

UNIVERSIDAD IBEROAMERICANA

Estudios con Reconocimiento de Validez Oficial por Decreto Presidencial
del 3 de abril de 1981



ARQUITECTURA DE NEGOCIOS ORIENTADA A SERVICIOS (SOBA-UIA)

TESIS

Que para obtener el grado de

MAESTRA EN INGENIERÍA DE SISTEMAS EMPRESARIALES

Presenta:

EDITH HERNÁNDEZ HERNÁNDEZ

Director: Dr. Alfonso Miguel Reyes

Lectores: Dr. Carlos Villegas Quezada

Mtro. Pedro Solares Soto

México, D.F.

2009

Resumen

Esta investigación considera la dificultad que existe en las áreas de negocio para realizar cambios o cubrir necesidades derivadas del entorno global y las soluciones que sus áreas tecnológicas han desarrollado por años con sistemas inflexibles, con una dificultad muy alta para la integración entre sistemas.

Algunas referencias proponen la incorporación de tecnologías recientes como BPM y SOA a modelar la arquitectura de negocios que ayude a las organizaciones a realizar cambios de forma rápida lo que representa flexibilidad, no obstante no existe la trazabilidad entre el modelo de negocio, modelo de servicios y el modelo computacional. En esta investigación se presenta un modelo unificado para la creación de una arquitectura de negocio orientada a servicios que aplica principios, del modelado de BPM, SOA, RM-ODP y el framework Zachman. El modelo se compone con cinco arquitecturas o vistas (negocio, información, computacional, ingeniería y tecnología) similares a RM-ODP, pero introduce el caso de los servicios básicos para el modelado de negocio en servicios y la arquitectura computacional.

El análisis de los procesos de negocios con BPM y la optimización de los mismos lleva a encontrar los componentes en vueltos en una metodología tecnológica los pasos para construir arquitecturas de negocios con las tecnología alienada a los objetivos del mismo.

La hipótesis principal es que existen componentes en las arquitecturas de negocio que permiten flexibilidad, reusabilidad, integración y agilidad al negocio de forma natural sin requerir proyectos costosos para tales fines.

La investigación se divide en cuatro capítulos:

Capítulo 1 Se expone cual será la metodología de investigación para la realización de este trabajo, la investigación es exploratoria y descriptiva ya que explica como llevar acabo la propuesta de la arquitectura SOBA-UIA.

Capítulo 2. Es el Marco Teórico sobre lo que hoy existe en metodologías y tecnologías como han evolucionando y expone el por que es necesario evolucionar hacia otras soluciones para el negocio y entender la terminología en conceptos importantes que serán utilizados en la nueva propuesta y a lo largo de la investigación.

Capítulo 3. Derivado del análisis de las metodologías en el capítulo anterior y las fases que debe cubrir el análisis y diseño de SOBA-UIA, se explica que implicaciones tiene cada fase y cual es la información como resultado.

Capítulo 4. Se realiza la construcción de la arquitectura con la herramienta NetBeans, con el fin de demostrar que SOA no solo aplica como una metodología tecnológica si no que también es aplicable al negocio agregando algunos matices de procesos de negocios bajo el contexto de servicios web que permiten la flexibilidad, reusabilidad y la integración.

Agradecimientos

Esta investigación nace de mi solicitud al Dr. Alfonso Miguel Reyes, a quien agradezco haya compartido sus conocimientos y experiencia durante la Maestría de Ingeniería en Sistemas Empresariales impartida en la Universidad Iberoamericana y me apoyará a incursionar en un tema que actualmente esta a la vanguardia en Tecnologías de Información.

El desarrollo de esta tesis estuvo bajo su constante supervisión, agradezco a los lectores, el Dr. Carlos Villegas y el Mtro. Pedro Solares, por sus observaciones las cuales fueron primordiales para el término de este trabajo.

De la misma manera a mi familia por su apoyo para concluir mis estudios de postgrado, a Gustavo por su paciencia durante el período de investigación y revisión de la última versión de este trabajo; A mi amiga Marle González Mil gracias por todo su apoyo y bondad.

Índice

Introducción	1
Capítulo 1 Planteamiento General	3
1.1 Descripción	3
1.2 Antecedentes de la Investigación	3
1.2.1 Internet y la web	5
1.2.2 Los servicios Web	5
1.2.3 Arquitectura Orientada a Servicios.	7
1.2.3.1 Elementos de la Arquitectura:	11
1.3 Problema Real.	12
1.4 Justificación	13
1.5 Pregunta de Investigación	14
1.5.1 Hipótesis de Trabajo	14
1.6 Delimitación del alcance y Metodología de Diseño.	14
1.6.1 Alcance	14
1.6.2 Diseño de la Investigación.	15
1.7 Objetivo.	17
Capítulo 2 Marco Teórico.	18
2.1 Arquitectura Empresarial	21
2.2 Zachman Framework.....	23
2.3 RM-ODP (Reference Model Open Distributing Processing).	26
2.4 UML Lenguaje de Modelado Unificado	27
2.5 Modelado Conceptual	29
2.6 Modelado Lógico	29

2.7	Modelado de Procesos de Negocio. (BPM).....	30
2.7.1	Características de BPM básicas:	33
2.8	Estado del Arte de SOBA (Arquitectura de Negocios Orientada a Servicios).....	34
2.8.1	Definición de SOBA.....	35
2.8.1.1	Características.....	36
2.8.1.2	Importancia del nuevo Enfoque	37
2.8.1.3	Cuadro Comparativo SOA vs SOBA	37
2.8.1.4	Componentes esenciales en SOBA.	38
2.9	Modelos Existentes.	39
2.9.1	Framework Conceptual “Fusión”	39
2.9.2	SOMA Un método para desarrollar soluciones orientadas a servicios.	41
Capítulo 3	Modelo de Arquitectura de Negocios Orientada a Servicios en Acambio.	46
3.1	Modelo SOBA- UIA	49
3.1.1	Antecedentes del Modelo SOBA-UIA	49
3.1.2	Formalización del Modelo.	50
3.1.2.1	Arquitectura Empresarial.....	50
3.1.2.2	Arquitectura de servicios y Computacional.....	53
3.2	Aplicación del Modelo SOBA-UIA.....	55
3.2.1	Breve reseña de Acambio.....	55
3.2.2	Identificación de necesidades	55
3.2.3	Modelo Conceptual.	56
3.2.4	Modelo Lógico	58
3.2.5	Modelo Físico	64
3.2.5.1	Modelado empleando BPEL.....	65
3.2.5.2	Conceptos Centrales para el modelado del proceso de Negocio en BPEL.	69
Capítulo 4	Construcción de SOBA-UIA.....	73

4.1	Qué es NetBeans	73
4.2	Iniciar el servidor de aplicaciones GlassFish V2.	74
4.3	Creación de Servicios Web y WSDL.....	75
4.3.1	Lo beneficios de usar WSDL son:.....	78
4.4	Creación del módulo BPEL.	79
4.4.1	Creación de Partner Links.....	82
4.4.2	Crear una composite application	88
4.4.3	Crear una prueba.....	90
4.5	Análisis de los componentes de SOBA que permiten la flexibilidad e Integración.....	92
4.5.1	Comparativo de la arquitectura actual de Acambio y SOBA.	93
4.5.2	Implicaciones con el Modelo SOBA.	93
	Conclusión	95
	Referencias.....	98
	Anexo 1.....	100
	Anexo 2.....	102
	Glosario	105

Lista de Figuras

Figura 1. Elementos de SOA (Microsoft, 2003)	11
Figura 2. Conceptos base de BPM (Francis Brudenell, 2007).....	18
Figura 3. UML (Selda, 2006)	19
Figura 4. Arquitectura SOA (The Gilbane Report, 2003)	19
Figura 5. Modelo (Malinverno, 2007)	20
Figura 6. Framework de Zachman (Young, 2007)	25
Figura 7. Vistas de RM-ODP. (Delgado, 2007).....	27
Figura 8. Tipos de Diagrama UML	29
Figura 9. SOBA-UIA.....	46
Figura 10. Modelo formal.....	50
Figura 11. Estructura de la Arquitectura Empresarial.....	51
Figura 12. Arquitectura Computacional.....	53
Figura 13. Sistema Empresarial	56
Figura 14. Ejecución de procesos entre diferentes entidades.....	58
Figura 15. Casos de uso de Negocio.....	59
Figura 16. Diagrama de Actividad	60
Figura 17. Descomposición de procesos.....	61
Figura 18. Diagrama en BPM.....	62
Figura 19. Subproceso Verifica datos del Cliente.....	63
Figura 20. Transición del modelo conceptual y lógico al modelo físico.....	64
Figura 21. Composición BPEL. (Matjaz, 2006).....	67
Figura 22. Orquestación de servicios web	68
Figura 23. Código fuente del proceso principal de negocio.....	70
Figura 24. Código de Partner Link	70
Figura 25. Código del documento WSDL.....	71

Figura 26. Actividades Estructuradas	71
Figura 27. Propuesta de SOBA	72
Figura 28. GlassFish	75
Figura 29. Creación del módulo EJB en NetBeans.....	76
Figura 30. Nombre y configuración del módulo EJB	76
Figura 31. Generar servicio web.....	77
Figura 32. Agregar operación en el servicio web	77
Figura 33. Campos que formarán los datos de entrada.....	77
Figura 34. Vista del servicio web y operación	78
Figura 35. Creación del módulo BPEL.....	79
Figura 36. Selección de Categoría y Módulo BPEL	80
Figura 37. Determinar nombre del módulo BPEL	80
Figura 38. Creación del documento BPEL	81
Figura 39. Asignar nombre al proceso	81
Figura 40. Actividades Básicas y Estructuradas.....	82
Figura 41. Configuración de un servicio web - partner link	82
Figura 42. Creación del segundo partner link - receive.....	83
Figura 43. Documento xsd.....	83
Figura 44. Definición de variables de entrada y salida.....	84
Figura 45. Configuración Concreta.....	84
Figura 46. Partner link y servicio web a invocar.....	85
Figura 47. Proceso de negocio BPEL.....	85
Figura 48. Configuración Receive	86
Figura 49. Configuración Invoke.....	86
Figura 50. Creación de la composición de aplicaciones.....	88
Figura 51. Definir Nombre al módulo.....	88

Figura 52. Agregar un módulo JBI	89
Figura 53. Seleccionar modulo BPEL para la creación del archivo .jar.....	89
Figura 54. Compilación.....	89
Figura 55. Compilación.....	90
Figura 56. Resultado de la compilación	90
Figura 57. Crear un caso de prueba	91
Figura 58. Nombre del caso de prueba	91
Figura 59. Selección de la definición de servicios web y operación.....	91

Listas de Tablas

Tabla 1. Comparativo	38
Tabla 2. Procesos de Negocio.....	57
Tabla 3. Notación BPM.....	104
Tabla 4. Glosario.....	106

INTRODUCCION

Actualmente el problema más crítico en cualquier tipo de empresa (comercial, gobierno, etc) es realizar cambios rápidos y requeridos en sus procesos de negocio.

La necesidad de transformar un ambiente de negocios inflexible a una empresa más ágil, que pueda hacer cambios de dirección rápidamente hace que los negocios modifiquen su dirección en el futuro.

La solución requiere metodologías y tecnologías para realizar cambios de negocios rápidos, con sistemas que puedan cambiar por lo que esto, representa un problema de tecnología o computación. El origen del problema radica en el negocio y la necesidad de tener una dirección estratégica.

Históricamente, los sistemas han tenido dificultad en la realización de cambios. Los sistemas y bases de datos que se construían en años pasados se resistían al cambio, por lo que las organizaciones no realizaban cambios para competir en el entorno global sino para sobrevivir.

Se requiere construir arquitecturas flexibles, las cuales puedan cambiar fácil y rápidamente, con la dirección e interacción que marca el negocio, creando arquitecturas empresariales con métodos y tecnologías que le permitan la interacción e integración.

La integración y flexibilidad del negocio provee la arquitectura empresarial, la cual está basada en tecnologías como XML (Lenguaje de Mercado Extensible) y la integración de aplicaciones esta a cargo de EAI (Integración de Aplicaciones Empresariales), web services (SOA) y BPM (Modelado de Procesos de Negocio), generando automáticamente procesos de negocios y el flujo que han de seguir (workflow), permitiendo redireccionar hacia los servicios web que se requieran. Por lo tanto SOA es una solución tecnológica que puede ser también aplicada para negocios SOBA (Arquitectura de Negocios Orientada a Servicios).

Existe un punto de convergencia entre el negocio y la tecnología, este punto es la Internet y asociado a las tecnologías actuales, han dado como resultado arquitecturas, en donde, la tecnología no es la respuesta a la necesidad del negocio, la integración de la misma debe ser alineada bajo los objetivos del negocio en cualquier organización o empresa.

Capítulo 1 Planteamiento General

1.1 Descripción

La presente investigación introduce un nuevo modelo de arquitectura empresarial incorporando conceptos de nuevos paradigmas como lo es SOBA (Schreiter, 2006), que prometen el alineamiento del negocio con la tecnología a través de la flexibilidad y agilidad, permitiendo mayor colaboración e integración con entidades externas o internas, reutilizando componentes tecnológicos que facilitan la gestión de la tecnología de información.

El contexto de SOBA es un concepto que evoluciona a medida que madura la adopción de este tipo de metodologías, en las cuales se pueden incorporar otras como BPM (Administración de Procesos de Negocios), (Francis Brudenell, 2007), el Modelo RM-ODP (Reference Model Open Distributing Processing), Metodología de Zachman (Zachman, 2006) etc, de tal forma que obtener nuevos modelos generan valor en el arte de desarrollo del modelado de arquitecturas.

El desarrollo de toda la arquitectura pasa por las etapas tradicionales del ciclo de vida, iniciando con la recopilación de información, análisis, diseño, construcción, pruebas e implementación. Incorporando en cada fase técnicas y herramientas que faciliten el modelado conceptual hasta llegar al modelo físico de la arquitectura.

1.2 Antecedentes de la Investigación

Las arquitecturas distribuidas, aparecen en los años 80, de tal forma, que durante este período el marco que se intentaba definir, era integrar diversos elementos que posibilitaran una solución descentralizada (Pelechano, 2005), es decir, que las aplicaciones o requerimientos de procesamiento son satisfechos por múltiples computadoras físicas o entornos separados y los datos almacenados en muchas localidades físicas comunicados a través de la red. Los elementos que conducen a este modelo de arquitectura de procesamiento y almacenamiento descentralizado son:

- Costos: No solo eran inversiones iniciales, si no tener un punto de procesamiento y almacenamiento representaba un riesgo.
- Propiedad de Datos: Otro factor para impulsar la descentralización eran las políticas de dueños de datos. (departamentos, divisiones, localidades geográficas).

El concepto de funcionalidad distribuida trae la promesa de reutilización.

Dentro de la evolución de la arquitectura distribuida, la invocación a procedimientos remotos (RPC por sus siglas en inglés), constituye un avance en la llamada de una función que reside en un sistema remoto, sin embargo, se presentaron algunos problemas como (BEA Systems, 2005):

- Redundancia: es el primer problema que es inherente en RPC, es el descubrimiento, en donde la aplicación busca la información que es necesaria para conectar a otro punto, en donde el punto final emplea código duro.
- Interrupciones temporales del servidor debido a fallas con el mismo.
- Necesidad de más hardware que se requiere para una carga típica, manejo infrecuente de puntos de carga.

El siguiente modelo fue el denominado Arquitecturas basadas en Mensajes, es decir el middleware orientado a mensajes proporciona aplicaciones con servicio de comunicaciones interprocesos usando tecnología de mensajes (message queuing technology) para garantizar el nivel de servicio para aplicaciones críticas (Microsoft, 2003).

La característica más positiva de esta arquitectura es que es asíncrona, basado en intercambios de mensajes que llaman a la función:

- Los mensajes pueden ser ruteados en carga y prioridad.

- Las llamadas asíncronas permiten a los clientes hacer trabajo productivo mientras espera para consumir una operación.
- El manejo de flujos de trabajo (workflow) que son definidos como secuencia de mensajes, iniciando el intercambio con múltiples computadoras.

La existencia de problemas con los protocolos binarios como la interacción con firewalls, la interoperabilidad y los formatos de datos, en donde se implementa esta arquitectura distribuida, como el Distributed Component Object Model (DCOM), el Java Remote Method Invocation (JRMI) y el Common Object Request Broker Architecture (CORBA), originan la aparición del World Wide Web como una solución universal y el uso fácil de documentos (Microsoft, 2003).

1.2.1 Internet y la web

Como el Protocolo de Control de Transmisión (TCP) y Protocolo de Internet (IP) eran originalmente desarrollados para conectar diferentes redes, a finales de 1990 Tim Berners Lee invento el WWW (Microsoft, 2003). La Web es una red de interconexión global de documentos de hipertexto y surgen con ello dos tecnologías HTML (Hypertext Markup Language) y HTTP (Hypertext Transfer Protocol).

HTML es el lenguaje que define como agregar marcado a documentos de texto proveyendo información al browser y HTTP es el protocolo que se usa para solicitar y recibir documentos de hipertexto en la Internet.

1.2.2 Los servicios Web

La evolución acelerada de Internet, así como la del comercio electrónico, necesita de tecnologías de información que faciliten la conexión de información, sistemas, hardware y personas. Desde la aparición de la Web los procesos de negocios ofrecidos a través de esta han mejorando (Selda, 2006). En una primera fase se inicia el interés por el desarrollo de portales fomentando la presencia de las empresas; en una segunda fase, se implementa el comercio electrónico y las transacciones vía Internet (eCommerce); en una tercera fase se continua con la

economía digital, con el desarrollo e implementación de los negocios (eBusiness), las empresas (eEmpresarial) y los mercados electrónicos (eMarket).

Los mercados electrónicos constituyen la nueva generación de negocios digitales, proporcionando bienes y servicios de carácter electrónico que sustituyen a los tradicionales. Dichos servicios pueden integrarlos a otras empresas en sus modelos de negocio. Los servicios web constituyen un esfuerzo para construir plataformas distribuidas para la web que brinden el soporte adecuado, como mecanismo de implementación a la economía digital (Pelechano, 2005).

Los servicios web son la última evolución tecnológica desde el modelo clásico Cliente/Servidor, el middleware distribuido mediante sistemas basados en RPC, mensajes, pasando por el middleware de integración de aplicaciones empresariales (EAI), los sistemas de workflow y por último la tecnología web clásica con los sitios web o portales. La evolución debe permitir la interoperabilidad universal, amplia y rápida adopción de un soporte eficiente de entornos abiertos (BEA Systems, 2005), por lo que tener la visión completa es enfocarse a una arquitectura orientada a servicios, como un paradigma de diseño que desemboca en la creación de unidades aisladas de funcionalidad de negocio que están débilmente ligadas mediante la conformidad a marcos de comunicación estándar (Cuenca, Ortiz & García, 2005). Debido a la independencia de que gozan los servicios dentro del marco, la lógica de programación que encapsulan no requiere ajustarse a ninguna plataforma o conjunto tecnológico.

El tipo más ampliamente aceptado de servicio es el Servicio Web XML; la referencia mundial W3C lo define como un sistema de software identificado por una URI, cuyas interfaces públicas y enlaces son definidos usando XML, su definición puede ser descubierta por otros sistemas los cuales pueden interactuar con el servicio web a través de mensajes basados en XML y protocolos estándares de Internet (Pelechano, 2005).

XML es un lenguaje conocido como lenguaje de marcado extensible y algunos de sus objetivos son: fácil de crear y usar en Internet, plataforma independiente con

una rápida adopción como formato universal de datos. La contribución más significativa es el Servicio Web, el cual emplea protocolos de Internet, enviando o recibiendo datos formateados como documentos XML.

La amplia aceptación del modelo de diseño de servicios Web ha determinado la emergencia de un conjunto de tecnologías suplementarias que se han convertido en estándares de facto. Un servicio estándar de la industria se supone:

- Suministrará una descripción del servicio que, como mínimo, consiste en un documento WSDL (Web Services Description Language), que es un formato basado en XML (desarrollado por IBM y Microsoft) para describir de manera formal servicios Web.
- Será capaz de transportar documentos XML utilizando Simple Object Access Protocol por sus siglas en inglés SOAP, protocolo basado en XML y sobre HTTP que es usado para el intercambio de información.

Estas tecnologías no alteran la funcionalidad central de un servicio Web, manteniendo su habilidad para representarse a sí mismos y comunicarse de una forma estándar. Muchas de las convenciones arquitectónicas asumen que tanto SOAP como WSDL forman parte del marco definido de servicios Web.

1.2.3 Arquitectura Orientada a Servicios.

Los sistemas de información necesitan soporte para cambios rápidos y eficientes para el negocio, sin embargo, ellos también necesitan adaptar desarrollos rápidos de nuevas tecnologías. La mayoría de los sistemas de información empresariales son heterogéneos, contienen un rango de diferentes sistemas, aplicaciones, tecnologías y arquitectura. La integración de todas estas tecnologías es crucial para que puedan dar valor al negocio (BEA Systems, 2005), es así como la eficiente toma de decisiones para integrar sistemas de información disminuye el costo de desarrollo de software y mantenimiento al mismo.

Los problemas relacionados con el cambio de requerimientos, desarrollo de la tecnología así como la integración de diferentes métodos. SOA es de las últimas arquitecturas que aproxima la integración, desarrollo y mantenimiento de los sistemas de información empresariales complejos.

SOA no es radicalmente una nueva arquitectura, pero ha evolucionado como arquitectura distribuida y como método de integración. La integración entre aplicaciones ha involucrado desde principios, métodos de integración (Delgado, 2007), hasta referencias EAI (Integración de Aplicaciones Empresariales). EAI inicialmente enfocado en la integración de aplicaciones incremento la necesidad de integración entre compañías (B2B-Business to Business).

SOA provee y extiende la flexibilidad de métodos de integración (EAI) y arquitectura distribuida, enfocadas en el concepto de reusabilidad de aplicaciones o sistemas existentes con interoperabilidad eficiente e integración de aplicaciones y la composición de procesos de negocio. Un importante objetivo de SOA es la capacidad de aplicar cambios en el futuro en forma relativamente fácil.

SOA, define el concepto, arquitectura y marco de trabajo para procesos, habilitando el costo-beneficio del desarrollo, integración, mantenimiento de los sistemas de información a través de la reducción de complejidad, la estimulación de integración y reutilización.

La definición de SOA por Bernhard Borges, Kerrie Holley

SOA es un tipo de arquitectura que soporta servicios débilmente acoplados, habilita flexibilidad al negocio en una plataforma interoperable. SOA consiste en la composición de servicios de negocio alineados, que soportan flexibilidad y dinámicamente reconfigura procesos de negocio de principio a fin, usando interfaces basadas en descripción de servicios. SOA es más que un conjunto de tecnologías, no esta directamente relacionado a cualquier tecnología, aunque es implementado con servicios web. Los servicios web son la tecnología más

apropiada para llevar cabo SOA. Los conceptos más importantes de SOA son (Matjaz, 2006):

- *Servicios*: Proveen las funcionalidades del negocio, así como una aplicación de agencia de viajes, los servicios deben proveer valor al negocio, ocultar los detalles de la implementación y ser autónomos, los consumidores de servicios son entidades de software, las cuales llaman al servicio y usan sus funcionalidades.
- *Interfaces*: Los consumidores acceden al servicio a través de su interfaz. La interfaz de un servicio define la firma de un conjunto de operaciones públicas. La interfaz es un contrato entre el proveedor de servicio y un consumidor del mismo. La interfaz es separada de la implementación, se autodescribe y es independiente de la plataforma. La descripción de la interfaz provee la base para la implementación de servicios por el proveedor de servicios y la base para la implementación del consumidor de servicios. Cada interfaz define un conjunto de operaciones. Para definir los servicios de negocio, se necesita enfocar una granulación correcta de operaciones.
- *Mensajes*: las operaciones son definidas como un conjunto de acciones, los mensajes especifican los datos a intercambiar y describirlos en una plataforma y lenguaje independiente usando esquemas, los servicios intercambian solo datos, el cual difiere considerablemente del paradigma orientado a objetos componentes, donde el comportamiento también puede ser intercambiado.
- *Sincronización*: consumidores de servicios acceden a los servicios a través del bus de servicios. Este puede ser síncrono o asíncrono. En modo síncrono una operación regresa una respuesta al consumidor de servicios después que el procesamiento es completo, El consumidor de servicio tiene que esperar para completar. Usualmente se usa el modo síncrono con operaciones procesadas en un corto tiempo. En el modo asíncrono una

operación de servicio no regresa una respuesta al consumidor, si una respuesta es necesaria, usualmente hace un callback.

- Débil acoplamiento. Son servicios que exponen solo las dependencias y reduce todo tipo de dependencias artificiales, esto es particularmente importante cuando los servicios son temas frecuentes de cambio. Las dependencias mínimas aseguran que existirá un monto pequeño de intercambio requerido con otros servicios cuando un servicio es modificado. Este acercamiento provee robustez en los sistemas y promueve la reutilización de los servicios.
- Registro: Simplifica y automatiza la búsqueda para el servicio apropiado. Los servicios son mantenidos en los registros de los servicios, el cual funciona como un directorio, los proveedores de servicios publican servicios en registros; los consumidores de servicios ven los servicios en los registros, las búsquedas pueden ser por nombre, funcionalidad o por propiedades del proceso de negocio. UDDI, por sus siglas en inglés Universal Description, Discovery and Integration es un ejemplo de un registro de servicios.
- Calidad de servicio: Los servicios usualmente tienen asociados atributos de servicios de calidad, así los atributos incluyen seguridad, mensajería confiable, transacción, correlación, administración, políticas y otros requerimientos. La infraestructura debe proveer soporte para estos atributos, los atributos de calidad de servicio son muy importantes en los sistemas de información. En los servicios web, los atributos de calidad de servicio son cubiertos por las especificaciones WS en general, así como WS-Security, WS-Addressing, WS-Coordination, en lo particular.
- Composición de servicios dentro de un proceso de negocio. Probablemente es el más importante concepto de SOA, como una composición de servicios dentro de un proceso de negocio. Los servicios son compuestos en un

orden particular y siguen reglas que proveen soporte para procesos de negocio.

1.2.3.1 Elementos de la Arquitectura:

Una arquitectura orientada a servicios consiste de tres roles primarios: proveedor de servicio, consumidor de servicios e intermediario de servicios.

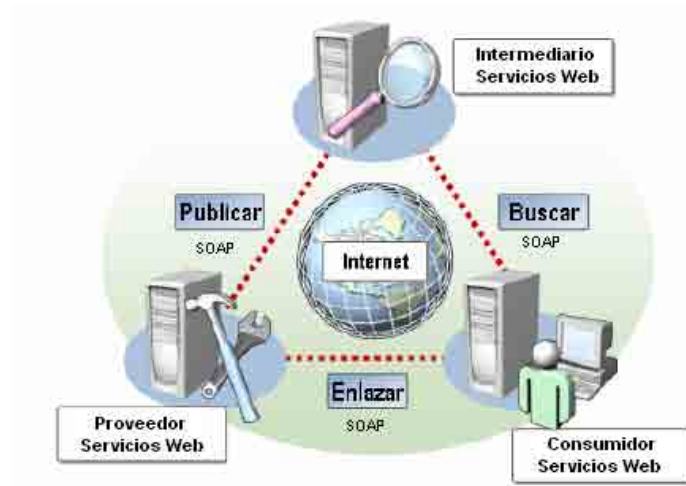


Figura 1. Elementos de SOA (Microsoft, 2003)

- Proveedor de servicios: es un nodo en la red intranet o internet que provee acceso a la interfase del servicio de software que permiten el establecimiento específico de operaciones. Este nodo provee acceso a los servicios de un sistema de negocios de un subsistema o un componente.
- Consumidor de servicios: es un nodo en la red que enlaza un servicio desde el proveedor y usa el servicio para implementar una solución de negocio. El cliente puede comunicarse por Hypertext Transfer Protocol (HTTP), los clientes incluyen browser, aplicaciones de consola y aplicaciones con interfaz grafica tradicionales (GUI).
- Intermediario de servicio: es un nodo en la red que es un repositorio de descripción de servicios y puede ser usado como un directorio para buscar la localidad de los servicios.

La interacción entre los roles y las operaciones básicas:

- **Publicación de servicios:** El proveedor de servicios publica sus servicios al intermediario de servicios. La información publicada incluye la definición de la interfase de servicios, localización del proveedor de servicios, información y/o documentación.
- **Buscar servicios:** Consumidor de servicios busca el servicio deseado o requerido.
- **Enlazar a servicios:** el consumidor de servicios enlaza a un servicio específico que es suministrado por un proveedor de servicios. El proceso de enlazar incluye autenticación de consumidores.

Los procesos de búsqueda y enlace de servicios pueden hacerse dinámicamente permitiendo que las aplicaciones se configuren por si solas.

1.3 Problema Real.

El modelado de arquitecturas empresariales ha representado por años uno de los mayores retos para las organizaciones. Estandarizar sus plataformas y que estas interoperen como una plataforma homogénea es uno de los principales retos del negocio y las áreas de tecnología (Selda, 2006), además, la integración de diferentes sistemas es otra parte a la cual las organizaciones invierten gran cantidad de recursos financieros y humanos.

La implementación de un modelo basado en SOBA en una organización no es sólo un aspecto de tecnología. La importancia de este tipo de modelos arquitectónicos radica en gran parte de la participación de los responsables del negocio.

Para modelar e implementar una arquitectura es indispensable identificar todos los elementos y/o componentes mínimos requeridos, de tal forma que el concepto de una plataforma heterogénea e integrada de sistemas se de en forma natural o lo más flexible obteniendo calidad de los datos, los procesos estén para el negocio

claros y bajo la misma visión, de tal forma que las metodologías apoyen en alinear las TI con los objetivos del negocio.

1.4 Justificación

El uso de metodologías apropiadas tanto para el modelado de procesos (BPM) como para la implementación de arquitecturas basada en SOBA (Arquitecturas de Negocio Orientada a Servicios) es la clave para el éxito del negocio el cual emprenda esta iniciativa. Sin embargo dentro del soporte al ciclo de vida siempre es parte fundamental la planificación y el análisis completo para tener todo el alcance y que éste no cambie de forma inmediata pese a las necesidades del negocio y el entorno sobre el cual está el mismo (Cranor,2003).

SOBA inicia con el antecedente de SOA, rápidamente se ha descubierto que una vez creado el proceso de elaboración de los servicios, es posible reducir la complejidad de la infraestructura de TI. SOBA está basado en estándares, los servicios web pueden ser reutilizados por distintas aplicaciones, la agilidad y flexibilidad del modelo podrá simplificar el trabajo, reducir tiempos, costos en las áreas de negocio y tecnológicas.

SOBA es recomendable para desarrollar aplicaciones totalmente nuevas que permitan aumentar la comunicación de los procesos de negocio dentro y fuera de la organización, ya que permite compartir, reutilizar servicios empresariales dentro y fuera de la organización (Malinverno, 2007), es decir, con los socios estratégicos, facilitando el intercambio de información entre aplicaciones de diferentes áreas según sea necesario.

Mediante SOBA se logra intercambiar información con los socios estratégicos, logrando un modelo de competitividad (modelo de Porter) que nos permite tener una cadena de valor importante para el negocio.

1.5 Pregunta de Investigación

¿Cuáles son los componentes de la arquitectura empresarial que proporciona la abstracción lógica de negocios permitiendo flexibilizar y mayor agilidad a la organización?

1.5.1 Hipótesis de Trabajo

El diseño de una arquitectura empresarial de negocios orientada a servicios permite a las empresas ser flexibles en la integración de sistemas de forma natural o solo son algunos componentes de la arquitectura que permiten esta agilidad a la organización.

1.6 Delimitación del alcance y Metodología de Diseño.

1.6.1 Alcance

La investigación extenderá los conceptos del modelo RM-ODP en la construcción de un modelo para el análisis y diseño de sistemas empresariales en el contexto de BPM-SOA, en la reciente área de investigación conocida como SOBA (Services Oriented Business Architecture).

El alcance se fundamentará de forma exploratoria, debido a que SOBA se encuentra en el estado del arte y son pocos los modelos que existen en investigaciones sobre este propósito o bien algunos aún no son concluidos.

Emplear la investigación exploratoria significa incursionar sobre un nuevo conocimiento, es decir examinar un tema o problema de investigación poco estudiado o que no ha sido abordado antes (Hernández, Fernández & Baptista, 2006), esto permitirá el acercamiento con las nuevas metodologías y modelos, que serán los esquemas a seguir por empresas que tratan de agilizar los procesos para mantenerse en una posición competitiva y es ahí donde los modelos de TI cobran su vital relevancia. El emplear la metodología de Zachman (Zachman, 2006), RM-ODP (Reference Model Open Distributing) e integrar bajo el contexto

SOBA es el objetivo de esta investigación para crear un nuevo método de modelado conceptual.

Se retomará la parte descriptiva para explicar como se construye la arquitectura, los modelos conceptuales, lógicos y físicos de los componentes que serán esenciales y que no pueden faltar para obtener un modelo basado en SOBA.

Se revisarán otras metodologías para resaltar los conceptos que son aplicables para SOBA y aporten mayor valor a la misma.

1.6.2 Diseño de la Investigación.

La investigación fundamentará el modelo, mediante la búsqueda de literatura congruente, para responder a la pregunta de investigación, y se desarrollará la Arquitectura Empresarial de Negocios, realizando el Modelado Conceptual, lógico y físico para una organización dedicada al comercio electrónico. El modelado físico corresponde a la implantación de la lógica de negocios en la organización a través de un Lenguaje de ejecución de negocios como BPEL (Matjaz, 2006) bajo la plataforma NetBeans 6.0 (Salter & Jennings, 2009). Una vez que se obtiene el primer modelo se construye el modelo físico en donde de igual forma se empleará la misma herramienta, pero ahora con un enfoque SOA, es decir, el prototipo evolucionará por etapas como tradicionalmente se emplea en ingeniería de software utilizando el ciclo de vida de sistemas.

La recopilación de información para lograr el propósito del modelado de la arquitectura fue por medio de la observación y explicación por parte de los miembros de la empresa “Acambio” involucrados en el proceso de intercambio de productos y/o servicios, esta información fue detallada de tal forma que el proceso se modelara en BPM y BPEL reflejando el detalle. Finalmente se obtiene la base que es el modelado conceptual. La entrevista también es parte importante de la recolección de datos, ver Anexo 1.

Antes, durante y después de realizar la entrevista como parte de la recolección de datos, es necesario cumplir con los siguientes pasos:

- Determinar la información requerida
- Seleccionar a la persona que será consultada
- Acordar el lugar y el horario de la entrevista
- Presentarse puntualmente a la entrevista
- Promover en el sujeto un estado de confianza y colaboración
- Si la persona no está dispuesta a ayudar, se intenta convencerla explicándole los beneficios de su colaboración.
- Formular las primeras preguntas.
- Aclarar todas las dudas propias o del encuestado
- La participación del entrevistador han de ser breves
- Tener en mente el objetivo de la entrevista.
- Durante la entrevista se toman notas
- Al terminar se tienen los comentarios y respuestas de los sujetos
- Los datos son valorados en términos de cantidad, calidad, representatividad y veracidad.
- Valoración en términos éticos respecto a cuál es la información

De los datos obtenidos en la entrevista y la visita a Acambio presentan una visión más amplia en donde el análisis de la información cualitativa, solo se toma como muestra a tres personas que intervienen en el proceso principal, de la unidad de expertos del negocio (unidad de análisis) y poder dar forma a sus comentarios y datos relevantes acerca del detalle del proceso.

La entrevista consta de preguntas abiertas del proceso de intercambio de productos o servicios en Acambio, la entrevista pretende que el tipo de información que se busca sea veraz y estandarizada por lo que la interacción directa nos aportará una mejor comunicación.

El vaciado de la información y la clasificación permitirán que la información sea manejable en términos de entendimiento y desarrollo para el diseño del proceso en BPM.

1.7 Objetivo.

Diseñar un método de modelado para una arquitectura empresarial que permita identificar los componentes a través de los cuales una organización puede llegar a ser flexible y ágil en la integración de procesos de negocio, sistemas internos o externos.

Capítulo 2 Marco Teórico.

La convergencia actual de las tecnologías de información en el contexto de los conceptos de administración en los procesos de negocios, tienen como objetivo hacer más ágiles los procesos de las organizaciones para colocarlas en el liderazgo (Francis Brudenell, 2007). Dos conceptos emergen como tales: Modelado de Procesos de Negocios (BPM por sus siglas en inglés) y la Arquitectura Orientada a Servicios (SOA por sus siglas en inglés). BPM involucra la convergencia de los conceptos de administración como TQM (Total Quality Management – Administración Total de calidad), BPR (Business Process Re-engineering- Reingeniería de Procesos de Negocio), y conceptos de Tecnología de información como ERP (Sistemas de plantación de recursos empresariales - Enterprise Resource Planning), WM (Workflow Management – Administración de flujos de trabajo), EAI (Enterprise Application Integration – Integración de Aplicaciones Empresariales) y BI (Business Intelligence /Dashboard and portals), la figura 2 muestra tal convergencia.

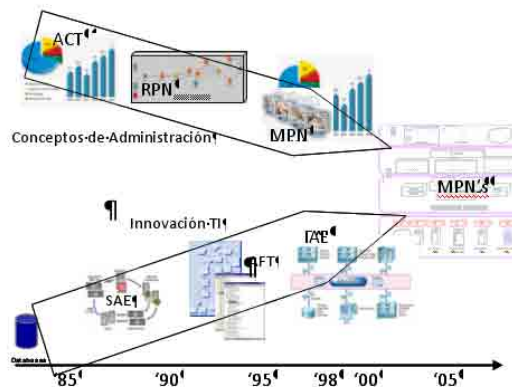


Figura 2. Conceptos base de BPM (Francis Brudenell, 2007)

La Arquitectura Orientada a Servicios (SOA) es un estilo arquitectónico que utiliza métodos y tecnologías que proporcionan a las organizaciones la capacidad de conectar y comunicar dinámicamente aplicaciones de software, entre diferentes socios de negocios y plataformas, para ofrecer servicios genéricos y confiables que pueden ser usados como elementos de construcción (Selda, 2006). La figura 3 muestra el modelo conceptual de dicha arquitectura.

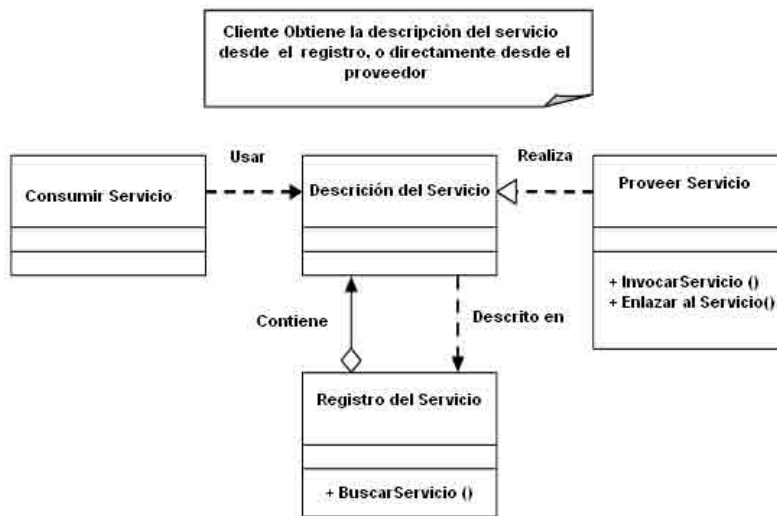


Figura 3. UML (Selda, 2006)

Otro modelo de arquitectura SOA propuesto en el contexto de la práctica de la industria es definida por (The Gilbane Report, 2003), la cual se muestra en la figura 4.



Figura 4. Arquitectura SOA (The Gilbane Report, 2003)

La cual es dividida en las capas de Web, de Aplicación, de Servicios e Infraestructura.

La capa de Web tiene como objetivo principal la visibilidad de todos los procesos y aplicaciones de la organización. (Malinverno, 2007), (Kamoun, 2006), (Madsen, 2006) exponen las ventajas de la integración o convergencia de la Arquitectura

Orientada a Servicios y la Administración de Procesos de Negocios, SOA-BPM. Aunque éstas son usualmente iniciadas independientemente, ambas comparten objetivos y recursos comunes, específicamente incrementar la agilidad empresarial. Al combinar proyectos SOA y BPM se obtienen mejores beneficios, los cuales son logrados en menor tiempo que cuando se inician independientemente, especialmente las grandes iniciativas. Aunque SOA puede ser exitoso sin BPM, nunca debe hacerse SOA sin involucrar analistas y administración de negocios. BPM crea un entendimiento profundo de los procesos que proporcionan una importante dimensión para entender que partes del portafolio de aplicaciones deben ser sujetas a reingeniería en los servicios SOA. Las tecnologías BPM pueden ser usadas para orquestar la ejecución de procesos, incluyendo el encadenamiento tardío de servicios SOA para que la infraestructura de TI habilite las partes del proceso. Sin SOA las iniciativas BPM tienden a crear puntos de integración externos a las interfaces de aplicaciones existentes. Sin embargo, cuando los servicios diseñados bajo SOA existen, menos integración personalizada es necesaria, ahorrando tiempo (Malinverno, 2007).

La figura 5 muestra un modelo de la arquitectura unificada SOA-BPM.

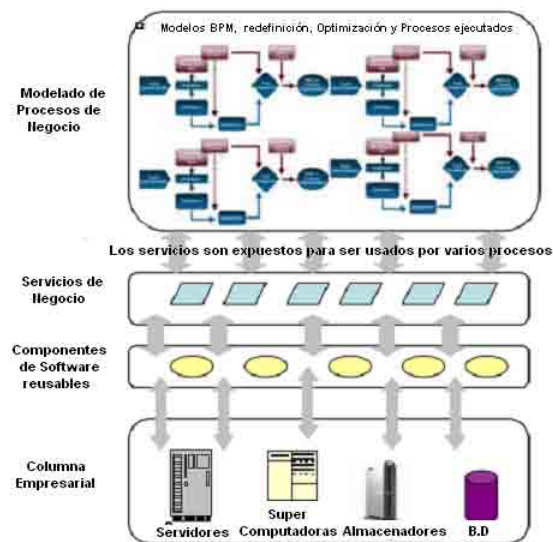


Figura 5. Modelo (Malinverno, 2007)

Por otro lado (Sánchez, Feroso & Joyanes, 2005)., (Schreiter, Laures, 2006), exponen la aplicación de BPM-SOA en el desarrollo de la Arquitectura Empresarial (AE) de una empresa, ya que éstas compiten con todo el mundo, y linearizando los procesos de negocios mejoran su competitividad, los cuales se encuentran completamente en la Arquitectura Empresarial.

2.1 *Arquitectura Empresarial*

La definición usada de AE es la siguiente:

Una Arquitectura Empresarial es la estructura actual y futura de los procesos de una organización, sistemas de información, unidades organizacionales y de personal, que se alinean con las metas y objetivos de la dirección estratégica de una organización (Cuenca, Ortiz & García, 2005), (Finkelstein, 2006).

El alcance de la arquitectura empresarial incluye personas, procesos, información, tecnología y la relación de cada uno, así como la relación con el ambiente externo, es decir, Un plano que explica como la tecnología de Información y los elementos de la Administración trabajan conjuntamente como una unidad.

Dentro del concepto de arquitectura empresarial hay dos elementos muy relacionados entre si: Modelado y Metodología.

El modelo (BPM) proporciona una representación simplificada o una abstracción de la realidad; puede incluir una representación de los conceptos y objetos físicos que ayudan a la organización a unificar el conocimiento, define una arquitectura como un conjunto finito de componentes interrelacionados. Una metodología es un enfoque estructurado para el seguimiento de las actividades que conducen, paso a paso, desde un sistema existente al futuro sistema teniendo en cuenta objetivos de evolución y limitaciones específicas, por lo tanto las empresas son entidades complejas compuestas por personas y procesos, que producen productos o servicios para los clientes. Para capturar la visión completa del sistema empresa en todas sus dimensiones además de su complejidad, surge el concepto de Arquitectura empresarial.

La arquitectura empresarial identifica los componentes principales de la organización y su relación para conseguir los objetivos de negocio. Actúa como fuerza integradora entre aspectos de planificación del negocio, aspectos de operación de negocio, aspectos tecnológicos, etc.

El marco referencial o framework es la estructura que permite almacenar y comunicar los diferentes elementos de la arquitectura empresarial. El framework de una arquitectura empresarial permite entender a una organización o una clase de empresas mediante la organización y presentación de componentes que conceptualizan, describen a la empresa y forman parte del framework completo. Para Zachman el framework es una estructura lógica para clasificar y organizar la representación descriptiva de una empresa.

Una característica o un principio específico de la arquitectura de una empresa y de su framework asociado es la definición de vistas. La complejidad de una empresa hace que sea difícil, por no decir imposible, su estudio bajo una única perspectiva.

Normalmente no hay una sola vista en una arquitectura, entre las vistas de una arquitectura destacan las siguientes.

- Negocio o empresarial
- Servicios
- Computacional
- Información
- Ingeniería y tecnología.

Prescribir una metodología para el diseño de una arquitectura empresarial considera diferentes niveles de abstracción o vistas de tal forma que las metodologías apoyan en la estructuración de arquitecturas empresariales, como es el caso del modelo RM-ODP (Reference Model Open Distributing Processing) y el framework de Zachman.

Las características que debe tener el framework son:

- Coherencia con los objetivos del negocio.

- Manejar complejidad del sistema vía abstracción por vistas.
- Una vista arquitectónica de un sistema o plataforma de negocio presenta primordialmente: Estructura, modularidad, componentes esenciales y flujos de control principales de un dominio ó perspectiva específica del problema de negocio que soporta la arquitectura. Es una ventana al interior del sistema desde una perspectiva específica enfatizando una idea en particular.

2.2 Zachman Framework

El **Zachman Framework** para arquitectura empresarial fue publicado en 1987 por John Zachman. Proporciona la arquitectura para la infraestructura de la información de una organización (Zachman, 2006). El marco contiene seis filas y seis columnas que componen treinta y seis celdas o aspectos distintos. No hay dirección establecida en la secuencia o proceso para la aplicación de la arquitectura. El objetivo es asegurarse de que todos los aspectos de una empresa estén cubiertos y muestra las relaciones que asegurarán un sistema completo sin importar el orden en el cual se establecen.

John Zachman describe el framework como una estructura lógica simple para clasificar y organizar la representación descriptiva de una empresa que significa la administración de una organización.

En la figura 6 se ilustra el framework, los renglones significan las diferentes descripciones de la empresa desde perspectivas diferentes, las columnas, Un aspecto de la organización de la parte alta a la parte inferior de perspectivas distintas (Young, 2007).







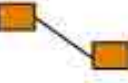


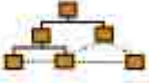


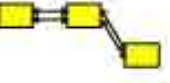
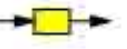
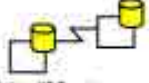
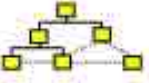


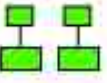


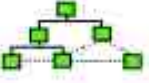








Las vistas o arquitecturas que cubre el framework de Zachman son las siguientes:

- La vista o arquitectura de negocio queda cubierta con las dos primeras filas del Zachman Framework, “*Scope*” y “*Business Model*” donde se define la dirección de la empresa y el propósito de su negocio (*scope*), así como, se

muestran todas las entidades de negocio y procesos, y las relaciones entre ellos (*business model*).

- Arquitectura o vista de información, la tercera fila corresponde con la vista de sistema de información, la cual define las funciones de negocio descrito en la fila dos, *business model*, pero en términos de información. Las funciones de negocio y las necesidades de la empresa son transformadas a datos y requerimientos de información.
- Arquitectura o vista tecnológica, Los requerimientos definidos en la arquitectura de sistemas o en la arquitectura de aplicación quedan concretados en la arquitectura tecnológica. En ella se determina la tecnología a utilizar, selección de bases de datos, tipo de lenguaje a utilizar, interfaz de usuario, etc. y cómo debe ser utilizada. Zachman definen en su framework una arquitectura tecnológica.
- Arquitectura Organizacional y Arquitectura de Recursos, en el Zachman Framework la vista de organización puede quedar representada por la columna "who" donde se indican las relaciones dentro de la empresa, indicando autoridad y responsabilidad en el trabajo.

ENTERPRISE ARCHITECTURE - A FRAMEWORK TM

	DATA	How	FUNCTION	How	NETWORK	Where	PEOPLE	Who	TIME	When	MOTIVATION	Why	
SCOPE (CONTEXTUAL)	List of Things Important to the Business 		List of Processes the Business Performs 		List of Locations in which the Business Operates 		List of Organizations Important to the Business 		List of Events/Cycles Significant to the Business 		List of Business Goals/Strategies 		SCOPE (CONTEXTUAL)
Planner	ENTITY = Class of Business Thing		Process = Class of Business Process		Node = Major Business Location		People = Major Organization Unit		Time = Major Business Event/Cycle		Ends/Means = Major Business Goal/Strategy		Planner
BUSINESS MODEL (CONCEPTUAL)	e.g. Semantic Model 		e.g. Business Process Model 		e.g. Business Logistics System 		e.g. Work Flow Model 		e.g. Water Schedule 		e.g. Games Plan 		BUSINESS MODEL (CONCEPTUAL)
Owner	Ent = Business Entity Reln = Business Relationship		Proc = Business Process IO = Business Resources		Node = Business Location Link = Business Linkage		People = Organization Unit Work = Work Product		Time = Business Event Cycle = Business Cycle		End = Business Objective Means = Business Strategy		Owner
SYSTEM MODEL (LOGICAL)	e.g. Logical Data Model 		e.g. Application Architecture 		e.g. Distributed System Architecture 		e.g. Human Interface Architecture 		e.g. Processing Structure 		e.g. Business Rule Model 		SYSTEM MODEL (LOGICAL)
Designer	Ent = Data Entity Reln = Data Relationship		Proc = Application Function IO = User Views		Node = IIS Function (Processor, Storage, etc.) Link = Line Characteristics		People = Role Work = Deliverable		Time = System Event Cycle = Processing Cycle		End = Strategic Assertion Means = Action Assertion		Designer
TECHNOLOGY MODEL (PHYSICAL)	e.g. Physical Data Model 		e.g. System Design 		e.g. Technology Architecture 		e.g. Presentation Architecture 		e.g. Control Structure 		e.g. Flow Design 		TECHNOLOGY MODEL (PHYSICAL)
Builder	Ent = Segment/Table/etc. Reln = Pointer/Key/etc.		Proc = Composite Function IO = Data Elements/Data		Node = Hardware/Systems Software Link = Line Specifications		People = Job Work = Screen Format		Time = Execution Cycle = Component Cycle		End = Condition Means = Action		Builder
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)	e.g. Data Definition 		e.g. Program 		e.g. Network Architecture 		e.g. Security Architecture 		e.g. Timing Definition 		e.g. Rule Specification 		DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)
Sub-Contractor	Ent = Field Reln = Address		Proc = Language Statement IO = Control Block		Node = Address Link = Protocol		People = Identity Work = Job		Time = Interrupt Cycle = Machine Cycle		End = Sub-condition Means = Step		Sub-Contractor
FUNCTIONING ENTERPRISE	e.g. DATA		e.g. FUNCTION		e.g. NETWORK		e.g. PEOPLE/MACHINES		e.g. SCHEDULE		e.g. MOTIVATION		FUNCTIONING ENTERPRISE

John A. Zachman, Zachman International

Figura 6. Framework de Zachman (Young, 2007)

2.3 RM-ODP (Reference Model Open Distributing Processing).

El modelo RM-ODP (Reference Model Open Distributing Processing) (Akira, 2005), (ISO/IEC 10746-2, ITU-T , 1996), (Kazi,1995), (Malinverno, 2007), prescribe una metodología conceptual o modelo de referencia que proporciona un marco de trabajo conceptual y una arquitectura que integra aspectos relacionados con la distribución, interoperabilidad y portabilidad de sistemas software, de tal forma que la heterogeneidad de plataforma se cumpla, en donde el diseño de sistemas distribuidos constan de diferentes niveles de abstracción llamados “puntos de vista”. La estructura de puntos de vista es muy genérica, incluye un conjunto de conceptos, estructuras y reglas para cada punto de vista, proporcionando un “lenguaje” para especificar sistemas. RM-ODP reconoce los siguientes puntos de vista: Empresarial, Información, Computacional, Ingeniería, y Tecnología (Madsen, 2006).

Puntos de vista, algunos autores las llaman perspectivas.

Cada uno de estos puntos de vista trata de satisfacer a una necesidad distinta, cada una interesada en aspectos diferentes del sistema. Y asociado a cada uno de los puntos de vista se define un lenguaje especializado.

- **Vista de empresa.** Se describen todas las actividades de negocio, objetivos y políticas, así como la forma en que pretende cumplir.
- **Vista de información:** relacionada con los datos que necesitará ser almacenada y procesada en el sistema, es decir, se muestran las estructuras de datos y los valores que podrá tomar.
- **Vista computacional:** Es la descripción de un sistema como un conjunto de componentes que interactúan mediante interfaces.
- **Vista de ingeniería:** relacionada con la infraestructura que soportan la distribución del procesamiento, datos del sistema.

- **Vista Tecnológica:** relacionado a todos los detalles de los componentes de infraestructura que consta de hardware y software con el que es construido el sistema distribuido.

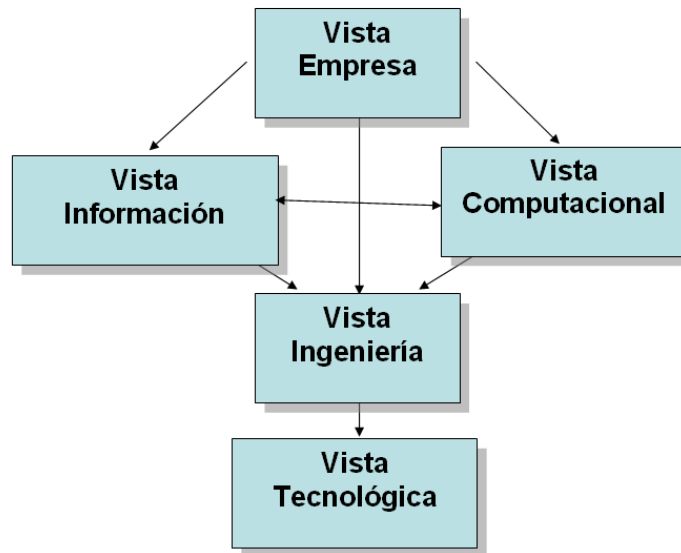


Figura 7. Vistas de RM-ODP. (Delgado, 2007)

2.4 UML Lenguaje de Modelado Unificado

La complejidad de representar el mundo real en términos de negocio y los objetivos de este en una arquitectura debe pasar por un proceso de abstracción de un espectro inusual de problemas inherentes a la alineación de la tecnología existente con los objetivos de negocio, consecuentemente existe una fuerte motivación por aplicar técnicas y tecnologías de diseño que permitan simplificar y proveer la confiabilidad del diseño de arquitecturas de negocio o bien tecnológicas que sean implementadas en escenarios del mundo real.

UML (Unified Modeling Language), Un lenguaje de propósito general para el modelado orientado a objetos, impulsado por el Object Management Group (OMG, www.omg.org), ha propuesto el enfoque que soporta a áreas de negocio de representarse con casos de negocio por medio de diagramas.

El lenguaje de modelado unificado, es un lenguaje basado en los paradigmas Orientado a Objetos, fue creado como una consolidación de conceptos probados que evolucionaron y fueron refinados durante el surgimiento de la tecnología orientada a objetos a finales de los 80's y principios de los 90's, durante este periodo nacieron muchos lenguajes de modelado (Lavagno, Grant & Selic, 2004), y como resultado se obtuvo una fuerte base para crear un lenguaje común de modelado. Este debería de capturar el estado de arte en la tecnología orientada objetos en una semántica común y una notación simple.

Los autores Grady Booch, Jim Rumbaugh y Ivan Jacobson, produjeron las primeras versiones de UML. Cuando la OMG encontró que el propósito era crear un lenguaje de modelado que pudiera ser usado para el análisis y diseño de aplicaciones Orientadas a Objetos Distribuidas, entonces UML nace como un libro de texto la mayor parte de las revisiones han estado sujetas en la versión UML 2.0 y la definición de adopción anticipada es en el 2003. Esta última versión agrega capacidades de modelado para describir arquitectura de software. Desde entonces la adopción de UML ha sido rápida tanto por la industria como académicamente.

La esencia del lenguaje unificado es la diversidad de cruzar los aspectos de los diferentes dominios de un negocio y tecnología, por esta razón UML colabora en el modelado de escenarios de negocio con algunos de sus diagramas que ayudan a representar y entender al mismo.

UML combina notaciones provenientes desde modelado Orientado a Objetos así como modelado de datos, componentes y Workflow. Los modelos y diagramas están completos desde el punto de vista de sistemas, sin embargo existen relaciones de trazabilidad para el negocio.

La propuesta de Rational Unified Process (RUP), ayuda a modelar y entender al negocio mediante diagramas que son útiles para definir los modelos conceptuales y lógicos.

Los tipos de diagramas que soporta UML son:

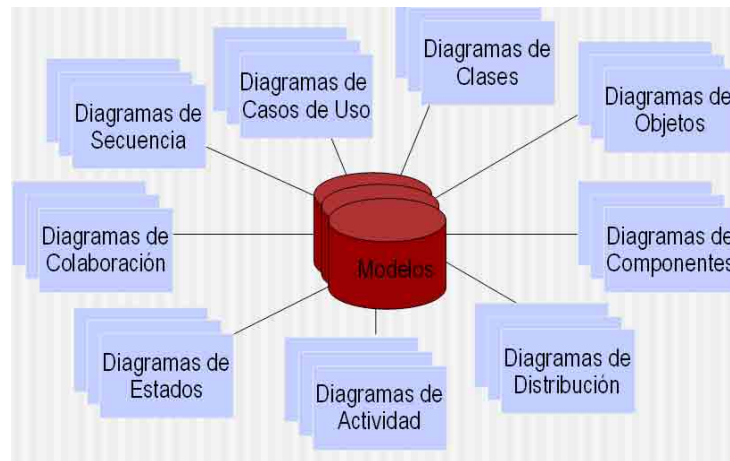


Figura 8. Tipos de Diagrama UML

Los diagramas son técnicas para capturar información respecto de los servicios que en un sistema proporciona a su entorno, es decir, son la especificaciones de requisitos con los que el negocio debe satisfacer sus necesidades, por lo tanto el modelado de negocios es la identificación de las características para la creación del modelado conceptual y lógico como primeros pasos de la creación de una arquitectura empresarial.

2.5 Modelado Conceptual

El modelado conceptual consiste en identificar el sistema empresarial, los procesos de negocio funcionales, la ejecución de los procesos a través de sus áreas funcionales principales, descubrir los procesos de tipo estratégico (Duffy, 2004), del negocio, de apoyo, para aplicar técnicas de descomposición sobre ellos para encontrar subprocesos. Aunque es la técnica más usada desde el paradigma estructurado aún tiene vigencia para aspectos de generalizar y después encontrar subprocesos particulares.

2.6 Modelado Lógico

Para el modelado lógico se debe de apoyar en el modelado de UML con casos de uso de negocio, identificar actores principales y procesos o actividades que ejecutan los autores principales o secundarios. El caso de uso debe ser sencillo, simple, claro y conciso o bien diagramas de actividades detallado de tal forma que

se tenga una secuencia del workflow de cómo los datos viajan a través de un flujo definido por el negocio.

2.7 Modelado de Procesos de Negocio. (BPM).

BPM es un paradigma para la administración de procesos de negocio. El concepto es algo abstracto y en realidad engloba numerosas actitudes y tareas a desarrollar en una organización. El papel de la administración de modelado de procesos y optimización de los mismos (Poirier, 2004) representa uno de los aspectos más importantes ya que forman la parte medular para la creación de la arquitectura empresarial basadas en cualquiera de los modelos de referencia presentados en el capítulo anterior y que SOA para negocios lo retoma con mayor importancia.

El concepto de proceso de Martín Ould, lo define como una serie de actividades coherentes que son realizadas por un grupo de colaboradores para alcanzar un objetivo (Flores, 2001). Los procesos han estado presentes en los negocios, por lo que el objetivo está relacionado al negocio dentro de una organización definiendo roles funcionales y relaciones, la importancia de poder clarificar la situación actual o futura desde la perspectiva del negocio, por lo tanto, el modelado de procesos de negocio es la representación y una técnica que visualiza negocios en el mundo real, principalmente en diagramas de flujo. La necesidad de modelar se ha incrementado en los últimos años y adoptado en muchas áreas de una organización (Sánchez, Feroso & Joyanes, 2005), apoyando a transformar mercados y relaciones con clientes internos (áreas internas) y externos (clientes potenciales), por ello emerge BPM (Business Process Management) como la pieza faltante en el rompecabezas del negocio especialmente para organizaciones que buscan mejorar su futuro.

BPM proveerá visibilidad, control y seguimiento a todos los procesos mediante la orquestación de las actividades de las personas, sistemas y participantes con otros; por lo que todas las áreas participantes en el flujo de información (workflow) puedan dar seguimiento en tiempo real y por lo tanto dar respuesta a cualquier

pregunta derivada del mismo negocio, esta capacidad era imposible ofrecerla en un tiempo menor por los ciclos de liberación.

La gestión de procesos de negocio a través de flujos de información, mediante BPM se persigue con el modelado de las actividades del negocio para lograr una mejor administración, automatización y optimización. El centrarse en los flujos de negocio o de información, es una técnica que permite mejorar la comprensión y la comunicación mediante diseños gráficos de secuencias de tareas, que conjuntamente realizan una función de alto nivel e importancia para los objetivos de una organización (Finkelstein, 2006). Representan una formalización de la ordenación requerida de este tipo de procesos, que mediante las herramientas o software adecuado puede ser explotado para mejorar desde varias perspectivas la secuenciación de tareas de negocio. Mediante un workflow se estudian los aspectos operacionales de una actividad de trabajo; cómo se estructuran las tareas, cómo se realizan, cual es su orden, cómo se sincronizan o la manera en que se relacionan unas con otras, por lo que todo el diseño esta basado en patrones para capturar buenas prácticas.

No es raro que los conceptos workflow y BPM se confundan, además de otras características en común, ambos contienen una representación de un flujo de trabajo (un workflow desde el punto de vista lógico), es decir, el flujo de trabajo puede ser representado en ambos por algún lenguaje de notación y de especificación. Pero el hecho es que el BPM no solo define las tareas y el seguimiento de las mismas, sino que tiene como objetivo solucionar el ciclo completo de un proceso de negocio.

Adicionalmente, BPM con su filosofía no es la misma que la de un diagrama de secuencias o de colaboración de UML. Este modela la interacción de una parte de la aplicación a nivel de negocio interno, especificando al detalle las llamadas a los métodos incluso con su signature y los objetos involucrados, de modo que teóricamente un programador no debe necesitar tomar decisiones arquitectónicas ni de diseño y se limitara a la implementación, ya que están pensados en principio para llegar a un nivel de detalle suficiente para evitarlo. BPM sin embargo está

pensado para una comprensión del negocio a alto nivel, diseñando tareas de alto nivel y flujos de alto nivel. Un nodo bien podría desembocar mensajes a cientos de objetos. Implementar un flujo a este nivel de detalle no es el objetivo y además provocaría un exceso.

Construir procesos de negocio basado en la coordinación de actividades iterativas o humanas, como servicios. BPM une lo mejor del mundo del Workflow, en un mismo flujo se integran tareas realizadas por personas, con tareas automatizadas (sistemas), entregándole al participante de un proceso de negocio una sola interfaz, ocultando la interacción con los sistemas legados.

El proceso de negocio (modelo BPM), consta principalmente de los siguientes componentes:

- **Actividades:** son las tareas que debe hacer una persona (tarea interactiva), o debe hacer un sistema (servicio) dentro del proceso de negocio. Por ejemplo “Revisar Antecedentes Financieros” (actividad interactiva), o “Imprimir Contrato” (servicio de un sistema), “Alta de clientes”.
- **Roles y Usuarios:** son los responsables de ejecutar las tareas interactivas, por ejemplo un “Ejecutivo” de un Banco o comercio electrónico.
- **Objeto de Negocio:** es la información o documento que fluye a través del proceso de negocio, por ejemplo la “Solicitud de Crédito”, o el “Crédito de Consumo” (en que se transforma la solicitud), o la “Alta del Cliente” o Alta de un producto.
 - **Flujos** (flechas): es la secuencia que se define entre las actividades.
 - **Decisiones:** criterios para tomar distintas opciones en el procesos, distintas direcciones en el flujo.
 - **Subproceso:** otro proceso interno.

La notación completa para el desarrollo de los diagramas que se emplean en BPM es sencilla y simple de seguir (ver Anexo 2 Notación completa de BPM).

2.7.1 Características de BPM básicas:

- Es un estándar internacional de modelado de procesos aceptado por la comunidad a nivel mundial.
- Independencia de cualquier metodología de modelado de procesos.
- Crea un puente estandarizado para disminuir la brecha entre los procesos de negocio y la implementación de estos.
- Modelado de los procesos de una manera unificada y estandarizada.

Los procesos de negocio deben ser planeados y desarrollados a través de personas y sistemas. El tiempo y costo en desarrollarlos se reduce por que los componentes existentes pueden ser accedidos para coordinarse a través de un tipo de interfaces estándares, permitiendo la integración entre tecnologías. El negocio puede ver los procesos de principio a fin usando BPM en combinación con SOA, pero BPM representará la parte base para la creación de una arquitectura empresarial, creando una nueva generación de modelado e integración de plataformas, es decir, BPM permite a los diseñadores de software hacer uso de herramientas como un componente en lugar de aplicaciones que trabajen con los procesos de negocio como un componente aislado.

La ejecución acertada de los procesos de negocio comunes en la organización es la relacionada directamente al funcionamiento eficaz de la infraestructura tecnológica. Dentro del negocio, prácticamente cada actividad es un proceso de negocio, por lo tanto BPM es parte del dominio de una arquitectura basada en un modelo de referencia para establecer un conjunto de aplicaciones compartiendo funcionalidades similares, comportamiento y estructura, esto describe las características esenciales de casos de negocio que en la mayoría de los casos se representa en UML. Aún el monitoreo de tecnologías de información están centrándose en una métrica técnica y de tendencias de aplicaciones individuales y de componentes de infraestructura, la mayor parte de los negocios tienen una

comprensión limitada del impacto de acoplamiento de tecnologías correlacionadas para el buen funcionamiento de procesos y servicios de negocio.

2.8 Estado del Arte de SOBA (Arquitectura de Negocios Orientada a Servicios).

El estado del arte se refiere al nivel más alto alcanzado por una determinada área del conocimiento científico o técnico, es decir, el nivel más alto en desarrollo o propuesto en un área específica, para el caso de SOBA, proporciona al negocio una arquitectura que no solo se basa en tecnología sino que puede ser aplicada al mismo como soporte de la tecnología que está alineada a los objetivos del negocio permitiendo la creación de arquitecturas flexibles e integración entre plataformas.

Muchas organizaciones tienen varios sistemas heterogéneos creando una red confusa de aplicaciones interconectadas a servicios y recursos de datos, surge entonces la clara necesidad de asociar estos sistemas incompatibles e integrarlos en aplicaciones empresariales.

La semántica de las arquitecturas de negocio orientadas a servicios (SE-SOBA), proveerá una Arquitectura dinámica reconfigurable, previendo a las organizaciones responder rápidamente y de manera flexible ante los cambios del mercado globalizado.

La fusión de procesos de negocio es la transformación de actividades de negocio que deben ser integrados a interfaces de procesos autónomos (Salaman, Stevens, 2006). El desarrollo de SOBA's y la publicación de servicios tal vez es la implementación de la visión de fusión de procesos de negocios, soportado por una capa de abstracción para envolver interfaces a través del lenguaje de descripción de servicios web por sus siglas en inglés (WSDL – Web Service Description Language).

La innovación en desarrollo de SE-SOBA, extiende la notación de SOA para aplicar la semántica de la tecnología de servicios web, usa la ontología y

semántica de un lenguaje web que describe la estructura para el paso de los mensajes, empleando XML como un lenguaje estándar y universal.

Las interfaces de servicios se combinan con reglas de formalización de escenarios de negocio y procesos permitiendo una dinámica reconfigurable de la arquitectura que dará a la empresa la respuesta rápida y flexible a los cambios del entorno exterior, soportando innovación y crecimiento del negocio, con lo que se incrementa el potencial para la recuperación de la inversión en tecnologías de información y finalmente proveer robustez en la arquitectura a nivel empresarial.

2.8.1 Definición de SOBA.

SOBA es un concepto reciente, pero aún no existe una definición uniforme (Yushun, Huang, 2008), se ha definido como un servicio de tecnología innovadora envolviendo funcionalidad de procesos de negocio o actividad, la cual es ofrecida por un proveedor de servicios. La naturaleza del acceso al servicio es transparente, con independencia de plataforma y orientación a negocios.

La Arquitectura de Negocios Orientada a Servicios, es una metodología que esta basada en la definición de procesos de negocio, transformándose en servicios web, es decir, SOBA esta soportada en lo que hoy se conoce como SOA como plataforma tecnológica, que permite la interoperabilidad e integración entre sistemas a través de servicios web, pero la pieza clave es aplicar el modelado de negocios empleando BPM.

SOBA elimina la complejidad del manejo de acceso a los múltiples tipos de servicio, incluyendo funcionalidad técnica, de negocio, y de datos de negocio. SOBA proporciona visibilidad y el acceso al ciclo vital del proceso completo (Fethi, 2006), dando a las organizaciones la capacidad de aplicar conceptos de SOA en diseño de procesos y servicios web, integración, ejecución, análisis, y mejora los procesos estratégicos del negocio.

Mientras que una organización puede tener muchos procesos que se integren en uno solo, la mayoría de los procesos de negocio críticos tienen datos complejos y funciones de negocio críticas, para los requerimientos de la integración. Las arquitecturas tradicionales de integración toman varias formas, entre ellas: código de integración personalizado que es replicado a través de diversos clientes (clientes = aplicaciones), las plataformas de integración de aplicaciones empresariales, e intermediarios ó mejor conocidos como brokers de mensajes. Todos estos acercamientos son absolutamente complejos y existen en un nivel por debajo de los procesos reales y de los usos que requieren los datos o las funciones de negocio ofrecidos por la integración.

2.8.1.1 Características

- Fácil integración entre entidades del negocio.
- Reuso de procesos de negocio
- Reduce costos del negocio total.
- Unifica negocios.
- Servicios compuestos por procesos de negocio
- Los procesos de negocio pueden ser encapsulados dentro de un servicio.
- Múltiples procesos interactúan con eventos/mensajes y comparten el mismo origen o datos.
- El proceso cambia automáticamente a través del tiempo, esto requiere asegurar el uso de servicios y seleccionar componentes de servicio en tiempo real.

2.8.1.2 Importancia del nuevo Enfoque

SOA tiene gran potencial para ayudar a organizaciones a implementar más rápidamente y productivamente en el entorno empresarial exigente de nuestros días. Alcanza los resultados que benefician perceptiblemente a la empresa:

- Agilidad: rápida adaptación a las condiciones cambiantes del mercado.
- Reuso usando el trabajo hecho en diferentes contextos y no reinvertiendo.
- Uniformidad. Basado en la habilidad de compartir sin importar la plataforma tecnológica.

La llave al éxito para SOA es en última instancia el valor que puede entregar al negocio. SOBA habrá establecido su equilibrio cuando los profesionales del negocio ganen la capacidad de controlar sus propios procesos y reglas directamente, así como la adaptación para cambiar rápidamente y correctamente. Más allá de eso, SOA entrega una prima ofreciendo una ocasión verdadera para que entregue la reutilización amplia y el desarrollo basado en componentes en un ambiente técnico uniforme.

BPM combina las ventajas estratégicas de la gestión del proceso del negocio con la tecnología de integración requerida para que SOA mantenga con eficacia las iniciativas con los objetivos estratégicos del negocio a todos los niveles dentro de la organización. Esta sinergia de gran alcance da como resultado SOBA's que prevé ayuda incomparable de SOA, la mejora de proceso verdadera y mayores resultados para el negocio (Salaman, Stevens, 2006).

2.8.1.3 Cuadro Comparativo SOA vs SOBA

Un diferenciador dominante de la arquitectura de SOBA y SOA es respecto a su capacidad de manejar procesos (Procesos de alto nivel, es decir más cercanos a

las actividades humanas) y sistemas primordiales. Las características de SOBA permiten el despliegue de servicios a los usuarios bajo la forma de procesos de negocio cohesivos, orquestados y sincronizados a través de la estrategia y objetivos de la organización.

Las organizaciones que colocan a BPM como un componente de su estrategia de SOA generan más valor al negocio. BPM optimiza el uso del SOA a través de los principales procesos de negocio que impactan directamente.

SOBA (SOA PARA NEGOCIO)	SOTA (SOA PARA TECNOLOGÍA)
Reusabilidad de servicios de negocio.	Reusabilidad de componentes de sistemas.
Acuerdos de Nivel de servicio	Especificación de Interfaces
Directorio de servicios	Repositorio/directorio de servicios.
Capacidad de interoperabilidad con otras áreas y servicios liberados.	Acoplamiento débil entre módulos o bloques.
Autonomía departamental	Contenido simple y abstracto
Liberar valor el negocio (en discretas y medibles unidades).	Servicios sin estado

Tabla 1. Comparativo

2.8.1.4 Componentes esenciales en SOBA.

Los servicios con funciones de negocio, son generalmente los que representan una transacción o una actividad con el significado valioso para el negocio (por ejemplo: crear el registro de un cliente, una factura, o cerrar un boleto abierto del servicio de atención al cliente). Los servicios con función de negocio se relacionan con una transacción requiriendo generalmente el planeamiento y el control apropiado de la transacción, tal como ubicación, concurrencia, etc. Estos tipos de servicios se distinguen debido a sus requisitos únicos.

Los datos del negocio o mensajes proporcionan el acceso a la información contenida en varias aplicaciones o sistemas y repositorios de la organización en una forma abierta pero segura, eliminan la réplica de la integración de punto a punto entre los sistemas y los procesos. Por ejemplo, los servicios de datos del negocio proveerían de un proceso de la entrada de datos de órdenes de venta sobre clientes activos, sistemas de productos disponibles, así como el estado del inventario actual, es decir los mensajes que entran y salen de cada proceso.

Los servicios de la tecnología ó servicios web, hacen las funciones fáciles, ofrecidas como parte de la plataforma, tal como una función para crear una herramienta de gestión. Las aplicaciones de BPM estándar se basaron en tecnologías para orquestar el consumo de las funciones de negocio, de los datos de negocio, y de las funciones de servicio técnico específicos dentro y a través de procesos de negocio múltiples.

2.9 Modelos Existentes.

2.9.1 Framework Conceptual “Fusión”

Es la solución de un marco de trabajo que facilita la integración de aplicaciones empresariales heterogéneas que existen en una misma empresa o en diferentes organizaciones, el diseño de fusión esta basado en SOA (Salaman, Stevens, 2006). Usando algunas tecnologías preexistentes (web services, UDDI, WSDL y XML), tomado lo mejor de cada uno de ellos. Esta innovación y técnica de integración de aplicaciones empresariales reduce los tiempos.

La arquitectura conceptual de Fusión esta basada en la infraestructura de servicios web, así como proveer la capa de integración, considerando la interfaz (WSDL) y la comunicación SOAP, y los servicios típicamente almacenados en la UDDI.

Contiene una segunda capa de integración que agrega forma y semántica a la funcionalidad de servicios y datos de negocio en la descripción de servicios web e

interfaces, haciendo más amplia la notación de SOA y servicios web aplicando una notación común.

La segunda capa de integración soporta la semántica de la descripción de servicios web (WSDL), tomando en cuenta que primero se debe proveer una descripción formal de la funcionalidad de servicios web para facilitar la categorización eficiente y descubrimiento de servicios web.

En los servicios es necesario identificar cada evento que pueda ocurrir en un ambiente de negocio y la organización de la lógica de negocios del dominio creando un diccionario conceptualizando los aspectos de funcionalidad de servicios potenciales del dominio de negocio.

- Registro de servicios de negocio: Los servicios Web son desarrollados y su WSDL creado, estos deben ser categorizados y publicados en el registro de servicios de negocio para permitirse ser usados y descubiertos. Normalmente se encuentran en la UDDI.
- Capa Orientada al negocio: la capa superior de abstracción es necesario en Fusión acercarla a EAI (Integración de aplicaciones empresariales) con SOA y servicios web, es un paso adelante por que los servicios se consideran inteligentes y la semántica de negocios con la misma característica. Esta última capa de integración invoca el uso de procesos de negocio que maneja flujos de trabajo (workflow) y modelado.

SOA obtiene la experiencia y conocimiento de consultores de negocio, además de expertos para conceptualizar un escenario típico de negocio, facilitando un modelado formal y ejecución de procesos de negocio empleando lenguaje de ejecución de procesos de negocio para servicios y lenguaje de modelado de workflow.

Los procesos de negocio, son el resultado del balance entre metas y significado. Dos tipos de estrategia debe derivarse de la estrategia principal: estrategia de negocio y de recursos.

2.9.2 SOMA Un método para desarrollar soluciones orientadas a servicios.

El autor Arsanjani muestra una arquitectura y modelado de servicios que se ha usado para conducir proyectos de varios alcances en múltiples industrias. El uso y la estructura de los métodos usados, diseñados e implementados y el desarrollo de arquitecturas orientadas a servicios como parte del desarrollo de software. El modelado de servicios a través de SOA (Arsanjani, 2008).

SOA es de las mejores prácticas y métodos para el desarrollo de aplicaciones distribuidas y su adopción es reciente. SOMA incorpora métodos de aspectos claves sobre el diseño y la integración con software existente desarrollado por otros métodos, pero esta orientado a un ciclo de vida de desarrollo de software.

Como un método del ciclo de desarrollo de software para construir soluciones basadas en SOA, o cualquier solución usando principios orientados al servicio. SOMA define las técnicas claves y describe los roles en un proyecto de SOA y estructura de trabajo. La estructura de trabajo incluye, la entrada, salida de productos de trabajo para tareas y la dirección prescriptiva para el análisis detallado, el diseño, la implementación, y el desarrollo de los servicios, componentes y los flujos necesarios para construir un ambiente SOA robusto y reutilizable.

El método SOMA incluye 7 fases. Es importante notar que las fases de SOMA no son lineales ellas son aplicadas como manejo de riesgo, iterativas e incrementales usando el ciclo de desarrollo de SOA, el ciclo de vida de un proyecto de SOA usando un matiz peculiar. El ciclo de vida de un proyecto de SOA usa la combinación de dos puntos clave: el principal desarrollar aplicaciones con el alcance similar, es decir, tareas hechas de forma similar en una línea de negocio o proyecto simple, la notación de indicativos similares es de aplicaciones, el método de SOA puede ser aplicado a soluciones de desarrollo, línea de negocio entre líneas de negocio, extender a la organización entre socios del negocio.

El método de SOMA es compuesto por patrones con técnicas aplicadas en todas las fases con diferentes grados de elaboración y precisión. Las decisiones están basadas en la información que se conoce acerca de los servicios durante la identificación y entonces se elaboran los servicios, se analizan los recursos en la fase de identificación y funcionalidad de los sistemas que requieren integración, se expande el análisis en la fase de especificación, se identifican componentes y objetos existentes, sus operaciones pueden ser reutilizables para realizar los servicios. La siguiente fase es determinar el modelo de dependencias de los servicios cuando se define el portafolio, para elaborar las dependencias con componentes y sistemas físicos, por lo que SOMA se considera un ciclo de vida de desarrollo.

El segundo principio del desarrollo de software es la iteración sucesiva. Los conceptos de desarrollo del ciclo iterativo e incremental ha existido por mucho tiempo, este se enfoca en la priorización y mitigación de factores de riesgos para asegurar la calidad del producto de la solución. Este se representa como un modelo espiral de software del autor Boehm (Arsanjani, 2008), las iteraciones sucesivas son conectadas con la notación de evolución de servicios e implica enfocarse no solo a la asociación de riesgos con la implementación, sino también con dependencias asociadas con el portafolio de servicios, como se envuelven los mismos a través del ciclo de vida. La notación del servicio de dependencias asociadas con otros servicios y potencialmente otras dependencias en otros servicios o en el código (back-end) de sistemas, base de datos y componentes que deben ser llamados.

Así en SOMA, una priorización del modelo de servicio es manejada y basada en diagramas de dependencia de servicio, tomando en cuenta los factores de riesgo que envuelven los aspectos en TI de la arquitectura. Un subconjunto de servicios son priorizados por la siguiente liberación de implementación.

Las siete fases se describen a continuación:

1. Modelado de Negocios y transformación: En esta fase el modelo es negocio, simulado y optimizado. El área clave para la transformación es identificada. La solución de SOA son híbridas y típicamente incluye múltiple tipos de solución, esto es porque los servicios se identifican y especifican la duración de la fase para poder realizar la siguiente fase y cubrir los diferentes escenarios, así como desarrollo personalizado, integración, transformación y la integración de aplicaciones.
2. Identificación: la fase de identificación pertenece a la identificación de 3 componentes fundamentales de SOA: servicios, componentes y flujos.

Un paso inicial es la identificación de servicios o posibles candidatos, la recomendación es iniciar por alinear los servicios con los objetivos del negocio, a este paso se le llama modelado de servicios-metas, esta alineación de TI es ejecutada con el negocio y permite realizar el modelado de procesos de negocio estableciendo el análisis apropiado. Identificación SOMA es un proceso de tipificación de servicios candidatos, creando un portafolio de servicios de TI alineados al negocio que colectivamente soportan procesos de negocio y metas de la organización, esto se hace a través de procesos. Para la identificación de servicios se emplean técnicas como la descomposición acercando al análisis en cambio de negocio y oportunidades, estrategia corporativa y metas de negocio. La técnica de descomposición usa la técnica top down que es enfocada al modelado de procesos de negocio, reglas e información.

La descomposición incluye una actividad llamado análisis de área funcional que provee una partición estructural del domino del negocio en distintas áreas funcionales, esto provee un significado natural identificando componentes después en el ciclo de vida. La descomposición de procesos y modelado de los mismos provee oportunidad no solo de identificar servicios si no también los flujos que serán usados en la orquestación de estos.

3. Especificación de servicios, componentes y flujos de información: SOA es diseño de alto nivel como bien significan partes del diseño detallado de componentes de servicio completos. Durante la fase de la especificación, se ampliará los recursos existentes y la elaboración de los servicios, flujos y componentes que desde la fase de identificación fueron analizados. El modelo de servicios más elaborados en términos de dependencias de servicios, flujos, composición, eventos, reglas y políticas, operaciones, mensajes y requerimientos no funcionales.

Primero se elabora y especifica el modelo de información, después se mejora el análisis y la especificación de los elementos existentes. Las entidades de negocio, su estructura de alto nivel y las relaciones (diagramas de relaciones de entidades son relevantes para el alcance de servicios) son identificados durante la fase de identificación. Se elabora un modelo de datos conceptual dentro de un modelo de datos lógico implementado, el cual debe ser definido en términos de su dominio y tipos de datos lógicos (carácter, numérico, fecha e imagen).

En la especificación debe enfocarse en el diseño de los mensajes, lo cual incluye entradas, salidas y mensajes de errores, para eliminar la transformación de datos múltiples en la capa de servicio, una de las mejores prácticas es usar un modelo de mensajes común que sea aceptado por la organización, el modelo define el flujo de los mensajes en la capa de servicio, después se selecciona el formato de los mensajes (ejemplo XML). Se establecen los tipos, elementos y atributos, representando las entidades de negocio y sus atributos de negocio, los tipos de mensajes son usados como bloques para entradas, salidas y mensajes de errores para los servicios.

Se analiza el sistema de interfaces y los parámetros de entrada y salida de los sistemas existentes. La especificación de servicios es el centro del modelado de servicios y diseño detallado de los mismos, las dependencias de los servicios, componentes, la composición de servicios y el flujo todo en

conjunto define la coreografía y orquestación de servicios habilitados en funciones de negocio o procesos, las operaciones de servicios son invocadas y ejecutadas por una función del negocio en una implementación de TI.

4. Construcción e Implementación, Monitoreo y Administración.
5. La fase de implementación, se construye, genera y unen los servicios, se crean los encapsulados necesarios u otros mecanismos para realizar la integración y las pruebas para servicios de componentes o flujos, En la fase de distribución, monitoreo y administración se enfocan a empaquetar y ejecutar pruebas de aceptación del usuario para finalmente distribuir en ambientes de producción.
6. Realización. Se valida que las decisiones a través del prototipo diseñado y desarrollado, la exploración de la viabilidad técnica es una forma de planear e implementar prototipos clave que sean un ejercicio para la construcción de la arquitectura, se planean factores de riesgos y cambios tecnológicos enfocados en el requerimiento.

SOMA es un elemento clave para la creación de arquitecturas que provee una vista de SOA, esta vista facilita comunicación y provee una representación de progreso y evolución de SOA en un diagrama de alto nivel. Las capas de la arquitectura de SOA son una instancia de SOMA pero en esta provee la parte de procesos de negocio, pero toma todos los aspectos de SOA para desarrollar e incluir el gobierno.

SOMA, provee una metodología de ingeniería de software para construir de principio a fin aplicaciones basadas en SOA, la motivación para desarrollar esta arquitectura, es que existen muchas arquitecturas, diseños y vistas técnicas de desarrollo, pero SOMA tiende a ser flexible a diferencias de las arquitecturas actuales. SOMA tiene un alcance empresarial puede extenderse y se basa en servicios web por lo que representa agilidad.

Capítulo 3 Modelo de Arquitectura de Negocios Orientada a Servicios en Acambio.

Los servicios como el nuevo campo de innovación en Tecnologías de Información, y basados en procesos de negocio o actividades que ofrece el mismo representa un marco de trabajo para la creación de nuevas arquitecturas que permitan flexibilidad y cambios de forma rápida ante el entrono global.

SOBA son los procesos de negocio transformados en servicios web que son ejecutados o consumidos por un cliente, es decir, SOBA es una composición de los servicios de Web con el fin de tener en ellos las políticas y todo a aquel elemento que este relacionado con el negocio, alineado con las nueva tendencias de tecnología y metodologías.

A continuación se muestra el diagrama general de la arquitectura propuesta en este trabajo, en el cual se representan los elementos principales y la combinación de las metodologías que están basadas en las mejores prácticas en el desarrollo de arquitecturas empresariales.

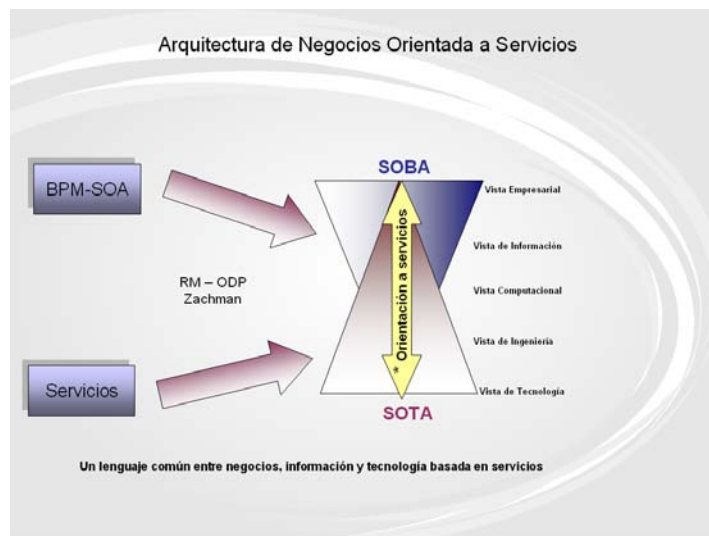


Figura 9. SOBA-UIA.

En la figura 9, se puede observar la integración del framework de Zachman, el cual es útil para la creación de la arquitectura empresarial, sin embargo, carece de una notación estándar. Tiene una dependencia de integrar UML para el modelado de procesos de negocio, mismo que afortunadamente disfruta de la aceptación universal, a pesar de que las notaciones pueden ser estructuradas o bien Orientadas a Objetos. La notación apropiada, principal aportación de este trabajo es la integración de modelado de procesos de negocio con BPM con el manejo de los mismo bajo en el contexto de SOA. Considerando las limitantes y ventajas de cada metodología como son el framework de Zachman y RM-ODP, con esta última se tiene asegurada una arquitectura totalmente distribuida que unida a la parte de servicios web y el framework de Zachman se logra una estructura que libera valor en el modelo conceptual de SOBA-UIA.

El framework de Zachman incorpora notaciones en UML o RUP (Casos de uso de negocio), no impone limitación o selección, no obstante, en la vista empresarial de RM-ODP es necesario elegir alguna notación común. La arquitectura emplea como base el framework de Zachman, para obtener los casos de uso de negocio, con en el modelo SOBA-UIA, se extiende el modelado con BPM, de tal forma que organizar y diseñar los procesos de negocio para la automatización de los mismos es más sencillo, mismo que soporta el diseño del workflow, procesos de negocio y XML como el lenguaje para la integración de la arquitectura además del desarrollo en UML. Bajo el contexto de servicios web, el framework de Zachman queda como una metodología Orientada a Objetos útil para identificar las clases en el diseño físico.

El modelo SOBA-UIA integra las cinco vistas de RM-ODP, estructurando con el modelo de Zachman, donde RM-ODP provee las vistas que deben contemplarse dentro de la arquitectura para marcar la dirección como parte del ciclo de desarrollo de la misma, obteniendo el modelado Conceptual, Lógico y Físico.

Con SOBA-UIA, se introduce un marco de trabajo con la capacidad de crear un modelo diferente, que se extienda a una arquitectura distribuida Orientada a Servicios (SOA), con la automatización de los procesos de negocio dentro del

framework de Zachman y RM-ODP, con la descripción de procesos de negocio asociados a la arquitectura de tal forma que las organizaciones puedan tener una colección de sus procesos de negocio, accediendo a ellos a través de la arquitectura de negocios empresarial.

No existe un estándar para ampliar la notación de los procesos de negocio obtenidos en los casos de uso. Los flujos de trabajo se obtienen a partir de los diagramas UML. Empleando BPM se tiene la estandarización tanto para las actividades propias del negocio como la de los flujos, por lo tanto, la notación de BPM es eficiente para crear el mapeo de procesos de negocio envueltos en servicios web.

XML, en la base del modelo, ha hecho que los modelos de UML obtenidos con el framework de Zachman sean flexibles, por lo que, la transición de la vista empresarial y de información hacia la vista computacional de RM-ODP, sea factible para la creación del modelo físico.

El Modelo SOBA-UIA provee las técnicas necesarias para desarrollar una arquitectura que permita la integración de los procesos de negocio (BPM) y extender el modelo Orientado a Objetos a un modelo distribuido mediante SOA, empleando componentes de modelado de UML con BPM para el análisis y diseño de la estructuración de la vista empresarial de RM-ODP y el framework de Zachman (Alcance y Business model). Estos conceptos se consideran de alto nivel, mismos que pueden ser asociados a los diagramas de procesos de negocio para su ejecución mediante algún lenguaje. En este trabajo se emplea BPEL. Técnicamente la descripción de los procesos de negocio, pueden ser ejecutados por la tecnología que soportará a los procesos.

En los modelos de SOMA y fusión contemplan los componentes que se tienen para SOA. SOMA en particular se enfoca al ciclo de desarrollo de servicios web lineal, el cual es similar al ciclo tradicional de desarrollo de sistemas llamado comúnmente de cascada que se emplea en ingeniería de Software, por lo tanto, este modelo esta dentro de la categoría SOTA. SOBA-UIA no solo emplea el ciclo

lineal para la construcción de la arquitectura. Si no que incorpora las vistas de RM-ODP que han de seguirse para obtener los diferentes modelos. Por otra parte el modelo de fusión tiene una similitud con SOBA-UIA, esta se encuentra en la capa orientada al negocio y la vista empresarial en donde SOA obtiene la experiencia de los expertos de negocio y facilita el modelado de proceso de negocio.

Finalmente, el análisis y diseño de la arquitectura se inicia seleccionando un proceso que la organización considere importante y que sirva como escenario de un proceso de negocio que genere valor al mismo, para el caso de la arquitectura propuesta, se toma el proceso de Intercambio, en el cual se requiere comunicación con clientes y servicios dentro de la empresa Acambio.

3.1 Modelo SOBA- UIA

3.1.1 Antecedentes del Modelo SOBA-UIA

En la clase de Análisis y Diseño de sistemas de la Maestría de Ingeniería de Sistemas Empresariales que imparte la Universidad Iberoamericana, se propone utilizar algunas de las metodologías RM-ODP, Zachman bajo un contexto SOA para explicar la evolución de las aplicaciones distribuidas, sin embargo esta propuesta se basa exclusivamente en la parte tecnológica, retomando algunos aspectos de negocio y ampliando la notación de SOA y adoptando BPM dentro del modelado se extiende el modelo para obtener una nueva propuesta para crear Arquitectura de negocios Orientada a Servicios (SOBA), empleando las técnicas tradicionales de Ingeniería de software, como top-down para la descomposición de procesos y ciclo de desarrollo secuencial.

Los participantes dentro de la primera definición se encuentran alumnos de la Universidad Iberoamericana Yosafat Moscoso, Andrés Morón, Edith Hernández y el Dr. Alfonso Miguel Reyes quién imparte la materia y taller de SOA, durante el periodo 2008.

3.1.2 Formalización del Modelo.

El modelo de SOBA-UIA se compone en cinco arquitecturas, siguiendo el concepto de la separación de los aspectos aplicados en RM-ODP: Arquitectura empresarial, servicios, computacional, de información, ingeniería y tecnológica ver figura 10.



Figura 10. Modelo formal.

3.1.2.1 Arquitectura Empresarial

Esta describe como opera el negocio, desarrolla una clara imagen del flujo de los procesos de negocio en la organización y como estos son soportados por la infraestructura tecnológica basada en servicios.

La arquitectura empresarial es el resultado de la definición de estrategias, funciones, proceso y requerimientos funcionales. Uno de los beneficios claves de tener una arquitectura empresarial desde la perspectiva de TI es que esta permite definir entregables relacionados a la información, bajo este contexto y requerimientos tecnológicos necesarios para soportar los cambios de los requerimientos del negocio.

El modelo de la arquitectura empresarial es la representación de los procesos de negocio en la manera en la cual son soportados de forma automatizada. La definición de un proceso consiste de una red de actividades y sus relaciones que

indican su inicio y fin de un proceso, la información de actividades individuales, así como los participantes y datos asociados.

El modelado de la arquitectura empresarial esta dividida en tres actividades, modelado conceptual, modelado lógico y el modelado físico.

La arquitectura empresarial tiene la siguiente estructura:

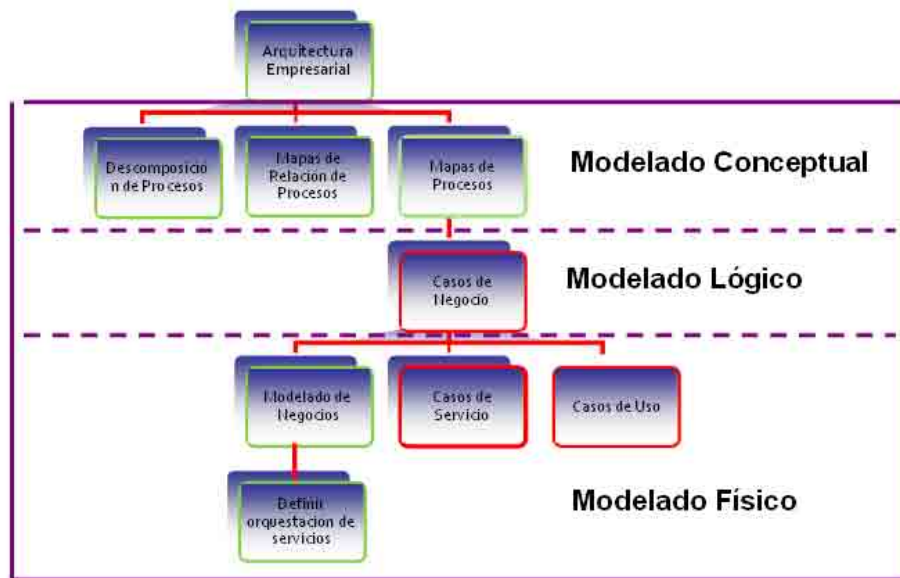


Figura 11. Estructura de la Arquitectura Empresarial.

Modelado Conceptual: provee los conceptos principales acerca de la existencia de la empresa así como los objetivos, misión de la organización. Estos aspectos son necesarios de ser identificados para desarrollar los procesos del negocio y los flujos de trabajo relacionados. Esta actividad consiste de una combinación de top down y bottom up, técnicas de descomposición de procesos.

Los elementos principales del modelado conceptual son:

- Modelado organizacional: Identificar a la organización y sus roles asociados a sus responsabilidades.
- Áreas de interés: descomposición de procesos representados en diagramas y mapas de procesos.

Especificación del modelo de procesos de negocio. Identificación de actividades de procesos y negocios con las áreas de interés definidas. Estas son descritas en detalle en los requerimientos y objetivos, durante esta etapa los casos de negocio son identificados (objetos de alto nivel de información).

Modelado de distribución de negocios: Identificar aspectos de distribución de la organización, se identifican las diferentes entidades y recursos de la empresa.

Modelado Lógico: los conceptos obtenidos son analizados y estructurados en una forma que proveerán un modelo lógico usando una notación técnica como BPM y UML.

Esta actividad consiste en una combinación de varias técnicas top down, casos de negocio y diagramas en UML, en donde se rehace un análisis de los procesos de negocio y casos de uso de negocio de alto nivel.

Los elementos principales del modelado de lógico son:

- Requerimientos de especificación de negocio, estos describen los requerimientos funcionales de los sistemas desde la vista de negocio soportados en los diagramas de caso de negocio y diagramas de actividades de UML.

Modelado Físico: Los diagramas obtenidos en el modelado lógico son los insumos para desarrollar el modelado físico, estos son mapeados en el módulo de BPEL (Lenguaje de Ejecución de Procesos de Negocio).

La notación de BPEL debe tener las siguientes actividades

- Identificación de partner links o colaboradores
- Declaración de partner links
- Identificación de servicios y construcción de los mismos.
- Diseño del workflow en BPEL

- Pruebas de funcionalidad del modelo.

El modelado físico es el puente entre la arquitectura empresarial, computacional y de servicios.

3.1.2.2 *Arquitectura de servicios y Computacional.*

Los casos de negocio y casos de servicios son la información de entrada necesaria para esta arquitectura, en donde se definen los diseños lógicos del sistema e interfaces de usuario, servicios, modelos de información y prototipos.

El diseño es visto desde la perspectiva computacional, componentes de software y tienen la siguiente estructura, cada proceso indica una actividad establecida que produce los elementos del sistema.

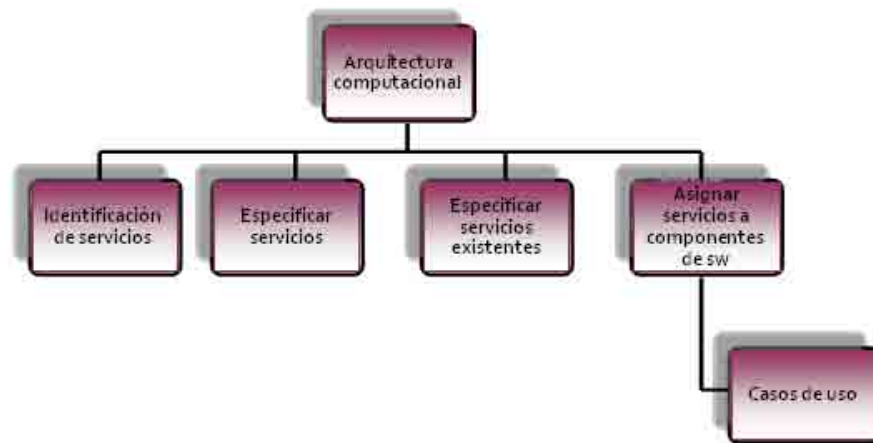


Figura 12. Arquitectura Computacional.

Identifica y categorizar los servicios.

La actividad es iniciada cuando los servicios han sido identificados, esto es importante clasificar los servicios dentro de una jerarquía, esto ayuda a determinar la composición y capas para coordinar la construcción de servicios independientes basados en una jerarquía,

También esto ayuda a no permitir el síndrome de la proliferación incrementando el número de servicios con un limitado gobierno, resultado un mejor desempeño, escalabilidad y administración de errores.

La especificación de servicios, esta actividad inicia con la inexistencia de servicios y deben de incluirse en la jerarquía o inventarios de servicios.

Construir servicios, en este paso reconoce cual es el software con el que deberán de construirse los servicios. Los servicios de asignación a los subsistemas que se han identificado hasta ahora. Estos subsistemas tienen componentes de la empresa que realicen su funcionalidad publicada.

Construcción de componentes de software

Los casos de uso especifican la entrada para modelar y construir los componentes de software que expondrán los servicios. La construcción del diseño del subsistema entonces será observada como construcción de la puesta en práctica de un componente que realiza los servicios en la actividad siguiente. Los detalles del componente que ejecutan los servicios se especifican:

- Datos
- Reglas
- Servicios
- Configuración de perfiles.

Por lo tanto RM-ODP propone un modelo integrado para la Arquitectura de Sistemas Empresariales, cuya base es el modelo RM-ODP. La Vista Empresarial indica un modelado orientado a BPM. Aunque las referencias son difusas en cuanto a la aplicación del modelo RM-ODP en el contexto de SOA-BPM, (Madsen, 2006), (Francis Brudenell, 2007), ofrecen las bases iniciales para la incorporación de RM-ODP como solución del desarrollo de la Arquitectura de Aplicación Empresarial que satisfaga la Arquitectura Empresarial basada en SOA-BPM.

3.2 Aplicación del Modelo SOBA-UIA

3.2.1 Breve reseña de Acambio

Acambio es una empresa dedicada al mercado virtual, es decir, a mercados que hacen uso de las tecnologías como Internet para establecer una posición estratégica en el comercio electrónico, en vez de los mercados tradicionales que se ubican en un área geográfica definida.

El mercado virtual tiene las ventajas que permiten un número casi ilimitado de ofertantes, un número mayor de visitantes o usuarios, y no está restringido a horarios de funcionamiento, las empresas pueden publicar las fotos de los productos y/o servicios, así como la información de contacto de forma que cuando un comprador esté interesado pueda realizar la transacción de intercambio.

La característica que hace diferente Acambio de otras empresas de comercio electrónico es el intercambio entre productos y/o servicio sin necesidad de utilizar dinero en efectivo.

El proceso de Intercambio representa el escenario sobre el cual se aplicarán los conceptos que se han manejado en los dos capítulos anteriores de este trabajo, con esto se pretende satisfacer el entendimiento de la arquitectura empresarial desde su concepción lógica y física hasta su aplicación bajo el contexto de SOBA.

3.2.2 Identificación de necesidades

La identificación de las necesidades es una forma de realizar un diagnóstico indispensable para elevar la calidad de los procesos que intervienen entre las diferentes entidades del negocio.

Algunas de las necesidades identificadas son:

- Falta de comunicación entre sistemas existentes y entidades.
- No existe la optimización y automatización de procesos para eliminar trabajos manuales.

- Duplicidad de procesos.

3.2.3 Modelo Conceptual.

La organización trata de reunir dos visiones por una parte, la de negocio y también la vista tecnológica, la definición de los objetivos sobre los que esta regido el negocio y la identificación de las necesidades son la base para entender como debe de constituirse el sistema empresarial, es decir, la parte funcional de la empresa y la comunicación que existe entre sus diferentes entidades, por lo tanto, el modelo conceptual debe concebirse como el sistema empresarial, en donde la metodología de Zachman aplica dos aspectos.

- **“Alcance”**

El propósito del negocio es posicionarse con una ventaja competitiva utilizando los recursos y medios para formar parte del modelo global de mercados virtuales.

- **“Modelo de Negocio o sistema empresarial”**

Se identifican las entidades que forman parte de Acambio y como interactúan unas con otras.

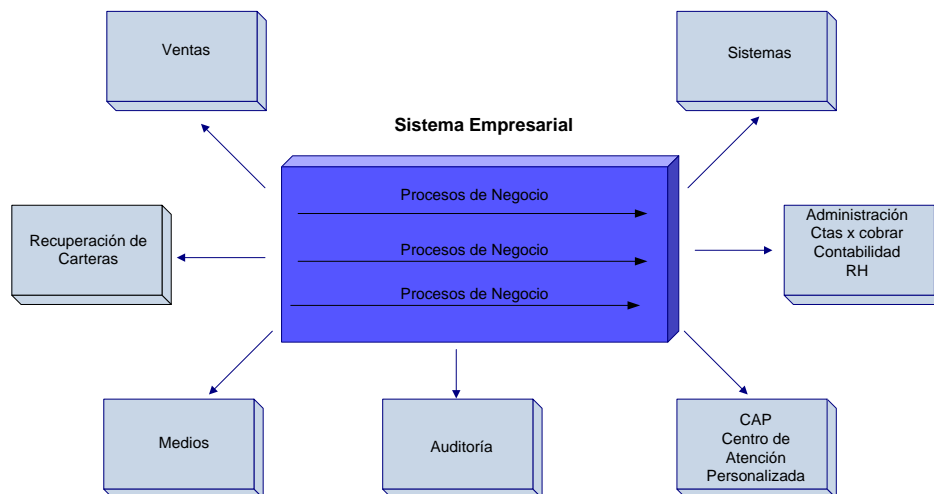


Figura 13. Sistema Empresarial

La parte medular de la arquitectura se centra en la identificación de los procesos, por esta razón la tipificación de los mismos constituyendo las actividades de negocios funcionales, un ejemplo de cómo realizar esta actividad de forma sencilla es mapear el área funcional y determinar cada una de las actividades que lleva a cabo.

Procesos de Negocio Funcionales	
Area Funcional	Procesos de Negocio
Ventas	Identifica clientes potenciales
	Alta de transacciones del intercambio de clientes corporativos
	Seguimiento a entregas del producto
	Seguimiento a transacciones
	Validación de las transacciones
	Registrar transacciones disponibles
	Registra comisiones de cada transacción
	Asegura que cada cliente este vendiendo o comprando
Recuperación de Cartera	Recuperación de clientes
Medios	Identifica clientes de medios masivos de comunicación (proveedores o consumidores)
	Alta de transacciones de intercambio de medios masivos de comunicación
	Planeación de campañas de publicidad
Auditoria	Establecer procesos de negocio y documentarlos
	Supervisar que se lleven acabo
	Aplicar sanciones preventivas y correctivas
Contabilidad	Elaborar y registrar facturas de comisiones de las transacciones
	Registra libros de mayor
	Entrega de facturas al cliente
	Administración de las cuentas por pagar y las cuentas por cobrar
	Administración del pago de nominas
CAP (Centro de Atención Personalizada)	Atención a Clientes corporativos
	Seguimiento a quejas y solicitudes
	Labor de promoción y preventa
RH	Contratación de personal
	Capacitación

Tabla 2. Procesos de Negocio

El modelo conceptual, esta describiendo la vista empresarial que combina los dos primeros elementos de la metodología de Zachman. Conocer como interactúan las entidades del negocio, permitirá a la organización alcanzar sus objetivos y en la medida que esta ejecución sea eficiente el resultado se estará alienado al objetivo.

Puede representarse la ejecución de procesos de la siguiente forma:

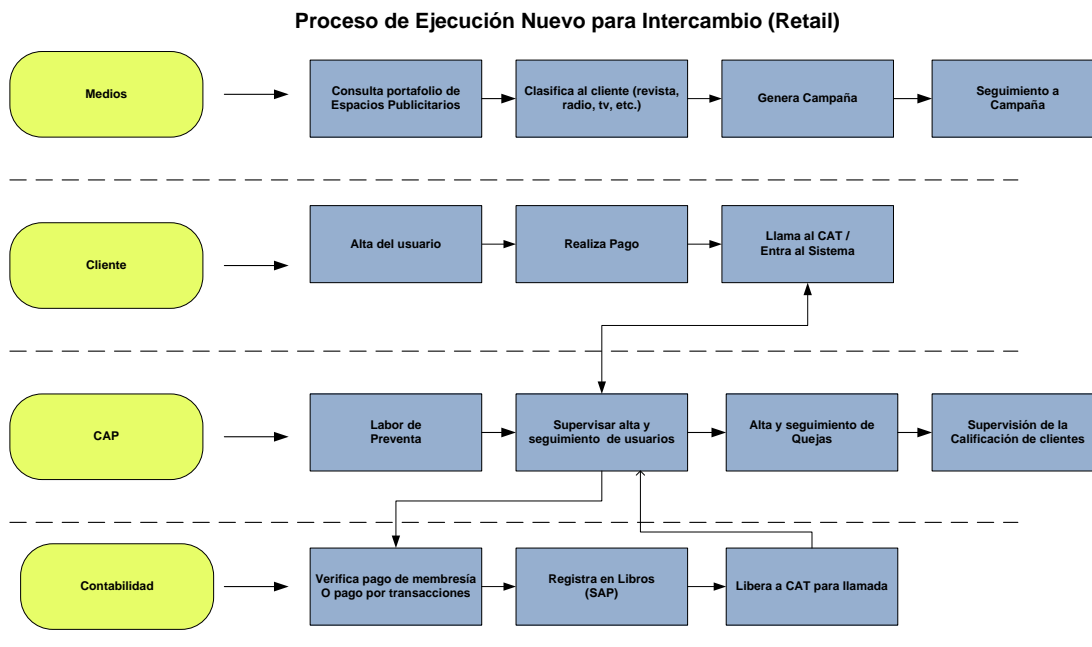


Figura 14. Ejecución de procesos entre diferentes entidades

Como resultado de esta ejecución es posible hacer la clasificación de los datos de entrada y salida, sin embargo en la vista de información se definirán las estructuras que deberán estar almacenando.

El diagrama anterior muestra un nivel de desagregación que permite llegar al nivel de procesos operativos, de bajo nivel, para comenzar un trabajo más detallado que conduzca a describir, mejorar o rediseñar.

3.2.4 Modelo Lógico

Entender los principios de la organización subyacentes puede ayudar a utilizar de forma más eficaz esta importante característica del diseño de la arquitectura, en esta fase se muestra las relaciones entre los componentes lógicos y base de la solución. La arquitectura lógica y el análisis de casos de uso, que es la información utilizada en la etapa de diseño lógico de la implementación.

El proceso seleccionado de intercambio en casos de uso de UML se centra de la siguiente forma en donde los actores que son algo o alguien externo a la empresa que interactúan con ella, en nuestro caso son los empleados internos y el cliente para las actividades son los procesos, ver figura 15.

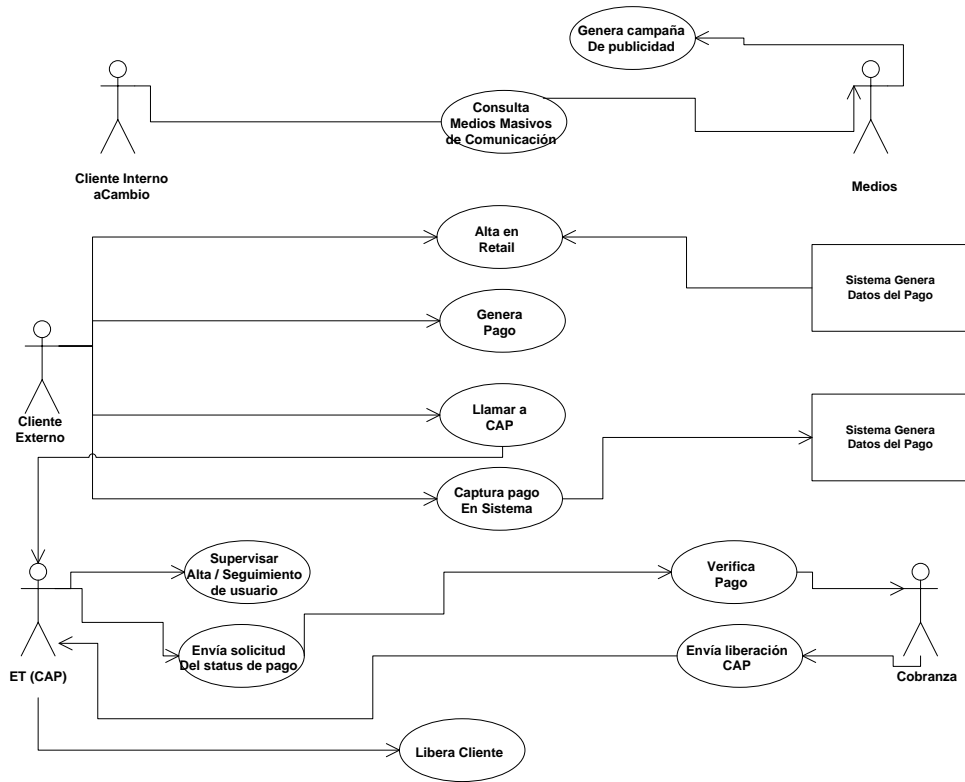


Figura 15. Casos de uso de Negocio

La información proporcionada por los casos de uso de negocio son todos aquellos escenarios que deben considerarse dentro del flujo o la secuencia del proceso, por tal razón las actividades o procesos en el modelado de la arquitectura funcionarían como las actividades principales y de ellas llevar a cabo la descomposición para alcanzar el nivel detalle para entender, así como para obtener los datos necesarios que se estarán intercambiando de pasar el flujo de trabajo de una entidad a otra.

Para determinar la secuencia o workflow que llevarán las actividades dentro del mismo caso de uso del negocio deben mostrarse de principio a fin de todas las actividades que intervienen, los diagramas de actividad UML nos apoyan a entender de forma gráfica esta secuencia y es otra forma de representar el caso de uso del negocio, pero adicional con la ventaja de incorporar las validaciones establecidas como reglas de negocio.

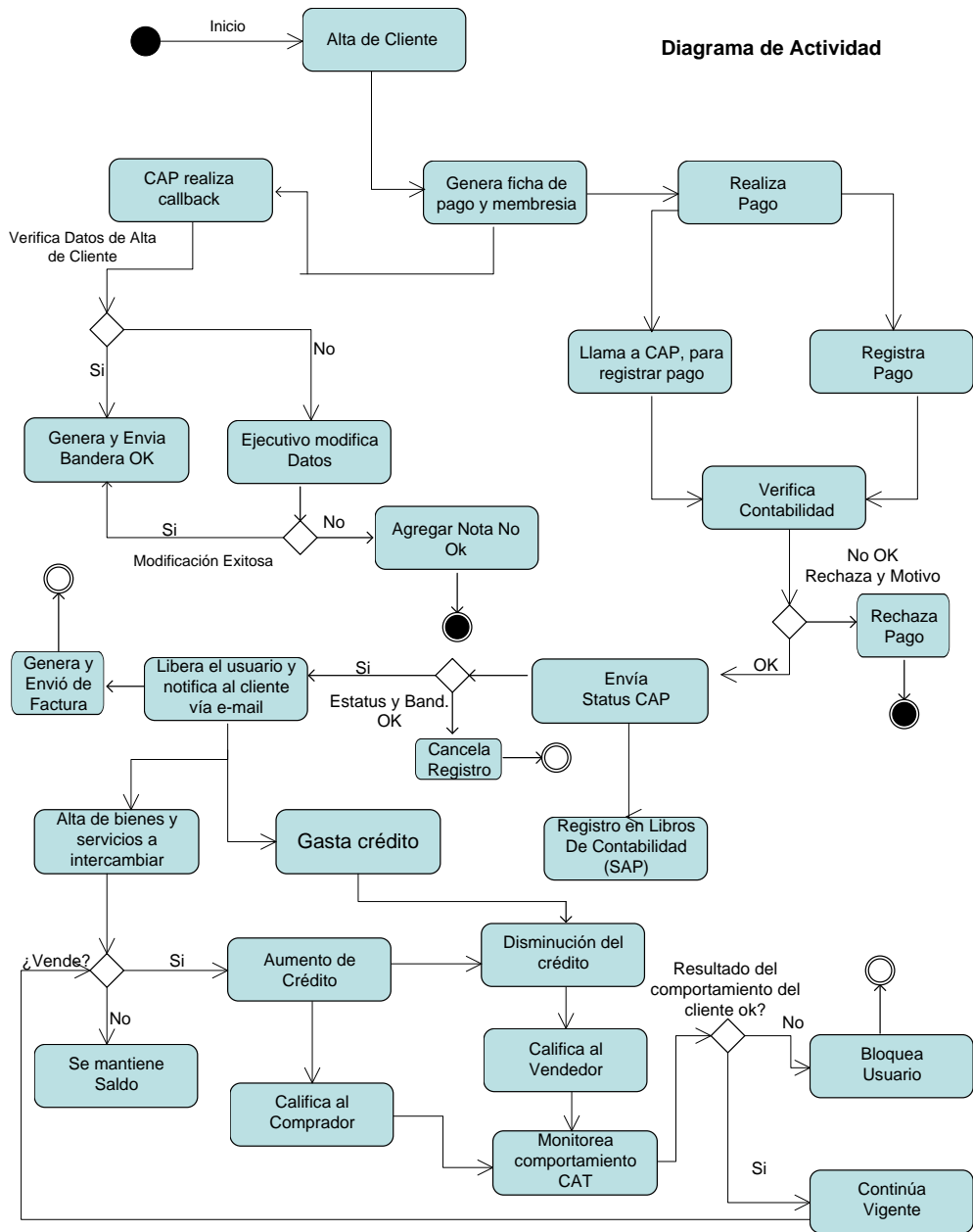


Figura 16. Diagrama de Actividad

Este diagrama tiene un estado inicial, final y las validaciones requeridas para pasar a otra actividad, de lo contrario se mantendrán hasta donde cumpla las condiciones.

Cada una de las actividades puede tener un detalle muy específico de tal forma que se puede aplicar la técnica top down, es decir, la descomposición de los procesos de negocio ha sido uno de los mejores enfoques que aún se usa para

representar la modularidad de las funciones o bien de los procesos, el enfoque top down enfatiza la planificación.

La modelo descendente consiste en efectuar una relación entre las sucesivas etapas de estructuración de modo que exista una relación entre ellas mediante entradas y salidas de información a un nivel más detallado.

Un ejemplo de descomposición del proceso de recuperación de clientes se muestra a continuación:

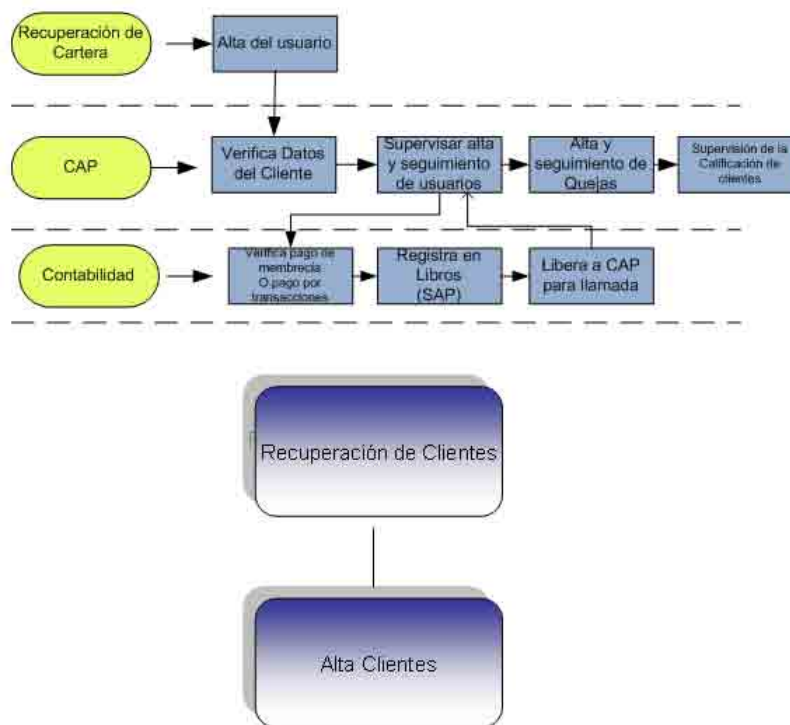


Figura 17. Descomposición de procesos

Con la descomposición de un proceso se obtienen los subprocesos, con el diagrama de actividad no es posible representar los subprocesos y cada actividad del subproceso es parte del proceso principal, empleando herramientas para el modelado de procesos de negocio BPM (Bizagi), es posible representar el esqueleto del diagrama general y los subprocesos derivado de la descomposición (top down).

Realizar el análisis y diseño de los procesos de negocio en BPM, el diagrama de la figura 18 proporciona una visión de los flujos de trabajo y los mensajes que se intercambian entre cada interacción, facilitando el diseño de los servicios web en la vista computacional.

Ejemplo de modelado en BPM, con el diagrama general.

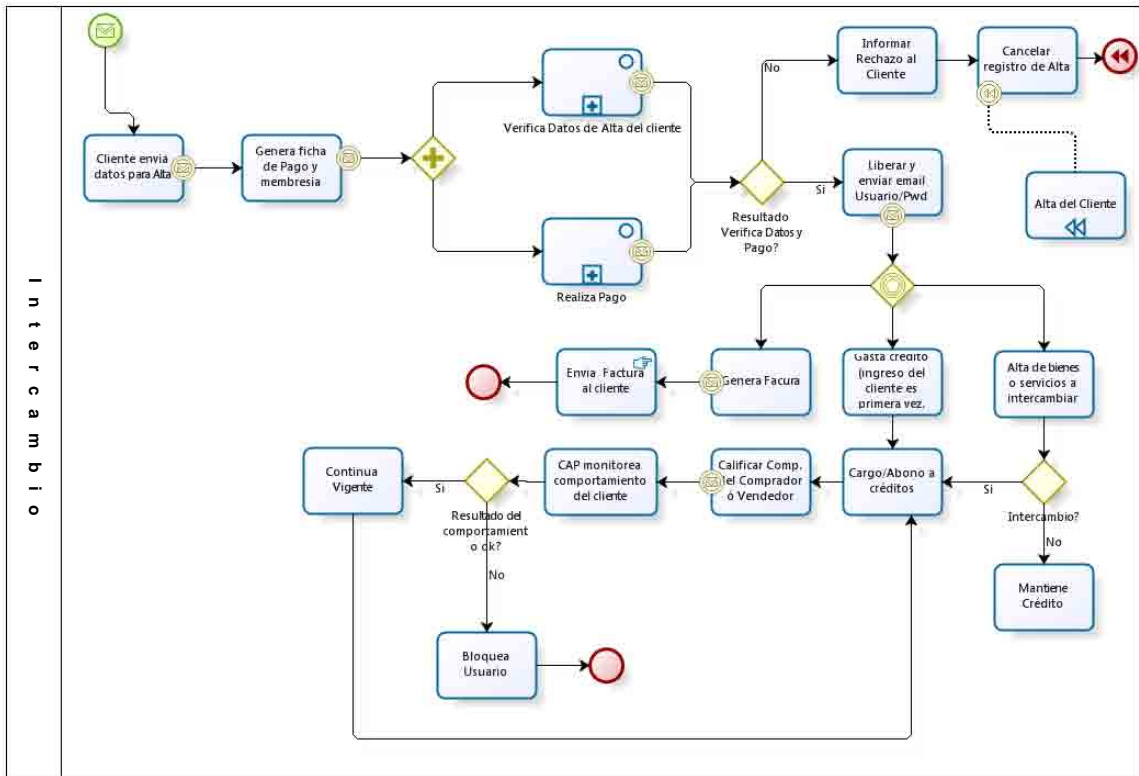


Figura 18. Diagrama en BPM

Del diagrama general se desprende un proceso verifica datos del cliente, es el subproceso que se muestra a continuación:

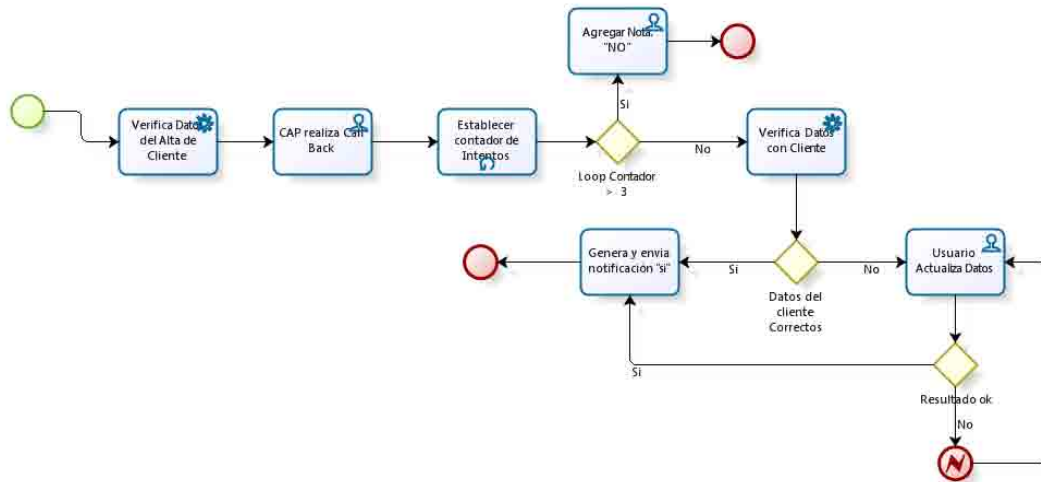


Figura 19. Subproceso Verifica datos del Cliente

Hasta este momento el modelado de la arquitectura de negocios orientada a servicios únicamente tiene los elementos de negocio esenciales que permitirán tener procesos inteligentes, sin embargo falta el contexto de SOA la parte tecnológica y computacional que permitirá implementar los servicios web conjugado con BPM, heredando la inteligencia de negocio, por lo que ambos conceptos son las piezas claves para obtener el diseño bajo el contexto de SOBA.

La transición de la vista empresarial y de información (esta última como referencia únicamente) a la vista computacional, es la fase en la cual se identifican y definen los servicios web que se emplearán para el modelado físico.

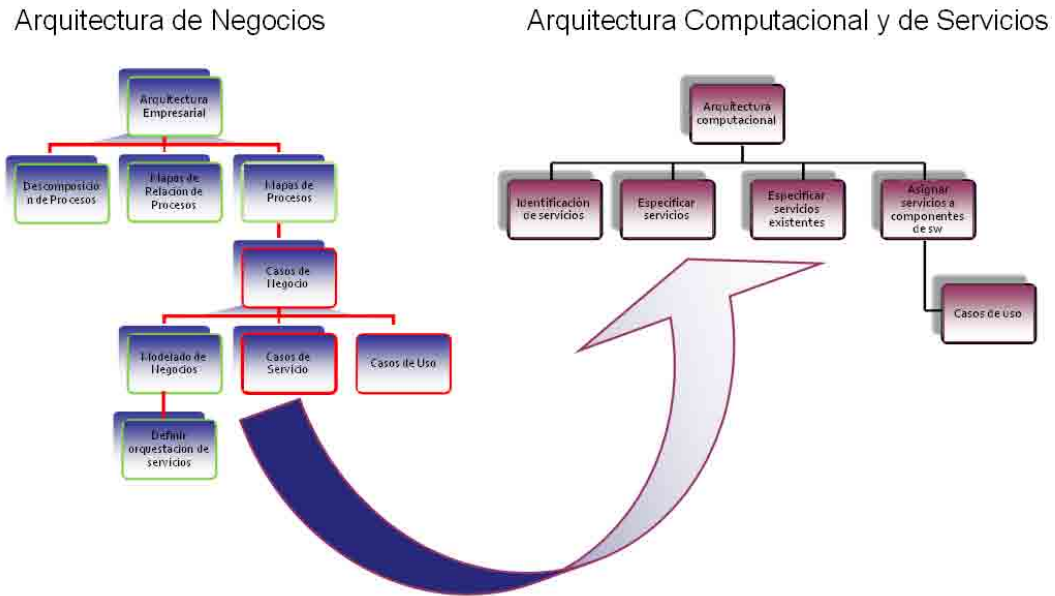


Figura 20. Transición del modelo conceptual y lógico al modelo físico.

3.2.5 Modelo Físico

La arquitectura de negocio puede mejorar la eficiencia de los procesos de negocio a través de la automatización de los mismos y en esta fase del modelado físico tiene como objetivo proveer la comprensión en términos de servicios, lo cual significa que la arquitectura debe proporcionar las funcionalidades expuestas dentro de procesos y transformarse en servicios.

Con el modelado físico se apoya la automatización de los procesos, esto significa eficientar los procesos a través de tecnologías de información y existen lenguajes que colaboren en este sentido. La automatización proveerá:

- una forma estandarizada de exponer y acceder a las funcionalidades como servicios.
- una arquitectura en la cual la comunicación y la administración de los servicios incluya intercambio de mensajes y transformación de los mismos.
- Facilitar la integración entre varios servicios existentes o nuevos usando procesos de negocio.

- Lenguaje especial para la composición de funcionalidades expuestas dentro de los procesos de negocio.

Exponer y acceder la funcionalidad de aplicaciones como servicios de forma remota es el resultado de transformar las arquitecturas en ambientes distribuidos y middleware. Los servicios web constituyen la mejor forma de exponer funcionalidad de aplicaciones con la lógica de negocios en una arquitectura distribuida, es decir la composición de servicios expuestos dentro de procesos de negocio y para ello existe un lenguaje especializado para la definición de procesos de negocio llamado BPEL, el cual provee un ambiente donde los procesos pueden ser desarrollados de una manera fácil y rápida, por lo tanto los cambios no requieren mucho esfuerzo.

3.2.5.1 Modelado empleando BPEL

BPEL significa Lenguaje de Ejecución de Procesos de Negocio con Servicios Web, el principal objetivo es estandarizar la automatización de los procesos entre servicios. Se definen los procesos de negocios que hacen uso de servicios y procesos de negocio que externalizan sus funciones como servicios. BPEL fue estandarizado por OASIS (Organization for the Advancement of Structured Information Standard) y representa una convergencia de dos lenguajes de flujo de trabajo, WSFL (Web Service Flow Language) y XLANG. WSFL fue diseñado por IBM y es basado en el concepto de gráficos dirigidos y XLANG fue diseñado por Microsoft, BPEL combina ambos y provee un vocabulario amplio para la descripción de procesos de negocio (Salter & Jennings, 2009).

BPEL usa el vocabulario base de XML para especificar y describir procesos de negocio. La versión 1.1 es basada en WSDL 1.1, esquema XML 1.0, y las especificaciones de XPath.

Empleando BPEL como el lenguaje para hacer las especificaciones dentro de la vista computacional podemos obtener:

- Descripción lógica de procesos de negocio a través de la composición de servicios.
- Manejo de invocación de operaciones de modo sincrónico y asíncrono.
- Invocación de las operaciones de servicios en secuencia o en paralelo.
- Manejo de compensación (funcionalidad que también expone BPM) y administración de fallas.
- Redireccionamiento de mensajes entrada a los procesos o actividades apropiados.
- Estructura de procesos de negocio.

La composición de servicios en BPEL dentro de los procesos de negocio requieren la definición de actividades de colaboración y mensajes de intercambio de datos entre los servicios. El WSDL (Lenguaje de Descripción de Servicios Web) proveerá la técnica básica para la descripción y especificación de los mensajes los cuales serán intercambiados, la mayor parte de estos mensajes ya se conocen con el apoyo del diagrama de BPM.

BPM toma un rol importante en el modelado de procesos de negocio y proporciona un modelo de ejecución abstracta para describir colaboraciones y transacciones. Se obtiene un modelo formal para expresar procesos abstractos (se especifican los mensajes públicos entre las partes, no incluyen detalles internos de flujo de procesos) y ejecutables (se especifica el detalle exacto de los procesos de negocio y siguen el paradigma de orquestación).

Comparando BPEL y BPM muestra que ambos comparten un origen similar en términos de servicios web y otras especificaciones, particularmente seguridad, coordinación; BPM, sin embargo, modela procesos complejos y transacciones de compensación.

La especificación, definición y desarrollo de los servicios con BPEL requerirán el uso de las tecnologías que soporta actualmente el lenguaje, los servicios web quedarán almacenados en la UDDI.

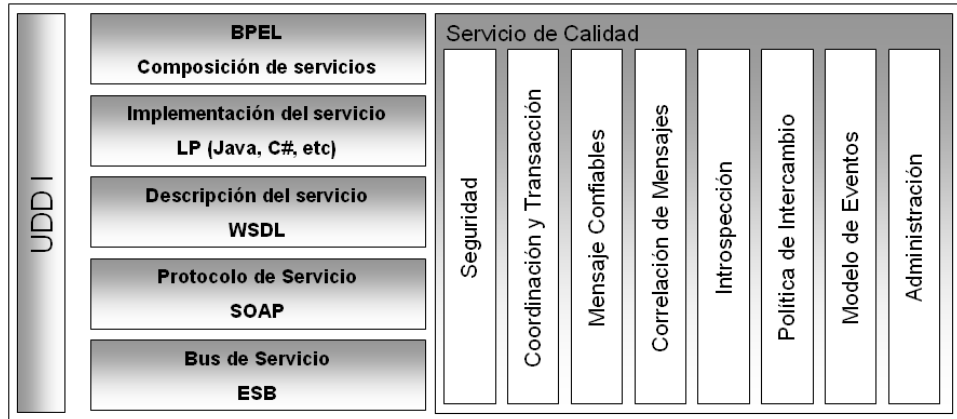


Figura 21. Composición BPEL. (Matjaz, 2006)

Las características previamente descritas son parte importante de lo que necesitamos que la vista computacional tenga dentro del alcance, pero una de las características más importantes de BPEL es la composición, la cual puede dirigir procesos públicos o privados y existen dos términos que se usan para este fin:

La Orquestación: un proceso central (el cual puede ser otro servicio web), toma el control sobre los servicios web y coordina la ejecución de las diferentes operaciones que se encuentran en los servicios web, esto es hacer un par de requerimientos de orquestación, el servicio web implicado no conoce al resto y no necesita conocerlo, por que ellos están dentro de la composición y forman parte de los procesos de alto nivel. Solo el coordinador central de la orquestación conoce esto, así que la orquestación es centralizar con definiciones explícitas de operación y ordenes de invocación de servicios web, La orquestación usualmente se utiliza en procesos privados y semánticamente se definen, como se muestra en la figura 22.

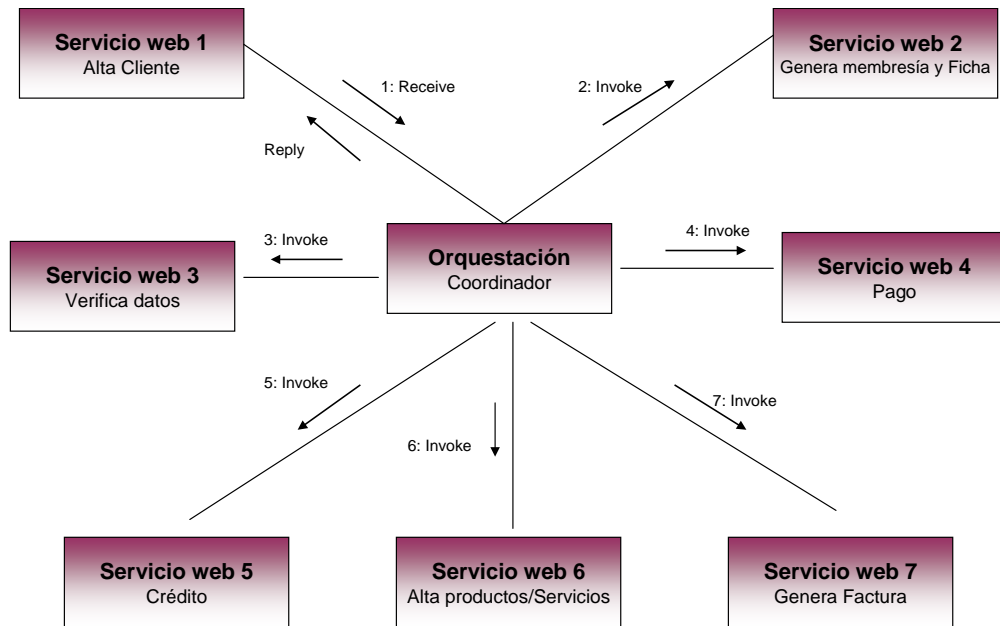


Figura 22. Orquestación de servicios web

Existe el término de coreografía, es un esfuerzo de colaboración enfocado en el intercambio de mensajes en procesos de negocio públicos, todos los participantes de la coreografía están conscientes de los procesos de negocio, operaciones a ejecutar, mensajes a intercambiar y el tiempo de intercambio de mensajes.

Desde la perspectiva de composición de servicios web y los procesos de negocio a ejecutar, la orquestación tiene una ventaja sobre la coreografía. La orquestación es un paradigma más flexible:

Ventajas:

- Se conoce quien es exactamente el responsable de la ejecución de los procesos de negocio.
- La incorporación de servicios web, no es necesario que otros conozcan que ahora son parte del proceso de negocio.

BPEL soporta coreografía y orquestación a través de procesos de negocio abstractos y ejecutables, por lo tanto el proceso de negocio central o coordinador para nuestro caso es "Intercambio", siendo este un proceso ejecutable.

3.2.5.2 Conceptos Centrales para el modelado del proceso de Negocio en BPEL.

Un proceso de BPEL consiste de pasos y cada uno es llamado actividad, BPEL soporta actividades básicas y estructuradas, las actividades básicas son usadas como actividades comunes:

Algunas de estas actividades son:

- Invocar a otros servicios web, se usa <invoke>
- Esperando para que el cliente invoke los procesos de negocio a través del envío de mensajes, se usa <receive>.
- Generando una respuesta para sincronizar operaciones, emplea <reply>.
- Manipulación de variables, usando <assign>
- Indicador de fallas y excepciones <throw>
- Terminar el proceso completo, <terminate>

Es posible combinar actividades básicas y definir algoritmos complejos y especificar exactamente los pasos de un proceso de negocio. La combinación soporta algunas estructuras, por ejemplo:

- Secuencia, para definir un conjunto de actividades que invocarán en una secuencia ordenada. <sequence>.
- Flow, para definir un conjunto de actividades que invocarán en paralelo.
- Constructor Case-switch, para implementar ramificaciones, <switch>
- While para definir ciclos condicionales.
- Pick, habilita la selección de una de varias rutas como alternativas <pick>.

Otros componentes que es necesario en la construcción del proceso de negocio y BPEL son los partners links, los links de las interacciones de BPEL son llamados

partner links o algunas veces llamados invoke partner link, cada proceso BPEL tiene al menos un partner link, porque tiene que hacer la primera invocación a los servicios.

Es importante entender que los tipos de partner link no son parte de los procesos BPEL, el lugar de ellos se encuentra descrito en el WSDL y usan este documento como mecanismo de extensibilidad.

Parte del código del proceso de negocio “Intercambio” luce de la siguiente forma:

```

<?xml version="1.0" encoding="UTF-8"?>
<process
  name="BP_Intercambio"
  targetNamespace="http://enterprise.netbeans.org/bpel/BPEL_Acambio/BP_Intercambio"
  xmlns="http://docs.oasis-open.org/wsbpel/2.0/process/executable"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:sxt="http://www.sun.com/wsbpel/2.0/process/executable/SUNExtension/Trace"
  xmlns:sxed="http://www.sun.com/wsbpel/2.0/process/executable/SUNExtension/Editor"
  xmlns:tns="http://enterprise.netbeans.org/bpel/BPEL_Acambio/BP_Intercambio">
  <import namespace="http://enterprise.netbeans.org/bpel/AltaWSWrapper" location="Partners/AltaWS/AltaWSWrapper.wsdl" importType="http://schema
  <import namespace="http://alta.acambio.org/" location="Partners/AltaWS/AltaWS.wsdl" importType="http://schemas.xmlsoap.org/wsdl/" />
  <import namespace="http://j2ee.netbeans.org/wsdl/WSDL_AltaCliente" location="WSDL_AltaCliente.wsdl" importType="http://schemas.xmlsoap.org/ws
  <import namespace="http://enterprise.netbeans.org/bpel/GeneraPagoWBWrapper" location="Partners/GeneraPagoWB/GeneraPagoWBWrapper.wsdl" importT
  <import namespace="http://alta.acambio.org/" location="Partners/GeneraPagoWB/GeneraPagoWB.wsdl" importType="http://schemas.xmlsoap.org/wsdl/"
  <import namespace="http://j2ee.netbeans.org/wsdl/WSDL_GeneraPM" location="WSDL_GeneraPM.wsdl" importType="http://schemas.xmlsoap.org/wsdl/" />
  <partnerLinks>
  <variables>
  <sequence>
</process>

```

Figura 23. Código fuente del proceso principal de negocio

Estructura de los partner link dentro del proceso BPEL:

```

<partnerLinks>
<partnerLink name="BPEL_AltaIntercambio" xmlns:tns="http://j2ee.netbeans.org/wsdl/WSDL_AltaCliente" partnerLinkType="tns:WSDL_AltaCliente" myRole="WSDL_AltaClientePortTypeRole"/>
<partnerLink name="WS_Alta" xmlns:tns="http://enterprise.netbeans.org/bpel/AltaWSWrapper" partnerLinkType="tns:AltaWSLinkType" partnerRole="AltaWSRole"/>
<partnerLink name="WS_GeneraPagyMem" xmlns:tns="http://enterprise.netbeans.org/bpel/GeneraPagoWBWrapper" partnerLinkType="tns:GeneraPagoWBLinkType" partnerRole="GeneraPagoWBRole"/>
</partnerLinks>

```

Figura 24. Código de Partner Link

Vista WSDL

70


```

<?xml version="1.0" encoding="UTF-8"?>
<definitions name="WSDL_AltaCliente" targetNamespace="http://j2ee.netbeans.org/wsdl/WSDL_AltaCliente"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:tns="http://j2ee.netbeans.org/wsdl/WSDL_AltaCliente" xmlns:ns="http://alta.acambio.org">
  <types>
    <xsd:schema targetNamespace="http://j2ee.netbeans.org/wsdl/WSDL_AltaCliente">
      <xsd:import namespace="http://alta.acambio.org/" schemaLocation="Partners/AltaWS/AltaWS_xsd_1.xsd"/>
    </xsd:schema>
  </types>
  <message>
  </message>
  <portType name="WSDL_AltaClientePortType">
    <operation name="WSDL_AltaClienteOperation">
      <input name="input1" message="tns:WSDL_AltaClienteOperationRequest"/>
      <output name="output1" message="tns:WSDL_AltaClienteOperationResponse"/>
    </operation>
  </portType>
  <binding>
  </binding>
  <service name="WSDL_AltaClienteService">
    <port name="WSDL_AltaClientePort" binding="tns:WSDL_AltaClienteBinding">
      <soap:address location="http://localhost:$(HttpDefaultPort)/WSDL_AltaClienteService/WSDL_AltaClientePort"/>
    </port>
  </service>
  <plink:partnerLinkType>
  </plink:partnerLinkType>
</definitions>

```

Figura 25. Código del documento WSDL

Estructura actividades básicas y compuestas:

```

<variables>
  <variable name="WSDL_AltaClienteOperationOut" xmlns:tns="http://j2ee.netbeans.org/wsdl/WSDL_AltaCliente" messageType="tns:WSDL_AltaClienteOperationResponse"/>
  <variable name="AltaClienteOut" xmlns:tns="http://alta.acambio.org/" messageType="tns:AltaClienteResponse"/>
  <variable name="AltaClienteIn" xmlns:tns="http://alta.acambio.org/" messageType="tns:AltaCliente"/>
  <variable name="WSDL_AltaClienteOperationIn" xmlns:tns="http://j2ee.netbeans.org/wsdl/WSDL_AltaCliente" messageType="tns:WSDL_AltaClienteOperationRequest"/>
</variables>

<sequence>
  <receive name="ReceiveFromClienteAlta" createInstance="yes" partnerLink="BPEL_AltaIntercambio" operation="WSDL_AltaClienteOperation" xmlns:tns="http://j2ee.netbeans.org/wsdl/WSDL_AltaCliente" portType="tns:WSDL_AltaClientePortType" variable="WSDL_AltaClienteOperationIn"/>

  <assign name="Assign1">
    <copy>
      <from>$WSDL_AltaClienteOperationIn.InputAlta/nombre</from>
      <to>$AltaClienteIn.parameters/nombre</to>
    </copy>
  </assign>

  <invoke name="Invoke_WSAltaCliente" partnerLink="WS_Alta" operation="AltaCliente" xmlns:tns="http://alta.acambio.org/" portType="tns:AltaWS" inputVariable="AltaClienteIn" outputVariable="AltaClienteOut"/>

  <assign name="Assign2"></assign>

  <reply name="Reply_WSAltaCliente" partnerLink="BPEL_AltaIntercambio" operation="WSDL_AltaClienteOperation" xmlns:tns="http://j2ee.netbeans.org/wsdl/WSDL_AltaCliente" portType="tns:WSDL_AltaClientePortType" variable="WSDL_AltaClienteOperationOut"/>
  <assign name="Assign4"/>

  <flow name="Flow1">
    <receive name="Receive1"/>
  </flow>
</sequence>

```

Variables necesarias para solicitar y responder SW

Combinación de actividades básicas para formar una actividad estructurada

Figura 26. Actividades Estructuradas

Durante el diseño físico es importante considerar los conceptos básicos que ayudan a entender de mejor forma la lógica que debe llevar el proceso de negocio para la composición de los servicios web con BPEL.

BPEL es un lenguaje basado en XML para definición de procesos de negocio, cada proceso tiene un conjunto de actividades e interacciones con colaboradores de servicio web, por lo que el proceso central que se encarga de la orquestación es también un servicio web.

En esta etapa se han definido las actividades para invocar operaciones, recibir mensajes y regresar una respuesta al cliente. La arquitectura completa muestra los procesos diseñados en BPM no solo como procesos de negocio sino estos se encuentran ahora en el contexto SOA, por lo que la arquitectura SOBA se muestra así:

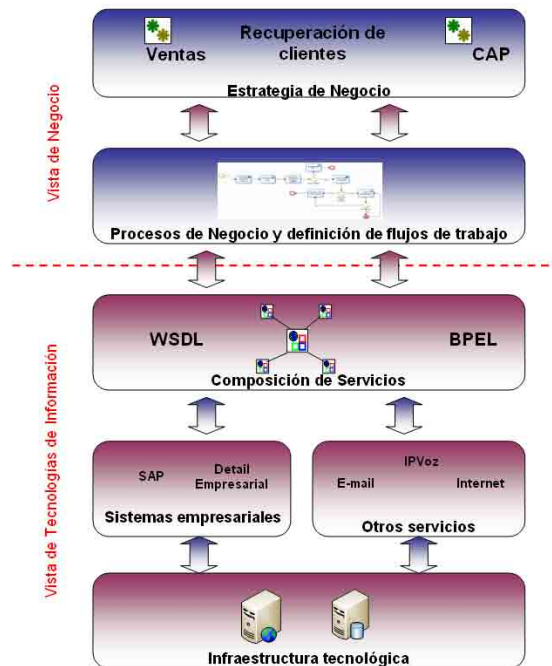


Figura 27. Propuesta de SOBA

Para la construcción de la arquitectura SOBA se pueden emplear algunas herramientas que soportan BPEL, SOA, servicios web, Aplicaciones empresariales (EJB), algunas de ellas son Oracle BPEL process manager and BPEL designer, MS Biztalk server de Microsoft y JDeveloper BPEL Designer, sin embargo para ocupar estas se requiere el licenciamiento adecuado, pero existen otras alternativas de software libre que cubren los requisitos mínimos para desarrollar los conceptos descritos en este trabajo, por lo que emplearemos la herramienta de NetBeans 6.5.1 la cual se puede descargar en la siguiente página web <http://www.netbeans.org/>.

Capítulo 4 Construcción de SOBA-UIA.

En este capítulo se proveerá una vista rápida de la aplicación de los conceptos que se han manejado a lo largo del desarrollo de este trabajo empleando las funcionalidades disponibles en la plataforma de NetBeans.

NetBeans tiene herramientas de SOA, Entorno de desarrollo integrado (IDE), diseñador BPEL, paleta de actividades básicas y estructuradas, mapeo BPEL y editores de esquemas XML entre otras funcionalidades propias del desarrollo en java EE de las cuales se harán uso.

4.1 *Qué es NetBeans*

Es una plataforma de desarrollo de aplicaciones empresariales, permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases (Salter& Jennings, 2009). Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente.

NetBeans viene con capacidades de SOA e incluye el servidor de aplicaciones GlassFish, la herramienta de SOA de NetBeans contiene características de código abierto de java enterprise con una nueva característica para crear composición de aplicaciones, BPEL, servicios web, servicios web Java EE y XML, esquemas XML y WSDL con estas herramientas es posible desarrollar la arquitectura de SOBA y durante la construcción tener disponibles:

- **Diseño Gráfico:** Se podrá visualizar de forma gráfica durante la etapa de diseño el esquema XML y la orquestación de los servicios basada en BPEL usando un diagrama de flujo como interfase.
- **Pruebas (test).** Se podrá realizar la prueba de composición de la aplicación sin dejar el ambiente de diseño y desarrollo, es decir, es una interfaz simple para construir, probar y desarrollar.

- Motor integrado de BPEL. Los servidores para compilar los servicios web están integrados y administrados desde el IDE y el motor de BPEL esta disponible como Motor de servicios JBI.

Es necesario entender algunos conceptos de aplicaciones SOA, procesos BPEL y aplicaciones JBI:

- Integración de negocios Java (JBI). Derivado de los problemas de integración de sistemas (EAI), y protocolos de comunicación cerrados es necesario integrar componentes proveedores o consumidores de de servicios.
- Motor de servicios es un componente, el cual provee o consume lógica de negocio y transforma servicio otros componentes, existen varios motores de servicios disponibles, así como el motor de servicios BPEL para la orquestación de procesos de negocio, Java EE para consumir servicios web EE.
- Componentes Binding es un componente que provee un protocolo de servicio de transporte independiente a los componentes JBI. Estos componentes se pueden ver en el documento WSDL. El componenete binding que se utilizara en este trabajo será SOAP, existen algunos otros como file BC, HTTP BC, JDBC BC, JMS BC, SMTP.

4.2 Iniciar el servidor de aplicaciones GlassFish V2.

En el navegador de Windows dentro del IDE de NetBeans, en la pestaña “Services” clic en Servers, clic derecho en GlassFish V2 y seleccionar Start.

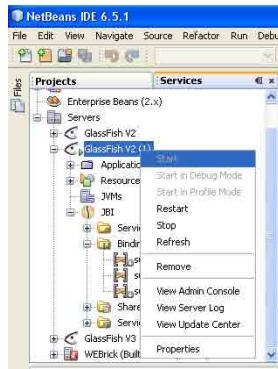


Figura 28. GlassFish

GlassFish v2 proveerá los JBI requeridos y los servicios en tiempo de ejecución, en tiempo de ejecución para aplicaciones basadas en BPEL son proporcionados por el motor de servicios de BPEL, el cual es un componente del JBI server.

El motor de servicio de BPEL se inicia junto con el servidor, ambos antes de probar y ejecutar la composición de la aplicación debe asegurarse que el GlassFish inició correctamente.

4.3 Creación de Servicios Web y WSDL

Durante la fase de diseño lógico y físico se obtuvo la definición de los servicios web, así como los mensajes participantes (estos como resultado del análisis de subprocesos de negocio), estos dos elementos deben quedar reflejados en el documento WSDL, el cual nos proveerá un estándar para la descripción de servicios, las operaciones que realizará cada servicio, todo en una plataforma y lenguaje independiente, esto permitirá que el desarrollo de aplicaciones empresariales consuman y construyan servicios débilmente acoplado, por lo que esto es el éxito de SOA.

En el documento WSDL es necesario describir la vista concreta y abstracta para describir el servicio, en la vista abstracta describes las operaciones de los servicios y en la vista concreta describe como las operaciones son mapeadas en un protocolo específico, por ejemplo SOAP.

En NetBeans se deberán de seguir los siguientes pasos para la construcción del servicio y WSDL indispensables para la diseño del proceso en el módulo BPEL.

Crear una aplicación EJB para generar el servicio web, en el IDE de NetBeans en el menú de file crear un nuevo proyecto. De la lista de categorías seleccionar el folder de Java EE y en proyectos seleccionar EJB Module.

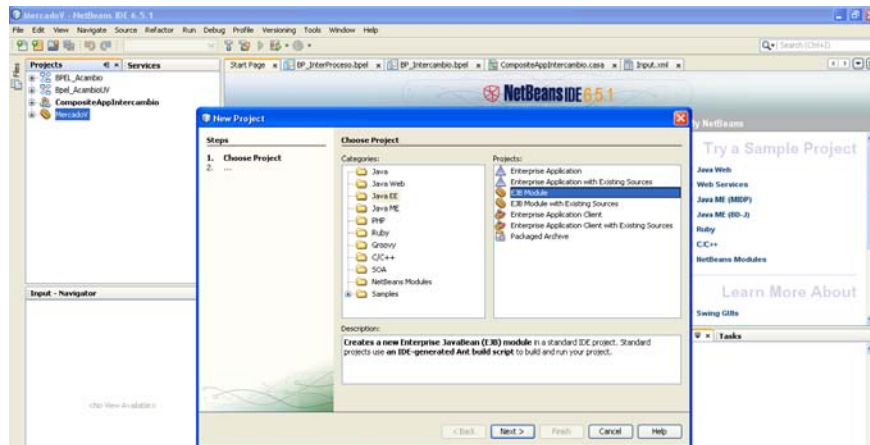


Figura 29. Creación del módulo EJB en NetBeans

Asignar nombre al módulo

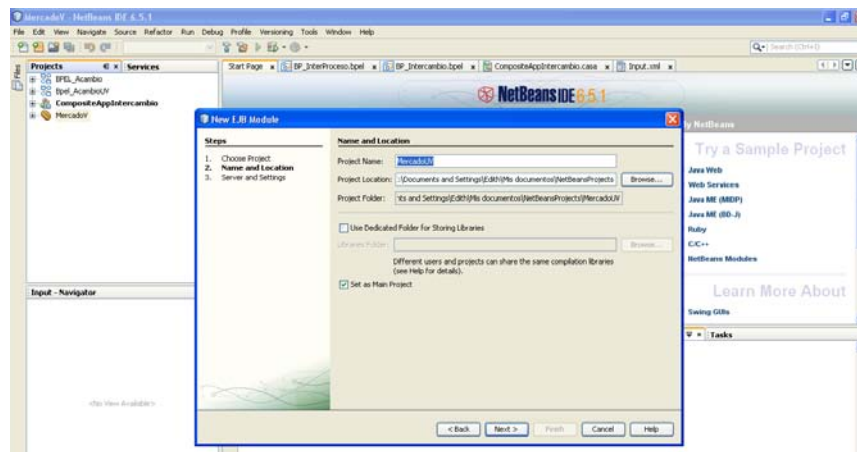


Figura 30. Nombre y configuración del módulo EJB

Ahora es necesario crear la estructura del servicio web como se muestra a continuación:

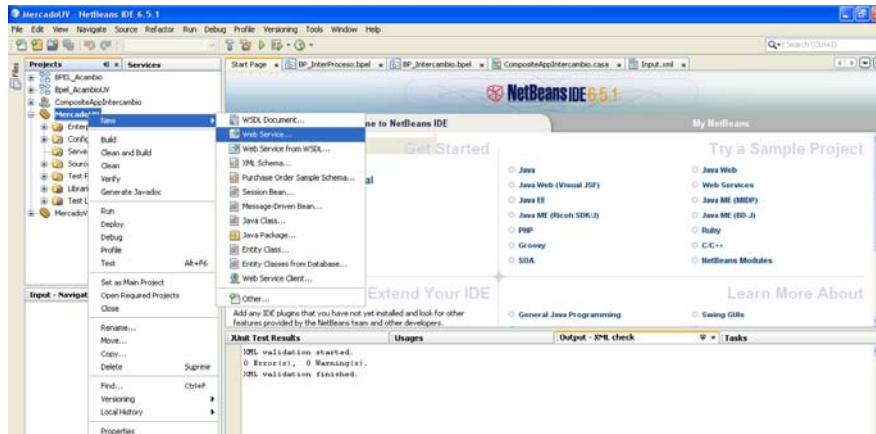


Figura 31. Generar servicio web

Se debe agregar el tipo de operación que necesitamos ejecutar el servicio, para este caso, es procesar el alta con los datos del cliente que se registrará en el sistema para el intercambio de bienes o servicios.

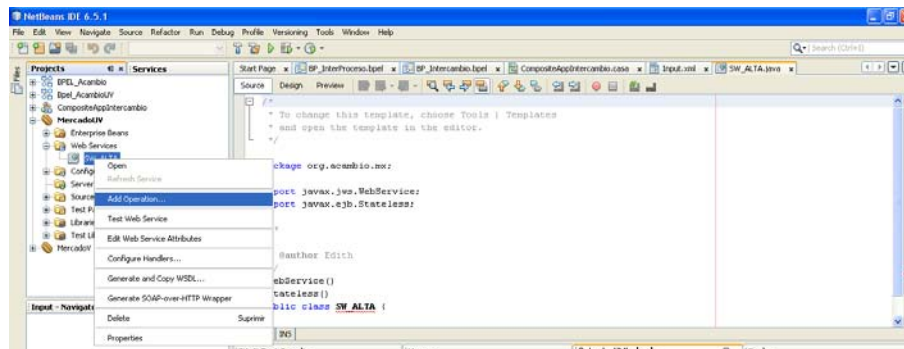


Figura 32. Agregar operación en el servicio web

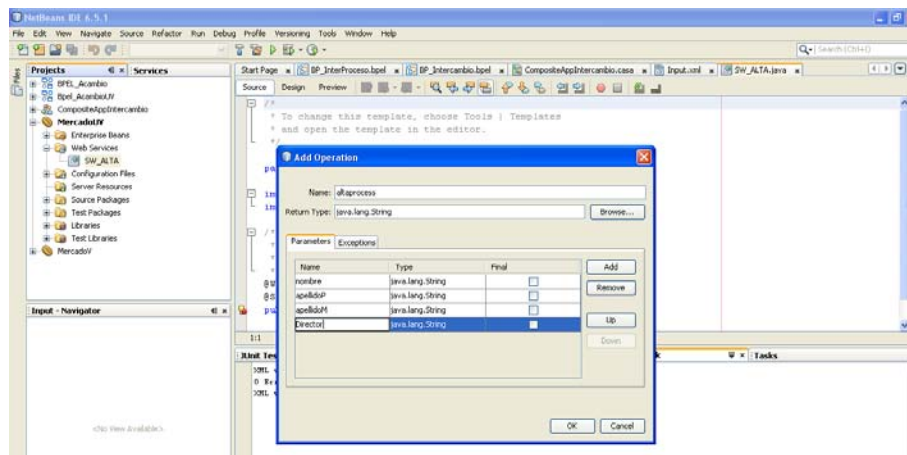


Figura 33. Campos que formarán los datos de entrada

Los campos agregados en esta sección serán aquellos que nos sirvan como variables para el intercambio de mensajes entre servicios, ahora el servicio tiene una operación como se muestra en el navegador de proyectos.

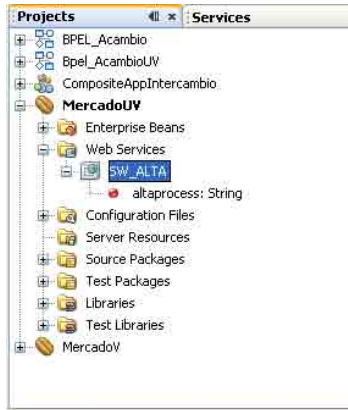


Figura 34. Vista del servicio web y operación

Finalmente para realizar la compilación en el modulo JBE seleccionar en el menú emergente deploy.

4.3.1 Lo beneficios de usar WSDL son:

- WSDL esta basado en XML, esto significa que cualquier sistema que puede leer e interpretar XML, podrá leer e interpretar los documentos WSDL.
- WSDL es independiente de la plataforma y lenguaje. No tiene dependencia de algún lenguaje de programación o plataforma. WSDL puede ser desarrollado y compilado en un ambiente Microsoft consumido por una aplicación de java y viceversa.
- Es extensible, es decir, no dependen de las comunicaciones o de un protocolo en específico.

Para crear la definición del WSDL es necesario crear el módulo BPEL

4.4 Creación del módulo BPEL.

BPEL es un ambiente de ejecución basado en XML e intenta habilitar definiciones simples o complejas de procesos de negocio, el objetivo de usar NetBeans es escribir un proceso de negocio y ejecutarlo en el mismo ambiente.

Es un tipo de proyecto en NetBeans, la herramienta de BPEL facilita un ambiente completo y habilitado para llevar acabo rápidamente y de forma eficiente la orquestación de servicios web.

Para el caso de Acambio realizar los siguientes pasos en NetBeans, Seleccionar en el menú File y submenú New Project

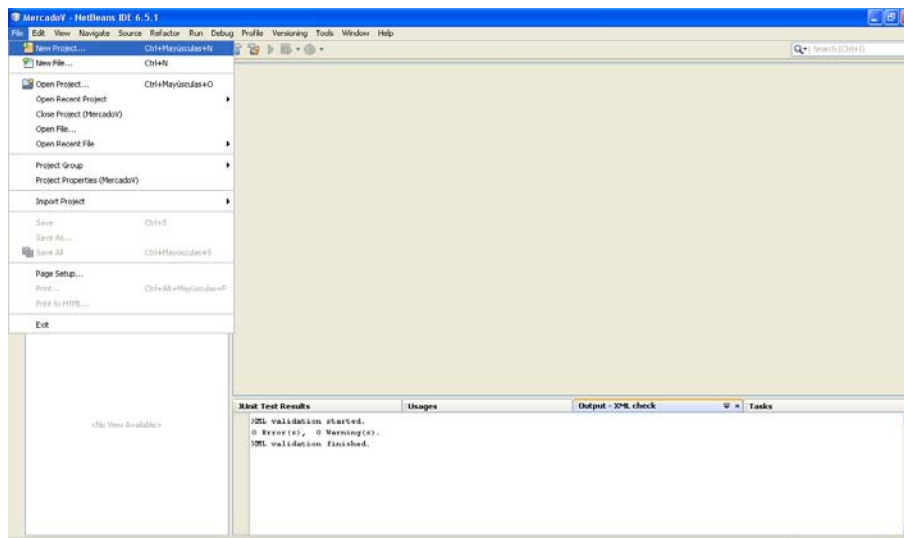


Figura 35. Creación del módulo BPEL

En la lista de Categorías seleccionar SOA y en la lista de proyectos clic en BPEL Module.

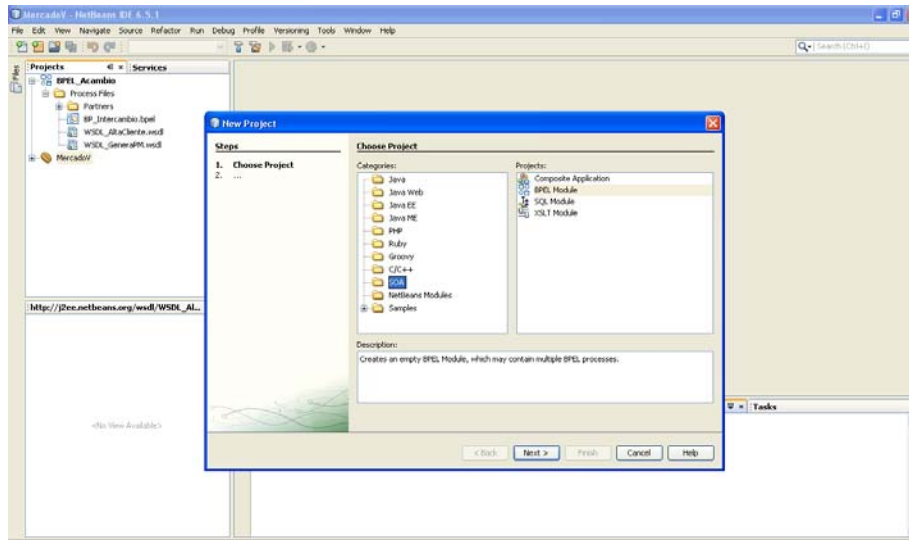


Figura 36. Selección de Categoría y Módulo BPEL

El wizard solicitará el nombre del proyecto, clic en el botón de finish y establecer como proyecto principal (verificar casilla).

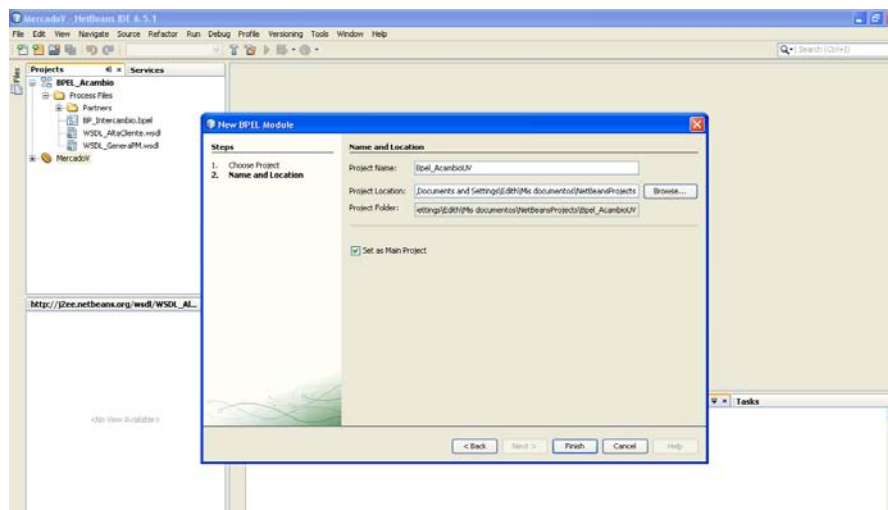


Figura 37. Determinar nombre del módulo BPEL

Para crear el archivo con el proceso central o coordinador, posicionarse en el fólder que lleva por nombre "process file" con el clic derecho del mouse seleccionar New y BPEL process.



Figura 38. Creación del documento BPEL

Una ventaja que tiene NetBeans es el wizard que tiene para crear el esqueleto cada uno de sus módulos.

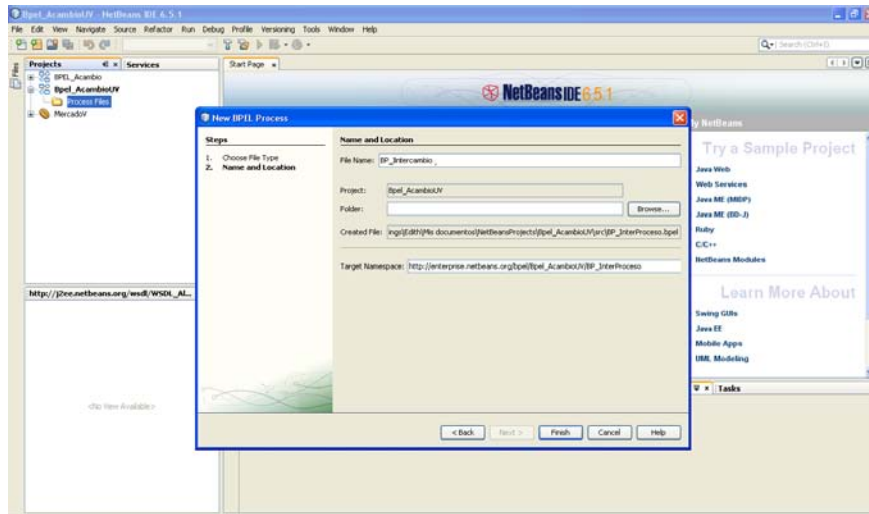


Figura 39. Asignar nombre al proceso

Al terminar de crear el documento con la extensión BP_Intercambio.bpel en el wizard, el documento se abre en la vista de diseño en el cual muestra un esqueleto de proceso en blanco para agregar la lógica del proceso de negocio.

Cuando se crea un nuevo modulo de BPEL, el IDE de NetBeans automáticamente muestra la ventana de actividades básicas y estructuradas, esta ventana consiste en las actividades que podrán ser arrastradas a la vista de diseño, la paleta esta clasificada en tres tipo, servicios web, actividades básicas y estructuradas, ver figura 40, sin embargo, también se puede elegir la opción de ir al código fuente en el menú de source en la misma vista de diseño.



Figura 40. Actividades Básicas y Estructuradas.

4.4.1 Creación de Partner Links

Ahora para iniciar el diagrama es necesario crear los partner links y de ellos se derivan de la definición del WSDL y la previa creación de los servicios web, como se muestra en la figura 41.

Se arrastrará del servicio web construido al panel del diagrama del modulo de BPEL, el cual será el servicio que invocará la actividad <invoke>.

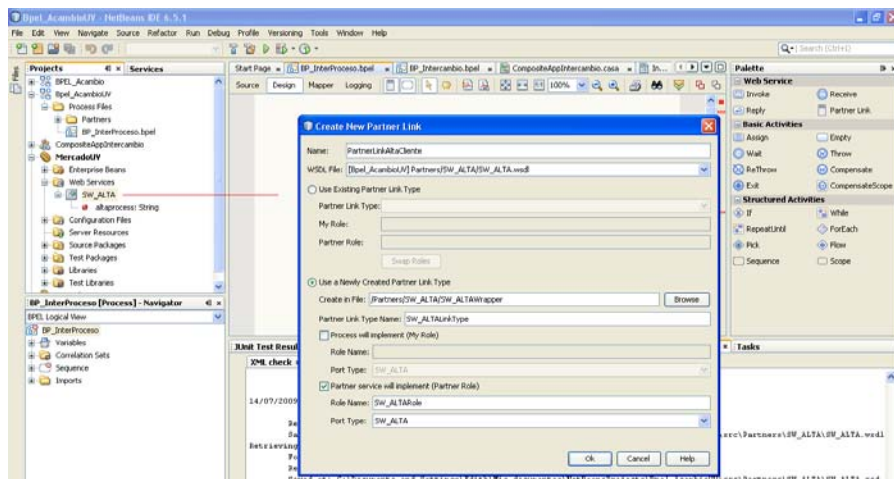


Figura 41. Configuración de un servicio web - partner link

Ahora generaremos el esqueleto del WSDL que será el partner link asociado a la actividad <receive>, para hacer esto del folder process file del menú emergente seleccionar New y WSDL document.

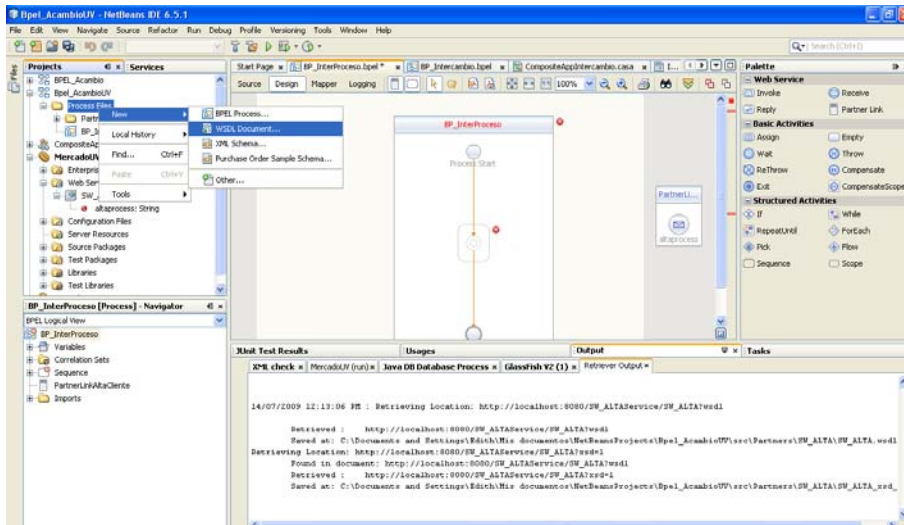


Figura 42. Creación del segundo partner link - receive

La definición del WSDL, también esta basada en el esquema de XLM (.xsd) y este se actualiza cuando compilamos el servicio web definido en el modulo EJB.

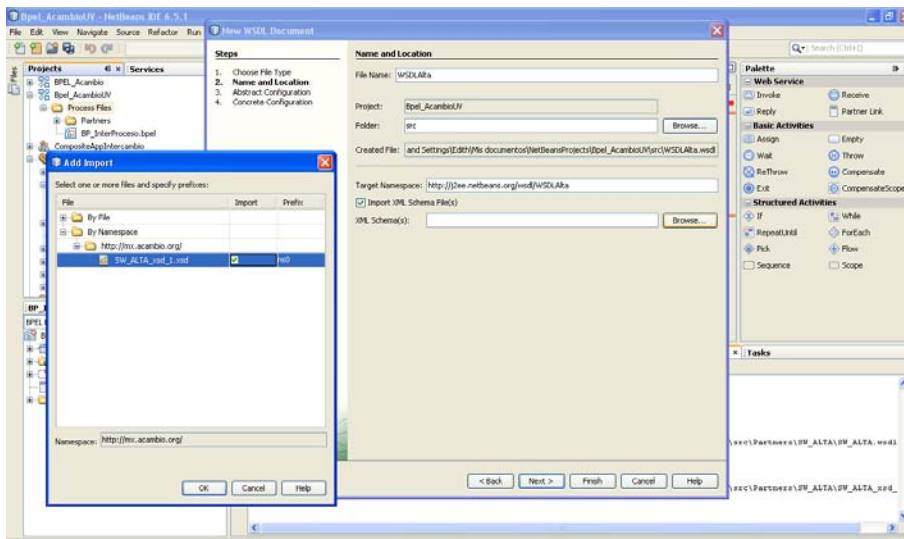


Figura 43. Documento xsd

En la configuración abstracta del WSDL, se debe especificar que tipo de operación, para este caso será petición-respuesta ver figura 44, y también se especificarán cuales serán los datos de entrada y salida.

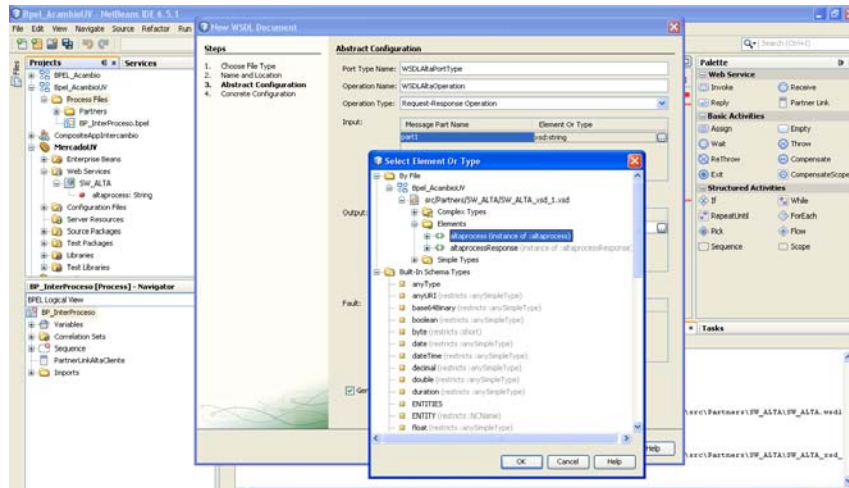


Figura 44. Definición de variables de entrada y salida.

En la configuración concreta, se definen los componentes binding, los cuales permitirán la comunicación de los mensajes mediante el protocolo SOAP.

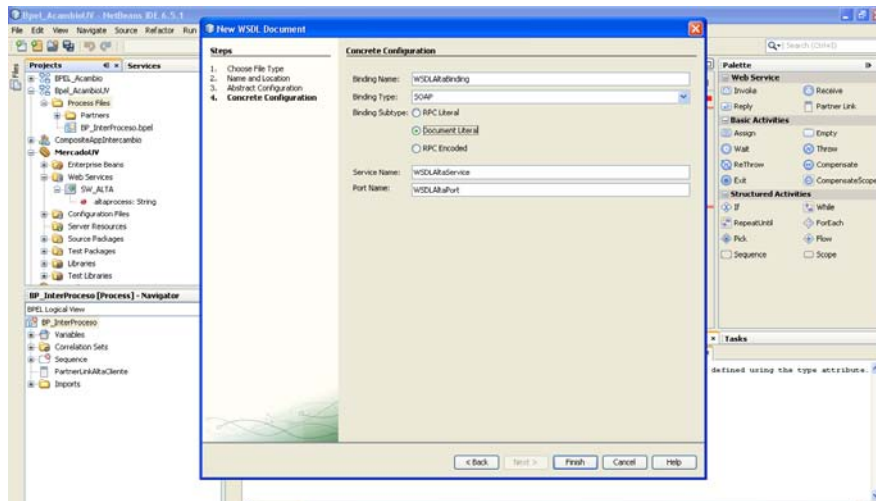


Figura 45. Configuración Concreta

Al finalizar la creación, en el navegador de proyectos el documento WSDLAlta.wsdl se encuentra creado y podrá ser visualizado en el IDE de NetBeans, ahora solo hay que arrastrar al panel del diagrama BPEL para que se genere el nuevo partner link y este estará asociado al Receive y reply.

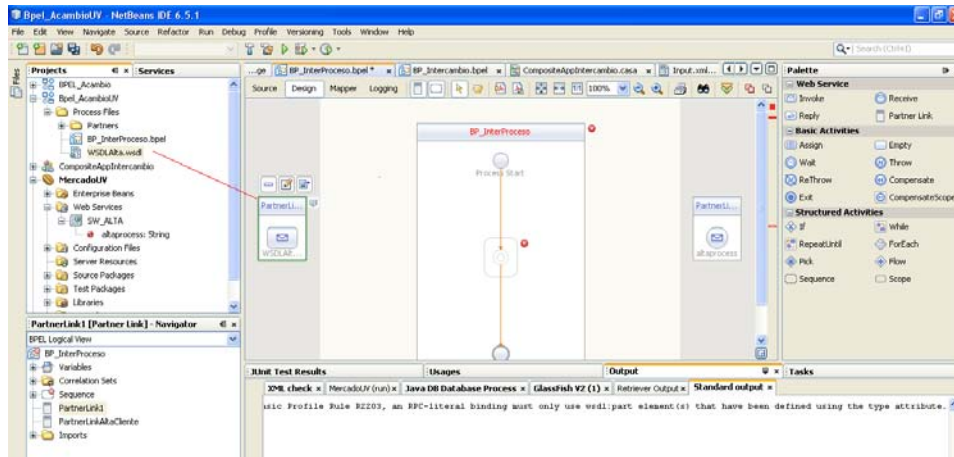


Figura 46. Partner link y servicio web a invocar

Para el diseño del proceso BPEL, en la ventana con el nombre de palette podrá iniciar el diseño agregando las actividades.

En la figura 47 se ilustra parte del proceso de Intercambio, en la parte central se encuentra la lógica de negocio, en la parte derecha dos de los servicios web, para este caso AltaCliente y GeneraMembresiaFicha, para el resto de los servicios se realizan los mismos pasos.

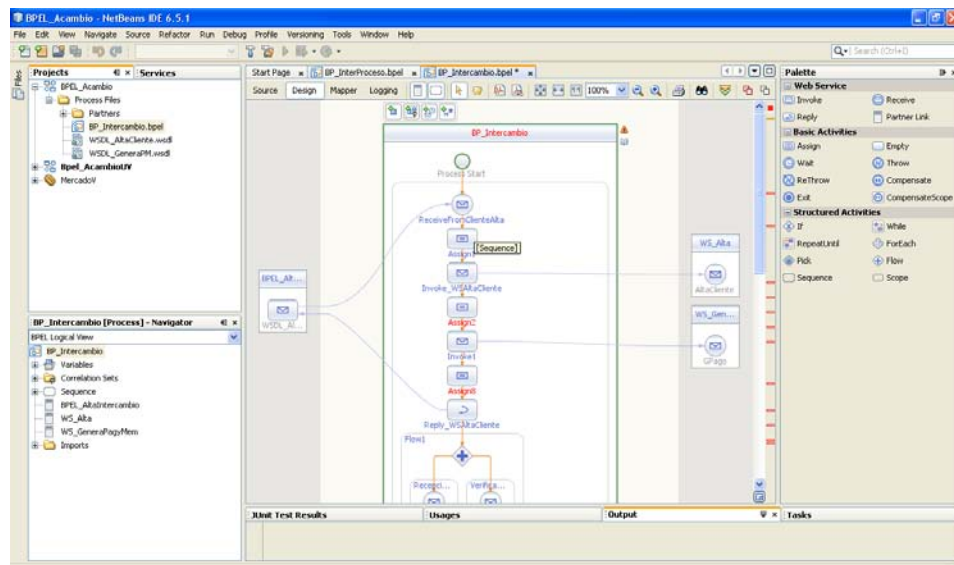


Figura 47. Proceso de negocio BPEL

Para crear la configuración correcta de Receive, Invoke y Reply para la invocación a los servicios y partner link se debe dar doble clic en cada actividad, se mostrará una ventana de diálogo para seleccionar el partner link, creación de variables de

entrada y salida según corresponda, al terminar quedarán asociados por medio de una línea en la vista de diseño.

Receive:

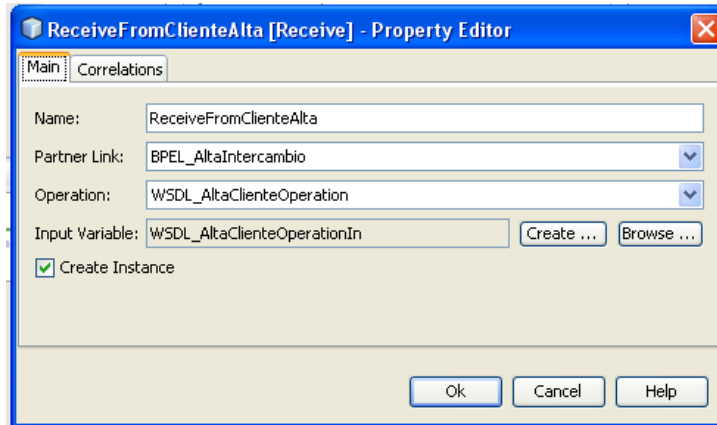


Figura 48. Configuración Receive

Invoke:

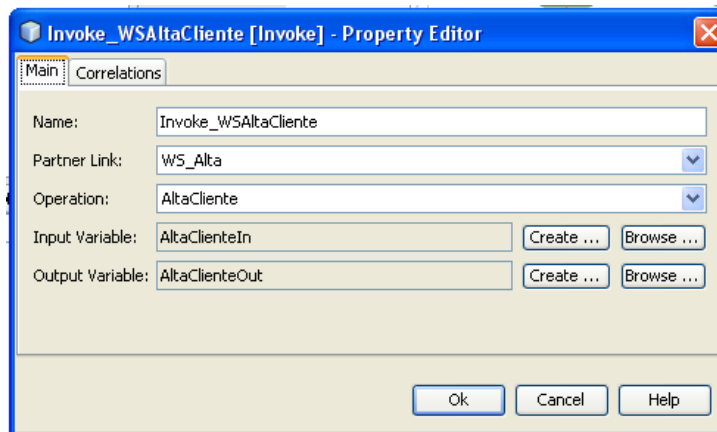
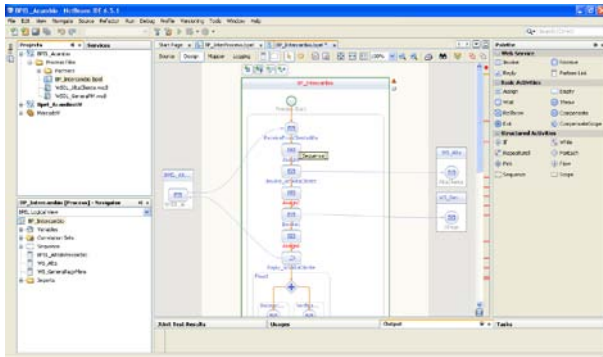


Figura 49. Configuración Invoke

NetBeans proporciona la vista de diseño y del código, adicional para realizar la validaciones o bien el intercambio de mensajes por medio de assign, esta la vista de mapper.

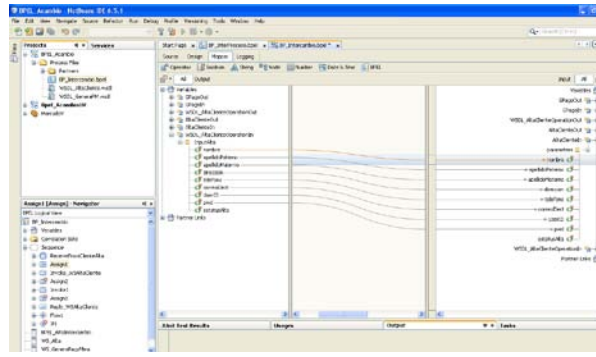
A continuación se presentan las 3 vistas para el proceso .bpel

Design



Es el diseño gráfico donde se muestra el workflow que seguirá el proceso de negocio de Acambio.

Mapper



En esta sección se realiza el paso de variables de entrada a las variables de salida, en el caso de requerir alguna validación en esta sección de mapper agregar las condicionantes.

Source (Código)

Es el documento donde se encuentra todo el código fuente XML.

Ahora el proceso BPEL esta completo, el cliente SOAP envía la respuesta mediante un mensaje al cliente, para realizar el compilado del proceso BPEL, se requiere crear una composite application. El proceso BPEL debe ser compilado como un modulo JBI, en una composite application es procesado por el motor de servicios de BPEL.

4.4.2 Crear una composite application

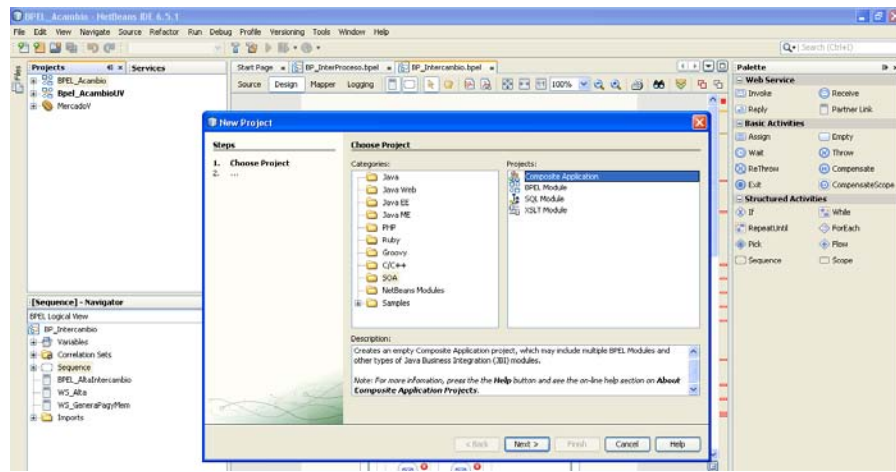


Figura 50. Creación de la composición de aplicaciones

Definir el nombre del proyecto

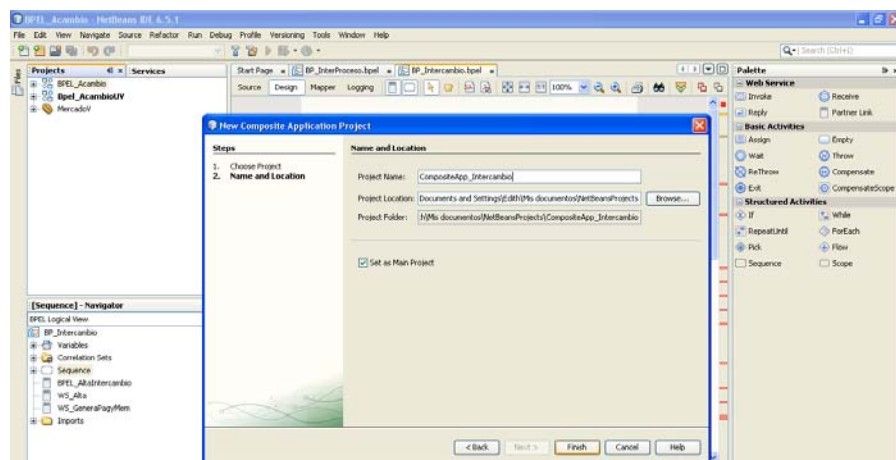


Figura 51. Definir Nombre al módulo

Después debe agregarse un módulo JBI

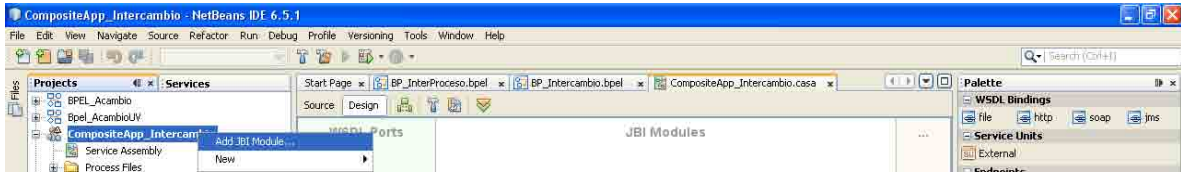


Figura 52. Agregar un módulo JBI

Mostrará un cuadro de diálogo en el cual debe seleccionar el módulo BPEL

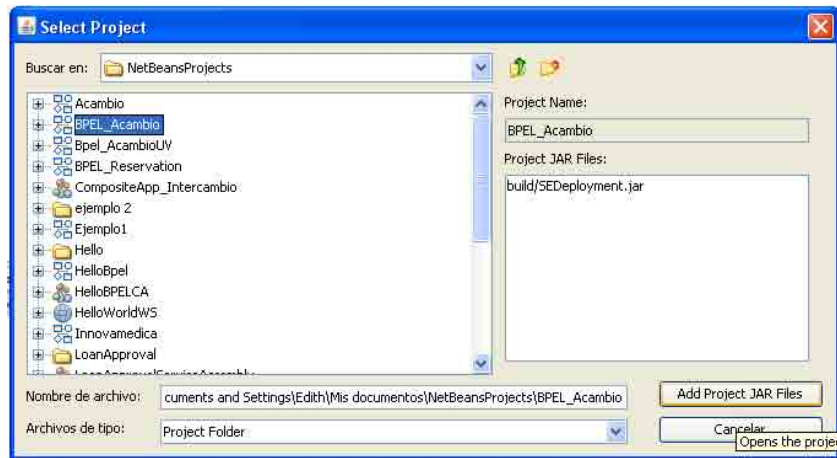


Figura 53. Seleccionar modulo BPEL para la creación del archivo .jar

Clic en el botón Add Project JAR files, de forma automática se creará el archivo .jar y se podrá ver dentro de la carpeta de JBI module.

Con el clic derecho del mouse en el módulo principal de la composición seleccionar clean and build.

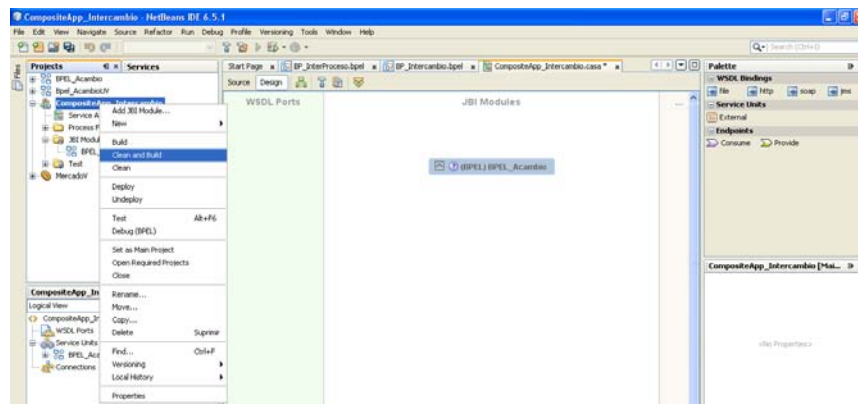


Figura 54. Compilación

Para compilar, seleccionar deploy.

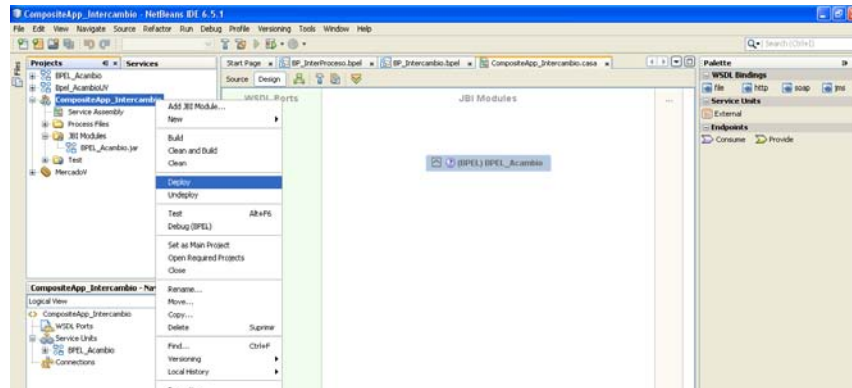


Figura 55. Compilación

Como resultado de la compilación queda de la siguiente forma:

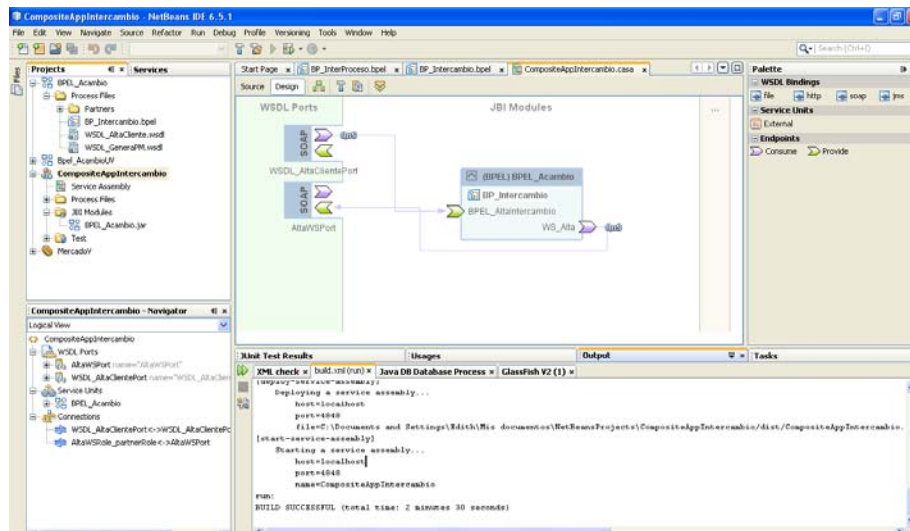


Figura 56. Resultado de la compilación

4.4.3 Crear una prueba

Ahora para probar que los mensajes de entrada y salida están funcionando adecuadamente, se realizará un caso de prueba, NetBeans dentro del IDE y la composición de la aplicación es posible llevarla acabo.

Ejemplo: Seleccionar la carpeta de test y crear un caso nuevo.

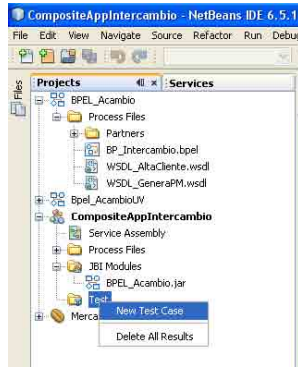


Figura 57. Crear un caso de prueba

Asignar un nombre al caso de prueba

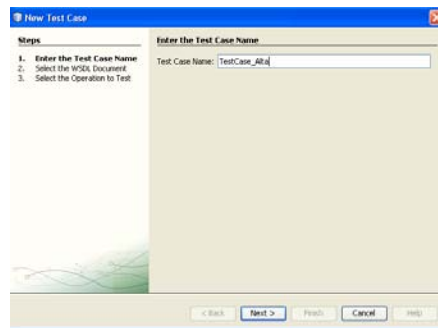


Figura 58. Nombre del caso de prueba

Para los casos de prueba se deben seleccionar las operaciones, la prueba se realizará al alta del Cliente.

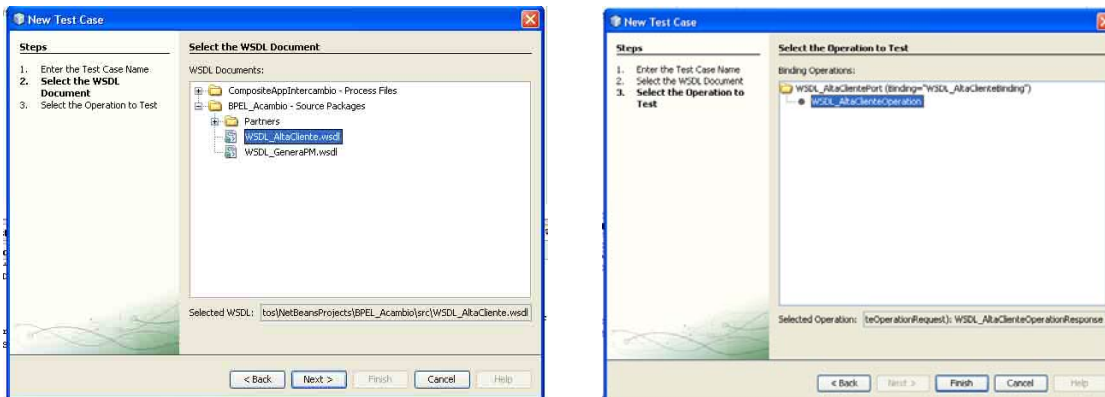


Figura 59. Selección de la definición de servicios web y operación.

Hasta este momento se tienen construidas dos piezas clave dentro la arquitectura, en donde el proceso de negocio ahora esta en el contexto de servicios web.

Usando BPEL para la definición del proceso de negocio tiene las siguientes ventajas:

- Administración hasta el punto final: BPEL introduce el concepto de partner links que puede ser definidos durante el tiempo de diseño, los partner links son el concepto de las primeras clases en BPEL, y pueden ser manipuladas directamente en el proceso.
- Abstracción de alto nivel: BPEL provee una abstracción de alto nivel, de tal forma que el o los analistas del negocio pueden componer y ejecutar procesos en un ambiente de modelado amigable.
- WSDL. La definición de las características técnicas de un servicio desde un punto de BPEL. Cada mensaje intercambiado describe en un proceso BPEL en términos de Port types y operaciones las cuales son definidas en WSDL. BPEL no asume que los servicios acceden vía SOAP o sobre HTTP.
- Minimizar complejidad. La complejidad seguirá siendo parte inevitable de la tecnología. El manejo y la reducción al mínimo de esta complejidad es el objetivo primario de BPEL.

4.5 Análisis de los componentes de SOBA que permiten la flexibilidad e Integración.

El elemento clave dentro de la arquitectura propuesta es la integración de BPM para el análisis y optimización de procesos de negocio de tal forma que se lleven a lo que hoy ya se conoce como SOA, de esta forma se tienen servicios web inteligentes, es decir que tienen la lógica del negocio para soportar las necesidades del entorno interno o externo.

Al diseñar un proceso de negocio en BPEL, este automáticamente se convierte también en un servicio web, que tiene centralizado el control y permite cambios, por lo que la orquestación permite la flexibilidad en cambios, ya que el cambio se desarrolla en el proceso central de la lógica y los servicios web tiene la

características de tener una funcionalidad débilmente acoplada para por lo que la dependencia con otros sistemas se reduce.

4.5.1 Comparativo de la arquitectura actual de Acambio y SOBA.

Actualmente en Acambio no se cuenta con una arquitectura empresarial que permita una adecuada administración de los sistemas de información y ERP, existen aplicaciones legadas y aisladas en donde ninguna se comunica entre si, esto tiene como resultado trabajos manuales y procesos duplicados muchos de ellos no tienen un fin.

Por lo que es necesario que sus procesos se sometan a un análisis y optimización, ya que el mercado al que ellos están dirigidos, los cambios son drásticos, y requerirán una arquitectura que les permita tener cambios rápidos y fáciles de mantener.

Pero SOBA no solo es para aquellas empresas que no cuentan con una arquitectura alienada a los objetivos del negocio, también aplica para aquellas que requieran realizar una reingeniería y empata con los conceptos de reusabilidad, flexibilidad e integración de sistemas.

Adicional la automatización de los procesos en conjunto con su flujo de trabajo mueven la agilidad del negocio por ende la arquitectura no solo es para tener una arquitectura sofisticada basada en las últimas tendencias de tecnología, en realidad la tecnología esta basada en los estándares pero hay que darle una alineación y esta es por medio de los objetivos del negocio.

4.5.2 Implicaciones con el Modelo SOBA.

Esta investigación tuvo que delimitarse a obtener un modelo de arquitectura de negocios útil que emplea parte de la tecnología que requiere su construcción, con la finalidad de demostrar que los conceptos y metodologías planteados en esta investigación proporcionan un marco de trabajo que puede ser explotado para eficientar el modelado de nuevas arquitectura con la alienación de la tecnología correspondiente.

Durante el desarrollo surgieron gran cantidad de temas y revisando la literatura existe poco material escrito y enfocado a tecnologías de información y en muchos de los casos la tecnología no solo es la respuesta a una necesidad del negocio.

Otra implicación fue el evaluar la herramienta de software libre existente en el mercado que soporte SOA, BPEL, EJB, en el mercado existen herramientas con mejor robustez, sin embargo su uso es por medio de licenciamiento, así que la mejor opción en este caso es NetBeans, se detecto que la herramienta tiene algunas limitaciones con el manejo de datos complejos.

En el mercado existen herramientas tanto de Oracle, Microsoft e IBM y con ellas poder ampliar la funcionalidad de la arquitectura propuesta.

Cabe mencionar que estos conceptos son nuevos y que prometen mejorar siempre y cuando sean bien implementados, algunas empresas están iniciando, sin embargo es un gran paso en el paradigma de implementación de arquitecturas y no solo tecnológicas, si no para negocios, debido a que es un cambio cultural en áreas de tecnologías de información en donde hay que caminar en la misma línea que el negocio, sin que exista un abismo entre ambas áreas.

Conclusión.

Con el desarrollo de una Arquitectura de Negocios Orientada a Servicios (SOBA) se obtienen procesos de negocio inteligentes, pero para llegar a estos se requiere de un análisis exhaustivo de información que se intercambia entre los expertos de negocio y los expertos de TI. La inversión de tiempo, provee beneficios significativos como:

- Acceso a la información, sobresaltando la relevancia, así como el uso de sistemas de seguridad para proporcionar perfiles de uso de la misma.
- Utilización de las mejores prácticas para la calidad, estandarización y publicación de los servicios.
- Los procesos de negocios analizados y optimizados, con oportunidades de mejora, que propicien cambios de forma natural.

En el transcurso de esta investigación, se ha propuesto la Arquitectura de Negocios Orientada a Servicios SOBA-UIA, basados en servicios y tecnologías web. El propósito fundamental de la arquitectura es alargar la notación SOA, adaptando algunos otros conceptos que proporcionen modelos, estructuras de mensaje y un coordinador de servicios de los procesos del negocio.

La arquitectura redefine además de complementar los arquetipos empresariales, tomando en cuenta que los negocios requieren más de un modelo empresarial para alcanzar sus objetivos. Si bien, las arquitecturas tratan de adaptarse a los negocios, con el fin de obtener progresos y mejoras en los procesos, no son absolutistas y pueden ser fácilmente modificados y adaptados. Esta visión que se ha impuesto en la creación de los modelos desde hace algunos años, ha permitido la evolución tanto de todos los procesos que se encuentran involucrados en el negocio (arquitecturas, procesos del negocio, lenguajes de programación, etc.).

Las capas intermedias en el modelo SOA que fueron adaptadas a las capas intermedias del modelo BPM, han hecho una barrera sólida para el negocio. Esta

barrera, sin embargo, es fácilmente adaptable con otros modelos, haciéndola robusta y soportable a cualquier cambio. Los cambios tecnológicos, los más visibles, demandan cambios rápidos y efectivos en el corto plazo. De la misma manera el comportamiento del mercado y el movimiento de las transacciones del negocio. A pesar de que no podemos predecir el futuro, podemos empezar a conceptualizar y adelantarnos a ubicar posibles acciones no contempladas. Estas acciones no deberán imperativamente modificar el núcleo estructural de nuestro planteamiento, sino por el contrario, deberán adaptarse al mismo.

Las notaciones WSDL, así como los registros de los servicios Web en combinación con los servicios empleados por el BPM, fundamentados en las metodologías reconocidas para el desarrollo de la arquitectura distribuida aunado a la integración de un marco de trabajo permiten la integración, flexibilidad y reutilización de todo el ciclo. La operatividad entre las plataformas, donde se desarrollo, se encuentra implícita, al igual que la automatización e invocación de los procesos del negocio.

La hipótesis se concreta verdadera, añadiendo, la tecnología por sí misma, no es la respuesta a las necesidades del negocio, por el contrario, se debe facilitar la comprensión para alinearlos a sus objetivos. Del mismo modo, como se planteó durante todo el desarrollo del presente trabajo, los modelos y conceptualizaciones arquitectónicas de los negocios, deberán adaptarse al negocio, así como a otras arquitecturas, para proporcionar una evolución en todos los procesos.

La conceptualización, documentación de análisis y diseño de SOBA-UIA, se realizó mediante la presente investigación, al mismo tiempo, para la organización que nos permitió extraer uno de sus principales procesos, creando y manipulando los escenarios para una nueva propuesta para arquitectura que permitan al negocio ser flexibles.

La estructura e integración del framework de Zachman y RM-ODP permiten extender el modelo que al inicio, desarrollaban únicamente un paradigma Orientado a Objetos y evolucionar hacia una arquitectura basada en procesos de

negocio, envuelta en servicios web, aplicando técnicas UML para conceptualizar al negocio, utilizando BPM y BPEL como herramienta fundamental para la construcción de SOBA-UIA.

Referencias

- Akira T, Yoshihide N, (2005). Applying ODP Enterprise Viewpoint Language to Hospital Information' Systems.
- Arsanjani, A. et al (2008), SOMA: A method for developing service-oriented solutions, IBM System journal. 21.
- BEA Systems. (2005). Domain Model for SOA Realizing the Business Benefit of Service Oriented Architecture
- Cranor, C. (2003) Design and Implementation of a Distributed Content Management System, , University of Kentucky University, College CorkFlorham.
- Cuenca, L., Ortiz, A., & García, A (2005). Arquitectura Empresarial. Visión general. Universidad pontificia de Valencia, España.
- Delgado, T. (2007) A multiple Perspective of an SDI/Múltiple Perspectiva de una IDE.
- Duffy, D (2004). Domain Architectures: Models and Architectures for UML Applications. NJ, USA. 314.
- Fernández García, N. (2007). Next Generation Web Technologies in Content Management, Universidad de Madrid, España.
- Fethi, A.(2006) A service-oriented architecture for financial business processes A case study in trading strategy simulation. 18.
- Finkelstein, C (2006). Enterprise Architecture for Integration: Rapid Delivery Methods and Technologies Artech House, Boston. 485.
- Flores, B. (2001) Diseño y Desarrollo de una herramienta de soporte para el estudio de procesos organizacionales, CICESE.
- Francis Brudenell, J. (2007). Enterprise Modelling: The Key to Successful Business Systems Integration on, Victoria University, Melbourne Australia.
- Hernández C, Fernández C & Baptista P. (2006). Metodología de la Investigación, México: Mc Graw Hill.
- ISO/IEC 10746-2, ITU-T. (1996). Information Technology – Open Distributed Processing – Reference Model: Foundations. USA.
- Kamoun F, (2006). A Roadmap towards the Convergence of Business Process Management and Service Oriented Architecture, University of Dubai, Dubai.
- Kazi F, (1995). ODP-Based Programming Environment in the Framework of ODP Viewpoint Models, Department of Computer Sciencies University Ottawa. Japón.
- Lavagno, L., Grant M., & Selic B (2004). UML for Real: Design of embedded real time system. (1a. ed.) California USA, 127, 1-17.
- Madsen M, (2006). Unlocking the Power of SOA with Business Process Modeling.
- Malinverno P, (2007). SOA and BPM Are Better Together, Gartner Research Publication, ID Number: G00145586.

- Matjaz B. (2006). Business process execution Language for Web Services. (2a. ed.). Inglaterra, 345, 12-26.
- Microsoft Official Course. (2003). 2524 C Developing XML Web Services using Microsoft ASP.NET (1a. Ed.). USA.
- Nordheim, Stig (2006). Customization of Enterprise Content Management Systems, Agder University College, Kristiansand, Norway.
- Pelechano, V. (2005). Servicios Web, Estándares, extensiones y perspectivas de Futuro. Universidad Politécnica de Valencia España.
- Poirier, C (2004). Networked Supply Chain: Applying Breakthrough Business Process Management Technology to meet relentless Customer Demands. J. Ross Publishing. USA, 131.
- Ravesteyn, P. (2006). A Study into the Critical Success Factors when Implementing Business Process Management Systems. Utrecht, The Netherlands,: University for Applied Science Utrecht, Institute for Process Innovation, Nijenoord .
- Salaman, A., Stevens, J (2006), Semantic Web Technologies and E-Business : Toward the Integrated Virtual Organization and Business Process Automation. IGI Global. The University of Carolina, USA. 101, 1-77.
- Salter, D. Jennings, F. (2009). Building SOA-Based Composite Applications Using NetBeans IDE 6. Birmingham UK, 301.
- Sánchez M, Feroso A, Joyanes L, (2005). Una recomendación basada en MDA, BPM y SOA para el desarrollo de software a partir de procesos del negocio en un contexto de Negocio Bajo Demanda, Universidad Pontificia de Salamanca, España.
- Sarkar S, Thonse S, (2004). An Integrated Modeling Approach to Enterprise Systems Architecture, USA.
- Selda, G. (2006). Architectural Approaches, Concepts and Methodologies of Service Oriented Architecture Master Thesis, Technical University Hamburg Harburg, Alemania.
- Schreiter T, Laures Guido (2006). A Business Process-centered Approach for Modeling Enterprise Architectures, University of Potsdam, Alemania.
- The Gilbane Report. (2 de 03 de 2003). Gilbane. Recuperado el 15 de 09 de 2008, de Gilbane: www.gilbane.com
- Young, S. (2007). Zachman Enterprise Architecture. Tampa USA.
- Yushun, F. Huang S (2008) A service-oriented business performance evaluation model and the performance-aware service selection method. Department of Automation, Tsinghua University, Beijing. 17.
- Zachman, J. (2006). The Zachman Framework for Enterprise Architecture Primer for Enterprise Engineering and Manufacturing. [en línea]. <http://www.zachmanframeworkassociates.com>.

Anexo 1. Guía para el protocolo de la Entrevista

1. La entrevista, es un instrumento de uso cotidiano para obtener información e ideas de un cierto grupo de expertos, indispensable para fortalecer y profundizar sobre algún tema de interés.
2. Requiere de una actitud profesional
3. El objetivo de la guía es establecer un protocolo que permita de forma ordenada obtener información sobre los procesos de Acambio.
4. Tiempo de duración de la sesión 2 horas y observación de cómo llevan acabo los procesos.

1. ¿Cuál es el giro del negocio?

El mercado digital

2. ¿De dónde surge la idea?

De la crisis económica que vivió Argentina en donde llevaban acabo intercambio de productos y servicios.

3. ¿Cuál es la misión de la empresa?

Crear, desarrollar y transmitir de forma transparente y eficiente, la industria de intercambios multilaterales de bienes y servicios entre empresas.

4. ¿Cuál es la visión?

Consolidar el liderazgo en el mercado nacional de intercambios, logrando el reconocimiento internacional por su modelo de operación.

5. ¿Cómo funciona el negocio?

Funciona como un mercado alternativo para clientes que tienen exceso de inventarios, falta de liquidez. Se lleva una cartera de clientes corporativos cobrando una comisión por el uso del mercado.

6.- *¿Qué tipo de clientes se tienen actualmente?*

Corporativos principalmente, sien embargo, se quiere extender al mercado (Retail) global dirigido a personas físicas.

7.- *¿Existen empresas dedicadas al mismo giro en México?*

Rush Business y Barter, pero esta última se encuentra en Nueva Zelanda.

8.- ¿Actualmente con que herramientas soportan sus operaciones y para el nuevo modelo como soportarán esas operaciones?

Se tienen sistemas legados en donde el ejecutivo de cuenta de cada empresa tiene acceso para actualizar los productos y servicios en el catálogo correspondiente, así como el mantenimientos al sistema de clientes, sin embargo se tienen algunas deficiencias debido a la falta de actualización de información.

Para el nuevo modelo se pretende que se tenga la nueva infraestructura integrada, de tal forma que los clientes no requieran de un ejecutivo.

9. ¿Cuáles son los procesos principales del negocio de Acambio?

La captación de clientes y el intercambio de productos o servicios.

10. ¿De los proceso cuál es el que consideran critico y cuál es la razón?

La transacción de intercambio, por temas regulatorios y es la base del negocio (ganancias).

11. ¿La empresa cuenta con una arquitectura empresarial (AE) si la respuesta es Si como esta constituida de lo contrario explique por que no?

No, debido a la falta de conocimiento.

12. ¿Los procesos de negocio se encuentran documentados en manuales operativos?

No, por falta de personal, sin embargo, el único proceso documentado es el de intercambio.

13. ¿Se cuenta con un sistema de gestión de calidad?

Si, se maneja a través de ISO 9000.

14. ¿Existe dentro de la estructura organizacional, alguna entidad de TI?

Si, actualmente solo se dedica al soporte.

15. ¿Con cuántos sistemas cuenta la organización para soportar sus procesos de negocio?

Con dos 2, SAP Business One y CRM Xo.






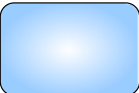


16. ¿Ambos sistemas conviven entre si?

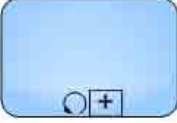







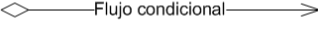
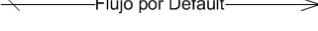
No, para mantener al ERP es necesario capturar la información por lo que existen procesos manuales de captura tanto en el CRM como en el ERP.

17. ¿Como llevarán acabo la integración de los sistemas al momento de liberación el modelo de negocio a nivel global?

Con tecnología Orientada a Servicios e integración de Base de datos.

Anexo 2 Notación Completa de BPM

OBJETOS DE FLUJO		
Nombre	Símbolo	Función
Evento	 Evento de Inicio  Evento Intermedio  Eventos de Fin Cada uno puede combinar tiempo o bien mensajes.  Evento Inicia Mensaje  Evento Inicia tiempo	Representar algo que sucede durante el curso de un proceso de negocio, afectan el flujo del proceso y usualmente tienen una causa y un resultado.
Actividades	 Tarea  Subproceso  Compensación Es usado para eliminar los efectos de una actividad previa (ejemplo cancelaciones).	Estas Representan el trabajo que es ejecutado dentro de un proceso de negocio. Las actividades pueden ser compuestas o no, por lo que dentro de los ejemplos utilizamos los dos tipos de actividades: Tarea y subproceso

	 Loop (Ciclo)  Subproceso Ad hoc <p>Nota las actividades pueden combinar loop, subproceso y subproceso Ad hoc.</p>	
<p>Compuertas (Gateways)</p>	 Decision  And  XOR  OR  Base del evento.	<p>Son elementos del modelado que se utilizan para controlar la divergencia y la convergencia del flujo.</p>
Objetos de Conexión		
<p>Conectores</p>	 Flujo de Secuencia  Flujo condicional  Flujo por Default	<p>Son los elementos usados para conectar dos objetos del flujo dentro de un proceso. Dentro de los ejemplos utilizamos la Líneas de secuencia, que conectan los objetos de flujo, y las asociaciones, que son las líneas punteadas que nos permitieron asociar</p>


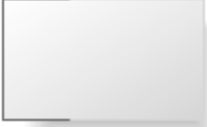

		anotaciones dentro de algunos flujos
Canales		Son elementos utilizados para organizar las actividades del flujo en diferentes categorías visuales que representan áreas funcionales, roles o responsabilidades.
Artefactos	 <p>Notación</p>  <p>Objetos</p>	Los artefactos son usados para proveer información adicional sobre el proceso. Dentro de los ejemplos se utilizaron algunas anotaciones dentro del flujo

Tabla 3. Notación BPM

Glosario.

CONCEPTO	DEFINICIÓN
Arquitectura empresarial	Un plano que explica como la tecnología de Información y los elementos de la Administración trabajan conjuntamente como una unidad.
SOA	Son las siglas de Arquitectura Orientada a Negocios.
Servicios web	Es un URL direccionable, conjunto de funcionalidades que están expuestas en la red para ser accedidas por aplicaciones distribuidas.
Procesos de negocio	Lo define como una serie de actividades coherentes que son realizadas por un grupo de colaboradores para alcanzar un objetivo.
BPM	Administración de procesos de negocio
WSDL	Es el lenguaje de descripción de servicios web.
BPEL	Lenguaje de Ejecución de Procesos de Negocio con Servicios Web).
XML	Lenguaje de Marcado extensible
HTTP	El protocolo de transferencia de hipertexto (HTTP, HyperText Transfer Protocol) es el protocolo usado en cada transacción de la Web (WWW). HTTP fue desarrollado por el consorcio W3C
SOAP	Por sus siglas de Simple Object Access Protocol, es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML
RM-ODP	El Modelo de Referencia de ISO para el

	Procesamiento Abierto y Distribuido.
EAI	Integración de aplicaciones empresariales.
Workflow	Es el flujos de trabajo de cómo fluye la información que soporta las tareas.

Tabla 4. Glosario