

Camada de Serviços

Leandro Garcia



Camada de Serviços



O que é?

- Exposição de funcionalidades para a camada de interfaces com usuário ou para sistemas externos
- Movimento de padrão SOA (Arquitetura Orientada a Serviços)



Onde estamos?

- Armazenamento de dados
- Regra de negócio
- Barramento de Serviços
- Interfaces com o usuário



Onde estamos?

- Armazenamento de dados

- Regra de negócio

- Barramento de Serviços

Interfaces

Sistemas Extern.

- ✓ Esta camada pode aparecer em topologias diferentes, dependendo da implementação



Por que utilizar camada de Serviços

- ✓ **Encapsula** as camadas de negócio e dados
 - ✓ A camada de interface não precisa conhecer as camadas de negócio e de dados, somente a camada de serviços
- ✓ **Suporta múltiplas interfaces (reuso)**
 - ✓ Web, App, Softwares externos, APIs, etc.
 - ✓ Reutilização dos serviços
 - ✓ A interface não deve interferir no funcionamento dos serviços
- ✓ **Ajuda no isolamento de interface**
 - ✓ Contribui para o baixo acoplamento
 - ✓ Dificulta a construção de código de baixa qualidade



Por que utilizar camada de Serviços

- ✓ Geram independência de tecnologia e facilidade de integrações com protocolos padrão



Características - chave

- Deve permitir concorrência
 - Chamadas em paralelo são premissas
 - Muitas chamadas simultâneas
- Não armazena Status (*Stateless*)
 - Cada chamada é independente
 - Envia o conjunto de dados completo
 - Recebe o retorno completo do processamento



Contrato – padrão *command* (SOAP)

- Desejável uso de objetos como insumo para chamadas
 - Objetos criados na camada de Domínio / Dados
 - Diminui mudanças na escrita dos métodos
 - Diminui

Desejável uso de objetos ou interfaces como retornos



SOAP vs REST



Diferenças

- SOAP é um protocolo; REST é uma arquitetura (usa protocolo existente)
- SOAP expõe comportamento; REST expõe dados
- SOAP é mais pesado e complexo que REST



SOAP - Simple object access protocol

- WSDL – Contrato entre consumidor e provedor
- Opção mais pesada, menor performance
- Independência de transporte (HTTP, TCP, etc)
- Tipos bem tipados
- Transfere dados em formato de XML
- Mais opções de segurança, autenticação e autorização
- Possibilidades de automação



REST - Representative state transfer

- Mais flexível (ou permissivo)
- Usa sempre HTTP
- Curva de aprendizado menor
- Qualquer formato (Json, XML, CSS, etc)
- Maior Eficiência



Mercado

Rest é utilizado em 70% das APIs públicas



Web Services ASMX



ASMX – Web Services

- Cria de forma fácil e rápida um conjunto de serviços SOAP
- O desenvolvedor só precisa escrever o método, que será convertido para um Serviço
- Cria-se os contratos dos serviços automaticamente

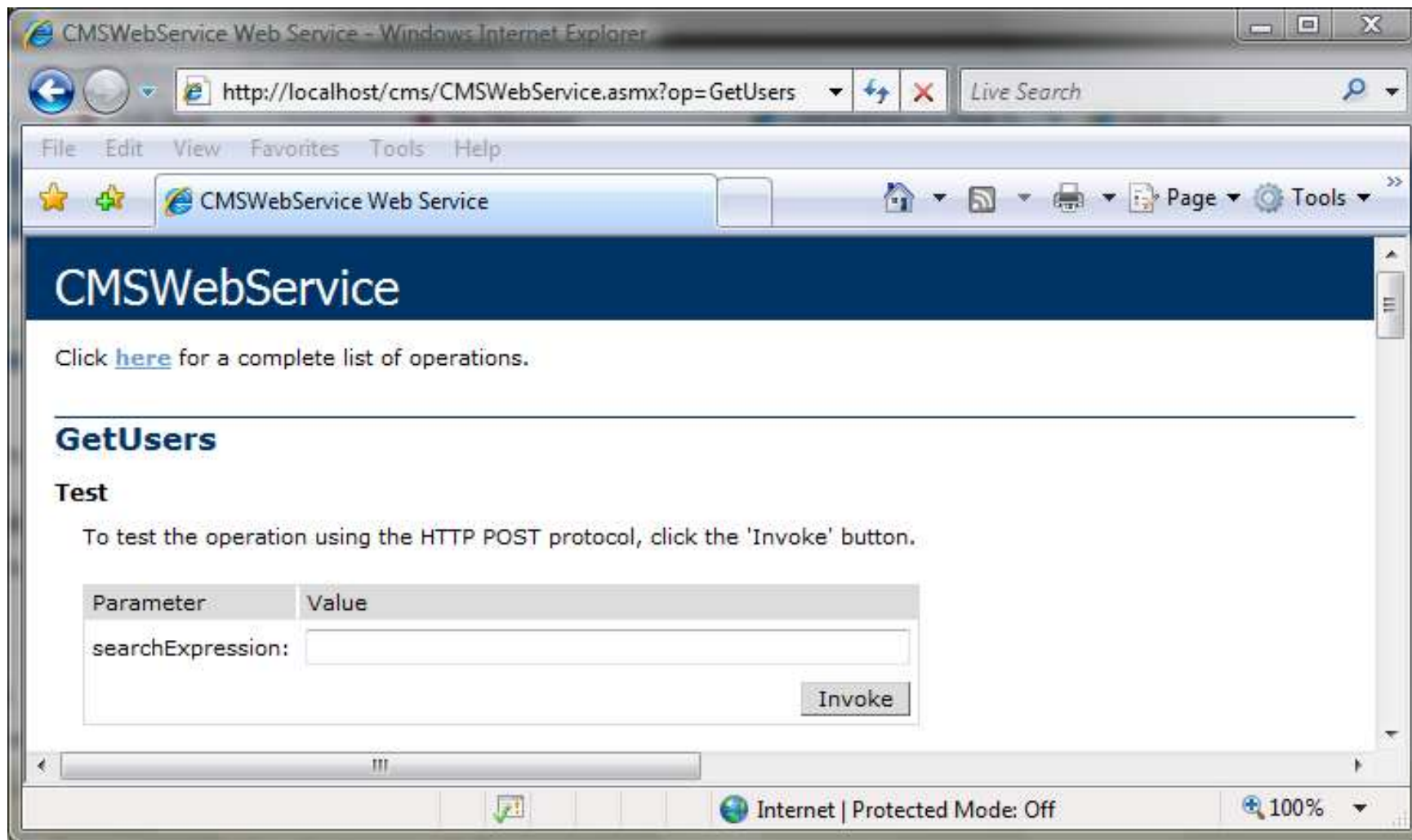


A cara do ASMX

```
namespace NetFw.Servico
{
    /// <summary>
    /// Descrição resumida de WebService
    /// </summary>
    [WebService(Namespace = "http://tempuri.org/")]
    [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
    [System.ComponentModel.ToolboxItem(false)]
    // Para permitir que esse serviço da web seja chamado a partir do script, usando ASP.NET AJAX,
    // [System.Web.Script.Services.ScriptService]
    public class WebService : System.Web.Services.WebService
    {
        [WebMethod]
        public string HelloWorld()
        {
            return "Olá, Mundo";
        }
    }
}
```



Simple e Rápido

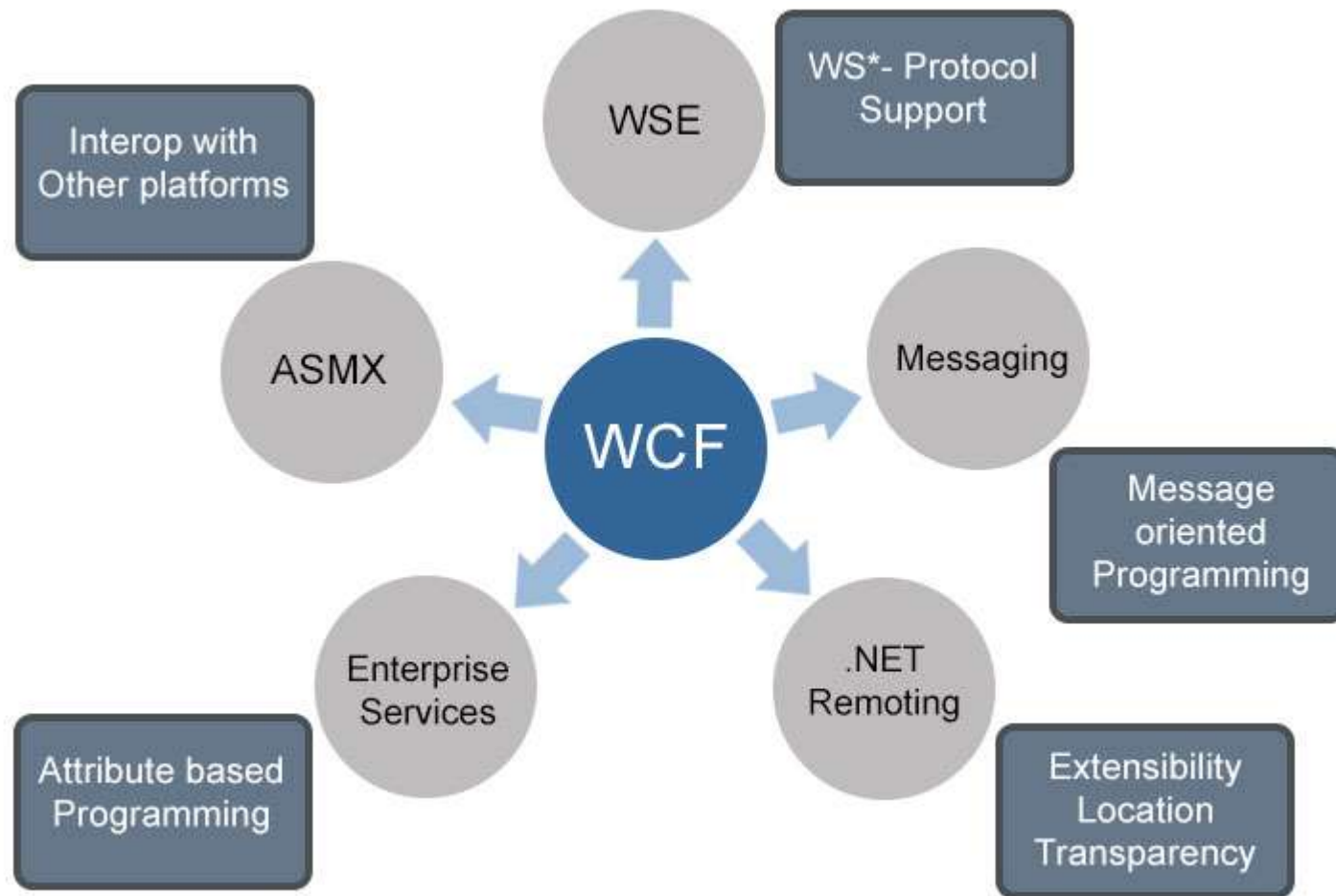


```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:s="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/" xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/" xmlns:tns="http://tempuri.org/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://tempuri.org/">
  <wsdl:types>
    <s:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/">
      <s:element name="CadastraSuspeita">...</s:element>
      <s:element name="CadastraSuspeitaResponse">...</s:element>
      <s:element name="CadastraFoco">...</s:element>
      <s:element name="CadastraFocoResponse">...</s:element>
      <s:element name="CadastraAvaliacao">...</s:element>
      <s:element name="CadastraAvaliacaoResponse">...</s:element>
      <s:element name="GetCoordenadas">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="latitude" type="s:string"/>
            <s:element minOccurs="0" maxOccurs="1" name="longitude" type="s:string"/>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetCoordenadasResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetCoordenadasResult" type="s:string"/>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="CadastraFocoTelegram">...</s:element>
      <s:element name="CadastraFocoTelegramResponse">...</s:element>
      <s:element name="CadastraSuspeitaTelegram">...</s:element>
      <s:element name="CadastraSuspeitaTelegramResponse">...</s:element>
      <s:element name="CadastraAvaliacaoTelegram">...</s:element>
      <s:element name="CadastraAvaliacaoTelegramResponse">...</s:element>
      <s:element name="registrarUsuario">...</s:element>
      <s:element name="registrarUsuarioResponse">...</s:element>
      <s:element name="AtualizaUsuario">...</s:element>
      <s:element name="AtualizaUsuarioResponse">...</s:element>
      <s:element name="TestNetwork">...</s:element>
      <s:element name="TestNetworkResponse">...</s:element>
    </s:schema>
  </wsdl:types>
  <wsdl:message name="CadastraSuspeitaSoapIn">...</wsdl:message>
  <wsdl:message name="CadastraSuspeitaSoapOut">...</wsdl:message>
  <wsdl:message name="CadastraFocoSoapIn">...</wsdl:message>
  <wsdl:message name="CadastraFocoSoapOut">...</wsdl:message>
```

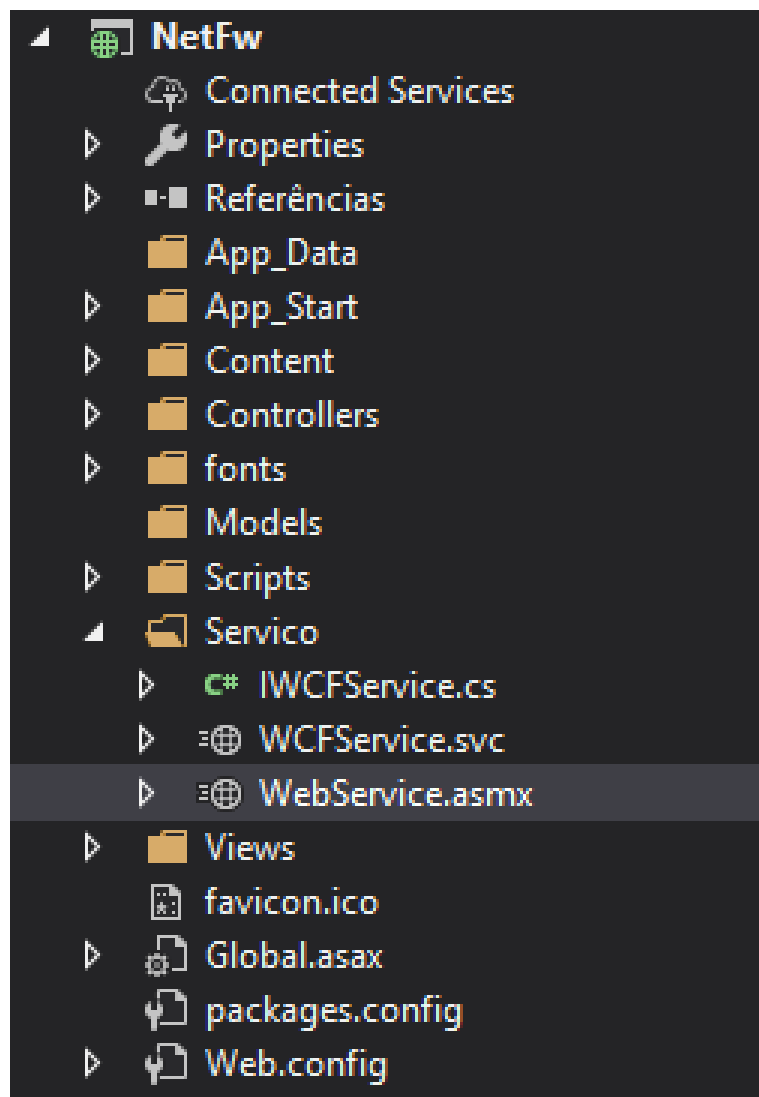
WCF – Windows Communication Foundation



WCF – Encapsulando Camada de Serviços

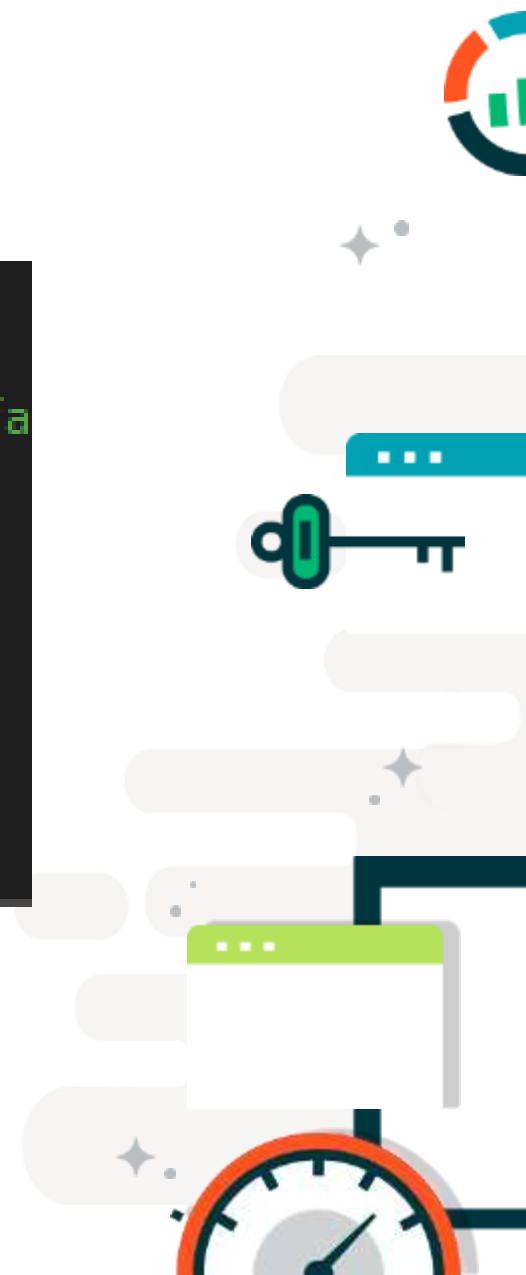


A cara do WCF



A cara do WCF

```
namespace NetFw.Servico
{
    // OBSERVAÇÃO: Você pode usar o comando "Renomear" no menu "Refa
    [ServiceContract]
    public interface IWCFService
    {
        [OperationContract]
        string DoWork();
    }
}
```



A cara do WCF

```
namespace NetFw.Servico
{
    // OBSERVAÇÃO: Você pode usar o comando "Renomear" no menu "Refatorar" para alterar o nome da classe "WCFService"
    // OBSERVAÇÃO: Para iniciar o cliente de teste do WCF para testar esse serviço, selecione WCFService.svc no menu "Executar"
    public class WCFService : IWCFService
    {
        public string DoWork()
        {
            return "Olá, Mundo";
        }
    }
}
```


Desenho da camada de domínio

- Desenho de um cenário para cada padrão, contextualizando e explicando a escolha pelo padrão.
- Exemplos reais ou fictícios



Desenho da camada de Serviço

- Desenho de um cenário para uso de REST e outro para uso de SOAP
- Exemplos reais ou fictícios



Desenho da camada de Serviço do seu projeto

- Quais os pontos que de utilização de serviços internos e externos
- Qual tecnologia será utilizada para cada ponto?
- Razões pelas quais se deu as escolhas



Camada de Domínio/Negócio e camada de Dados

Leandro Garcia



