SOA = Service oriented architecture --- > 1. CORBA (common object request broker architecture ) 2. Jini (Or Appache River) 3. Web Services (mostly used in SOA)

What is webservice ?

A web services is a software system designed to support interoperable machine to machine interaction over a network . or making application interact with each other on the web through HTTP .

Difference between Web application and webservice

if we have User Interface for providing two numbers and then getting the result, it should be called a web application. But if we have an API exposed to receive two numbers and return result over http , then it should be called a web service.  
At low level, both Web application and web service are kind of same thing. But the main point is that web services are for machine/program to machine/program communication whereas Web application is for Users.

Types of webservice :

SOAP : Simple object access protocol , it is heavily relied on XML and Schema

RESTful : representational state Transfer

**Restful webservice :**

Restful webservice can be developed and consumed in java using **(1). JAX-RS (2). Spring Rest** .

**JAX-RS Implementation by :**

1. Weblogic Application server implementation 2. Apache CXF 3. Jersy 4 .Resteasy 5.WebSphere Application server implementation

Spring Rest:

Spring MVC :

Flow :

Note : The Rest controller are different from MVC controller because REST controller’s method return results which can be mapped to representation rather than a view . for this , spring 3.0 introduced @RespoonseBody annotation , the method annotated with @ResponseBody tells that the result of execution need not to be mapped with view .

@ResponseBody annotation automatically converts the response to a JSON string litteral by applying serilization on the return value of the method to an http response.

@RequestBody annnotaiton helps map our HTTP request body to java DTO .

In spring 4, the @RestController annotation was introduced . When we used this @RestController class bounds all the values returned by controller method to the response body . Here @RestController is combination of Controller + ResponseBody . When we give @Restcontroller , no need to provide @ResponseBody in method level

NOTE: we can use PostMapping or @GetMapping in method level if there is @RequestMapping annotation is given in class level otherwise does not work .

**Difference between Spring MVC and Spring Rest:** Spring MVC response is a view/page by default , in spring REST data is returned directly back to the client. Eg.

@PostMapping

Public String createCustomer(@RequestBody CutomerDTO customerDTO){

}

Here in this example , Incomming HTTP request body is deserialized to CustomerDTO . and BY default MIME type is json .

If we don’t give @RequestBody annotation then it customerDto will give null .

**Dependencies in pom.xml :**

1. Spring-boot-starter-web : it is used to build a spring MVC application including the support fro REST . It also ensure application deployed on tomcat embedded server .