

The logo for CPQ, featuring the letters 'CPQ' in a bold, stylized, white font. The 'C' and 'P' are connected, and the 'Q' has a distinctive shape with a small tail.

CPQ

The text '35 anos' in a white, cursive script font, with a horizontal line underneath the '35' and 'anos'.

35 anos

The background is a solid dark blue. It features a large, abstract, wavy shape that resembles a stylized wave or a series of overlapping spheres. This shape is composed of a fine, light blue grid of dots, giving it a textured, mesh-like appearance. A bright, horizontal light flare or lens flare effect cuts across the middle of the image, passing through the center of the wavy shape.

Desenvolvendo e Integrando
Serviços Multiplataforma de
TV Digital Interativa

Agenda

- Introdução
- Aplicações interativas de TV Digital
- Desafios de *layout* e usabilidade
 - Laboratório de usabilidade
- Desafios e problemas de codificação
- Componentes
- *Web Services*
- Arquitetura de referência
 - *Browser* de serviço e *Gateway* de Serviço
 - STID
 - Serviço de governo eletrônico
- Criação de interfaces de integração com Axis2
- Demonstração de aplicações interativas
- Conclusões finais

Sistema de transmissão de aplicações Ginga



- Demonstração

Cenário atual

- O Brasil está atualmente passando por uma fase de transição no sistema de transmissão de TV, da transmissão analógica para a digital.
- Teremos que passar por uma mudança de paradigma no uso da TV no país, através da crescente demanda por serviços de TV interativos.
- Haverá também a necessidade de integração entre diferentes formas de acesso. Além da TV, os serviços interativos serão acessados por celulares e computadores pela internet.
- Para a realização de integrações entre diferentes formas de acesso e serviços o CPqD escolheu a tecnologia de *Web Services*.
- Aplicações “ricas” serão disponibilizadas e a aplicação no lado da camada de negócio e persistência deverá ter atenção especial para suportar a demanda de acesso.

Aplicações interativas de TV Digital

- Serviço de governo eletrônico

Escolha uma opção

- 1 Marcar consulta médica
- 2 Consultas que já marquei
- 3 Informações de saúde



Heródoto Barbeiro



Sair



Selecionar



Ajuda

Aplicações interativas de TV Digital

- Serviço de conteúdo sob demanda – Notícias



Selecione um assunto para ver as notícias

Amazônia	Música
Brasil	Planeta Bizarro
Carros	Política
Ciência e Saúde	Pop & Arte
Cinema	Quadrinhos
Concursos e Emprego	Rio de Janeiro
Economia e Negócios	São Paulo
Games	Tecnologia
Mundo	Vestibular e Educação



Sair



Selecionar

Aplicações interativas de TV Digital

- Serviço de conteúdo sob demanda – Clima



Aplicações interativas de TV Digital

- Serviço de conteúdo sob demanda – Chat

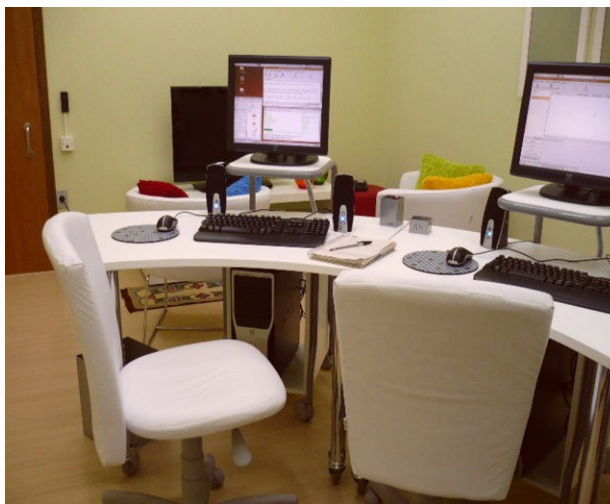
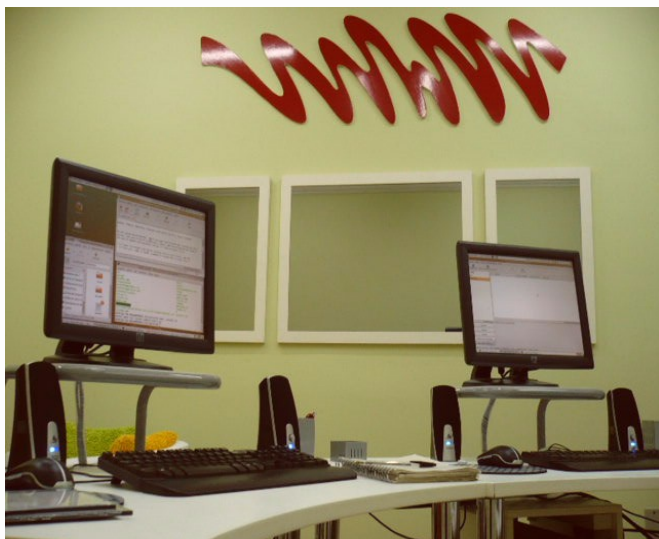


Desafios de *layout* e usabilidade

- Navegação de telas restrita a poucas opções do controle remoto.
- Falta de botões coloridos e necessidade de adaptação das interfaces gráficas com elementos focáveis.
- Foco na navegação entre as telas dos serviços de forma que o usuário saiba onde está e para onde vai. Não terá a mesma flexibilidade de uma aplicação em ambiente de internet.
- Consideração da proporção 4:3 pelo fato de ter menos espaço em tela e resolução ser mais baixa do que a 16:9.

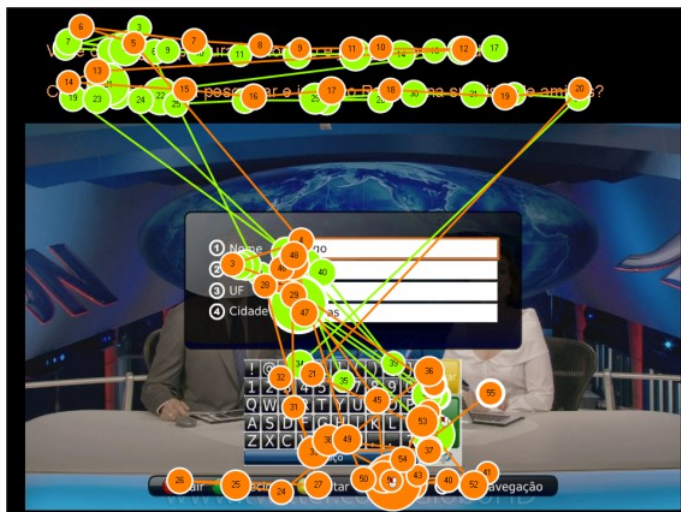
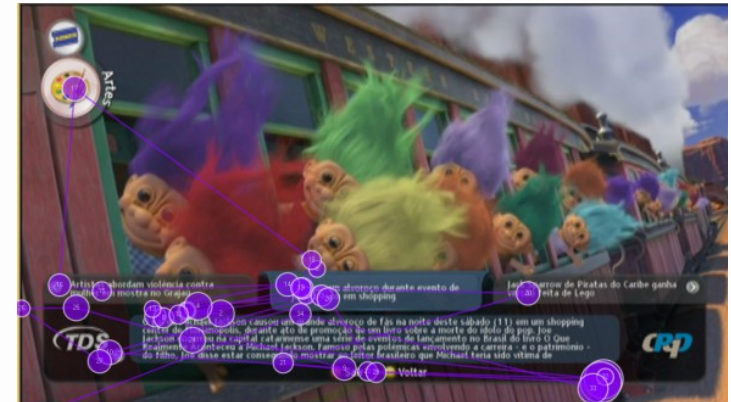
Laboratório de usabilidade

- Tem por objetivo validar, com testes de usabilidade, as interfaces de usuário propostas para projetos e soluções de software, independente de plataformas e tecnologias
- Laboratório do CPqD é composto por dois ambientes:
 - Sala de testes
 - Sala de observação



Laboratório de usabilidade

- São utilizadas ferramentas assistivas e software de apoio como forma de comprovar os aspectos observados e auxiliar na coleta de informações dos testes.
- Exemplos de ferramentas e softwares:
 - Câmeras de monitoramento
 - Morae como software de apoio
 - Equipamento de Eyetracking



Laboratório de usabilidade

- Principais testes realizados:
 - **Usabilidade** – facilidade de uso e aprendizado
 - **Acessibilidade** – eliminação de barreiras
 - **Inteligibilidade** – clareza de textos, ícones, botões de ação e comando
- Exemplos de problemas encontrados e recomendações:
 - Retirar cores amarelas de textos e substituir por negrito pois o usuário confunde com *hiperlink*
 - Não utilizar o ícone de navegação de páginas para direita e esquerda porque os usuários confundem com a tecla '*play*'
 - Manter as cores dos botões da aplicação na mesma ordem das cores dos botões do controle remoto
 - Quando houver um texto que deve ser lido na sequência, não utilizar diferentes telas, e sim o scroll vertical
 - Entre outros

Desafios e problemas de codificação

- Desenvolver serviços que tenham apelo rodando em STB com pouca capacidade de memória e processamento e sem capacidade de armazenamento.
- Poucas opções de plataformas de desenvolvimento. Atualmente desenvolvemos com máquina virtual e STB que possuem ambiente de desenvolvimento estáveis.
 - No passado tínhamos um STB que esquentava e travava a aplicação.
 - Hoje temos ambiente de outros fornecedores, porém não rodam parte das aplicações.
 - Determinamos que há problemas na execução da uma mesma aplicação entre diferentes fornecedores da implementação do *middleware*.
- A fonte de consulta de desenvolvimento é a norma da ABNT, porém nem tudo o que está na norma está implementado no *middleware*.
 - Ex: Alguns parâmetros especificados na norma, no momento do desenvolvimento foi verificado que estavam invertidos na implementação do *middleware*.

Desafios e problemas de codificação

Escolha outro dia



Dezembro



DOM	SEG	TER	QUA	QUI	SEX	SAB
	01	02	03	04	05	06
07	08	09	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			



Heródoto Barbeiro



Sair



Selecionar



Voltar

Desafios e problemas de codificação

- Memória das aplicações vem se mostrando um dos grandes desafios para o desenvolvimento de aplicações interativas ricas.
 - Geralmente um limite de 500kb se mostrou suficiente para o Lua, sendo que o *Garbage Collector* funciona regularmente.
 - Algo em torno de 90% é o que corresponde o uso de imagens dentro das aplicações
 - Atualmente no *middleware* em que desenvolvemos existe 16Mb para a memória gráfica. Ainda assim é possível executar a aplicação que o *middleware* consegue gerenciar até 40 Mb de memória, porém acima disso é nítido ao usuário que o desempenho da aplicação é prejudicado.
 - Mesmo associando '*nil*' pouco se nota com relação ao desempenho e o *middleware* não consegue gerenciar e liberar a memória gráfica.
 - Independentemente do tamanho físico (Kb) da imagem chegamos ao cálculo que cada *pixel* da imagem corresponde a 4 *bytes*. O que é relevante é a dimensão da imagem.
 - Solução temporária para o caso da memória gráfica é finalizar a mídia Lua através do NCL. Em um exemplo, de 46 Mb de memória gráfica utilizada, caiu para menos do que 16 Mb.

Componentes

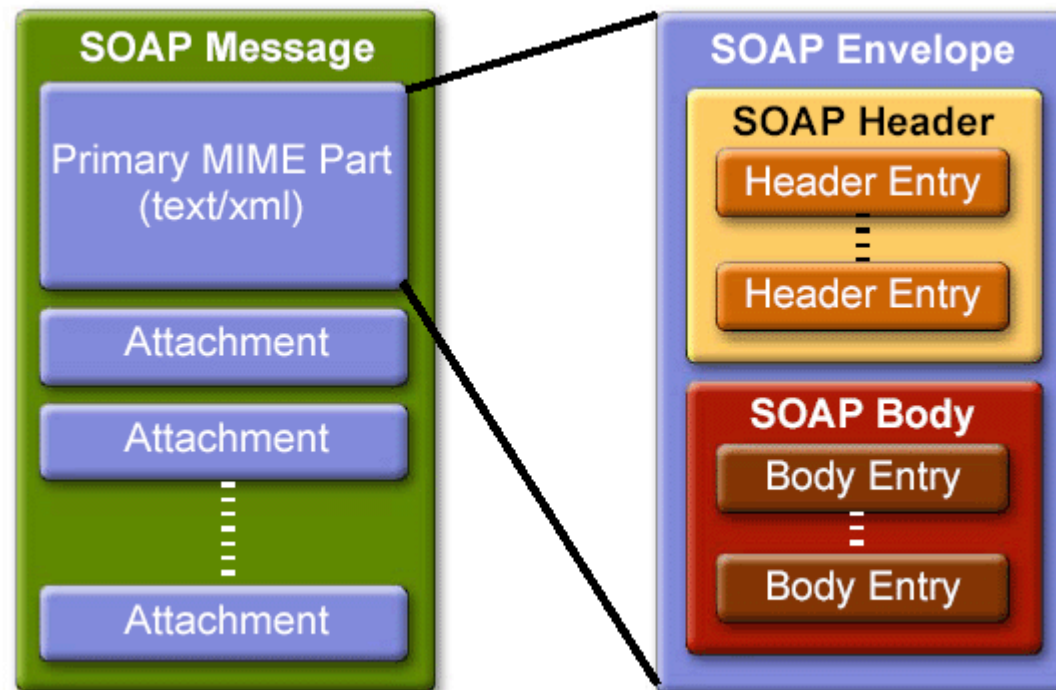
- Foram criados para agilizar o processo de desenvolvimento, principalmente da camada gráfica.
- Atualmente temos componentes para:
 - Menus, Botões, Abas, Teclado (QWERTY, SMS e T9), Scroll, Foco
 - Imagens, Input, Mensagens de texto para usuários, entre outros...
- A cada serviço desenvolvido, novas necessidades são identificadas e convertidas em componentes para a utilização pelos desenvolvedores.
- Para se desenvolver 4 telas com todos os seus elementos e navegação, todas funcionais, utilizava-se 5 dias. Agora o mesmo trabalho leva algo relativo a 1 (um) dia e meio.
- E o mais importante: Comunicação via *Web Services* entre aplicações NCLua e regras de negócio e persistência desenvolvidos em Java.

Tecnologia *Web Services*

- Implementa as características da arquitetura **SOA** e são descritos e definidos usando a linguagem **XML** e protocolo **HTTP** com padrão **SOAP**.
- Propõe a exposição das transações e regras de negócios por meio de protocolos que podem ser acessados por:
 - Qualquer linguagem de programação
 - Qualquer sistema operacional
 - Qualquer dispositivo de acesso
- **SOAP** é o padrão de troca de mensagens entre as aplicações e os *Web Services*. Utilizado independente de plataforma e linguagem de programação.
- Também garante a interoperabilidade e intercomunicação entre sistemas através de uma linguagem (**XML**) e mecanismo de transporte (**HTTP**).

Tecnologia *Web Services*

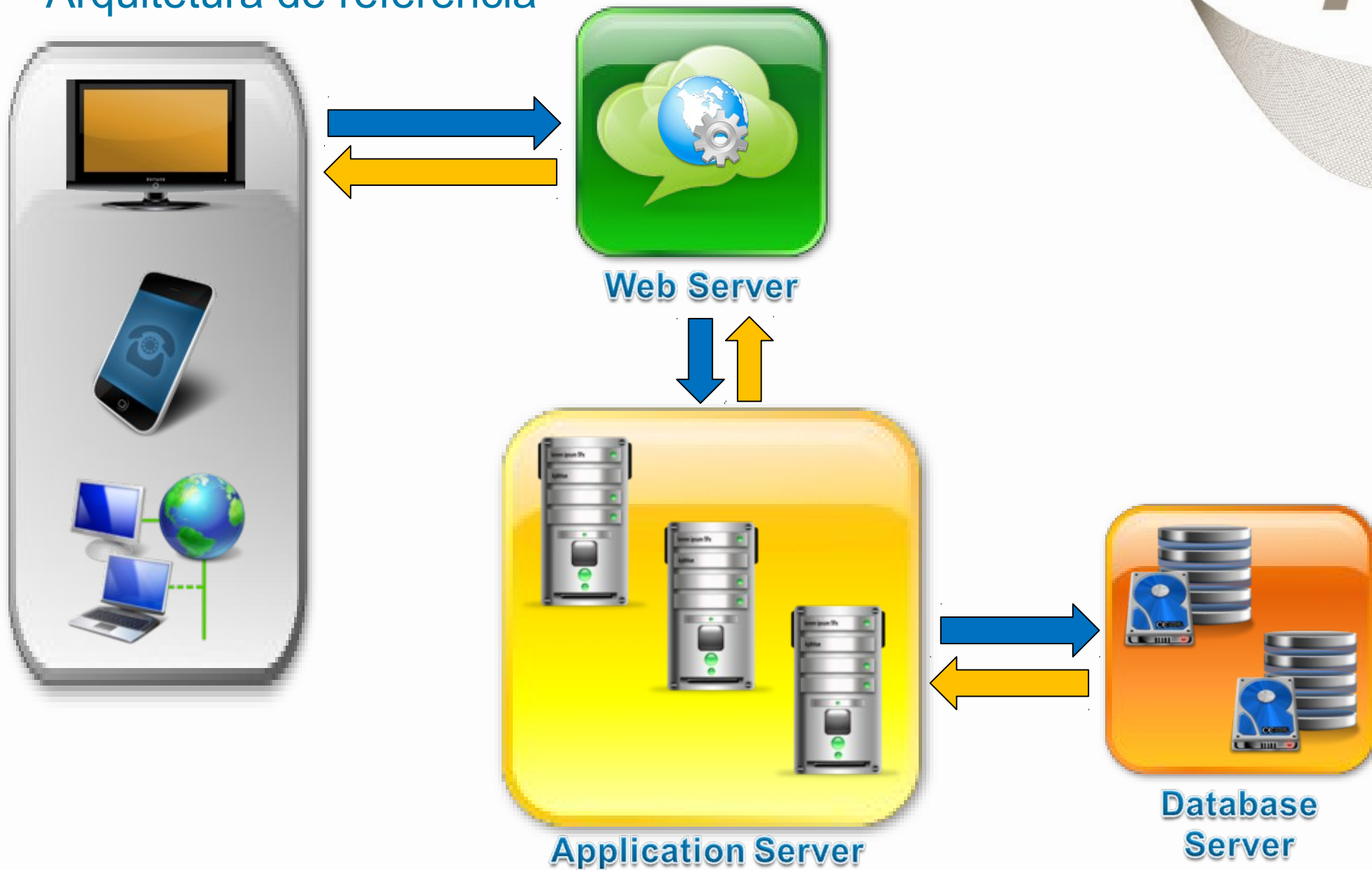
- Composição de uma mensagem **SOAP**:
 - **Envelope** – Elemento raiz do XML
 - **Header** – Contém informações adicionais
 - **Body** – A informação a ser transportada. Pode conter um elemento *Fault*, usado para carregar mensagens de status e erros retornados



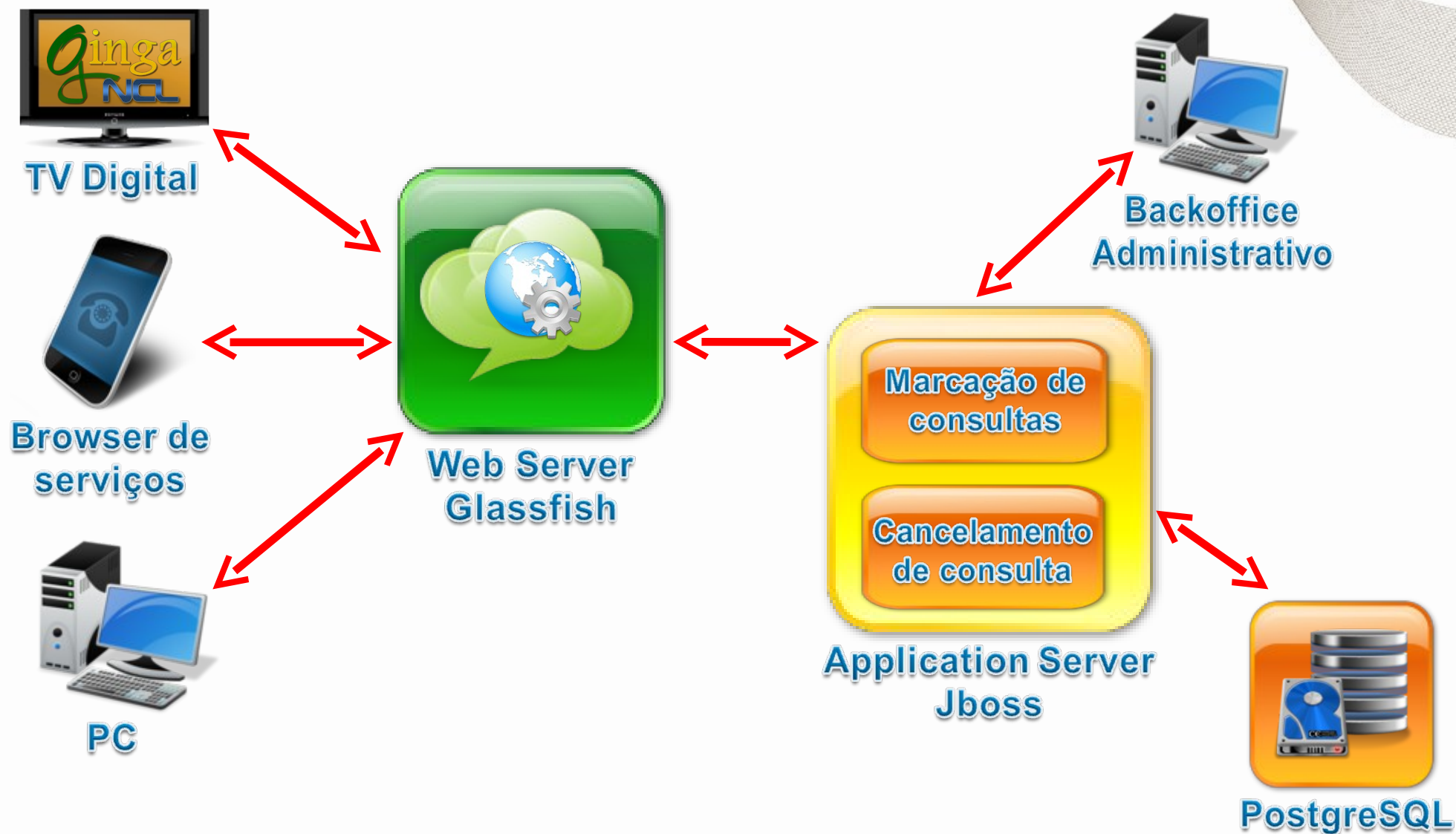
Arquitetura de referência

- Objetivo de garantir a interoperabilidade dos serviços. Independe se o serviço é transmitido via radiodifusão, satélite, IPTV, etc ou acessado por meio de vários dispositivos (celulares ou computadores).
- Valor agregado em duas frentes:
 - No desenvolvimento (componentização das regras de negócios e uso do **SOA**);
 - Na execução (regras de negócio únicas e acessadas por vários dispositivos);
- É funcional para sistemas legados e que seja de interesse a sua utilização por dispositivos de acesso atuais.
 - Necessário o desenvolvimento da camada de apresentação e uma camada de integração;

Arquitetura de referência



Arquitetura – Serviço de governo eletrônico

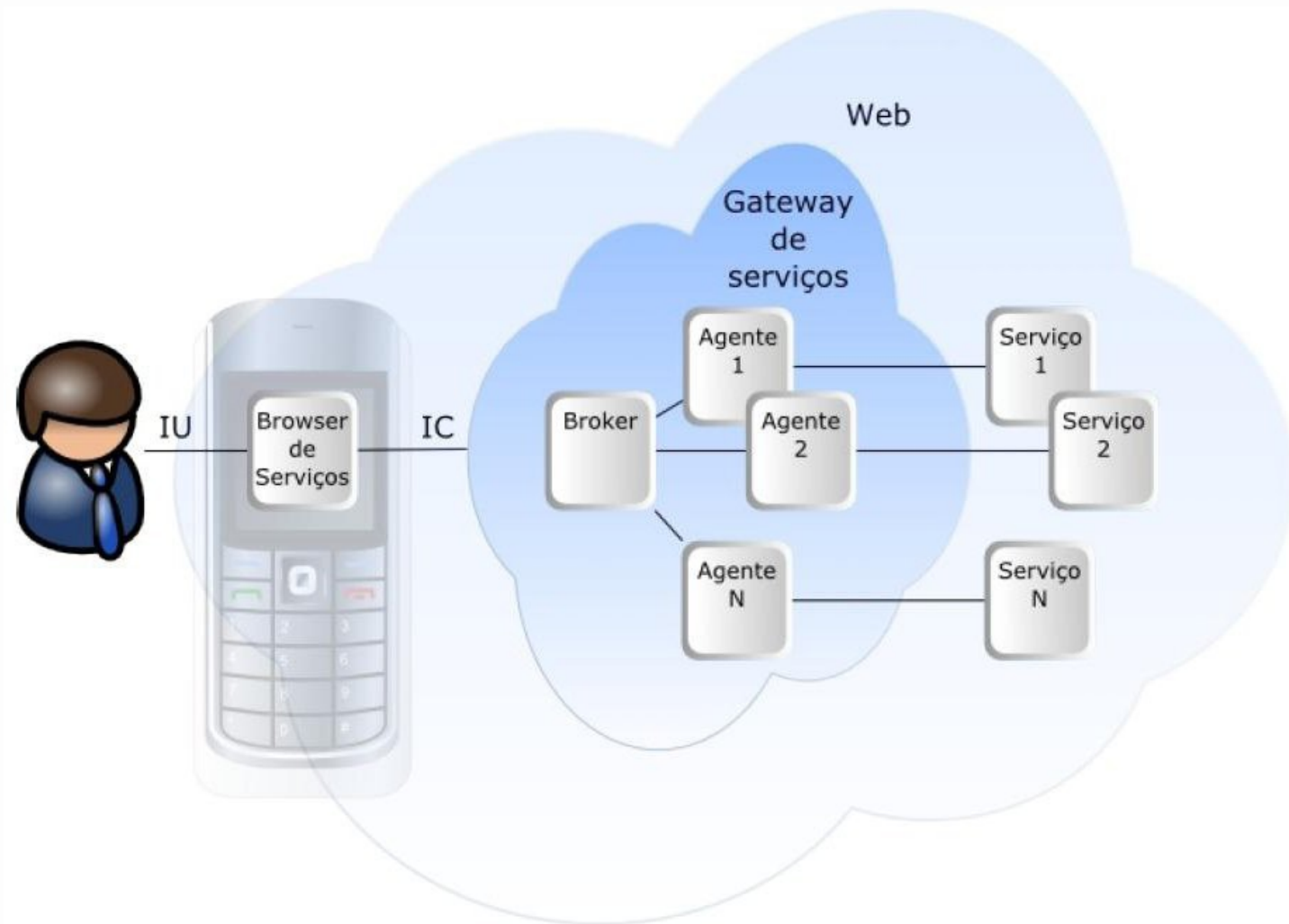


Browser de Serviços e Gateway de Serviços

- Execução do serviço de governo eletrônico no celular através de solução comercial do próprio CPqD chamada **CPqD Gateway de Serviços**.
- Principais características:
 - Arquitetura orientada a serviços;
 - O celular possui um software instalado chamado **CPqD Browser de Serviços**;
 - **CPqD Gateway de Serviços** é composto por agentes e acessam serviços através de suas interfaces de consumo;
 - **O CPqD Browser de Serviços** constrói as interfaces gráficas para o usuário do celular;
 - As interfaces são criadas a partir de metadados;

Browser de Serviços e Gateway de Serviços

- Representação gráfica do Gateway de Serviços e Browser de Serviços



Browser de Serviços e Gateway de Serviços

- Exemplo de interface gráfica:
 - Obtenção dos horários disponíveis dos médicos





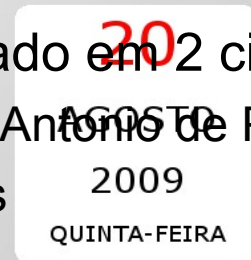
- Software executado em ambiente Web.

HORAS DISPONÍVEIS

- Foco no processo de marcação e visualização

- Implantado em 2 cidades do estado de São Paulo

- Santo Antonio de Posse
- Bastos



TROCAR O DIA

MANHÃ		
09:00		
10:00		
11:00		
TARDE		
13:00		
14:00		
15:00		
16:00		



Serviço de governo eletrônico

- Interface gráfica no serviço de TV utilizando **Ginga-NCL**.

The interface is titled "Escolha a hora da consulta" (Choose the consultation hour). It features a grid of available time slots for appointments. The next available date is displayed as 14/02/2011, Segunda feira, Fevereiro. The interface includes a video feed of a man in a suit, identified as Heródoto Barbeiro. At the bottom, there are four navigation buttons: Sair (Exit), Selecionar (Select), Voltar (Back), and Trocar dia (Change day), each with a corresponding colored circle icon. The CPQ logo is also present in the bottom right corner.

Escolha a hora da consulta

Próximo dia disponível
14/02/2011
Segunda feira
Fevereiro

Manhã			
08:30	09:00	09:30	10:00
10:30	11:00	11:30	

Tarde			
13:00	13:30	14:00	14:30
15:00	15:30	16:00	

Heródoto Barbeiro

Sair Selecionar Voltar Trocar dia

CPQ

Criação de interfaces de integração com Axis2

- Projeto da Apache e é uma implementação **SOAP**.
- Possui suporte a **WSDL** (*Web Service Description Language*).
- Opção de geração automática de classes Java a partir de um arquivo **WSDL** e vice-versa.
- Pode ser utilizado em vários servidores Web e de aplicações:
 - **Apache Tomcat**;
 - **Glassfish**;
 - **Jboss**;
 - entre outros;

Arquivo XSD

- Arquivo responsável por definir os parâmetros e os atributos dos parâmetros para cada operação que for existir no serviço.

Schema : <http://www.cpqd.com.br/xsd/TGOVSAude>

Directives

Elements

- ReqAgendaHorario : reqAgendaHorario
- ReqCancelamentoAgendamento : reqCancelamentoAgendamento
- ReqConfirmacaoAgendamento : reqConfirmacaoAgendamento
- ReqConsultasAgendadas : string
- ReqDependentes : string
- ReqEspecialidadesAtendimento : string
- ReqMedicos : reqMedicos
- RespAgendaHorario : respAgendaHorario
- RespCancelamentoAgendamento : string
- RespConfirmacaoAgendamento : string
- RespConsultasAgendadas : respConsultasAgendadas
- RespDependentes : respDependentes
- RespEspecialidadesAtendimento : respEspecialidadesAtendimento
- RespMedicos : respMedicos

Types

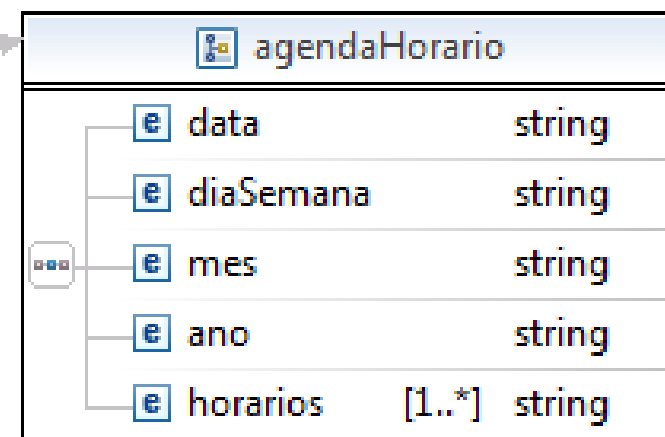
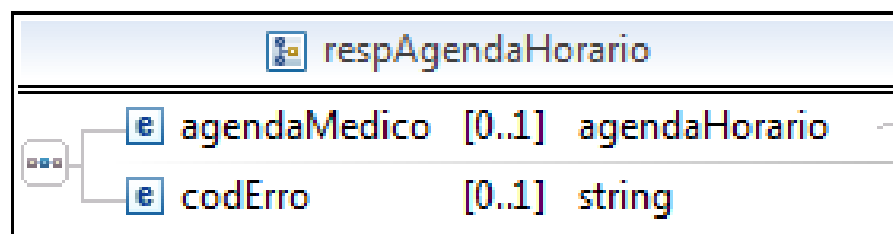
- agendaHorario
- listaConsultasAgendadas
- listaDependentes
- listaEspecialidadesAtendimento
- listaMedicos
- reqAgendaHorario
- reqCancelamentoAgendamento
- reqConfirmacaoAgendamento
- reqMedicos
- respAgendaHorario
- respConsultasAgendadas
- respDependentes
- respEspecialidadesAtendimento
- respMedicos

Attributes

Groups

Arquivo XSD

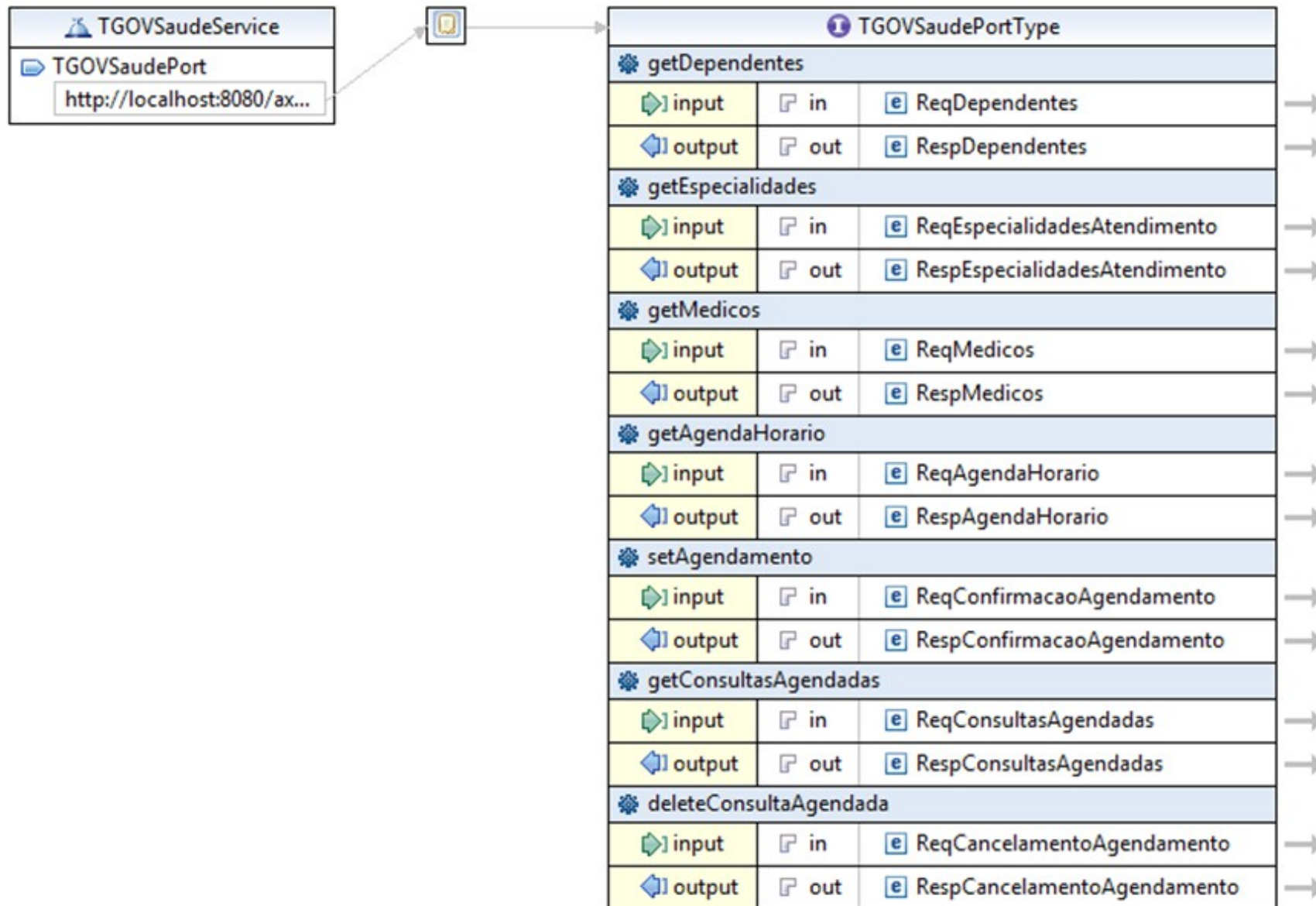
- Detalhamento de um parâmetro no arquivo **XSD**.



Arquivo WSDL

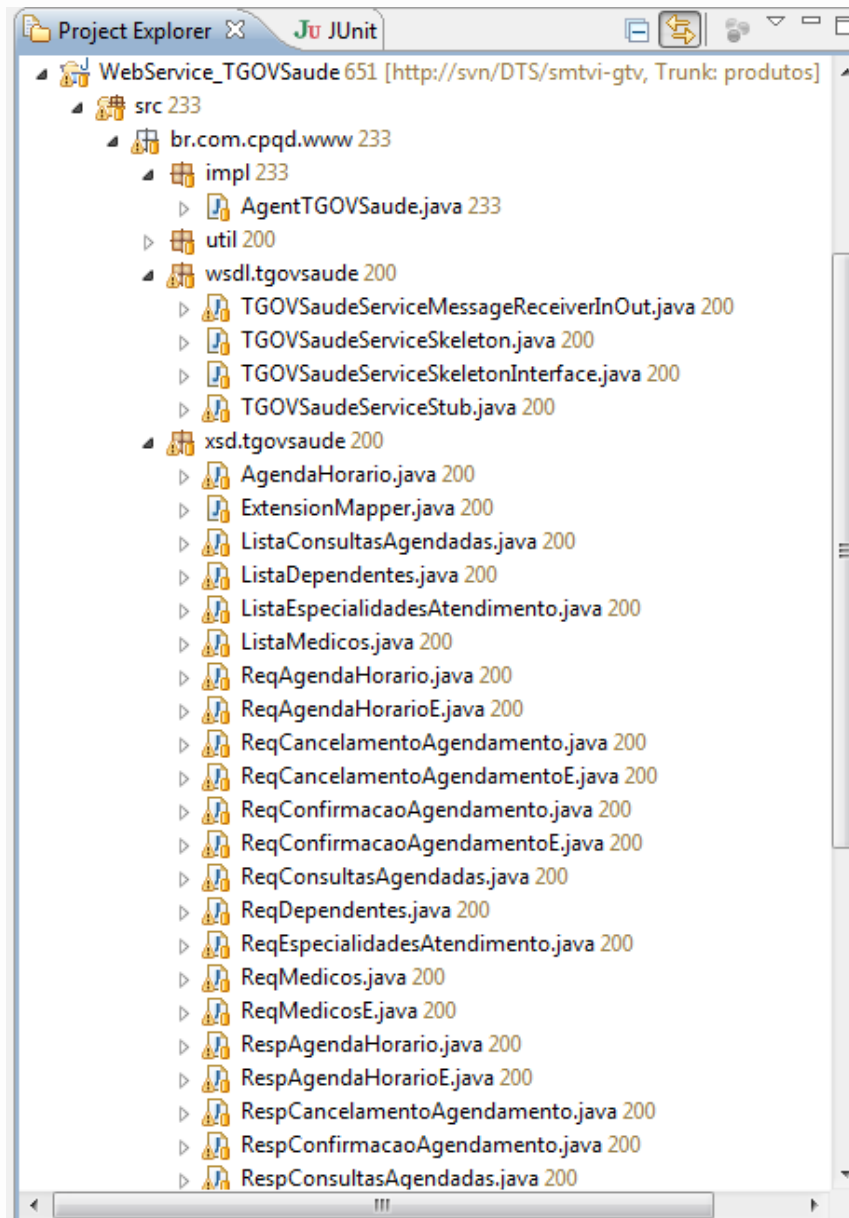
- Arquivo responsável por definir detalhadamente um *Web Services*.
- Especifica as operações que compõem o *Web Services* e define de forma clara como deve ser o formato de entrada e saída de cada operação.

Arquivo WSDL



Classes do serviço geradas através do Axis2

- Visão através de ferramenta de desenvolvimento Java com as classes que foram geradas através do **Axis2**.



Demonstração de aplicações interativas

- Comércio eletrônico
- Inclua Saúde (Marcação e visualização de consultas e informações de saúde)
- Emprego
- Previdência

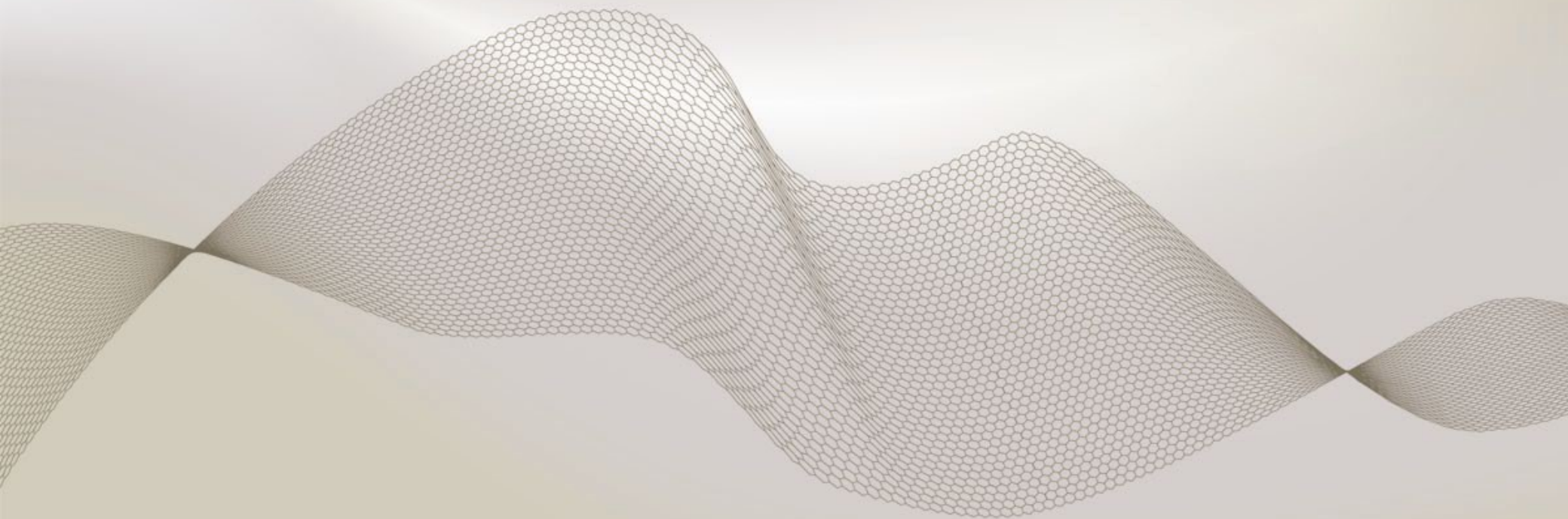
Conclusões finais

- Há uma crescente demanda por serviços interativos de TV Digital e convergência com outros dispositivos de acesso como celulares e computadores.
- Para os desenvolvedores apenas a TV e uma máquina virtual bastam, porém para uma visão mais profissional no desenvolvimento de aplicativos é necessário ter equipamentos para emular toda a cadeia do serviço desde da transmissão até a recepção.
- Há a necessidade de melhoria nos ambientes de desenvolvimento, principalmente com relação a *debug* de aplicações.
- Precisamos de certificação com relação aos *middlewares* disponibilizados para a comunidade de desenvolvedores. Há a necessidade de garantir que a aplicação seja executada em qualquer equipamento, independentemente do fornecedor.
- Aplicação interativa não é apenas desenvolver e disponibilizar a aplicação ao usuário, é preciso levar em conta a experiência do usuário (QoE) na manipulação do aplicativo (fundamental a questão de usabilidade, inteligibilidade e acessibilidade)

Raphael de Carvalho Barbosa

raphaelb@cpqd.com.br

(19) 3705-4151



www.cpqd.com.br