

UNIVERSIDADE FEDERAL DE PERNAMBUCO

GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO

CENTRO DE INFORMÁTICA

2008.2

---

UTILIZAÇÃO DE ARQUITETURA ORIENTADA A SERVIÇOS  
EM PEQUENOS NEGÓCIOS

---

**TRABALHO DE GRADUAÇÃO**

**CAIO TIZEI DE ANDRADE LIRA**

Novembro de 2008

UNIVERSIDADE FEDERAL DE PERNAMBUCO

GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO

CENTRO DE INFORMÁTICA

2008.2

---

UTILIZAÇÃO DE ARQUITETURA ORIENTADA A SERVIÇOS  
EM PEQUENOS NEGÓCIOS

---

Este trabalho foi apresentado ao programa de Graduação em Engenharia da Computação pelo Centro de Informática da Universidade Federal de Pernambuco, como requisito parcial para obtenção do título de Engenheiro da Computação.

Orientador: Carlos André Guimarães Ferraz

Co-orientador: Marcelo Corrêa de Moraes

## ASSINATURAS

---

Este Trabalho de Graduação representa o esforço do aluno **Caio Tizei de Andrade Lira**, orientado pelo professor **Carlos André Guimarães Ferraz** e co-orientado por **Marcelo Corrêa de Moraes**, sob o título “**Utilização de Arquitetura Orientada a Serviços em Pequenos Negócios**”. Todos abaixo estão de acordo com o conteúdo deste documento e os resultados deste trabalho de graduação.

---

Carlos André Guimarães Ferraz  
(orientador)

---

Caio Tizei de Andrade Lira  
(aluno)

A minha família

## AGRADECIMENTOS

---

A toda a minha família. Em particular, a minha mãe, por todo amor que me deu. A meu pai, por sempre ter acreditado em mim e me apoiado em minhas decisões. A minhas irmãs, por terem me aturado durante todos estes anos. A minhas avós por todo carinho e atenção.

Aos meus amigos do *Cinetrash*: Bruno, Guilherme, Léo e Lucas. Em especial a Juliana, por todos os ótimos momentos que vivemos ao longo de vários anos. Acho que temos bastante história para contar.

Aos meus amigos de Engenharia da Computação: Aretakis, Boneco, Breno, Flora, Hugo, João Victor, Nelson, Renata, Tiaguinho, por todos os momentos bons e ruins que passamos dentro e fora do CIn. São experiências que não irei me esquecer. Também gostaria de agradecer a nosso estimado amigo Baracho, que desistiu do curso precocemente, mas deixou muitas histórias para serem contadas.

A meus demais amigos de várias partes do mundo: Ada, Ana, Danilo, Douglas, Érico, Helena, Lucas, Rafael e vários outros que não citei. Obrigado por fazerem parte da minha vida.

A meus orientadores: Prof. PhD Carlos Ferraz, por ter aceitado o desafio de orientar este trabalho à distância e Prof. MSc Marcelo Moraes, por toda a contribuição com o seu conhecimento.

A meus companheiros de trabalho na *Chemtech*, por terem me colocado no caminho que conduziu ao desenvolvimento deste Trabalho de Graduação.

## RESUMO

---

As grandes companhias buscaram durante muito tempo por uma solução eficiente e economicamente viável para integração de sistemas e obtenção de maior alinhamento entre o negócio e a Tecnologia da Informação, até que o conceito de SOA apareceu e vem sendo adotado em larga escala por estas empresas para tal finalidade. SOA foi elaborado pensando-se nestas grandes corporações, mas recentemente passou-se a discutir sua aplicabilidade a ambientes de menor porte. Neste contexto, este trabalho se propõe analisar a aplicabilidade de SOA a pequenos negócios, propondo algumas idéias para que esta adequação possa ocorrer e realizando um estudo de caso na implantação para pequenos negócios, fazendo uso de tecnologias e ferramentas gratuitas.

**Palavras-chave:** Arquitetura Orientada a Serviços, Pequenos Negócios.

## ABSTRACT

---

For a long time, big corporations sought for an efficient and economically practicable solution for systems integration and obtainment of a better alignment between business and Information Technology. SOA came to resolve these problems and has been used in large scale by these companies for such end. SOA was conceived in consideration of these problems by big corporations, nevertheless its applicability for small environments has been discussed recently. This work aims to analyze the adoption of SOA by small businesses, proposing some ideas to allow it to happen and realizing a case study, making use of open source tools and technologies.

**Keywords:** Service-Oriented Architecture, Small Business.

# SUMÁRIO

---

|  |           |
|--|-----------|
| <b>1. INTRODUÇÃO.....</b>                              | <b>11</b> |
| 1.1. CONTEXTO .....                                    | 11        |
| 1.2. OBJETIVOS.....                                    | 12        |
| 1.3. ESTRUTURA DO TRABALHO.....                        | 12        |
| <b>2. ARQUITETURA ORIENTADA A SERVIÇOS .....</b>       | <b>13</b> |
| 2.1. CONCEITOS .....                                   | 13        |
| 2.1.1. SERVIÇO .....                                   | 13        |
| 2.1.2. SOFTWARE AS A SERVICE (SAAS).....               | 14        |
| 2.1.3. COMPUTAÇÃO ORIENTADA A SERVIÇOS.....            | 14        |
| 2.1.4. ARQUITETURA ORIENTADA A SERVIÇOS .....          | 15        |
| 2.2. PRINCÍPIOS DE DESIGN .....                        | 16        |
| 2.2.1. PRINCÍPIOS QUE IMPLEMENTAM CARACTERÍSTICAS..... | 16        |
| 2.2.2. PRINCÍPIOS REGULADORES .....                    | 17        |
| 2.2.3. PRINCÍPIOS NA ELABORAÇÃO DE UM SERVIÇO .....    | 18        |
| 2.3. MODELOS DE SERVIÇOS.....                          | 19        |
| 2.3.1. SERVIÇOS DE UTILIDADE .....                     | 20        |
| 2.3.2. SERVIÇOS DE ENTIDADE .....                      | 20        |
| 2.3.3. SERVIÇOS DE TAREFAS.....                        | 20        |
| 2.4. TECNOLOGIAS UTILIZADAS.....                       | 21        |
| 2.4.1. WEB SERVICES.....                               | 21        |
| 2.4.2. ENTERPRISE SERVICE BUS (ESB) .....              | 25        |
| 2.5. ESTRATÉGIAS DE ENTREGA DE SOA .....               | 27        |
| 2.5.1. CICLO DE VIDA BÁSICO.....                       | 27        |
| 2.5.2. ESTRATÉGIAS ESPECÍFICAS.....                    | 29        |
| 2.6. BENEFÍCIOS .....                                  | 30        |
| 2.6.1. BENEFÍCIOS DE TECNOLOGIA .....                  | 31        |
| 2.6.2. BENEFÍCIOS DE NEGÓCIO .....                     | 32        |
| <b>3. PEQUENOS NEGÓCIOS.....</b>                       | <b>34</b> |
| 3.1. CARACTERIZAÇÃO .....                              | 34        |
| 3.2. VANTAGENS .....                                   | 35        |
| 3.2.1. PESSOALIDADE.....                               | 35        |
| 3.2.2. EXCELÊNCIA EM ATIVIDADES ESPECÍFICAS .....      | 35        |
| 3.2.3. EFICIÊNCIA OPERACIONAL.....                     | 36        |
| 3.3. DESAFIOS .....                                    | 36        |
| 3.3.1. FALTA DE CAPITAL .....                          | 36        |
| 3.3.2. PLANEJAMENTO .....                              | 37        |
| 3.3.3. INSERÇÃO NO MERCADO .....                       | 37        |
| 3.4. A TI PARA OS PEQUENOS NEGÓCIOS.....               | 37        |
| 3.4.1. UTILIZAÇÃO .....                                | 38        |
| 3.4.2. OBJETIVOS.....                                  | 38        |
| 3.4.3. RISCOS.....                                     | 39        |
| <b>4. ADOTANDO SOA EM PEQUENOS NEGÓCIOS.....</b>       | <b>41</b> |
| 4.1. ESCOLHA DOS SERVIÇOS NA ADOÇÃO DE SOA .....       | 41        |

|  |           |
|--|-----------|
| 4.1.1. <i>INSIDE-OUT</i> .....                                 | 41        |
| 4.1.2. <i>OUTSIDE-IN</i> .....                                 | 42        |
| <b>4.2. APLICABILIDADE DE SOA EM PEQUENOS NEGÓCIOS</b> .....   | <b>43</b> |
| 4.2.1. <i>COMÉRCIO VAREJISTA</i> .....                         | 44        |
| 4.2.2. <i>COMÉRCIO ELETRÔNICO (E-COMMERCE)</i> .....           | 44        |
| 4.2.3. <i>PEQUENAS INDÚSTRIAS</i> .....                        | 44        |
| 4.2.4. <i>PRESTAÇÃO DE SERVIÇOS</i> .....                      | 45        |
| 4.2.5. <i>TURISMO</i> .....                                    | 45        |
| 4.2.6. <i>DESENVOLVIMENTO DE SOFTWARE</i> .....                | 46        |
| 4.2.7. <i>FRANQUIAS</i> .....                                  | 46        |
| <b>4.3. ESTRATÉGIA PARA SOA EM PEQUENOS NEGÓCIOS</b> .....     | <b>47</b> |
| 4.3.1. <i>DEFINIÇÃO DO MODELO DE NEGÓCIOS</i> .....            | 48        |
| 4.3.2. <i>ANÁLISE ORIENTADA A SERVIÇOS</i> .....               | 48        |
| 4.3.3. <i>DESIGN DE SERVIÇOS</i> .....                         | 49        |
| 4.3.4. <i>DESENVOLVIMENTO</i> .....                            | 49        |
| 4.3.5. <i>TESTES</i> .....                                     | 49        |
| 4.3.6. <i>INSTALAÇÃO</i> .....                                 | 49        |
| 4.3.7. <i>ADMINISTRAÇÃO</i> .....                              | 50        |
| <b>5. ESTUDO DE CASO</b> .....                                 | <b>51</b> |
| 5.1. <i>DESCRIÇÃO</i> .....                                    | 51        |
| 5.2. <i>IMPLANTAÇÃO DA CONSULTA A PACOTES DE VIAGEM</i> .....  | 52        |
| 5.2.1. <i>DEFINIÇÃO DO MODELO DE NEGÓCIO</i> .....             | 52        |
| 5.2.2. <i>ANÁLISE ORIENTADA A SERVIÇOS</i> .....               | 54        |
| 5.2.3. <i>DESIGN ORIENTADO A SERVIÇOS</i> .....                | 59        |
| 5.2.4. <i>DESENVOLVIMENTO</i> .....                            | 63        |
| 5.2.5. <i>TESTES</i> .....                                     | 65        |
| 5.2.6. <i>INSTALAÇÃO</i> .....                                 | 66        |
| 5.2.7. <i>ADMINISTRAÇÃO</i> .....                              | 67        |
| <b>6. CONCLUSÃO</b> .....                                      | <b>68</b> |
| 6.1. <i>CONTRIBUIÇÕES</i> .....                                | 68        |
| 6.2. <i>TRABALHOS FUTUROS</i> .....                            | 68        |
| <b>REFERÊNCIAS</b> .....                                       | <b>69</b> |
| <b>APÊNDICE I - ESQUEMAS XSD DOS SERVIÇOS DA TRAVELX</b> ..... | <b>73</b> |
| <b>APÊNDICE II - DESCRIÇÕES WSDL ABSTRATAS</b> .....           | <b>78</b> |

## ÍNDICE DE FIGURAS

---

|   |    |
|---|----|
| FIGURA 2.1 – EXEMPLO DE SERVIÇO COTIDIANO.....  | 13 |
| FIGURA 2.2 – EXEMPLO DE SERVIÇO EM CONTEXTO COMPUTACIONAL.....                                | 14 |
| FIGURA 2.3 – ATUAÇÃO DE PRINCÍPIOS NA ELABORAÇÃO DE UM SERVIÇO.....                           | 19 |
| FIGURA 2.4 – HIERARQUIA DOS TIPOS DE SERVIÇOS.....  | 21 |
| FIGURA 2.5 – EXEMPLO DE ARQUIVO XML.....  | 22 |
| FIGURA 2.6 – RELACIONAMENTO ENTRE <i>WEB SERVICES</i> E OUTRAS TECNOLOGIAS<br>AUXILIARES..... | 25 |
| FIGURA 2.7 – EXEMPLO DE UTILIZAÇÃO DE ESB.....  | 26 |
| FIGURA 2.8 – CICLO DE VIDA BÁSICO.....  | 27 |
| FIGURA 2.9 – ESTRATÉGIA <i>TOP-DOWN</i> .....   | 29 |
| FIGURA 4.1 – ADOÇÃO <i>INSIDE-OUT</i> .....   | 42 |
| FIGURA 4.2 – ECOSSISTEMA DE NEGÓCIOS.....   | 43 |
| FIGURA 4.3 – ESTRATÉGIA PARA ADOÇÃO DE SOA EM PEQUENOS NEGÓCIOS.....                          | 48 |
| FIGURA 5.1 – PROCESSO DE NEGÓCIO DA CONSULTA DE PACOTES.....                                  | 53 |
| FIGURA 5.2 – SERVIÇOS AGNÓSTICOS DE ENTIDADE.....   | 55 |
| FIGURA 5.3 – SERVIÇO CANDIDATO PARA CONSULTAR PACOTE VIAGEM.....                              | 56 |
| FIGURA 5.4 – SERVIÇOS MODIFICADOS PARA MAIOR REUSO.....                                       | 57 |
| FIGURA 5.5 – SERVIÇOS CANDIDATOS APÓS APLICAÇÃO DOS PRINCÍPIOS DE SOA.....                    | 58 |
| FIGURA 5.6 – COMPOSIÇÃO DOS SERVIÇOS CANDIDATOS.....  | 59 |
| FIGURA 5.7 - SERVIÇO DE CONSULTA AO CPF DA CONSULTACPF.COM.....                               | 60 |
| FIGURA 5.8 - XSD DO VALIDAÇÃO DE CPF.....   | 61 |
| FIGURA 5.9 - XSD DA VERIFICAÇÃO DE RESTRIÇÕES NO CPF.....                                     | 61 |
| FIGURA 5.10 - WSDL DO SERVIÇO DE CPF.....   | 62 |
| FIGURA 5.11 - DOCUMENTAÇÃO DO SERVIÇO DE CPF.....   | 63 |
| FIGURA 5.12 – COMPARAÇÃO DE DESEMPENHO DO WSO2 ESB COM OUTRAS SOLUÇÕES<br>.....               | 64 |
| FIGURA 5.13 – GERAÇÃO <i>TOP-DOWN</i> DE CÓDIGO ESQUELETO JAVA PARA <i>WEB SERVICES</i> ..... | 65 |
| FIGURA 5.14 - TESTE FUNCIONAL DA OPERAÇÃO VALIDARNUMERO DE CPF.....                           | 66 |
| FIGURA 5.15 - SERVIÇO DE CPF INSTALADO NO WSO2 APPLICATION SERVER.....                        | 66 |
| FIGURA 5.16 - ESTATÍSTICAS DO WSO2 ESB.....   | 67 |
| FIGURA 5.17 - LOGS DO WSO2 ESB.....   | 67 |

# 1. INTRODUÇÃO

---

## 1.1. Contexto

As grandes companhias buscaram durante muito tempo por uma solução eficiente para integração de seus sistemas e conseqüente obtenção de um maior alinhamento entre o negócio executado e a tecnologia da informação (TI). Neste cenário apareceu o conceito de Arquitetura Orientada a Serviços (SOA, do inglês *Service Oriented Architecture*), que vem sendo adotado em larga escala por estas empresas para estas e outras finalidades.

Segundo a consultoria *Gartner*, líder mundial em pesquisa e análise sobre tecnologia, 50% das novas aplicações operacionais de missão crítica e processos de negócio desenvolvidas em 2007 devem ter utilizado SOA. Para 2010 a expectativa do *Gartner* é que SOA marque presença em 80% das aplicações desenvolvidas nas grandes corporações [2].

SOA foi elaborado pensando-se justamente nas grandes corporações, não sendo tido como aplicável a negócios de menor porte. Isto se deveria ao fato de que a adoção de SOA dentro de uma empresa requereria uma grande infra-estrutura e pessoal altamente capacitado. Além disso, muito esforço seria gasto e o retorno não seria suficiente para justificá-lo, pois este tipo de empresa não teria tantas necessidades de integração [4].

No entanto, recentemente passou-se a discutir a aplicabilidade de SOA a estas empresas de menor porte, havendo alguns estudos relacionados, como em [1] e [3]. De acordo com estes artigos, a implantação de SOA nestas empresas não deve ocorrer da mesma forma que nas grandes corporações, de forma que uma modificação na forma de adoção viabilizaria o uso de SOA em pequenos negócios.

## **1.2. Objetivos**

O objetivo geral deste trabalho é analisar e traçar caminhos para a adoção de SOA por parte de pequenos negócios. A partir daí, será elaborado um estudo de caso para simulação de um pequeno negócio adotando SOA de acordo com as idéias desenvolvidas neste trabalho.

## **1.3. Estrutura do Trabalho**

A fim de atingir seu objetivo, este trabalho é composto de seis capítulos, com a seguinte composição:

- Capítulo 1 - Apresenta o contexto no qual este trabalho está inserido, bem como seus objetivos e estruturação.
- Capítulo 2 - Mostra os conceitos da arquitetura orientada a serviços, destacando seus princípios e benefícios que podem ser obtidos com a sua utilização.
- Capítulo 3 - Realiza uma discussão sobre pequenos negócios, analisando suas características e desafios enfrentados.
- Capítulo 4 - Analisa e propõe idéias para adoção de SOA por parte de pequenos negócios.
- Capítulo 5 - Realiza um estudo de caso sobre a adoção de SOA em pequenos negócios.
- Capítulo 6 - Apresenta as conclusões deste trabalho, ressaltando suas contribuições e projetando trabalhos futuros.

## 2. ARQUITETURA ORIENTADA A SERVIÇOS

---

Neste capítulo discutiremos sobre a Arquitetura Orientada a Serviços, analisando inicialmente alguns conceitos relacionados. Em seguida serão descritos os princípios de *design* nos quais SOA se baseia, os modelos de serviços, tecnologias utilizadas e as formas de adoção de SOA. Por fim, falaremos sobre os benefícios que SOA se propõe a trazer.

### 2.1. Conceitos

A fim de melhor entender a Arquitetura Orientada a Serviços, alguns conceitos iniciais serão apresentados a seguir nesta seção.

#### 2.1.1. Serviço

A palavra serviço em um contexto geral pode ser definida como “ato ou ação útil aos interesses de alguém” [5]. Tomando como exemplo o cenário da Figura 2.1, no qual Alex deseja saber o telefone de uma localidade e pergunta para Carlos, que lhe responde. Pode-se dizer que Carlos prestou um serviço a Alex ao informá-lo o telefone.

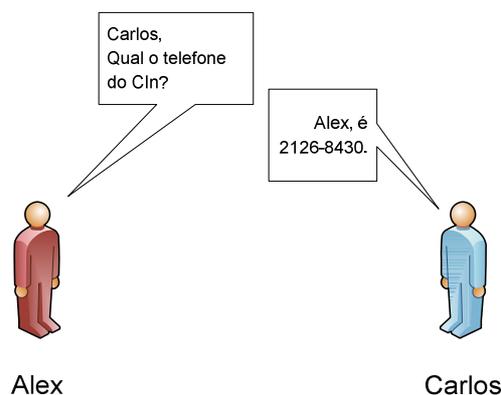


Figura 2.1 - Exemplo de serviço cotidiano

Dentro do contexto de *software* e sistemas computacionais não é muito diferente o que se espera de um serviço. Pode-se definir mais formalmente um serviço em computação como uma unidade de trabalho realizado por um provedor de serviços para atingir os resultados desejados por um consumidor de serviços [7].

### 2.1.2. *Software as a Service (SaaS)*

Remodelando, em alto nível, o cenário descrito anteriormente para um sistema computacional, podemos imaginar que um determinado sistema, Sistema X, precise obter um telefone, o que o caracteriza como um consumidor de serviços. Então pode existir um serviço que seja capaz de lhe prover esta informação, como ilustrado na Figura 2.2.

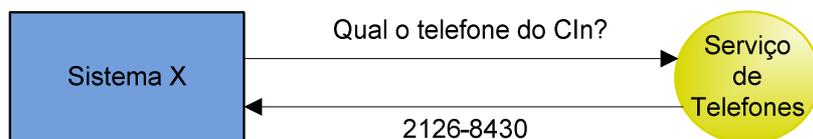


Figura 2.2 - Exemplo de serviço em contexto computacional

Em SaaS, o *software* deve ser fornecido na forma de um serviço, como no caso acima. Seu modelo de entrega essencialmente separa governança do *software* e usuário, de forma que o dono do serviço é quem hospeda o *software* e permite que os usuários executem o serviço sob demanda [12].

### 2.1.3. *Computação Orientada a Serviços*

Computação Orientada a Serviços, do inglês *Service-Oriented Computing (SOC)*, representa uma plataforma para computação distribuída de nova geração. Esta plataforma engloba seu próprio paradigma de *design*, a orientação a serviços,

bem como padrões de linguagens, princípios de *design*, *frameworks*, tecnologias e outros conceitos relacionados [6].

SOC utiliza serviços como unidade básica de construção para permitir uma fácil composição de aplicações distribuídas, inclusive em ambientes bastante heterogêneos. Sua proposta é criar um mundo no qual serviços cooperam entre si, com baixo acoplamento, permitindo a criação processos de negócio dinâmicos e flexíveis [8]. Desta forma, as aplicações orientadas a serviços são montadas pela composição de vários serviços independentes, que reunidos conseguiriam executar todas as funcionalidades desejadas.

#### **2.1.4. Arquitetura Orientada a Serviços**

SOA é um estilo de arquitetura de software que beneficia a eficiência, agilidade e produtividade no desenvolvimento de aplicações, de acordo com os objetivos da computação orientada a serviços [6]. Ou seja, SOA não se trata de uma tecnologia ou produto para criação de *software*, mas sim da forma como este deve ser construído.

O *design* de uma Arquitetura Orientada a Serviços se baseia em um conjunto de princípios para garantir que seus objetivos sejam alcançados. Estes princípios definem e regulam como as aplicações devem ser implementadas de acordo com a orientação a serviços. Na seção 2.2 descreveremos individualmente cada um destes princípios.

É importante ressaltar que a Arquitetura Orientada a Serviços abrange tanto as arquiteturas de cada aplicação específica como também a arquitetura corporativa, no entanto não é necessário que toda a empresa passe a ser orientada a serviços, apenas as áreas nas quais SOA realmente traga benefícios [9].

## **2.2. Princípios de *Design***

O paradigma da orientação a serviços se baseia em um conjunto de princípios de *design* a serem seguidos por arquiteturas que queiram ser aderentes a SOA. Existem certas diferenças em relação à definição de cada princípio individualmente de autor para autor, no entanto a idéia geral contida ao se reunir todos os princípios é a mesma. Neste trabalho, será adotada a definição dos princípios de acordo com Thomas Erl [6] [9].

Estes princípios podem ser classificados em dois grupos básicos: princípios que implementam características e princípios que regulam a aplicação de outros princípios. A seguir serão definidos estes grupos.

### **2.2.1. Princípios que Implementam Características**

São aqueles princípios cuja aplicação resulta em propriedades específicas da arquitetura final. A seguir cada um será descrito individualmente.

#### a) Padronização do Contrato de Serviços

Os serviços são acessados através de um contrato que deve conter informações sobre suas funcionalidades e propósitos. Ao realizar o *design* de um serviço, todas as suas possibilidades de utilização devem ser analisadas, de forma que este contrato não precise ser modificado no futuro, quando novas entidades precisem acessar este serviço já criado anteriormente.

#### b) Reusabilidade

Os serviços devem ser construídos de forma que possam ser reutilizados em diferentes processos. Para tal, eles devem ser construídos ignorando o máximo possível de detalhes do contexto de negócio ao qual se encontram. Serviços que

conseguem atingir esta característica costumam ser chamados de serviços agnósticos.

c) Autonomia

Para que os serviços realizem suas funcionalidades consistentemente e de forma confiável, a sua lógica subjacente deve ter um nível significativo de controle sobre o ambiente e seus recursos [6]. Isto é, os serviços devem possuir um certo grau de liberdade na sua atuação em relação aos demais serviços.

d) Ausência de Estado (*Statelessness*)

Serviços devem minimizar a quantidade de informação de estado que gerenciam e a duração pela qual eles a guardam. Caso contrário, a sua disponibilidade para outros solicitantes será comprometida [9].

e) Possibilidade de Descobrimto (*Discoverability*)

Para que um dado serviço possa ser usado e reusado dentro de um ambiente de negócio, ele deve ter a capacidade de ser encontrado, identificado e compreendido facilmente.

### **2.2.2. Princípios Reguladores**

Tratam-se de princípios que não produzem características diretas da sua aplicação, mas que influenciam nas características de outros princípios. A seguir descrevemos cada um.

#### a) Baixo acoplamento

Este princípio consiste em garantir que cada serviço seja o mais independente possível de outros. Desta forma, ao se modificar um determinado serviço, os outros não deverão ser afetados.

#### b) Abstração

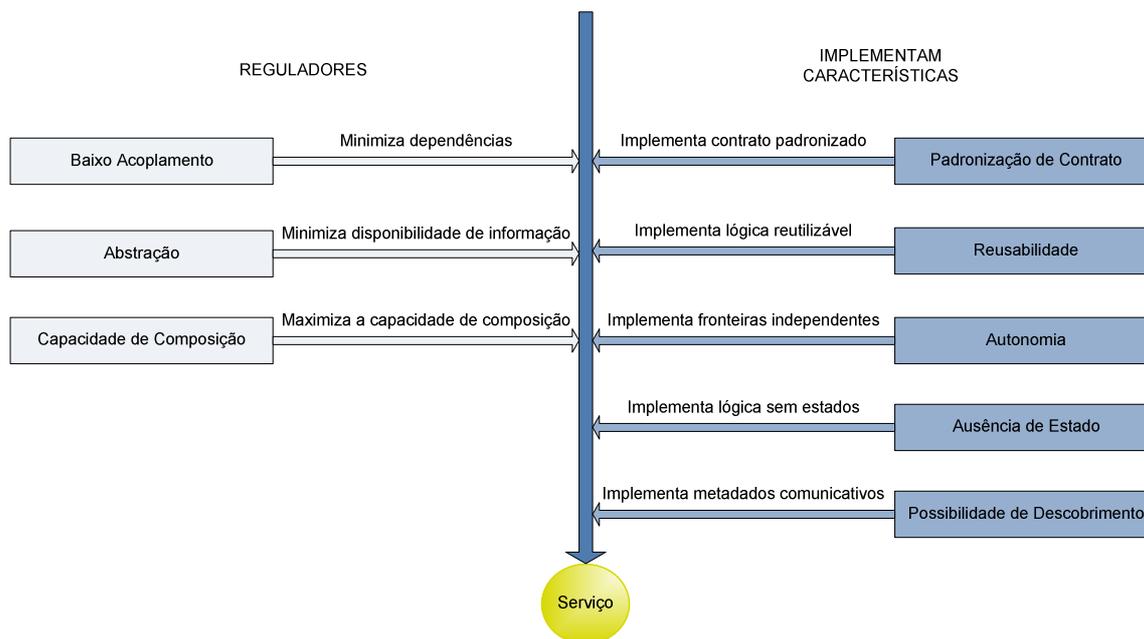
Consiste em garantir que o serviço expõe o mínimo necessário de informações a seu respeito. Desta forma, se este princípio for seguido corretamente, um serviço criado em *Java*, por exemplo, poderá ter sua implementação trocada para *C#* sem impactos aos demais serviços.

#### c) Capacidade de Composição (*Composability*)

De modo a permitir que os serviços executem funcionalidades complexas e completas dentro de um determinado negócio, os serviços devem ter a capacidade de serem compostos.

### **2.2.3. Princípios na Elaboração de um Serviço**

Para desenvolvimento de um serviço, estes princípios devem ser levados em consideração. Cada princípio tem uma diferente atuação sobre a solução final, como pode ser visto na Figura 2.3.



**Figura 2.3 - Atuação de Princípios na Elaboração de um Serviço [6]**

Ao se analisar cada princípio individualmente, pode-se perceber que há influências dele com alguns outros, de forma que ao se tentar seguir um determinado princípio, outro estará sendo afetado indiretamente. Por exemplo, ao buscar aumentar a reusabilidade entre serviços, se está favorecendo a capacidade de composição dos mesmos.

Existem vários relacionamentos entre todos os princípios, no entanto foge ao escopo deste trabalho detalhar cada um deles. Muita informação a respeito pode ser encontrada em [6] e [9].

### 2.3. Modelos de Serviços

Ao se modelar serviços, fica claro que eles podem ser classificados dependendo do modelo de lógica que encapsulam, do potencial de reuso que esta lógica tem e de como ela se relaciona com os domínios existentes dentro do

ambiente empresarial [6]. Desta forma, os serviços podem ser classificados nos seguintes modelos:

- Serviços de Tarefa
- Serviços de Entidade
- Serviços de Utilidade

A seguir descreveremos cada um.

### **2.3.1. Serviços de Utilidade**

Tratam-se de serviços que contém lógica independente de negócio, ou seja, responsáveis por realizar atividades que podem ser aplicadas a diversos tipos de negócio. Desta forma, possuem um potencial de reuso bastante elevado.

Algumas funcionalidades que estes serviços podem contemplar seriam registro de *logs*, autenticação de usuários ou tratamento de exceções.

### **2.3.2. Serviços de Entidade**

São os serviços relacionados às entidades envolvidas no negócio, como, por exemplo, funcionários, clientes, fornecedores, produtos, etc. Estes serviços também possuem um alto grau de reuso, porém menor do que os Serviços de Utilidade, devido à sua relação mais próxima com o negócio.

### **2.3.3. Serviços de Tarefas**

Constituem nos serviços responsáveis pelos processos específicos de cada negócio, como, por exemplo, uma análise de lucro. Nestes serviços, pode ser necessário obter informações das várias entidades do negócio, bem como realizar

tarefas atividades mais genéricas, de forma que serviços de Entidade e de Utilidade podem ser orquestrados para auxiliar na execução da Tarefa.

Por se tratarem de serviços relacionados a processos de negócio específicos, possuem baixo potencial de reuso. Na Figura 2.4 temos ilustrada a hierarquia dos serviços de acordo com cada modelo, destacando o sentido do crescimento do potencial de reuso.

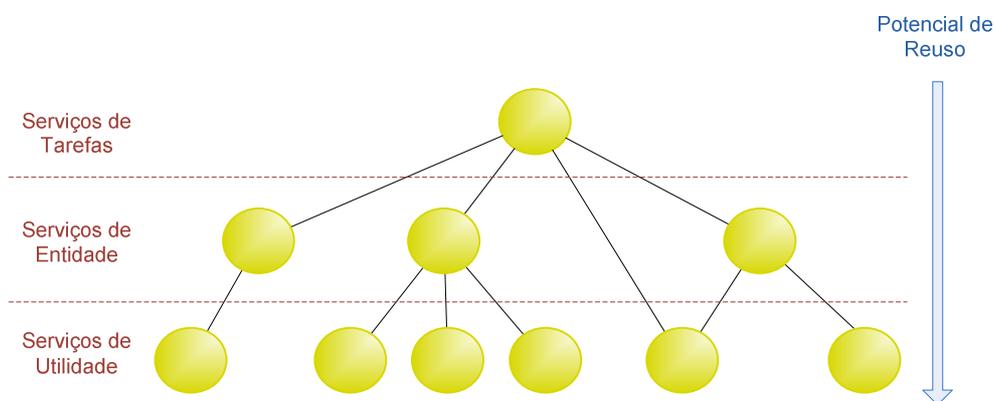


Figura 2.4 - Hierarquia dos Tipos de Serviços [6]

## 2.4. Tecnologias Utilizadas

Para que se possa construir uma arquitetura orientada a serviços de acordo com tudo o que foi visto até agora, algumas tecnologias precisam ser utilizadas.

### 2.4.1. Web Services

Trata-se de um tipo específico de serviço que se auto-contém e se auto-descreve. Suas funcionalidades podem ser encontradas e acessadas por outras aplicações utilizando padrões abertos [10].

A tecnologia de *Web Services* é a mais utilizada para a criação de serviços em SOA por possuir várias características que vão ao encontro dos princípios pregados pelo paradigma de orientação a serviços.

#### a) XML

XML, acrônimo de *Extensible Markup Language*, trata-se de uma metalinguagem, ou seja, uma linguagem utilizada para descrever outras linguagens [14].

Tal linguagem descreve documentos XML, que são documentos em um formato de texto simples e bastante flexível elaborado originalmente para atender aos desafios da publicação eletrônica de larga escala [13]. Estes documentos são constituídos de *tags*, que por sua vez podem encapsular dados ou outras *tags*. Na Figura 2.5 tem-se um exemplo desta estrutura.

```
<?xml version="1.0" encoding="UTF-8"?>
<Curriculo>
  <DadosPessoais>
    <NomeCompleto>Fulano de Tal</NomeCompleto>
    <DataNascimento>23-07-80</DataNascimento>
    <Contatos>
      <Endereco>
        <Rua>Av. Domingos Ferreira</Rua>
        <Num>1234</Num>
        <Cidade>Recife</Cidade>
        <Pais>Brasil</Pais>
      </Endereco>
      <Telefone>1234-5678</Telefone>
      <CorreioEletronico>fulano@email.com</CorreioEletronico>
    </Contatos>
    <Nacionalidade>Brasileiro</Nacionalidade>
    <Sexo>M</Sexo>
  </DadosPessoais>
  <Objetivo>Atuar em TI</Objetivo>
  <Experiencia>
    <Cargo>Suporte de TI</Cargo>
    <Empregador>Empresa, Recife - PE</Empregador>
  </Experiencia>
  <Formacao>Superior Completo</Formacao>
</Curriculo>
```

Figura 2.5 - Exemplo de arquivo XML

A extensibilidade e a natureza estruturada de XML permitem sua utilização na comunicação entre diferentes sistemas, que de outra forma não poderiam se comunicar [14]. Devido principalmente a estas características, *Web Services* normalmente são implementados utilizando XML como formato padrão de troca de dados.

#### b) Esquemas

Para padronizar o formato das mensagens XML, faz-se uso de estruturas bem definidas chamadas Esquemas. O formato mais comum de esquemas é o XSD, acrônimo de *XML Schema Definition*.

#### c) WSDL

WSDL, do inglês *Web Services Description Language*, constitui em uma linguagem para descrição de *Web Services* baseada em XML. Ou seja, WSDL é a linguagem utilizada para elaborar os contratos dos serviços construídos em *Web Services*.

Existem basicamente dois tipos de WSDL: abstrato e concreto. O primeiro define apenas as funcionalidades do serviço, ignorando onde o mesmo estará localizado para acesso, enquanto o segundo possui também estas informações. A idéia por trás disso é que caso um serviço tenha apenas que mudar de lugar, o seu WSDL abstrato pode ser reutilizado para gerar um novo WSDL concreto.

#### d) UDDI

UDDI (*Universal Description, Discovery and Integration*) é uma especificação para registros de informação distribuídos de *Web Services*. UDDI define uma forma de publicar e descobrir informações sobre *Web Services* [15].

#### e) SOAP

SOAP, do inglês *Simple Object Access Protocol*, é um protocolo para troca de mensagens que fornece interoperabilidade entre uma grande variedade de plataformas através de uma especificação generalizada para invocação de métodos em objetos e componentes, fazendo uso de chamadas XML, geralmente sobre HTTP [16].

Uma das grandes vantagens de SOAP sobre outros protocolos é a sua capacidade de comunicar aplicações executando em sistemas operacionais distintos, utilizando diferentes tecnologias e linguagens de programação [17]. Por isso que SOAP geralmente é utilizado sobre HTTP, um protocolo que todo navegador e servidor de internet conhece.

Existem alternativas a SOAP, como, por exemplo, o REST. Cada um possui vantagens e desvantagens, no entanto SOAP ainda é mais utilizado, sendo considerado o protocolo padrão para utilização com *Web Services*.

Na Figura 2.6 está ilustrado o relacionamento destas tecnologias envolvendo *Web Services*.

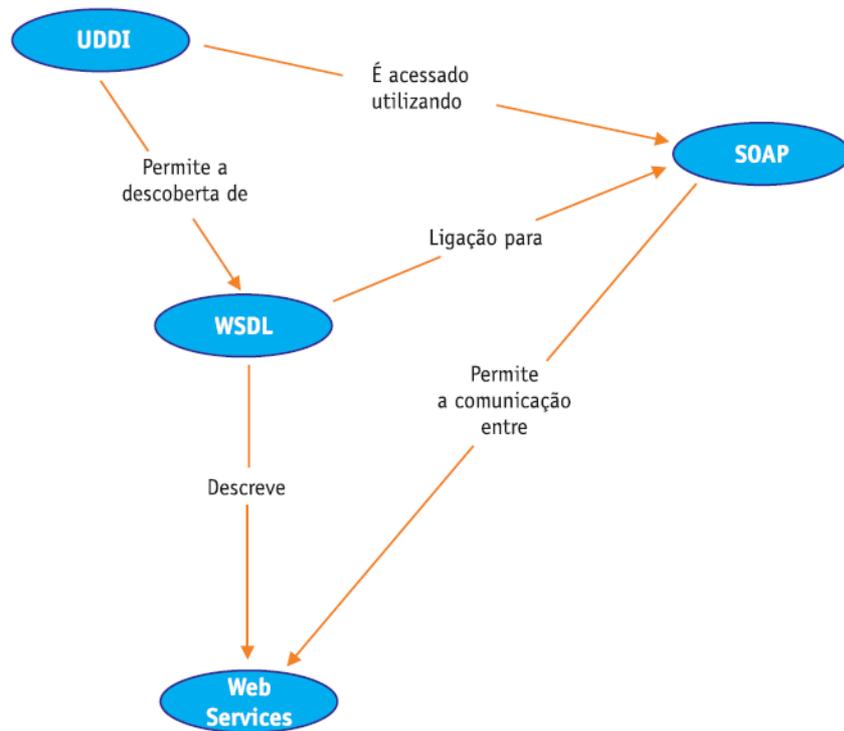


Figura 2.6 - Relacionamento entre *Web Services* e outras tecnologias auxiliares [18]

#### f) WS-BPEL

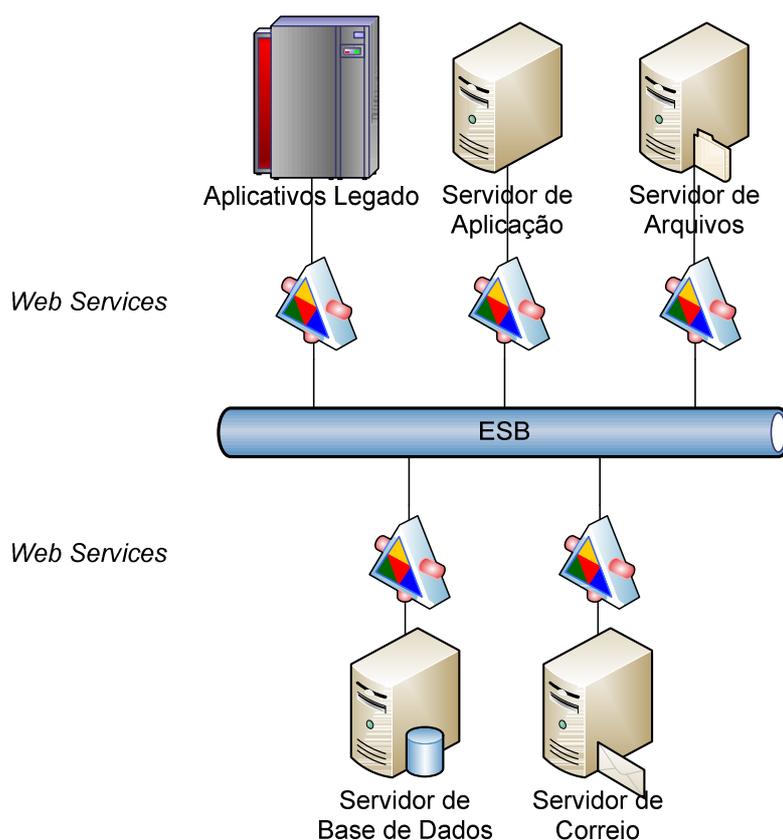
WS-BPEL, acrônimo de *Web Services Business Process Language*, é uma linguagem para definição de interações entre *Web Services*, compondo processos de negócios. Este processo de criar tais relacionamentos é chamado de orquestração, sendo WS-BPEL, ou simplesmente BPEL, uma das formas de descrevê-lo.

#### 2.4.2. Enterprise Service Bus (ESB)

Consiste de uma aplicação que fornece a SOA uma infra-estrutura de conectividade para permitir a comunicação entre as aplicações executando em diferentes plataformas, escritas em diferentes linguagens e utilizando diferentes modelos de programação. Funciona como um uma versão de um barramento

tradicional de um PC, movendo dados entre aplicações através de um caminho comum [11].

Na Figura 2.7 temos um exemplo de utilização de ESB em um ambiente corporativo.



**Figura 2.7 - Exemplo de utilização de ESB**

Geralmente os ESBs são desenvolvidos em *software*, podendo ser classificados como *Middlewares*, ou seja, uma camada de *software* que permite a comunicação entre outros *softwares*. No entanto, um ESB normalmente oferece muitas outras funcionalidades bastante úteis para SOA, como roteamento de mensagens, orquestração, recursos de segurança, dentre outros [19].

O escopo dos ESBs ainda é algo bastante variável, enquanto alguns possuem uma infinidade de recursos, outros apenas servem para simples trocas de mensagens, e ambos se auto-denominam ESB da mesma forma. Gigantes do mundo da Tecnologia da Informação, como IBM e Oracle, estão investindo bastante no desenvolvimento de suas soluções proprietárias de ESB, porém também há soluções gratuitas e *open source*.

## 2.5. Estratégias de Entrega de SOA

A maneira escolhida na implantação de uma Arquitetura Orientada a Serviços pode ser determinante no sucesso ou não da mesma. Dependendo das necessidades específicas de cada um, diferentes abordagens podem ser escolhidas. A seguir descreveremos o ciclo de vida básico, bem como algumas abordagens específicas.

### 2.5.1. Ciclo de Vida Básico

O ciclo de vida SOA trata-se das etapas básicas a serem realizadas durante a construção de serviços numa arquitetura orientada a serviços. Tal ciclo está ilustrado abaixo na Figura 2.8.

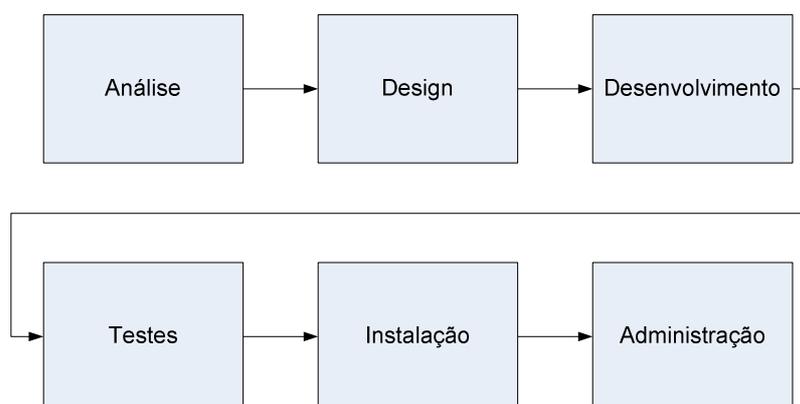


Figura 2.8 - Ciclo de Vida Básico [9]

a) Análise

Consiste em analisar as necessidades e iniciar a modelagem de serviços candidatos para que a solução proposta seja capaz de atingir seus objetivos.

b) *Design*

É a etapa de definição dos serviços, que determinará como os serviços irão ser compostos e orquestrados para resolver atividades mais complexas.

c) Desenvolvimento

Construção de fato dos serviços. Trata-se de uma etapa comum na produção de *software*, na qual alguma linguagem de programação será utilizada. No entanto, detalhes da implementação não devem ficar expostos, de modo que a implementação poderá mudar futuramente sem afetar o relacionamento entre os serviços.

d) Testes

Etapa responsável por garantir que os serviços criados realmente cumprem com aquilo que se espera deles.

e) Instalação

Deve colocar os serviços para serem utilizados em produção. Leva em consideração como distribuir estes serviços na infra-estrutura disponível.

## f) Administração

Uma vez que os serviços estejam prontos para serem utilizados, é necessária uma preocupação para que eles continuem funcionando perfeitamente. A administração é responsável por garantir isso.

### 2.5.2. Estratégias Específicas

Existem diferentes estratégias utilizadas na implantação de SOA. A escolha da estratégia adequada deve levar em consideração aspectos como a qualidade desejada da arquitetura final, tempo disponível e quantidade de recursos disponíveis. A seguir cada uma será definida.

#### a) *Top-down*

Esta estratégia consiste em realizar inicialmente uma análise profunda do modelo de negócios, podendo até gerar mudanças no mesmo, para depois modelar os serviços de forma totalmente alinhada com o negócio. Ou seja, trata-se de partir de uma visão em alto nível de como tudo funciona no ambiente para depois detalhar em mais baixo nível como tudo será feito. Na Figura 2.9 temos as etapas da estratégia *top-down*.

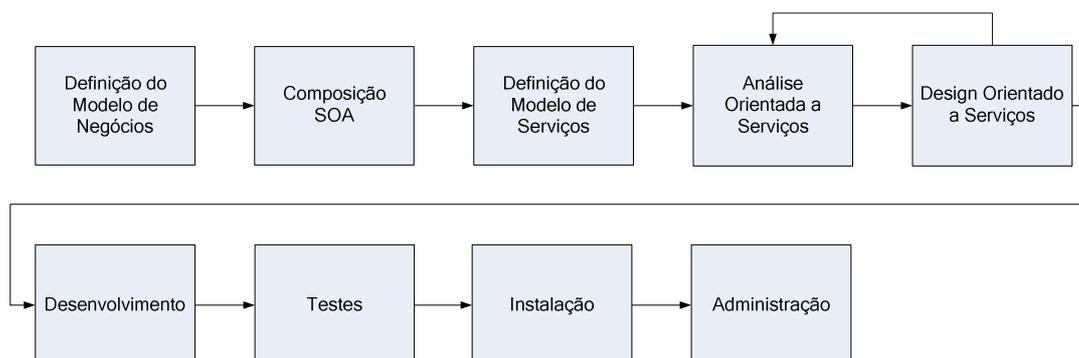


Figura 2.9 - Estratégia *Top-Down* [9]

Esta estratégia pode ser considerada a forma ideal de se implantar uma Arquitetura Orientada a Serviços, do ponto de vista da qualidade do resultado final. No entanto, ela demanda bastante tempo e recursos, além de causar grandes impactos no modo como uma empresa funciona cotidianamente, o que pode requerer um período de adaptação.

#### b) *Bottom-up*

Ambientes corporativos normalmente possuem uma grande quantidade de *software* legado que não pode deixar de ser usado repentinamente. Uma forma de adotar SOA e continuar utilizando estes *softwares* é criar serviços que encapsulam estes produtos, são os chamados *wrappers*. As etapas desta estratégia são fundamentalmente apenas as do ciclo de vida básico, ilustrado anteriormente na Figura 2.8.

Esta estratégia provém uma forma mais rápida de se implementar orientação a serviços dentro de um ambiente corporativo, fazendo uso do legado existente, economizando desta forma boa parte de desenvolvimento. No entanto, o resultado final obtido com esta estratégia dificilmente estará totalmente de acordo com todos os princípios de SOA.

## **2.6. Benefícios**

A utilização de um modelo de Arquitetura Orientada a Serviços pode trazer uma série de benefícios para quem opte por adotá-la. Estes benefícios podem ser vistos sob diferentes perspectivas, tanto do ponto de vista da tecnologia quanto do negócio [20]. Descreveremos a seguir estes benefícios de forma separada, embora ganhos de tecnologia possuam impactos diretos e indiretos sobre o negócio.

## 2.6.1. Benefícios de Tecnologia

### a) Flexibilidade

Com a utilização do conceito de orientação a serviços, obtém-se uma grande flexibilidade dentro da arquitetura. Por exemplo, caso um determinado processo tenha que mudar, uma alteração na composição de serviços existentes neste ou a modificação em um destes serviços poderá resolver o problema sem impactar nos demais serviços.

### b) Independência de Tecnologia e Fabricantes

Devido à abstração promovida pela orientação a serviços, as tecnologias subjacentes utilizadas na implementação de tais serviços podem ser as mais diversas de serviço para serviço sem que isso cause problemas. Isto é bastante relevante, pois dependendo da funcionalidade desejada, uma tecnologia pode ser melhor que outra em diferentes circunstâncias, e isso permite que a melhor seja escolhida.

### c) Integração de Funcionalidades Internas

Nas arquiteturas convencionais, as atividades dos processos de negócio, aplicativos e dados estavam fechados em locais independentes e incompatíveis, com alto custo de manutenção. Com SOA, os usuários não precisam se conectar a múltiplos sistemas nem integrar resultados manualmente, pois os dados para as atividades de negócio são oferecidos como serviços integrados.

### d) Integração de Dados

Além de promover a integração entre sistemas e funcionalidades, SOA possibilita um *framework* ideal para integração de dados, devido à camada de

abstração que faz com que seus componentes e serviços sejam encapsulados e reusados sem grande esforço de desenvolvimento de código manual [21].

#### e) Reuso no Desenvolvimento de Aplicações

Do ponto de vista do desenvolvedor, uma das grandes vantagens é que os serviços devem ser reutilizáveis. Desta forma o desenvolvimento será bastante facilitado, trazendo maior agilidade e aumentos de produtividade.

#### f) Facilidade ao Desenvolvimento de Portais

Várias empresas vêm adotando em larga escala a utilização de portais corporativos como centralizador e distribuidor de informações sobre a própria empresa. Com a utilização de orientação a serviços, a construção destes portais se torna bastante simplificada, pois o portal poderá acessar facilmente os serviços que precisar para obter os dados que irá disponibilizar.

### **2.6.2. Benefícios de Negócio**

#### a) Facilidade de Integração com Outras Empresas

Devido à grande heterogeneidade normalmente existente nos sistemas computacionais de diferentes empresas, realizar integração entre estas sempre foi bastante difícil. Com a utilização de SOA, o que passa a existir são diferentes serviços em cada empresa, que podem ser compostos para realizarem novos processos de negócio.

#### b) Fornecimento de Serviços a Clientes

Pode ser bastante comum que o cliente tenha papel ativo dentro de determinados processos de negócios, dependendo da área de atividade da empresa. Serviços podem ser criados ou compostos para atender a estes clientes

dentro da própria arquitetura orientada a serviços, o que além de facilitar bastante o desenvolvimento dos mesmos através da utilização de outros serviços já existentes, pode proporcionar novas oportunidades de negócio.

#### c) Maior Eficácia no Uso de Serviços de Provedores Externos

Da mesma forma que fornecer serviços a clientes se torna mais fácil, a utilização de serviços oferecidos por outros provedores também é favorecida ao se utilizar SOA. Pode-se dizer que é como incorporar mais um serviço ao repositório de serviços da empresa.

#### d) Redução de Custos de TI em Longo Prazo

Com o amadurecimento da arquitetura desenvolvida ao longo do tempo, chegar-se-á a um estágio em que as novas atividades de negócio serão em sua maioria compostas por serviços já existentes, desta forma o custo para desenvolvimento destes novos processos será bastante reduzido.

## 3. PEQUENOS NEGÓCIOS

---

O presente capítulo tem o intuito de apresentar informações sobre Pequenos Negócios, de modo a fornecer subsídios a este trabalho para o desenvolvimento de uma adequação por parte de SOA. Inicialmente será feita uma caracterização deste tipo de negócio. Em seguida serão discutidas vantagens e desafios enfrentados. Por fim, será feita uma análise da utilização de Tecnologia da Informação neste contexto.

### 3.1. Caracterização

Pequenos negócios variam bastante sob diferentes aspectos, sendo uma tarefa difícil colocá-los em uma definição única. Diversos autores de diferentes países apresentam uma variedade de definições para pequeno negócio [22], enquanto uns analisam do ponto de vista da quantidade de funcionários, outros levam mais em conta o grau de importância do negócio dentro de seu ramo de atividade e o seu faturamento. No entanto, estas características estão relacionadas, de forma que um pequeno negócio geralmente possui até 50 pessoas trabalhando, dentre as quais uma ou duas são as responsáveis por tomar as decisões – quase sempre os donos – e estas empresas normalmente não possuem dominância dentro de seu ramo de atividade.

Apesar de seu tamanho, tais negócios são de extrema importância para a economia global. No Brasil, especificamente, de acordo com o Censo de 2002 do Instituto Brasileiro de Geografia e Estatística (IBGE), as micro e pequenas empresas possuíam 20% de participação no Produto Interno Bruto (PIB) e geravam 60% do total de empregos no setor privado [23].

Este tipo de negócio normalmente trata-se de uma empresa privada, composta por uma pequena sociedade, podendo ser inclusive uma propriedade individual. Algumas de suas características são a independência de ação, existência de diferentes estruturas organizacionais e também uma grande variedade de estilos de gerenciamento [24], não se tratando, no entanto, apenas de versões em menor escala de grandes negócios. Embora o tamanho seja de fato uma diferença, pequenos negócios possuem uma série de peculiaridades que os distinguem [25].

## **3.2. Vantagens**

Pelo formato da sua estrutura, pequenos negócios possuem algumas vantagens que devem ser corretamente exploradas para seu sucesso. A seguir descreveremos algumas destas.

### **3.2.1. Pessoalidade**

Por serem compostos por um número mais restrito de pessoas, estes negócios conseguem oferecer aos clientes um nível de atenção mais próximo e pessoal, que possibilita o estabelecimento de laços de confiança maiores. Alguns grandes empresas inclusive vêm tentando englobar esta abordagem para seus clientes [28].

### **3.2.2. Excelência em Atividades Específicas**

Pequenos negócios podem se especializar em determinado segmento específico, de forma que consiga obter um nível de excelência dentro daquela atividade superior ao de grandes empresas [29], que normalmente atuam em um domínio mais amplo.

### **3.2.3. Eficiência Operacional**

Por possuírem uma máquina operacional menor em comparação a grandes corporações, pequenos negócios demandam que seus funcionários tenham alto grau de eficiência. Isto proporciona que a empresa como um todo seja capaz de realizar suas atividades de forma ágil e eficaz.

A eficiência operacional é crucial para a sobrevivência de um pequeno negócio, de modo que além de ser uma vantagem, trata-se de um requisito para o mesmo.

## **3.3. Desafios**

Pequenos negócios enfrentam uma série de dificuldades, ocasionando que muitos deles fechem rapidamente. Um estudo realizado pelo IBGE em 2005 constatou que as pequenas empresas são as mais atingidas pela passagem do tempo: entre as empresas criadas em 1997 com até quatro pessoas, apenas pouco mais de 50% permaneciam ativas em 2005 [30]. A seguir descreveremos alguns desafios que estes negócios precisam superar para que possam prevalecer.

### **3.3.1. Falta de Capital**

Este é um fator limitante no crescimento de um pequeno negócio, pois muitas vezes há ótimas idéias e iniciativas sendo desenvolvidas, mas que ficam emperradas pela falta de recursos financeiros para executá-las. A utilização de empréstimos e outras modalidades de crédito costumam ser utilizadas para contornar este problema, no entanto caso algo não ocorra como planejado, pode-se estar decretando o fim do negócio.

### **3.3.2. Planejamento**

Planejamento adequado é a melhor forma de permitir que um pequeno negócio possa crescer com sustentabilidade, pois através dele pode ser possível detectar inconsistências que necessitam melhor gerenciamento [31]. Alguns pequenos negócios conseguem expandir bastante em um curto intervalo de tempo, sem que haja um planejamento adequado para isso. Isto pode levar a uma perda de controle do negócio, que pode ser irreversível.

O não cumprimento de um planejamento também pode resultar em vários problemas. À medida que novas oportunidades apareçam ao longo do tempo, pode ocasionar que um planejamento feito inicialmente seja descartado e não seja refeito adequadamente para priorizar as demandas de momento, o que pode levar à perda de foco dos negócios.

### **3.3.3. Inserção no Mercado**

Conseguir clientes é sempre um desafio para negócios de pequeno porte, pois geralmente há uma desconfiança por parte daqueles em relação à confiabilidade da empresa e qualidade do trabalho, uma espécie de preconceito. Tentar causar uma boa impressão, buscando mostrar competência naquilo que faz é um primeiro passo. Dependendo do ramo de atividade, algum investimento em marketing pode ser uma boa solução para tornar a empresa conhecida.

## **3.4. A TI para os Pequenos Negócios**

A participação da tecnologia da informação nos pequenos negócios vem crescendo bastante, de modo que é importante entender essa crescente demanda para poder conduzi-la da melhor forma.

### **3.4.1. Utilização**

As aplicações de pequenos negócios geralmente possuem baixa complexidade algorítmica e são compostas por entidades de negócios bem definidas, como em aplicações de gerenciamento de impostos, vendas, contabilidade, entre outras [26]. Esta facilidade deve ser aproveitada para permitir um rápido desenvolvimento destas aplicações e até uma maior facilidade de alinhamento da mesma com a atividade de negócio relacionada.

Alguns fatores que influenciam na adoção de TI por parte de pequenos negócios são o setor no qual o negócio está envolvido, a natureza dos produtos ou serviços oferecidos, o quão ligada a empresa está com fornecedores e parceiros, e até o grau de instrução dos donos [27].

### **3.4.2. Objetivos**

A adoção de soluções de tecnologia da informação em pequenos negócios costuma ser bastante analisada e questionada pelos donos e gestores destes, pois se tratam de investimentos que nem sempre tem seu retorno claro para a empresa, principalmente em curto prazo. Os principais objetivos que estes negócios têm ao utilizar TI são a busca de maior eficiência operacional, aumento de mercado, melhoria da relação com clientes e outras organizações e ganho de vantagem estratégica [22].

#### **a) Ganho de Eficiência Operacional**

A execução de atividades no menor tempo possível é algo que todas as empresas sempre buscam, e é justamente para isso que se desenvolveu a Informática.

#### b) Aumento de Mercado

A partir do uso de soluções de TI, um pequeno negócio pode atingir um nível organizacional maior que permita sua expansão e conseqüente aumento de mercado, possibilitando, por exemplo, a criação de filiais sem desalinhamento do negócio.

#### c) Melhoria da Relação com o Cliente

Em negócios que lidam diretamente com clientes, sobretudo as prestadoras de serviço, a satisfação do cliente é fundamental. Com o uso de Tecnologia da Informação é possível oferecer um atendimento mais eficiente, que pode ser decisivo na escolha do cliente entre uma empresa ou sua concorrente.

#### d) Ganho de Vantagem Estratégica

A partir do uso de tecnologia da informação, o modelo organizacional pode ser remodelado e melhorado, possibilitando que novas estratégias de negócio sejam desenvolvidas.

### **3.4.3. Riscos**

Dentro de um pequeno negócio, nem tudo precisa passar a ser resolvido através da Tecnologia da Informação, apenas aquilo que realmente traga benefícios reais. Caso a TI cresça mais do que a empresa em si, ela pode passar a ser um grande fardo, o que pode trazer conseqüências bastante drásticas.

A questão do suporte em cima de soluções que estão sendo desenvolvidas também é bastante crítica. Muitas vezes o suporte não é planejado adequadamente, tanto para mais, quanto para menos. Desta forma pode-se ter um gasto

desnecessário com uma solução de suporte superdimensionada, ou simplesmente tudo pode parar e não haver ninguém responsável para recuperar a solução.

## 4. ADOTANDO SOA EM PEQUENOS NEGÓCIOS

---

Este capítulo analisará como SOA pode ser adotado em pequenos negócios. Para tal, inicialmente serão vistos critérios para escolha dos serviços a serem criados em SOA. Em seguida discutiremos a aplicabilidade de SOA a alguns diferentes tipos de pequenos negócios. Por fim, será derivada uma estratégia para adoção de SOA nestes pequenos negócios.

### 4.1. Escolha dos Serviços na Adoção de SOA

Na seção 2.5 do Capítulo 2 foram apresentadas algumas estratégias para a implantação de SOA. Na prática, tais estratégias costumam ser aplicadas em várias etapas a fragmentos das atividades de negócio. A seguir discutiremos duas formas de escolha destas atividades.

#### 4.1.1. *Inside-out*

No processo de adoção de uma Arquitetura Orientada a Serviços, as grandes empresas costumam optar por uma adoção gradual, escolhendo estrategicamente subconjuntos de seus processos. Tal escolha é feita de forma a minimizar os impactos da adoção, o que faz com que nestas empresas ela se dê de dentro para fora (*inside-out*), ou seja, escolhendo inicialmente processos internos da empresa, sobre os quais se tem um maior controle. Tal abordagem está ilustrada na Figura 4.1.

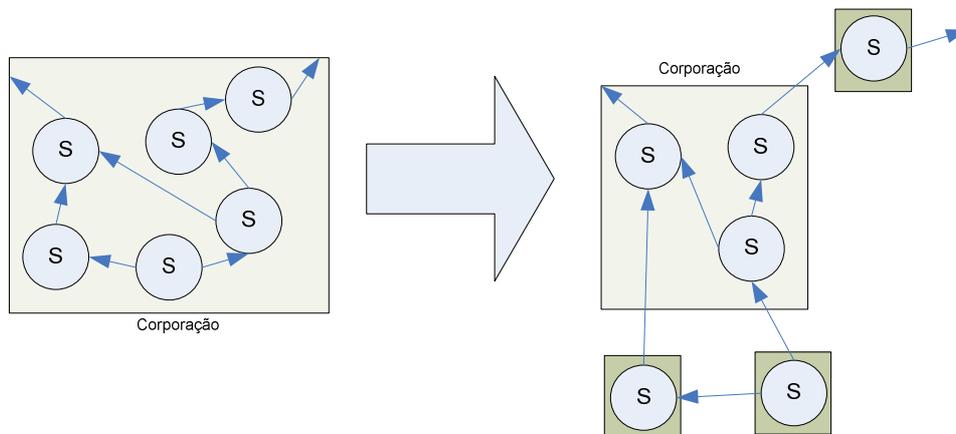


Figura 4.1 - Adoção Inside-out [1]

#### 4.1.2. Outside-in

A adoção nos moldes de *Inside-out* não é adequada para utilização em Pequenos Negócios, pois os modelos de negócios destes são mais focados justamente em integrações *business-to-business* (B2B) com empresas parceiras e clientes [3], além de interações diretamente com os consumidores na forma de *business-to-consumer* (B2C). Por esse motivo, a adoção de SOA para estes negócios deve ser mais voltada justamente para o seu mundo exterior.

Uma adoção que se inicia de fora para dentro (*outside-in*) é focada justamente na integração com entidades externas, podendo-se utilizar de serviços fornecidos por parceiros e também visando disponibilizar novos serviços para estes. A idéia por trás disso é que se possa formar um ecossistema entre negócios [1], com as interações entre seus participantes ocorrendo a partir dos serviços disponibilizados por cada um. Como ilustrado na Figura 4.2, não existem apenas pequenos negócios neste ecossistema, mas também há grandes corporações, que costumam atuar em conjunto com pequenos negócios.

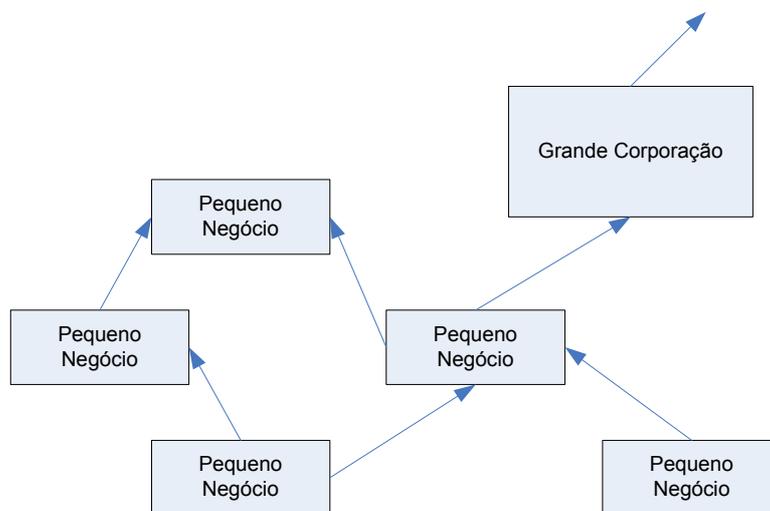


Figura 4.2 - Ecossistema de Negócios [1]

## 4.2. Aplicabilidade de SOA em Pequenos Negócios

A idealização da orientação a serviços foi desenvolvida com o foco em resolver dificuldades enfrentadas pelas grandes corporações. No entanto, os benefícios que arquiteturas baseadas neste paradigma oferecem também são de interesse de muitos pequenos negócios.

Como visto na seção 4.1, a utilização de arquiteturas orientadas a serviço nestas empresas não deve ocorrer da mesma forma que normalmente ocorre nas grandes corporações. No entanto, nem todo pequeno negócio conseguirá tirar benefícios que justifiquem a utilização de orientação a serviços, mas para outras estes benefícios serão bastante claros e proveitosos. A seguir analisaremos certos aspectos sobre a aplicabilidade de SOA a alguns diferentes tipos de pequenos negócios. Vale a pena ressaltar que utilizar SOA não é trivial, de modo que sua utilização é recomendada caso o negócio realmente esteja enfrentando problemas ou planeje se expandir.

#### **4.2.1. Comércio Varejista**

A Tecnologia da Informação utilizada neste setor não costuma ser muito complexa nem possuir muitos sistemas interligados. Normalmente faz-se uso apenas de sistemas bastante simples de automação dos processos de venda e controle de clientes com algum controle financeiro agregado.

Em um cenário simples como o descrito acima, a utilização de SOA não irá trazer grandes benefícios. No entanto, caso o negócio tenha pretensões de se expandir com ganho de agilidade operacional, a utilização de orientação a serviços pode servir como instrumento para tal.

#### **4.2.2. Comércio Eletrônico (*E-commerce*)**

Atualmente estão surgindo várias lojas virtuais trabalhando com venda de produtos na *Internet*, através do chamado *e-commerce*. No entanto, da mesma maneira que muitas estão surgindo, outras rapidamente estão fechando, sendo uma das causas disso justamente a falta de estrutura e organização adequadas para suportar o negócio da melhor forma.

Por trabalharem basicamente em cima de operações B2C e B2B, a utilização de SOA neste tipo de negócio possibilita uma redução de custos de integração em até 10 vezes, retirando barreiras com o objetivo de fornecer aos consumidores canais de interação consistentes [32].

#### **4.2.3. Pequenas Indústrias**

Indústrias costumam trabalhar com processos bastante específicos, sendo estes normalmente auxiliados por alguma forma de tecnologia da informação. No entanto, tais processos costumam sofrer otimizações e pequenas alterações com

certa freqüência, o que costuma gerar dificuldades no alinhamento da TI com os mesmos.

Devido à característica de SOA promover facilidade no alinhamento entre TI e negócio, sua utilização na área industrial é encorajada. Através de modificações na composição de serviços e alterações em serviços individualmente, é possível reproduzir mais facilmente as modificações realizadas no processo na TI.

#### **4.2.4. Prestação de Serviços**

Existe uma grande diversidade de áreas de prestação de serviços por parte de negócios, muitas delas podendo ser feitas através de TI. Para oferecer tais serviços, pode ser utilizada a orientação a serviços, que permite um maior controle sobre todos os serviços fornecidos.

Mesmo em algumas atividades que não seja possível oferecer diretamente o serviço em forma de TI, indiretamente serviços de TI podem ser utilizados para auxiliar e suportar de melhor forma seguindo os princípios de SOA.

#### **4.2.5. Turismo**

Os negócios relacionados a turismo costumam envolver várias entidades, como agências de viagens, companhias aéreas, hotéis, entre outros. A integração de tudo isso pode resultar em problemas caso não haja uma preparação adequada para isso.

Por essas características, este ramo de atividades pode se beneficiar bastante com a orientação a serviços. Grandes companhias aéreas e redes hoteleiras já vêm disponibilizando suas informações na forma de serviços, de forma que cabe

também às empresas de menor porte deste setor se inserir neste contexto para utilizá-los da melhor forma.

#### **4.2.6. Desenvolvimento de Software**

Devido à onda SOA que se instaurou na TI [33], as empresas de desenvolvimento de software que não olharem atentamente para este conceito de arquitetura começarão a perder clientes importantes. Isto devido às grandes empresas estarem adotando SOA em larga escala, o que faz com que passem a requerer que suas novas aplicações sejam aderentes a orientação a serviços.

Além de desenvolver de acordo com os princípios de SOA, estas empresas podem passar a se estruturar também seguindo este modelo de arquitetura. Processos de desenvolvimento de software costumam ser bastante complexos, e a orientação a serviços pode ajudar neste contexto.

#### **4.2.7. Franquias**

Franquias podem ser consideradas um caso a parte dentro de pequenos negócios, se encaixando dentro de alguns dos tipos citados anteriormente. Mas o que as faz serem tão particulares é que a franquia como um todo geralmente é uma grande empresa, enquanto que as unidades locais, de diferentes donos, são pequenas empresas.

Os franqueadores geralmente são responsáveis por fornecer as soluções de TI utilizadas pelos franqueados. É importante observar que se trata de um desenvolvimento feito por uma grande empresa para ser utilizado em pequenas. Este é um bom cenário para utilização de SOA, pois pode fazer uso das vantagens de uma grande empresa, como maior disponibilidade de recursos e também das

vantagens de pequenos negócios, como uma menor complexidade dos modelos de negócios.

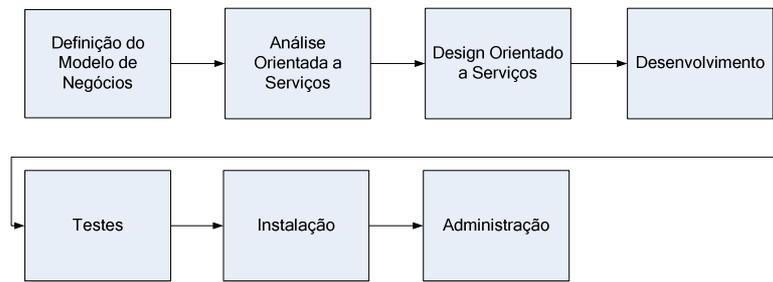
### 4.3. Estratégia para SOA em Pequenos Negócios

Devido à agilidade requerida em pequenos negócios, a adoção de SOA na estratégia *top-down* tradicional não é adequada, pois demandará muito tempo e recursos, que são sérias restrições para estas entidades. Já a *bottom-up* se enquadra melhor nestes dois critérios, no entanto os objetivos desejados podem não ser alcançados com a permanência de modelos e legados já existentes.

Neste contexto, se faz necessária uma estratégia alternativa que permita uma arquitetura de serviços consistente e aderente a SOA, que possa, no entanto, ser desenvolvida rapidamente e com menos recursos. Diante deste cenário, propomos neste trabalho a utilização de um modelo híbrido baseado em uma redução do *top-down*, porém como alguns serviços são externos, estes seriam encaixados de forma *bottom-up*. Desta forma será possível entregar serviços mais rapidamente, ou seja, *quick wins*.

A utilização de um modelo híbrido não é novidade, existindo a estratégia conhecida como ágil [9]. No entanto normalmente ela é feita com as duas estratégias em paralelo para todo o contexto de serviços, o que demanda ainda mais recursos.

Na Figura 4.3 está ilustrado o fluxo desta estratégia, que consiste em definir nas fases iniciais quais e como serão os serviços e em seguida, de forma iterativa, desenvolver subconjuntos de serviços de acordo com a prioridade definida nas etapas iniciais.



**Figura 4.3 – Estratégia para Adoção de SOA em Pequenos Negócios**

Observa-se que o fluxo de etapas não muda muito, sendo similar ao *top-down*, no entanto com algumas etapas a menos, que foram incorporadas dentro de outras. A seguir descreveremos um pouco de cada etapa de acordo com esta estratégia.

#### **4.3.1. Definição do Modelo de Negócios**

Esta etapa tem como objetivo gerar uma visualização macro do modelo de negócios envolvido. Pode ocorrer que este modelo já esteja bem definido de antemão, devendo então ser revisto aqui. Com o modelo em mãos, será possível definir quais são as prioridades do negócio, e onde que SOA trará mais benefícios e será mais aplicável.

O alinhamento dos serviços com o negócio envolvido depende diretamente da correta elaboração deste modelo. A partir dele serão preparados os serviços a serem criados para atender às necessidades por ele levantadas.

#### **4.3.2. Análise Orientada a Serviços**

Nesta etapa será montado o escopo de SOA a ser utilizado, com uma modelagem inicial dos serviços através de suas funcionalidades, gerando serviços

candidatos. Também aqui serão identificados os modelos de cada serviço: tarefa, entidade ou utilidade.

Por não possuir controle sob os serviços externos, estes poderão ser encapsulados dentro de novos serviços para garantir que a interface do serviço permaneça a mesma para os demais serviços.

#### **4.3.3. Design de Serviços**

Aqui ocorrerá a definição de cada serviço, ou seja, a definição de como cada um será construído e seus relacionamentos com outros serviços através da orquestração. Devem ser definidos esquemas para padronização da troca de mensagens, bem como as interfaces dos serviços a serem criados.

#### **4.3.4. Desenvolvimento**

Implementação propriamente dita dos serviços, na qual eles serão de fato construídos. Processos de desenvolvimento já utilizados para *softwares* em pequenos negócios podem ser adotados aqui também.

#### **4.3.5. Testes**

Como todo componente de *software*, os serviços precisam ser testados. Algumas coisas que podem ser analisadas são o seu correto funcionamento, desempenho, resposta sob condições de *stress*, aderência aos padrões adotados, etc.

#### **4.3.6. Instalação**

Os serviços devem ser instalados e configurados no ESB escolhido. Estas soluções podem chegar a custar centenas de milhares de dólares, dependendo do fornecedor, o que o tornaria impraticável para pequenos negócios.

No entanto, existem soluções *open source* de ESBs e de várias outras ferramentas para SOA que atendem às necessidades destes negócios. Uma das grandes vantagens das soluções *open source* é que se pode testá-las facilmente antes de usar e escolher a solução que melhor se adéqüe às necessidades de cada um.

Um detalhe sobre ESBs é que as funcionalidades do ESB, dependendo da solução escolhida, podem estar distribuídas em mais de uma aplicação, como *Application Servers* ou Registros baseados em UDDI.

É importante ressaltar que caso se opte posteriormente por uma troca de solução utilizada no ESB, os serviços desenvolvidos deverão continuar sendo os mesmos, sem necessidade de desenvolvê-los novamente. Apenas deverão ser instalados e configurados no novo ambiente.

#### **4.3.7. Administração**

Uma das grandes vantagens de SOA é a governança que ela proporciona sobre seus serviços, o que é levado adiante através da correta administração destes. Aqui os serviços serão monitorados e terão dados coletados sobre sua utilização, fornecendo assim elementos que permitam um maior controle e possibilitem futuras otimizações baseadas nestas informações.

## 5. ESTUDO DE CASO

---

No presente capítulo será desenvolvido um estudo de caso para um pequeno negócio utilizando-se das idéias apresentadas no Capítulo 4. Inicialmente será descrito o cenário a ser desenvolvido. Em seguida será realizada a criação de um processo de negócio.

### 5.1. Descrição

Por possuir características interessantes para a utilização de SOA de forma *outside-in*, foi escolhido como estudo de caso o ambiente de uma agência de viagens fictícia, que chamaremos daqui em diante de *TravelX*. Por tratarem diretamente com consumidores, agências de viagem necessitam oferecer seus serviços da melhor forma para estes, de forma ágil e eficiente.

Companhias aéreas já vêm oferecendo suas informações na forma de serviços, bem como algumas redes hoteleiras. Diante deste cenário, a utilização de orientação a serviços na *TravelX* permitirá a criação de processos automatizados para atendimento com qualidade a seus clientes. No entanto, não foi possível utilizar de serviços reais de empresas para estas funcionalidades, de forma que serão criados serviços para simular estes.

Agências de viagens também costumam vender seus pacotes através de parcelamentos e outras facilidades de crédito, o que torna necessária a consulta a sistemas de decisão de crédito, como a SERASA [34]. A empresa ConsultaCPF.com vende esta consulta na forma de *web service*, possibilitando que outras empresas automatizem este processo internamente. Este é um recurso bastante interessante para a *TravelX*, de forma que neste estudo de caso será utilizada uma conta de testes, que nos foi gentilmente cedida pela ConsultaCPF.com.

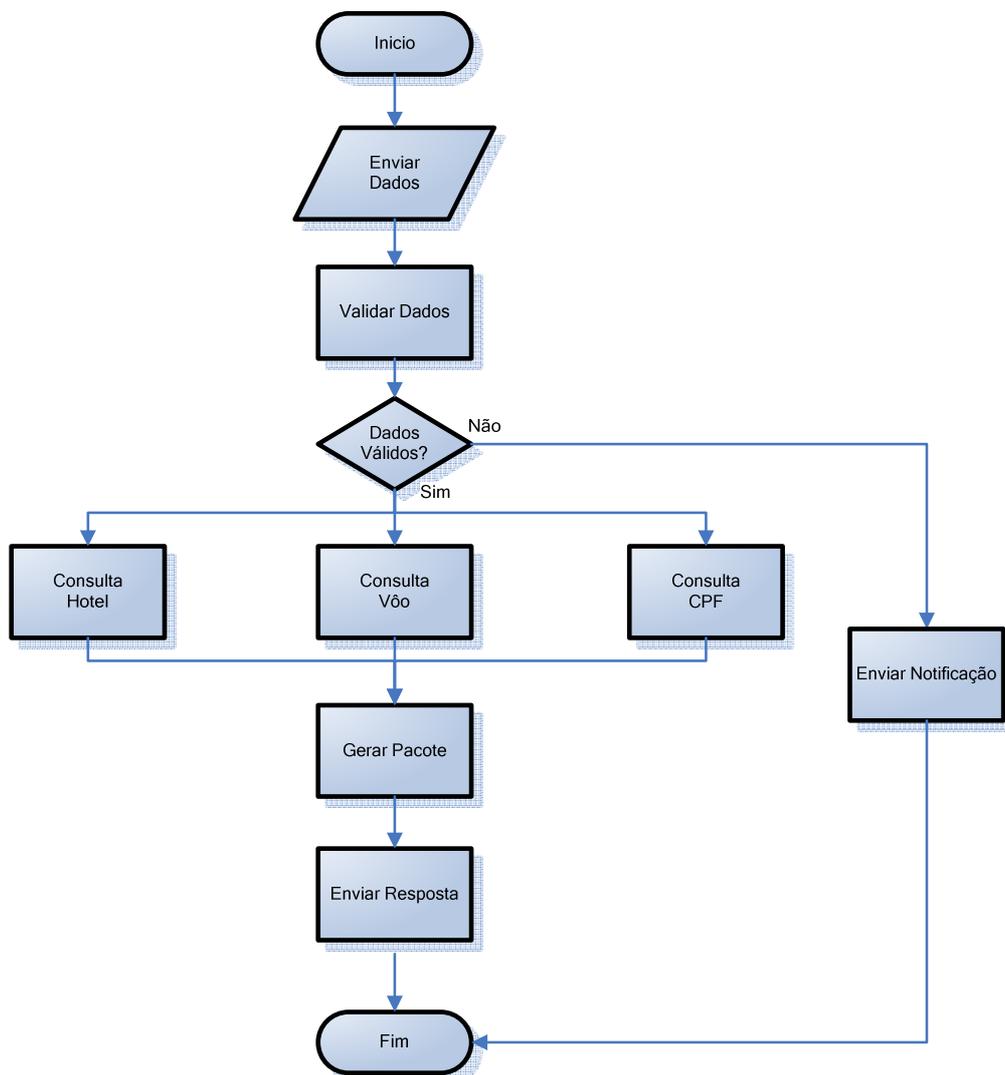
Inicialmente a *TravelX* disponibilizará a seus clientes apenas consultas a pacotes de viagem na forma de serviços, em um processo de negócio completamente novo e independente de outros sistemas já utilizados na empresa. Tal processo tem o intuito de validar uma nova arquitetura nos moldes de SOA, que poderá ser implantada posteriormente para outros processos da empresa. Este processo será implantado utilizando as idéias de adoção para pequenos negócios apresentadas no Capítulo 4.

## **5.2. Implantação da Consulta a Pacotes de Viagem**

A seguir serão descritas as execuções de cada uma das etapas da estratégia de adoção.

### **5.2.1. Definição do Modelo de Negócio**

Ao analisar o processo desejado pela *TravelX*, foi mapeado o modelo de negócio representado na Figura 5.1. É importante que este modelo seja validado com todos os *stakeholders* do processo, pois a partir de tal modelo é que serão criados todos os serviços a serem utilizados.



**Figura 5.1 - Processo de Negócio da Consulta de Pacotes**

As consultas ao Hotel e Vão tem o objetivo de encontrar hotéis e vôos que sirvam para a viagem desejada, enquanto que a consulta ao CPF será utilizada para indicar as formas de pagamento, que caso não conste com restrições na SERASA, poderá ser de forma parcelada.

## 5.2.2. Análise Orientada a Serviços

Nesta fase se dará início à modelagem dos serviços. A seguir será realizado um passo a passo baseado em [9].

### a) Decomposição do Processo de Negócio

Para melhor dar prosseguimento à análise orientada a serviços, inicialmente é interessante decompor o processo de negócio em passos bem definidos. A seguir seguem os passos que foram identificados:

- Enviar dados sobre o cliente e viagem desejada.
- Verificar validade dos dados fornecidos. Se forem inválidos, envia uma notificação.
- Consultar Hotel para os dados fornecidos.
- Consultar Vão para os dados fornecidos.
- Consultar se o CPF possui restrições na SERASA.
- Gerar Pacote de Viagem com os resultados recebidos das 3 consultas anteriores.
- Enviar pacote para o solicitante.

### b) Eliminar etapas que não são apropriadas para serem serviços

Inicialmente na fase de análise devem ser eliminadas as etapas do processo de negócio que não farão parte dos serviços a serem criados, como ações tomadas pelos usuários ou atividades que por algum outro motivo não se adéquem a serem integrados na forma de serviços. Especificamente no processo em questão, identificou-se apenas a seguinte etapa a ser retirada:

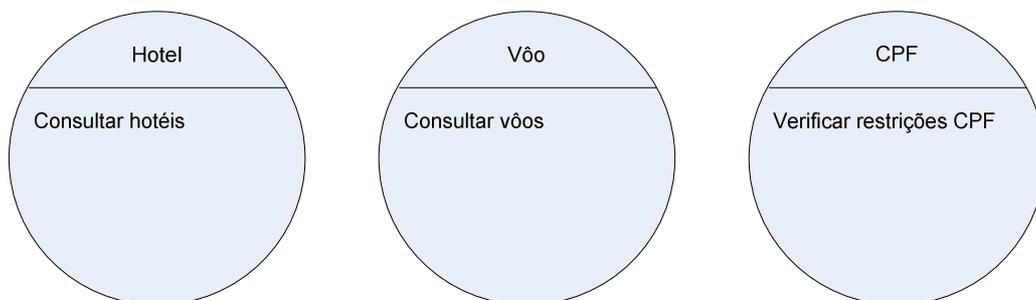
- Enviar dados sobre o cliente e viagem desejada. (Os dados fornecidos pelo usuário.)

### c) Identificar Serviços Agnósticos

Em seguida deve-se tentar identificar serviços agnósticos, ou seja, aqueles que são independentes do processo de negócio em questão. Facilmente são identificados inicialmente os seguintes serviços:

- Consultar disponibilidade de Hotel para os dados fornecidos.
- Consultar disponibilidade de Vôos para os dados fornecidos.
- Consultar se o CPF possui restrições na SERASA.

Por acaso, estes são justamente os serviços que encapsulam serviços externos ao ambiente da *TravelX* e poderão ser facilmente reusados em outros processos de negócio. Estes serviços podem ser modelados como serviços de entidade, de acordo com a Figura 5.2.



**Figura 5.2 – Serviços Agnósticos de Entidade**

Estes serviços atuam como mediadores para com os serviços externos à *TravelX*, atuando como *wrappers* da estratégia *bottom-up*. Este artifício foi utilizado devido à ausência de controle sobre os serviços externos ou sobre a arquitetura nos quais eles estão localizados, pois podem ocorrer modificações em suas interfaces, o

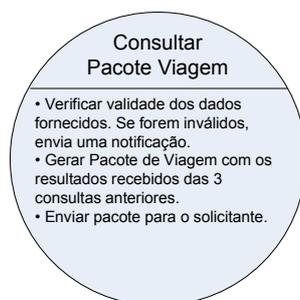
que acarretaria em modificações nos demais serviços da *TravelX*. Com este modelo, apenas estes serviços aqui descritos necessitariam ser modificados nesta situação.

#### d) Identificar Serviços Específicos do Negócio

O próximo passo consiste em tentar identificar os serviços específicos para o processo de negócio em questão, que no caso seriam relacionados às demais atividades, excluídas da etapa anterior:

- Verificar validade dos dados fornecidos. Se forem inválidos, envia uma notificação.
- Gerar Pacote de Viagem com os resultados recebidos das 3 consultas anteriores.
- Enviar pacote para o solicitante.

Desta forma, o serviço candidato à atividade de “Consultar Pacote Viagem” seria como ilustrado na Figura 5.3.



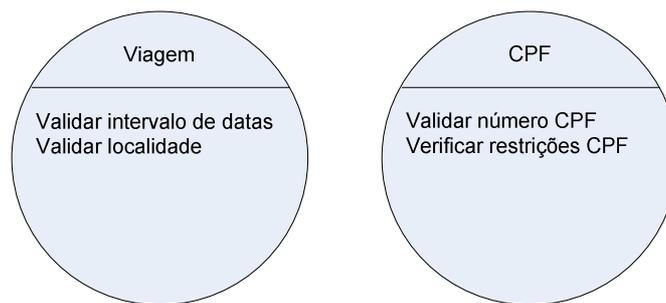
**Figura 5.3 – Serviço Candidato para Consultar Pacote Viagem**

#### e) Aplicar Princípios de Orientação a Serviços

Nesta etapa que se inicia o procedimento para tornar a solução realmente aderente à orientação a serviços, através da remodelagem dos serviços começando a aplicar os princípios de SOA.

A etapa de validação dos dados está bastante específica para este processo, por validar todos os dados de uma só vez. No entanto, se ela for separada de acordo com os dados fornecidos, poderá se tornar mais reutilizável e colocada em cada serviço específico.

Os dados fornecidos para a consulta são o CPF, data de ida, data de volta, cidade de origem e destino. O CPF poderá ser validado através de uma nova atividade no serviço CPF, chamada Validar número CPF, que verifica se a entrada é válida. Já as datas e localidades de ida e volta estão relacionadas dentro do contexto da viagem propriamente dita, desta forma estas validações poderão ser feitas em um novo serviço de entidade chamado Viagem. Desta forma ficamos com estes serviços conforme ilustrado na Figura 5.4.



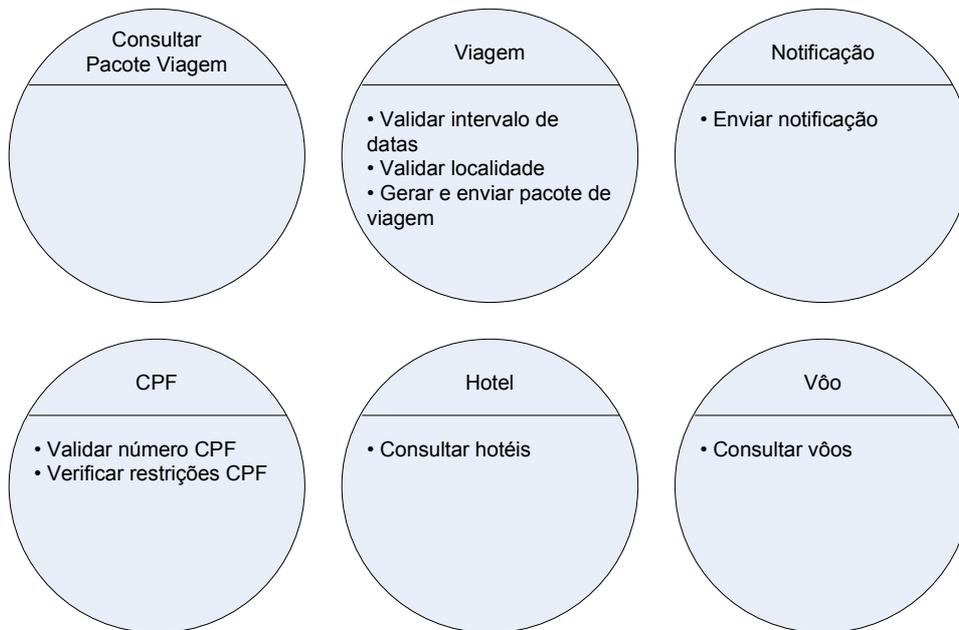
**Figura 5.4 - Serviços modificados para maior reuso**

Situações de erro e exceção que geram notificações são comuns dentro de qualquer processo de negócio em qualquer lugar, então um serviço de utilidade

para envio de notificações com alto grau de reuso pode ser criado para realizar essa atividade.

As atividades de geração do pacote de viagem e o envio do mesmo podem ser unidas, pois não fazem sentido independentemente, uma vez que sempre que o pacote for gerado, ele deverá ser enviado para algum lugar. Esta atividade também fica melhor localizada no serviço de Viagem.

De acordo com as modificações propostas, os serviços ficaram definidos como ilustrado na Figura 5.5.



**Figura 5.5 - Serviços Candidatos após aplicação dos Princípios de SOA**

#### f) Identificar composições dos serviços candidatos

Nesta etapa será elaborada uma composição dos serviços candidatos de acordo com as definições que foram realizadas até então. Desta forma, obtém-se a composição ilustrada na Figura 5.6.

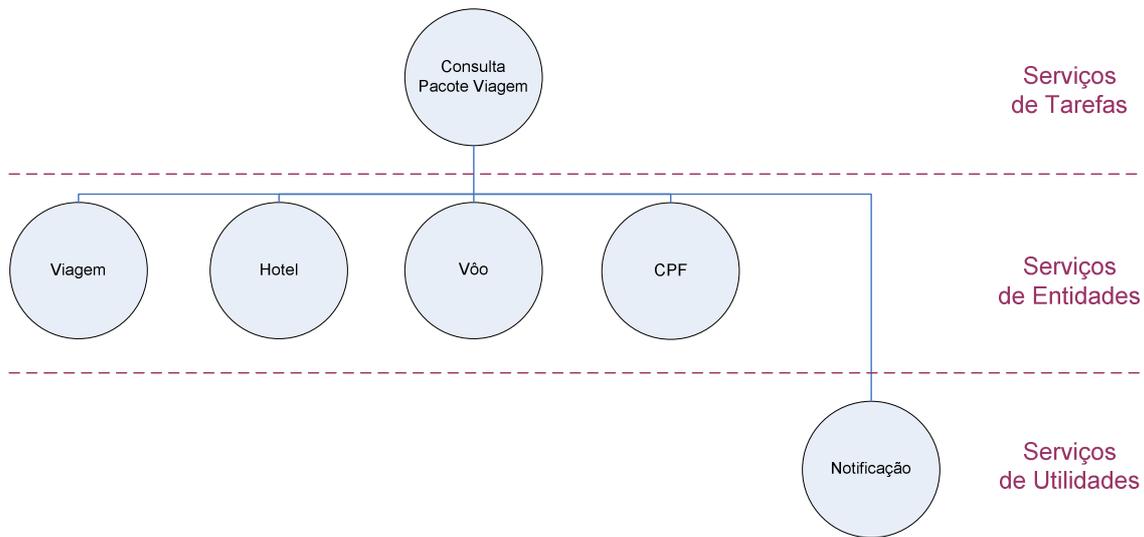


Figura 5.6 - Composição dos Serviços Candidatos

### 5.2.3. Design Orientado a Serviços

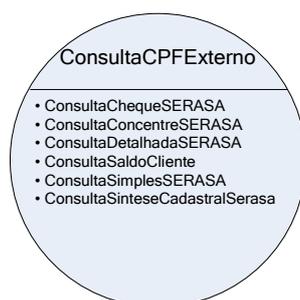
Nesta fase serão definidos os escopos dos serviços. Os serviços serão preparados na forma de *web services*, seguindo também um passo a passo baseado em [9].

#### a) Revisão dos Serviços Existentes

Antes de dar segmento ao design dos serviços, é importante analisar se algum dos serviços candidatos resultantes da etapa de análise não se sobrepõe a nenhum já existente. Como inicialmente não possuíamos nenhum serviço, este caso não se aplica.

Quanto aos serviços que estão relacionados com serviços externos, é importante analisar como são os serviços externos de forma a verificar a consistência entre os serviços e efetuar eventuais correções, caso necessário.

Como neste estudo de caso apenas o serviço de CPF está de fato acessando um sistema externo, este último precisa ser levado em consideração nesta etapa. Na Figura 5.7 temos representado este serviço, que já está pronto. Para utilização de nosso propósito, será utilizada a operação ConsultaSimplesSERASA. De forma que não é necessária nenhuma modificação para adequação.



**Figura 5.7 - Serviço de Consulta ao CPF da ConsultaCPF.com**

#### b) Definição dos Esquemas de Mensagens

Para definição dos formatos das mensagens recebidas e enviadas pelos serviços, serão definidos esquemas XSD. Nesta análise detalharemos o serviço de CPF, no entanto os demais XSDs gerados podem ser visualizados no Apêndice I. Na Figura 5.8 e na Figura 5.9 temos representados os XSDs das operações do serviço de CPF.

Existem ferramentas que auxiliam na elaboração destes esquemas, no entanto caso estas sejam utilizadas, é recomendável que se observe o código gerado para garantir que o modelo da mensagem está de fato definido como desejado. Estes formatos de mensagem não deverão ser modificados uma vez que sejam colocados em produção, por isso deve-se ter bastante cuidado nesta elaboração.

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.travelx.com/schemas/cpf/validacao/"
  targetNamespace="http://www.travelx.com/schemas/cpf/validacao/"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:element name="ValidarNumeroCPFRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="CPF" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ValidarNumeroCPFResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Valido" type="xs:boolean"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Figura 5.8 - XSD do Validação de CPF

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.travelx.com/schemas/cpf/restricao/"
  targetNamespace="http://www.travelx.com/schemas/cpf/restricao/"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:element name="VerificarRestricoesCPFRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="CPF" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="VerificarRestricoesCPFResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="TemRestricoes" type="xs:boolean"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Figura 5.9 - XSD da Verificação de Restrições no CPF

### c) Definição das interfaces abstratas

A partir dos esquemas XSD definidos, pode-se gerar as descrições em WSDL abstratas dos serviços a serem implementados. Estas descrições não levam em conta a localização efetiva dos serviços, sendo por isso chamadas de abstratas. Na Figura 5.10 temos o WSDL do Serviço de CPF.

```
<?xml version="1.0" encoding="UTF-8" ?>
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/"
             xmlns:tns="http://www.travelx.com/wsdl/cpf/"
             xmlns:xsd="http://www.w3.org/2001/XMLSchema"
             xmlns:val="http://www.travelx.com/schemas/cpf/validacao/"
             xmlns:rst="http://www.travelx.com/schemas/cpf/restricao/"
             targetNamespace="http://www.travelx.com/wsdl/cpf/">
  <import namespace="http://www.travelx.com/schemas/cpf/validacao/"
          location=" ../Esquemas/ValidarNumeroCPF.xsd" />
  <import namespace="http://www.travelx.com/schemas/cpf/restricao/"
          location=" ../Esquemas/VerificarRestricoesCPF.xsd" />
  <message name="ValidarNumeroCPFRequestMessage">
    <part name="RequestParameter" element="val:ValidarNumeroCPFRequest" />
  </message>
  <message name="ValidarNumeroCPFResponseMessage">
    <part name="ResponseParameter" element="val:ValidarNumeroCPFResponse" />
  </message>
  <message name="VerificarRestricoesCPFRequestMessage">
    <part name="RequestParameter" element="rst:VerificarRestricoesCPFRequest" />
  </message>
  <message name="VerificarRestricoesCPFResponseMessage">
    <part name="ResponseParameter" element="rst:VerificarRestricoesCPFResponse" />
  </message>
  <portType name="CPFInterface">
    <operation name="ValidarNumero">
      <input message="tns:ValidarNumeroCPFRequestMessage" />
      <output message="tns:ValidarNumeroCPFResponseMessage" />
    </operation>
    <operation name="VerificarRestricoes">
      <input message="tns:VerificarRestricoesCPFRequestMessage" />
      <output message="tns:VerificarRestricoesCPFResponseMessage" />
    </operation>
  </portType>
</definitions>
```

Figura 5.10 - WSDL do Serviço de CPF

Todos os serviços deverão ter seus WSDLs gerados desta forma, uma particularidade ocorre apenas no caso do serviço “Consulta Pacote Viagem”, que será responsável por orquestrar todos os demais serviços.

Assim como para a geração dos esquemas, existem ferramentas para geração dos WSDLs. Algumas destas ferramentas geram WSDLs a partir de códigos de desenvolvimento já prontos, o que não é recomendável, pois nesta etapa os serviços ainda estão sendo modelados para posteriormente serem desenvolvidos.

#### d) Aplicação dos Princípios de Orientação a Serviços

De modo a promover uma maior capacidade de descobrimento para quando os serviços forem instalados, é interessante documentá-los, o que pode ser feito com tags “<documentation>” no WSDL. Na Figura 5.11 está representada a documentação criada para o serviço de CPF.

```
<documentation>  
  ValidarNumeroCPF - Verifica se o CPF fornecido é um número válido.  
  VerificarRestricoes - Verifica se o CPF fornecido possui restricoes na SERASA.  
</documentation>
```

Figura 5.11 - Documentação do Serviço de CPF

### 5.2.4. Desenvolvimento

Uma vez definidos os serviços de acordo com SOA, é possível iniciar o processo de desenvolvimento dos mesmos. Por se tratarem de *web services*, várias linguagens de programação podem ser utilizadas para tal. Neste trabalho, optamos por desenvolver em Java, principalmente por ser uma linguagem de alta portabilidade e suportada por diversas ferramentas.

Já nesta etapa, devemos ter definido as soluções de software que serão utilizadas, inclusive o ESB, de modo a permitir a execução de pequenos testes ao longo do desenvolvimento. Após analisar algumas soluções *open source*, optou-se por utilizar a solução de ESB da WSO2 [36]. O WSO2 ESB possui uma interface bastante amigável de fácil utilização. Além disso, em determinadas operações possui desempenho comparado ao de soluções proprietárias, como em *proxy* de serviços, de acordo com Figura 5.12. No entanto ele não hospeda *web services*, tarefa desempenhada pelo WSO2 Application Server.

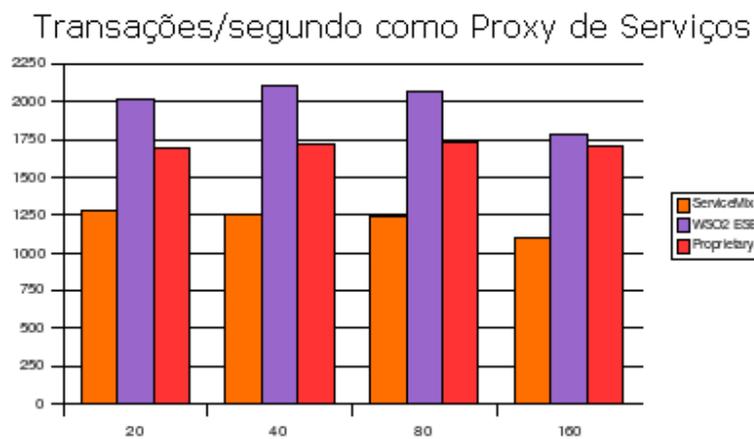


Figura 5.12 - Comparação de Desempenho do WSO2 ESB com outras soluções [36]

Para o desenvolvimento utilizou-se a IDE Eclipse [37] e alguns de seus plugins para desenvolvimento de *web services* como o WTP [38]. Também foi utilizado um plugin de integração fornecido pelo próprio WSO2 Application Server. A partir daí WSDLs concretos, contendo as informações de localização dos *web services*, foram gerados a partir dos abstratos já definidos anteriormente, o que possibilitou a geração de código esqueleto dos *web services* na forma *top-down*, como pode ser visto na Figura 5.13.

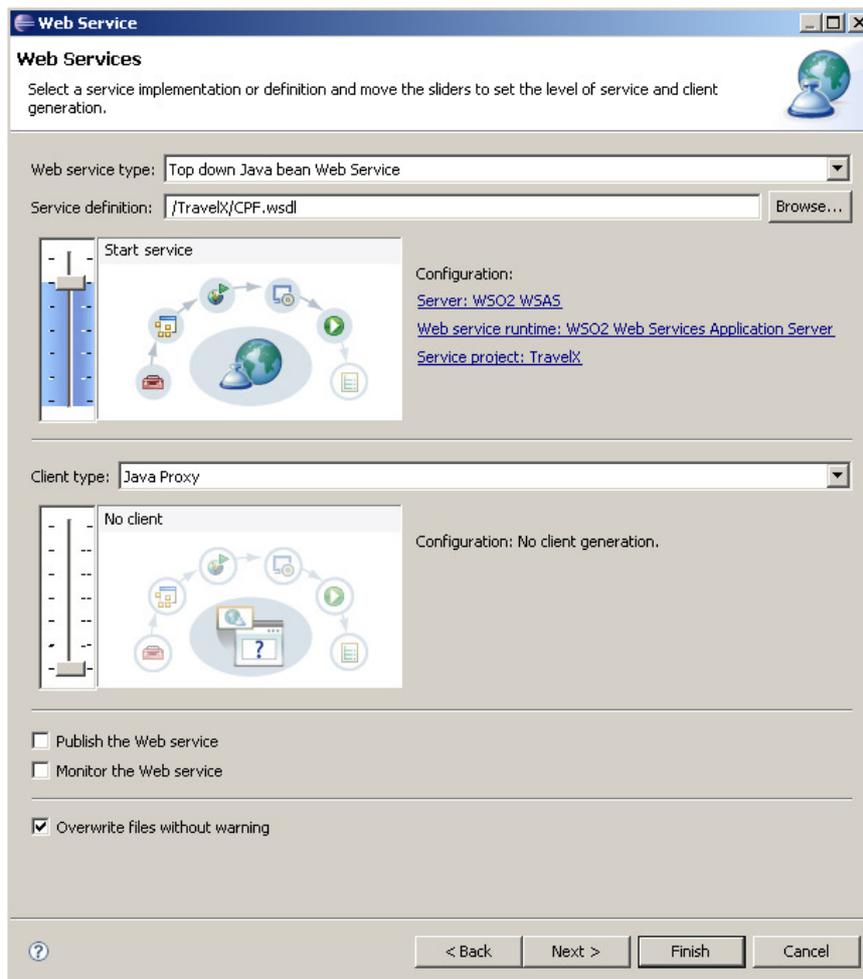


Figura 5.13 - Geração *Top-down* de código esqueleto Java para *Web Services*

A partir daí trata-se de um processo de desenvolvimento de software comum, no qual a lógica dos *web services* foi desenvolvida.

### 5.2.5. Testes

Uma vez desenvolvidos os *web services*, os mesmos deverão ser testados. Existem várias ferramentas *open source* para execução dos mais variados testes, destacando-se o SOAPUI [39]. Realizaram-se aqui apenas testes funcionais, como ilustrado na Figura 5.14.



Figura 5.14 - Teste Funcional da Operação ValidarNumero de CPF

## 5.2.6. Instalação

Uma vez validados, os *web services* poderão ser instalados em definitivo. De acordo com a solução escolhida, inicialmente eles deverão ser colocados no WSO2 Application Server. Como pode ser visto na Figura 5.15. A partir daí ele pode ser acionado a partir do WSO2 ESB.

**Services > CPF**

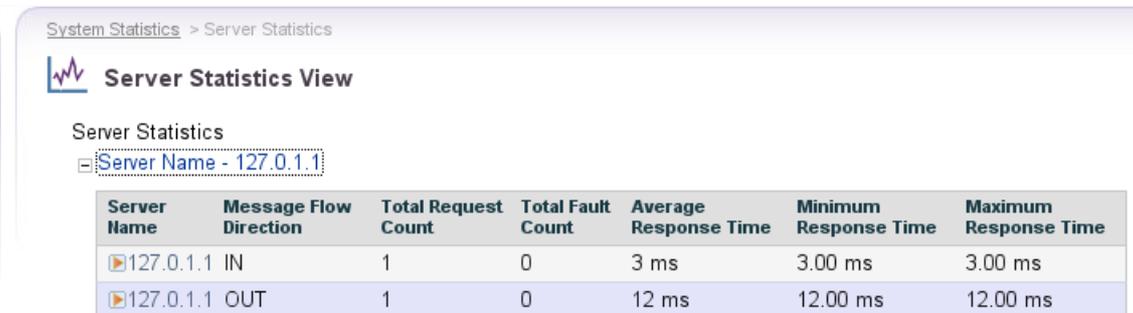
Service Information

| General             |   |
|---------------------|---|
| Service Name        | CPF   |
| Service Description | CPF   |
| Service Group Name  | CPF   |
| Deployment Scope    | request   |
| Service Type        | Normal  |
| Service Status      | Active  |
| MTOM                | Optional  |
| End Points          |   |
| Endpoint References | https://192.168.254.5:9443/services/CPF<br>http://192.168.254.5:9762/services/CPF |

Figura 5.15 - Serviço de CPF Instalado no WSO2 Application Server

## 5.2.7. Administração

Uma vez que os serviços foram instalados corretamente, eles devem ser mantidos em funcionamento. Para isso auxiliar isto, as soluções de ESB, inclusive o WSO2 ESB, dispõem de diversas estatísticas de tudo o que passa com ele e registro de logs, que podem ser utilizados para descobrir problemas, como pode ser visto na Figura 5.16 e na Figura 5.17, respectivamente.



System Statistics > Server Statistics

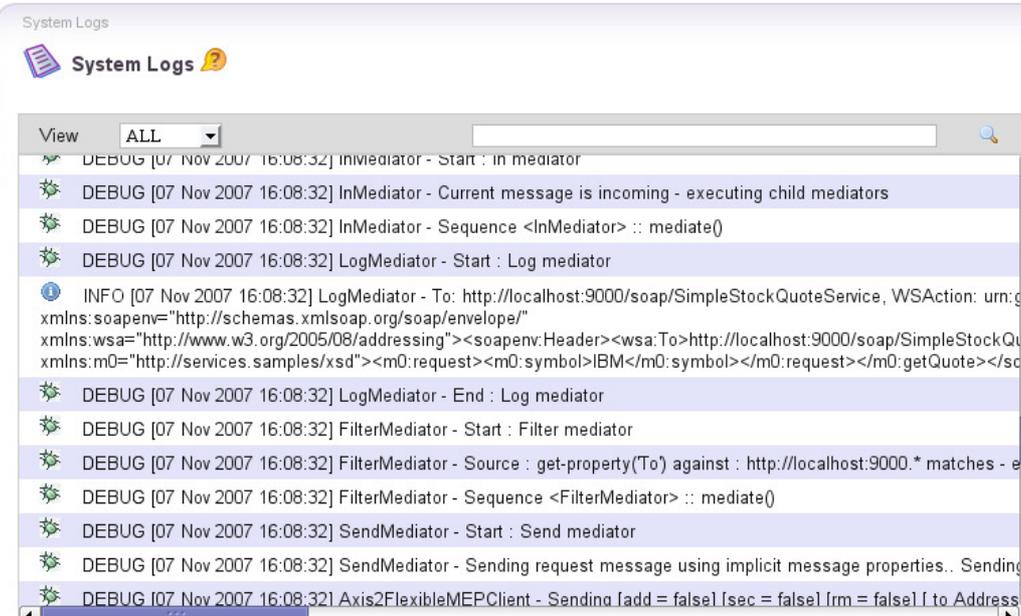
**Server Statistics View**

Server Statistics

Server Name - 127.0.1.1

| Server Name | Message Flow Direction | Total Request Count | Total Fault Count | Average Response Time | Minimum Response Time | Maximum Response Time |
|-------------|------------------------|---------------------|-------------------|-----------------------|-----------------------|-----------------------|
| 127.0.1.1   | IN                     | 1                   | 0                 | 3 ms                  | 3.00 ms               | 3.00 ms               |
| 127.0.1.1   | OUT                    | 1                   | 0                 | 12 ms                 | 12.00 ms              | 12.00 ms              |

Figura 5.16 - Estatísticas do WSO2 ESB [36]



System Logs

**System Logs**

View: ALL

- DEBUG [07 Nov 2007 16:08:32] InMediator - Start : In mediator
- DEBUG [07 Nov 2007 16:08:32] InMediator - Current message is incoming - executing child mediators
- DEBUG [07 Nov 2007 16:08:32] InMediator - Sequence <InMediator> :: mediate()
- DEBUG [07 Nov 2007 16:08:32] LogMediator - Start : Log mediator
- INFO [07 Nov 2007 16:08:32] LogMediator - To: http://localhost:9000/soap/SimpleStockQuoteService, WSAction: urn:sc...  
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"  
xmlns:wsa="http://www.w3.org/2005/08/addressing"><soapenv:Header><wsa:To>http://localhost:9000/soap/SimpleStockQu...  
xmlns:m0="http://services.samples/xsd"><m0:request><m0:symbol>IBM</m0:symbol></m0:request></sc...
- DEBUG [07 Nov 2007 16:08:32] LogMediator - End : Log mediator
- DEBUG [07 Nov 2007 16:08:32] FilterMediator - Start : Filter mediator
- DEBUG [07 Nov 2007 16:08:32] FilterMediator - Source : get-property(To) against : http://localhost:9000.\* matches - e...
- DEBUG [07 Nov 2007 16:08:32] FilterMediator - Sequence <FilterMediator> :: mediate()
- DEBUG [07 Nov 2007 16:08:32] SendMediator - Start : Send mediator
- DEBUG [07 Nov 2007 16:08:32] SendMediator - Sending request message using implicit message properties.. Sending
- DEBUG [07 Nov 2007 16:08:32] Axis2FlexibleMEPClient - Sending [add = false] [sec = false] [rm = false] to Address...

Figura 5.17 - Logs do WSO2 ESB [36]

## 6. CONCLUSÃO

---

Ao longo do presente trabalho foram apresentados conceitos sobre SOA e pequenos negócios. Abordou-se a questão da utilização de SOA por parte destes pequenos negócios, levantando algumas questões particulares que deveriam ser levadas em consideração para que esta adoção pudesse ocorrer da melhor forma. Foram propostas algumas idéias e uma modificação na estratégia de adoção de SOA, que foram colocadas em prática através de um estudo de caso.

### 6.1. Contribuições

A utilização de SOA está revolucionando a Tecnologia da Informação, pois em vez de se preocupar com orientação a objetos, herança, polimorfismo, a TI está passando a trabalhar mais em cima da execução de processos de negócio, objetivando lucratividade e alta produção, tornando a computação mais ágil, fácil e objetiva. Este trabalho contribuiu possibilitando que pequenos negócios possam obter crescimento e evolução através do uso desta mentalidade.

### 6.2. Trabalhos Futuros

Ferramentas *open source* específicas para SOA ainda não o mesmo grau de maturidade de outras soluções proprietárias. Um futuro esforço pode ser direcionado neste sentido, de como elas podem ser melhoradas através de uma melhor combinação e adição de novos recursos.

## REFERÊNCIAS

---

- [1] CASTRO-LEON, E.; HE, J.; CHANG, M., "Scaling Down SOA to Small Businesses," Service-Oriented Computing and Applications, 2007. SOCA '07. IEEE International Conference on , vol., no., pp.99-106, 19-20 June 2007.
- [2] INFO Online. Apresenta informações sobre estudo do Gartner na adoção de SOA. Disponível em: <<http://info.abril.com.br/aberto/infonews/042007/27042007-8.shl>>. Acesso em 18 out. 2008.
- [3] ROBOOSTOFF, A., "Leveraging SOA for your SMB", Disponível em: <<http://www.ebizq.net/topics/soa/features/7152.html>>. Acesso em 18 out. 2008.
- [4] MCKENDRICK, J., "Small companies don't have time for SOA". Disponível em: <<http://blogs.zdnet.com/service-oriented/?p=941>>. Acesso em 18 out. 2008.
- [5] Michaelis - Moderno Dicionário da Língua Portuguesa. Disponível em <<http://michaelis.uol.com.br>>. Acesso em 19 out. 2008.
- [6] ERL, T., SOA: Principles of Service Design. Upper Saddle River: Prentice Hall PTR. ISBN 0-13-234482-3. 2007.
- [7] DAN, X.; SHI, Y.; TAO, Z., XIANG-YANG, J.; ZAO-QING, L.; JUN-FENG, Y., "An Approach for Describing SOA," Wireless Communications, Networking and Mobile Computing, 2006. WiCOM 2006. International Conference on , vol., no., pp.1-4, 22-24 Sept. 2006.
- [8] PAPAZOGLU, M.; TRAVERSO, P.; DUSTDAR, S.; LEYMANN, F.; KRAMER, B., "Service-oriented computing: A research roadmap". In Francisco Cubera, Service Oriented Computing (SOC), No. 05462 in Dagstuhl Seminar Proceedings, 2006.
- [9] ERL, T., Service-Oriented Architecture: Concepts, Technology, and Design. Upper Saddle River: Prentice Hall PTR. ISBN 0-13-185858-0. 2005.

- [10] AL BELUSHI, W.; BAGHDADI, Y., "An Approach to Wrap Legacy Applications into Web Services," *Service Systems and Service Management*, 2007 International Conference on , vol., no., pp.1-6, 9-11 June 2007.
- [11] ORTIZ JR., Sixto, "Getting on Board the Enterprise Service Bus," *Computer* , vol.40, no.4, pp.15-17, April 2007.
- [12] LAPLANTE, P. A.; ZHANG, J.; VOAS, J., "What's in a Name? Distinguishing between SaaS and SOA," *IT Professional* , vol.10, no.3, pp.46-50, May-June 2008
- [13] W3C. Apresenta informações sobre XML. Disponível em <<http://www.w3.org/XML/>>. Acesso em 1 nov. 2008.
- [14] DAILY, P. G., "XML Basics and Benefits". Disponível em <[http://www.intranetjournal.com/articles/200312/ij\\_12\\_08\\_03a.html](http://www.intranetjournal.com/articles/200312/ij_12_08_03a.html)>. Acesso em 1 nov. 2008.
- [15] UDDI Technical White Paper. Disponível em <[http://www.uddi.org/pubs/Iru\\_UDDI\\_Technical\\_White\\_Paper.pdf](http://www.uddi.org/pubs/Iru_UDDI_Technical_White_Paper.pdf)>. Acesso em 1 nov. 2008.
- [16] JEPSEN, T., "SOAP cleans up interoperability problems on the Web," *IT Professional* , vol.3, no.1, pp.52-55, Jan/Feb 2001.
- [17] W3C Schools. Possui informações sobre tecnologias XML. Disponível em <<http://www.w3schools.com>>. Acesso em 1 nov. 2008.
- [18] ERL, T., "Introdução às Tecnologias Web Services". Disponível em <<http://www.devmedia.com.br/articles/viewcomp.asp?comp=2873>>. Acesso em 1 nov. 2008.
- [19] WILKES, L., "Applying ESB", *CBDi Journal* 2006 / April.
- [20] ROTH, E., "SOA Benefits, Challenges and Risk Mitigation". Disponível em <<http://it.toolbox.com/blogs/the-soa-blog/soa-benefits-challenges-and-risk-mitigation-8075>>. Acesso em 8 nov. 2008.

- [21] "Data Integration in a Service-Oriented Architecture". Disponível em <[http://www.informatica.com/NL/products/powercenter/arch/data\\_services/infa\\_soa\\_wp\\_j506956646101405.pdf](http://www.informatica.com/NL/products/powercenter/arch/data_services/infa_soa_wp_j506956646101405.pdf)>. Acesso em 8 nov. 2008.
- [22] MACGREGOR, R. C.; WANGH, P.; BUNKER, D. J.; COURTNEY, J. F., "Adoption of EDI by small business: are the advocates in tune with the views of small business-a pilot study," *System Sciences*, 1997, Proceedings of the Thirtieth Hawaii International Conference on , vol.4, no., pp.42-47 vol.4, 7-10 Jan 1997
- [23] Instituto Brasileiro de Geografia e Estatística - IBGE. "As Micro e Pequenas Empresas Comerciais e de Serviços no Brasil". Disponível em <<http://www.ibge.gov.br/home/estatistica/economia/microempresa/default.shtm>>. Acesso em 15 nov. 2008.
- [24] CHURCHILL, N.; LEWIS, V., *The Five Stages of Small Business Growth*. Harward Business Review. 61 (3), 30-50, 1983.
- [25] MACGREGOR, R.; VRAZALIC, L., "Sector Driven Variations on E-commerce Adoption Barriers in Regional Small Businesses: An Australian Study," *Innovations in Information Technology*, 2006 , vol., no., pp.1-5, Nov. 2006.
- [26] LODHI, F.; SALEEM, M. M., "A domain-specific component-based methodology for the rapid development of small business applications," *Multitopic Conference, 2004. Proceedings of INMIC 2004. 8th International* , vol., no., pp. 423-427, 24-26 Dec. 2004.
- [27] CHESTER, M., SKOK, W., Roadmap for successful information technology transfer for small businesses, Proceedings of the 2000 ACM SIGCPR conference on Computer personnel research, p.16-22, April 2000.
- [28] "The 10 Natural Advantages of Small Business". Disponível em <<http://www.ducttapemarketing.com/blog/2005/11/04/the-10-natural-marketing-advantages-of-small-business-small-is-the-new-big/>>. Acesso em 15 nov. 2008.

- [29] "Small Business Advantages". Disponível em <<http://www.startupinternetmarketing.com/ezines/smallbiz.html>>. Acesso em 15 nov. 2008
- [30] Instituto Brasileiro de Geografia e Estatística - IBGE. "Demografia das Empresas 2005". Disponível em <<http://www.ibge.gov.br/home/estatistica/economia/demografiaempresa/2005/default.shtm>>. Acesso em 15 nov. 2008.
- [31] KING, J., "Overcoming Challenges that Prevent Small Business Growth". Disponível em <[http://www.canadaone.com/ezine/april07/small\\_business\\_growth\\_challenges.html](http://www.canadaone.com/ezine/april07/small_business_growth_challenges.html)>. Acesso em 15 nov 2008.
- [32] KAHN IV, A., "Simplifying E-Commerce with SOA for Payments". Disponível em <<http://www.ecommercetimes.com/story/60295.html>>. Acesso em 24 nov 2008.
- [33] FERNANDES, D., "A onda SOA". Disponível em: <[http://imasters.uol.com.br/artigo/8140/gerencia/a\\_onda\\_soa/](http://imasters.uol.com.br/artigo/8140/gerencia/a_onda_soa/)>. Acesso em 24 nov 2008.
- [34] SERASA. Disponível em <<http://www.serasa.com.br>>. Acesso em 25 nov. 2008.
- [35] ConsultaCPF.com. Disponível em <<http://www.consultacpf.com>>. Acesso em 25 nov 2008.
- [36] WSO2. Disponível em <<http://www.wso2.com>>. Acesso em 25 nov. 2008.
- [37] Eclipse. Disponível em <<http://www.eclipse.org>>. Acesso em 25 nov. 2008.
- [38] WTP - Web Tools Platform. Disponível em <<http://www.eclipse.org/webtools/>>. Acesso em 25 nov. 2008.
- [39] SOAPUI. Disponível em <<http://www.soapui.org>>. Acesso em 25 nov. 2008.

# APÊNDICE I - ESQUEMAS XSD DOS SERVIÇOS DA TRAVELX

---

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.travelx.com/schemas/notificacao/"
  targetNamespace="http://www.travelx.com/schemas/notificacao/"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:element name="EnviarNotificacaoRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Mensagem" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="EnviarNotificacaoResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Valido" type="xs:boolean"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.travelx.com/schemas/viagem/pacote/"
  targetNamespace="http://www.travelx.com/schemas/viagem/pacote/"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:element name="GerarEnviarPacoteViagemRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="DataInicio" type="xs:date"/>
        <xs:element name="DataFim" type="xs:date"/>
        <xs:element name="LocalidadeOrigem" type="xs:string"/>
        <xs:element name="LocalidadeDestino" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GerarEnviarPacoteViagemResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Hotel" minOccurs="0" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Nome" type="xs:string"/>
              <xs:element name="Preco" type="xs:int"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="Voo" minOccurs="0" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Numero" type="xs:short"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```

        <xs:element name="Companhia" type="xs:string"/>
        <xs:element name="Preco" type="xs:int"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns="http://www.travelx.com/schemas/viagem/datas/"
    targetNamespace="http://www.travelx.com/schemas/viagem/data/"
    elementFormDefault="qualified"
    attributeFormDefault="unqualified">
    <xs:element name="ValidarIntervaloDatasRequest">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="DataInicio" type="xs:date"/>
                <xs:element name="DataFim" type="xs:date"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="ValidarIntervaloDatasResponse">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="Valido" type="xs:string"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns="http://www.travelx.com/schemas/viagem/localidade/"
    targetNamespace="http://www.travelx.com/schemas/viagem/localidade/"
    elementFormDefault="qualified"
    attributeFormDefault="unqualified">
    <xs:element name="ValidarLocalidadeRequest">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="Localidade" type="xs:string"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="ValidarLocalidadeResponse">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="Valido" type="xs:boolean"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns="http://www.travelx.com/schemas/cpf/validacao/"
    targetNamespace="http://www.travelx.com/schemas/cpf/validacao/"
    elementFormDefault="qualified"

```

```

        attributeFormDefault="unqualified">
        <xs:element name="ValidarNumeroCPFRequest">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="CPF" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="ValidarNumeroCPFResponse">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="Valido" type="xs:boolean"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:schema>

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns="http://www.travelx.com/schemas/cpf/restricao/"
    targetNamespace="http://www.travelx.com/schemas/cpf/restricao/"
    elementFormDefault="qualified"
    attributeFormDefault="unqualified">
    <xs:element name="VerificarRestricoesCPFRequest">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="CPF" type="xs:string"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="VerificarRestricoesCPFResponse">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="TemRestricoes" type="xs:boolean"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns="http://www.travelx.com/schemas/consultapacote/"
    targetNamespace="http://www.travelx.com/schemas/consultapacote/"
    elementFormDefault="qualified"
    attributeFormDefault="unqualified">
    <xs:element name="ConsultaPacoteRequest">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="CPF" type="xs:string"/>
                <xs:element name="LugarOrigem" type="xs:string"/>
                <xs:element name="LugarDestino" type="xs:string"/>
                <xs:element name="DataInicio" type="xs:date"/>
                <xs:element name="DataFim" type="xs:date"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="ConsultaPacoteResponse">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="Parcelamento" type="xs:boolean"/>
                <xs:element name="Voo" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>

```

```

                <xs:sequence>
                    <xs:element name="Numero" type="xs:short"/>
                    <xs:element name="Companhia" type="xs:string"/>
                    <xs:element name="Preco" type="xs:int"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="Hotel" minOccurs="0" maxOccurs="unbounded">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="Nome" type="xs:string"/>
                    <xs:element name="Preco" type="xs:int"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.travelx.com/schemas/hotel/"
  targetNamespace="http://www.travelx.com/schemas/hotel/"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:element name="ConsultarHoteisRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="DataInicio" type="xs:date"/>
        <xs:element name="DataFim" type="xs:date"/>
        <xs:element name="Localidade" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ConsultarHoteisResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Hotel" minOccurs="0" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Nome" type="xs:string"/>
              <xs:element name="Preco" type="xs:int"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.travelx.com/schemas/voo/"
  targetNamespace="http://www.travelx.com/schemas/voo/"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:element name="ConsultarVoosRequest">
    <xs:complexType>
      <xs:sequence>

```

```

        <xs:element name="DataInicio" type="xs:date"/>
        <xs:element name="DataFim" type="xs:date"/>
        <xs:element name="LocalidadeOrigem" type="xs:string"/>
        <xs:element name="LocalidadeDestino" type="xs:string"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ConsultarVoosResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="Voo" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="Numero" type="xs:short"/>
                        <xs:element name="Companhia" type="xs:string"/>
                        <xs:element name="Preco" type="xs:int"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:schema>

```

## APÊNDICE II – DESCRIÇÕES WSDL ABSTRATAS

---

```
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:plnk="http://schemas.xmlsoap.org/ws/2003/05/partner-link/" xmlns:tns="http://www.travelx.com/wsdl/hotel/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:hot="http://www.travelx.com/schemas/hotel/" targetNamespace="http://www.travelx.com/wsdl/hotel/">
```

```
  <import namespace="http://www.travelx.com/schemas/hotel/" location="../../Esquemas/ConsultarHoteis.xsd"/>
```

```
  <message name="ConsultarHoteisRequestMessage">
    <part name="RequestParameter" element="hot:ConsultarHoteisRequest"/>
  </message>
  <message name="ConsultarHoteisResponseMessage">
    <part name="ResponseParameter" element="hot:ConsultarHoteisResponse"/>
  </message>

  <portType name="HotelInterface">
    <operation name="ConsultarHoteis">
      <input message="tns:ConsultarHoteisRequestMessage"/>
      <output message="tns:ConsultarHoteisResponseMessage"/>
    </operation>
  </portType>
</definitions>
```

```
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:plnk="http://schemas.xmlsoap.org/ws/2003/05/partner-link/" xmlns:tns="http://www.travelx.com/wsdl/notificacao/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:not="http://www.travelx.com/schemas/notificacao/" targetNamespace="http://www.travelx.com/wsdl/notificacao/">
```

```
  <import namespace="http://www.travelx.com/schemas/notificacao/"
location="../../Esquemas/EnviarNotificacao.xsd"/>

  <message name="EnviarNotificacaoRequestMessage">
    <part name="RequestParameter" element="not:EnviarNotificacaoRequest"/>
  </message>
  <message name="EnviarNotificacaoResponseMessage">
    <part name="ResponseParameter" element="not:EnviarNotificacaoResponse"/>
  </message>

  <portType name="NotificacaoInterface">
    <operation name="EnviarNotificacao">
      <input message="tns:EnviarNotificacaoRequestMessage"/>
      <output message="tns:EnviarNotificacaoResponseMessage"/>
    </operation>
  </portType>
</definitions>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:plnk="http://schemas.xmlsoap.org/ws/2003/05/partner-link/" xmlns:tns="http://www.travelx.com/wsdl/viagem/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:loc="http://www.travelx.com/schemas/viagem/localidade/"
xmlns:dat="http://www.travelx.com/schemas/viagem/data/"
xmlns:pac="http://www.travelx.com/schemas/viagem/pacote/"
targetNamespace="http://www.travelx.com/wsdl/viagem/">
  <import namespace="http://www.travelx.com/schemas/viagem/localidade/"
location="../../Esquemas/ValidarLocalidade.xsd"/>
  <import namespace="http://www.travelx.com/schemas/viagem/pacote/"
location="../../Esquemas/GerarEnviarPacoteViagem.xsd"/>
```

```

        <import namespace="http://www.travelx.com/schemas/viagem/data/"
location="../Esquemas/ValidarIntervaloDatas.xsd"/>
        <message name="ValidarLocalidadeRequestMessage">
            <part name="RequestParameter" element="loc:ValidarLocalidadeRequest"/>
        </message>
        <message name="ValidarLocalidadeResponseMessage">
            <part name="ResponseParameter" element="loc:ValidarLocalidadeResponse"/>
        </message>
        <message name="ValidarIntervaloDatasRequestMessage">
            <part name="RequestParameter" element="dat:ValidarIntervaloDatasRequest"/>
        </message>
        <message name="ValidarIntervaloDatasResponseMessage">
            <part name="ResponseParameter" element="dat:ValidarIntervaloDatasResponse"/>
        </message>
        <message name="GerarEnviarPacoteViagemRequestMessage">
            <part name="RequestParameter" element="pac:GerarEnviarPacoteViagemRequest"/>
        </message>
        <message name="GerarEnviarPacoteViagemResponseMessage">
            <part name="ResponseParameter" element="pac:GerarEnviarPacoteViagemResponse"/>
        </message>
        <portType name="ViagemInterface">
            <operation name="ValidarLocalidade">
                <input message="tns:ValidarLocalidadeRequestMessage"/>
                <output message="tns:ValidarLocalidadeResponseMessage"/>
            </operation>
            <operation name="ValidarIntervaloDatas">
                <input message="tns:ValidarIntervaloDatasRequestMessage"/>
                <output message="tns:ValidarIntervaloDatasResponseMessage"/>
            </operation>
            <operation name="GerarEnviarPacoteViagem">
                <input message="tns:GerarEnviarPacoteViagemRequestMessage"/>
                <output message="tns:GerarEnviarPacoteViagemResponseMessage"/>
            </operation>
        </portType>
    </definitions>

<definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:plnk="http://schemas.xmlsoap.org/ws/2003/05/partner-link/" xmlns:tns="http://www.travelx.com/wsdl/voo/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:voo="http://www.travelx.com/schemas/voo/" targetNamespace="http://www.travelx.com/wsdl/voo/">

        <import namespace="http://www.travelx.com/schemas/voo/" location="../Esquemas/ConsultarVoos.xsd"/>

        <message name="ConsultarVoosRequestMessage">
            <part name="RequestParameter" element="voo:ConsultarVoosRequest"/>
        </message>
        <message name="ConsultarVoosResponseMessage">
            <part name="ResponseParameter" element="voo:ConsultarVoosResponse"/>
        </message>

        <portType name="VooInterface">
            <operation name="ConsultarVoos">
                <input message="tns:ConsultarVoosRequestMessage"/>
                <output message="tns:ConsultarVoosResponseMessage"/>
            </operation>
        </portType>
    </definitions>

<?xml version="1.0" encoding="UTF-8"?>
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/"
xmlns:tns="http://www.travelx.com/wsdl/cpf/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:val="http://www.travelx.com/schemas/cpf/validacao/"

```

```

        xmlns:rst="http://www.travelx.com/schemas/cpf/restricao/"
        xmlns:plnk="http://schemas.xmlsoap.org/ws/2003/05/partner-link/"
        targetNamespace="http://www.travelx.com/wsd/cpf/">
    <import namespace="http://www.travelx.com/schemas/cpf/validacao/"
        location="../Esquemas/ValidarNumeroCPF.xsd"/>
    <import namespace="http://www.travelx.com/schemas/cpf/restricao/"
        location="../Esquemas/VerificarRestricoesCPF.xsd"/>
    <message name="ValidarNumeroCPFRequestMessage">
        <part name="RequestParameter" element="val:ValidarNumeroCPFRequest"/>
    </message>
    <message name="ValidarNumeroCPFResponseMessage">
        <part name="ResponseParameter" element="val:ValidarNumeroCPFResponse"/>
    </message>
    <message name="VerificarRestricoesCPFRequestMessage">
        <part name="RequestParameter" element="rst:VerificarRestricoesCPFRequest"/>
    </message>
    <message name="VerificarRestricoesCPFResponseMessage">
        <part name="ResponseParameter" element="rst:VerificarRestricoesCPFResponse"/>
    </message>
    <portType name="CPFInterface">
        <operation name="ValidarNumero">
            <input message="tns:ValidarNumeroCPFRequestMessage"/>
            <output message="tns:ValidarNumeroCPFResponseMessage"/>
        </operation>
        <operation name="VerificarRestricoes">
            <input message="tns:VerificarRestricoesCPFRequestMessage"/>
            <output message="tns:VerificarRestricoesCPFResponseMessage"/>
        </operation>
    </portType>
</definitions>

```