Webservices:

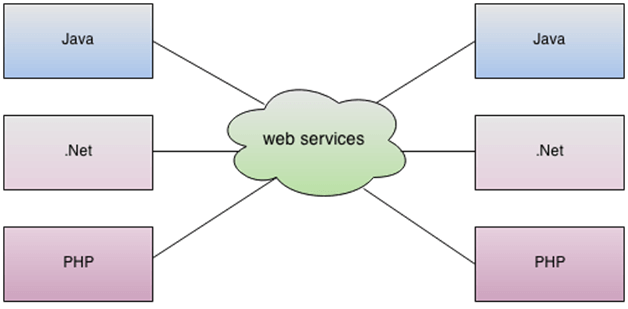
Web service is a technology to communicate one programming language with another. For example, java programming language can interact with PHP and .Net by using web services.

What is Web Service

A **Web Service** is can be defined by following ways:

* It is a client-server application or application component for communication.
* The method of communication between two devices over the network.
* It is a software system for the interoperable machine to machine communication.
* It is a collection of standards or protocols for exchanging information between two devices or application.

Let's understand it by the figure given below:



As you can see in the figure, Java, .net, and PHP applications can communicate with other applications through web service over the network. For example, the Java application can interact with Java, .Net, and PHP applications. So web service is a language independent way of communication.

Types of webservices:

1. SOAP web services.
2. RESTful web services.

# RESTful Web Services

REST stands for REpresentational State Transfer.

REST is an architectural style not a protocol.

## **Advantages of RESTful Web Services**

**Fast**: RESTful Web Services are fast because there is no strict specification like SOAP. It consumes less bandwidth and resource.

**Language and Platform independent**: RESTful web services can be written in any programming language and executed in any platform.

2) **JAX-RS**: for RESTful web services. There are mainly 2 implementation currently in use for creating JAX-RS application: Jersey and RESTeasy.

|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **URI** | http://localhost:8080/UserManagement/rest/UserService/users |
| **Operation** | Get list of users |
| **Operation Type** | Read Only |

|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **URI** | http://localhost:8080/UserManagement/rest/UserService/users/1 |
| **Operation** | Get user of Id 1 |
| **Operation Type** | Read Only |

|  |  |
| --- | --- |
| **HTTP Method** | POST |
| **URI** | http://localhost:8080/UserManagement/rest/UserService/users/2 |
| **Operation** | Insert user with Id 2 |
| **Operation Type** | Non-Idempotent |

|  |  |
| --- | --- |
| **HTTP Method** | PUT |
| **URI** | http://localhost:8080/UserManagement/rest/UserService/users/2 |
| **Operation** | Update User with Id 2 |
| **Operation Type** | N/A |

|  |  |
| --- | --- |
| **HTTP Method** | DELETE |
| **URI** | http://localhost:8080/UserManagement/rest/UserService/users/1 |
| **Operation** | Delete User with Id 1 |
| **Operation Type** | Idempotent |

|  |  |
| --- | --- |
| **HTTP Method** | OPTIONS |
| **URI** | http://localhost:8080/UserManagement/rest/UserService/users |
| **Operation** | List the supported operations in web service |
| **Operation Type** | Read Only |

|  |  |
| --- | --- |
| **HTTP Method** | HEAD |
| **URI** | http://localhost:8080/UserManagement/rest/UserService/users |
| **Operation** | Returns only HTTP Header, no Body |
| **Operation Type** | Read Only |

1. **POST** – This would be used to create a new employee using the RESTful web service
2. **GET** – This would be used to get a list of all employee using the RESTful web service
3. **PUT** – This would be used to update all employee using the RESTful web service
4. **DELETE** – This would be used to delete all employee using the RESTful services