

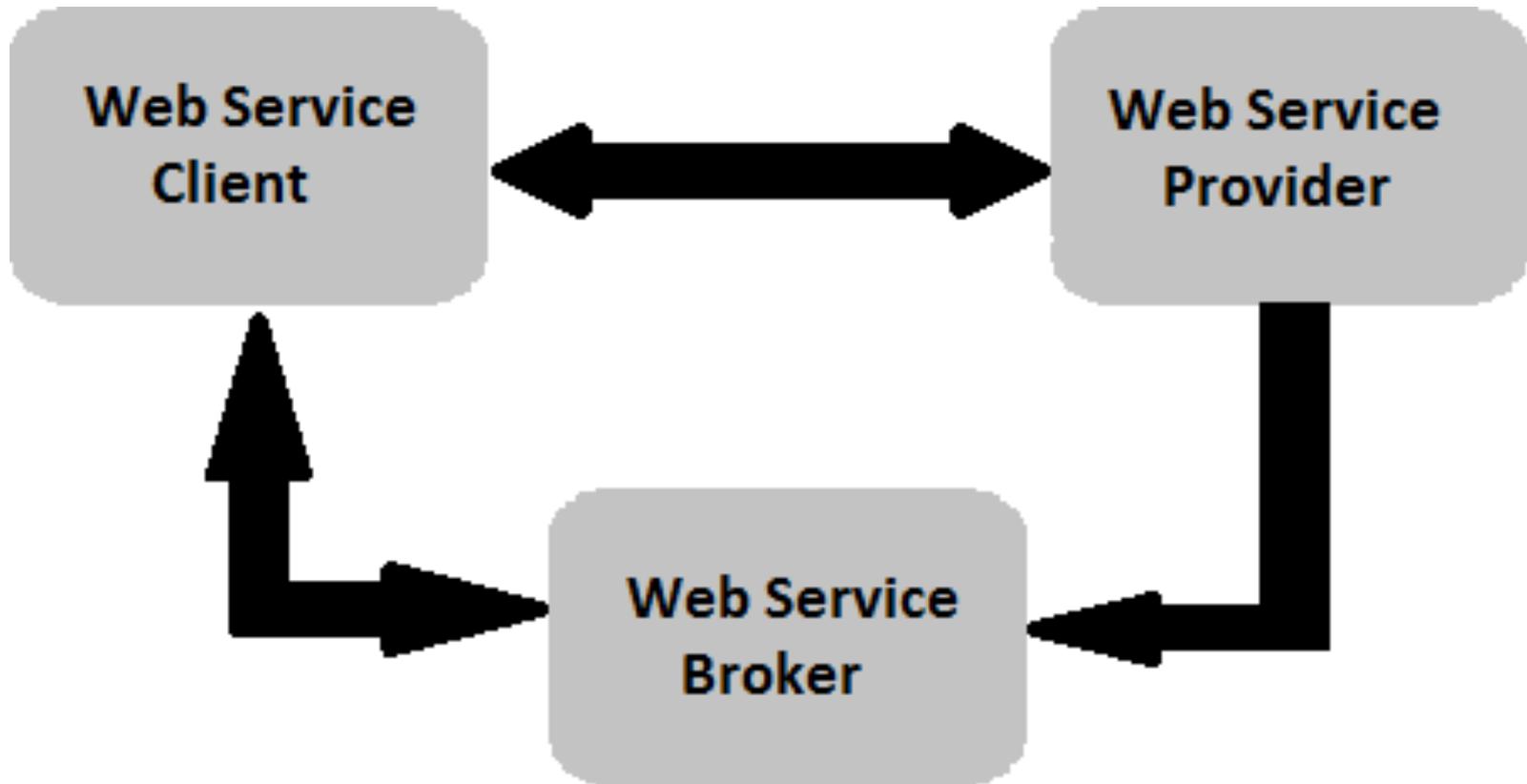
# Разработка XML Веб-сервисов

Залятов А.Э.

# Определение Web-сервиса

- ▶ Web-сервисы – это модульные бизнес-приложения, имеющие стандартные интерфейсы для работы через Интернет, другими словами, являются слабосвязанными компонентами ПО, доступными для использования через стандартные Интернет-протоколы.

# Схема взаимодействия Web-сервисов



# Общие сведения об XML-веб-службах

- ▶ XML-веб-служба — это программируемое средство предоставления определенного элемента или функциональных возможностей, например логики приложения, доступное любому количеству потенциально различных систем с помощью широко применяемых стандартов Интернета, таких как XML и HTTP.

# Общие сведения об XML-веб-службах

- ▶ XML-веб-службу можно использовать для одного приложения на компьютере или предоставить к ней доступ через Интернет любому числу приложений.

# Общие сведения об XML-веб-службах

- ▶ XML-веб-службы устраняют необходимость работы над переносимостью кода, обеспечивая эффективное взаимодействие между системами и данными.
- ▶ Механизм передачи данных XML-веб-службами основан на обмене сообщениями на базе XML, что позволяет обойти различия между системами, в которых используются несовместимые модели компонентов, операционные системы и языки программирования.

# Общие сведения об XML-веб-службах

- ▶ Одной из ключевых характеристик XML-веб-службы является высокая степень абстракции между реализацией службы и ее использованием.

# Soap

- ▶ Soap – это протокол, основанный на обмене XML-документами.

# WSDL

- ▶ WSDL – это стандартный XML–документ, описывающий фундаментальные свойства Web службы.

# Создание веб-сервиса

- ▶ `<@% WebService class="MathServiceJit" language="C#" %>`
- ▶ `using System.Web.Services;`
- ▶ `public class MathServiceJit {`
  - `[WebMethod]`
  - `public double Add(double x, double y)`
    - `{ return x + y; }`
  - `[WebMethod]`
  - `public double Subtract(double x, double y)`
    - `{ return x - y; }`
- `}`

# Создание веб-сервиса

- ▶ `<@% WebService class="MathServiceJit" language="C#" %>`
- ▶ В описании `WebService` должен быть также указан и язык программирования, чтобы обработчик мог выбрать в среде выполнения правильный компилятор.

# Создание веб-сервиса

```
[WebMethod]
```

```
public double Add(double x, double y)
```

```
{ return x + y; }
```

Оболочка `WebMethods` занимается преобразованиями SOAP сообщений в методы класса `.NET`. Это делается, прежде всего, путем аннотирования ваших методов атрибутом `[WebMethod]`.

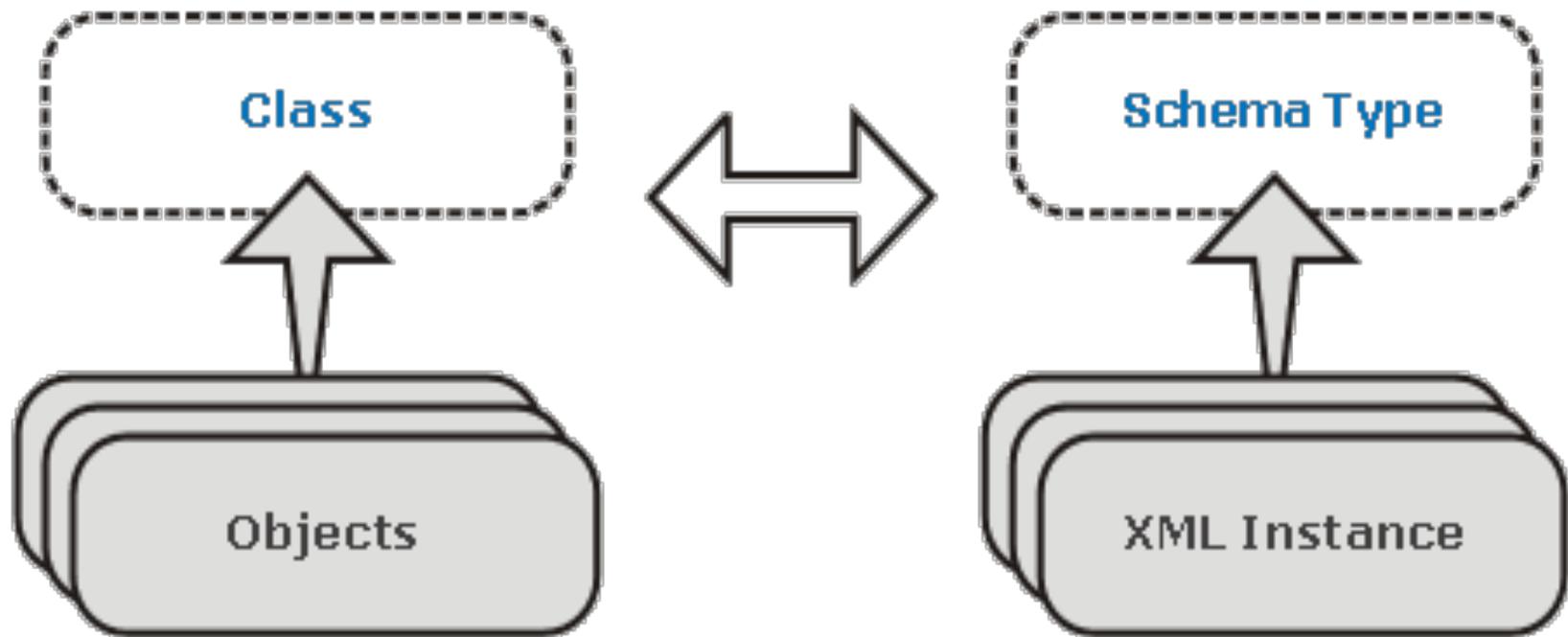
# Диспечеризация сообщений

- ▶ POST /math/math.asmx HTTP/1.1
- ▶ Host: localhost
- ▶ Content-Type: text/xml; charset=utf-8
- ▶ Content-Length: length
- ▶ SOAPAction: "http://tempuri.org/Add"
  
- ▶ <soap:Envelope
- ▶   xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
- ▶ >
- ▶   <soap:Body>
- ▶     <Add xmlns="http://tempuri.org/">
- ▶       <x>33</x>
- ▶       <y>66</y>
- ▶     </Add>
- ▶   </soap:Body>
- ▶ </soap:Envelope>

# Диспечеризация сообщений

- ▶ using System.Web.Services;
- ▶ using System.Web.Services.Protocols;  
[WebService(Namespace="http://example.org/math")]
- ▶ public class MathService {
- ▶     [WebMethod]
- ▶     public double Add(double x, double y)
- ▶     { return x + y; }
- ▶     [WebMethod]
- ▶     [SoapDocumentMethod(Action="urn:math:subtract")]
- ▶     public double Subtract(double x, double y)
- ▶     { return x - y; }
- ▶     ...
- ▶ }

# Преобразование XML в объекты



# Автоматическое генерирование WSDL

```
<configuration>  
  <system.web>  
    <webServices>  
      <wsdlHelpGenerator  
        href="MyDocumentation.aspx"/>  
    </webServices> ...
```

```
<configuration>  
  <system.web>  
    <webServices>  
      <protocols> <remove  
        name="Documentation"/>  
    </protocols> ...
```

# Создание клиента

- ▶ `using MathServiceJit;`
- ▶ `private int calc()`
- ▶ `{`
- ▶ `MathServiceJit service = new MathServiceJit();`
- ▶ `int res = service.Add(2,3) ;`
- ▶ `return res;`
- ▶ `}`