

Unidad 6 – Arquitectura Orienta a Servicios (SOA)

Introducción y orientaciones para el estudio

En el desarrollo de este módulo abordaremos:

- Introducción a conceptos de esta arquitectura y sus tecnologías usuales.
- Presentación de otros conceptos organizativos (contratos, gobierno, ciclo de vida y desarrollo, etc).

Objetivos

Pretendemos que al finalizar de estudiar esta unidad el alumno logre:

- Realizar un tour por los conceptos de SOA, definiendo sus principales conceptos.
- Entender las generalidades sus términos, criterios de organización y gobierno.
- Presentación de sugerencias, buenas prácticas y consejos para su implementación.

Aclaraciones previas al estudio

En este módulo, el alumno encontrará:

- Contenidos
- Conceptualizaciones centrales

Se debe tener presente que los contenidos presentados en el módulo no ahondan profundamente en el tema, sino que pretenden ser un recurso motivador para que, a través de la lectura del material, la bibliografía sugerida y el desarrollo de las actividades propuestas se alcancen finalmente los objetivos planteados.

Cada módulo constituye una guía cuya finalidad es facilitar el aprendizaje.

¿Qué es SOA?

Es un conjunto de patrones, principios y prácticas para construir piezas de software que puedan inter-operar independientemente de la tecnología.

Explora a los principios de la orientación a servicios para lograr una relación más estrecha entre el negocio y los sistemas de información lo soporta

Fundamentos de Servicios

Los servicios son el corazón de este estilo arquitectónico.

Una definición basada provista por IBM (Rational Method) nos dice:

“Un servicio es un recurso que realiza una tarea repetible, y es descrito por una especificación o contrato de servicios externalizados”.

“SOA no se trata solo de la “tecnología”, sino también sobre los requisitos y necesidades empresariales”.

El **TOGAF** The Open Group Architecture Framework ofrece dos definiciones para la arquitectura, en función del contexto:

“Una descripción formal de un sistema o un plan detallado del sistema a nivel de componente para orientar su aplicación.”

“La estructura de los componentes, sus interrelaciones y los principios y directrices que rigen su diseño y evolución en el tiempo. “



Figura 1 - Elementos de un servicio en SOA

Consideraciones para una arquitectura SOA

El siguiente gráfico nos muestra una serie de consideraciones para garantizar una arquitectura SOA:

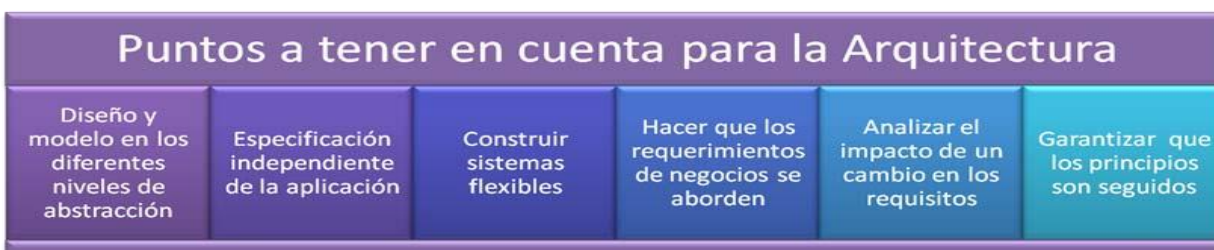


Figura 2 - Consideraciones para el diseño de una buena arquitectura SOA

Arquitectura Empresarial

Podemos pensar SOA dentro y fuera de un ámbito empresarial. De cualquier modo, para pensar SOA hacia fuera de una empresa, primero debemos pensarlo (o implementarlo) internamente dentro de la empresa. Para eso debemos ver que componentes son usados en las distintas formas de implementación.



Figure 3 – Arquitectura: amiga de los negocios

Componentes de SOA

El gráfico nos muestra algunos de los componentes que conforman una arquitectura SOA.

Nota: Este es un modelo de implementación, pero no necesariamente deben implementarse estos componentes en todos los escenarios. Es posible implementar conceptualmente SOA sin ellos.

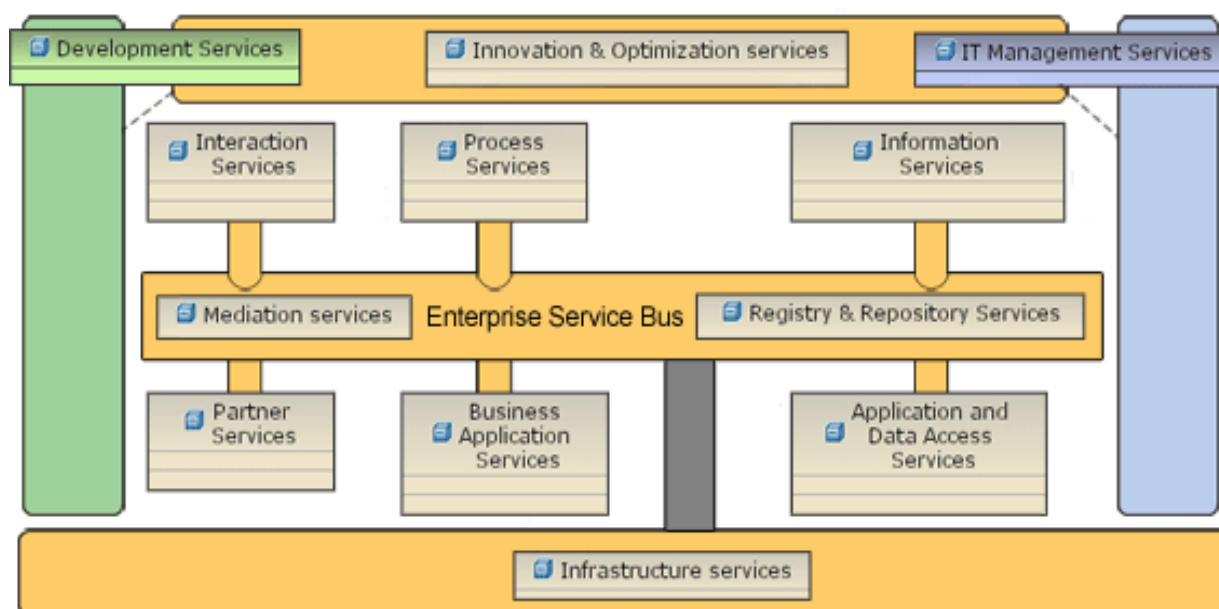


Figure 4 - Componentes de y servicios de una arquitectura SOA

- **ESB:** Enterprise Service Bus (por sus siglas en inglés), o bus empresarial de servicios, consta de una combinación de hardware y/o software que da conectividad a los servicios la empresa pública y consume internamente.
- **Publicación de Servicios:** algunas implementaciones cuentan con un catálogo de servicios que permite publicar los servicios disponibles para ser consumidos.
- **Mediación de Servicios:** esta funcionalidad permite, con lenguajes propios de la tecnología de BUS que se utiliza, desarrollar adaptadores que permitan conectar servicios pre-existentes que no fueron pensados para ser conectados. A esta característica se la llama “mediación”.
- **Distintos tipos de servicios:** los servicios a ser vinculados pueden ser de múltiples tipos (tal como se los enumera). En gráfico anterior muestra distintos tipos y niveles de servicios y su interconexión (pasando por servicios de negocio de desarrollo interno, servicios de infraestructura, acceso a datos, etc).

Beneficios de una arquitectura basada en SOA

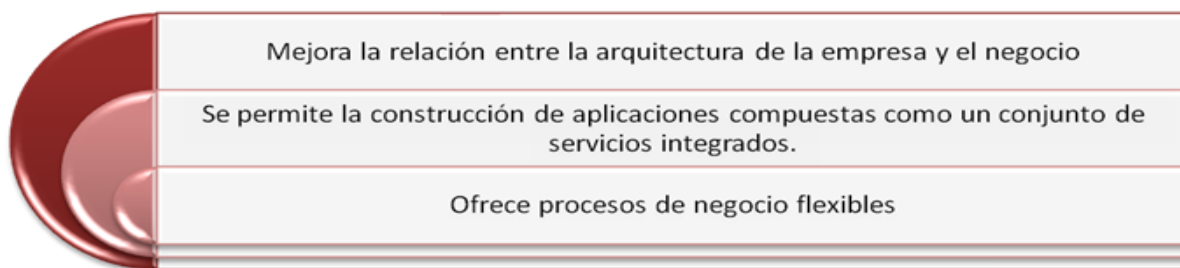


Figure 5 - Beneficios claves de una buena arquitectura basada en SOA

- **Mejor Relación Empresa Negocio:** determinados negocios conllevan a la utilización de distintos tipos de tecnología (móviles, o de pequeños dispositivos) que requieren una arquitectura SOA para poder integrarse con el resto de la empresa.
- **Integración:** SOA ha sido una de las primeras arquitecturas que permitieron el desarrollo e integración compuesto de soluciones.
- **Procesos de Negocio Flexibles:** esta arquitectura ha sido el punto de partida para modelos de negocio flexibles y/o de desarrollo rápidos como los que postula la propuesta de BPN (Business Process Model, o Modelado de Procesos de Negocio), que ofrece “orquestración de servicios” basados en SOA con herramientas específicas.

Capas Funcionales

El diagrama nos muestra dimensiones en las que se desarrolla una arquitectura SOA. En forma vertical, se ilustran los consumidores de servicios (Service Consumers), pasando por los procesos de negocio, proceso que “orquestran” o hacen coreografías (llaman a proceso en secuencia), pasando por servicios más atómicos, los servicios componentes, llegando a los sistemas componentes o proveedores de servicios.

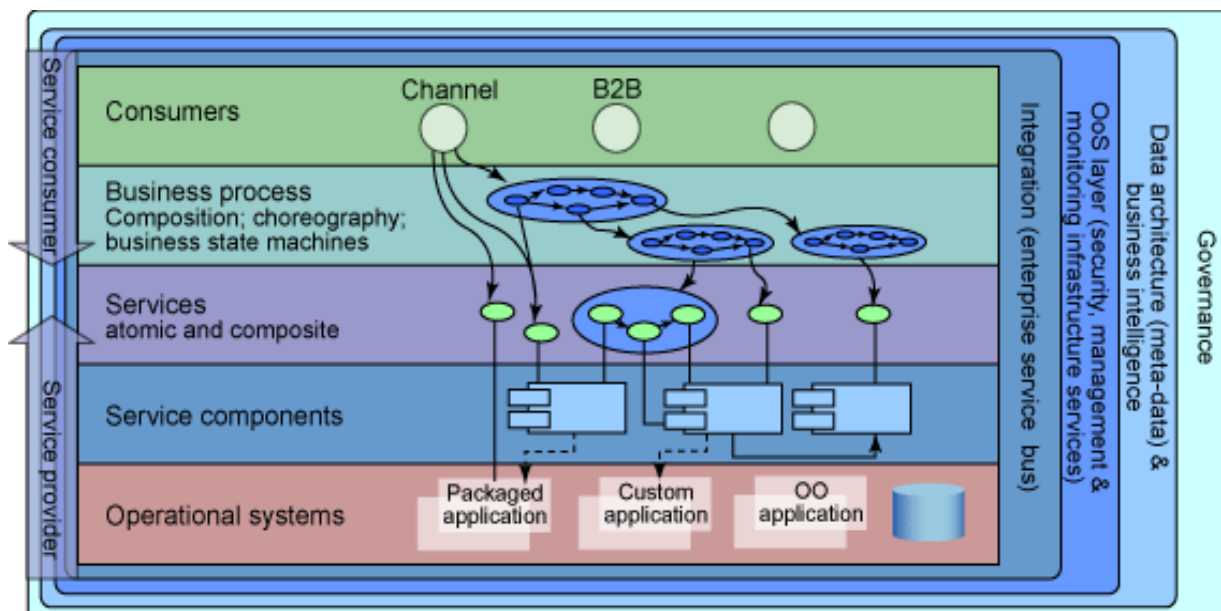


Figure 5 – Capas Funcionales y sus dimensiones de interacción.

En forma horizontal, se muestra la dimensión que va desde el gobierno de la arquitectura SOA (representando los acuerdos y definiciones de interacción), pasando por la gestión de la meta-data, la interacción de los servicios más básicos (de infraestructura) hasta el encuentro con el ESB, quien conecta con otros servicios.

Gobierno de una Arquitectura SOA

El gobierno de SOA es una extensión del gobierno de TI, se basa específicamente en la gestión de los servicios dentro del ciclo de vida de los artefactos SOA.



Figure 6 – Interacciones de gobierno para SOA en una organización.

La gobernabilidad de SOA dentro de una organización, aborda desafíos tales como:

- ⇒ ¿Qué roles y estructuras de la organización facilitan la identificación, el diseño y la distribución de servicios?
- ⇒ ¿Qué métricas apoyan a la inversión, el mantenimiento, la vitalidad, y el intercambio de servicios?
- ⇒ ¿Cuánto las empresas deciden invertir en la creación y mantenimiento de servicios?
- ⇒ ¿Cuál es la madurez de los servicios una empresa?
- ⇒ ¿Qué educación, capacitación o asesoramiento se necesita?

Ciclo de Vida SOA

SOA utiliza un ciclo de vida para asegurarse de que los procesos y políticas establecidos para cada servicio en el SOA Governance se cumplan.

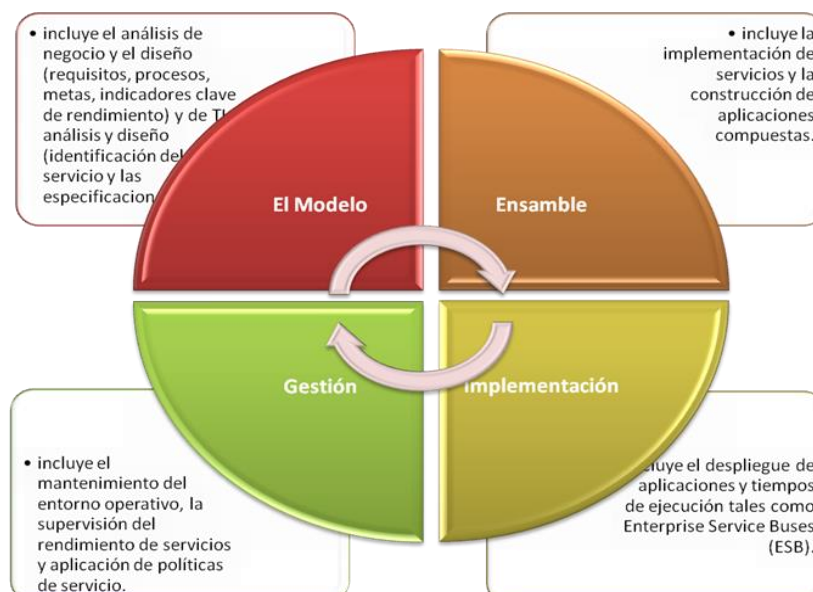


Figure 7– Ciclo de Vida en SOA.

Principios de SOA

El gráfico nos resume un orden de principios que se postulan para el buen uso de la arquitectura SOA. Sus conceptos son triviales, y están alineados con otras buenas prácticas del diseño y/o desarrollo de software.



Figure 8– Principios en SOA (Relación y secuencia).

Contrato de Servicio

El contrato de servicios implica definir con claridad entre las partes tres términos principales:

⇒ **Cabecera:**

- Nombre del Servicio
- Versión
- Dueño
- Responsabilidades (funcionales, técnicas, y de negocio)

⇒ **Requerimientos Funcionales:**

- Métodos de Provee
- Formas de llamarlos

⇒ **Requisitos No Funcionales:**

- Restricciones de Roles
- Información Transaccional
- Información de Proceso
- Disponibilidades



Figure 9 – Representación de la etapa del contrato para un servicio.

Protocolos usados en SOA

Remote Procedure Call - RPC

Se trata de un método simple de llamar subrutinas, alojado en la Capa de Aplicación. En donde una aplicación llama a una subrutina alojado en otro lugar.

Fue uno de los primeros métodos de intercambio de servicios. Está más orientado al intercambio de servicios dentro de un mismo servidor.

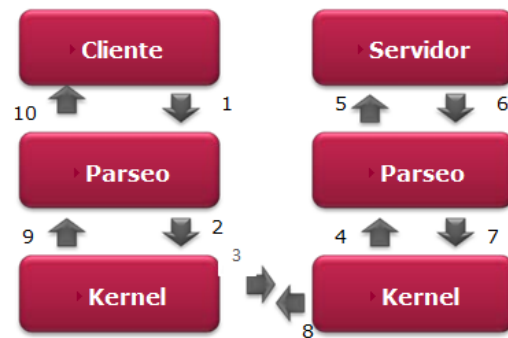


Figura 10 – Dinámica RPC

Distributed Component Object Model - DCOM

Se trata de una tecnología propietaria de Microsoft para comunicar objetos COM (interfaces de servicio) distribuidos, creado para reemplazar el uso de RPC en aplicaciones complejas.

Se basa en tener componentes distribuidos que se llaman y ejecutan subrutinas entre ellos.

Esta tecnología ya permitía intercambiar servicios en más de un servidor (de la misma tecnología).

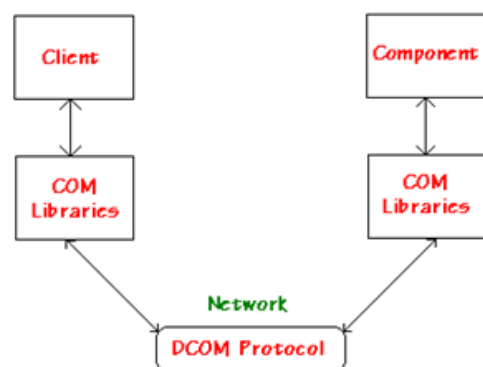


Figura 10 – Dinámica DCOM

Common Object Request Broker Architecture (CORBA)

Es una tecnología que, a contraposición de DCOM, tiene como meta definir un estándar de acceso y pasaje de información con el objetivo de que una serie de aplicaciones alojadas en distintos puestos actúen como un solo programa o set de servicios.

Funciona sobre la base de un agente (o broker) que es nexa con otros servicios. Fue postulado como solución multiplataforma.

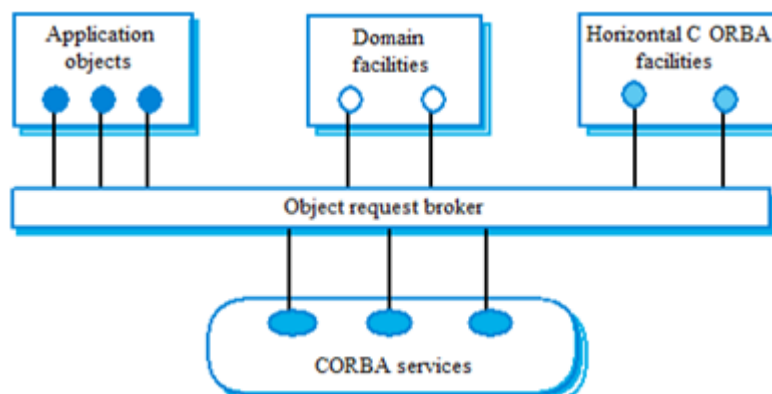


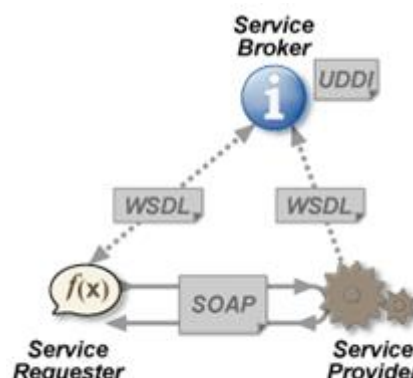
Figura 11 – Dinámica CORBA

Web Services

Es un método de comunicación de servicios organizado para una red interna o externa (intranet o extranet).

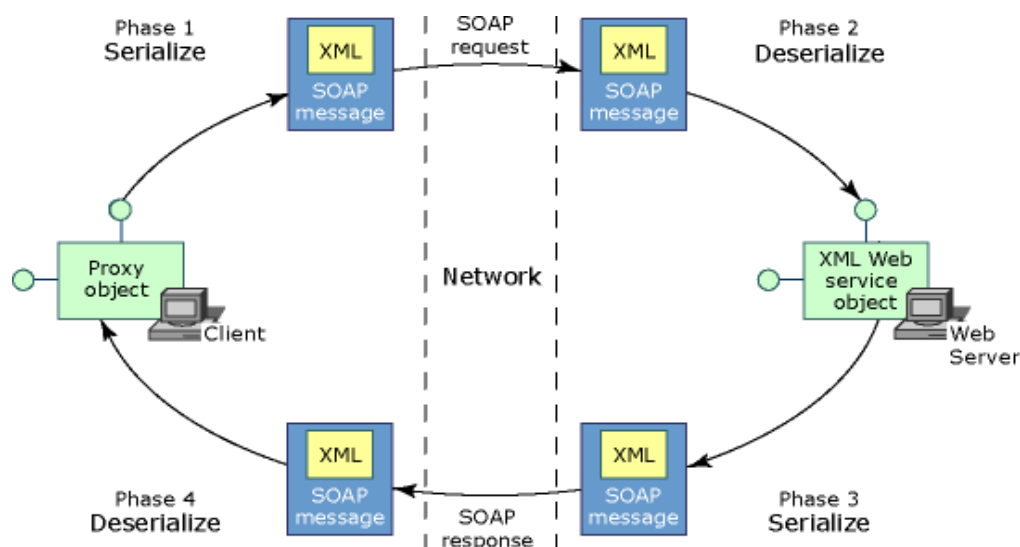
Actualmente es el método más difundido. Está usualmente compuesto por:

- Un Broker que publica información del servicio WSDL (Web Service Definition Language)
- El proveedor del servicio.
- El cliente que consume el servicio utilizando SOAP (Simple Object Access Protocol)



SOA Remote Procedure Call

Es un estilo de Web Service que se basa en la publicación llamadas de métodos o rutinas como interfaces. Ej. traerCliente([id]), grabarCliente(id), borrarCliente(id). Suele utilizar XML como medio de comunicación y envío de datos.



REST

A diferencia de SOAP RPC, REST es un estilo de Web Service que se basa en la publicación de recursos y provee un mecanismo de acceso y manipulación ya sea de colecciones o de elementos singulares.

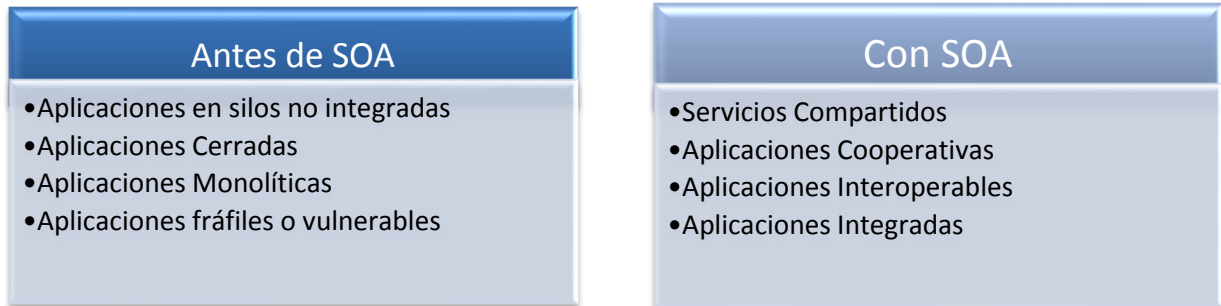
El mecanismo de acceso está compuesto por dos elementos: una URI (Uniform Resource Identifier) para la ubicación de uno o varios elementos dentro de un repositorio; un método de acceso que define la acción a realizar por el proveedor del servicio.

Este tipo de relacionamiento, es actualmente uno de los más difundidos y una tendencia para la integración entre servicios de distintas plataformas (y tecnologías).

Sin embargo, para la comunicación aún persisten los mecanismos anteriores cuando se trata de integrar aplicaciones dispositivos pequeños o de comunicación específica.

| Resource | GET | PUT | POST | DELETE |
|---|--|---|--|---|
| Collection URI, such as <code>http://example.com/resources/</code> | List the URIs and perhaps other details of the collection's members. | Replace the entire collection with another collection. | Create a new entry in the collection. The new entry's URL is assigned automatically and is usually returned by the operation. | Delete the entire collection. |
| Element URI, such as <code>http://example.com/resources/ef7d-xj36p</code> | Retrieve a representation of the addressed member of the collection, expressed in an appropriate Internet media type. | Update the addressed member of the collection. | Treat the addressed member as a collection in its own right and create a new entry in it. | Delete the addressed member of the collection. |

Beneficios de SOA



Claves para implementar SOA en la organización

En los siguientes links, referencias notas con claves para su implementación, no obstante un resumen de los principales puntos en cada caso:

⇒ [¿Por dónde empezar?](#)

⇒ [¿Cómo lo implemento?](#)

⇒ [¿Qué no debo hacer?](#)

Fuentes:

- http://en.wikipedia.org/wiki/Distributed_Component_Object_Model
- <http://en.wikipedia.org/wiki/CORBA>
- http://en.wikipedia.org/wiki/Distributed_Component_Object_Model
- http://en.wikipedia.org/wiki/Web_Services
- <http://en.wikipedia.org/wiki/SOAP>