

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)

 Powered by 	Hosting Patrocinado por enREDados.com 
---	--

[Home](#) | [Quienes Somos](#) | [Empleo](#) | [Tutoriales](#) | [Contacte](#)



CoNcept

Lanzado

TNTConcept versión 0.4.1 (04/06/2007)

Desde [Autentia](#) ponemos a vuestra disposición el software que hemos construido (100% gratuito y sin restricciones funcionales) para nuestra gestión interna, llamado TNTConcept (auTeNTia).

Construida con las últimas tecnologías de desarrollo Java/J2EE (Spring, JSF, Acegi, Hibernate, Maven, Subversion, etc.) y disponible en licencia GPL, seguro que a muchos profesionales independientes y PYMES os ayudará a organizar mejor vuestra operativa.

Las cosas grandes empiezan siendo algo pequeño Saber más en: <http://tntconcept.sourceforge.net/>

<p>Tutorial desarrollado por: Francisco Javier Martínez Páez</p> <p>Puedes encontrarme en Autentia Somos expertos en Java/J2EE Contacta en: fjmpaez@autentia.com</p>	<p>NUEVO CATÁLOGO DE SERVICIOS DE AUTENTIA (PDF 6,2MB)</p> <p>www.adictosaltrabajo.com es el Web de difusión de conocimiento de www.autentia.com</p>  <p>autentia real business solutions</p> <p style="text-align: center;">Catálogo de cursos</p>
--	---

Descargar este documento en formato PDF [SOA2.pdf](#)

Firma en nuestro libro de Visitas <-----> [Asociarme al grupo AdictosAlTrabajo en eConozco](#)

Master Experto Java

100% alumnos se colocan. Incluye Struts, Hibernate, Ajax
www.grupoatrium.com

SUN java

Formación y título con SUN ¡Aumente el nivel de su carrera!
es.sun.com/training

Integra datos dinámicos

en aplicaciones Java. Descarga una versión prueba de Crystal Reports.
www.businessobjects.es

Anuncios Google

Fecha de creación del tutorial: 2007-07-09

SOA (Service-Oriented Architecture)

INTRODUCCIÓN

Para empezar este tutorial, me gustaría comenzar con una serie de afirmaciones que creo que nadie me podría negar:

- Los procesos de negocio de las empresas son cada día más complejos.
- La evolución del mercado y la fuerte competencia exige a las empresas una respuesta más ágil para aprovechar la curva de oportunidad, a la hora de ofrecer distintos servicios a los clientes.
- Esta fuerte competencia implica que las empresas tengan que ofrecer servicios de mayor valor añadido apoyándose en acuerdos con otras empresas (por ejemplo, una línea aérea quiere ofrecer a través de su vez, no sólo reservar billetes en un vuelo, sino ofrecer una solución completa en base a itinerarios, hoteles, coches de alquiler, etc...)

Sin embargo, gran parte de los problemas con los que se encuentran las empresas para evolucionar acorde al mercado es la infraestructura tecnológica, o dicho de otro modo, la arquitectura.

A lo largo de los años, en nuestras empresas se han ido acumulando una gran cantidad de aplicaciones distintas que se fueron desarrollando para tratar de resolver las distintas necesidades que iban surgiendo: ERPs, CRMs, Bases de datos, Mainframes, Sistemas CICS junto con aplicaciones WEB / J2EE, .NET, etc... Pero estas aplicaciones no se crearon con la idea de interactuar entre ellas, sino como herramientas para resolver un problema en un momento dado. En definitiva, el paisaje que nos encontramos es parecido a un archipiélago de aplicaciones.

La realidad del mercado nos ha llevado a la siguientes aseveraciones:

- Nuestras aplicaciones internas están de alguna manera condenadas a entenderse, si queremos dar una respuesta ágil, a las necesidades de negocio que surgen.
- Las aplicaciones de nuestra empresa están condenadas a entenderse de alguna manera ágil con las aplicaciones de las empresas con las que queremos cooperar para ofrecer mejor servicio a nuestros clientes.

Es, en este cajón de sastre, donde SOA pretende hacer su aparición.

La palabra arquitectura en el mundo del software se podría definir como un conjunto de decisiones que hemos de tomar a la hora de organizar nuestras aplicaciones, como van a interactuar entre ellas, qué software se va a usar a la hora de comunicar entre ellas (MOMs, EAls etc...), qué plataformas, máquinas, sistemas operativos, lenguajes de programación, qué tipo de mensajes se van a intercambiar, etc...

Las decisiones que tomamos a la hora de decidir nuestra arquitectura son fundamentales, no tanto a corto plazo, sino más bien a largo plazo, y pueden ser a veces una trampa mortal.

SOA pretende ayudarnos a la hora de tomar este tipo de decisiones.

¿ QUÉ NO ES SOA ?

La mejor manera de comenzar a explicar SOA, es explicar qué NO es:

- SOA no es un software, no es un MOM, no es un EAI, aunque una arquitectura SOA puede apoyarse en un conjunto de herramientas como MOMs o EAls para conseguir su objetivo.
- SOA no es una metodología de proyecto, aunque a la hora de iniciar un proyecto orientado a conseguir una arquitectura SOA en mi empresa, algunas metodologías se ajustan mejor que otras: es preferible seguir un modelo en iteraciones (no confundir con iterativo como UP), como las metodologías ágiles (XP, etc..), que seguir metodologías Waterfall
- SOA no es otra forma de llamar a los WebServices, aunque los webservices son una herramienta válida para conseguir una arquitectura SOA, incluso una arquitectura SOA podría apoyarse en la programación distribuida (DCOM, CORBA, RMI, EJBs...)

¿ QUÉ ES SOA ?

Entonces, ¿ Qué es SOA ?.

Yo me atrevería a definir SOA "una arquitectura donde la pieza fundamental es el servicio."

Al ser una arquitectura, entendemos que es un conjunto de decisiones que debemos adoptar para montar nuestra infraestructura tecnológica.

Para poder continuar explicando la definición, urge definir previamente qué es un servicio. Primeramente, antes de continuar con la exposición, os planteo la siguiente pregunta: ¿ Cual es la duración media de los componentes que forman parte de la infraestructura tecnológica de una empresa ?

Una respuesta aproximada podría ser:

- Las innovaciones tecnológicas y productos nuevos aparecen aproximadamente cada 5 años.
- Nuestras aplicaciones empresariales tienen una duración de unos 8 a 10 años
- La lógica de negocio o los procesos de negocio permanecen en vigor unos 17 o 18 años.
- Los datos (o la estructura de los datos) perduran aproximadamente 25 años.

En base a esas respuestas, podemos intuir que a la hora de elegir como organizar nuestra infraestructura tecnológica, si nuestro objetivo es obtener la máxima reusabilidad y permanencia en el tiempo, debemos adoptar decisiones que vayan orientadas hacia la independencia tecnológica y al acercamiento al proceso de negocio y los datos.

Entonces, ¿ Qué es un servicio?.

Un servicio lo podríamos definir como la resolución de una necesidad de negocio, que debe ser autocontenida (es decir que no contenga la resolución de otra necesidad en el mismo) y que está constituido por tres partes bien diferenciadas:

1. Un contrato: que define la especificación del propósito del servicio, así como restricciones en su uso, calidad que debe ofrecer etc..., pero sin especificar nada acerca de la tecnología subyacente.
2. Un interfaz físico donde los clientes que quieren usar el servicio pueden invocarlo (podría ser una URL)
3. Una implementación: un servicio se apoya en alguna tecnología para realizar lo que se expone en su contrato. La implementación de un servicio podrá consistir en la interacción de distintos artefactos (EJBs, CORBA, Bases de datos ...), y estará compuesta de:
 1. Una lógica de negocio
 2. Una serie de datos.

La idea que presenta SOA es construir nuestra arquitectura en base a un gran conjunto de servicios independientes, localizables e invocables de manera transparente y agnósticos de la tecnología. Estos servicios que podríamos denominar básicos, permitirán construir servicios de mayor valor añadido mediante la composición o el engranaje de estos servicios básicos.

¿ ACTORES DE SOA ?

Una vez que hemos visto lo que es un servicio, os voy a presentar los demás actores que deberían participar en una arquitectura SOA:

1. **Frontales de aplicación.** Estos serán no tanto partes funcionales de una arquitectura SOA sino aquellos que se van a beneficiar de ella, es decir serán aquellos que quieren hacer uso de los servicios ofrecidos dentro de la arquitectura. Frontales de aplicación podrán ser tanto aplicaciones WEB, CRMs, ERPs..., así como procesos Batch que se ejecutan de manera nocturna, etc...
2. **Servicios.**
3. **Repositorio de servicios.** Un repositorio de servicios será algún componente de mi arquitectura que permitirá tanto a los frontales de aplicación como a otros servicios, descubrir que servicios existen, cual es su interfaz y donde se encuentran físicamente. Los objetivos de este componente serán:
 1. Crear un nivel de indirección para localizar a los servicios
 2. Servir como repositorio de información de los servicios existentes, contratos, calidad de los mismos, etc...
4. **Bus de servicios (ESB).** Este será un componente que permitirá a todos los participantes o actores de la arquitectura SOA comunicarse entre ellos. Este componente fundamental en la arquitectura SOA debe ofrecer:
 1. Conectividad entre frontales de aplicación y los servicios.
 2. Debe ser agnóstico de lenguajes de programación y tecnologías. Es decir debe ofrecer una forma de comunicación universal, para que todos puedan entenderse (por ejemplo, puede usar XML como formato de comunicación de los mensajes)
 3. Debe ser capaz de ofrecer diferentes paradigmas de comunicación (sincronismo y asincronismo).
 4. Debería ser capaz de ofrecer otra serie de funcionalidades transversales como:
 - Trazabilidad de las operaciones (logging)
 - Mecanismos de seguridad (autenticación, autorización, encriptación, no repudio ...)
 - Mecanismos de transaccionalidad: protocolo de commit en dos fases (2PC) o transacciones en cadena y mecanismos de compensación (SAGA) etc...
 - Enriquecimiento de mensajes, adaptación etc...
 - Control centralizado, mecanismos de monitorización.
 - Que incluyese un procesador de BPM, que permitiese construir servicios de mayor valor añadido en base a servicios básicos, simplemente definiendo la lógica en algún lenguaje del tipo BPEL o BPEL4WS, etc...

EPÍLOGO

Conseguir reorganizar la infraestructura tecnológica de nuestra empresa para que se encamine hacia una arquitectura SOA es un proceso muy complejo. He leído en algún sitio que el 70% de los proyectos que se embarcan en esta tarea fracasan.

No sólo nos encontraremos problemas tecnológicos, sino también muchos problemas debido a la política y cultura empresarial. Además, es un proceso largo (de años) que debe ser iniciado de manera muy cuidadosa. Es muy recomendable llevarlo a cabo de una manera gradual, y no todo de una vez, acompañando la tarea de llevarlo a cabo con una continua "evangelización" de las ventajas de SOA a todos los niveles de la empresa.

Es probable que debamos cambiar la metodología de proyectos para orientarlo hacia metodologías más ágiles (XP) etc...

Es probable también que debamos empezar a modelar siguiendo la filosofía de MDA (Model Driven Architecture), de una manera independiente de la

tecnología, partiendo desde los procesos de negocio para ir gradualmente obteniendo el modelado específico de la plataforma en la que vamos a realizar la implementación del proceso de negocio (o servicio).

Por mi parte nada más, espero que os haya servido para entender un poco de que va esto del SOA.



This work is licensed under a [Creative Commons Attribution-NonCommercial-No Derivative Works 2.5 License](https://creativecommons.org/licenses/by-nc-nd/2.5/).
[Puedes opinar sobre este tutorial aquí](#)



Recuerda

que el personal de [Autentia](#) te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#))

¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?

¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?

info@autentia.com

Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos

Autentia = Soporte a Desarrollo & Formación

[Autentia S.L.](#) Somos expertos en:
J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ..
y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
<i>e-mail</i>	<input type="text"/>
	<input type="button" value="Enviar"/>

Otros Tutoriales Recomendados ([También ver todos](#))

Nombre Corto

[Motivación para la mejora de procesos basada en CMMI](#)

[CRM: MODELO Y HERRAMIENTAS PARA DESARROLLO DE SOLUCIONES CRM](#)

[CRM: E-BUSINESS Y LOS NEGOCIOS EN LÍNEA](#)

[CMMI: Niveles de madurez 2 y 3](#)

[CMMI. Modelo de Madurez Software](#)

[SOA y el modelo de Negocio](#)

[Arquitectura SOA e Integración de aplicaciones](#)

Descripción

Gracias a la colaboración de Giuseppe Satriani podemos saber más sobre la motivación para la mejora de procesos basada en CMMI.

Cristhian Herrera comparte con nosotros su trabajo de fin de estudios sobre CRM. Es un extenso trabajo que seguro os gustará

Cristhian Herrera comparte con nosotros su trabajo de fin de estudios sobre CRM. Es un extenso trabajo que seguro os gustará

En este artículo os mostramos una pequeña redacción sobre los niveles de madurez 2 y 3 del CMMI

Os introducimos a CMMI o Capability Maturity Model Integration. CMMI es un modelo de calidad exigido por el gobierno americano a sus proveedores para el desarrollo de Software. Su conocimiento es esencial para reducir costes de desarrollo.

En este tutorial podemos ver una documento de Santiago Pereira sobre SOA, los estándares que se usan y la unión del SOA con los procesos de negocio.

Os presentamos un breve pero intenso pdf sobre la tecnología Arquitectura SOA y la Integración de aplicaciones

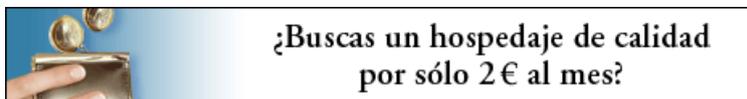
Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.

[Patrocinados por enredados.com Hosting en Castellano con soporte Java/J2EE](#)



www.AdictosAlTrabajo.com Optimizado 800X600