

# Estudo de caso:



**Café Expresso Ltda.**

A estória de João...



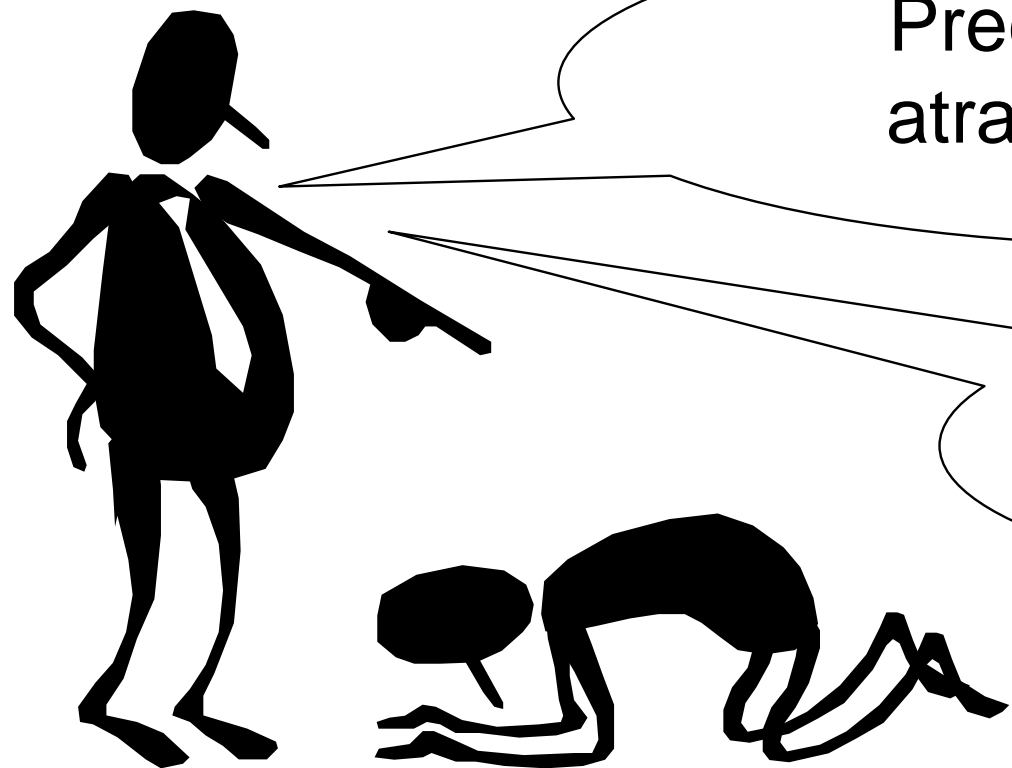
# Café Expresso Ltda.

- Características:
  - Sistema de controle de vendas de café:
    - Implementado em Delphi
    - Interface Windows p/ Desktop
    - Possui um SGBD



# Café Expresso Ltda.

Um dia o chefe de João traz novos requisitos...



Preciso vender café  
através da Internet!!

**Estou perdendo  
dinheiro! Quero isso  
PARA ONTEM!**



# Café Expresso Ltda.

João pensa...



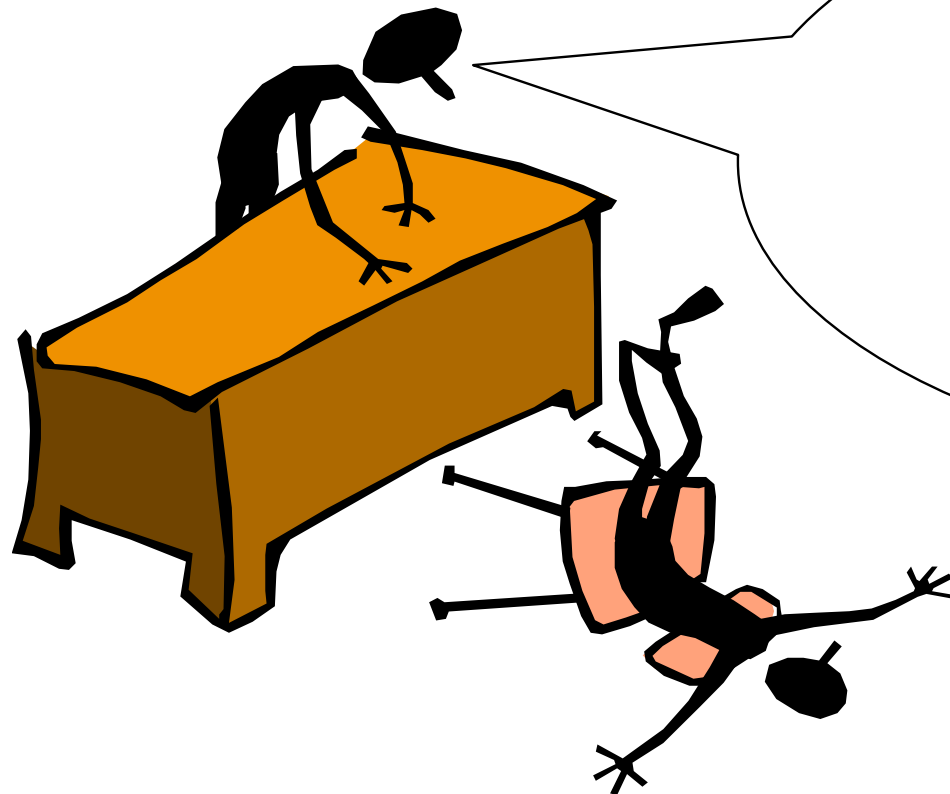
Como vou integrar o sistema existente com um novo módulo de acesso à Internet?

Terei que reescrever o sistema em Java? ASP?



# Café Expresso Ltda.

Enquanto isso, o chefe traz mais requisitos...



Meus concorrentes estão vendendo mais variedades de café do que eu! Preciso integrar nosso sistema de venda pela internet com os sistemas de meus fornecedores, e de novos fornecedores, para aumentar meus lucros!!!

**PARA ONTEM!!!**



# Café Expresso Ltda.

João desesperado...

Já estou  
reescrevendo o  
sistema em  
Java...

Pelo menos o  
sistema dos  
fornecedores é  
escrito em  
Java??





# Café Expresso Ltda.

O chefe esclarece...

Um dos  
fornecedores tem  
o sistema em  
Delphi...

... o outro em C++ ...

... o outro em  
Visual Basic ...

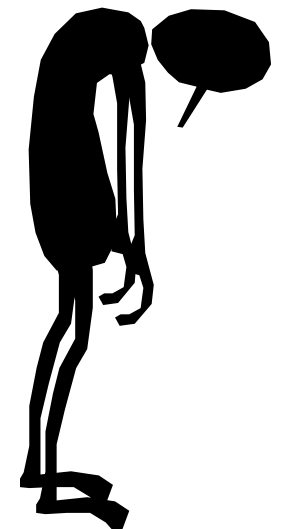
Preciso disso  
**PRA ONTEM!**





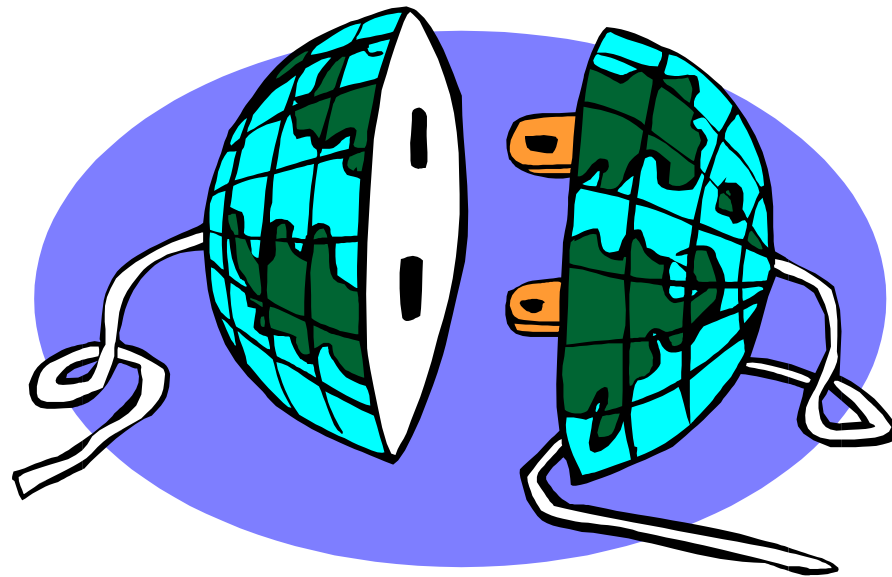
# Café Expresso Ltda.

Precisamos ajudar João...



# *Web Services*

## *Parte 1 – Introdução*

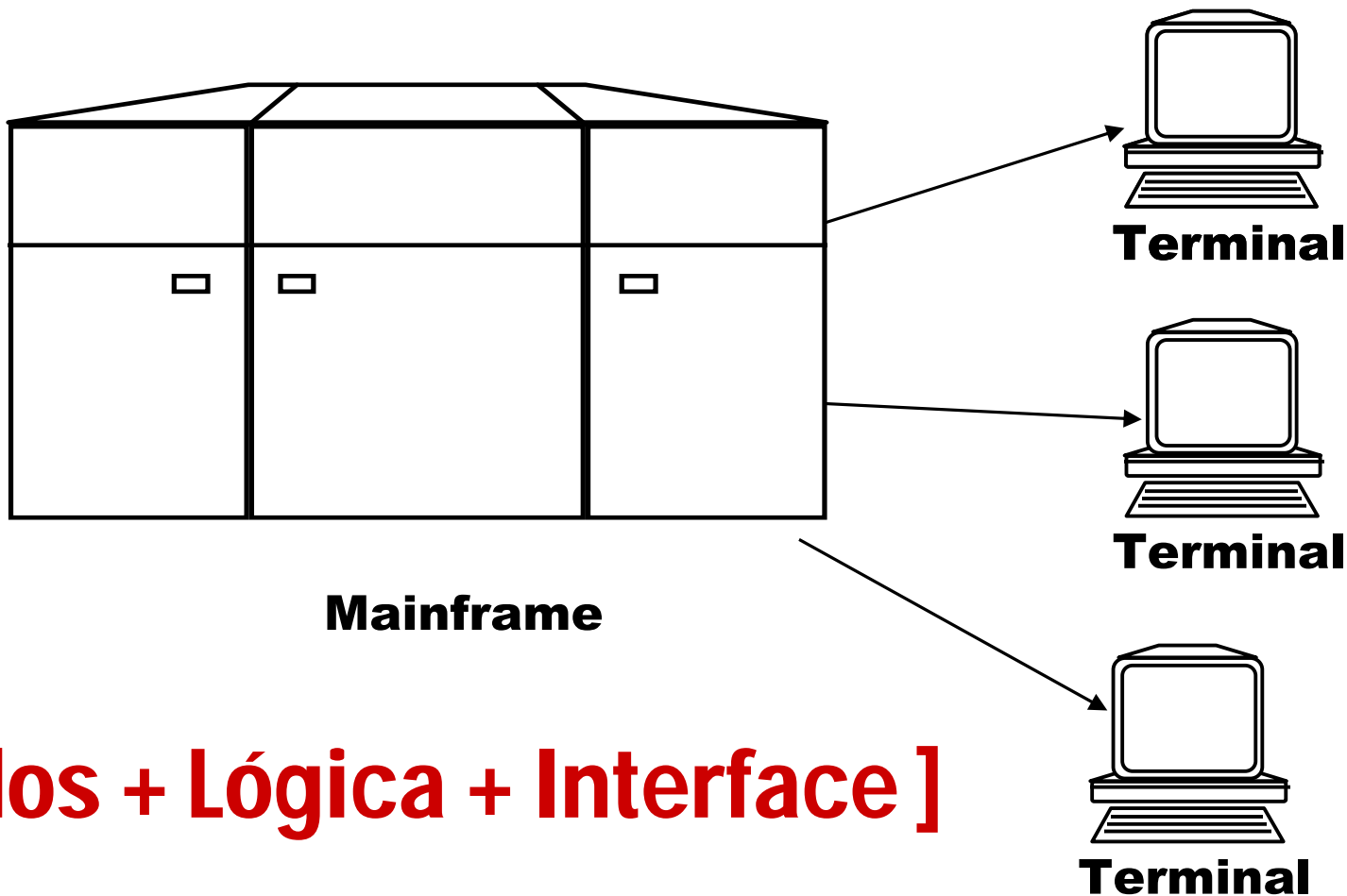


Rodrigo Rebouças de Almeida  
rodrigor@rodrigor.com

*4a Reunião PBJUG*

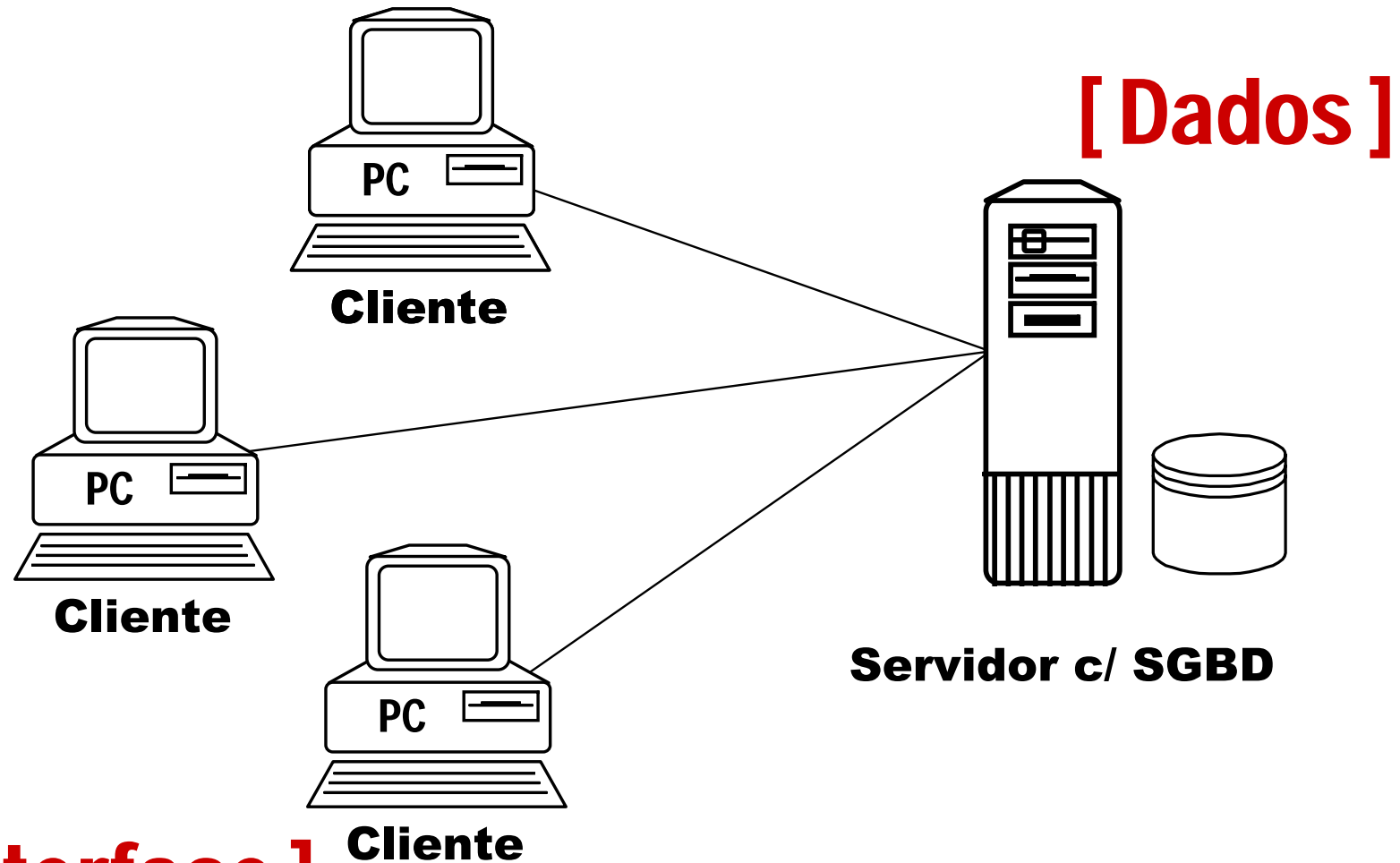
# Arquiteturas - Histórico

## ■ MainFrames



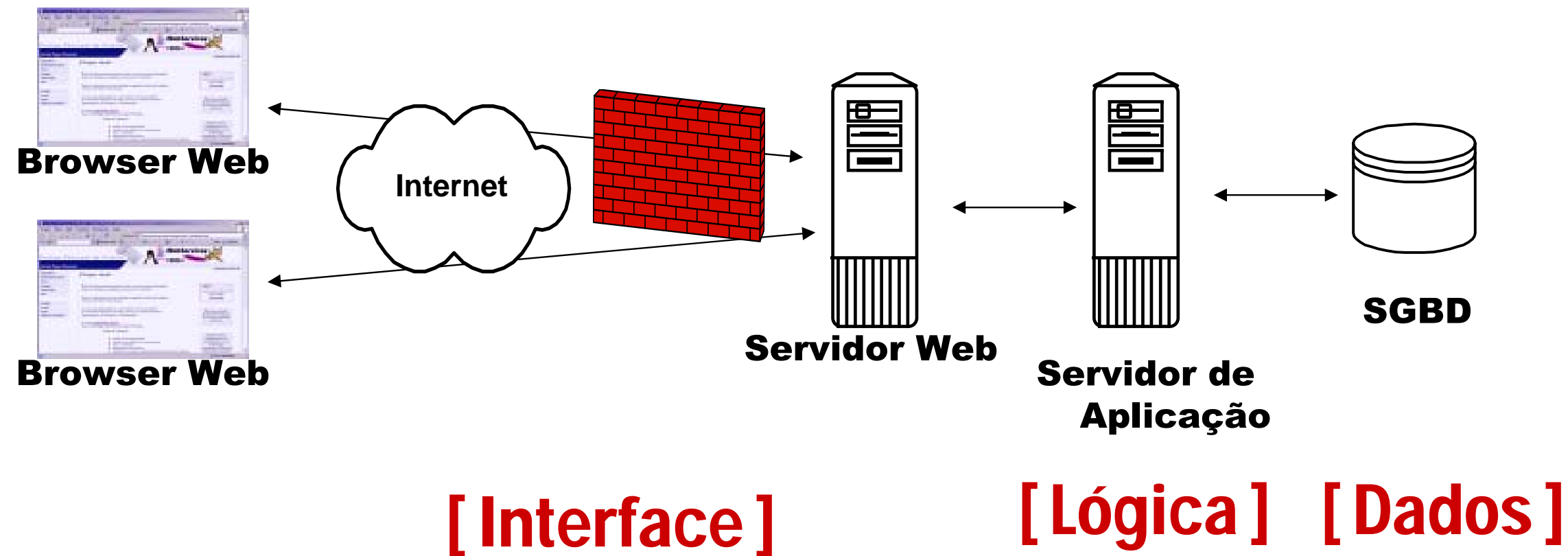
# Arquiteturas - Histórico

## ■ Arquitetura Cliente-Servidor



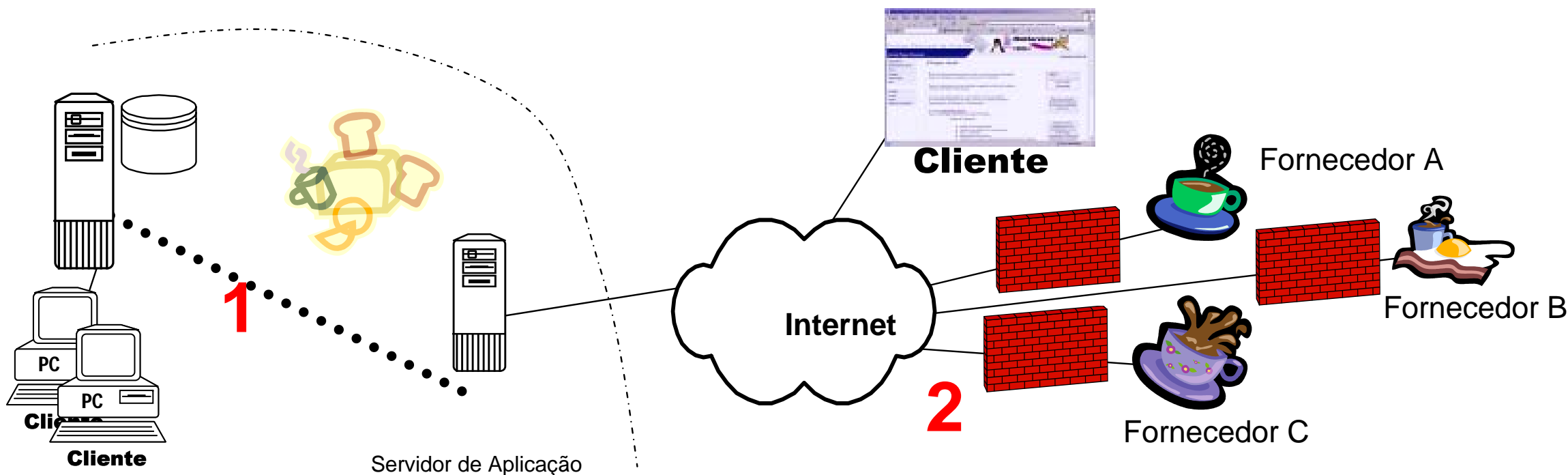
# Arquiteturas - Histórico

- Arquitetura em n-camadas



# O Problema de João...

1. Integrar o sistema legado com um novo sistema
2. Integrar o sistema da empresa com os sistemas dos fornecedores A, B, C



Vamos tentar resolver o problema de João usando Web services.

# O que é um Web Service?

- Um Web Service é um ponto de acesso a funcionalidade que pode ser:
  - Localizado dinamicamente
  - Ter sua interface descoberta automaticamente, porque o serviço sabe se descrever
  - Ser chamado na Web

# Web services

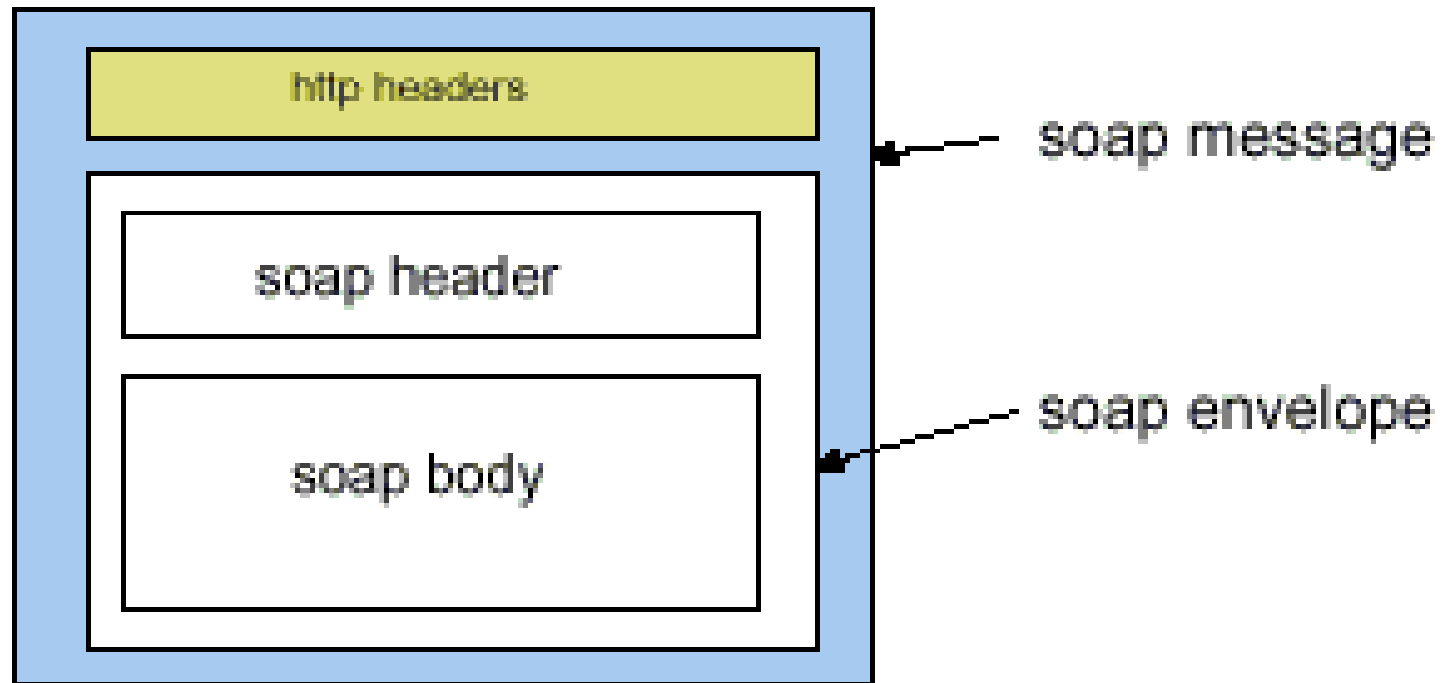
- Se fundamenta basicamente em três tecnologias:
  - Web Services Description Language (WSDL)
    - Define a interface de acesso ao serviço
  - Simple Object Access Protocol (SOAP)
    - Um protocolo baseado em XML. Permite que os clientes se comuniquem com os provedores de serviço.
  - Universal Description, Discovery and Integration (UDDI)
    - Permite Web services registrarem suas características, permitindo que outras aplicações as encontrem.

# Usar XML para toda a comunicação

- Extended Markup Language (XML)
- Oferece um formato ASCII para trocar qualquer tipo de informação estruturada
- Usa o “estilo” HTML de *markup* com tags
  - ```
< Pessoa nome="João">  
  < frutasFavoritas>  
    < fruta>Manga</fruta>  
    < fruta>Maçã</fruta>  
    < fruta>Uva</fruta>  
  </frutasFavoritas>  
</ Pessoa>
```
- Os tags podem ser definidos para criar uma “Aplicação XML” ou “Linguagem XML”

# Usar SOAP fazer RPC

- SOAP é uma forma de fazer Remote Procedure Call (RPC) usando documentos XML



# Descrever o serviço remoto usando WSDL

- WSDL = Web Services Description Language
- Pronunciado “wisdle”
- É uma linguagem XML que contém informação sobre a interface, a semântica, e outros detalhes de chamadas a um Web Service

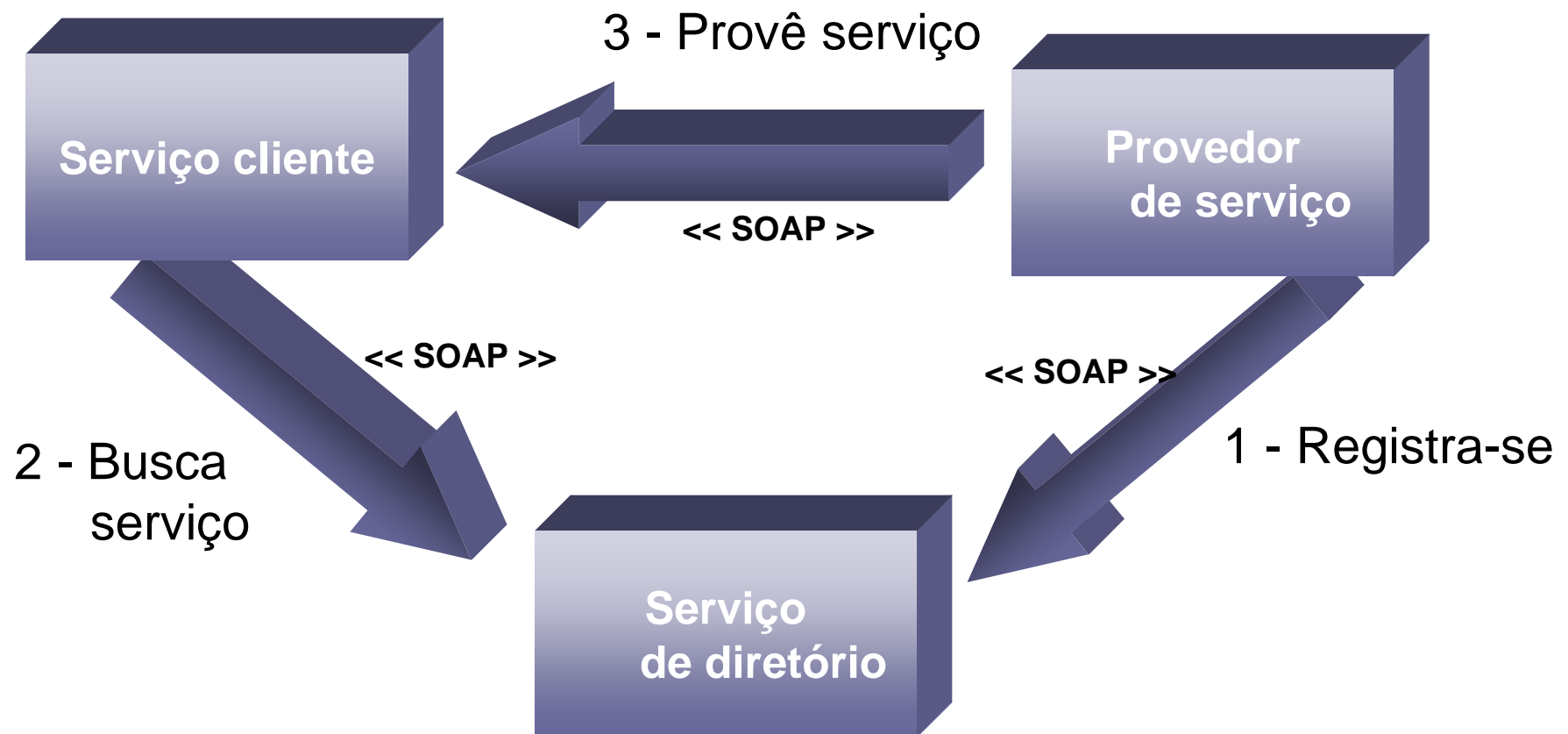
# Definir ontologias

- Precisamos saber o que as coisas significam
- Exemplo: como mandar uma ordem de serviço se não sei o que cada campo deve conter?
- Várias organizações definem e padronizam ontologias
  - ebXML
  - RosettaNet
  - (...)

# Usar UDDI para localizar serviços

- UDDI = Universal Description, Discovery, and Integration
- Permite cadastrar serviços e localizá-los
- Não é necessário usar UDDI se o cliente já tiver o documento WSDL

# Web services, Arquitetura



# SOAP

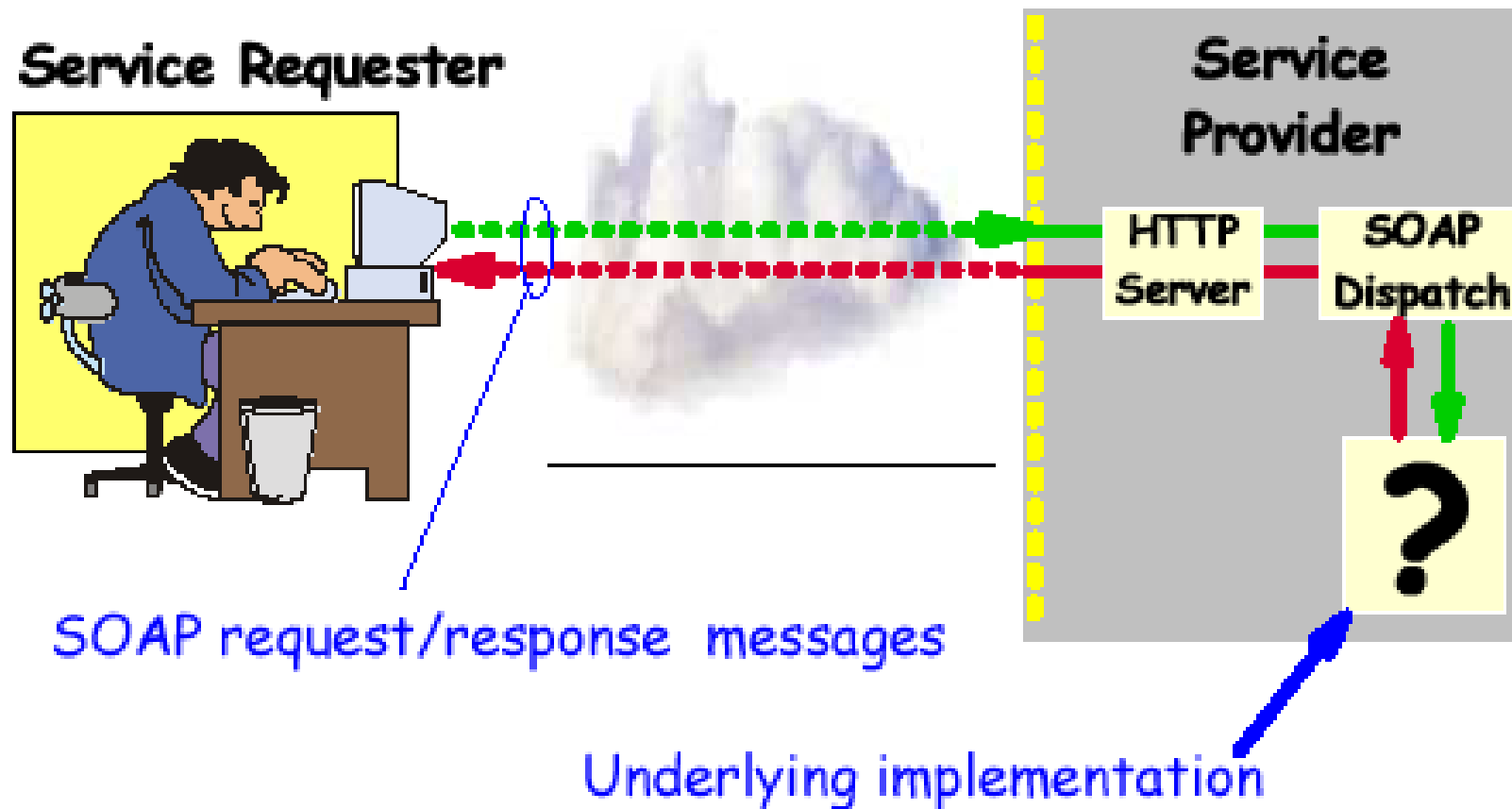
- Simple Object Access Protocol
  - Define uma forma de transferir documentos XML através da Internet;
  - Aplicações cliente se comunicam com Web services usando SOAP como protocolo;
  - SOAP define um envelope para transmissão de mensagens, oferece regras para representar "remote procedure calls" (RPCs);

# SOAP

- Está se tornando um padrão para troca de mensagens usando XML;
- Desenvolvido sob a supervisão da W3C;
- Permite a transferência de mensagens através de protocolos de alto nível, como HTTP e SMTP;
- É mais usado sobre HTTP, pois **consegue atravessar firewalls!**



# SOAP



# SOAP

## ■ Estrutura de uma mensagem SOAP:

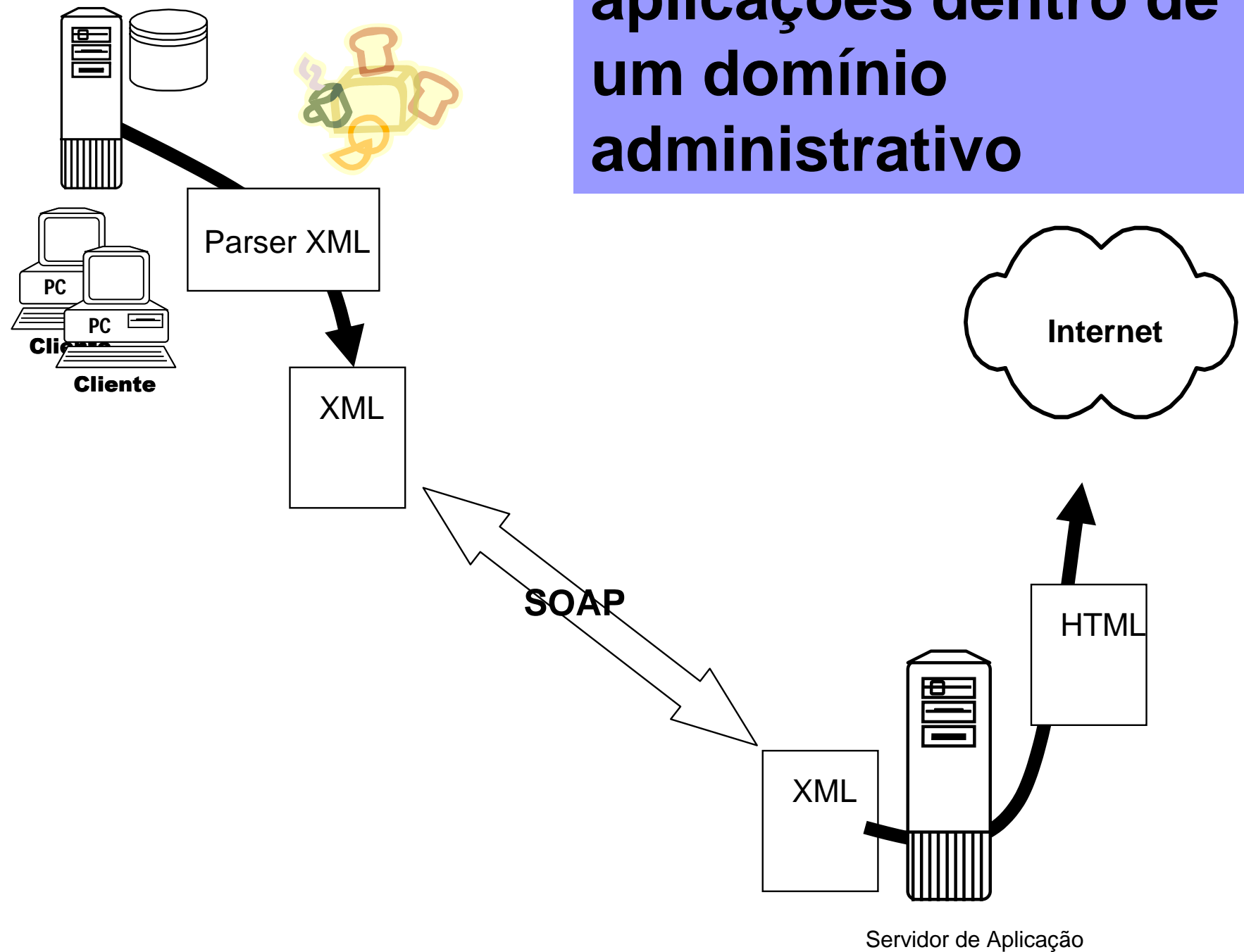
```
<SOAP:Envelope xmlns:SOAP= http://schemas.xmlsoap.org/soap/envelope/>  
  <SOAP:Header>  
    <!conteudo do cabeçalho >  
  </SOAP:Header>  
  <SOAP:Body>  
    <!conteudo do corpo>  
  </SOAP:Body>  
</SOAP:Envelope>
```

João agora não precisa  
reimplementar o sistema  
de controle de vendas de  
café para este ser  
acessível via Internet...

Basta fazer o sistema  
existente e o módulo de  
acesso a Internet  
conversarem usando  
SOAP e descrever os  
dados em XML



# Integração entre aplicações dentro de um domínio administrativo



Resolvemos o 1o problema,  
publicamos o sistema existente na  
Internet, agora precisamos ligar  
nosso sistema aos sistemas dos  
nossos fornecedores!

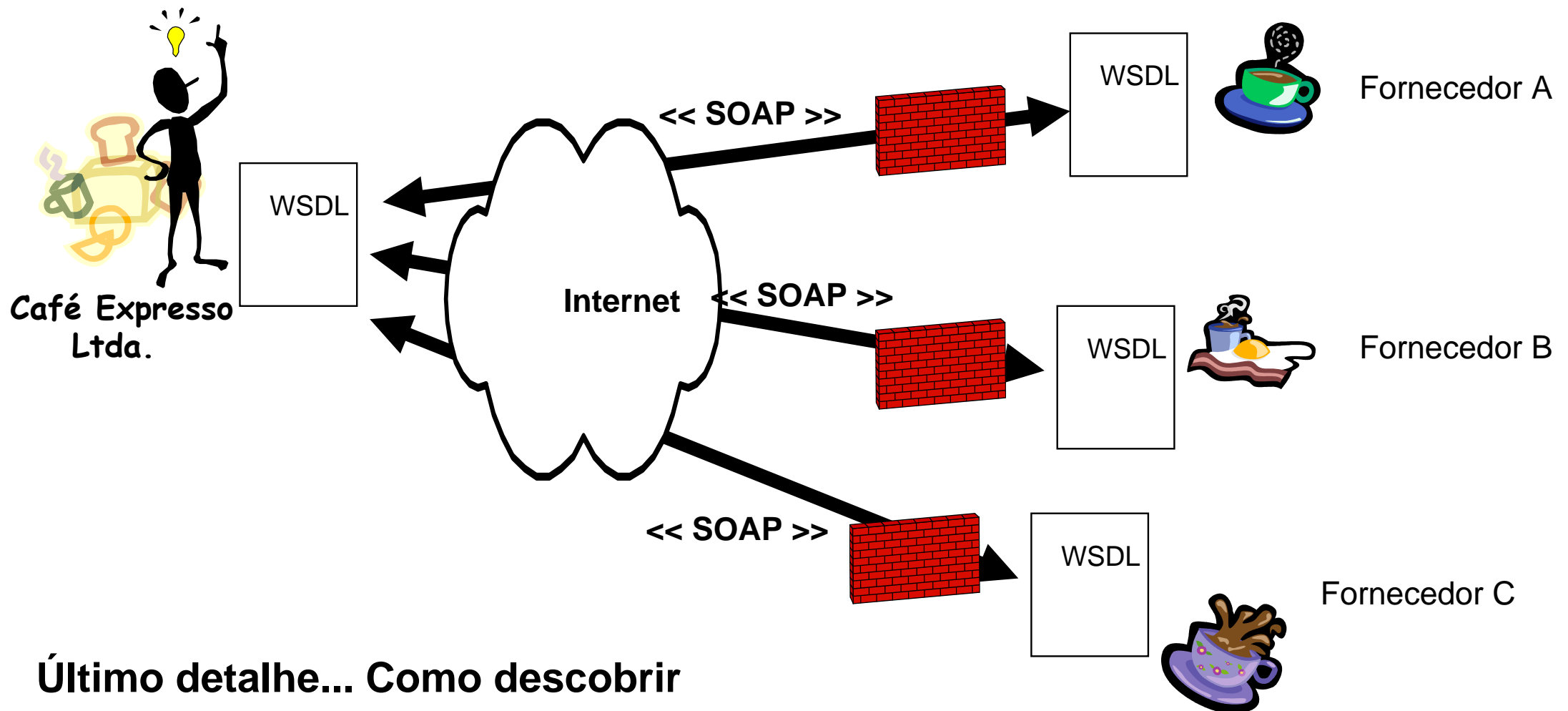
Podemos usar SOAP como  
protocolo de comunicação! Mas,  
como saber a interface do  
sistema do meu fornecedor?  
Como posso fazer meu sistema  
interagir com um sistema escrito  
em outra linguagem, em outra  
plataforma?



# WSDL

- Web Service Description Language
  - É uma linguagem XML para descrever Web services;
  - Segue um conceito parecido da “Interface Definition Language” (IDL) usado em CORBA;
  - Descreve detalhes como definição de tipos de dados, operações suportadas pelo Web service, formato das mensagens, endereço de rede do Web service, etc.

# Resolvido o problema, agora João tem seu Sistema integrado com os fornecedores.



**Último detalhe... Como descobrir novos fornecedores??**

# UDDI

- Universal Description, Discovery and Integration
- Provê um mecanismo para registro e localização de Web services;
- É um diretório que contém vários Web services registrados, onde associamos um nome a um serviço, como um Web service.

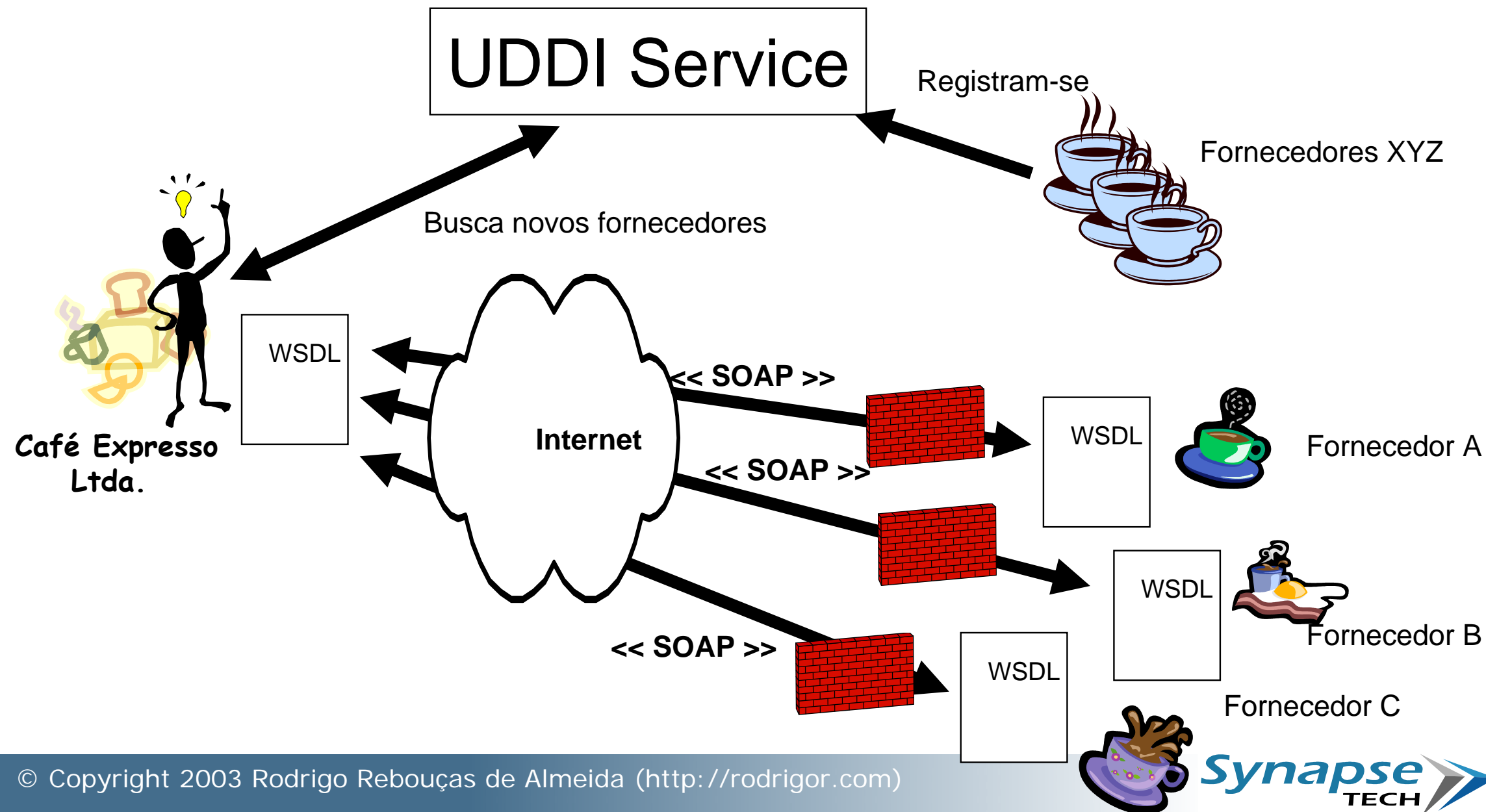
# UDDI

- Permite que empresas registrem seus serviços e possam interagir com outras empresas interessadas.
- UDDI é um Webservice baseado em XML e SOAP;
- A interação com um serviço UDDI é baseado em interfaces SOAP pré-definidas.

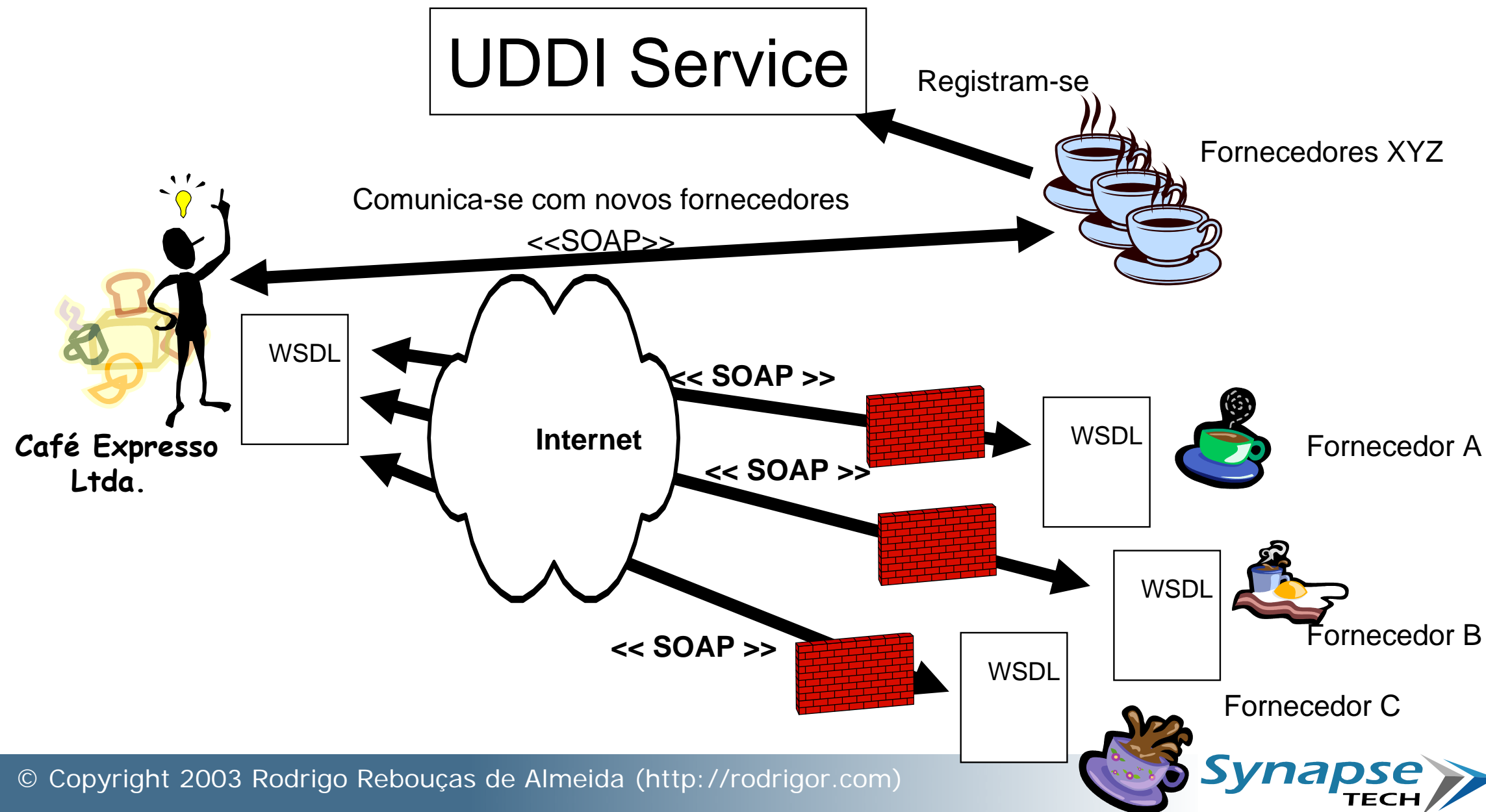
# UDDI

- Um registro UDDI é organizado em duas entidades fundamentais:
  - `businessEntity` : provê informações a sobre a empresa que está publicando o serviço. Contém um ou mais elementos `businessService`;
  - `businessService`: representa um serviço provido pela `businessEntity`;
  - Cada entidade é identificada por uma chave única, permitindo que haja referências a serviços, definições WSDL;

# João agora pode encontrar outros Fornecedores através de um serviço UDDI.



# João agora pode encontrar outros Fornecedores através de um serviço UDDI.



# Ferramentas - Como desenvolver Web Services?

- Tem ferramentas da Sun
  - JWSDP com muitas APIs:
    - JAXP, JAXB, JAX-RPC, SAAJ, JAXR, JWSDL
- Tem ferramentas da IBM
- Tem ferramentas da Microsoft (.NET)
- Tem ferramentas de outros lugares
  - Axis do projeto Apache
    - WSDL2Java converte WSDL para Java
  - SOAP::Lite para Perl
  - etc.

# Conclusões

- João não foi despedido



João com seu salário

# Conclusões

- O chefe de João conseguiu barrar a concorrência..



# Fontes de consulta

- <http://java.sun.com/webservices>
- [developers.ibm.com/webservices](http://developers.ibm.com/webservices)
- JavaWorld: [www.javaworld.com](http://www.javaworld.com)
- [webservices.org](http://webservices.org)
- [www.uddi.org](http://www.uddi.org)
- [xml.apache.org](http://xml.apache.org)

# Obrigado!!

Rodrigo Rebouças de Almeida

[rodrigor@rodrigor.com](mailto:rodrigor@rodrigor.com)

<http://rodrigor.com>