos lleva a crear servicios de manera indiscriminada hasta el punto que se hacen inmanejables y no se logra uno de los pilares de esta arquitectura que es la reutilización. Es muy importante tener el gobierno de de los servicios.
- La interoperabilidad de las aplicaciones en Telecom se mejoró enormemente, muchos procesos comunes son reutilizados en los diversos aplicativos Front-End de la empresa.
- Los costos de mantenimiento bajaron en 47%. Al no tener que implementar adaptadores en cada aplicación y tener todo en un entorno centralizado y configurable, el mantenimiento se hizo muy sencillo.
- La Arquitectura Orientada a Servicios es muy eficiente en entornos empresariales donde generalmente se tienen aplicaciones en diversas plataformas implementadas en diferentes lenguajes de programación, es la mejor decisión si lo que se quiere es integrar las aplicaciones empresariales.
- La implementación del enfoque SOA no se puede llevar a cabo sin el apoyo de la alta gerencia, es muy importante convencerlos de que esta solución les traerá muchos beneficios.

RECOMENDACIONES

- Se debe realizar un análisis para determinar si es necesario implementar el resto de capas que propone SOA, como por ejemplo la capa de componentes de servicio.
- Definir estándares de diseño de creación de servicios.
- Definir lineamientos para tomar decisiones arquitectónicas, respecto a cuándo es recomendable crear un servicio y cuando no.
- Se debe realizar capacitaciones sobre las nuevas tecnologías implementadas, sobre todo a las personas que van a dar mantenimientos a las mismas.
- Para aplicar todas las recomendaciones mencionadas es necesario crear un área o grupo separado que solo se encargue de temas relacionados a SOA.
- Por último es recomendable implementar un mecanismo para que el bus de servicios tenga alta disponibilidad, esto es muy importante pues al ser el único punto de integración, una falla podría ocasionar el colapso de muchas aplicaciones.

GLOSARIO DE TÉRMINOS

Enterprise Service Bus:

En informática un bus de servicios de empresa consiste en un combinado de arquitectura de software que proporciona servicios fundamentales para arquitecturas complejas a través de un sistema de mensajes basado en las normas y que responde a eventos.

Stakeholders:

Es un término que se utiliza para referirse a quienes pueden afectar o son afectados por las actividades de una empresa.

Servicio Web:

Es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones.

Dblink:

Es una definición de cómo establecer una conexión de una base de datos Oracle a otra.

API:

Interfaz de programación de aplicaciones (Application Programming Interface), es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Acuerdos de nivel de servicio:

Es un contrato escrito entre un proveedor de servicio y su cliente con objeto de fijar el nivel acordado para la calidad de dicho servicio.

KPI:

Indicadores Clave de desempeño (Key Performance Indicators), miden el nivel de desempeño de un proceso, enfocándose en el “cómo” e indicando el rendimiento de los procesos, de forma que se pueda alcanzar el objetivo fijado.

IDEN:

Red Mejorada Digital Integral (Integrated Digital Enhanced Network) es una tecnología inalámbrica desarrollada por Motorola en 1994, proporciona a los usuarios múltiples servicios en un único e integrado sistema de comunicaciones móviles.

3G:

Es la abreviación de tercera generación de transmisión de voz y datos a través de telefonía móvil mediante UMTS (Universal Mobile Telecommunications System o servicio universal de telecomunicaciones móviles)

BIBLIOGRAFÍA

- Architecture Building Block Definition, The Open Group Architecture Framework Version 9, <http://www.opengroup.org/architecture/togaf9-doc/arch/chap03.html>, Definitions, 2009.
- Ali Arsanjani, Service-oriented modeling and architecture: How to Identify, Specify and Realize your Services, IBM developerWorks, (<http://www.128.ibm.com/developerworks/webservices/library/ws-soa-design1/>).
- Gartner Defines Enterprise Information Architecture. By David Newman, Nicholas Gall, Anne Lapkin, Febrero 2008.
- SOA Reference Architecture. The Open Group, Abril 2009.
- Oracle SOA Suite Developer's Guide by Matt Wright, Antony Reynolds Published by Packt Publishing Ltd.
- Understanding Service-Oriented Architecture. By David Sprott and Lawrence Wilkes, January 2004.

Anexo 1 - Implementación del Servicio Web

Todos los servicios Web se implementaron utilizando el entorno de desarrollo Eclipse. En la siguiente imagen se puede apreciar la estructura del proyecto, así como los archivos de configuración del mismo.

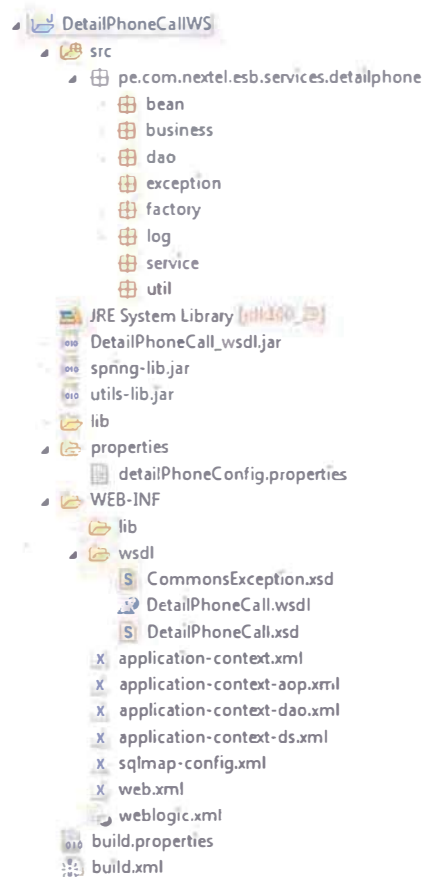


Figura 27. Proyecto en Eclipse

Fuente: Telecom.

A continuación se detallan los principales componentes del proyecto.

Interface WSDL DetailPhoneCall.wsdl:

```
<?xml version="1.0" encoding="UTF-8"?>
<wSDL:definitions xmlns:apachesoap="http://xml.apache.org/xml-soap"
  xmlns:wSDLsoap="http://schemas.xmlsoap.org/wSDL/soap/"
  xmlns:wSDL="http://schemas.xmlsoap.org/wSDL/"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:det="http://telecom.com.pe/eai/ebs/xsd/DetailPhoneCall"
  xmlns:ex="http://telecom.com.pe/eai/ebs/xsd/CommonsException"
  xmlns:tns="http://telecom.com.pe/eai/ebs/ws/DetailPhoneCall"
  targetNamespace="http://telecom.com.pe/eai/ebs/ws/DetailPhoneCall">
  <wSDL:types>
    <wSDL:documentation>
      <description>
        Descripción del servicio: Servicio que entrega
        información detallada de llamadas
        facturadas y no facturadas
        Version:
        1.0
      </description>
    </wSDL:documentation>
    <xs:schema>
      <xs:import schemaLocation="DetailPhoneCall.xsd"
        namespace="http://telecom.com.pe/eai/ebs/xsd/DetailPhoneCall"/>
      <xs:import schemaLocation="CommonsException.xsd"
        namespace="http://telecom.com.pe/eai/ebs/xsd/CommonsException"/>
    </xs:schema>
  </wSDL:types>
  <wSDL:message name="DetailPhoneCallRequest">
    <wSDL:part name="body" element="det:DetailPhoneCallRequest"/>
  </wSDL:message>
  <wSDL:message name="DetailPhoneCallResponse">
    <wSDL:part name="body" element="det:DetailPhoneCallResponse"/>
  </wSDL:message>
  <wSDL:message name="IntegrationException">
    <wSDL:part name="fault" element="ex:IntegrationError"/>
  </wSDL:message>
  <wSDL:portType name="DetailPhoneCall">
    <wSDL:operation name="getDetailPhoneCall">
      <wSDL:documentation>
        <description>Operacion que permite obtener el detalle de
        llamadas</description>
      </wSDL:documentation>
      <wSDL:input message="tns:DetailPhoneCallRequest"/>
      <wSDL:output message="tns:DetailPhoneCallResponse"/>
      <wSDL:fault message="tns:IntegrationException"
        name="IntegrationException"/>
    </wSDL:operation>
  </wSDL:portType>
  <wSDL:binding name="DetailPhoneCallSoapBinding" type="tns:DetailPhoneCall">
    <wSDLsoap:binding transport="http://schemas.xmlsoap.org/soap/http"
      style="document"/>
    <wSDL:operation name="getDetailPhoneCall">
```

```

        <wsdlsoap:operation
soapAction="http://telecom.com.pe/eai/ebs/ws/DetailPhoneCall/getDetailPhoneCall"/>
        <wsdl:input>
            <wsdlsoap:body use="Literal"/>
        </wsdl:input>
        <wsdl:output>
            <wsdlsoap:body use="Literal"/>
        </wsdl:output>
        <wsdl:fault name="IntegrationException">
            <wsdlsoap:fault name="IntegrationException" use="Literal"/>
        </wsdl:fault>
    </wsdl:operation>
</wsdl:binding>

    <wsdl:service name="DetailPhoneCallService">
        <wsdl:port name="DetailPhoneCallPort"
binding="tns:DetailPhoneCallSoapBinding">
            <wsdlsoap:address

location="http://172.19.24.39:7011/DetailPhoneCallWS/DetailPhoneCallService"/>
        </wsdl:port>
    </wsdl:service>

</wsdl:definitions>

```

Esquema con los objetos request y response.

DetailPhoneCall.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema version="1.0" elementFormDefault="qualified"
targetNamespace="http://telecom.com.pe/eai/ebs/xsd/DetailPhoneCall"
xmlns:tns="http://telecom.com.pe/eai/ebs/xsd/DetailPhoneCall"
xmlns:xs="http://www.w3.org/2001/XMLSchema">

    <xs:element name="DetailPhoneCallRequest" type="tns:DetailPhoneCallRequest" />
    <xs:element name="DetailPhoneCallResponse" type="tns:DetailPhoneCallResponse" />

    <xs:complexType name="DetailPhoneCallRequest">
        <xs:sequence>
            <xs:element name="idTransaction" type="xs:string"
minOccurs="1" />
            <xs:element name="phone" type="xs:string" minOccurs="1" />
            <xs:element name="invoiceNumber" type="xs:string"
minOccurs="0" />
            <xs:element name="flagCall" type="xs:boolean" minOccurs="0" />
            <xs:element name="typeClient" type="xs:string" minOccurs="0" />
        </xs:sequence>
    </xs:complexType>

    <xs:complexType name="DetailPhoneCallResponse">
        <xs:sequence>
            <xs:element name="idTransaction" type="xs:string"
minOccurs="1" />
            <xs:element name="listDetailPhoneCall" type="tns:DetailPhoneCall"
minOccurs="0" maxOccurs="unbounded" />
        </xs:sequence>
    </xs:complexType>

    <xs:complexType name="DetailPhoneCall">

```

```

        <xs:sequence>
            <xs:element name="phone" type="xs:string" minOccurs="0" />
            <xs:element name="date" type="xs:date" minOccurs="0" />
            <xs:element name="destinationPhone" type="xs:string"
                minOccurs="0" />
            <xs:element name="consumption" type="xs:string" minOccurs="0" />
            <xs:element name="originalCharge" type="xs:string"
                minOccurs="0" />
            <xs:element name="callType" type="xs:string" minOccurs="0" />
            <xs:element name="destination" type="xs:string" minOccurs="0" />
            <xs:element name="operation" type="xs:string" minOccurs="0" />
            <xs:element name="callDate" type="xs:date" minOccurs="0" />
            <xs:element name="serviceType" type="xs:string" minOccurs="0" />
            <xs:element name="hour" type="xs:string" minOccurs="0" />
        </xs:sequence>
    </xs:complexType>
</xs:schema>

```

Clase java con las anotaciones JAX-WS:

```

package pe.com.telecom.esb.services.detailphone.service;

import javax.jws.WebService;
import javax.xml.ws.BindingType;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.context.support.SpringBeanAutowiringSupport;

import pe.com.telecom.esb.services.detailphone.business.DetailPhoneCallInterface;
import pe.com.telecom.esb.services.detailphone.factory.FaultFactory;

/**
 * This class was generated by the JAX-WS RI. Oracle JAX-WS 2.1.5 Generated
 * source version: 2.1
 *
 */
@WebService(portName = "DetailPhoneCallPort", serviceName = "DetailPhoneCallService", targetNamespace =
"http://telecom.com.pe/eai/ebs/ws/DetailPhoneCallInterface", wsdlLocation = "/wsdls/DetailPhoneCallInterface.wsdl",
endpointInterface = "pe.com.telecom.esb.services.detailphone.service.DetailPhoneCall")
@BindingType("http://schemas.xmlsoap.org/wsdl/soap/http")
public class DetailPhoneCallService_DetailPhoneCallPortImpl extends
    SpringBeanAutowiringSupport implements DetailPhoneCall {

    @Autowired
    private DetailPhoneCallInterface detailPhoneCall;

```

```

public DetailPhoneCallService_DetailPhoneCallPortImpl() {
}
/**
 *
 * @param body
 * @return returns
 *     pe.com.telecom.esb.services.detailphone.service.DetailPhoneCallResponse
 * @throws IntegrationException
 */
public DetailPhoneCallResponse getDetailPhoneCall(
    DetailPhoneCallRequest body) throws IntegrationException {
    try {
        return detailPhoneCall.getDetailPhoneCall(body);
    } catch (Exception ex) {
        (new FaultFactory()).getException(ex);
    }
    return null;
}
}
}

```

Archivos de despliegue

web.xml:

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://java.sun.com/xml/ns/javaee"
    xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
    xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
    http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
    id="WebApp_ID" version="2.5">
    <display-name>OrdenServicioTecnicoWS</display-name>

    <context-param>
        <param-name>contextConfigLocation</param-name>
        <param-value>
            /WEB-INF/application-context.xml
            /WEB-INF/application-context-dao.xml
            /WEB-INF/application-context-ds.xml
            /WEB-INF/application-context-aop.xml
        </param-value>
    </context-param>

    <listener>
        <listener-
class>org.springframework.web.context.ContextLoaderListener</listener-class>
    </listener>

    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
    </welcome-file-list>

```

```

        <welcome-file>index.htm</welcome-file>
        <welcome-file>index.jsp</welcome-file>
        <welcome-file>default.html</welcome-file>
        <welcome-file>default.htm</welcome-file>
        <welcome-file>default.jsp</welcome-file>
    </welcome-file-list>
</web-app>

```

weblogic.xml:

```

<?xml version="1.0" encoding="UTF-8"?>
<wls:weblogic-web-app
    xmlns:wls="http://xmlns.oracle.com/weblogic/weblogic-web-app"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd http://xmlns.oracle.com/weblogic/weblogic-
web-app http://xmlns.oracle.com/weblogic/weblogic-web-app/1.3/weblogic-web-app.xsd">
    <wls:weblogic-version>10.3.6</wls:weblogic-version>
    <wls:context-root>DetailPhoneCallWS</wls:context-root>
    <wls:library-ref>
        <wls:library-name>spring-lib</wls:library-name>
        <wls:specification-version>2.0</wls:specification-version>
        <wls:exact-match>true</wls:exact-match>
    </wls:library-ref>
    <wls:library-ref>
        <wls:library-name>utils-lib</wls:library-name>
        <wls:specification-version>2.0</wls:specification-version>
        <wls:exact-match>true</wls:exact-match>
    </wls:library-ref>
</wls:weblogic-web-app>

```

Archivo XML que utiliza la herramienta Apache Ant para realizar diversas tareas, como el despliegue y la compilación de las clases java del servicio.

build.xml:

```

<?xml version="1.0"?>
<!-- define the project -->
<project name="DetailPhoneCallWS" default="build" basedir=".">

    <property name="projectname" value="DetailPhoneCallWS" />

    <!-- define project properties -->
    <property name="compiler" value="modern" />
    <property name="fork" value="no" />
    <property name="verbose" value="no" />
    <property name="debug" value="on" />
    <property name="optimize" value="on" />
    <property name="deprecation" value="on" />
    <property name="target" value="1.5" />
    <property name="source" value="1.5" />
    <property name="includeantruntime" value="false" />
    <property file="build.properties" />
    <property environment="env" />

    <!-- define properties to refer to directories in the project -->

```

```

<property name="webinf.dir" value="WEB-INF" />
<property name="webinf.lib.dir" value="WEB-INF/lib" />
<property name="lib.dir" value="lib" />
<property name="src.dir" value="src" />
<property name="build.dir" value="build" />
<property name="webinf.classes.dir" value="${webinf.dir}/classes" />
<property name="web.xml" value="${webinf.dir}/web.xml" />
<property name="properties.dir" value="properties"/>

<!-- define generate-from-wsdl properties -->
<property name="wsdlc.packageName"
value="pe.com.telecom.esb.services.detailphone.service" />
<property name="wsdlc.destImplDir" value="src" />
<property name="wsdlc.destJwsDir" value="WEB-INF/lib" />
<property name="wsdlc.srcWsdl" value="WEB-INF/wsdl/DetailPhoneCall.wsdl" />

<!-- define generate-client-from-wsdl properties -->
<property name="clientgen.wsdl" value="WEB-INF/wsdl/DetailPhoneCall.wsdl" />
<property name="clientgen.destfile" value="build/DetailPhoneCallWSDLClient.jar" />

<fileset id="webapp.libs" dir="${webinf.lib.dir}">
  <include name="*.jar" />
</fileset>

<fileset id="libs" dir="${lib.dir}">
  <include name="*.jar" />
</fileset>

<path id="class.path">
  <pathelement path="${webinf.classes.dir}" />
  <fileset refid="webapp.libs" />
  <fileset refid="libs" />
</path>

<path id="classpathweblogic">
  <pathelement location="${oracle.server.lib}\wsse.jar" />
  <pathelement location="${oracle.server.lib}\weblogic.jar" />
</path>

<pathconvert pathsep=":" property="class.path" refid="class.path" />

<fileset id="war.files" dir=".">
  <include name="${webinf.dir}/**" />
  <exclude name="${web.xml}" />
</fileset>

<taskdef name="wldeploy" classname="weblogic.ant.taskdefs.management.WLDeploy">
  <classpath refid="classpathweblogic" />
</taskdef>

<taskdef name="wsdlc" classname="weblogic.wsee.tools.anttasks.WsdlcTask">
  <classpath refid="classpathweblogic" />
</taskdef>

<taskdef name="clientgen" classname="weblogic.wsee.tools.anttasks.ClientGenTask">
  <classpath refid="classpathweblogic" />
</taskdef>

<target name="generate-from-wsdl">
  <wsdlc type="JAXWS"
    packageName="${wsdlc.packageName}"
    destImplDir="${wsdlc.destImplDir}"
    destJwsDir="${wsdlc.destJwsDir}"
    srcWsdl="${wsdlc.srcWsdl}"/>
</target>

<target name="generate-client-from-wsdl">
  <clientgen type="JAXWS"

```

```

        wsdl="${clientgen.wsdl}"
        destfile="${clientgen.destfile}"
    </clientgen>
</target>

<!-- target to clean up all files created by various tasks -->
<target name="clean">
    <delete quiet="true" includeemptydirs="true">
        <fileset dir="${webinf.classes.dir}" includes="**/*" />
        <fileset dir="${build.dir}" />
        <fileset dir="${work.dir}" />
    </delete>
</target>

<!-- compile target to compile the sources -->
<target name="compile">
    <mkdir dir="${webinf.classes.dir}" />
    <javac srcdir="${src.dir}" destdir="${webinf.classes.dir}" fork="${fork}"
verbose="${verbose}" deprecation="${deprecation}" debug="${debug}" optimize="${optimize}"
compiler="${compiler}" target="${target}" source="${source}"
includeantruntime="${includeantruntime}">
        <classpath refid="class.path" />
    </javac>
    <copy todir="${webinf.classes.dir}" preservelastmodified="true">
        <fileset dir="${src.dir}">
            <include name="**/*.properties" />
            <include name="**/*.xml" />
        </fileset>
    </copy>
</target>

<!-- target to create the project WAR file -->
<target name="build" depends="clean,compile">
    <mkdir dir="${build.dir}" />
    <war destfile="${build.dir}/${projectname}.war" webxml="${web.xml}">
        <fileset refid="war.files" />
    </war>
</target>

<!-- target to deploy war files to weblogic server -->
<target name="deploy-to-weblogic" depends="build">
    <scp todir="${host.user}:${host.password}@${host.ip}:${host.propertiesDir}"
trust="true">
        <fileset dir="${properties.dir}">
            <include name="**/*.properties" />
        </fileset>
    </scp>
    <wldeploy action="deploy" verbose="true" debug="true" name="${projectname}"
source="${build.dir}/${projectname}.war" user="${weblogic.admin.user}"
password="${weblogic.admin.password}" adminurl="${weblogic.admin.url}"
targets="${weblogic.targets}" failonerror="true" upload="true" />
</target>

<!-- target to deploy war files to weblogic local -->
<target name="deploy-to-local" depends="build">
    <wldeploy action="deploy" verbose="true" debug="true" name="${projectname}"
source="${build.dir}/${projectname}.war" user="${weblogic.local.user}"
password="${weblogic.local.password}" adminurl="${weblogic.local.url}"
targets="${weblogic.local.targets}" failonerror="true" upload="true" />
</target>

</project>

```

Configuración del servicio en el Oracle Service Bus.

Proyecto en Eclipse:

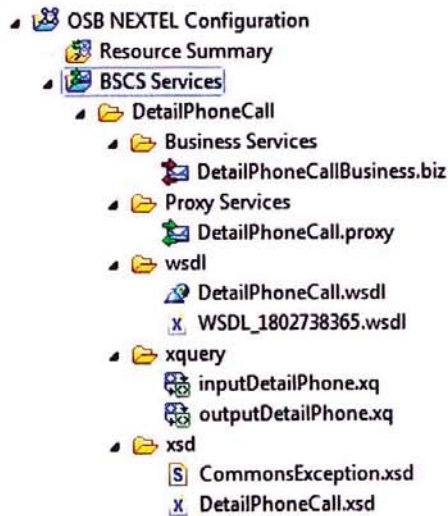


Figura 28. Proyecto OSB en Eclipse.

Fuente: Telecom.

Transformación de las entradas del proxy a entradas del Business Service

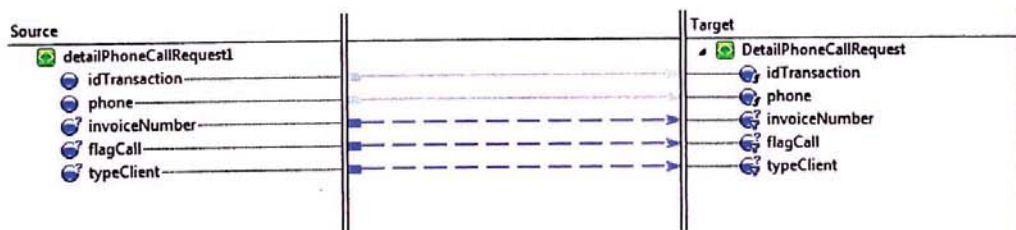


Figura 29. Transformación en OSB.

Fuente: Telecom.

Código del archivo de transformación:

```
(:: pragma bea:global-element-parameter  
parameter="$detailPhoneCallRequest1" element="ns0:DetailPhoneCallRequest"  
location="../xsd/DetailPhoneCall.xsd" ::)  
(:: pragma bea:global-element-return element="ns0:DetailPhoneCallRequest"  
location="../xsd/DetailPhoneCall.xsd" ::)
```

```
declare namespace ns0 =  
"http://telecom.com.pe/eai/ebs/xsd/DetailPhoneCall";
```



```

declare namespace xf =
"http://tempuri.org/BSCS%20Services/DetailPhoneCall/xquery/inputDetailPhone/";

declare function xf:inputDetailPhone($detailPhoneCallRequest1 as
element(ns0:DetailPhoneCallRequest))
  as element(ns0:DetailPhoneCallRequest) {
  <ns0:DetailPhoneCallRequest>
    <ns0:idTransaction>{
data($detailPhoneCallRequest1/ns0:idTransaction) }</ns0:idTransaction>
    <ns0:phone>{ data($detailPhoneCallRequest1/ns0:phone)
}</ns0:phone>
    {
      for $invoiceNumber in
$detailPhoneCallRequest1/ns0:invoiceNumber
      return
        <ns0:invoiceNumber>{ data($invoiceNumber)
}</ns0:invoiceNumber>
    }
    {
      for $flagCall in $detailPhoneCallRequest1/ns0:flagCall
      return
        <ns0:flagCall>{ data($flagCall) }</ns0:flagCall>
    }
    {
      for $typeClient in $detailPhoneCallRequest1/ns0:typeClient
      return
        <ns0:typeClient>{ data($typeClient) }</ns0:typeClient>
    }
  }
  </ns0:DetailPhoneCallRequest>
};

declare variable $detailPhoneCallRequest1 as
element(ns0:DetailPhoneCallRequest) external;

xf:inputDetailPhone($detailPhoneCallRequest1)

```

Anexo 2 - Ejecución de un servicio Web

A continuación se muestran imágenes del Servicio Web desplegado en el servidor de aplicaciones Weblogic:

<input type="checkbox"/>	DetailPhoneCallWS	Activo	✓ OK	Aplicación Web	100
<input type="checkbox"/>	Servicios Web				
<input type="checkbox"/>	DetailPhoneCallService			Servicio Web	

Figura 30. Servicio Web desplegado 1.

Fuente: Telecom.

Como se puede apreciar el servicio está activo y listo para que algún cliente lo utilice.

Valores para DetailPhoneCallService

Visión General Configuración Seguridad **Prueba** Supervisión

Utilice esta página para probar que el servicio web está desplegado y funciona según lo esperado. En la tabla, amplíe el nombre del servicio web para ver el WSDL dinámico en una ventana de explorador independiente. Haga clic en **Probar Cliente** para abrir una nueva ventana de explorador introduciendo los valores de los parámetros, ejecutando las operaciones y viendo los resultados.

Pruebas de Despliegue


Nombre 	Punto de Prueba	Comentarios
<input type="checkbox"/> DetailPhoneCallService		Test points for this
/DetailPhoneCallWS/DetailPhoneCallService	?WSDL	WSDL page on ser
/DetailPhoneCallWS/DetailPhoneCallService	Test client	Test client on serv

Figura 31. Servicio Web desplegado 2.

Fuente: Telecom.

Ejecución del Servicio Web desde la consola del servidor de aplicaciones Weblogic.

Parámetros de entrada del servicio:



Figura 32. Prueba del Servicio Web 1.
Fuente: Telecom.

Respuesta del Servicio Web:



Figura 33. Prueba del Servicio Web 2.
Fuente: Telecom.

- Front End: Desarrollo de la interface usuario, se utilizo el Web Center de Oracle para implementar el portal WEB.
- Back End: Desarrollo de los procesos internos a través de servicios web, se utilizo el Oracle Service Bus para la integración de servicios web.

A modo de resumen se detalla todo el proceso de desarrollo de un requerimiento en la parte Back End, por lo que se presentara las interfaces de los servicios más no interfaces usuario.

3.6.1. Requerimiento Funcional de ejemplo

Permitir al cliente que consulte su detalle de llamadas de Interconexión Telefónica

Se requiere que un cliente de Telecom pueda consultar el detalle de sus llamadas de interconexión telefónica para el ciclo de facturación actual o los últimos 4 ciclos.

Sobre la autenticación:

Para poder utilizar esta funcionalidad del Web Portal, es necesario que el usuario sea un cliente de Telecom y que haya ingresado su clave de registro.

Sobre la consulta:

Por defecto se le debe mostrar al cliente un resumen de consumo por día del ciclo actual y así como también el detalle de llamadas por cada día. Los campos a mostrar son los mismos que se muestran actualmente en la web de Telecom.