

Introducción a los Servicios Web

Que es una API?

- Application Programming Interface
- Interacción entre dos piezas de software
- Lista de
 - Métodos para invocar (requests)
 - Sus parámetros
 - Valores de retorno
 - Formato de datos
- Lista de métodos que puedes invocar
- API Local
 - Interacción en la propia máquina
 - API del sistema operativo
 - Biblioteca de C
- API remota (servicio web)
 - Interacción en máquinas diferentes
 - Comunicación a través de una red
 - Ejemplo
 - API de twitter
- Proveedor (provider)
 - La entidad que ofrece sus servicios a través de la API
- Consumidor (consumer)
 - La entidad que solicita los servicios

Que es un servicio web?

- A Web service is a software service used to communicate between two devices on a network.
- More specifically, a Web service is a software application with a standardized way of providing interoperability between disparate applications. It does so over HTTP using technologies such as XML, SOAP, WSDL, and UDDI.
- A Web service is a software system designed to support interoperable machine-to-machine interaction over a network.
- It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.

Tecnologías

- REST
- Webhooks
 - User-defined callbacks made with HTTP
 - Mensajes automatizados resultado en un evento concreto
 - Mensajes específicos

- Para comunicación uno a uno entre dos apps
 - usado por ejemplo en CI/CD
- SOAP
 - SOAP es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. Este protocolo deriva de un protocolo creado por Dave Winer en 1998, llamado XML-RPC. SOAP fue creado por Microsoft, IBM y otros. Está actualmente bajo el auspicio de la W3C
- GraphQL
 - GraphQL es un lenguaje de consulta y manipulación de datos para APIs, y un entorno de ejecución para realizar consultas con datos existentes. GraphQL fue desarrollado internamente por Facebook en 2012 antes de ser liberado públicamente en 2015.
- Websockets
 - WebSocket es una tecnología que proporciona un canal de comunicación bidireccional y full-duplex sobre un único socket TCP. Está diseñada para ser implementada en navegadores y servidores web, pero puede utilizarse por cualquier aplicación cliente/servidor.
- MQTT
 - MQTT es un protocolo de red ligero, de publicación y suscripción, de máquina a máquina para cola de mensajes/servicio de cola de mensajes. Está diseñado para conexiones con ubicaciones remotas que tienen dispositivos con restricciones de recursos o ancho de banda de red

Formato de los datos

- XML
- JSON

Ventajas

- Interoperabilidad
 - independiente de plataforma
 - Independiente del lenguaje
- Estandarización
 - Las APIs son contratos que no deberían romperse
 - El proveedor promete no cambiar la API
 - Protocolos basados en texto, que hacen que sea más fácil acceder a su contenido y entender su funcionamiento
- Ubicuidad
 - Los servicios web se comunican utilizando HTTP, XML, JSON por lo tanto cualquier dispositivo que soporte estas tecnologías puede implementar o acceder a estos servicios
- Encapsulamiento (information hiding)
 - ninguna de las partes conoce la implementación de la otra
- Integración y composición de servicios
 - Reutilización de servicios
 - Creación de servicios más complejos

Ventajas de Aplicaciones Web

ventajas de Aplicaciones web

- Ahorran tiempo de instalación y despliegue
- No hay problemas de compatibilidad
- No hay necesidad de instalarlas y no ocupan espacio
- Actualizaciones inmediatas
- Consumo de recursos bajo
- Multiplataforma
- Portables
- Alta disponibilidad
- Difícil de atacar mediante virus
- Ideales para colaboración

Desventajas

- Soporte de transacciones
- Rendimiento
 - Por su uso de texto y XML
- Problemas derivados del uso de HTTP
- Interfaces inmutables
- Respuestas no garantizada

Desventaja (Aplicaciones web)

- Menor funcionalidad que las aplicaciones de escritorio
- Disponibilidad supeditada al proveedor de red
- Alta dependencia del servidor
- Limitación por el protocolo HTTP utilizado
- Cuello de botella en el ancho de banda para grandes cantidades de datos
- Necesidad de cifrar los datos

Los servicios web pueden ser

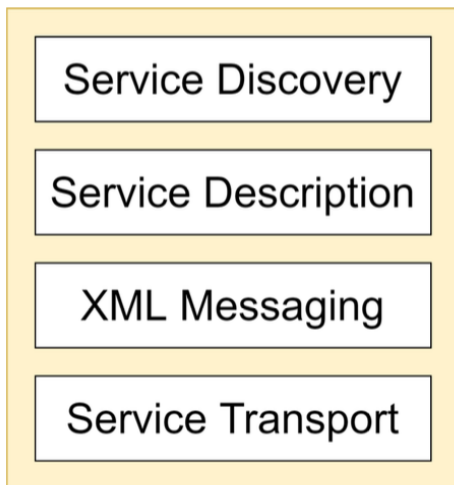
- Descritos
- Publicados
- Localizados
- Utilizados

A través de una red, generalmente internet

Arquitectura

- El stack del protocolo
 - Descubrimiento de servicios
 - UDDI
 - Descripción de servicios
 - WSDL
 - Mensajes XML
 - XML - RPC y SOAP
 - Capa de transporte

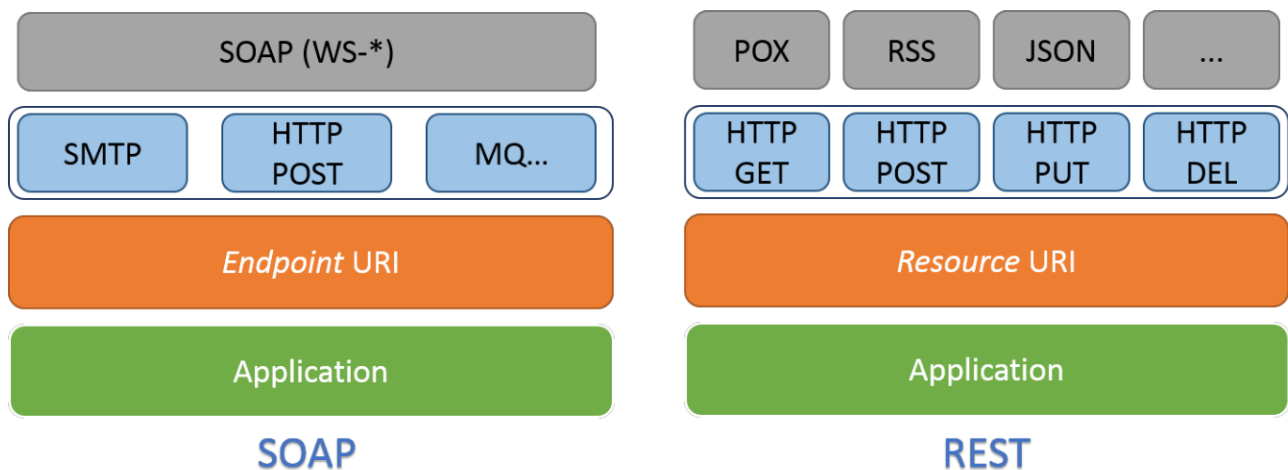
- HTTP, SMTP, FTP



Tipos de servicios web

- **SOAP**
 - Simple Object Access Protocol
 - Es un protocolo
 - Expone operaciones que representan lógica
 - Las operaciones se muestran con WSDL
- **REST**
 - REpresentational State Transfer
 - Es una arquitectura de SW
 - Se desarrollo paralelamente a HTTP/1.1
 - Cada URL representa un objeto

Diferencias SOAP vs REST



SOAP	REST
A XML-based message protocol	An architectural style protocol
Uses WSDL for communication between consumer and provider	Uses XML or JSON to send and receive data
Invokes services by calling RPC method	Simply calls services via URL path
Does not return human readable result	Result is readable, which is just plain XML or JSON
Transfer is over HTTP or other protocols such as SMTP, FTP, etc.	Transfer is over HTTP only
JavaScript can call SOAP, but it is difficult to implement	Easy to call from JavaScript
Performance is not great compared to REST	Performance is much better compared to SOAP. Less CPU intensive, leaner code, etc.

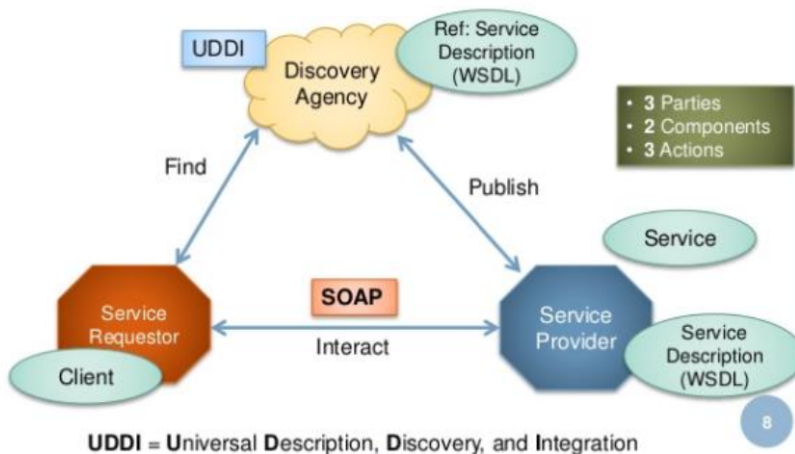
SOAP

- Expone operaciones que representan lógica
- Codifica todo en XML con adjuntos
- Soporta operaciones con estado (Stateful)
- Muy estandarizado
- Define su propia seguridad
- No puede usar REST

REST

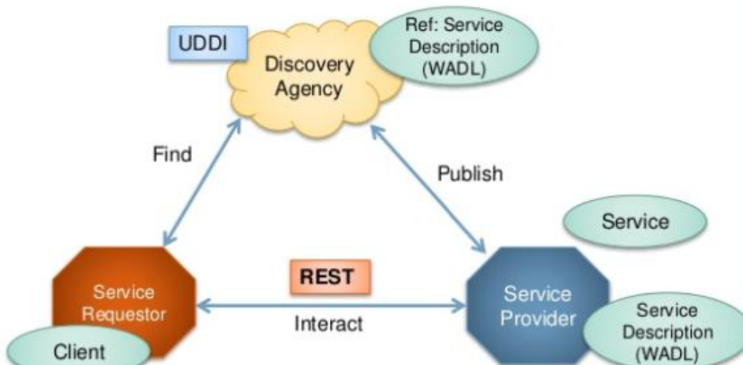
- Expone recursos que representan datos
- Utiliza varios formatos pero basa todo en texto plano
- Diseñado para operaciones sin estado (Stateless)
- Evita estándares
- Hereda la seguridad del transporte
- Puede usar SOAP

WEB SERVICE ARCHITECTURE BY W3C (SOAP-BASED)



UDDI = Universal Description, Discovery, and Integration

WEB SERVICE ARCHITECTURE (REST-BASED)



REST

- REST APIs should be independent of the protocol
 - Stateless
 - Cacheable
 - Uniform Interface
 - Idempotent
- RESTful
 - Orientadas al recurso
 - La función va en el método HTTP
- RPC-Style
 - Orientadas al proceso
 - Toda la información va en el paquete
 - El encapsulamiento no cambia
- REST -RPC hybrid
 - mix

UDDI

- Universal Description, Discovery and Integration
- Basado en XML
- Pretendía ser un repositorio central de servicios web
- Impulsado por la industria pero prácticamente abandonado en su idea original
- Se sigue utilizando pero de forma privada

WSDL

- Web Services Description Language
- Basado en XML
- Describe la interfaz pública de los servicios web
- Un cliente puede conectarse a un servicio web para leer el WSDL y determinar qué funciones están disponibles en el servidor

WADL

- Web Application Description Language
- Basado en XML
- Describe la interfaz pública de los servicios web
- WSDL en principio no estaba diseñado para REST al no admitir distintos métodos HTTP que POST, por ello surgió WADL
- Menos flexible que WSDL pero más ligero y fácil de usar
- Sin embargo para servicios REST no es necesaria una descripción por lo que su uso no está muy extendido

Aplicación web vs servicios web

Aplicación web

- Diseñadas para humanos
- Tienen interfaces
- Human to Machine
- Tienen usuario
- Información y presentación (HTML)

Servicios web

- Diseñados para máquinas
- Exponen APIs
- Machine to Machine
- Tienen aplicaciones
- Información sin presentación (XML)