



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES

ARAGÓN

***“SOA E INTEGRACIÓN A TRAVÉS DE PROGRESS
SONIC ESB”***

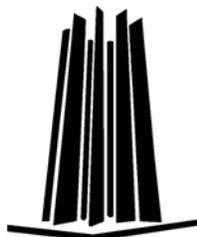
TRABAJO ESCRITO EN LA MODALIDAD
DE ALTO NIVEL ACADÉMICO QUE PARA
OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN

P R E S E N T A:

LAUREANO LEYVA BARRERA

ASESOR DE TESIS:

M. EN C. MARCELO PÉREZ MEDEL



MÉXICO, 2009.



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

ÍNDICE

ÍNDICE	2
INTRODUCCIÓN	3
CAPÍTULO 1. ¿QUÉ ES SOA?	5
1.1 Nivel 1. Servicios Básicos.	6
1.2 Nivel 2. Servicios Estructurados (Architected Services por sus siglas en inglés)	9
1.3 Nivel 3. Procesos de Negocios y Colaborativos	11
1.4 Nivel 4. Medición de Procesos de Negocios	15
1.5 Nivel 5. Optimización de Procesos de Negocio.....	16
1.6 En conclusión.....	17
CAPÍTULO 2. ¿QUÉ ES SONIC ESB?	18
2.1 Plataformas soportadas.	26
2.2 Sonic ESB V7.6	26
CAPÍTULO 3. SOA GOVERNANCE	29
3.1 Lo más importante de SOA Governance: Asegurar la confiabilidad de las operaciones en la SOA.....	29
3.2 Logrando los objetivos de negocio con SOA Governance.	30
3.3 Reduciendo el riesgo en la seguridad y el cumplimiento de las reglas a través de la implementación de SOA Governance.....	30
3.4 Descubriendo y controlando los servicios que generan problemas.	31
CAPÍTULO 4. ESCENARIOS DE INTEGRACIÓN	33
4.1 Caso 1. Proceso Continuo.....	34
4.2 Caso 2. Proceso Continuo.....	37
CAPÍTULO 5. CASOS REALES.....	40
CONCLUSIONES	44
GLOSARIO	45
BIBLIOGRAFIA	50

INTRODUCCIÓN

Este trabajo mostrará una serie de temas relacionados con la tendencia actual de sistemas, la cual pretende como principal objetivo la conjunción, comunicación, interrelación e intercambio de información a través de las plataformas tecnológicas actuales, sin tomar en cuenta como una limitante los sistemas operativos, tecnología de procesadores, lenguajes de programación e infraestructura sobre la cual operan los sistemas empresariales.

A través de la recapitulación de este trabajo me he dado a la tarea de visitar diferentes empresas las cuales reflejan la problemática actual, visualizada desde el punto de vista de negocios se traduce hacia ganancias o pérdidas, las cuales podrán verse impactadas por medio de la Implementación de arquitecturas de sistemas robustas, confiables, escalables y orientadas a servicios.

Este trabajo recolecta una serie de conceptos que hoy día se vienen manejando en la industria, tales como SOA (Service Oriented Architecture) por sus siglas en inglés, ESB (Enterprise Service Bus), WSDL (Web Service Definition Language) y algunos otros que se irán detallando conforme se vayan tocando cada uno de los conceptos.

En la parte inicial del trabajo nos enfocamos a la definición conceptual de SOA, en este apartado podremos encontrar definiciones, tendencias, modelos y recomendaciones en la adopción de una arquitectura orientada a servicios o SOA.

El siguiente apartado abordará el tema de Progress SONIC ESB, producto perteneciente a la empresa Progress Software Corp. Quien ha innovado en el concepto de integración de sistemas a través de esta suite.

Posteriormente se tocará y profundizará sobre el tema de SOA Governance, en el cual se abordarán además de los temas de definición y aplicación, las diferentes ofertas que hoy día tienen los principales competidores como Progress Software Corp, Oracle e IBM.

Un apartado muy importante y dejará a lector con un mejor sabor de boca una vez que haya leído o recolectado la cadena de información proporcionada al inicio de este trabajo, será una lista de escenarios de aplicación para integración, ya hoy día sabemos que existen diferentes alternativas para la implementación de soluciones que permitan entrelazar y habilitar canales para el intercambio de información, también habrá que considerar que hay estándares y escenarios que son comunes a la mayoría de las empresas que requieren no solo comunicarse internamente, sino también enviar y recibir información de diferentes fuente externas a los sistemas propietarios. Este apartado proporcionará una serie de escenarios de integración los cuales proporcionarán una visión más apegada a la realidad sobre los casos de uso de las tecnologías antes mencionadas.

En el apartado final, se podrán encontrar algunos casos reales, que aún cuando no será posible mencionar los nombres de las empresas que han adoptado estas buenas prácticas

y que dicho por ellos mismos se han optimizado recursos humanos y materiales una vez que se han implementado soluciones enfocadas hacia la integración y particularmente cuando se ha habilitado su red de sistemas hacia SOA.

CAPÍTULO 1. ¿QUÉ ES SOA?

La Arquitectura Orientada a Servicios (en inglés Service-Oriented Architecture o SOA), es un concepto de arquitectura de software que define la utilización de servicios para dar soporte a los requerimientos de software del usuario.

SOA es una arquitectura de software que permite la creación y/o cambios de los procesos de negocio desde la perspectiva de TI de forma ágil, a través de la composición de nuevos procesos utilizando las funcionalidades de negocio que están contenidas en la infraestructura de aplicaciones actuales o futuras (expuestas bajo la forma de webservices).

SOA define las siguientes capas de software:

- ✚ aplicativa básica, sistemas desarrollados bajo cualquier arquitectura o tecnología, geográficamente dispersos y bajo cualquier figura de propiedad;
- ✚ de exposición de funcionalidades, donde las funcionalidades de la capa aplicativas son expuestas en forma de servicios (webservices);
- ✚ de integración de servicios, facilitan el intercambio de datos entre elementos de la capa aplicativa orientada a procesos empresariales internos o en colaboración;
- ✚ de composición de procesos, que define el proceso en términos del negocio y sus necesidades, y que varía en función del negocio; de entrega, donde los servicios son desplegados a los usuarios finales.

Dicho de otra forma, se puede conceptualizar el modelo SOA a través de la representación del modelo de madurez de SOA (SOA Maturity Model en inglés) el cual se describe a continuación:

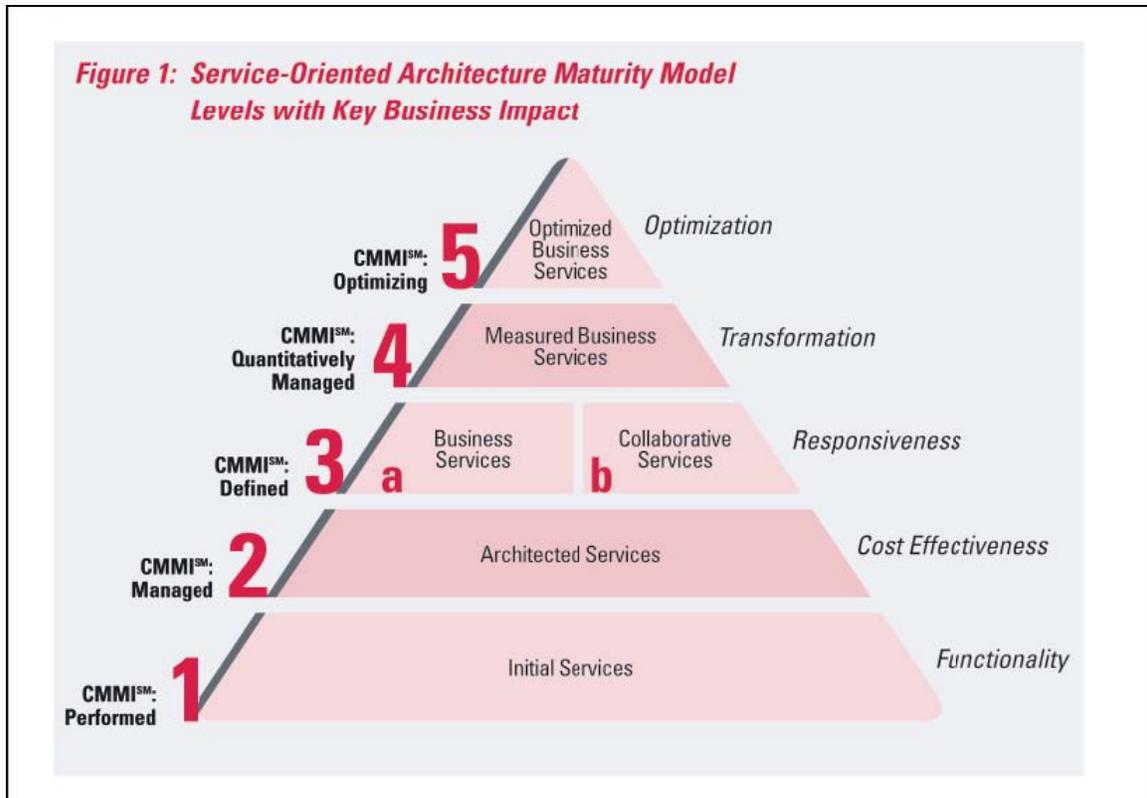


Figura 1. Modelo de madurez de SOA.

1.1 Nivel 1. Servicios Básicos.

El nivel 1 del modelo de madurez de SOA es llamado Servicios Básicos (Initial Services), este nivel representa el conocimiento inicial y la primera fase en la adopción del concepto SOA. Los proyectos en esta fase están típicamente realizados en base a necesidades específicas y funcionalidad determinada mientras se obtiene mayor profundidad en los conceptos de SOA.

Usualmente la introducción inicial sobre SOA es manejada a través del desarrollo de aplicaciones como parte de un proyecto de integración. Se adquieren nuevas habilidades y se realizan cuantificaciones iniciales sobre el retorno de inversión de las aplicaciones o arquitectura que se está diseñando.

Este es el nivel en el cual se emplean los estándares básicos de SOA indicados por W3C los cuales son introducidos como XML para la definición de formatos de mensaje, el WSDL (WebService Definition Lenguaje) para la definición de la interfase y SOAP (Simple Object Access Protocol) para la invocación de servicios.

Un ejemplo claro de este nivel es mostrado en la Figura 2, en el cual se realiza la liga directamente entre un portal Web y un par de aplicaciones, una aplicación de pronóstico de ventas la cual se encuentran operando en un sistema mainframe y también hacia una aplicación CRM la cual es expuesta o publicada como un Web Service.

En este pequeño ejemplo se utilizan solamente 2 componentes de la infraestructura SOA:

- ✚ Servicio de integración con el Mainframe, el cual se encargará mediante estándares, de la comunicación entre el portal y los servicios que operan en el mainframe.
- ✚ Plataforma de desarrollo altamente productiva, la cual deberá ser la óptima para la creación o desarrollo de aplicaciones de negocio orientadas a servicios, tales como Progress OpenEdge para el desarrollo y la exposición de aplicaciones o lógica de negocios accesible a través de Web Services.

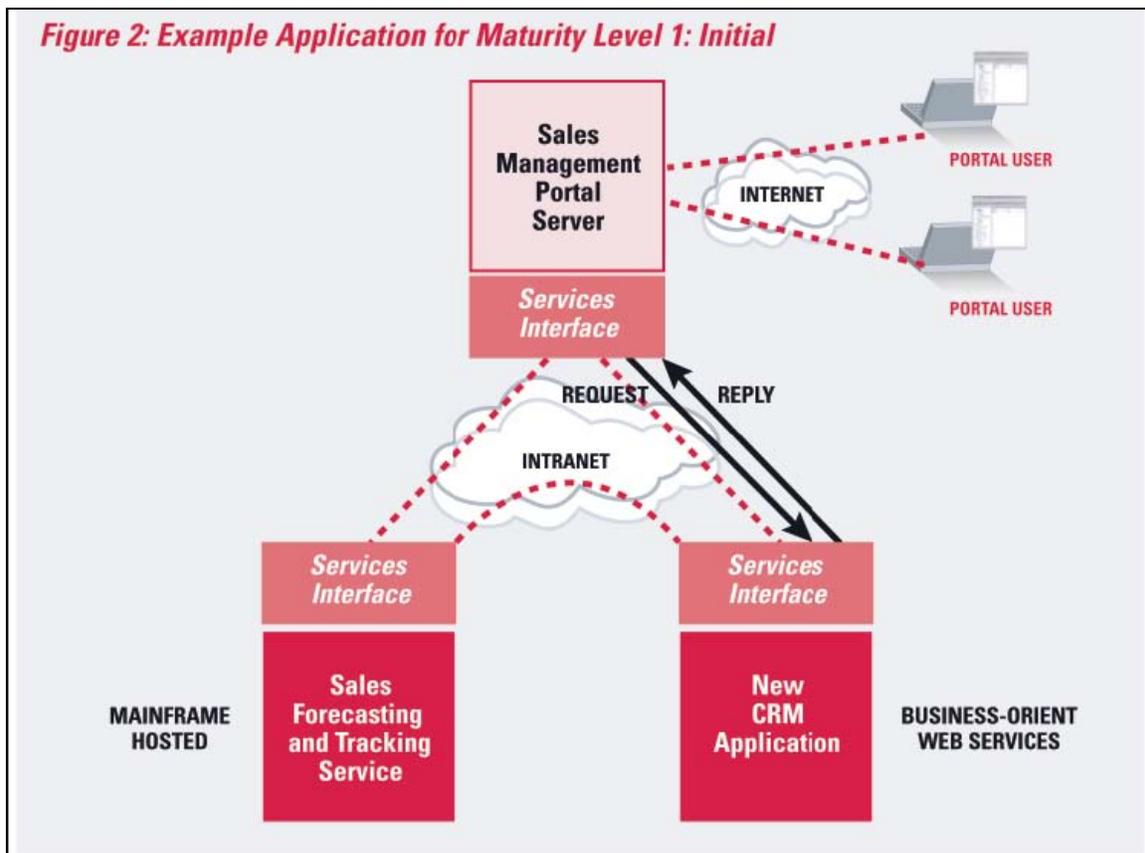


Figura 2. Ejemplo de aplicación para SOA nivel de madurez 1.

Los principales beneficios en este tipo de arquitecturas o modelos de integración son el poder proporcionar la funcionalidad de negocios requerida mientras tanto se adquiere conocimiento sobre como desarrollar e implementar aplicaciones SOA básicas, tales como la implementación de SOAP en el protocolo de intercambio de información entre el portal y las aplicaciones de ventas.

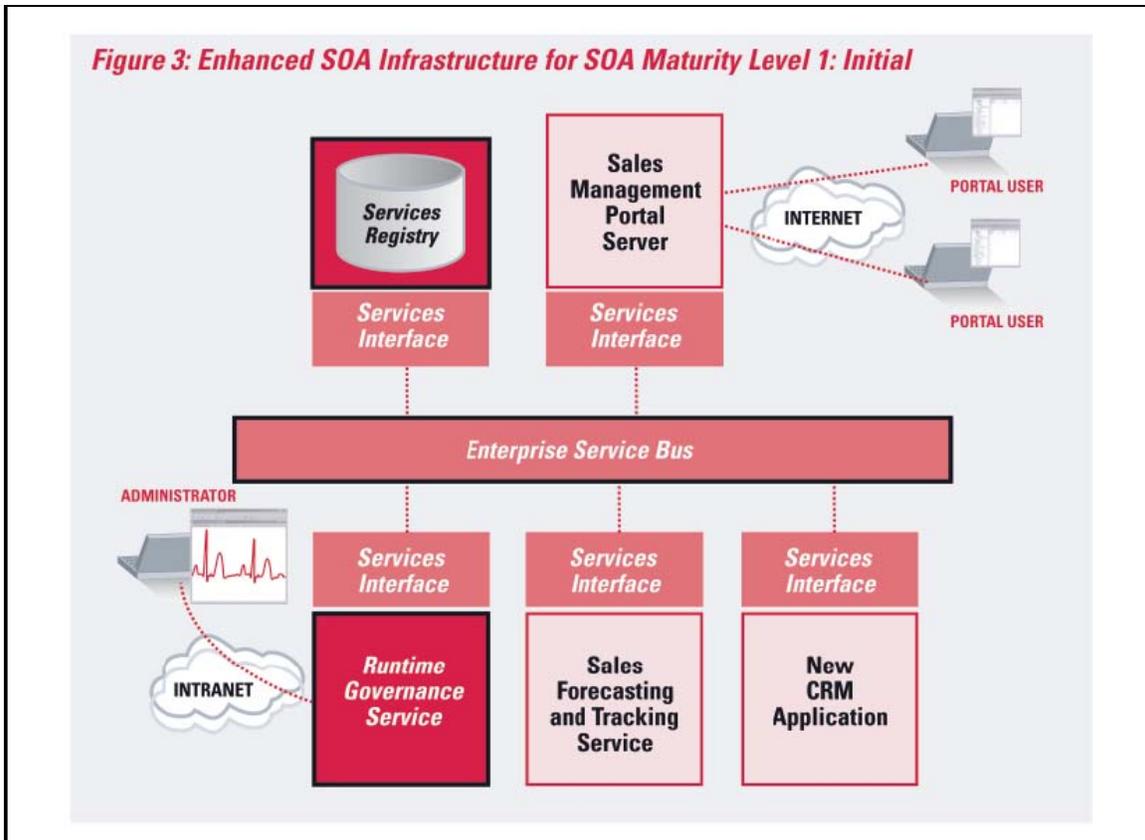


Figura 3. Infraestructura SOA extendida.

1.2 Nivel 2. Servicios Estructurados (Architected Services por sus siglas en inglés)

El nivel 2 del modelo de madurez SOA es un nivel en el cual los estándares están definidos y agrupados de acuerdo a las técnicas de implementación de “SOA governance”, típicamente bajo el mando del arquitecto de una organización.

Los principales beneficios de este nivel están basados en el desarrollo e implementación (reducción de) a través del uso de estándares SOA para la infraestructura y en el uso de componentes, comparado con el uso de tecnologías obsoletas o sobre la acumulación de costos a través del uso de proyectos que utilizan arquitecturas poco flexibles o monolíticas. Estos beneficios son de gran valor en ambientes totalmente heterogéneos en gran parte de las empresas de diferentes sectores.

Basados en el conocimiento y retroalimentación que se ha obtenido del primer nivel del modelo de madurez SOA, los estándares estructurados y las tecnologías para la implementación basada en estándares, son definidas, por ejemplo los estándares están definidos a través de los siguientes conceptos:

- ✚ Protocolos empresariales SOA los cuales serán utilizados y seleccionados con base a los estándares de la industria, típicamente en los de W3C, OASIS y WS-I.
- ✚ Plataformas que serán utilizadas para la implementación de servicios.
- ✚ Políticas, incluyendo la reutilización de componentes, administración o control en tiempo de ejecución, el cumplimiento de las reglas y políticas de seguridad.
- ✚ Proceso técnico de revisión para la definición de nuevos servicios y la reutilización de los ya existentes.

La figura 4 muestra un ejemplo de la puesta en operación de SOA referente al nivel 2 del modelo de madurez SOA para un ejemplo de “financiamiento”. El nivel 2 incluye el uso de componentes clave mencionados con anterioridad y mostrados en la figura 3 incluyendo servicios para la administración o control en tiempo real y un Bus Empresarial de Servicios (ESB ó Enterprise Service Bus).

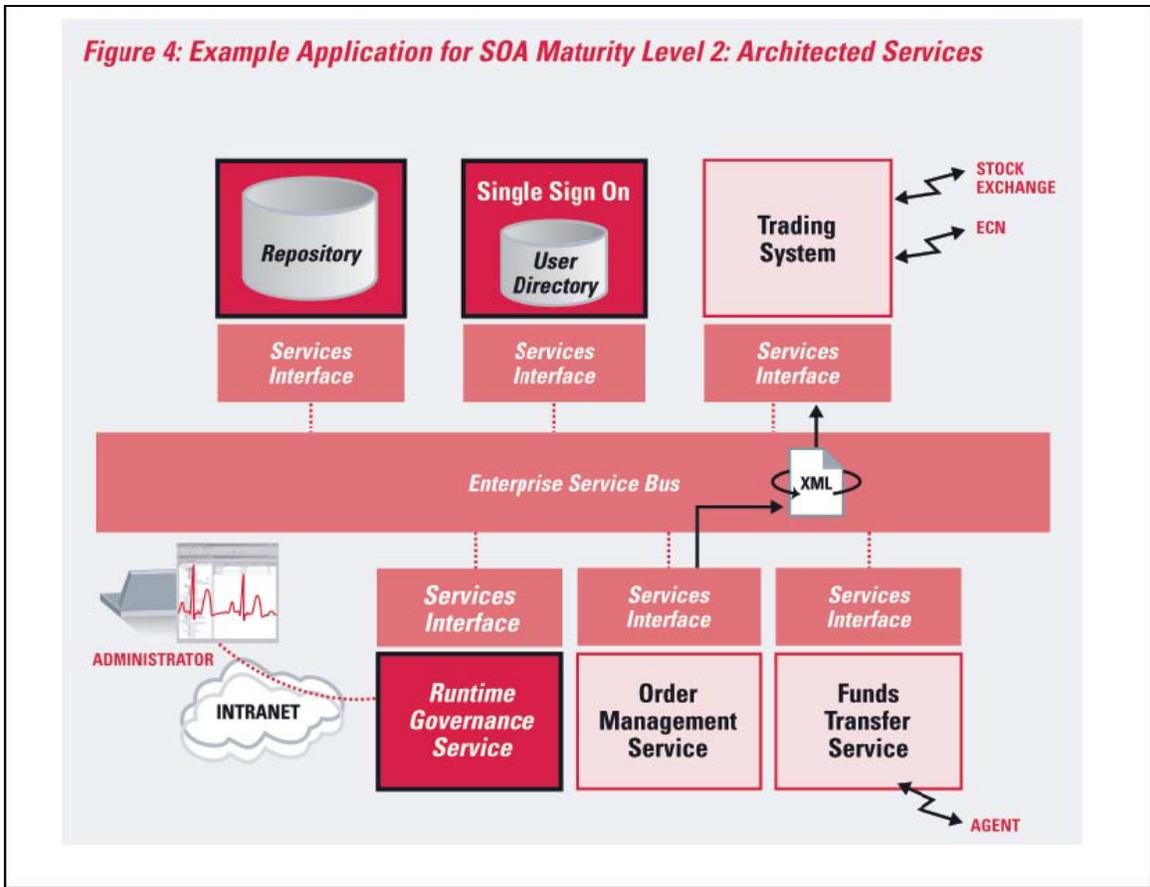


Figura 4. Ejemplo de aplicación para SOA nivel de madurez 2.

1.3 Nivel 3. Procesos de Negocios y Colaborativos

El principal objetivo o foco del nivel 3 del modelo de madurez es la alianza entre la tecnología y las organizaciones empresariales a fin de asegurar que el uso de la SOA proporciona respuestas de negocio claras. El valor principal de SOA es la vinculación entre los procesos de negocios y procesos digitales tal como se muestra en el Figura 5.

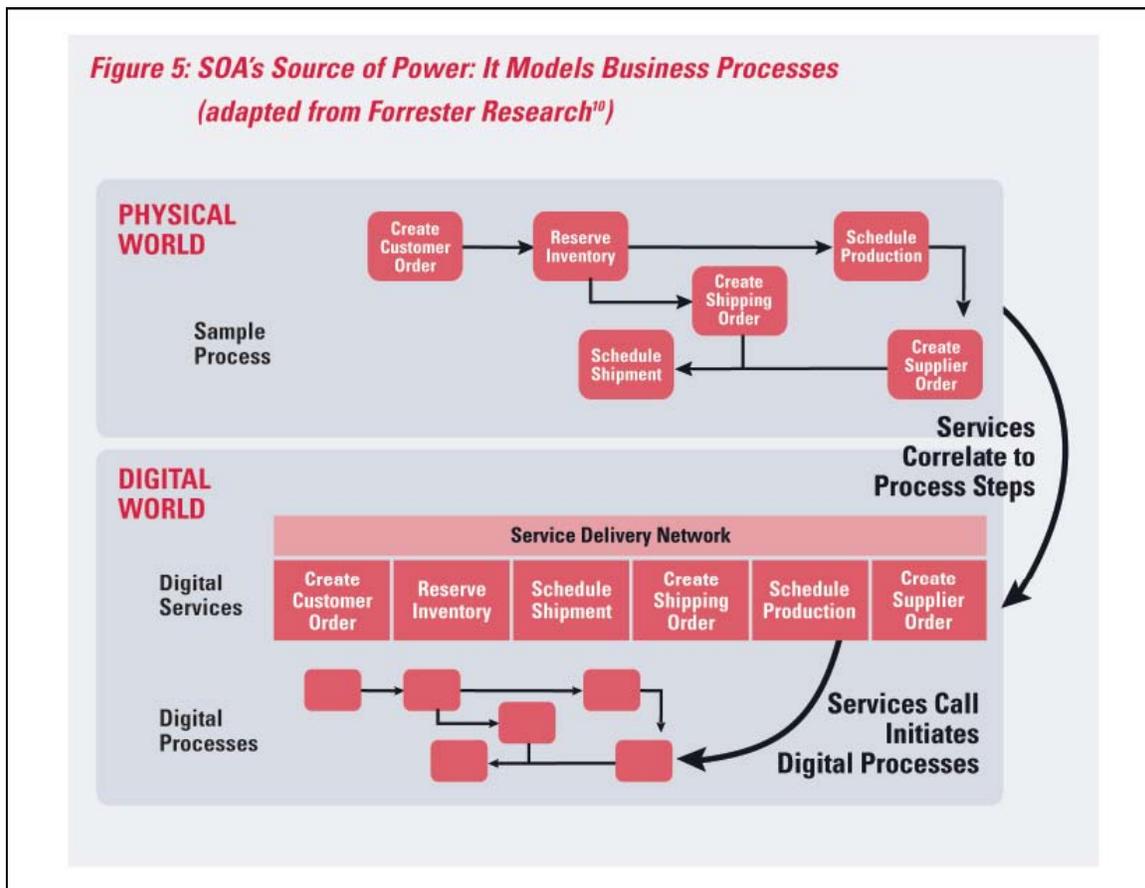


Figura 5. Fuente de energía para SOA.

El nivel 3 del modelo de madurez está definido con dos rutas complementarias para la atención de los objetivos como uno solo, servicios de negocio, focalización en la mejora de procesos de negocio internos y servicios colaborativos, enfocados en la optimización de procesos de colaboración con elementos o proveedores de información externos.

La figura 6 muestra un ejemplo de la implementación del SOA referente al nivel 3 del modelo de madurez para “financial trading” o negociación financiera. Los puntos más relevantes en este tipo de implementación son:

- Business Process Management (BPM) o Administración de Procesos de Negocio. La administración de procesos de larga duración incluyen mensajes secuenciales entre cada servicio. Esta tarea puede ser realizada por ejemplo a través de Progress BPEL Server el cual puede administrar el estado para cada proceso.

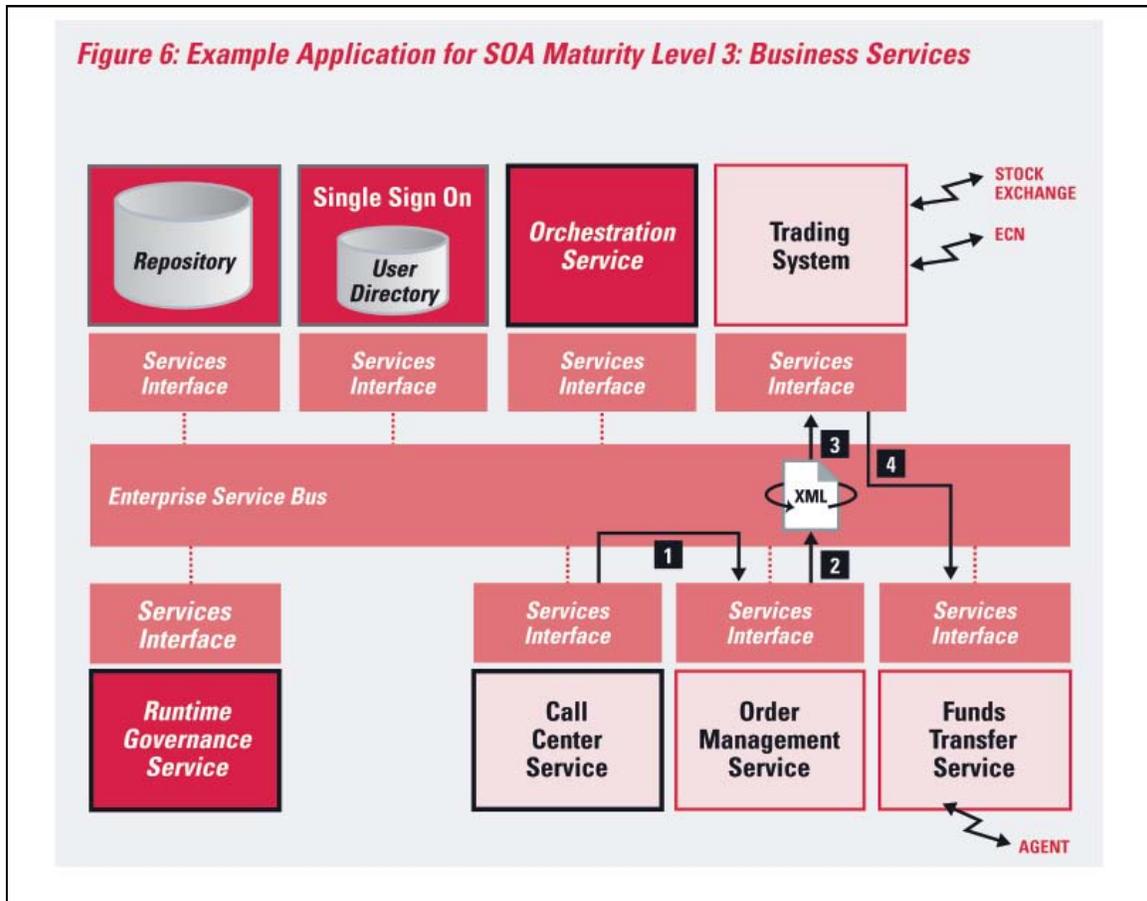


Figura 6. Ejemplo de aplicación para SOA nivel de madurez 3.

La figura anterior muestra claramente la secuencia que debe seguir un proceso de negocio que hace uso de los estándares como:

- ✚ **BPEL (Business Process Execution Language).** *Business Process Execution Language* (BPEL), en castellano Lenguaje de Ejecución de Procesos de Negocio es un lenguaje que se desarrolló a partir de WSDL y XLANG, ambos lenguajes orientados a la descripción de servicios Web. Básicamente consiste en un lenguaje basado en XML diseñado para el control centralizado de la invocación de diferentes servicios Web, con cierta lógica de negocio añadida que ayudan a la programación en gran escala (programming in the large).
- ✚ **BPM (Business Process Management).** Se llama *Business Process Management* a la metodología empresarial cuyo objetivo es mejorar la eficiencia a través de la gestión sistemática de los procesos de negocio (BPR), que se deben modelar, automatizar, integrar, monitorear y optimizar de forma continua.

Como su nombre sugiere, Business Process Management (BPM) se enfoca en la administración de los procesos del negocio. A través del modelado de las actividades y procesos puede lograrse un mejor entendimiento del negocio y muchas veces esto presenta la oportunidad de mejorarlos. La automatización de los procesos reduce errores, asegurando que los mismos se comporten siempre de

la misma manera y dando elementos que permitan visualizar el estado de los mismos. La administración de los procesos permite asegurar que los mismos se ejecuten eficientemente, y la obtención de información que luego puede ser usada para mejorarlos. Es a través de la información que se obtiene de la ejecución diaria de los procesos, que se puede identificar posibles ineficiencias en los mismos, y actuar sobre las mismas para optimizarlos.

Para soportar esta estrategia es necesario contar con un conjunto de herramientas que den el soporte necesario para cumplir con el ciclo de vida de BPM. Este conjunto de herramientas son llamadas Business Process Management System y con ellas se construyen aplicaciones BPM.

- ✚ WSDL (Web Service Definition Language). Son las siglas de Web Services Description Language, un formato XML que se utiliza para describir servicios Web (se lee wisdel). La versión 1.0 fue la primera recomendación por parte del W3C y la versión 1.1 no alcanzó nunca tal estatus. La versión 2.0 se convirtió en la recomendación actual por parte de dicha entidad.

WSDL describe la interfaz pública a los servicios Web. Está basado en XML y describe la forma de comunicación, es decir, los requisitos del protocolo y los formatos de los mensajes necesarios para interactuar con los servicios listados en su catálogo. Las operaciones y mensajes que soporta se describen en abstracto y se ligán después al protocolo concreto de red y al formato del mensaje.

Así, WSDL se usa a menudo en combinación con SOAP y XML Schema. Un programa cliente que se conecta a un servicio web puede leer el WSDL para determinar que funciones están disponibles en el servidor. Los tipos de datos especiales se incluyen en el archivo WSDL en forma de XML Schema. El cliente puede usar SOAP para hacer la llamada a una de las funciones listadas en el WSDL.

El foco alternativo para el nivel 3 del modelo de madurez es el uso de servicios colaborativos (Collaborative Services) con especial atención en la interconexión con proveedores de información externos. La figura 8 muestra un ejemplo donde una compañía comercial extiende sus operaciones hacia nuevos negocios de intercambio financiero (compra venta de acciones y monedas) el cual es ofrecido a través de Internet. Los principales beneficios obtenidos en este tipo de implementación utilizando servicios colaborativos son:

- ✚ Use de protocolos estándares de SOA para el soporte de funcionalidad específica para un B2B (Business to Business) o Negocio a Negocio.
- ✚ Servicios de colaboración para la implementación de protocolos B2B y para el soporte en servicios de transformación necesarios entre mensajes utilizados internamente hacia el resto de la empresa y hacia todos aquellos procesos externos que requieren de esta información.

- ✚ La conexión de (ECN) Electronic Crossing Network o redes de intercambio electrónico se ha movido de un protocolo propietario hacia un protocolo estándar en la industria y de ésta forma es administrado a través de servicios de colaboración.

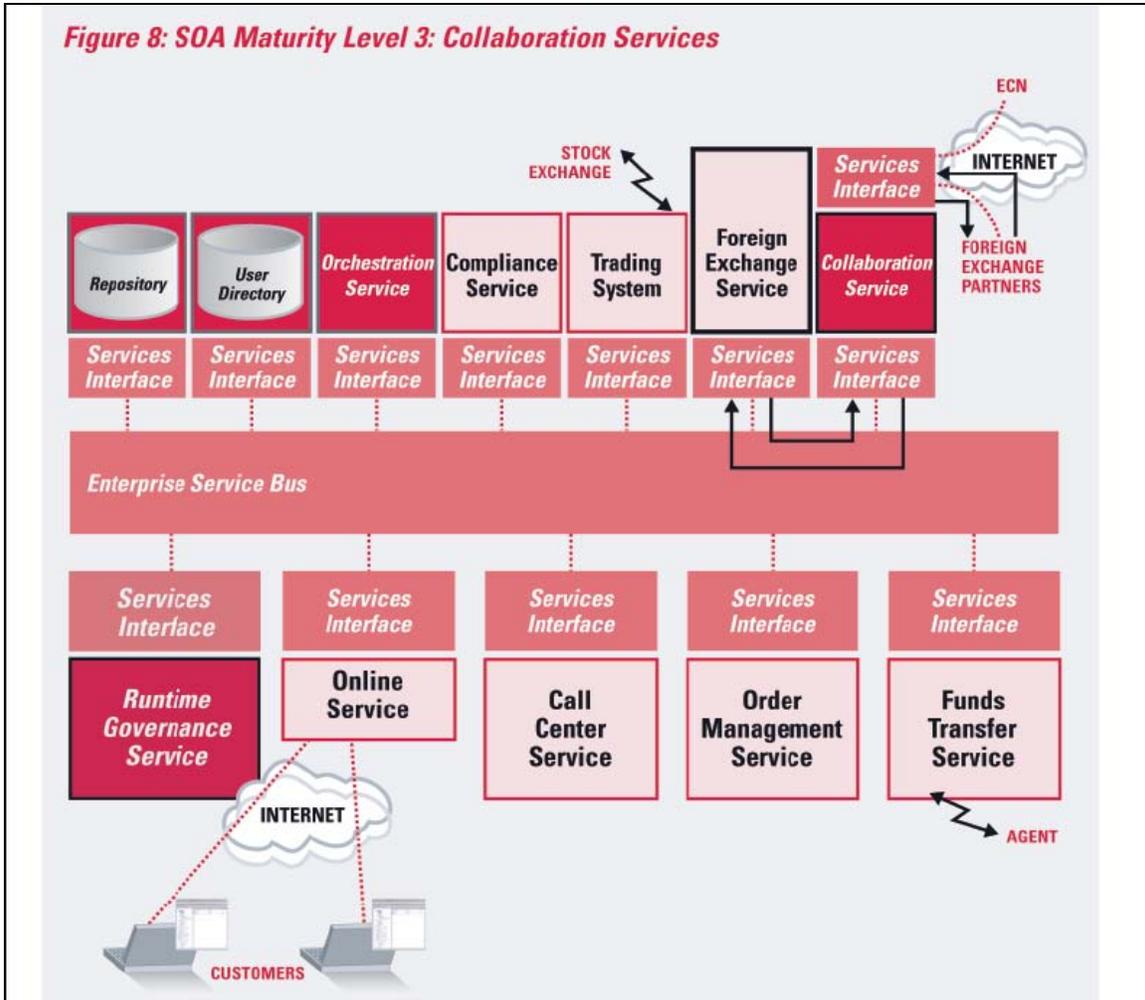


Figura 8. Nivel 3 del modelo de madurez de SOA.

1.4 Nivel 4. Medición de Procesos de Negocios

Mientras el nivel 3 del modelo de madurez se focaliza o concentra en la implementación de procesos de negocio internos y/o externos, el nivel 4 del modelo de madurez SOA se focaliza en la medición y presentación de esos procesos a nivel de negocio y no de tecnología propiamente, de tal forma que continuamente se obtiene retroalimentación sobre el desempeño e impacto en el negocio de los procesos implementados en el nivel 3. Dicho de otra forma, en este nivel se podrán observar los umbrales y elementos de valor obtenidos a partir de la implementación de servicios básicos, web services, servicios colaborativos, etc. Esto permitirá de manera tangible, observar los indicadores que permitirán tomar decisiones sobre los procesos de negocio optimizados, automatizados e integrados a través de éste modelo de madurez.

La figura 9 ilustra un ejemplo del proceso para configuración, ordenes y manufactura, con servicios para cada funcionalidad requerida, distribuido a través de diferentes localidades geográficas.

Los elementos clave en este ejemplo son:

- ✚ Procesamiento de eventos en tiempo real (Real-time Event Stream Processing). En este ejemplo se recolectan todos los eventos “RFID” (Radio Frequency Identification) en eventos de base de datos y filtrados hacia eventos basados en reglas, los cuales son enviados hacia otros puntos para ser utilizados por otros componentes o servicios.
- ✚ Visibilidad en procesos de negocio (BPV) el cual habilita para la administración de la SOA desde una perspectiva de negocio.

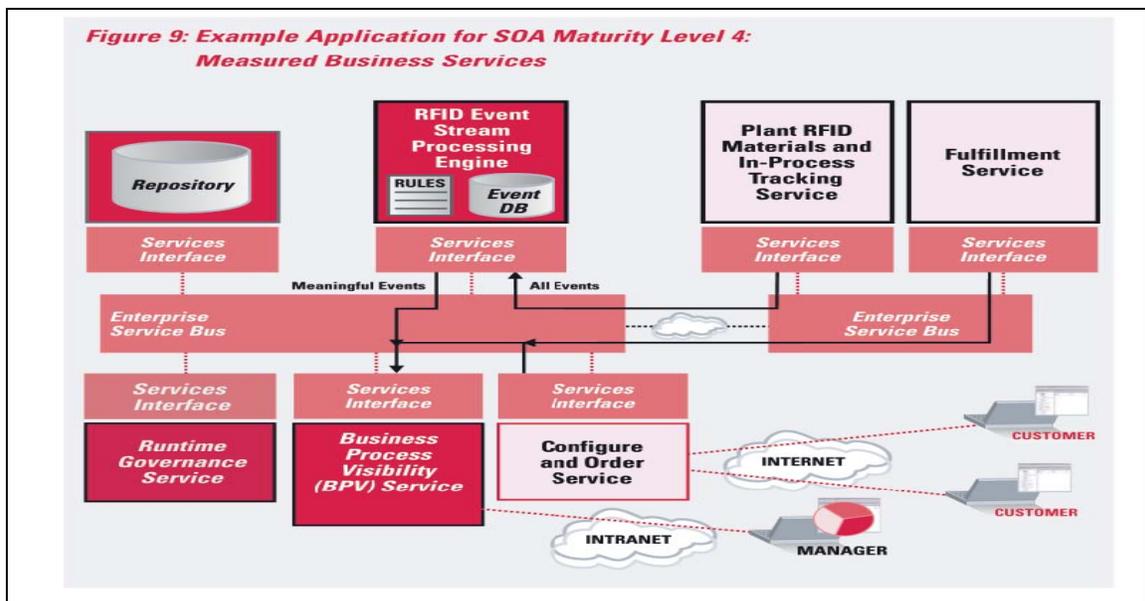


Figura 9. Ejemplo de aplicación para SOA nivel de madurez 4.

1.5 Nivel 5. Optimización de Procesos de Negocio

El nivel 5 del modelo de madurez SOA agrega respuesta automática con base en las mediciones e indicadores generados en el nivel 4, en este sentido los sistemas de información basados en SOA se convierten en “*Sistemas Nerviosos Empresariales*” y actúan de acuerdo a eventos los cuales ocurren a nivel de negocio y con base en ciertas reglas que se han planteado con el fin de lograr los objetivos de la empresa.

La figura 10 muestra un ejemplo del proceso de configuración, ordenes y manufactura mejorado para proporcionar dinámicamente los precios de acuerdo al estado de los materiales y al estado del proceso de manufactura. Por ejemplo, si las partes para una versión particular de un producto se encuentran escasas, el precio puede ser calculado para estimular a los compradores a ordenar el producto utilizando partes adicionales o similares. Este ejemplo está inspirado en el modelo de trabajo de Dell Corporation’s.

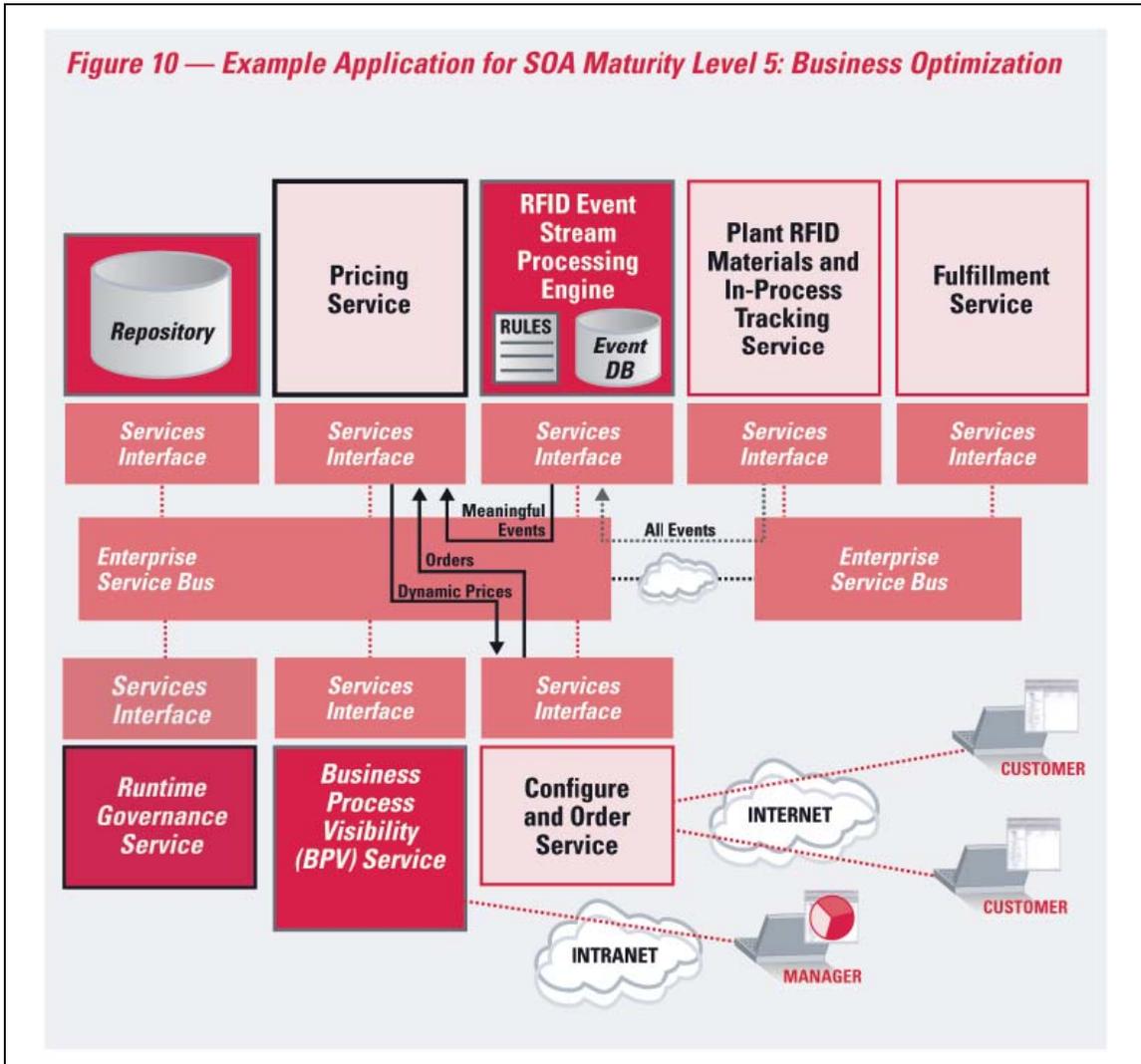


Figura 10. Ejemplo de aplicación para SOA nivel de madurez 5.

1.6 En conclusión.

Los beneficios que puede obtener una compañía que adopte SOA son:

- ✚ Mejora en los tiempos de realización de cambios en procesos.
- ✚ Facilidad para evolucionar a modelos de negocios.
- ✚ Facilidad para abordar modelos de negocios basados en colaboración con otros entes (socios, proveedores).
- ✚ Poder para reemplazar elementos de la capa aplicativa SOA sin interrupción en el proceso de negocio

SOA proporciona una metodología y un marco de trabajo para documentar las capacidades de negocio y puede dar soporte a las actividades de integración y consolidación.

En un ambiente SOA, los nodos de la red hacen disponibles sus recursos a otros participantes en la red como servicios independientes a los que tienen acceso de un modo estandarizado. La mayoría de las definiciones de SOA identifican la utilización de Servicios Web (empleando SOAP y WSDL) en su implementación, no obstante se puede implementar una SOA utilizando cualquier tecnología basada en servicios.

Al contrario de las arquitecturas orientado a objetos, las SOAs están formadas por servicios de aplicación débilmente acoplados y altamente interoperables. Para comunicarse entre sí, estos servicios se basan en una definición formal independiente de la plataforma subyacente y del lenguaje de programación (p.ej., WSDL). La definición de la interfaz encapsula (oculta) las particularidades de una implementación, lo que la hace independiente del fabricante, del lenguaje de programación o de la tecnología de desarrollo (como Plataforma Java o Microsoft.NET). Con esta arquitectura, se pretende que los componentes software desarrollados sean reutilizables, ya que la interfaz se define siguiendo un estándar; así, un servicio C Sharp podría ser usado por una aplicación Java.

Los lenguajes de alto nivel como BPEL o WS-coordinación llevan el concepto de servicio un paso adelante al proporcionar métodos de definición y soporte para flujos de trabajo y procesos de negocio.

CAPÍTULO 2. ¿QUÉ ES SONIC ESB?

La expectativa de IT en las organizaciones normalmente está basada sobre el hecho de poder alcanzar y hacer más con menos recursos, responder rápidamente a los cambios, atender de manera efectiva cualquier requerimiento de negocio y transformarlo hacia un modelo tecnológico que permita la reorganización controlada, el cumplimiento de las políticas y bajar los costos al mismo tiempo.

SOA promete de manera impactante el mejoramiento alineado a IT de los procesos de negocio. Sin embargo, para alcanzar esta visión SOA requiere de infraestructura que pueda integrar cualquier recurso con el que cuente IT, en cualquier parte que se encuentre y en cualquier momento que se requiera alcanzar. Para ser flexible, esto requiere de una infraestructura que pueda ser fácilmente escalable y extenderse sin ninguna perturbación, además no menos importante, la infraestructura deberá ser robusta y segura, esta infraestructura es llamada Enterprise Service Bus o Bus de Servicios Empresariales.

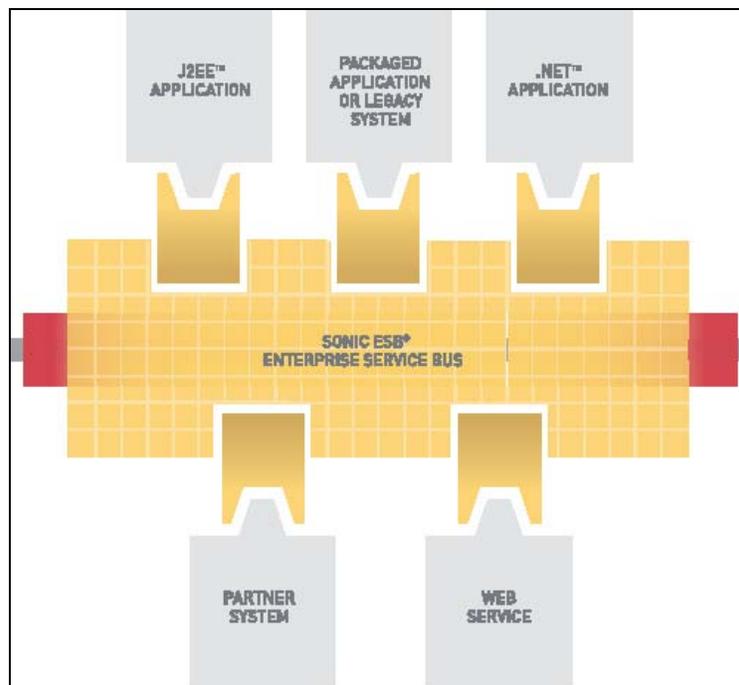


Figura 11. Esquema general de operación de un ESB.

Progress Sonic ESB elimina la rigidez y fragilidad de la integración punto a punto con una arquitectura robusta, manejada por eventos la cual puede evolucionar, escalar y extenderse a través de toda la empresa. A través de redes amplias como LAN o WAN, dominios de seguridad y límites organizacionales, Progress Sonic ESB permite la administración de la implementación de servicios distribuidos y la ejecución de servicios de integración escalables. Además cuenta con una edición que soporta la alta disponibilidad, llamada Progress Sonic Continuous Availability Architecture (CAA), en

estos casos solo Sonic puede garantizar la entrega en tiempo y continua de eventos de negocio de misión crítica.

Sonic ESB es una suite de productos destinados entre otras cosas a la integración de aplicaciones heterogéneas en ambientes totalmente heterogéneos y también para la habilitación de una SOA, lo cual puede visualizarse como la base de la pirámide tecnológica sobre la cual se implementarán y evolucionarán los procesos de negocio de una empresa.

Progress Sonic ESB es un bus de servicios empresariales, basado en mensajería el cual simplifica la integración y flexible reutilización de aplicaciones de negocio dentro de una arquitectura orientada a servicios o mejor conocida como SOA (Service Oriented Architecture).

Algunos de los principales beneficios de Sonic ESB son:

- ✚ Integración de tecnologías y aplicaciones fuera de rango
- ✚ Manejo de arquitecturas distribuidas para el manejo de gran número de servicios de integración y servicios orquestados
- ✚ Comunicación basada en mensajería JMS, rápida, confiable y tolerante a fallas
- ✚ Integración dentro de los límites organizacionales y también a través de sitios remotos, utilizando protocolos estándares y lenguajes universales
- ✚ Interoperabilidad confiable, asíncrona y segura a través del uso de estándares avanzados para el uso de Web Services
- ✚ Solución en el mercado madurada, líder y recomendada por usuarios y analistas de negocios
- ✚ Integración flexible de nuevas aplicaciones en el ambiente de IT existente
- ✚ Elimina la rigidez y fragilidad de la integración punto a punto de aplicaciones y servicios
- ✚ Altos niveles de confiabilidad y disponibilidad para asegurar la continuidad de los procesos de negocio
- ✚ Flexibilidad para adaptar nuevos procesos de negocio
- ✚ Fácil adopción basada en estándares tecnológicos

Familia de Productos Progress Sonic ESB.

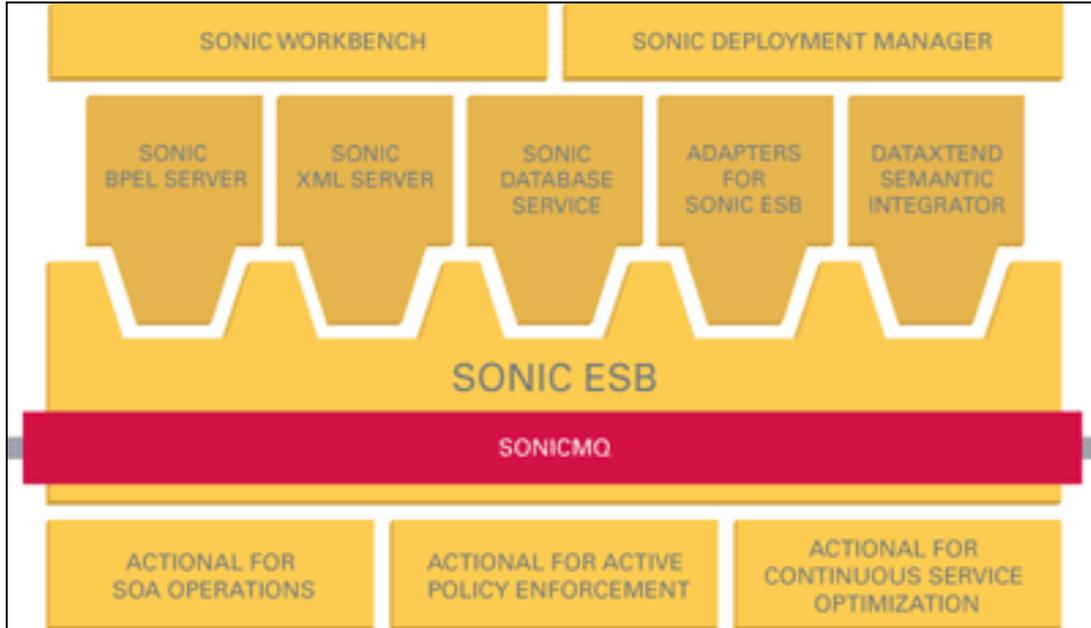


Figura 12. Familia de productos ESB.

Sonic ESB cuenta con una serie de productos que conforman la familia de productos enfocados hacia la orquestación de servicios, administración operacional de datos e integración de diferentes fuentes de datos relacionales, aplicaciones y tecnologías empaquetadas. La familia de productos Sonic ESB incluye:

- ✚ **Sonic ESB.** Es un bus de servicios empresariales o Enterprise Service Bus que simplifica la integración y la flexible reutilización de aplicaciones dentro de una arquitectura orientada a servicios (SOA).
- ✚ **Sonic Workbench.** Es una herramienta de integración SOA basada en Eclipse la cual sirve para modelar, configura, probar, desarrollar servicios y orquestar servicios utilizando los productos de la familia Sonic ESB, dicho de otra forma, Sonic Workbench es el ambiente de desarrollo de servicios preparados para la implementación sobre un ESB.
- ✚ **Sonic BPEL Server.** Este producto adiciona el estándar basado en la orquestación de servicios para el ruteo inteligente de componentes y servicios Sonic ESB.
- ✚ **Sonic Database Service.** Este componente simplifica y reutiliza los componentes para el acceso a fuentes de datos en una SOA.
- ✚ **Sonic Deployment Manager.** Es una herramienta que permite realizar de manera rápida y flexible la instalación y actualización de forma incremental, de todos los ambientes de trabajo hacia un ambiente de producción, apoyándose e integrándose con herramientas de control de versiones para facilitar el deployment

y la administración de entornos de desarrollo, calidad y producción a los equipos de desarrollo y administración de proyectos.

- ✚ **Sonic XML Server.** Proporciona procesamiento XML de alta velocidad, almacenamiento y servicios de búsqueda basados en XML como estándar.
- ✚ **Adaptadores para Sonic ESB.** Se cuenta con una lista de adaptadores para la integración de aplicaciones los cuales simplifican la incorporación de más de 200 tipos de sistemas de TI internos y externos dentro de una arquitectura orientada a servicios.
- ✚ **Actional.** La familia de productos Sonic ESB aprovecha Progress Actional para la administración de SOA, incrementar la visibilidad y control en Sonic ESB y en la infraestructura que conecta. Actional detecta automáticamente el nivel de servicio, confirma inmediatamente las causas, detecta los rastros de manera visual. Sólo Actional puede proporcionar esta capacidad con bajos costos y con la capacidad de escalar al mismo nivel que la arquitectura orientada a servicios SOA.
- ✚ **DataXtend SI.** La familia de productos Sonic ESB también incluye el desarrollo e integración en tiempo de ejecución con Progress DataXtend Semantic Integrator (SI), el cual simplifica de manera impactante la problemática de la administración del ciclo de vida de modelos de datos, transformación y validación de los mismos, basados en el estándar DSI. DataXtend SI perfecciona tanto el desarrollo como la ejecución a través de herramientas contenidas en Eclipse, además habilita a Sonic ESB para la ejecución de servicios destinados a la integración semántica (SI – Semantic Integration) de manera automática. Al igual que Sonic ESB proporciona la habilidad para eliminar la rigidez en las arquitectura de integración punto a punto, DataXtend SI soluciona el problema de transformaciones punto a punto, haciendo esta tarea mucho más sencilla para la integración de datos y para la evolución en conjunto con la arquitectura orientada a servicios SOA con diversos y diferentes sistemas interconectados.

SONIC ESB

Sonic ESB, es el primer Canal de Servicios Empresariales (Enterprise Service Bus), el cual combina mensajería basada en estándares, Web Services, transformación XML y ruteo inteligente para hacer una conexión confiable y coordinar la interacción de las aplicaciones en toda la Empresa.

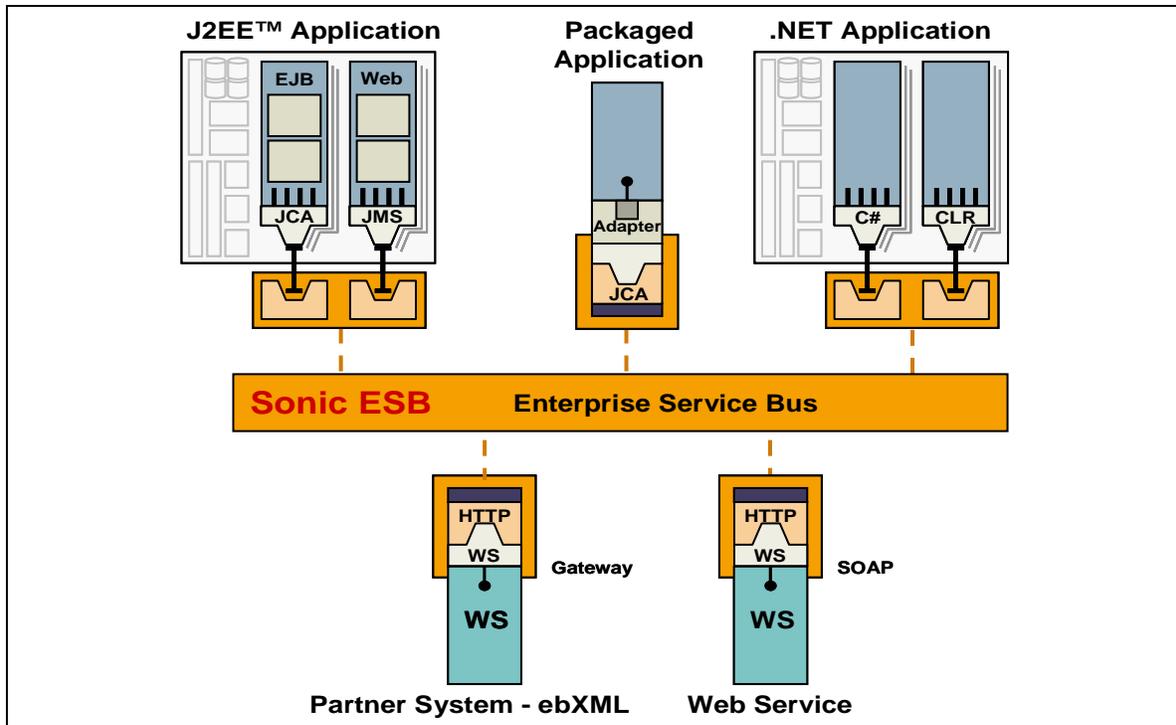
El ESB es un preempacado de una implementación SOA, diseñando para hacer fácil y efectiva la liberación de proyectos a nivel departamental y empresarial.

A diferencia de las soluciones tradicionales de integración de administración de la Cadena de Suministro, las cuales son pobres en Escalabilidad, Sonic ESB emplea un canal flexible y ligero el cual no esta ligado a ninguna arquitectura. Esta arquitectura del Canal permite a las organizaciones de Tecnología de la Información, ir creciendo de manera incremental, su base instalada, conforme se requieren resolver las necesidades y las demandas de Servicios en la Empresa.

Compañías de tipo Fortune 500, las compañías industriales más grandes en EEUU, han reconocido la necesidad de migrar a una arquitectura orientada a servicios para ligar sus aplicaciones dentro de la empresa y exponer sus sistemas en Internet. Algunas aplicaciones son expuestas como servicios que fácilmente pueden ser conectados con otras aplicaciones a lo largo de la empresa, creando así de manera efectiva una espina dorsal de estándares, o un Canal de Servicios Empresarial.

Consecuentemente, las integraciones punto a punto pueden ser eliminadas, ahora cuales quiera dos aplicaciones, pueden interactuar una con otra a través del Canal. Además, introducir una aplicación de un socio promoverá un efecto en la red, una aplicación ahora puede interactuar con otras en el Canal, y ellas pueden interactuar con esta a su vez.

Los WebServices y la arquitectura de conector J2EE (JCA) soportan este valor agregado ofreciendo estándares base para la integración de interfaces de aplicaciones departamentales que usan J2EE y .NET así como aplicaciones empaquetadas y legalizadas. Sonic ESB construye estas tecnologías de punta proveyendo las comunicaciones necesarias para conexiones confiables y seguras. En adición, Sonic ESB provee servicios XML, ruteo basado en contenido, transformación y procesos de administración requeridos para mejorar el desempeño entre aplicaciones, eliminando la necesidad de costosos brokers de integración. Sonic ESB además entrega un flujo de clase empresarial, escalabilidad global y el ser administrable, todo lo requerido para la integración de proyectos a gran escala.



El ESB (Enterprise Service Bus) contiene 7 áreas principales de funcionalidad.

Columna Vertebral de Clase Empresarial. Provee comunicación basada en estándares, confiable y segura, entre cualquier número de servicios y terminales de aplicación siendo cooperativo en transacciones a través de la Empresa Extendida. Utilizando una Arquitectura para el Ruteo Dinámico (DRA, tecnología por patentar), Web Services y procesos distribuidos que pueden ser escalables para resolver las demandas de las grandes empresas multinacionales. Esta tecnología termina con los riesgos de operaciones asociadas con el servicio y la integración de aplicaciones a través de las redes empresariales de escala global.

Ruteo Inteligente. Automatiza el ruteo de los documentos de negocio entre servicios dentro del ESB usando expresiones de reglas, contenido de documentos y atributos de mensaje. El ruteo de la información viaja con los mensajes y habilita a las terminales (endpoints) para rutear dinámicamente las comunicaciones sin la necesidad de tener un broker centralizado de integración. Esta metodología de ruteo elimina los cuellos de botella, un punto simple de falla y modelos rígidos de seguridad como el tradicional broker de Integración de la cadena de suministro.

Arquitectura de Servicios Distribuidos. Provee un marco coherente y administrable de publicación de servicios a través de los nodos cooperativos. Central a la arquitectura ESB, esta permite a los servicios ser administrables y escalables de manera independiente, y permite la expansión e integración a la red en cualquier tiempo con un costo bajo.

Soporte a WebServices. Anticipando una gran disponibilidad de WebServices y aplicaciones integradas, el ESB permite que las terminales Web Services sean fácilmente integradas en el ambiente del ESB.

Infraestructura de Seguridad Flexible. Seguridad en los servicios, aplicaciones y la comunicación entre ellos, ya que es apropiadamente restringida, dentro y fuera del Firewall. Provee capacidades para una comprensible, autenticación añadida, autorización, y encriptación a lo largo del ESB. La combinación de la embebida encriptación RSA y del amplio rango de suites que se pueden agregar, se facilita la integración siendo altamente segura y con un desempeño alto también. Estas características permiten la entrega de los proyectos de integración de nueva generación los cuales soportan las actuales políticas de seguridad de la empresa.

Servicio de transformación de XML. El servicio de transformación de ESB facilita la integración de múltiples fuentes de datos para la distribución a diversos destinos. La transformación de documentos XML a través de los servicios del ESB, se realiza por medio de XSLT o Extensible Stylesheet Language Transformations, lo cual es un estándar. Esto facilita el cumplir con formatos de datos entre terminales sin cambiar las aplicaciones de envío o recibimiento.

Marco Administrativo. Utiliza Estándares de distribución para configuración, liberación, administración y monitoreo de miles de servicios y terminales a través de la empresa desde una consola central de administración. Esto permite el mantenimiento de un significativo número de sistemas heterogéneos y servidores desde cualquier punto en el ESB. El marco administrativo facilita la expansión sistemática de la red de trabajo permitiendo a los administradores mantener el control del sistema, sin importar que tanto pueda crecer.

Sonic ESB resuelve los problemas de integración desde la conexión de aplicaciones departamentales hasta la integración de sistemas globales y procesos del negocio y sus asociados.

Sonic ESB es un Bus de servicios empresariales basado en estándares que combina mensajería, web services, transformación y Ruteo inteligente de mensajes, para conectar confiadamente y coordinar la integración de un número significativo de diversas aplicaciones a través de una alta distribución, extensión empresarial e integridad transaccional.

Soporte en Internet:

-  Compatible con firewalls
-  Alto rango de opciones de Implementación.
-  DRA permite comunicación cluster a cluster sobre Internet

Soporte a servidores de aplicación:

-  Soporte a los principales servidores de aplicaciones J2EE a través del mejor sistema de mensajería JMS.

Factores clave:

1. Maximiza las inversiones existentes en tecnología y ROI.
2. Mejora la productividad con un desempeño superior en aplicaciones y disponibilidad.
3. Incrementa la colaboración entre procesos, y socios de negocio a través de una siguiente generación de integración de aplicaciones.
4. Permite a los servicios ser fácilmente administrados, permitiendo una escalabilidad independiente.

Habilita la expansión de la red en cualquier momento, reduciendo el costo de propiedad.

2.1 Plataformas soportadas.

- Sonic ESB V7.6
- Sonic Workbench V7.6
- Sonic BPEL Server V7.6
- Sonic Database Service V7.6
- Sonic Deployment Manager V7.6
- Sonic XML Server V7.6

2.2 Sonic ESB V7.6

SISTEMA OPERATIVO Y VERSION	PROCESADOR	JVM Y VERSION
Microsoft Vista	x86-32	IBM V1.4.2 [32-bit]
		Sun V1.5.0 [32-bit]
Microsoft Windows XP Professional SP2	x86-32	IBM V1.4.2 [32-bit]
		Sun V1.4.2 [32-bit]
		Sun V1.5.0 [32-bit]
Microsoft Windows Server 2003 SP2	x86-32	IBM V1.4.2 [32-bit]
		Sun V1.4.2 [32-bit]
		Sun V1.5.0 [32-bit]
	x64	Sun V1.5.0 [64-bit]
Sun Solaris 9	SPARC	Sun V1.4.2\ [32-bit]
		Sun V1.5.0 [64-bit]
Sun Solaris 10	x86-32	Sun v1.4.2

		[32-bit]
	SPARC	Sun V1.5.0 [64-bit]
	x64	Sun V1.5.0 [64-bit]
Red Hat Enterprise Linux 4 Update 5	x86-32	Sun V1.4.2 [32-bit]
		Sun V1.5.0 [32-bit]
Red Hat Enterprise Linux 5	x86-32	Sun V1.4.2 [32-bit]
		Sun V1.5.0 [32-bit]
	x64	Sun V1.5.0 [64-bit]
SuSE Enterprise Linux V9.0 SP3	x86-32	Sun V1.4.2 [32-bit]
		Sun V1.5.0 [32-bit]
IBM AIX 5L V5.2 (Patch Level: 5200-01)	pSeries	IBM V1.4.2 [32-bit]
		IBM V1.5.0 [32-bit]
IBM AIX 5L V5.3 (Patch Level: 5300-04)	pSeries	IBM V1.4.2 [32-bit]
		IBM V1.5.0 [32-bit]
		IBM V1.5.0 [64-bit]
HP-UX V11i V2	PA-RISC	HP V1.4.2.11 [32-bit]
		HP V1.5.0

		[32-bit]
	Itanium	HP V1.5.0 [64-bit]
HP-UX V11i V3	Itanium	HP V1.5.0.03 [64-bit]

CAPÍTULO 3. SOA GOVERNANCE

Las definiciones de SOA Governance están de acuerdo en su propósito de ejercer el control, pero difieren en las responsabilidades que debería tener. Algunas definiciones se centran en imponer políticas y control de los servicios, mientras que otras definiciones utilizan una perspectiva orientada a negocios.

Con el objetivo de optimizar la inversión en SOA y cumplir los objetivos de negocio tales como incrementar la agilidad para hacer negocios, las organizaciones hoy día están volteando hacia SOA governance. Una estrategia efectiva para SOA governance no solo debe asegurar que el comportamiento de los proyectos sea alineado a los objetivos corporativos, sino también debe incluir temas de seguridad y cumplimiento de políticas que provienen de las regulaciones y demandas de negocio tales como Sarbanes-Oxley, MIFID y Visa PCI, asegurando que nada falle y controlando lo que no funcione de forma adecuada.

La familia de productos *Progress Actional* complementa las herramientas para SOA governance haciendo funciones de repositorio y almacenador de datos el cual provee la funcionalidad necesaria para el monitoreo de operaciones en una SOA con el objetivo de optimizar la salida de los procesos de negocio y con ello asegurar la ejecución de reglas de seguridad, políticas y reglas de negocio.

3.1 Lo más importante de SOA Governance: Asegurar la confiabilidad de las operaciones en la SOA.

Como un fundamento de control, las organizaciones deben estar habilitadas para entender que WebServices se encuentran operando dentro de su SOA, como operan o funcionan y quienes hacen uso de los servicios, con el objetivo de asegurar la confiabilidad de las operaciones. Pero una pregunta muy importante es, ¿Cómo se puede monitorear el desempeño de la SOA y la disponibilidad de la misma en un ambiente pobremente acoplado, en donde los servicios pueden ser reutilizados y ejecutados a través de diversos recursos de cómputo?

Progress Actional SOA Operations proporciona a las organizaciones la visibilidad sobre las necesidades que se tienen para asegurar la confiabilidad de la SOA mientras se elimina la mayoría de los códigos y configuraciones manuales requeridas por otras soluciones. Esta herramienta automáticamente descubrirá y monitoreará los servicios y disparará alertas derivadas de la disponibilidad y el desempeño. Estas alertas generan flujos gráficos que permitirán identificar los problemas o violaciones que han ocurrido dentro de la SOA.

3.2 Logrando los objetivos de negocio con SOA Governance.

Una vez que la SOA se encuentra operando de manera satisfactoria, la gobernabilidad debe ser establecida para controlar la calidad del servicio que la SOA debe aportar a los usuarios, socios de negocio y otros clientes y de esta forma alcanzar los niveles de servicio definidos a través de los SLA's (service level agreements).

Progress Actiona Continuous Service Optimization proporciona unicamente la perspectiva detallada de negocios para poder visualizar como es que la SOA está operando con todos los usuarios a nivel procesos. Actional puede capturar métricas a nivel de servicio, procesos e indicadores de negocio (key business indicators) relacionados con la capa inferior de la infraestructura de TI en múltiples dimensiones, por ejemplo, para clientes, grupos de clientes, regiones y segmentos de clientes previamente definidos. Esta información proporciona la base para la optimización del comportamiento de una SOA de manera dinámica o manual, a través del uso de mecanismos de control con Actional, esto con el objetivo de asegurar la calidad del servicio para los usuarios.

3.3 Reduciendo el riesgo en la seguridad y el cumplimiento de las reglas a través de la implementación de SOA Governance.

No existen áreas más importantes de una SOA que las que se encargan de implementar la seguridad y el cumplimiento de las reglas. Las faltas o violaciones en estas áreas pueden derivar en perdidas financieras, litigios y en general en pérdidas económicas para las empresas, pero SOA Governance en éstas áreas no resulta nada fácil de implementar ya que esto requiere de monitoreo estricto de información contenida dentro del contexto de los mensajes y la ejecución a través de diferentes servicios que conforman los procesos de negocio dentro de la SOA. Cuando los desarrolladores son los responsables en la determinación de las políticas en los servicios que están generando puede ocasionar un riesgo muy grande de inconsistencia de políticas y derivar en áreas de oportunidad muy importantes.

Progress Actional for Active Policy Enforcement permite centralizar la seguridad y acatar las políticas de administración con la ejecución de políticas distribuidas, esto permite a las organizaciones dejar las políticas en las manos de expertos en seguridad y cumplimiento de normas y autorizarlos para la generación de políticas, además de asegurar la aplicación de las mismas dentro de la SOA lo cual garantiza la reducción del riesgo y de los costos.

3.4 Descubriendo y controlando los servicios que generan problemas.

Los servicios maliciosos no detectados pueden ocasionar grandes problemas y más aún cuando se tienen expuestos y en contacto con personas no autorizadas o poco confiables, eludiendo el cumplimiento de políticas y de la auditoría. Aún cuando los servicios no sean maliciosos, éstos pueden afectar notablemente la capacidad planeada de los sistemas, entonces, que es SOA Governance si no es el control de lo que está ocurriendo y de quien está utilizando la SOA?

Progress Actional Governance Integration Module se integra con herramientas de gobernabilidad tales como Systinet y además proporciona contoles para la detección, interrupción y control de los servicios. Este módulo automáticamente descubre todos los servicios maliciosos y los detiene hasta que hayan sido revisados y autorizados.

A continuación se muestra una gráfica sobre los diferentes criterios de monitoreo de una SOA:

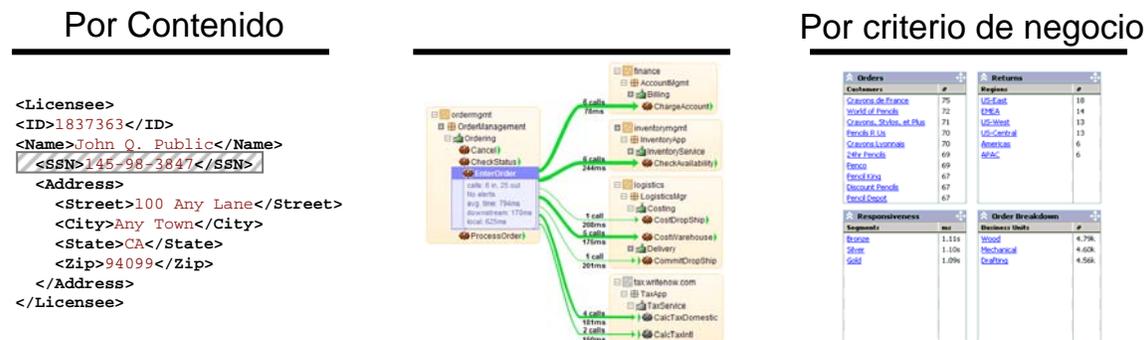


Figura 13. Criterios de monitoreo de SOA.

La siguiente pantalla muestra el panorama general de una SOA a través del uso de Progress Actional, herramienta que proporciona una vista gráfica sobre el uso de los servicios y habilita al equipo de TI para la habilitación de SOA Governance:

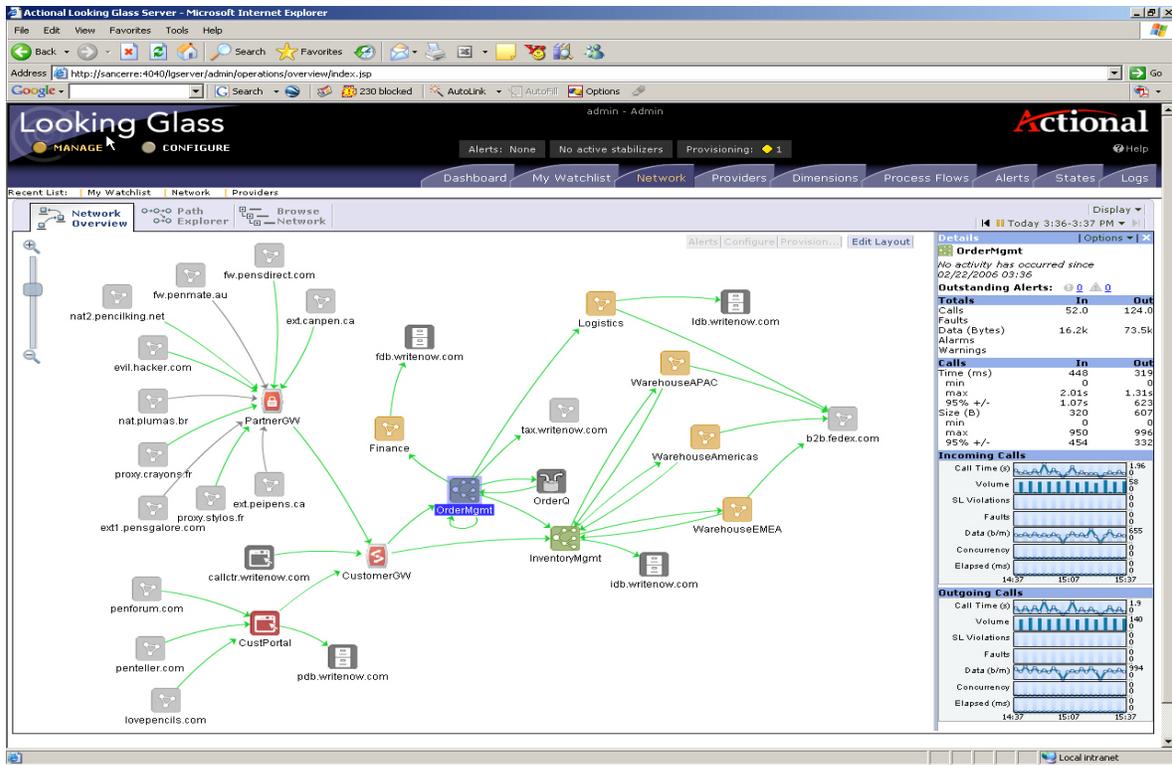


Figura 14. Vista del autodiscovery map.

CAPÍTULO 4. ESCENARIOS DE INTEGRACIÓN

En este apartado se verá brevemente algunos de los casos de uso más comunes en empresas mexicanas y norteamericanas, las cuales basan su operación en el comercio a través del uso de diferentes sistemas, protocolos y estándares.

Con este resumen de escenarios de integración, se pretende transmitir la visión tecnológica de Progress Software para la solución de problemas de integración, tomando ventaja de las características de una SOA (Flexibilidad, Escalabilidad, Seguridad y Desempeño).



Figura 15. Casos de uso comunes.

En la figura 15 se mencionan 4 escenarios de integración, basados por una parte en la distribución de datos, mientras que en otras se ve la complejidad para la consulta, transformación, extracción y garantía de entrega de los datos.

4.1 Caso 1. Proceso Continuo.

Problema: Garantizar el cumplimiento del ciclo del proceso y agilizar el envío y recepción de datos.

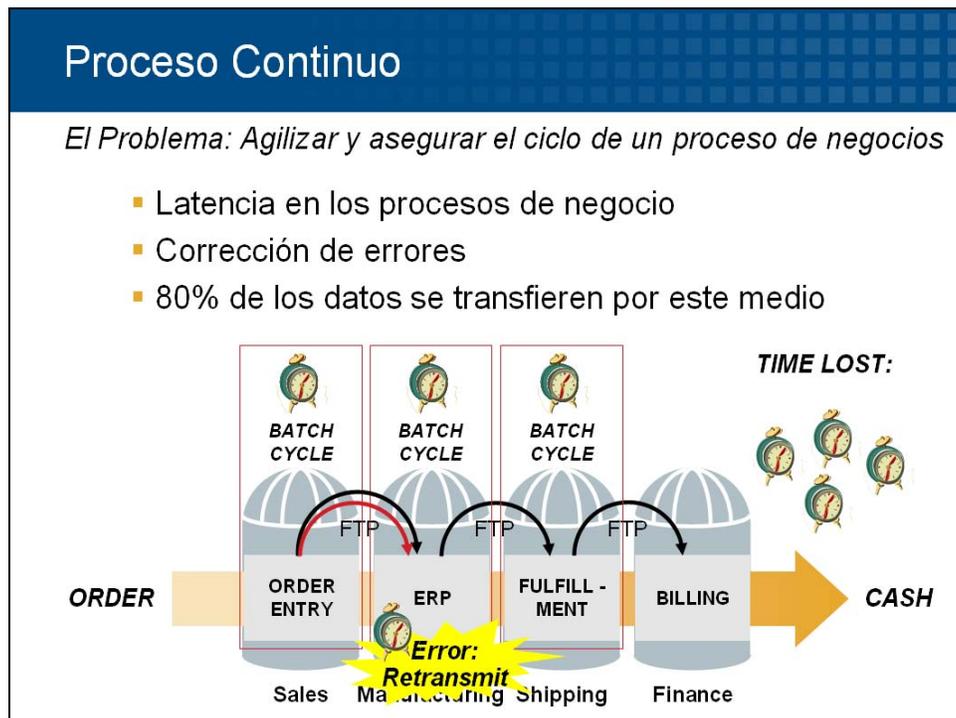


Figura 16. Requerimiento de proceso continuo.

En la figura 16 se observa un caso muy común y éste es cuando se cuenta con procesos de negocio (Business Processes) y no cuenta con los mecanismos adecuados para garantizar la entrega de información y la ejecución secuencial de una serie de pasos en su totalidad conforman el proceso.

Además en este caso se utilizan protocolos como FTP y sobre los cuales no se cuenta con puntos de control, también se requiere de la participación directa de una persona con conocimientos básicos del protocolo y que pueda asegurar el envío y recepción de información mediante mecanismos manuales de ejecución y también para el monitoreo.

El resultado típico de una arquitectura de este tipo es la pérdida de tiempo y dinero, factores que en la actualidad conforman el conjunto de métricas sobre las cuales son evaluados los dueños de los procesos de negocio. A la larga empresas que utilicen estos mecanismos no serán capaces de reaccionar con oportunidad a los requerimientos expuestos por la inercia del negocio o por las necesidades del equipo que toma las decisiones. Vale la pena señalar que en todos estos casos es muy importante e imprescindible la participación de la gente involucrada en el diseño y administración de procesos de negocio (decision makers) ya que de esto parte la necesidad de instrumentar

tecnologías y arquitecturas que satisfagan las oportunidades de negocio que harán subsistir a las empresas hoy en día.

Solución: Analizar, diseñar e implementar una arquitectura orientada a servicios, basada en estándares y servicios comunicados a través de mensajería, los cuales aporten visibilidad, control, agilidad y garantía de entrega de información.

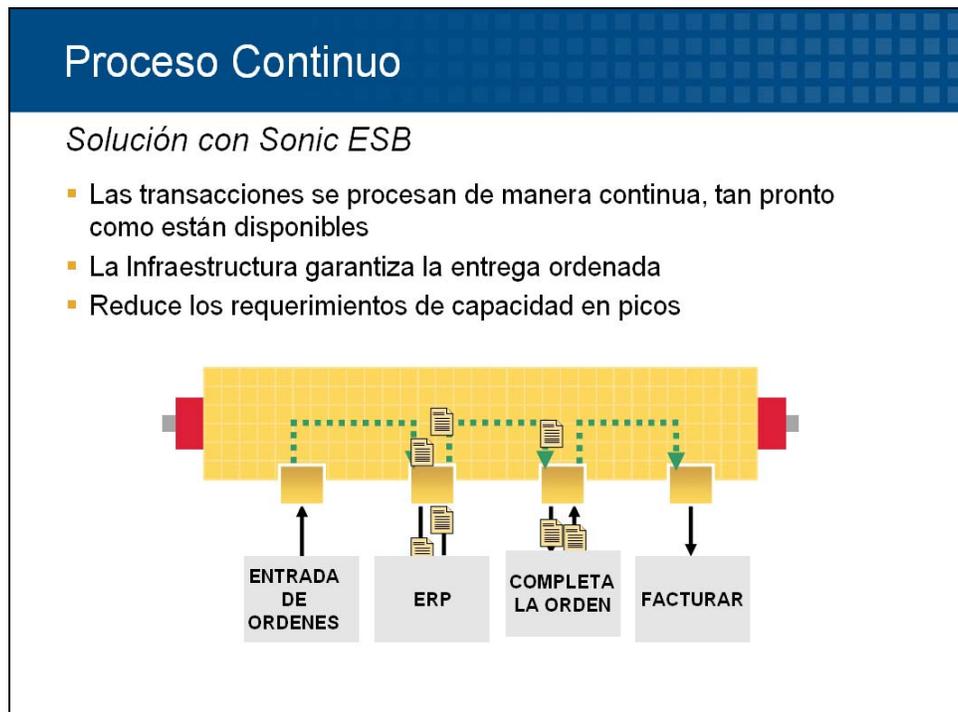


Figura 17. Arquitectura para proceso continuo.

Con base en las premisas de una SOA, la cual necesariamente tiene que cumplir con las siguientes características, se ha determinado que la solución al problema de proceso continuo es la implementación de servicios que permitan la interoperabilidad basando el esquema de comunicación y transporte a través de un Enterprise Service Bus (ESB):

- Confiabilidad
- Disponibilidad
- Optimo desempeño
- Viabilidad para distribuirla geográficamente
- Permitir la comunicación e integración de datos de plataformas y tecnologías totalmente heterogéneas
- Administrable
- Controlable

La arquitectura mostrada en la figura 17 ejemplifica la instrumentación de un proceso continuo, el cual delega la responsabilidad de garantía de entrega, transformación de

datos y conexión a las fuentes destino, a través de servicios (conjunto de lógica de negocios):

- ✚ Servicio de detección de archivos (File Polling)
- ✚ Servicio de transformación (xsl)
- ✚ Servicio de base de datos (DB Service)
- ✚ Servicio de escritura de archivos (File Writer)
- ✚ Servicio de invocación de servicios web (WebService invocation)

A través del mapeo, construcción e integración del flujo de proceso basado en los servicios mencionados con anterioridad, se cumple con los requisitos para garantizar el óptimo funcionamiento de la cadena de suministro, evitando puntos no controlables, dependencia de personas en horarios establecidos fuera de las jornadas de trabajo oficiales y modificación a la lógica de negocio de las aplicaciones.

4.2 Caso 2. Proceso Continuo.

Problema: Actualización de información en sitios y sistemas remotos.

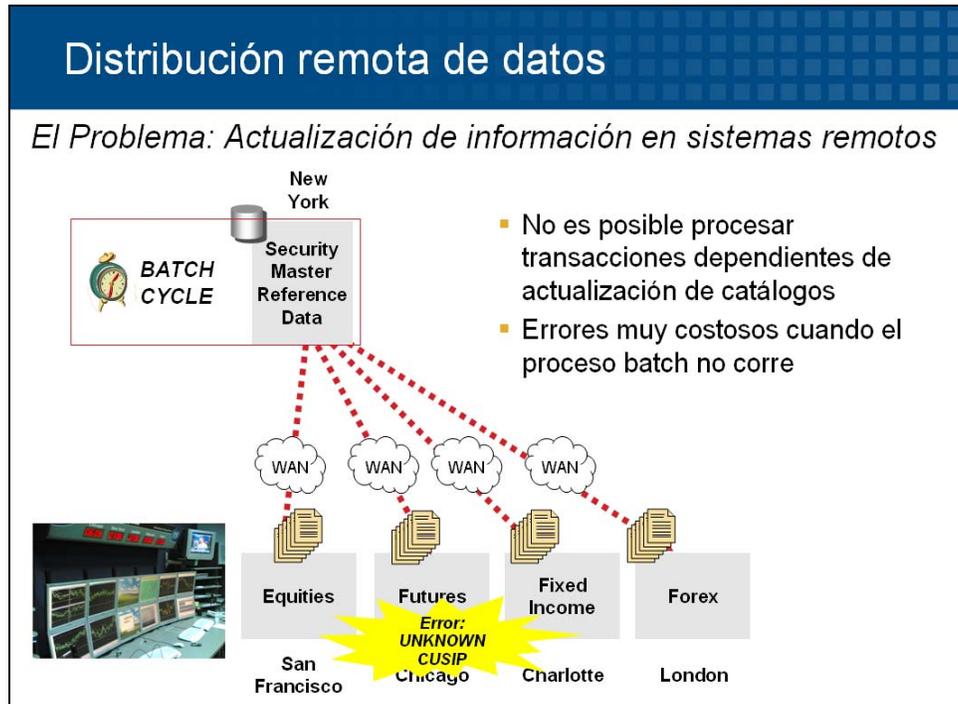


Figura 18. Requerimiento de distribución remota de datos

En la figura 18 se observa una problemática también muy común sobre empresas que cuentan con oficinas centrales y en diferentes entidades cuentan con sucursales o franquicias. En este caso es muy común la necesidad de distribuir información tal como catálogos de clientes, listas de precios, catálogo de empleados, promociones, etc.

Comúnmente se utilizan esquemas para el envío de información en los cuales se obliga a las aplicaciones generadoras de información para que se conecten directamente a los destinos y realicen el envío, sin embargo en otros casos se recurre a herramientas y métodos que no fueron diseñados para garantizar en tiempo y forma la entrega y recepción de información tan valiosa y confidencias para las empresas. Actualmente se utilizan los siguientes métodos:

- ✚ Correo electrónico
- ✚ FTP
- ✚ CGI
- ✚ Procesos manuales (CD, DVD, diskette)

El uso de éstos mecanismos obliga al uso de métodos manuales en los cuales se involucran puntos de error y manipulación de información que a la larga representan

pérdidas millonarias para las empresas, ya que los errores son muy comunes por el mal manejo de la información por una parte, mientras que en otros casos la información no llega a su destino, lo cual representa pérdida de tiempo, consumo excesivo de recursos y pérdidas en las ventas y facturación de los negocios.

La solución: La implementación de una arquitectura orientada a servicios (SOA por sus siglas en inglés) que permita agilizar el envío de información y garantice la entrega en tiempo y forma.

Distribucion Remota de Datos

Solución con Sonic ESB

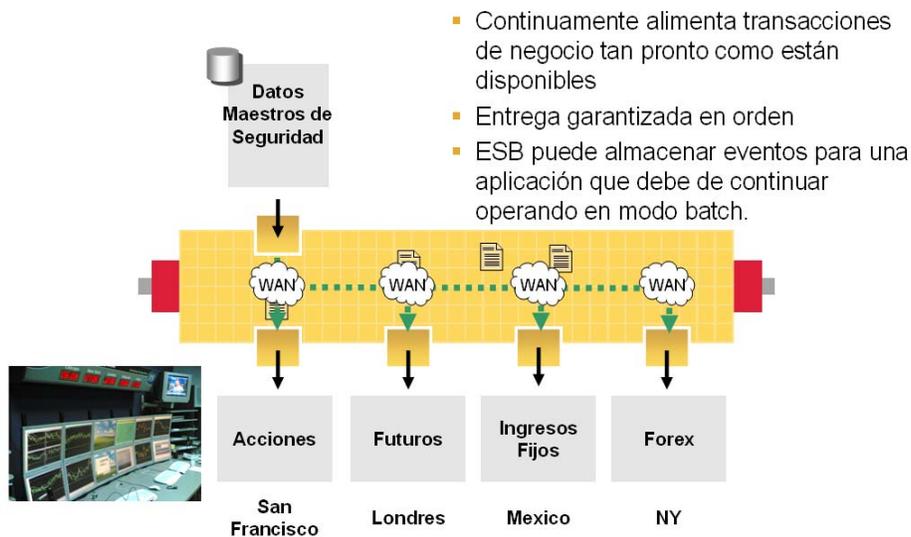


Figura 19. Escenario para distribución remota de datos

La solución a este problema de integración está basada además de los supuestos que debe tener una SOA (Service Oriented Architecture) o Arquitectura Orientada a Servicios, es la reutilización, misma que se alcanza al utilizar componentes estándar tales como JMS, XML, SQL, HTTP y SOAP.

Los servicios requeridos para la implementación de esta arquitectura son:

- 🔗 Servicios web (Web Services) basados en WSDL (Web Service Definition Language)
- 🔗 Servicio de base de datos (DB Service)
- 🔗 Servicio de notificación (Notification Service)

En este caso el diseño del flujo de procesos se basa en la comunicación síncrona de cada uno de los pasos, esperando la notificación de entrega y garantizando el envío utilizando arquitecturas de alta disponibilidad.

Adicionalmente se utilizarán conceptos como JMS para la conexión vía tcp hacia el Bus de datos (ESB), con ello utilizarán también mecanismos de persistencia y durabilidad, garantizando con ello que la información tendrá un factor de 0% de pérdida de datos.

Un servicio muy importante es la notificación, ya que una vez que se hayan cumplido las reglas de negocio y los niveles de servicio esperados, deberá notificarse a través de un medio basado en las tecnologías y reglas establecidas por las empresas, éstos mecanismos pueden basarse en notificaciones SMTP, SNMP o incluso mensajes que se transformen en la entrada hacia otro proceso transaccional.

CAPÍTULO 5. CASOS REALES

Problema: Eliminar los costos asociados por el manejo de inventarios de proveedores, agilizar el proceso de cadena de suministro e interconectar los diferentes sistemas que intervienen en el proceso de venta.

Arquitectura inicial (antes de la implementación de una SOA)

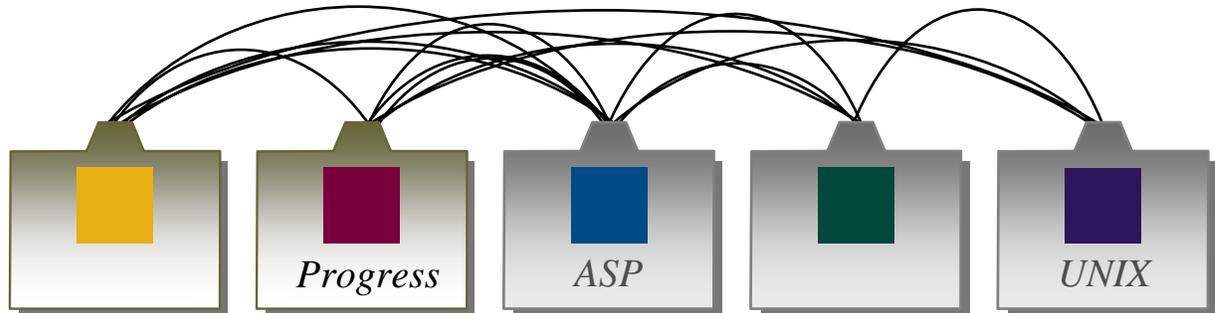


Figura 20. Arquitectura inicial tradicional.

Como se puede observar en la figura 20, la empresa dedicada a la venta de armazones de lentes y construcción de micas para las armazones (en este caso le llamaremos Ópticas Alex), cuenta con diferentes sistemas en plataformas totalmente heterogéneas tales como Microsoft, Unix, Perl, Progress y algunas otras de uso libre.

Ópticas Aex, además cuenta con aproximadamente 300 sucursales distribuidas en la República Mexicana y Centro América, distribuye diferentes marcas de renombre y mantiene relaciones comerciales con distribuidores localizados en Estados Unidos de Norte América, Asia y Europa, lo cual le exige tener mecanismos de intercambio de información ágiles, sin embargo no contaba con infraestructura que le permitiera comunicarse en línea ni en tiempo real.

Por otra parte contaba con manejo de inventarios de diferentes productos en todas sus sucursales, con el objetivo de satisfacer las necesidades de surtimiento basados en los niveles de ventas.

Conclusión:

- ✚ Se tienen niveles muy altos de transaccionalidad
- ✚ Se cuenta con un número muy elevado de sucursales lo cual deriva en el tráfico excesivo de información entre las sucursales y oficinas centrales
- ✚ Se cuenta con diferentes plataformas operativas y de sistemas, lo cual representa un gasto excesivo en el desarrollo de interfaces para la integración de información
- ✚ Se requiere de respuestas en línea y tiempo real con diferentes proveedores, lo cual exige la implementación de una arquitectura confiable y robusta para el intercambio de información
- ✚ Se tiene la necesidad de disminuir costos asociados por el manejo de inventarios

- ✚ Se deberán diseñar sistemas que permitan reducir los tiempos de implementación de nuevas aplicaciones y que además estén preparadas para operar de acuerdo a los estándares de comunicación actuales

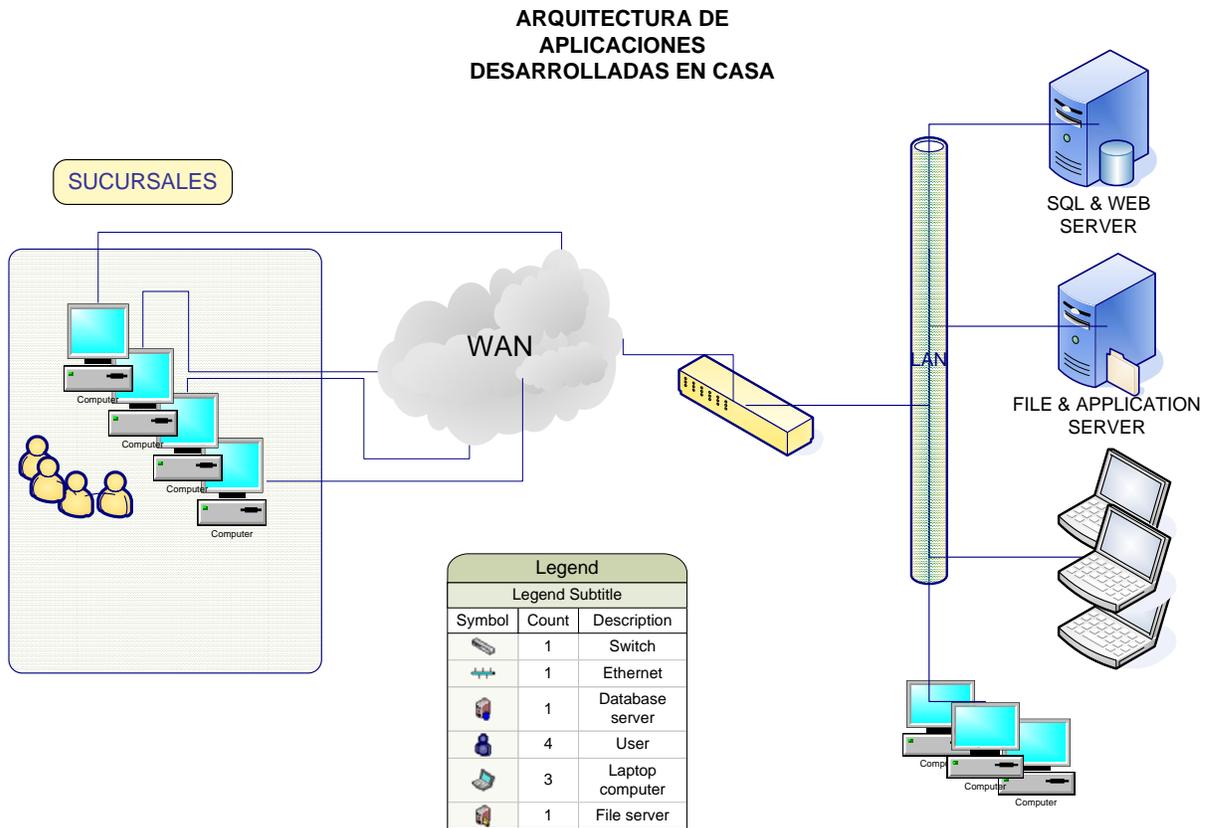


Figura 21. Arquitectura general de ópticas Alex.

La figura 21 ejemplifica de forma general la arquitectura bajo la cual operan Ópticas Alex y sobre las cuales deberá analizarse y diseñarse una arquitectura orientada a servicios, la cual principalmente deberá contar con elementos de reutilización, control y garantía.

Solución: Implementación de una arquitectura que incluya las siguientes características:

- ✚ Comunicación basada en estándares JMS, HTTP, SOAP, XML, WSDL
- ✚ Reutilización de componentes actuales y evitar la recodificación de sistemas
- ✚ Habilidad de mecanismos basados en protocolos redundantes para evitar la pérdida de información
- ✚ Servicios diseñados con base en las tecnologías que utilizan y enfocados a la transaccionalidad y garantía de entrega

Arquitectura propuesta.

Integración mediante una sola plataforma

- Flexibilidad para el mapeo de procesos de negocio que requieren integrarse
- Confiabilidad para conectar y orquestar la interacción de servicios distribuidos
- Agilidad para la incorporación de nuevos sistemas
- Administración centralizada

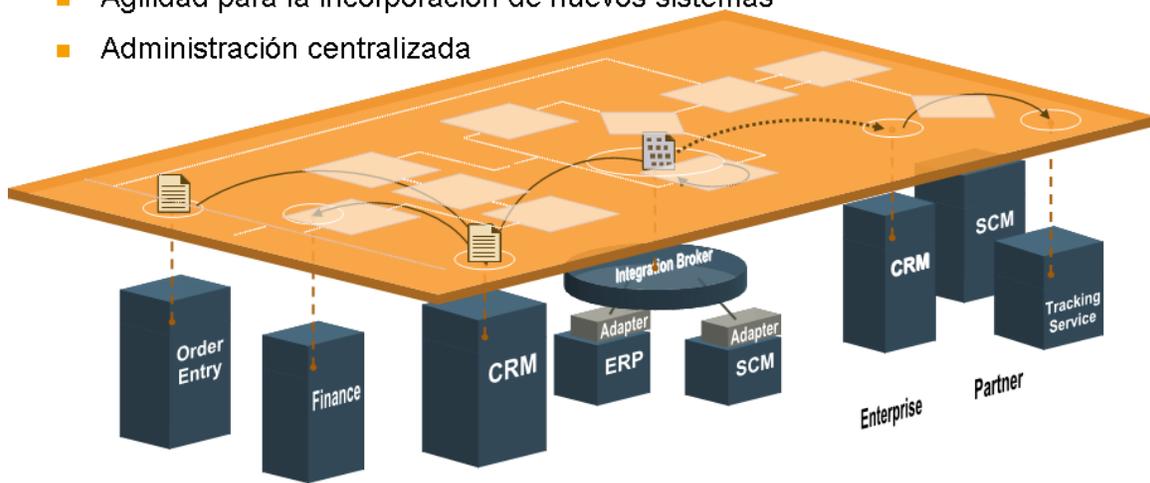


Figura 22. Integración mediante una sola plataforma.

Ópticas Alex realizó la implementación de una arquitectura basada en servicios a través de Progress Sonic ESB, utilizando componentes (servicios) que permitieron la comunicación síncrona a través de la red de las ópticas, reutilizando las tecnologías actuales y sacando ventaja de las características de los sistemas implementados.

Esta arquitectura permitirá establecer de manera dinámica los puntos de acceso a los servicios y operaciones que son requeridos y expuestos a través de la red, ésta arquitectura además permitirá implementar herramientas que en la pirámide de madurez (SOA Maturity Model) habilite la arquitectura para la evolución e implementación de nuevas características tales como el monitoreo, control y administración de servicios.

Adicionalmente se tendrá la capacidad de reaccionar ante diferentes requerimientos o necesidades propias de los procesos de negocio, con este esquema Opticas Alex estableció mecanismos de comunicación basada en servicios, mismos que le permiten

reaccionar en tiempo, consultar y enviar información a través de todos los miembros de la arquitectura, sin necesidad de recodificar sistemas ni programar interfaces punto a punto.

CONCLUSIONES

Los sistemas actuales, las empresas, los procesos de negocio y las personas han tratado de relacionar todo el ámbito en el que se relacionan a través de prácticas buenas, regulares y malas, sin embargo, la relación costo – beneficio se ha visto beneficiada al implementar herramientas y métodos que permitan visualizar como un todo el ambiente de trabajo y familiar y además el poder observar el entorno, su comportamiento y las áreas de oportunidad que esto genera para el bienestar general.

A través de éste trabajo, he resumido las diferentes formas que existen para hacer convivir el mundo de sistemas a través de un entorno que cada día evoluciona y que exige la implementación de buenas prácticas para poder llegar a los objetivos.

Definitivamente SOA es una filosofía y un planteamiento que permite pensar y diseñar sistemas capaces de evolucionar y cambiar conforme el entorno de negocios va cambiando, también hemos podido comprobar que es necesario poner especial atención en el comportamiento de las arquitecturas con el objetivo de mejorar y garantizar los niveles de servicio que las empresas exigen.

A través Progress Sonic ESB, Progress Accional y el modelo de madurez que se plantea es posible llegar a integrar cualquier cantidad de sistemas sin importar la tecnología, procesadores y sistemas operativos sobre los cuales operen.

GLOSARIO

BPEL. *Business Process Execution Language* (WS-BPEL), en castellano Lenguaje de Ejecución de Procesos de Negocio, es un lenguaje estandarizado por OASIS para la composición de servicios web. Está desarrollado a partir de WSDL y XLANG, ambos lenguajes orientados a la descripción de servicios Web. Básicamente consiste en un lenguaje basado en XML diseñado para el control centralizado de la invocación de diferentes servicios Web, con cierta lógica de negocio añadida que ayudan a la programación en gran escala (programming in the large).

BPEL es un lenguaje de orquestación, no un lenguaje coreográfico. La diferencia mayor entre ambos es el ámbito. Un modelo de orquestación provee un ámbito específicamente enfocado en la vista de un participante en particular (ej: un modelo par-a-par). En cambio, un modelo coreográfico abarca todos los participantes y sus interacciones asociadas, dando una vista global del sistema.

Las diferencias entre orquestación y coreografía están basadas en analogías: la orquestación describe un control central del comportamiento como un director en una orquesta, mientras que la coreografía trata sobre el control distribuido del comportamiento donde participantes individuales realizan procesos basados en eventos externos, como en una danza coreográfica donde los bailarines reaccionan a los comportamientos de sus pares.

A través de un documento XML BPEL, un analista de negocio es capaz de representar la lógica asociada y los elementos con los que se verá relacionado. Estos elementos serán servicios Web y la lógica el proceso BPEL.

BPM. Se llama Business Process Management (BPM) a la metodología empresarial cuyo objetivo es mejorar la eficiencia a través de la gestión sistemática de los procesos de negocio, que se deben modelar, automatizar, integrar, monitorizar y optimizar de forma continua.

Como su nombre sugiere, BPM se enfoca en la administración de los procesos del negocio. A través del modelado de las actividades y procesos puede lograrse un mejor entendimiento del negocio y muchas veces esto presenta la oportunidad de mejorarlos. La automatización de los procesos reduce errores, asegurando que los mismos se comporten siempre de la misma manera y dando elementos que permitan visualizar el estado de los mismos. La administración de los procesos permite asegurar que los mismos se ejecuten eficientemente, y la obtención de información que luego puede ser usada para mejorarlos. Es a través de la información que se obtiene de la ejecución diaria de los procesos, que se puede identificar posibles ineficiencias en los mismos, y actuar sobre las mismas para optimizarlos.

CGI. Interfaz de entrada común (en inglés Common Gateway Interface, abreviado CGI) es una importante tecnología de la World Wide Web que permite a un cliente (explorador web) solicitar datos de un programa ejecutado en un servidor web. CGI especifica un estándar para transferir datos entre el cliente y el programa. Es un mecanismo de

comunicación entre el servidor web y una aplicación externa cuyo resultado final de la ejecución son objetos MIME. Las aplicaciones que se ejecutan en el servidor reciben el nombre de CGIs.

FTP. FTP (File Transfer Protocol) es un protocolo de transferencia de archivos entre sistemas conectados a una red TCP basado en la arquitectura cliente-servidor, de manera que desde un equipo cliente nos podemos conectar a un servidor para descargar archivos desde él o para enviarle nuestros propios archivos independientemente del sistema operativo utilizado en cada equipo.

HTML. HTML, sigla de HyperText Markup Language (Lenguaje de Etiquetas de Hipertexto), es el lenguaje de marcado predominante para la construcción de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de "etiquetas", rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo Javascript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

HTML también es usado para referirse al contenido del tipo de MIME text/html o todavía más ampliamente como un término genérico para el HTML, ya sea en forma descendida del XML (como XHTML 1.0 y posteriores).

LAN. Una red de área local, o red local, es la interconexión de varios ordenadores y periféricos. (LAN es la abreviatura inglesa de Local Area Network, 'red de área local'). Su extensión está limitada físicamente a un edificio o a un entorno de hasta 100 metros. Su aplicación más extendida es la interconexión de ordenadores personales y estaciones de trabajo en oficinas, fábricas, etc., para compartir recursos e intercambiar datos y aplicaciones. En definitiva, permite que dos o más máquinas se comuniquen.

El término red local incluye tanto el hardware como el software necesario para la interconexión de los distintos dispositivos y el tratamiento de la información.

MAINFRAME. Una computadora central o mainframe es una computadora grande, potente y costosa usada principalmente por una gran compañía para el procesamiento de una gran cantidad de datos; por ejemplo, para el procesamiento de transacciones bancarias.

OASIS. OASIS, acrónimo de (Organization for the Advancement of Structured Information Standards) es un consorcio internacional sin fines de lucro que orienta el desarrollo, la convergencia y la adopción de los estándares de comercio electrónico y servicios web. Los miembros del consorcio deciden cómo y qué trabajo se realiza mediante un proceso abierto y democrático.

El trabajo técnico se lleva a cabo en las siguientes categorías: Web Services, e-Commerce, Security, Law & Government, Supply Chain, Computing Management, Application Focus, Document-Centric, XML Processing, Conformance/Interop, e Industry Domains.

PROCOLO. Protocolo de red o también Protocolo de Comunicación es el conjunto de reglas que especifican el intercambio de datos u órdenes durante la comunicación entre las entidades que forman parte de una red.

RED. Una red de computadoras (también llamada red de ordenadores o red informática) es un conjunto de equipos (computadoras y/o dispositivos) conectados, que comparten información (archivos), recursos (CD-ROM, impresoras, etc.) y servicios (acceso a internet, e-mail, chat, juegos), etc.

Para simplificar la comunicación entre programas (aplicaciones) de distintos equipos, se definió el Modelo OSI por la ISO, el cual especifica 7 distintas capas de abstracción. Con ello, cada capa desarrolla una función específica con un alcance definido.

SMTP. Simple Mail Transfer Protocol (SMTP), o protocolo simple de transferencia de correo. Protocolo de red basado en texto utilizado para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos (PDA's, teléfonos móviles, etc.). Está definido en el RFC 2821 y es un estándar oficial de Internet.

SNMP. El Protocolo Simple de Administración de Red o SNMP es un protocolo de la capa de aplicación que facilita el intercambio de información de administración entre dispositivos de red. Es parte de la familia de protocolos TCP/IP. SNMP permite a los administradores supervisar el desempeño de la red, buscar y resolver sus problemas, y planear su crecimiento.

Las versiones de SNMP más utilizadas son dos: SNMP versión 1 (SNMPv1) y SNMP versión 2 (SNMPv2). Ambas versiones tienen un número de características en común, pero SNMPv2 ofrece mejoras, como por ejemplo, operaciones adicionales.

SNMP en su última versión (SNMPv3) posee cambios significativos con relación a sus predecesores, sobre todo en aspectos de seguridad, sin embargo no ha sido mayoritariamente aceptado en la industria.

SOAP. Acrónimo del inglés Simple Object Access Protocol "Protocolo Simple de Acceso a Objetos". Protocolo para el intercambio de mensajes que utiliza XML y es independiente de la plataforma o sistema operativo, utilizado para codificar información en los diálogos de los Servicios Web.

SQL. El Lenguaje de consulta estructurado (SQL Structured Query Language) es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre las mismas. Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo lanzar consultas con el fin de recuperar -de una forma sencilla- información de interés de una base de datos, así como también hacer cambios sobre la misma. Es un lenguaje de cuarta generación (4GL).

W3C. El World Wide Web Consortium, abreviado W3C, es un consorcio internacional que produce estándares para la World Wide Web. Está dirigida por Tim Berners-Lee, el creador original de URL (Uniform Resource Locator, Localizador Uniforme de

Recursos), HTTP (HyperText Transfer Protocol, Protocolo de Transferencia de HiperTexto) y HTML (Lenguaje de Marcado de HiperTexto) que son las principales tecnologías sobre las que se basa la Web.

WAN. Una Red de Área Amplia (Wide Area Network o WAN, del inglés), es un tipo de red de computadoras capaz de cubrir distancias desde unos 100 hasta unos 1000 km, dando el servicio a un país o un continente. Un ejemplo de este tipo de redes sería RedIRIS, Internet o cualquier red en la cual no estén en un mismo edificio todos sus miembros (sobre la distancia hay discusión posible). Muchas WAN son construidas por y para una organización o empresa particular y son de uso privado, otras son construidas por los proveedores de Internet (ISP) para proveer de conexión a sus clientes.

WS (WEBSERVICE). Un servicio web (en inglés Web service) es un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de computadoras como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios Web. Para mejorar la interoperabilidad entre distintas implementaciones de servicios Web se ha creado el organismo WS-I, encargado de desarrollar diversos perfiles para definir de manera más exhaustiva estos estándares.

WS-I. La Organización para la Interoperabilidad de Servicios Web (Web Services Interoperability Organization) es un esfuerzo de la industria en este sentido. Su objetivo es fomentar y promover la Interoperabilidad de Servicios Web (Web Services Interoperability - WS-I) sobre cualquier plataforma, sobre aplicaciones, y sobre lenguajes de programación. Su intención es ser un integrador de estándares para ayudar al avance de los servicios web de una manera estructurada y coherente. La WS-I ha organizado los estándares que afectan a la interoperabilidad de los servicios web en una pila basada en funcionalidades.

WSDL. WSDL son las siglas de Web Services Description Language, un formato XML que se utiliza para describir servicios Web. La versión 1.0 fue la primera recomendación por parte del W3C y la versión 1.1 no alcanzó nunca tal estatus. La versión 2.0 se convirtió en la recomendación actual por parte de dicha entidad.

WSDL describe la interfaz pública a los servicios Web. Está basado en XML y describe la forma de comunicación, es decir, los requisitos del protocolo y los formatos de los mensajes necesarios para interactuar con los servicios listados en su catálogo. Las operaciones y mensajes que soporta se describen en abstracto y se ligan después al protocolo concreto de red y al formato del mensaje.

Así, WSDL se usa a menudo en combinación con SOAP y XML Schema. Un programa cliente que se conecta a un servicio web puede leer el WSDL para determinar que funciones están disponibles en el servidor. Los tipos de datos especiales se incluyen en el archivo WSDL en forma de XML Schema. El cliente puede usar SOAP para hacer la llamada a una de las funciones listadas en el WSDL.

XML. XML, sigla en inglés de Extensible Markup Language («lenguaje de etiquetas extensible»), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C). Es una simplificación y adaptación del SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML). Por lo tanto XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades. Algunos de estos lenguajes que usan XML para su definición son XHTML, SVG, MathML.

XML no ha nacido sólo para su aplicación en Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo y casi cualquier cosa imaginable.

XML es una tecnología sencilla que tiene a su alrededor otras que la complementan y la hacen mucho más grande y con unas posibilidades mucho mayores. Tiene un papel muy importante en la actualidad ya que permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil.

XSL. (siglas de Extensible Stylesheet Language, expresión inglesa traducible como "lenguaje extensible de hojas de estilo") es una familia de lenguajes basados en el estándar XML que permite describir cómo la información contenida en un documento XML cualquiera debe ser transformada o formateada para su presentación en un medio.

Esta familia está formada por tres lenguajes:

- ✚ XSLT (siglas de Extensible Stylesheet Language Transformations, lenguaje de hojas extensibles de transformación), que permite convertir documentos XML de una sintaxis a otra (por ejemplo, de un XML a otro o a un documento HTML).
- ✚ XSL-FO (lenguaje de hojas extensibles de formateo de objetos), que permite especificar el formato visual con el cual se quiere presentar un documento XML, es usado principalmente para generar documentos PDF.
- ✚ XPath, o XML Path Language, es una sintaxis (no basada en XML) para acceder o referirse a porciones de un documento XML.

Estas tres especificaciones son recomendaciones oficiales del W3C.

BIBLIOGRAFIA

1. <http://www.progress.com>
2. <http://www.sonicsoftware.com>
3. <http://www.actional.com>
4. <http://wikipedia.es>
5. Service Oriented Architecture for Dummies
Judith Hurwitz
Robin Bloor
Carol Baroudi
Marcia Kaufman
6. Enterprise Service Bus
Theory in Practice
David A. Chappell
7. Service Oriented Architecture
Getting it right
David Besemer, Paul Butterworth, Luc Clément, Jim Green, Hemant
Ramachandra, Jeff Schneider, Hub Vandervoort