WEB API with .NET 5+

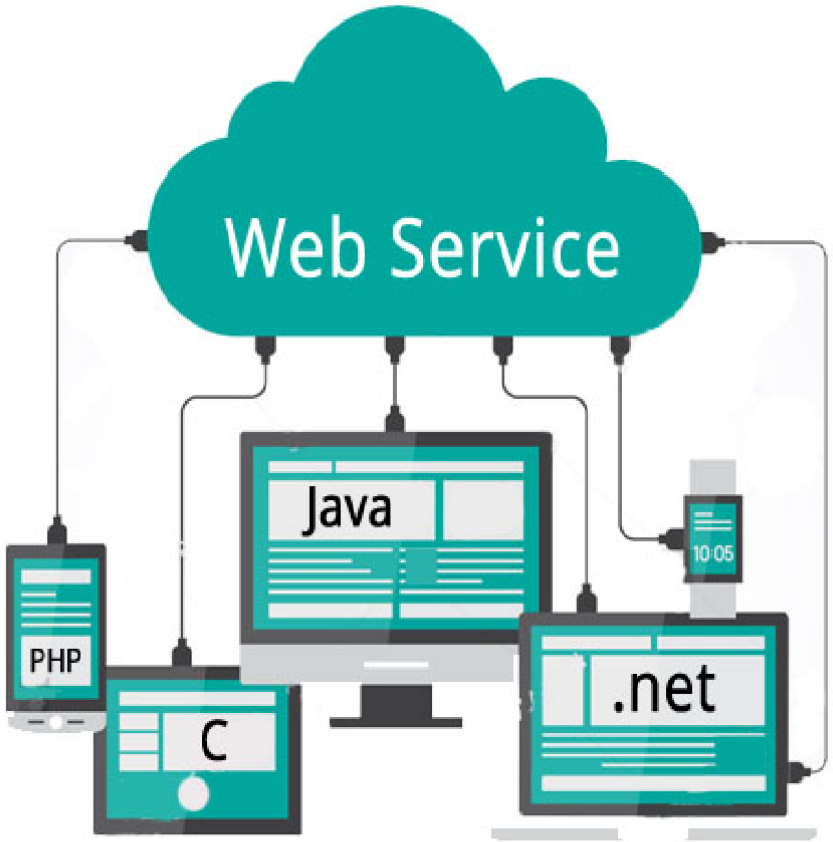
Arctech Info Private Limited

Topics

- What is a Web Services
- Web Service vs Web Sites
- Windows Communication Foundation (WCF)
- Web API

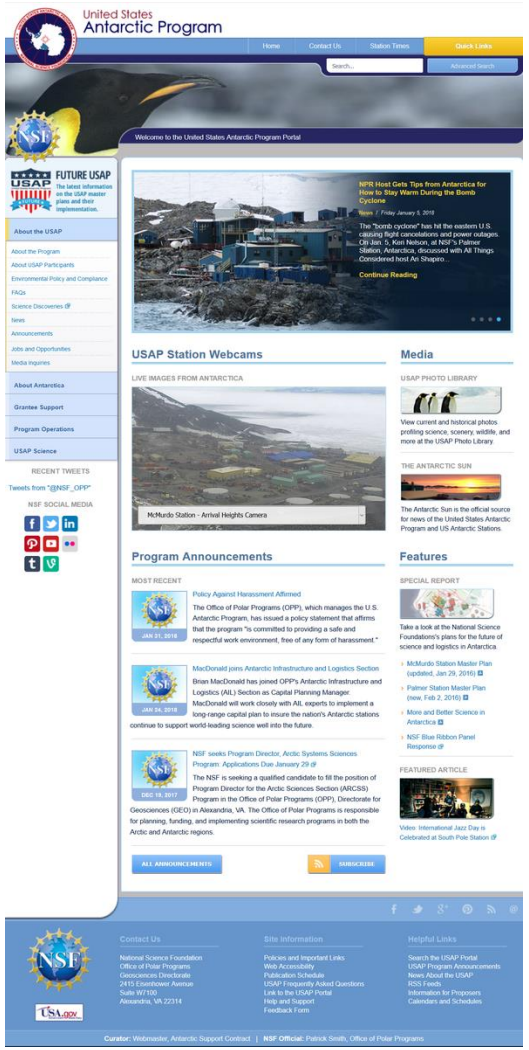
What is a Web Service

- A standardized way for developing interoperable applications
- Allows invoking a method from another application
- These applications can be on the same computer or on different computers
- Most common usage is a client device connecting to a web service over the internet



- Consider a weather station which provides a Web Service interface for accessing weather updates
- Clients can access this webservice to get the weather updates.
  - mobile apps
  - websites
  - IOT devices
- Any Client based on .NET Framework can access a Web Service
- Others that understand SOAP can also access a Web Service

Web Site or Web Service



Web Site

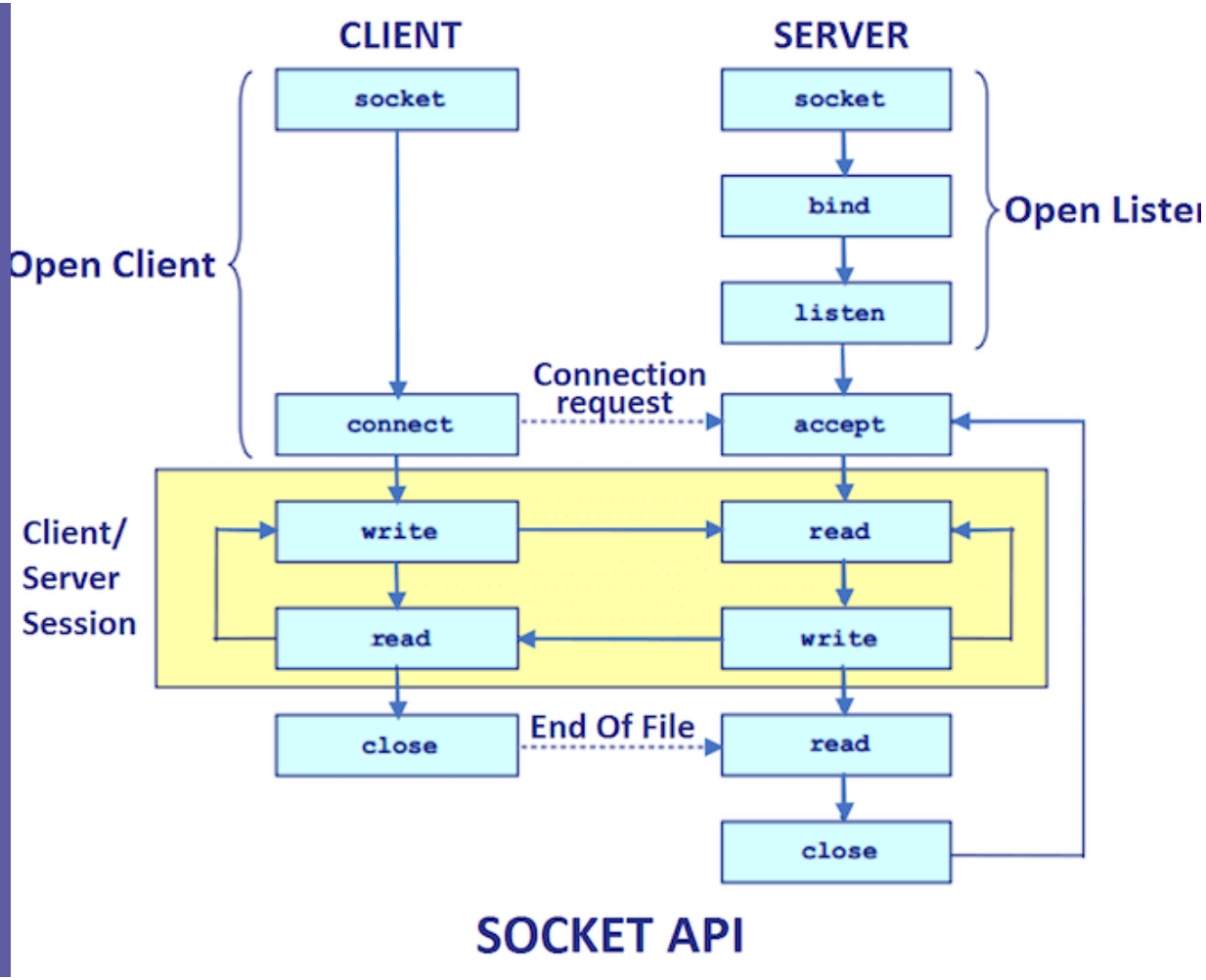
- A website has a user interface or GUI.
- Websites are meant for use by humans.
- Websites are cross-platform as they require tweaking to operate on different browsers, operating systems, etc.
- E.g., w3schools.com is a website to educate web developers

Web Service

- A web service doesn't have a user interface.
- Web services are meant for other applications to be interacted with over the internet.
- Web services are platform independent as they use open protocols
- E.g., Google maps API is a web service that can be used by websites to display Maps by passing coordinates to it.

Before Web Services

- Computers applications always needed to communicate with other computers or applications.
- For example to get stock updates or weather applications.
- Before Web Services standardized inter application communication
  - Programmers had to develop their own protocols and techniques
  - Use low level Socket programming to connect to the server and send/receive data.



ASP.NET Web Services uses SOAP

- Simple Object Access Protocol
- Application communication protocol
- Format for sending and receiving messages
- Platform & Language Independent
- Based on XML
- Originally developed by Microsoft

```
POST /InStock HTTP/1.1
Host: www.example.org
Content-Type: application/soap+xml; charset=utf-8
Content-Length: nnn

<?xml version="1.0"?>

<soap:Envelope
xmlns:soap="http://www.w3.org/2003/05/soap-envelope/"
soap:encodingStyle="http://www.w3.org/2003/05/soap-encoding">

<soap:Body xmlns:m="http://www.example.org/stock">
  <m:GetStockPrice>
    <m:StockName>IBM</m:StockName>
  </m:GetStockPrice>
</soap:Body>

</soap:Envelope>
```

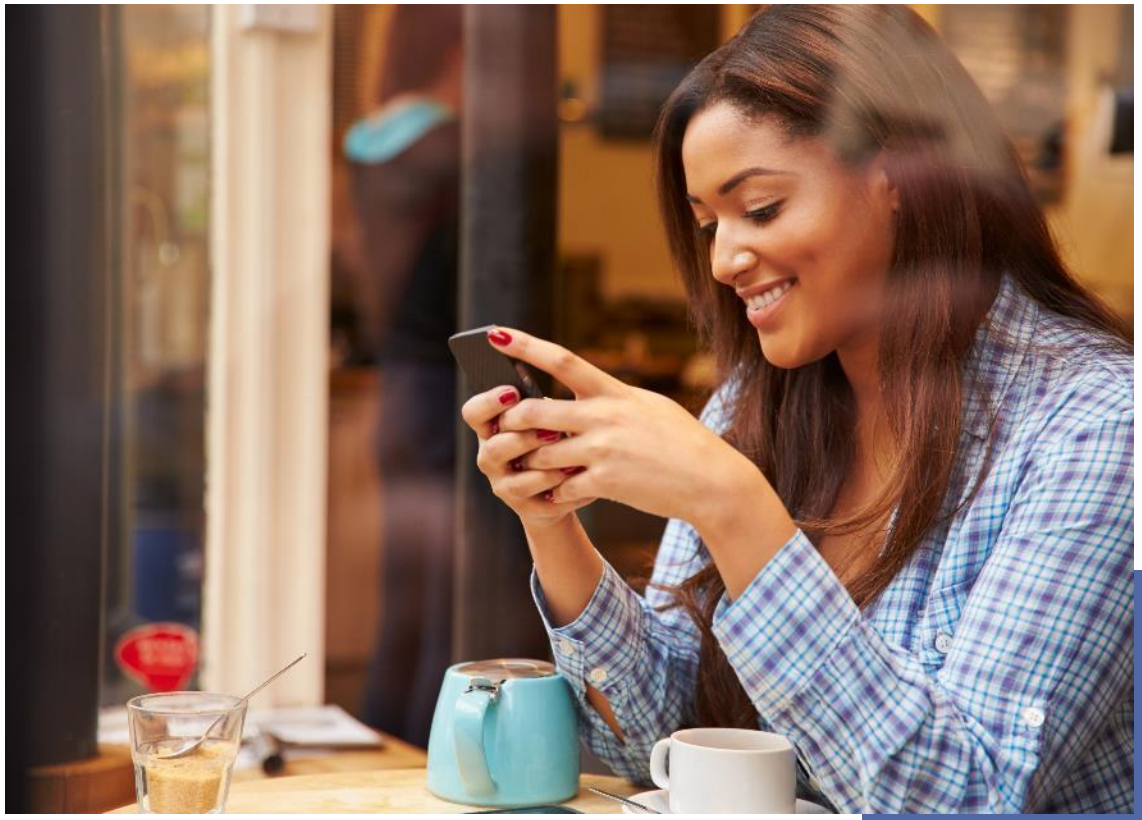
Create an ASP.NET Web Service (ASMX)

- ASMX provides the ability to build Web Services using SOAP
- Consumers of ASMX do not need to know platform, model or programming language used to build the web service
- Consumers only need to understand how to send and receive SOAP messages.

- Consume an existing SOAP WebService
- Build a SOAP WebService

Security - Authentication

Security - Confidentiality



Security - Confidentiality

SOAP runs primarily on top of HTTP POST

HTTP has support for Secure Sockets Layer (SSL)

So SOAP communication can be encrypted via SSL

SSL is a proven technology and widely deployed

Security - Authentication

Windows-Based Security

Basic Windows Authentication

Digest Windows Authentication

Integrated Windows Authentication

Custom Authentication

Log In Method

SOAP Header

SOAP Extensions

SOAP Authentication Request

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetBankList
      xmlns="http://arctechinfo.com/webservies/worldline-
      training">
      <offset>int</offset>
      <rowCount>int</rowCount>
    </GetBankList>
  </soap:Body>
</soap:Envelope>
```

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <UserAuthSoapHeader
      xmlns="http://arctechinfo.com/webservies/worldline-
      training">
      <Username>string</Username>
      <Password>string</Password>
    </UserAuthSoapHeader>
  </soap:Header>
  <soap:Body>
  ...
  </soap:Body>
</soap:Envelope>
```

Demo Site (Free Access)
https://bankdetailsdemo.azurewebsites.net/WebServicesDev/BankData.asmx

Production Site (Paid Access)
https://bankdetailsdemo.azurewebsites.net/WebServices/BankData.asmx

Example: Adding authentication in the web service

```
0 references
public class UserAuthSoapHeader : SoapHeader
{
    1 reference
    public string Username { get; set; }
    1 reference
    public string Password { get; set; }

    0 references
    public bool IsValid()
    {
        // This is hardcoded for testing
        // In real application, this would be checked in database
        // or using any authentication libraries
        return Username == "raman" && Password == "gujral@123";
    }
}
```

```
6 references
public UserAuthSoapHeader UserAuthSoapHeader { get; set; }

[WebMethod]
[SoapHeader(nameof(UserAuthSoapHeader))]
0 references
public PopulationInfo GetPopulationInfo(string city)
{
    AuthenticateUser();

    var random = new Random();
    return new PopulationInfo
    {
        Males = random.Next(999999),
        Females = random.Next(999999),
        Others = random.Next(999999),
        Below18 = random.Next(999999),
        Above65 = random.Next(999999),
    };
}

2 references
private void AuthenticateUser()
{
    if (UserAuthSoapHeader == null)
        throw new SoapHeaderException(
            "Soap Header [UserAuthSoapHeader] is missing!!"
            , new XmlQualifiedName(nameof(UserAuthSoapHeader)));

    if (!UserAuthSoapHeader.IsValid())
        throw new SoapHeaderException(
            "Username and/or password is invalid!!"
            , new XmlQualifiedName(nameof(UserAuthSoapHeader)));
}
```

Example: Consuming a web service with authentication

- The commented section shows consumption of a web service which does not require authentication
- Note the differences
  - The authenticated web service method has an additional parameter
  - The Web Method return value is different when the SOAP header is present
  - See
    - response.Body.GetTotalPopulationResult
    - Versus
    - response.GetTotalPopulationResult

```
private readonly UserAuthSoapHeader userAuth = new UserAuthSoapHeader
{
    Username = "raman",
    Password = "gujral@123"
};

1 reference
public async Task<int> GetTotalPopulationAsync(string city)
{
    var client = new PopulationLiveSoapClient(basicHttpBinding, endpointAddress);

    //var soapResponse = await client.GetTotalPopulationAsync(city);
    //return soapResponse.Body.GetTotalPopulationResult;

    var soapResponse = await client.GetTotalPopulationAsync(userAuth, city);
    return soapResponse.GetTotalPopulationResult;
}
```

SOAP vs REST

- SOAP - Simple Object Access Protocol
- SOAP is a protocol which
  - Uses HTTP protocol
  - Includes a WSDL file
  - Permits XML data format only
  - Requires more bandwidth
- You can build SOAP based Web Service using ASP.NET (asmx)

- REST - Representational State Transfer
- REST is an Architectural style
  - Usually uses HTTP
  - Can use any data format like plain text, xml, json, etc.
  - Requires less bandwith
- You can build REST Services using The ASP.NET Web API

SOAP vs REST – Data Packet Examples with HTTP headers

```
POST /WebServices/Population.asmx HTTP/1.1
Host: census.gov.in
Content-Type: text/xml
Content-Length: 418

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope
xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">

  <soap:Body>

    <GetCityPopulation
xmlns="http://census.gov.in/Population">

      <city>Mumbai</city>

      <state>Maharashtra</state>

    </GetCityPopulation>

  </soap:Body>

</soap:Envelope>
```

- GET example

```
GET /WebApi/Population?city=Mumbai&state=Maharashtra HTTP/1.1
Host: census.gov.in
```

- POST example

```
POST /WebApi/Population HTTP/1.1
Host: census.gov.in
Content-Type: application/json
Content-Length: 54

{
    "city": "Mumbai",
    "state": "Maharastra“
}
```

Thank you

Avinash Tauro

Arctech Info Private Limited