



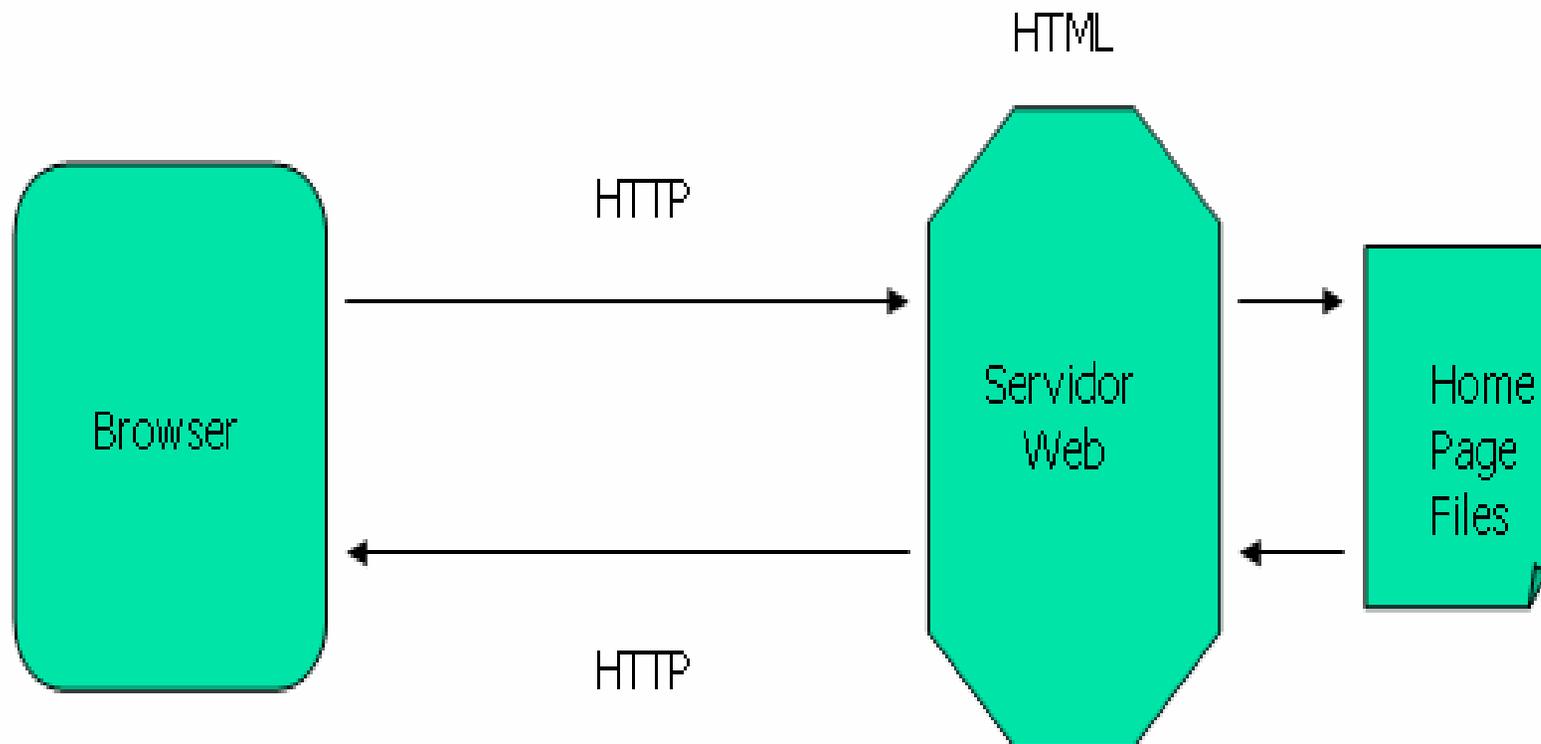
O Essencial em Serviços Web

Prof. João Bosco M. Sobral

UFSC

Departamento de Informática e Estatística

A Evolução da Web

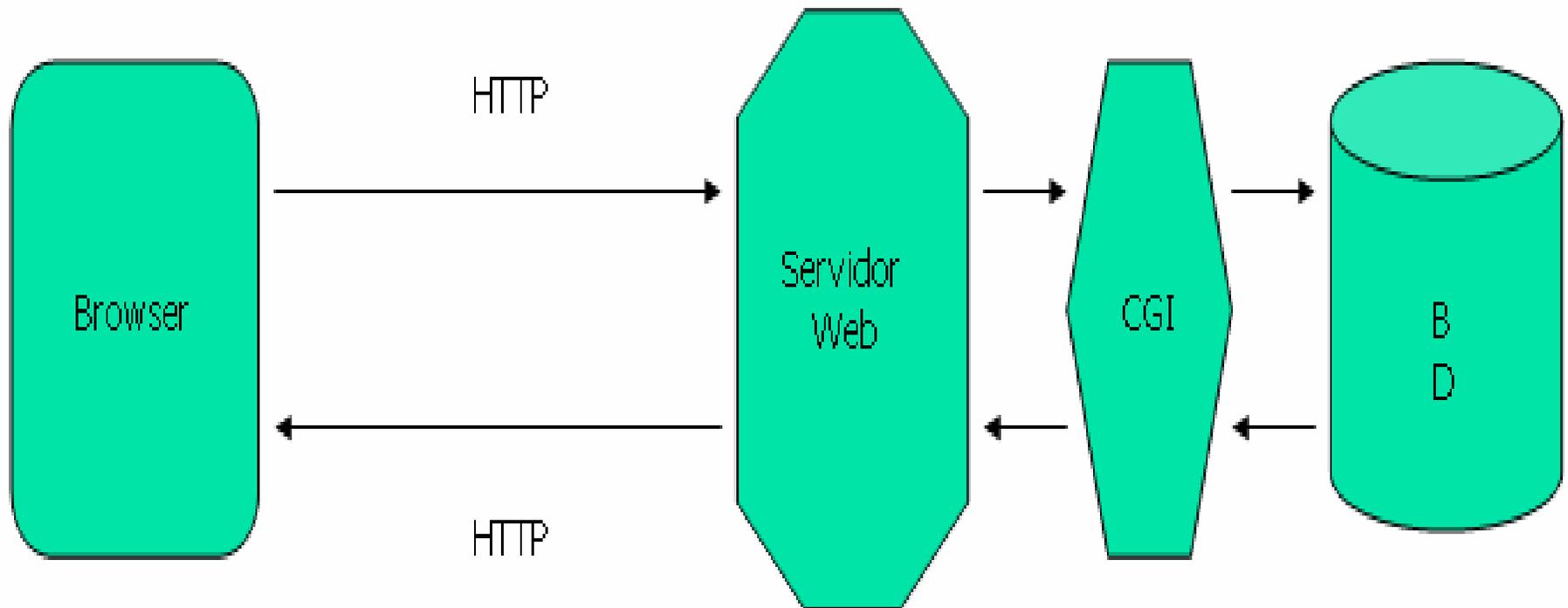


A Evolução da Web: Páginas Estáticas

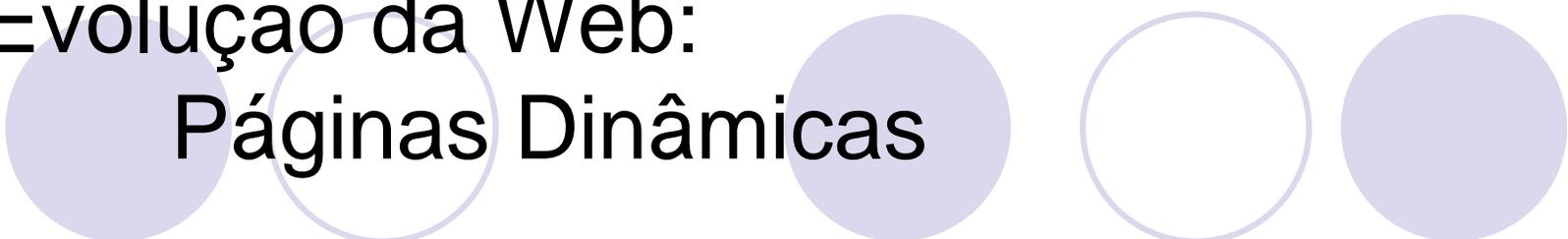


- Browser,
- Servidor Web,
- HTTP,
- HTML,
- JavaScript.

A Evolução da Web

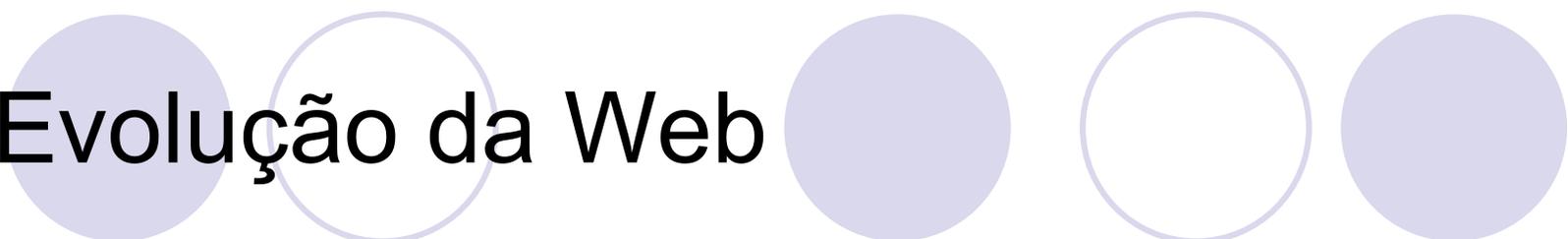


A Evolução da Web: Páginas Dinâmicas



- Browser,
Servidor Web,
HTTP, HTTPS
HTML, ...
- Servidor Web e mais Programas Externos
do tipo CGI (C, Perl, Python)
- Mais complexidade.

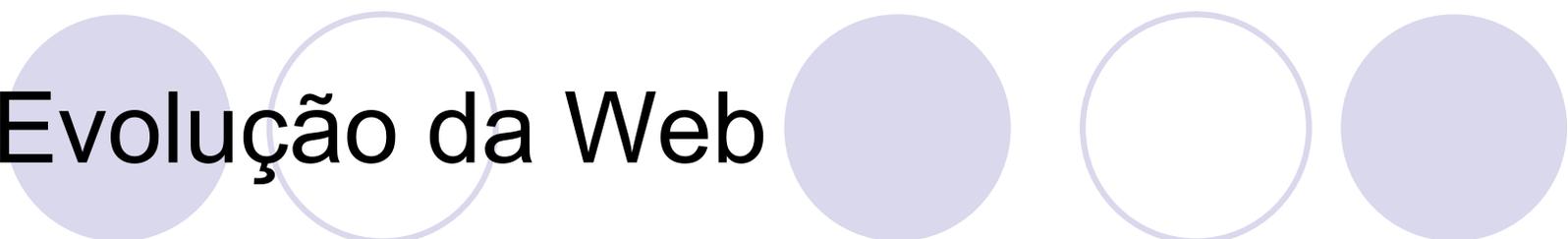
A Evolução da Web



- Linguagem Perl.
- Todas as plataformas de OS.
- Prototipação rápida.
- Ferramentas.
- Acesso à bases de dados.
- **Programação para a Web.**

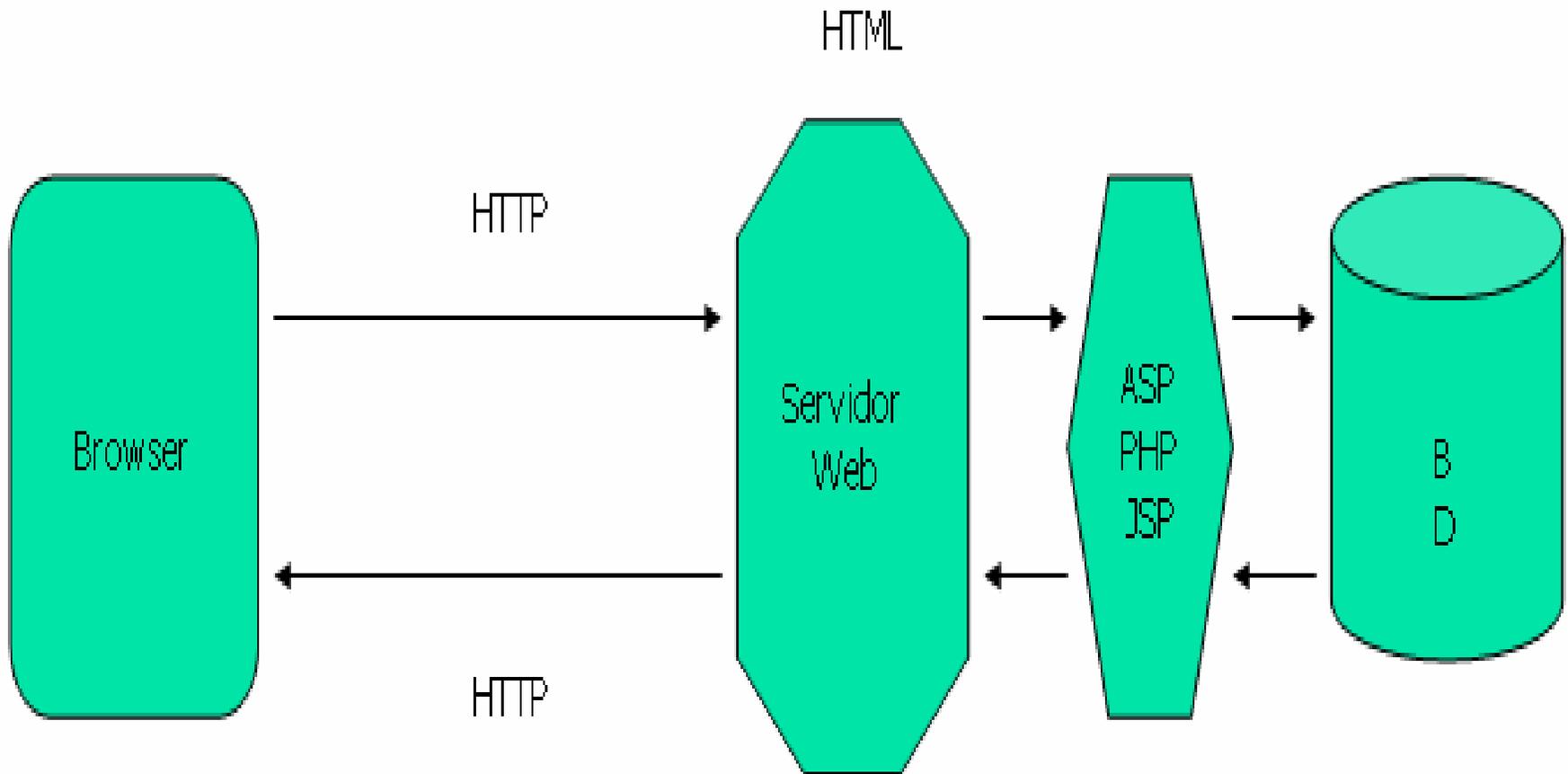
<http://www.perl.com/>

A Evolução da Web



- Linguagem Python.
- Python é uma linguagem **OO scripting** que faz tudo.
- Que você pode fazer com Perl.
- Projetada para ser orientada a objeto.
- Documentação em <http://www.python.org> para UNIX, Macintosh e Windows.

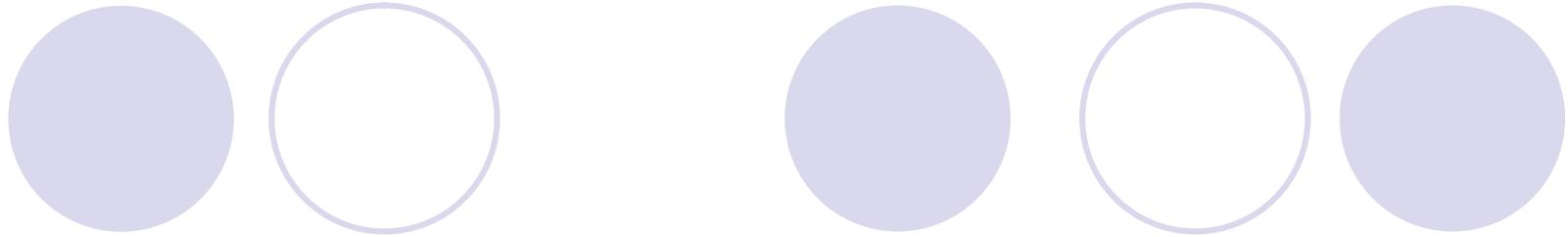
A Evolução da Web



A Evolução da Web: Páginas Dinâmicas



- Servidor Web mais Programas Externos
..., PHP, ASP, ...
- Object Web
Tecnologias Java (Servlets, JSP, J2EE)



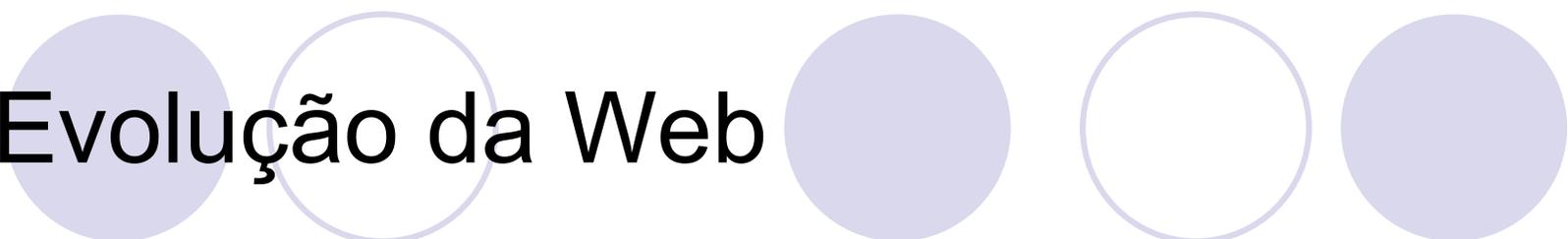
- Paradigma da Computação com Objetos Distribuídos:
 - dificuldades apresentadas por CORBA para Web,
 - middlewares com protocolo binário (CORBA-IIOP, RMI-IIOP),

- O problema são os Firewalls.

Em direção aos Serviços na Web

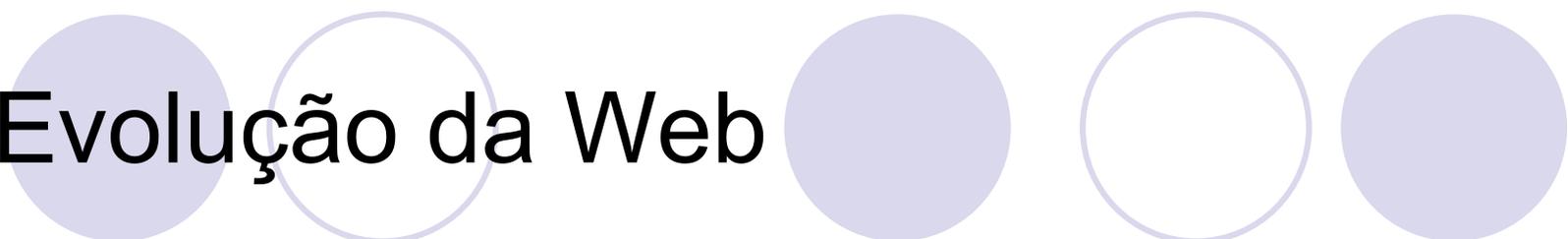
- Paradigma da Computação Distribuída para Serviços Web:
 - Cliente/Servidor
 - Objetos
 - XML

A Evolução da Web



- O inventor da Web, Tim Bernes-Lee, vislumbrou a próxima geração da Web será sobre dados, e não texto.
- HTML para texto.
- XML é para dados.
(descrição de informação).

A Evolução da Web

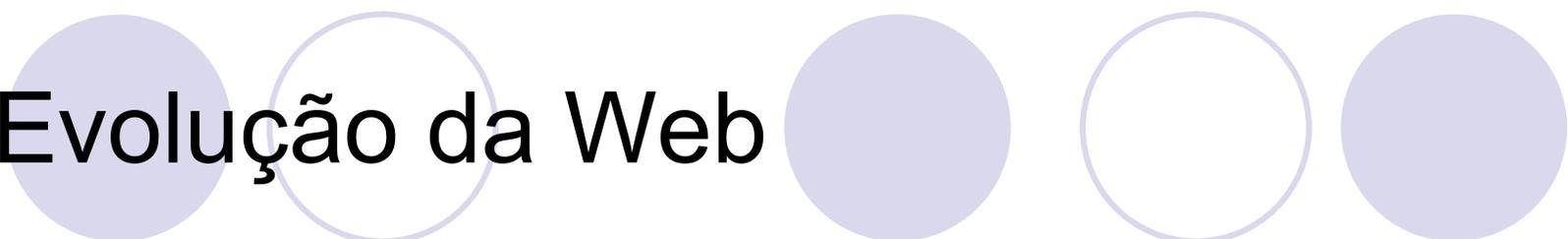


- A próxima geração da Web é voltada para objetivar diversas imperfeições da Web existente, notadamente a dificuldade de busca de informação exata sobre strings de texto embutidas em páginas Web.
- Web Semantics (Semântica da Web).

Integração de Aplicações: Legado

- Soluções para comunicação aplicação-a-aplicação devem ser derivadas das tecnologias Internet existentes.

A Evolução da Web



- **Conceito de Serviço:**

Uma abstração de um conjunto de operações providas a clientes, as quais permitem a eles realizarem uma tarefa particular.

Endereços de Serviços

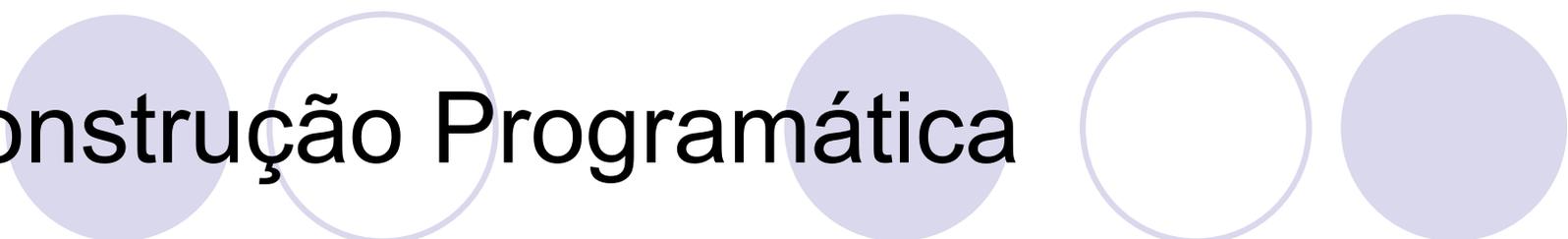


- Os pontos finais da Web, os endereços URL, proporcionarão **serviços** que processarão dados XML, da mesma maneira que browsers processam texto HTML.



Endereços de Serviços

- Estes **endereços de serviços** também poderão ser incluídos em qualquer aplicação de Web Services.



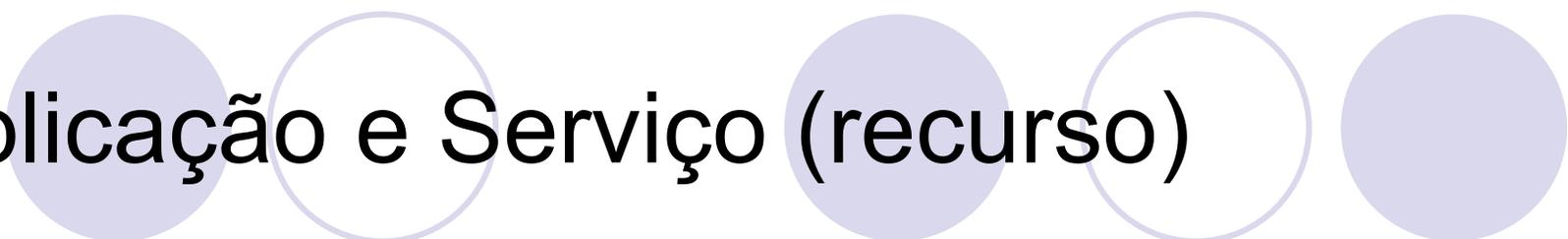
Construção Programática

- Serviços na Web devem referenciar dinamicamente a pontos finais na Web, ou endereços Web (URLs).

Mapeamento XML

A decorative graphic consisting of six circles arranged in a horizontal line. The first circle is solid light purple. The second circle is a light purple outline. The third circle is solid light purple. The fourth circle is a light purple outline. The fifth circle is solid light purple. The sixth circle is a light purple outline.

- E também mapear dados para/de XML, transparentemente.



Aplicação e Serviço (recurso)

- Assim, é possível se comunicar de uma aplicação para um recurso remoto de dados.

Serviços Pré-definidos



- Tipos de serviços padrões, acessíveis de qualquer aplicação, para criar e integrar aplicações que usam serviços pré-definidos na Web.



W3C

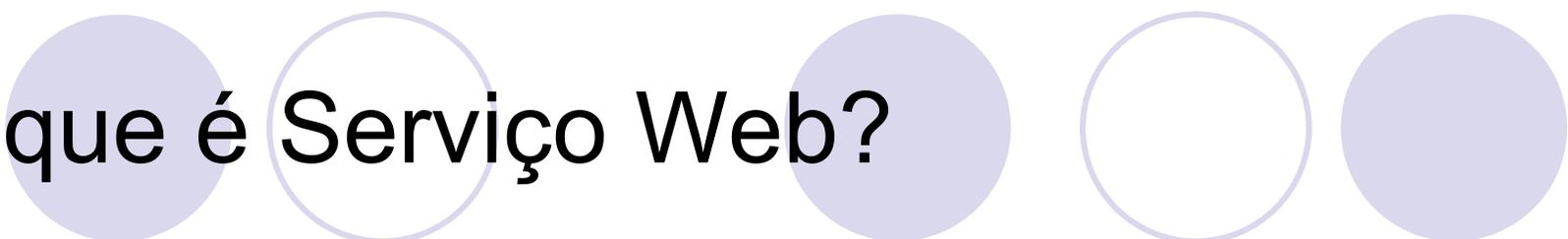
- Mas, para acontecer isto, é requerida uma padronização significativa.
- Padronização do W3C para a Web.
- Serviços Web requerem diversas tecnologias baseadas em XML.

Tecnologias Essenciais para Serviços na Web



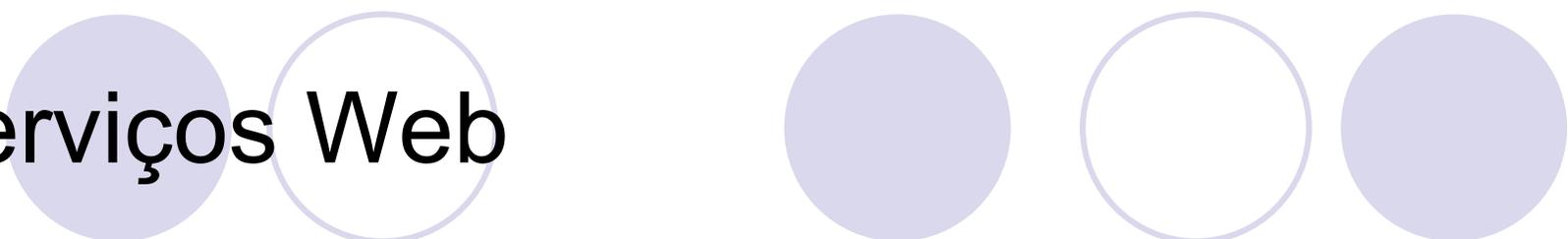
- XML (eXtended Markup Language)
- WSDL (Web Services Description Language)
- SOAP (Simple Object Access Protocol)
- UDDI (Universal Description, Discovery and Integration)

O que é Serviço Web?



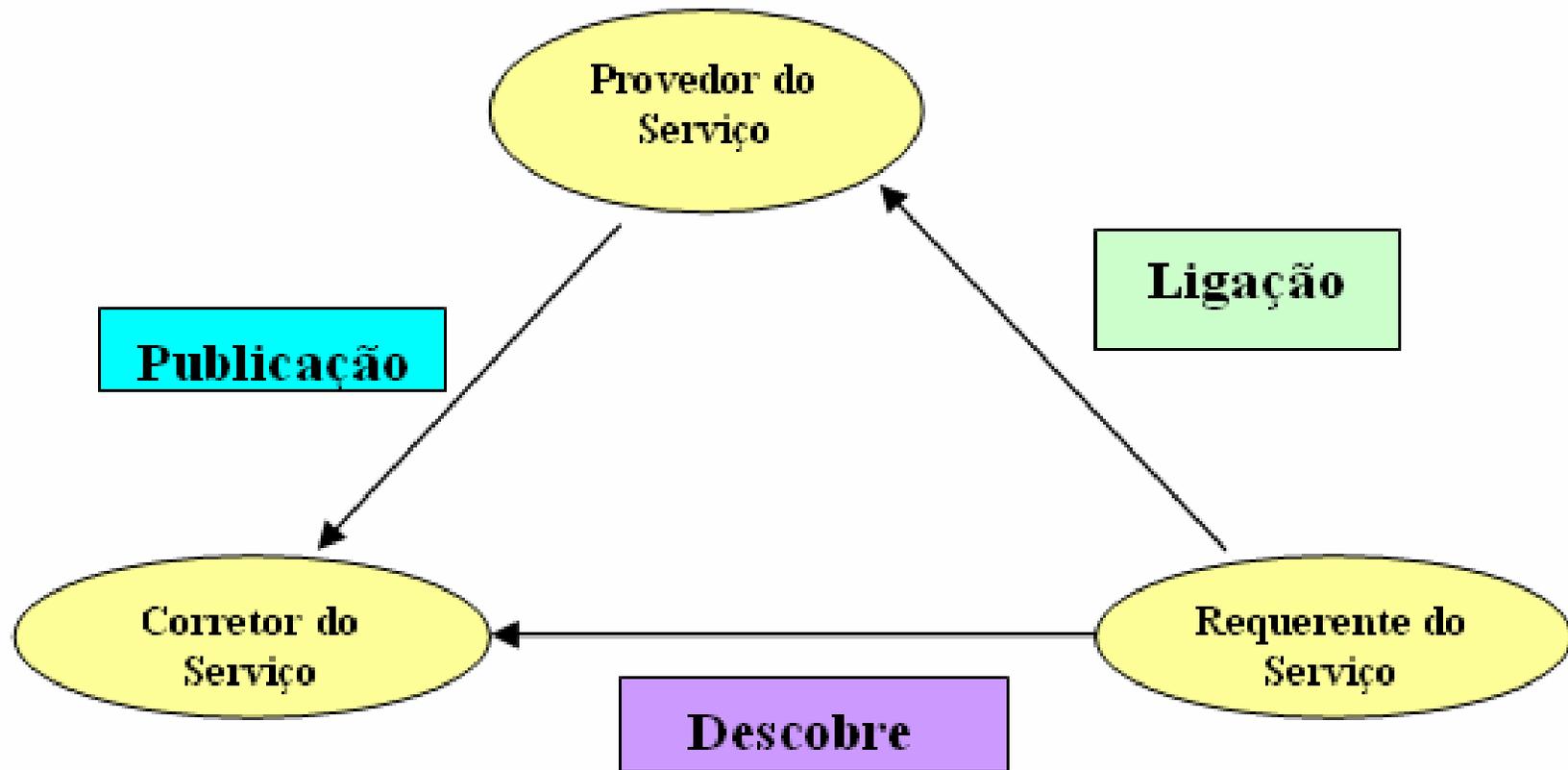
- Um **Serviço Web** é a lógica de uma aplicação (lado do servidor), disponível programaticamente e acessível sobre a Internet/Web, através da padronização de XML (W3C) e construída sobre protocolos padronizados sob esta linguagem.

Serviços Web

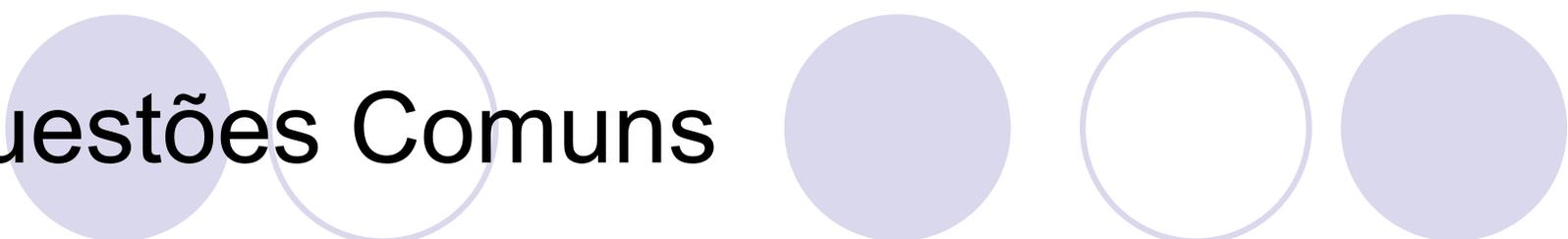


- Acesso de informação é através de um browser.
- Independência de linguagens.
- Torna a construção de aplicações distribuídas na Web menos difíceis (em relação à CORBA, DCOM).

Arquitetura Orientada-a-Serviço



Questões Comuns



- Como descrever um serviço Web ?
 - Que protocolos ele suporta ?
XML-RPC, HTTP+SOAP-XML
 - Que tipos de dados ele usa ?
Schemas XML
- Como publicar a localização de um Serviço Web ?

Questões relacionadas ao uso de Serviços Web



- Para conectividade de aplicação para aplicação.
- Para habilitar aplicações existentes à Web.
- Para inter-operar através de plataformas heterogêneas.

Qual dos benefícios de Web Services é mais atrativo para você?

- Provê um modo de construir aplicações dinâmicas.
- Torna o processo de desenvolvimento mais rápido.
- Simplifica o processo de aplicação.

Qual dos benefícios de Web Services é mais atrativo para você?

- Provê um modo de reutilizar aplicações existentes.
- Provê um modo de usar padrões abertos.
- Provê um modo de interoperar através de plataformas (OS) distintos.

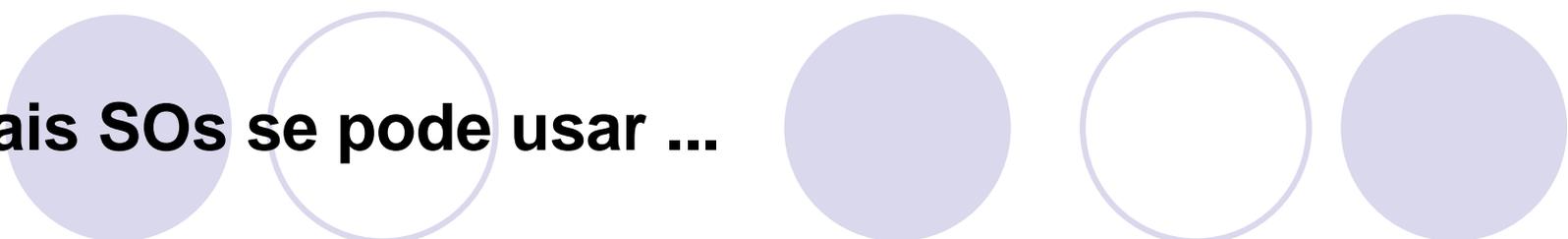
Quais linguagens se pode usar ?

- C or C++ tools
- VB tools
- C# tools
- Java tools

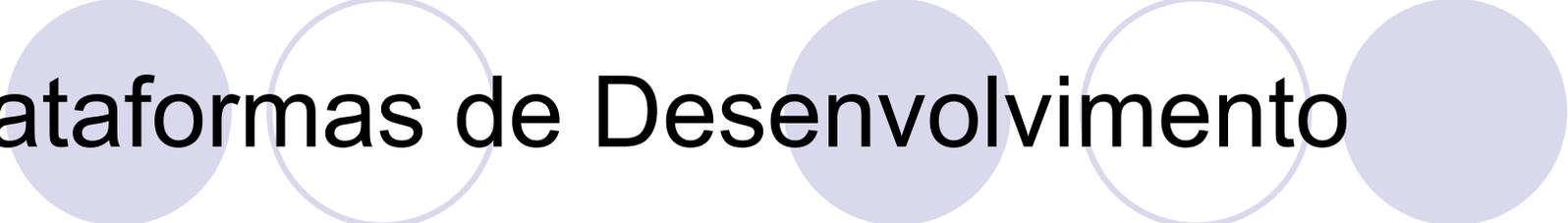
Quais linguagens se pode usar ?

- COBOL or PL1
- Scripting Languages
- HTML or XML
- outras

Quais SOs se pode usar ...

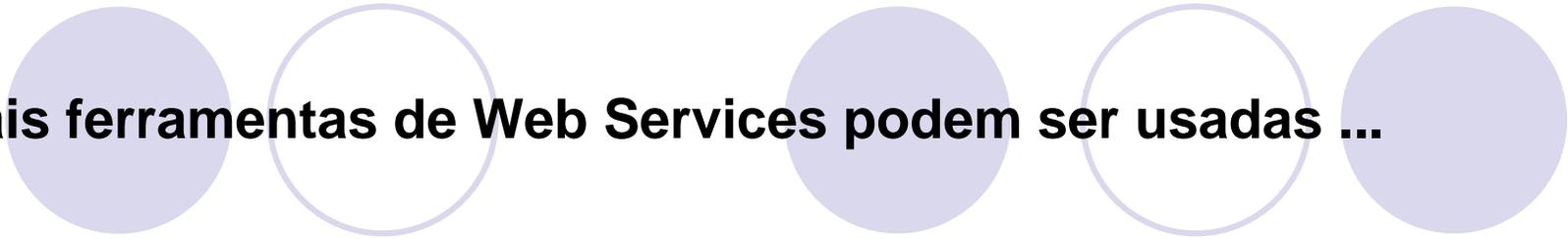


- Windows 2000 or NT
- Windows XP
- Solaris
- Linux
- HP-UNIX
- Mac OS
- IBM AIX



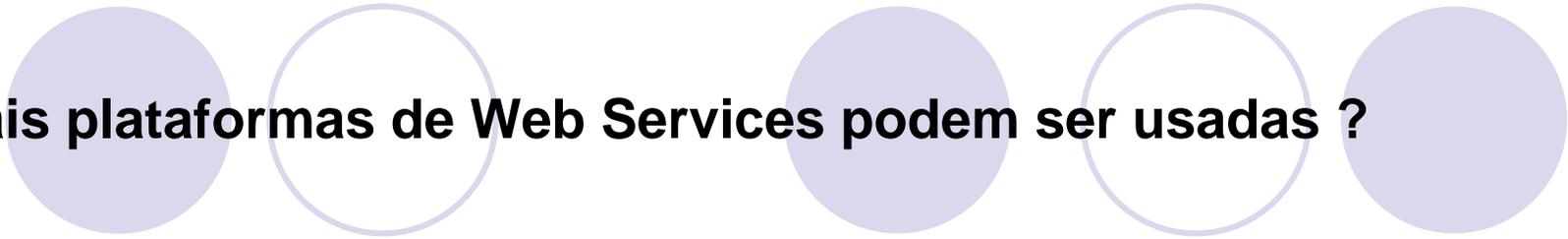
Plataformas de Desenvolvimento

- Qual das plataformas de desenvolvimento de WS você escolheria ?



Quais ferramentas de Web Services podem ser usadas ...

- **Apache Project SOAP - AXIS**
- Module BEA WebLogic Workshop
- Borland JBuilder
- Cape Clear Studio
- HP NetAction Toolkit
- IBM WebSphere SDK for Web Services
- IBM WebSphere Studio Application Developer.



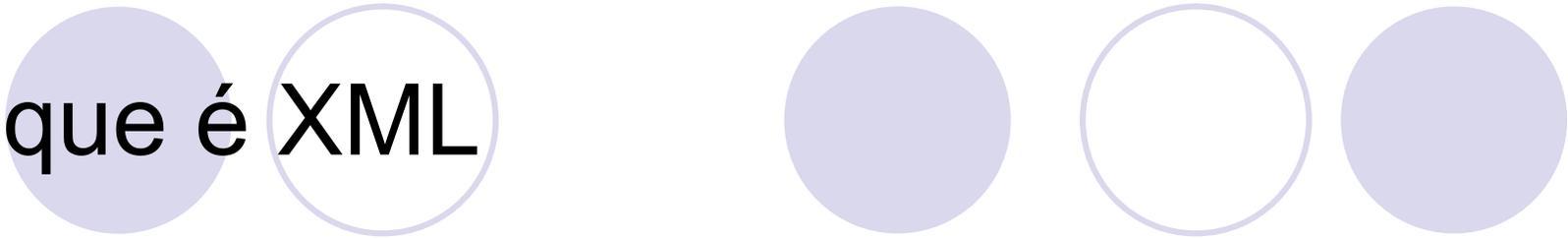
Quais plataformas de Web Services podem ser usadas ?

- IBM WebSphere Studio Site Developer
- IBM WebSphere Studio Integration Edition
- IBM ETTK
- Microsoft .NET
- Microsoft .NET SDK
- Oracle .NOW Dynamic Web Services
- Sun Studio ONE
- Sun Java Web Services Development Pack
- Systinet Developer
- The Mind Electric Glue

Quais Servidores de Aplicação se pode usar ... ?

- WebSphere AS (IBM)
- BEA
- Sun
- Oracle
- J2EE
- JBOSS

O que é XML

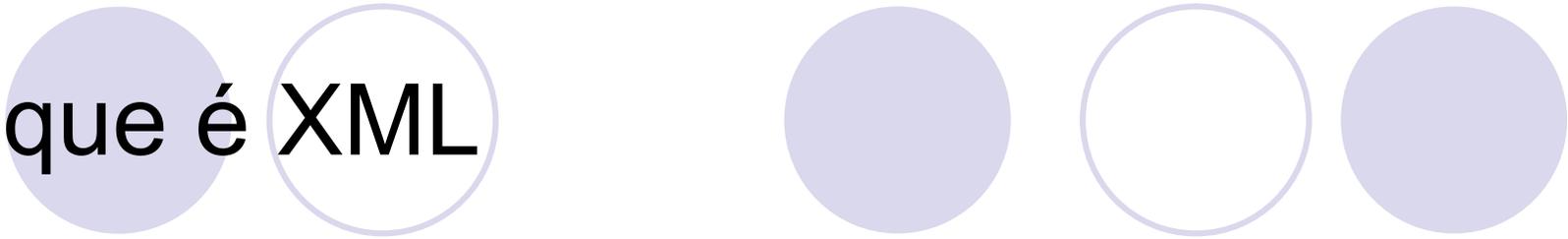


- XML é uma sigla. É acrônimo de **eXtensible Markup Language**.
- É uma linguagem com a finalidade de **descrever informações**.
- XML representa um aperfeiçoamento da abordagem da HTML.
- O propósito original de sua existência é a **WWW**.

O que é XML

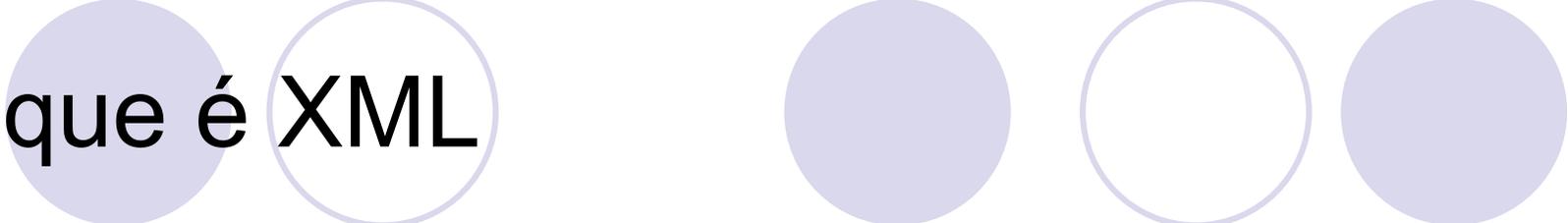
- XML não é uma “HTML mais rápida”.
- XML libera informações das “algemas” de um conjunto de tags fixo, como existe em HTML.
- Por exemplo, se você estiver descrevendo uma fatura, por que não chamá-la de `<Invoice>` em vez de um título de nível 1 ?

O que é XML

A decorative graphic at the top of the slide consists of a horizontal row of five circles. The first circle is solid light purple. The second circle is white with a light purple outline. The third circle is solid light purple. The fourth circle is white with a light purple outline. The fifth circle is solid light purple.

- XML é uma linguagem que oferece uma abordagem padrão para descrição, captura, processamento e publicação de informações.

O que é XML

A decorative graphic consisting of six circles arranged in a horizontal line. The first circle is solid light purple. The second circle is white with a light purple outline. The third circle is solid light purple. The fourth circle is white with a light purple outline. The fifth circle is white with a light purple outline. The sixth circle is solid light purple.

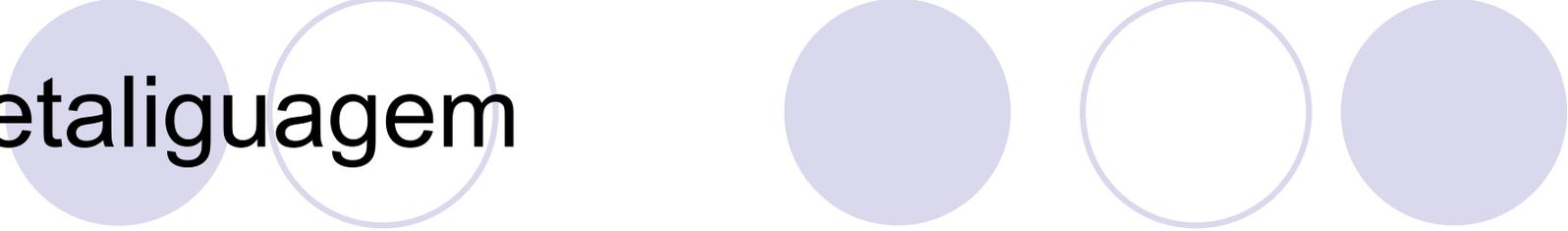
- **HTML**, por exemplo, possui um conjunto fixado de *tags* com o qual podemos elaborar documentos.
- **XML**, por outro lado, **não define** qualquer conjunto de *tags* em particular.

Domínios de Aplicação



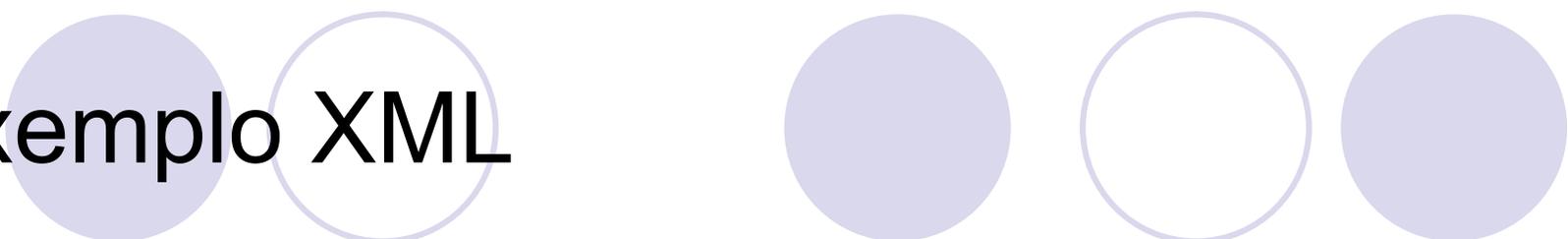
- Diversas **linguagens de marcação** podem ser construídas com base no que a XML oferece.

Metalinguagem

A decorative graphic consisting of six circles arranged in two rows. The top row has three circles: a solid light purple circle, a hollow light purple circle, and a solid light purple circle. The bottom row has three circles: a solid light purple circle, a hollow light purple circle, and a solid light purple circle.

- XML proporciona uma **estrutura padrão** que lhe possibilita criar sua própria estrutura para seus documentos (XML Schemas), por que ela é uma **metalinguagem**.

Exemplo XML



```
<! -- XML document -- >
```

```
<PCforSale>
```

```
  <Maker>Acme PC Inc</Maker>
```

```
  <Model>Blaster 555</Model>
```

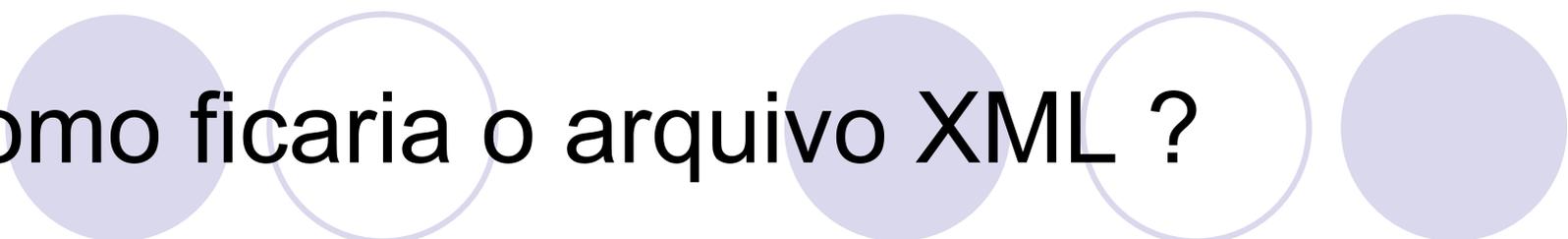
```
  <Storage>
```

```
    <RAM Units = "MB">128</RAM>
```

```
    <HardDisk Units = "GB">20</HardDisk>
```

```
  </Storage>
```

```
</PcforSale>
```



Como ficaria o arquivo XML ?

- **Book Description**

- *Title:*

- Author:*

- Binding type:*

- Number of pages:*

- Price:*



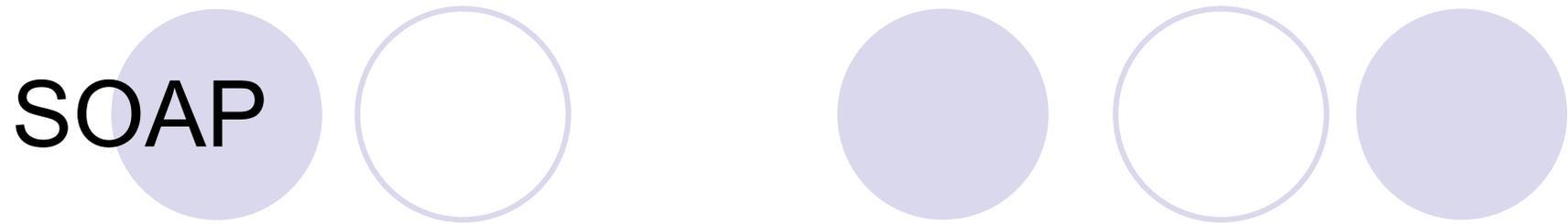
SOAP

- SOAP é para **comunicação entre aplicações.**
- SOAP é um simples protocolo baseado em XML para permitir aplicações trocarem informação sobre HTTP ou outro protocolo.



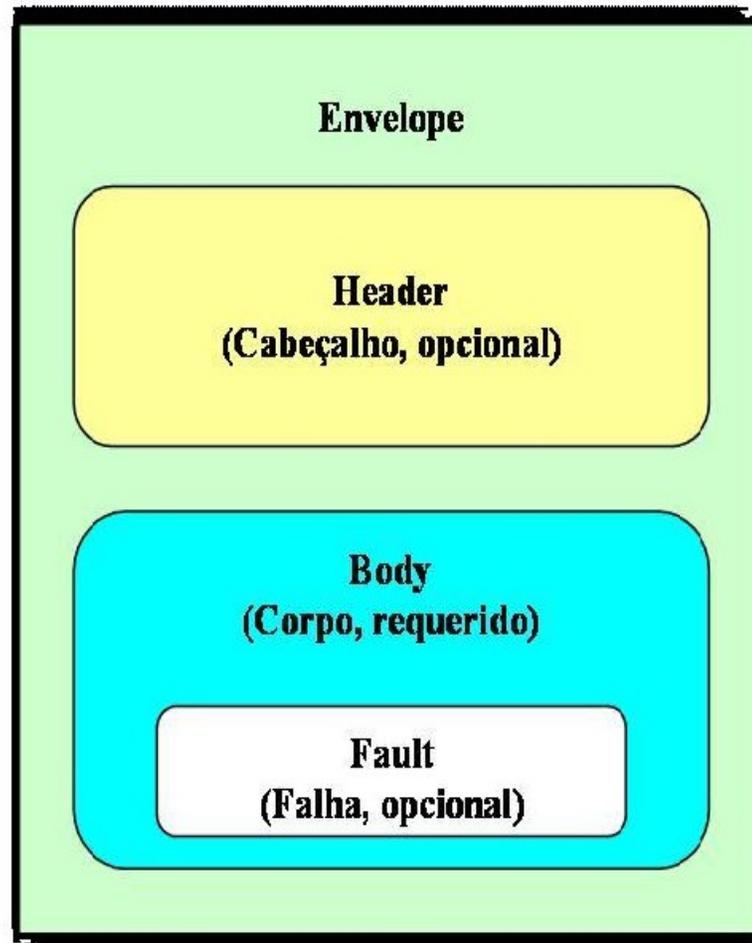
SOAP

- SOAP é projetado para a Internet.
- SOAP é independente de plataforma.



- SOAP permite passar por **firewalls**.
- SOAP será desenvolvido como um **padrão W3C**.

SOAP



O Cliente: uma solicitação SOAP

- `<?xml version='1.0' encoding='ISO8859-1' ?>`
- `<SOAP-ENV: Envelope`
- `xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"`
- `xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"`
- `xmlns:xsd="http://www.w3.org/2001/XMLSchema">`
- `<SOAP-ENV:Body>`
- `<ns1:getCEP>`
- `xmlns:ns1="urn = correios-cep"`
- `SOAP-ENV:encodingStyle=`
- `"http://schemas.xmlsoap.org/soap/encoding"`
- `>`
- `<end-postal>Rodovia SC-401</end-postal>`
- `</ns1:getCEP>`
- `</SOAP-ENV:Body>`
- `</SOAP-ENV: Envelope>`

O Servidor: a Resposta SOAP

- `<?xml version='1.0' encoding='ISO8859-1' ?>`
-
- `<SOAP-ENV: Envelope`
- `xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"`
- `xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"`
- `xmlns:xsd="http://www.w3.org/2001/XMLSchema">`
- `<SOAP-ENV:Body>`
- `<ns1:getCEPResponse`
- `xmlns:ns1="urn=Correios-CEP"`
- `SOAP-`
- `ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding" >`
- `<returnCEP xsi:type="xsd:string">88055</returnCEP>`
- `</ns1:getCEPResponse`
- `</SOAP-ENV:Body>`
- `</SOAP-ENV: Envelope>`

WSDL

<definitions>: Root WSDL Element

<types>: What data types will be transmitted?

<message>: What messages will be transmitted?

<portType>: What operations (functions) will be supported?

<binding>: How will the messages be transmitted on the wire?
What SOAP-specific details are there?

<service>: Where is the service located?

UDDI

Universal Description, Discovery and Integration

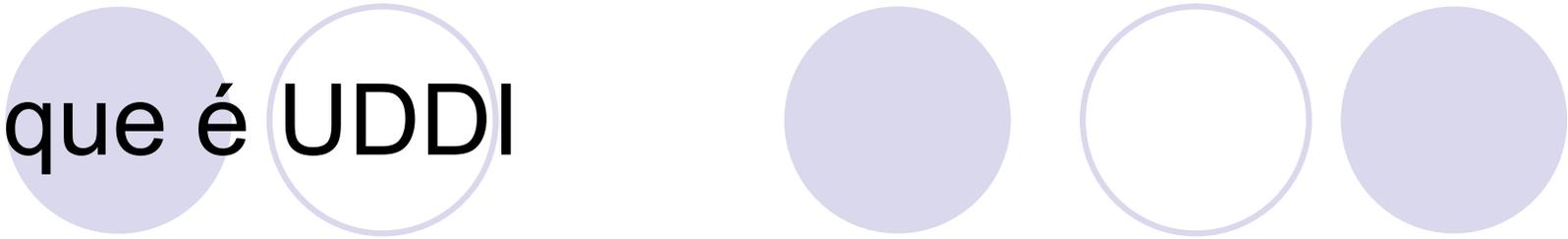
● Origem UDDI

- Criado em parceria com Ariba, IBM e Microsoft
- Versão 1.0 lançada em Setembro 2000

● Objetivo

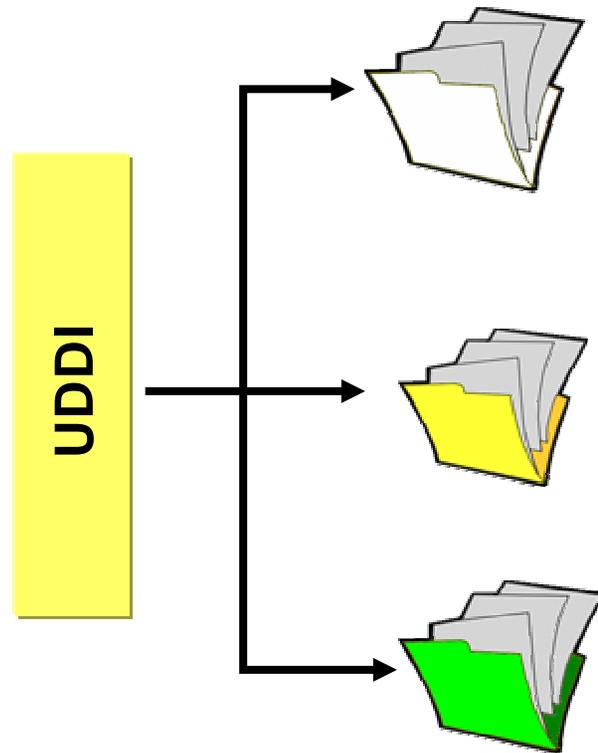
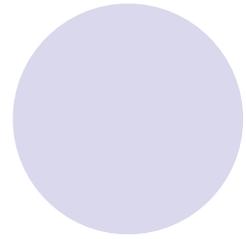
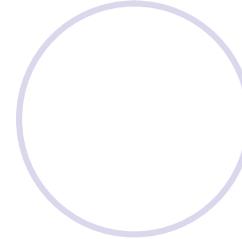
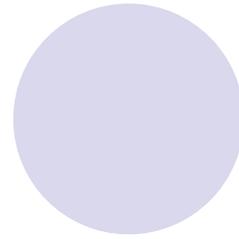
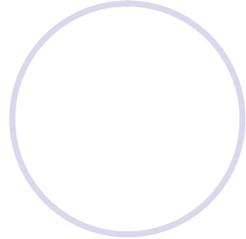
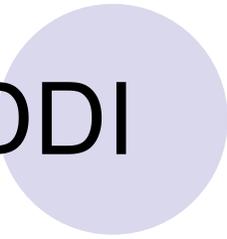
Publicação e Busca de serviços na Web.

O que é UDDI



- Um arquivo XML utilizado para descrever uma entidade de negócio e seus serviços.
- Um diretório universal de registro de Web Services, onde seriam encontradas informações sobre as empresas fornecedoras de serviços.

UDDI

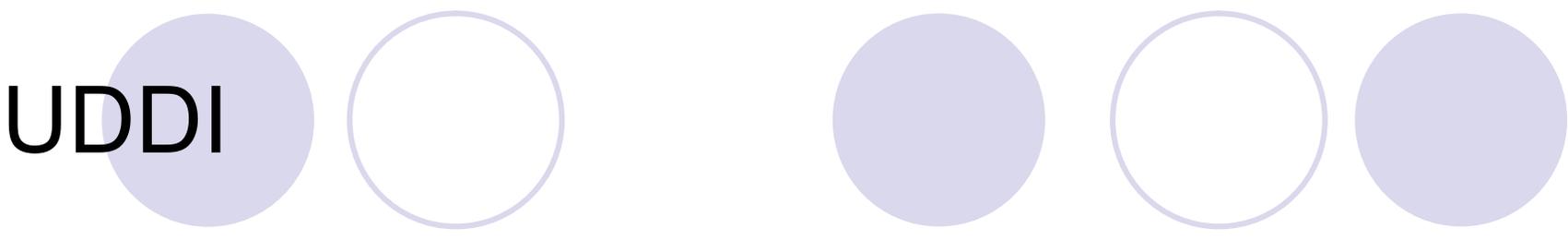




UDDI

○ Páginas brancas:

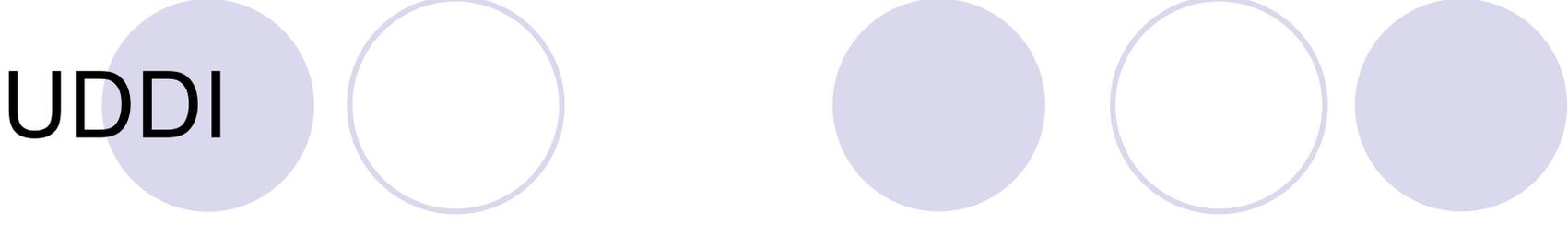
Contêm o endereço, pessoas de contato e outros identificadores relativos ao **negócio** onde a empresa atua;



UDDI

○ Páginas amarelas:

Incluem categorias industriais baseados na organização do serviço;

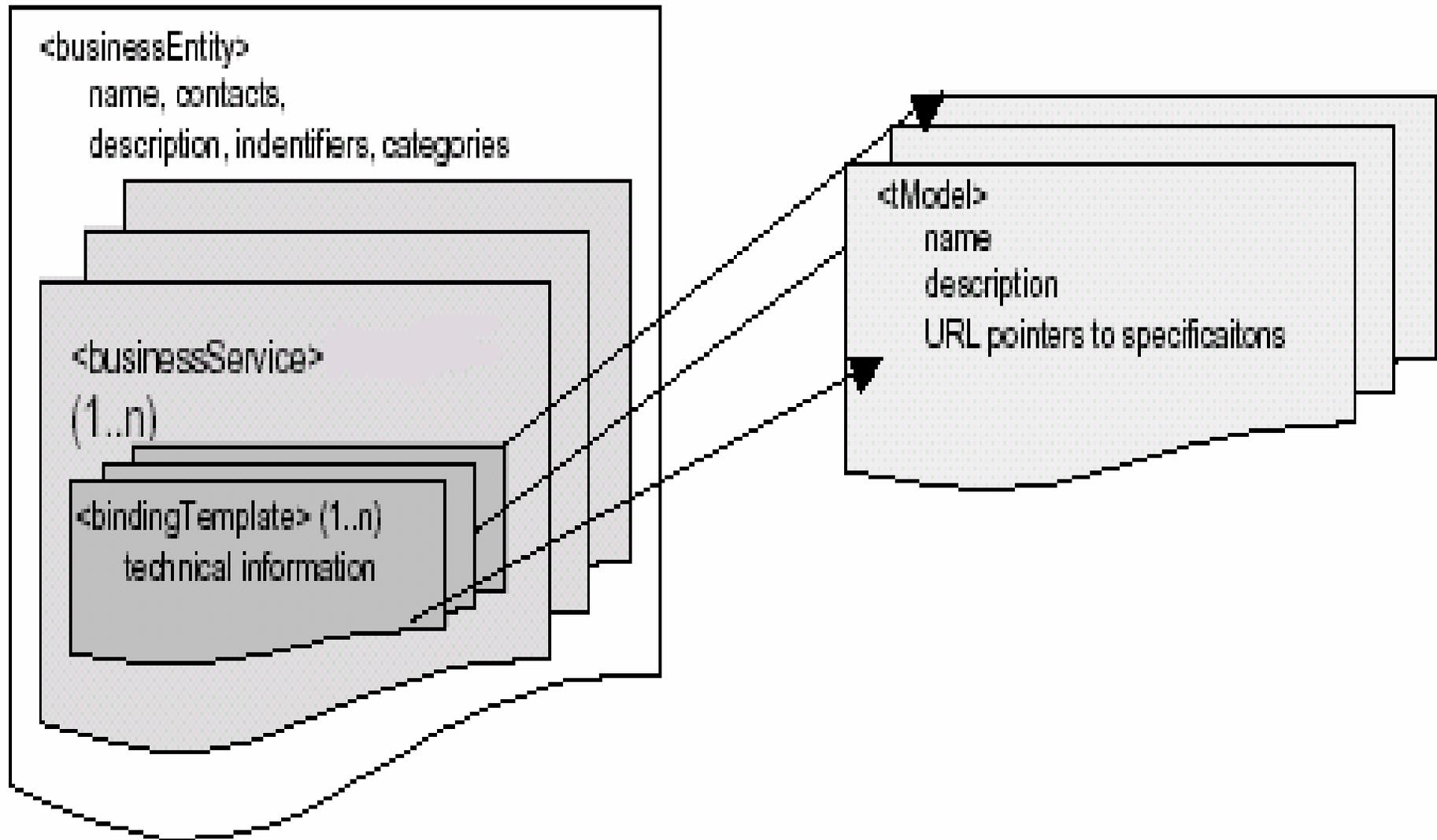


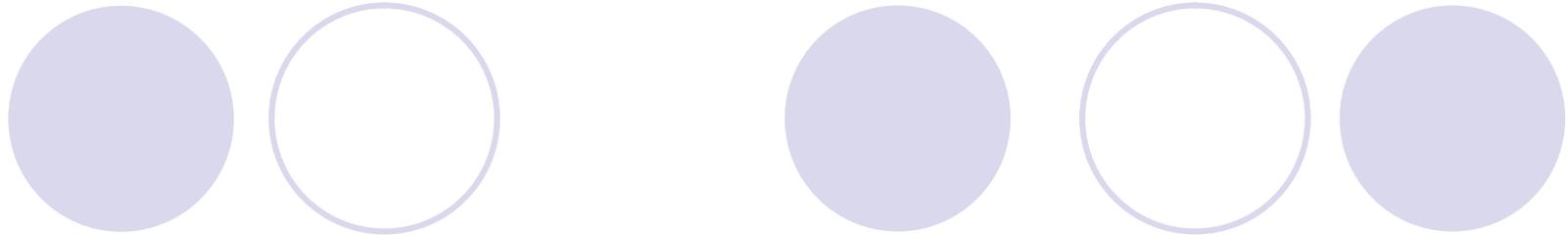
UDDI

○ **Páginas verdes:**

Contêm **informações técnicas** sobre os **serviços** expostos pelo provedor.

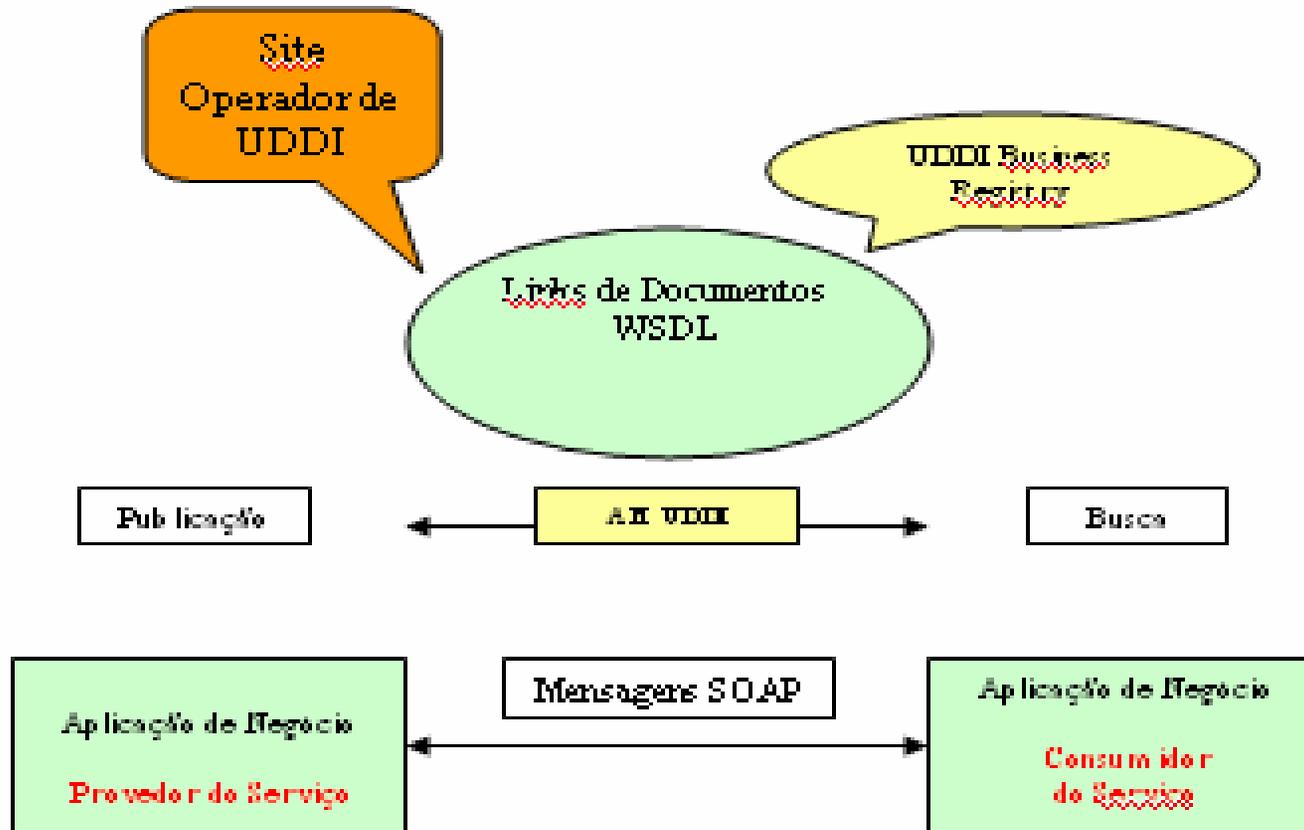
O Modelo de Informação do UDDI

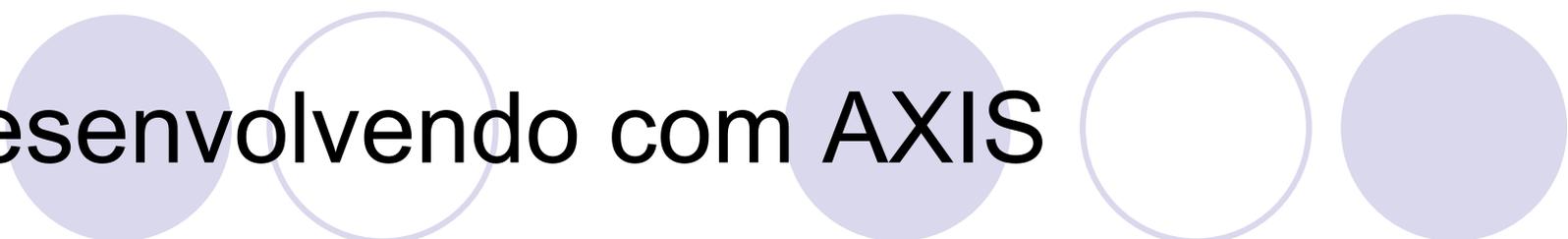




- businessEntity: Informações sobre uma família de serviços.
- businessService: Informações sobre um serviço particular.
- bindingTemplate: Informações técnicas sobre um serviço.
- tModel: Descrições específicas do serviço.

Arquitetura Orientada-a-Serviço e UDDI





Desenvolvendo com AXIS

- Uso da Plataforma

AXIS: [AXIS.rar](#) ou
[AXIS.doc](#) (Informações sobre Instalação
e Exemplo de Aplicação).

Plataforma : [AXIS 1.2rc3.zip](#)

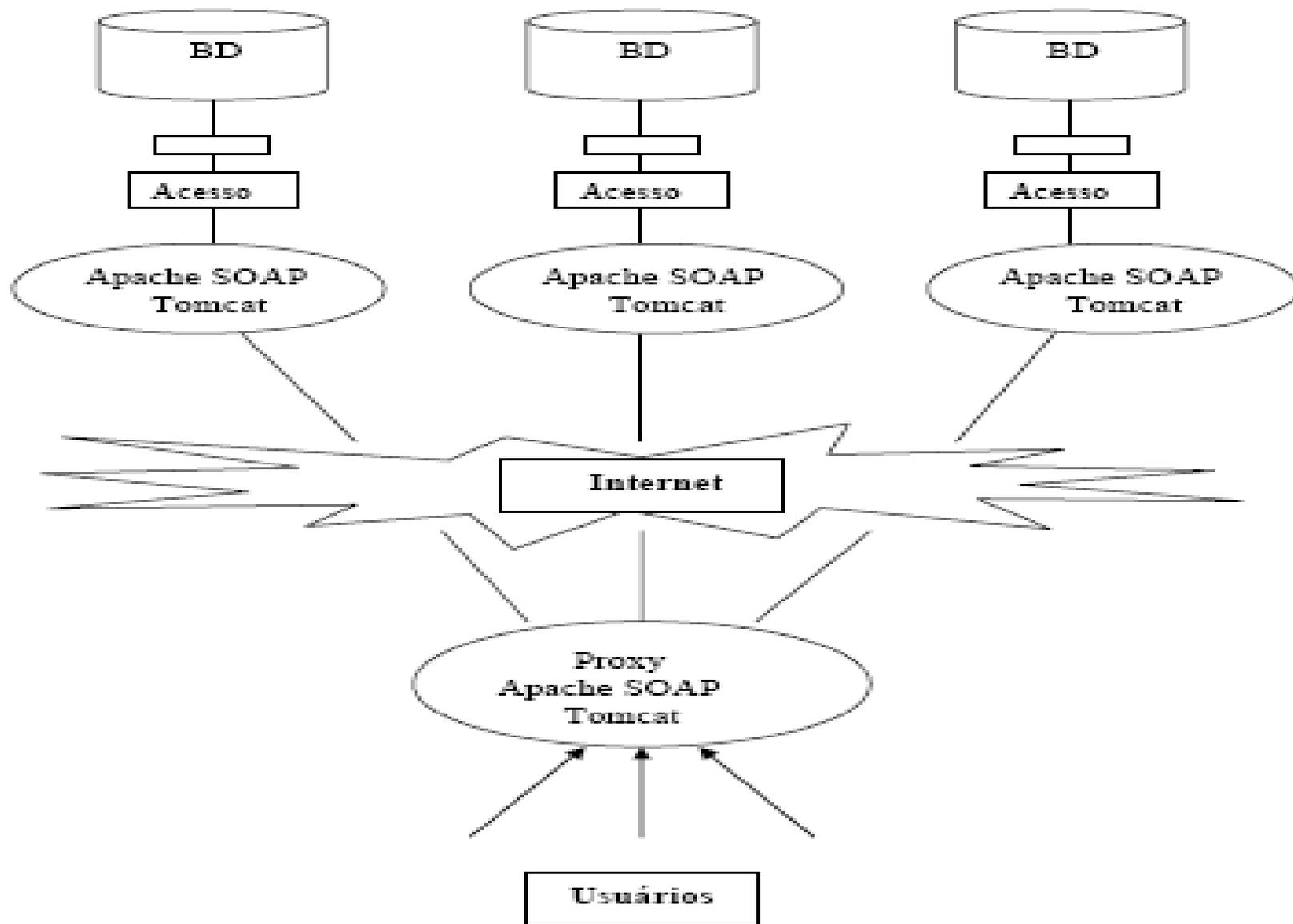
Tomcat : [Tomcat 5.0.28.zip](#)

Mais Informações

- <http://www.inf.ufsc.br/~bosco/ensino/ine5626.html>
- Conteúdo de disciplina
- Links e Ferramentas
- TCC sobre Avaliação de Plataformas: AXIS, JWSDP, ETTK e GLUE.
(Carlos Alberto Furtado e Maurício Botelho -
Curso de Graduação em Sistemas de Informação da UFSC)

Exemplo de um Serviço Web em AXIS

```
● package app.ws.calc;
● public class Calculadora implements Calculadora {
●     /** Creates a new instance of Calculadora */
●     public int getSoma(int i1, int i2) {
●         return i1 + i2;
●     } //getSoma
●     //-----
●     public int getSub(int i1, int i2) {
●         return i1 - i2;
●     } // getSub
●     //-----
●     public int getDiv(int i1, int i2) {
●         return i1/i2;
●     } // getDiv
●     //-----
●     public int getMult(int i1, int i2) {
●         return i1*i2;
●     } //getMult
● } //Calculadora
```



- 
- 1. XML - Step by Step, Michael Young, Microsoft Editora, 2000.**
 - 2. XML, Aplicações Práticas, Sean McGrath, Editora Campus, 1998.**
 - 3. Programming Web Services with XML-RPC, Simon St. Laurent, Joe Johnston e Edd Dumbill, O'Reilly, 2001.**
 - 4. SOAP: Cross-Platform Web Service Development Using XML, Scott Seely, PH PTR, 2002.**
 - 5. Understanding Web Services, XML, WSDL, SOAP e UDDI, Eric Newcomer, Addison Wesley, 2002.**
 - 6. Web Services Essentials, Ethan Cerami, O'Reilly, 2002.**
 - 7. Professional Java Web Services, Mack Hendricks at al. Alta Books/Wrox, 2002.**