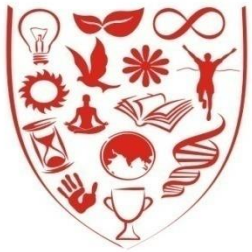
Cloud computing

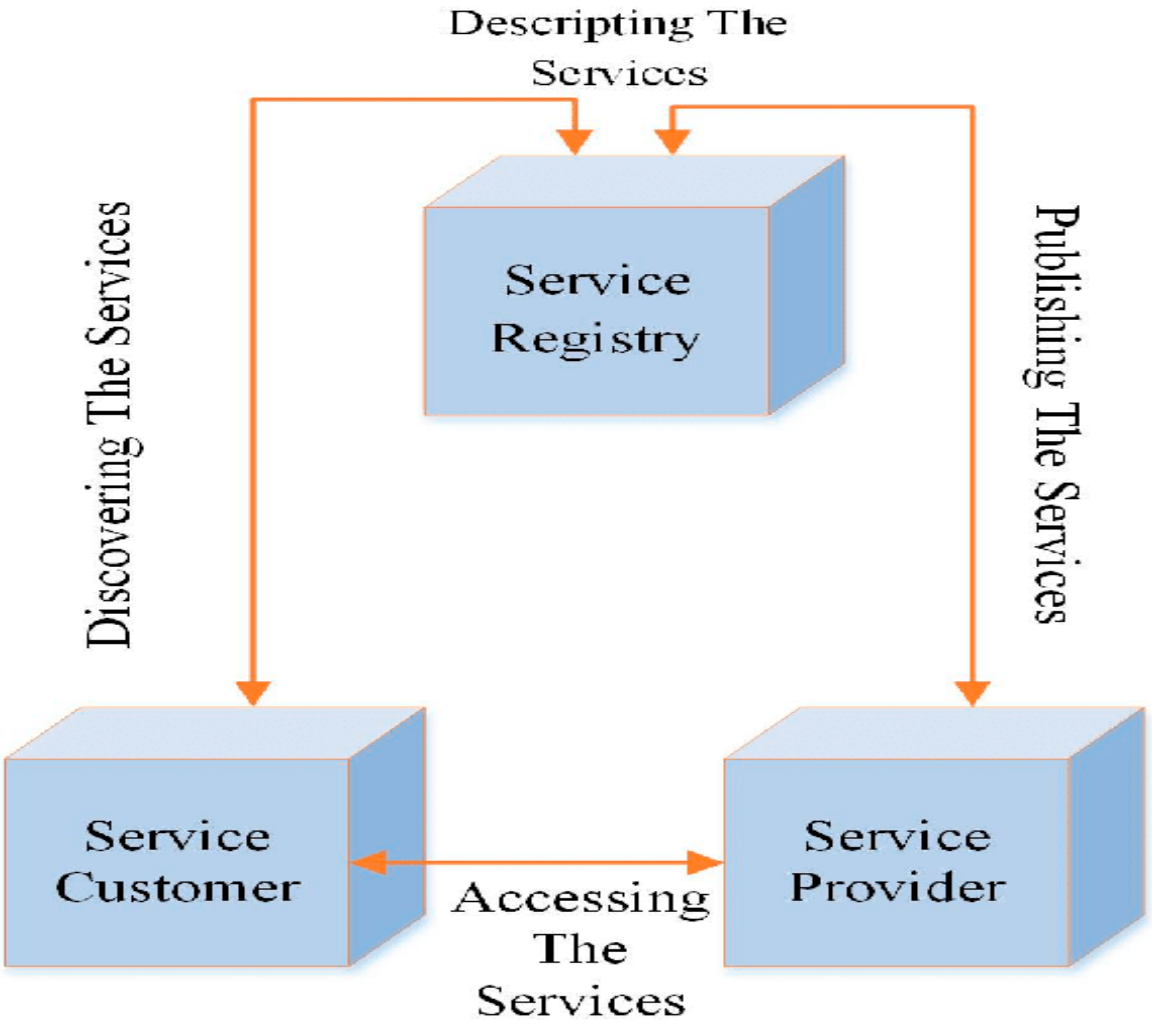


JECRC™
UNIVERSITY
BUILD YOUR WORLD

CONTENTS

- SOA & Cloud
- What is difference?
- What is common?
- SOAP vs REST

SOA


Descripting The Services
Service Registry
Discovering The Services
Publishing The Services
Service Customer
Service Provider
Accessing The Services

What is SOA?

- Service Oriented Architecture (SOA): is a flexible set of design principles used during the phases of systems development and integration.

- A deployed SOA-based architecture will provide a loosely-integrated suite of services that can be used within multiple business domains.

What is common?

- First, both emphasize the service concept. By definition, a service is performance of work by one for another.
- Both Cloud and SOA delegate work to other parts of the system, either by the service provider or other business components. With that delegation, people can use the services without worrying about the implementation details and scalability.

- Most important, services in CAA and SOA can be shared by multiple applications and users, thus optimizing resource utilization.

What is common?

- Second, both promote loose coupling. Each architecture demands minimum dependencies among different parts of the system.

- As a result, any single change on one part of the system has limited impact on the overall system.

Difference between SOA & Cloud

- Horizontal versus vertical services.
- The services in SOA mainly focus on business. Each service may represent one aspect of the business. Combined together, these services consist of a business application or solution.
- In this sense, the services are horizontal.

- The services in cloud are mainly layered according to typical software stacks. The lower services support the upper services to deliver applications.
- Therefore, these are called vertical services.

Difference between SOA & Cloud

- Application vs. infrastructure
- SOA is for application architecture.

- The dividing of different components is based on their roles in the SOA applications. More often than not, you start with a business problem and then abstract out the services.

- These services can be re-used by other applications in the future.

Difference between SOA & Cloud

- Application vs. infrastructure
- Cloud Computing is for IT delivery.
- The dividing of different services is based on their roles in a software stack that is mostly well defined.
- You don't need a problem before defining the cloud services.
- The services can be easily re-used by all applications.

SOAP & REST

- REST stands for **Representational State Transfer** where as SOAP stands for **Simple Object Access Protocol**.
- SOAP defines its own **security** where as REST inherits security from underlying transport.
- SOAP does not support error handling but REST has built-in **error handling**.
- REST is lightweight and does not require **XML parsing**.

SOAP & REST

- REST can be consumed by any client, even a web browser with Ajax and Javascript.

- REST consumes **less bandwidth**, it does not require a SOAP header for every message.

SOAP Vs REST

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

REST

- **Successful request:**
- { "success": true,
- "payload": {
- /* Application-specific data would go here. */
- }
- }

REST

- **Failed request:**
- { "success": false,
- "payload": {
- /* Application-specific data would go here. */
- },
- "error": {
- "code": 123,
- "message": "An error occurred!"
- }
- }

Pros and Cons

- Pros
- Reusability
- Maintainability
- Scalability
- Dynamic Provisioning

- Cons
- Complexity
- Increased Overhead

THANK YOU!!!