



**PROTÓTIPO PARA ATUALIZAÇÃO
ASSÍNCRONA DE DADOS UTILIZANDO
WEB SERVICES**



Sérgio Koch Van-Dall
sergiod@inf.furb.br

Orientador: Prof. Paulo Fernando da Silva

**UNIVERSIDADE REGIONAL DE BLUMENAU
CURSO DE CIÊNCIAS DA COMPUTAÇÃO - BACHARELADO**

ROTEIRO

- Introdução
- Objetivos
- Fundamentação Teórica
 - ✓ .NET
 - ✓ XML
 - ✓ SOAP
 - ✓ Web Services
- Desenvolvimento
- Resultados e Discussão
- Conclusão
- Extensões

INTRODUÇÃO

- Enorme crescimento no número de redes de computadores e a complexidade de seus softwares;
- Necessidade da comunicação entre eles para que possam compartilhar informações e trabalharem de forma cooperativa;
- Independência de arquitetura, sistema operacional e linguagem de programação;

- Uso de Web Services para resolver a interoperacionalidade das aplicações;

- Uso do XML como um formato padrão na comunicação e troca de informação entre as aplicações.

OBJETIVOS

O objetivo principal é a criação de um protótipo para atualização assíncrona de dados no servidor de banco de dados utilizando *Web Services* através do protocolo SOAP.

Objetivos específicos:

- utilizar XML como uma linguagem de marcação para a confecção dos documentos utilizados no intercâmbio das informações;
- criar um *Web Service* para guardar as informações de forma estruturada no arquivo XML;
- gerar atualização assíncrona dos dados no servidor de banco de dados através da chamada do *Web Service*;
- criar uma aplicação cliente para demonstração do protótipo.

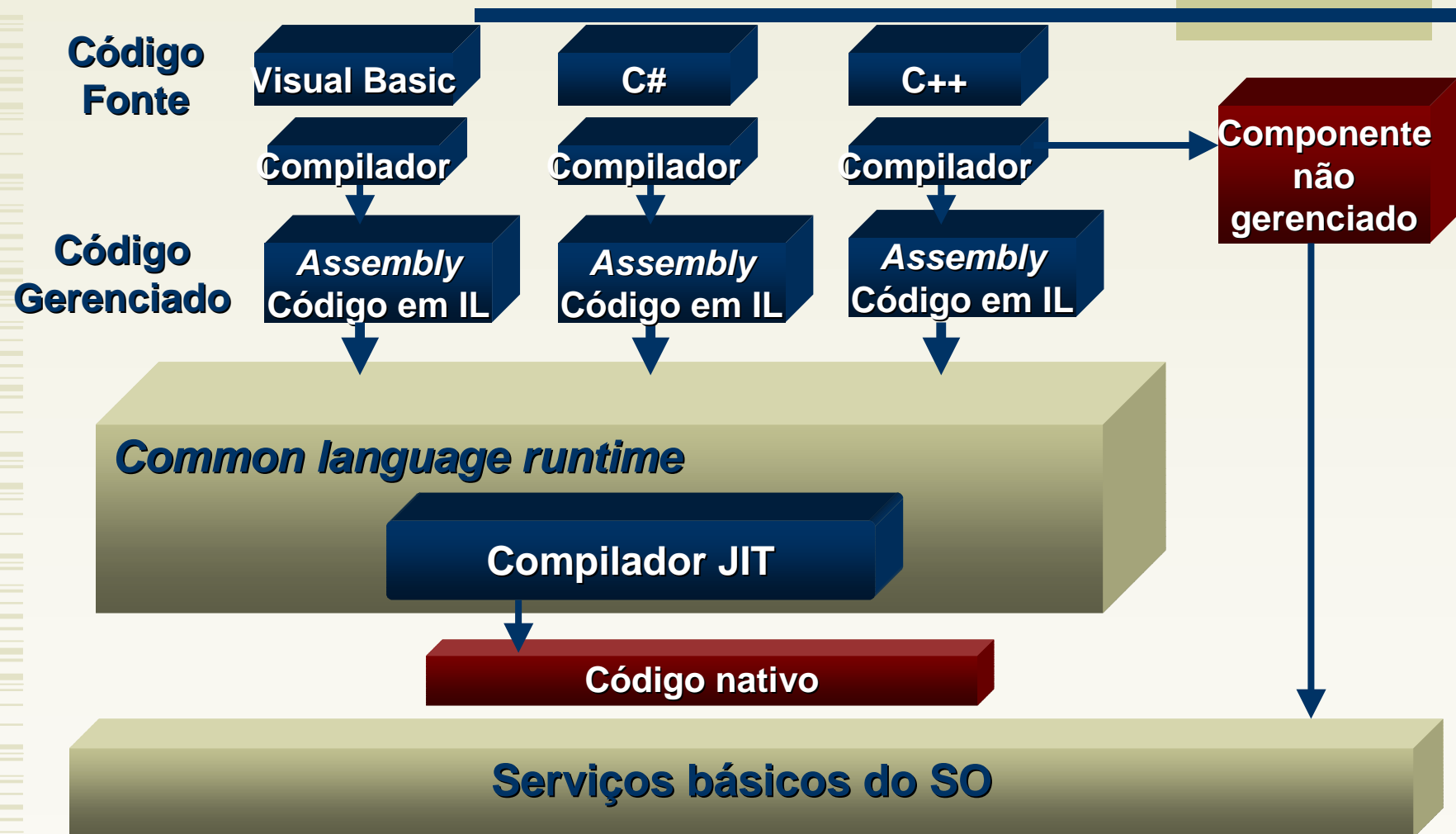
FUNDAMENTAÇÃO TEÓRICA .NET

É uma plataforma para desenvolvimento e execução de software, criada visando o desenvolvimento de software para a internet e aplicações distribuídas.

Características:

- Independência de Sistema Operacional;
- Integração entre Linguagens;
- Confiabilidade;
- Facilidade de Distribuição.

FUNDAMENTAÇÃO TEÓRICA .NET



FUNDAMENTAÇÃO TEÓRICA

XML (eXtensive Markup Language)

É uma linguagem de marcação com um formato padrão para comunicar dados estruturados entre aplicações e sistemas de computadores, mesmo quando estão envolvidas múltiplas arquiteturas, linguagens de programação e padrões.

Consiste de elementos e atributos que implementam uma hierarquia de árvore.

▪Pode ser usado para:

- Formato de ligação (para transferência de dados entre cliente e servidor);
- Formato de troca de informação (por exemplo: entre a aplicação e a base de dados);
- Formato persistente (para guardar dados).

FUNDAMENTAÇÃO TEÓRICA

SOAP (*Simple Object Access Protocol*)

É um protocolo padrão para troca de informações entre aplicações, independente de sistema operacional, linguagem de programação ou modelo de objeto.

Utilizado para implementar chamadas de procedimentos remotos e trocar dados entre o código que utiliza um serviço web (o cliente) e o serviço web propriamente dito (o servidor).

Vantagens:

- Capacidade de transpor firewalls facilmente ao ser utilizado sobre o HTTP;
- Utiliza a estruturação de dados XML;
- É satisfatoriamente mapeado no padrão solicitação/resposta do HTTP.

FUNDAMENTAÇÃO TEÓRICA

SOAP (*Simple Object Access Protocol*)

Troca de Mensagens:

É feita através da sintaxe de envelope para envio e recebimento de mensagens XML que são enviadas sobre HTTP entre clientes e Web Services.

Contêm os seguintes elementos SOAP:

Envelope (elemento raiz obrigatório)
Cabeçalho (elemento opcional)
Corpo (elemento obrigatório)

FUNDAMENTAÇÃO TEÓRICA WEB SERVICES

É uma tecnologia de chamada remota de objetos, usado para disponibilizar serviços interativos na web.

Características:

- Plataforma Web;
- Interoperacionalidade;
- Reuso;
- Fraco acoplamento.

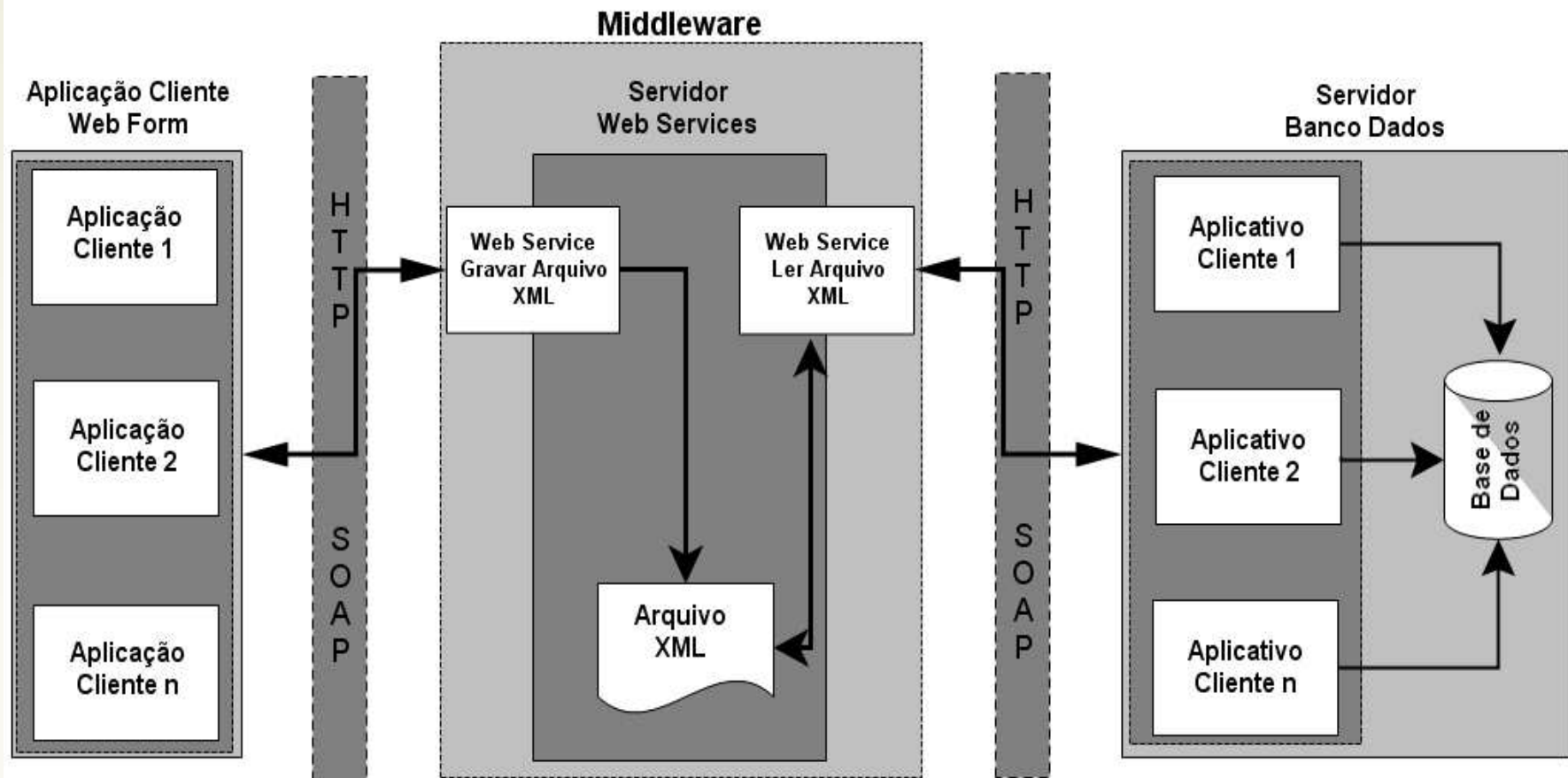
Vantagens:

- Simplicidade;
- Padrões abertos;
- Flexibilidade;
- Escopo.

DESENVOLVIMENTO MOTIVAÇÃO

- Dificuldade na comunicação entre as aplicações de diferentes empresas;
- Comunicação e troca dos dados de forma simples e transparente em um sistema heterogêneo;
- Desenvolver um protocolo independente de arquitetura, sistema operacional e linguagem de programação, para comunicar-se em um sistema distribuído.

DESENVOLVIMENTO CENÁRIO



DESENVOLVIMENTO PRICIPAIS REQUISITOS

Requisitos Funcionais:

- realizar o acesso aos *Web Services* através de uma aplicação cliente;
- realizar o acesso aos *Web Services* através de um cliente servidor de banco de dados;
- realizar atualização assíncrona no servidor de banco de dados através de *Web Services* utilizando o protocolo SOAP.

Requisitos Não Funcionais:

- utilizar um arquivo XML como uma linguagem de marcação para a confecção das informações que serão compartilhadas.

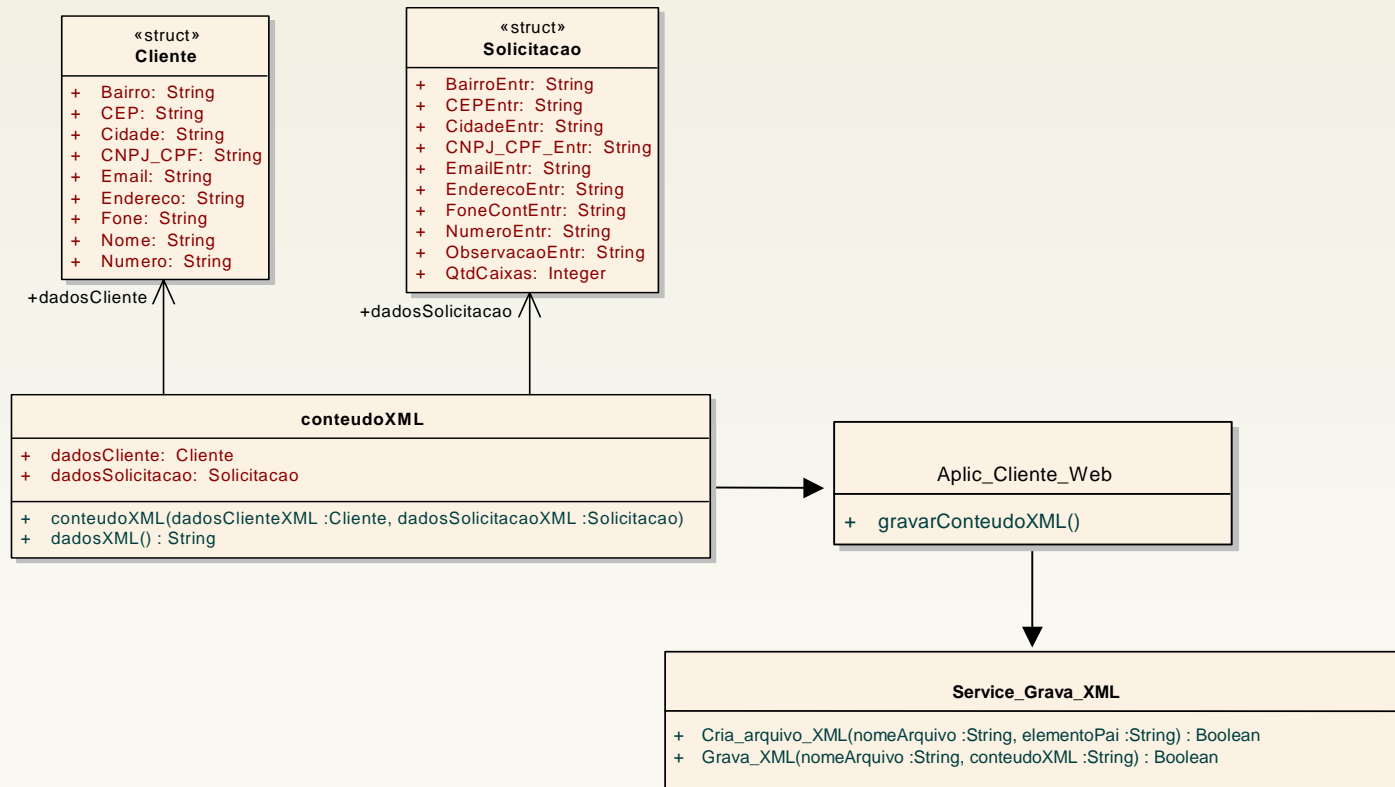
DESENVOLVIMENTO ESPECIFICAÇÃO

Utiliza um servidor de *Web Services* que disponibiliza métodos para gravação e leitura de um arquivo no formato XML.

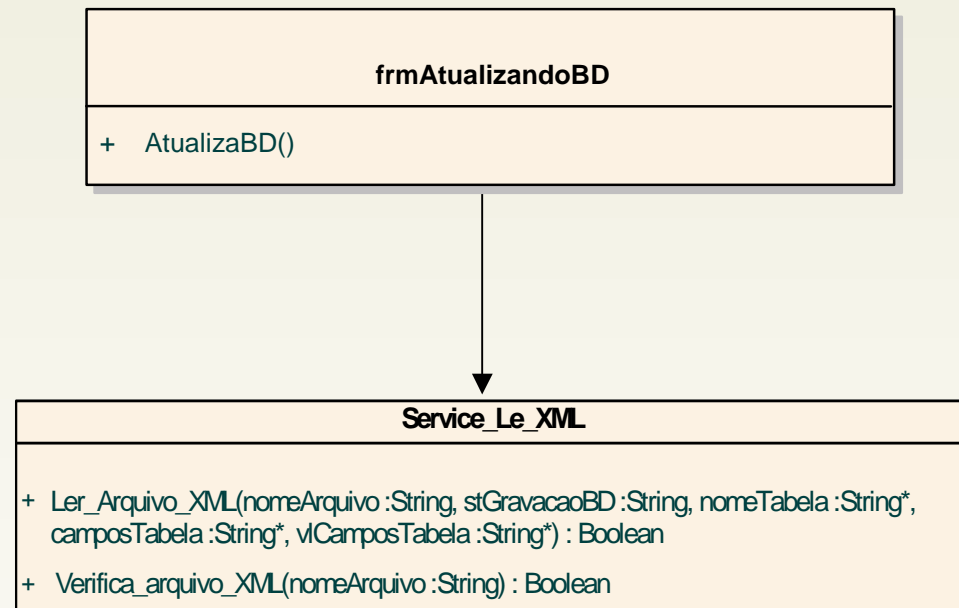
Consiste em classes que fazem referência as respostas e tratamentos efetuados no servidor de *Web Services* aos métodos invocados remotamente pela aplicação cliente e pelo servidor de banco de dados.

As classes são apresentadas através de diagramas UML.

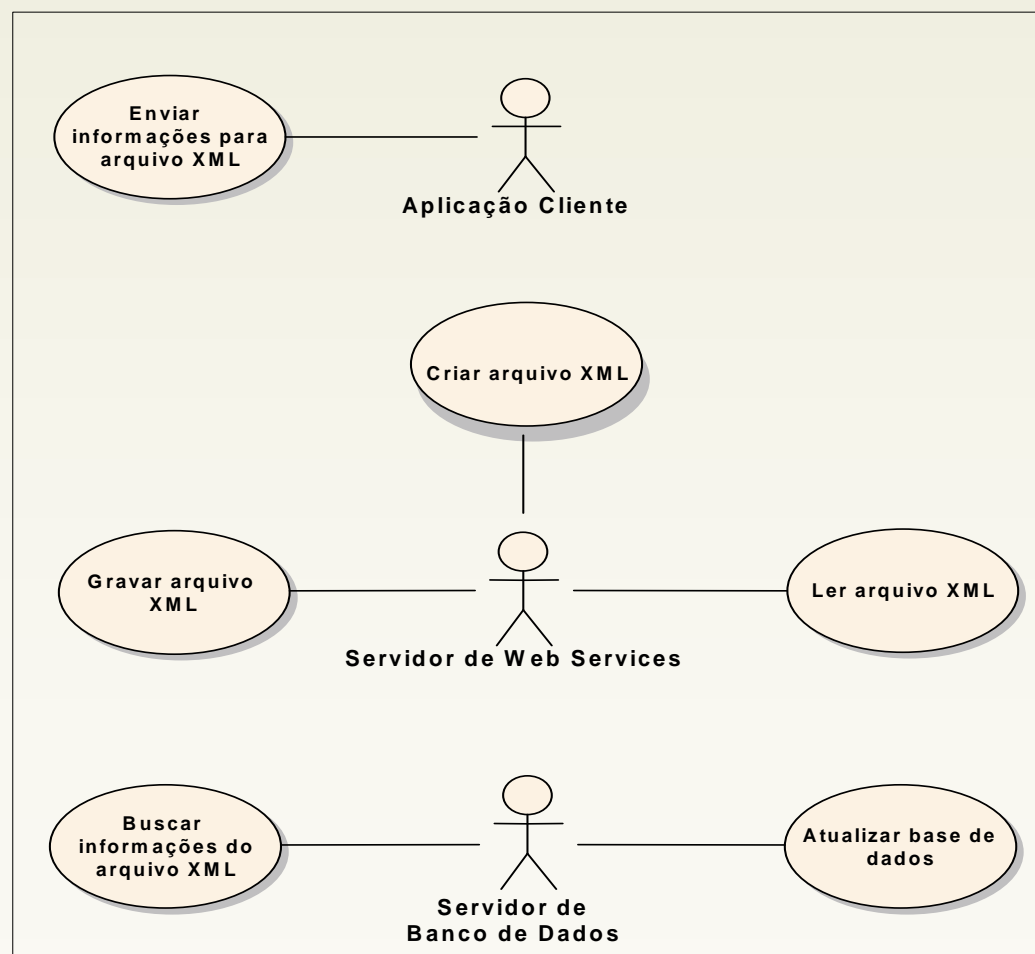
DESENVOLVIMENTO APLICAÇÃO CLIENTE (Diagrama de Classe)



DESENVOLVIMENTO SERVIDOR DE BD (Diagrama de Classe)



DESENVOLVIMENTO CASOS DE USO



DESENVOLVIMENTO IMPLEMENTAÇÃO

- Aplicação cliente consiste em uma aplicação *Web Form* desenvolvida em VB .Net no formato ASP;
- Os *Web Services* consistem em classes em VB .Net que contém métodos que são invocados através de mensagens SOAP;
- O servidor de banco de dados é uma aplicação cliente baseada em *Windows Form*, utiliza o banco de dados MSDE;
- Foi utilizado o servidor *Internet Information Server* (IIS), responsável por disponibilizar as classes que fornecem os serviços *web* e gerenciar a troca de mensagens.

RESULTADOS E DISCUSSÃO

- Realiza o envio de informações de forma estruturada para um arquivo no formato XML através de uma aplicação *web*, disponível em um local remoto comum entre as aplicações;
- Permite a leitura destas informações e atualização da base de dados de forma assíncrona, através de um aplicativo servidor de banco de dados;
- A linguagem XML com o protocolo SOAP pode integrar sistemas distintos mais facilmente se comparado com outros padrões de integração existentes atualmente;

RESULTADOS E DISCUSSÃO

- Forneceu suporte adequado à atualização das informações contidas no arquivo XML em um banco de dados *SQL Server* (MSDE), podendo ser facilmente substituído por outros, tais como, *MySQL*, *Oracle*, *Interbase* entre outros;
- Pode ser visto como um *middleware* responsável por disponibilizar um arquivo no formato XML, podendo assim, efetuar a integração de sistemas distintos através de um ambiente distribuído, utilizando um formato padrão na comunicação e troca de informações entre os mesmos.

CONCLUSÃO

- Este trabalho alcançou na íntegra seu objetivo principal;
- O uso de servidor IIS para gerenciar a troca de informações e a comunicação entre as aplicações no cenário composto demonstrou uma funcionalidade satisfatória;
- O uso do XML e .NET é sem dúvida muito interessante, pois além de conhecer funcionalidades do XML, também foi possível descobrir e utilizar uma plataforma de desenvolvimento que oferece suporte a múltiplas linguagens de programação.

CONCLUSÃO

Dificuldades:

- falta de experiência em trabalhar com XML;
- não conhecimento da linguagem de programação VB .NET e das funcionalidades da plataforma .NET.

Limitações:

- requer o .NET *Framework* SDK;
- rodar somente no servidor IIS.

EXTENSÕES

- Possibilitar a criação de novas tabelas no servidor de banco de dados a partir do arquivo XML;
- Efetuar a alteração dos dados já inseridos na tabela no servidor de banco de dados a partir do arquivo XML;
- Utilizar outras linguagens de programação para a aplicação *web*, como PHP ou JSP, possibilitando assim, o uso de outros servidores, como o *Apache* ou o *Tomcat*.



**PROTÓTIPO PARA ATUALIZAÇÃO
ASSÍNCRONA DE DADOS UTILIZANDO
WEB SERVICES**



OBRIGADO !!!