

UNIDADE
Nº CÍVIL
TÍTULO
V.
TOMBO
PROC.
C
PREÇO
DATA
1971

Uma Abordagem Baseada em Características para o Estabelecimento de Contratos Eletrônicos para Serviços Web

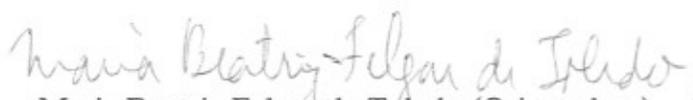
Este exemplar corresponde à redação final da Tese devidamente corrigida e defendida por Marcelo Fantinato e aprovada pela Banca Examinadora.

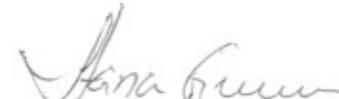
Este exemplar corresponde à redação final da Tese/Dissertação devidamente corrigida e defendida por: MARCELO FANTINATO

e aprovada pela Banca Examinadora.
Campinas, 10 de FEVEREIRO de 08


COORDENADOR DE POS-GRADUAÇÃO
CPG-IC

Campinas, 10 de dezembro de 2007.


Maria Beatriz Felgar de Toledo (Orientadora)


Itana Maria de Souza Gimenes (Co-orientadora)

Tese apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de doutor em Ciência da Computação.

**FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DO IMECC DA UNICAMP**

Bibliotecária: Maria Júlia Milani Rodrigues – CRB8a / 2116

F18a	Fantinato, Marcelo Uma abordagem baseada em características para o estabelecimento de contratos eletrônicos para serviços web / Marcelo Fantinato -- Campinas, [S.P.:s.n.], 2007. Orientador: Maria Beatriz Felgar de Toledo; Itana Maria de Souza Gimenes. Tese (doutorado) - Universidade Estadual de Campinas, Instituto de Computação. 1. Serviços na web. 2. Comercio eletrônico. 3. Engenharia de software. I. Toledo, Maria Beatriz Felgar de. II. Gimenes, Itana Maria de Souza. III. Universidade Estadual de Campinas. Instituto de Computação. IV. Título.
------	---

Título em inglês: A feature-based approach to web services e-contract establishment.

Palavras-chave em inglês (Keywords): 1. Web services. 2. Electronic commerce. 3. Software engineering.

Área de concentração: Sistemas de informação.

Titulação: Doutor em Ciência da Informação.

Banca examinadora: Profª. Dra. Itana Maria de Souza Gimenes (DIN-UEM)
Prof. Dr. Marcos Roberto da Silva Borges (NCE-UFRJ)
Profª. Dra. Regina Maria Maciel Braga (DCC-UFJF)
Prof. Dr. Edmundo Roberto Mauro Madeira (IC-UNICAMP)
Profª. Dra. Islene Calciolari Garcia (IC-UNICAMP)
Prof. Dr. Marcelo Morandini (EACH-USP)
Prof. Dr. Neucimar Jerônimo Leite (IC-UNICAMP)

Data da defesa: 10/12/2007

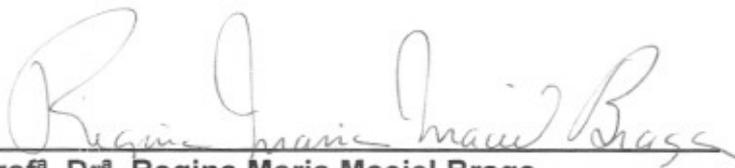
Programa de Pós-Graduação: Doutorado em Ciência da Informação

TERMO DE APROVAÇÃO

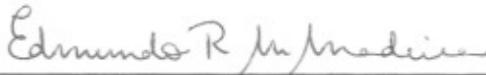
Tese Defendida e Aprovada em 10 de dezembro de 2007, pela Banca examinadora composta pelos Professores Doutores:



Prof. Dr. Marcos Roberto da Silva Borges
NCE - UFRJ.



Prof.ª Dr.ª Regina Maria Maciel Braga
DCC - UFJF.



Prof. Dr. Edmundo Roberto Mauro Madeira
IC - UNICAMP.



Prof.ª Dr.ª Islene Calciolari Garcia
IC - UNICAMP.



Prof.ª Dr.ª Itana Maria de Souza Gimenes
DIN - UEM.

Uma Abordagem Baseada em Características para o Estabelecimento de Contratos Eletrônicos para Serviços Web

Marcelo Fantinato

Dezembro de 2007

Banca Examinadora:

- Dra. Itana Maria de Souza Gimenes – Co-orientadora (Presidente) (DIN-UEM)
- Dr. Marcos Roberto da Silva Borges (NCE-UFRJ)
- Dra. Regina Maria Maciel Braga (DCC-UFJF)
- Dr. Edmundo Roberto Mauro Madeira (IC-Unicamp)
- Dra. Islene Calciolari Garcia (IC-Unicamp)
- Dr. Marcelo Morandini (Suplente) (EACH-USP)
- Dr. Neucimar Jerônimo Leite (Suplente) (IC-Unicamp)

Resumo

Contratos eletrônicos descrevem processos de negócio interorganizacionais em termos de fornecimento e consumo de serviços eletrônicos (atualmente serviços Web). O estabelecimento de contratos eletrônicos em um determinado domínio de negócios normalmente envolve um conjunto de pontos comuns e pontos de variação bem definidos. Essas propriedades não são completamente exploradas pelas abordagens existentes para o estabelecimento de contratos eletrônicos. Modelos de características têm sido amplamente usados para capturar e gerenciar pontos comuns e variabilidades em famílias de produtos no contexto de linha de produto de software. Esta tese apresenta uma abordagem baseada em modelos de características para ser aplicada no estabelecimento de contratos eletrônicos para serviços Web. Seu objetivo é melhorar a estruturação e o reuso de informação em contratos eletrônicos, incluindo atributos de qualidade de serviço. Características são usadas para representar possíveis elementos de contratos eletrônicos para serviços Web com o objetivo de orientar a criação de instâncias de molde de contrato eletrônico e atuar, portanto, como um gerenciador de espaço de configuração. A abordagem proposta usa as linguagens WS-BPEL e WS-Agreement para especificar, respectivamente, processos de negócio e atributos de qualidade de serviço. Para oferecer apoio automatizado à abordagem proposta, um conjunto de ferramentas, chamado FeatureContract, foi desenvolvido como parte deste trabalho. Para demonstrar a aplicabilidade da abordagem, foi realizado um estudo de caso dentro do contexto de sistemas para telecomunicações.

Abstract

Electronic contracts describe inter-organizational business processes in terms of supply and consumption of electronic services (commonly Web services). The establishment of electronic contracts in a particular business domain usually involves a set of well-defined common and variation points. These properties are not fully exploited by the existing electronic contract establishment approaches. Feature modeling is a software engineering technique that has been widely used for capturing and managing commonalities and variabilities of product families in the context of software product line. This thesis presents a feature-based approach to be applied in the establishment of Web services electronic contracts. The approach aims at improving the structure and reuse of information in electronic contracts, including quality of service attributes. Features are used to represent possible elements in Web services electronic contracts in order to drive contract template instantiation and act, therefore, as a configuration space manager. The proposed approach uses WS-BPEL and WS-Agreement languages to specify business processes and quality of service attributes respectively. To provide automatic support for the proposed approach, a toolkit, named FeatureContract, was developed as part of this work. Aiming at demonstrating the approach applicability, a case study was carried within the telecommunication systems context.

Agradecimentos

Agradeço a minhas orientadoras Beatriz e Itana pelo excelente apoio oferecido. Suas atuações, de forma harmoniosamente complementares, foram de valiosa importância para a realização deste trabalho.

Agradeço a Sara, minha esposa e companheira inseparável, pelos estímulos oferecidos e pelo uso de sua experiência na indicação de melhores caminhos a seguir.

Sumário

CAPÍTULO 1	INTRODUÇÃO.....	1
1.1	OBJETIVOS DA TESE	4
1.2	ESTRUTURA DA TESE.....	5
CAPÍTULO 2	CONTRATOS ELETRÔNICOS PARA SERVIÇOS WEB.....	9
2.1	EXEMPLO DE DOMÍNIO DE APLICAÇÃO.....	10
2.2	CONTRATOS ELETRÔNICOS.....	12
2.2.1	<i>Partes de Contratos Eletrônicos.....</i>	<i>14</i>
2.2.2	<i>Ciclo de Vida de Contratos Eletrônicos.....</i>	<i>17</i>
2.2.3	<i>Meta-Modelos e Moldes de Contrato Eletrônico.....</i>	<i>18</i>
2.2.4	<i>Linguagem de Especificação de Contratos Eletrônicos.....</i>	<i>20</i>
2.3	SERVIÇOS WEB.....	21
2.4	CONTRATOS ELETRÔNICOS PARA SERVIÇOS WEB.....	23
2.4.1	<i>WSDL.....</i>	<i>25</i>
2.4.2	<i>WS-BPEL.....</i>	<i>28</i>
2.4.3	<i>WS-Agreement.....</i>	<i>30</i>
2.5	CONSIDERAÇÕES FINAIS.....	34
CAPÍTULO 3	LINHA DE PRODUTO DE SOFTWARE E MODELOS DE CARACTERÍSTICAS.....	35
3.1	LINHA DE PRODUTO DE SOFTWARE.....	36
3.1.1	<i>Engenharia de Domínio.....</i>	<i>38</i>
3.1.2	<i>Engenharia de Aplicação.....</i>	<i>41</i>
3.1.3	<i>Gerência da Linha de Produto de Software.....</i>	<i>43</i>
3.2	MODELOS DE CARACTERÍSTICAS.....	44
3.2.1	<i>Meta-Modelo de Características Baseado em Cardinalidade.....</i>	<i>47</i>
3.2.2	<i>Restrições Adicionais de Características.....</i>	<i>49</i>
3.2.3	<i>Apoio Automatizado a Modelos de Características.....</i>	<i>50</i>
3.2.4	<i>Método FORM.....</i>	<i>53</i>
3.3	CONSIDERAÇÕES FINAIS.....	54
CAPÍTULO 4	ABORDAGEM PARA O ESTABELECIMENTO DE CONTRATOS ELETRÔNICOS PARA SERVIÇOS WEB.....	57
4.1	PREMISSAS DA ABORDAGEM.....	58
4.2	VISÃO GERAL DA ABORDAGEM.....	59
4.2.1	<i>Estágios do Processo.....</i>	<i>60</i>
4.2.2	<i>Artefatos Produzidos.....</i>	<i>63</i>
4.2.3	<i>Papéis Envolvidos.....</i>	<i>64</i>
4.3	ELABORAÇÃO DOS MODELOS DE CARACTERÍSTICAS PARA SERVIÇOS ELETRÔNICOS.....	66
4.3.1	<i>Estrutura de Modelo de Características.....</i>	<i>69</i>

4.3.2	<i>Restrições Adicionais de Características</i>	74
4.4	CRIAÇÃO DO MOLDE DE CONTRATO ELETRÔNICO PARA SERVIÇOS WEB.....	74
4.4.1	<i>Meta-modelo de Contrato Eletrônico para serviços Web</i>	79
4.4.2	<i>Mapeamento entre Meta-modelos de Características e de Contrato Eletrônico</i>	84
4.5	CONFIGURAÇÃO DOS MODELOS DE CARACTERÍSTICAS PARA SERVIÇOS ELETRÔNICOS	90
4.6	CRIAÇÃO DA INSTÂNCIA DE CONTRATO ELETRÔNICO PARA SERVIÇOS WEB.....	93
4.7	TRABALHOS RELACIONADOS.....	95
4.8	CONSIDERAÇÕES FINAIS.....	101
CAPÍTULO 5	CONJUNTO DE FERRAMENTAS FEATURECONTRACT	105
5.1	ARQUITETURA E INTERAÇÕES DO CONJUNTO DE FERRAMENTAS	106
5.2	FERRAMENTAS DE FEATURECONTRACT	108
5.2.1	<i>FeaturePlugin</i>	109
5.2.2	<i>XSLTransformer</i>	113
5.2.3	<i>WSDL Editor</i>	116
5.2.4	<i>XML Editor</i>	117
5.2.5	<i>ActiveBPEL Designer</i>	118
5.2.6	<i>WS-Contract Factory</i>	121
5.3	CONSIDERAÇÕES FINAIS.....	124
CAPÍTULO 6	ESTUDO DE CASO	127
6.1	DOMÍNIO DA APLICAÇÃO.....	127
6.2	ARTEFATOS PRODUZIDOS.....	130
6.2.1	<i>Modelos de Características para Serviços Eletrônicos</i>	130
6.2.2	<i>Molde de Contrato Eletrônico para Serviços Web</i>	139
6.2.3	<i>Configurações dos Modelos de Características para Serviços Eletrônicos</i>	151
6.2.4	<i>Instância de Contrato Eletrônico para Serviços Web</i>	156
6.3	ANÁLISE DO GRAU DE VARIABILIDADE	163
6.4	ANÁLISE DOS RESULTADOS DO ESTUDO DE CASO.....	165
6.5	CONSIDERAÇÕES FINAIS.....	169
CAPÍTULO 7	CONCLUSÕES.....	171
7.1	TRABALHOS FUTUROS.....	174
APÊNDICE A	MODELOS DE CARACTERÍSTICAS.....	177
A.1	SISTEMA COB	177
A.2	SISTEMA CRM.....	182
APÊNDICE B	MOLDE DE CONTRATO ELETRÔNICO.....	185
B.1	SEÇÃO DE DEFINIÇÕES WSDL	185
B.2	SEÇÃO DE TERMOS DE WS-AGREEMENT	191
B.3	SEÇÃO DE PROCESSO DE NEGÓCIO WS-BPEL.....	199
APÊNDICE C	ARTIGOS PUBLICADOS E EM AVALIAÇÃO	203
C.1	ARTIGOS PUBLICADOS.....	203
C.2	ARTIGO EM AVALIAÇÃO.....	204
REFERÊNCIAS BIBLIOGRÁFICAS.....		205

Lista de Figuras

Figura 01: Processo de negócio “compra de bilhetes”.....	10
Figura 02: Processo de negócio “compra de bilhetes”.....	11
Figura 03: Exemplo de meta-modelo de contrato eletrônico.	19
Figura 04: Processo de desenvolvimento de uma LP [34].	39
Figura 05: Exemplo de modelo de características [67].....	45
Figura 06: Meta-modelo de características baseado em cardinalidade [33].....	47
Figura 07: Exemplo de modelo de características – ferramenta FeaturePlugin.....	51
Figura 08: Exemplo de configuração de modelo de características – FeaturePlugin.	52
Figura 09: Processo de desenvolvimento do método FORM [37].....	54
Figura 10: Processo de estabelecimento de contrato eletrônico com base em características. .	61
Figura 11: Relacionamento entre os artefatos produzidos pelo processo.	64
Figura 12: Exemplo de modelo de características para serviços eletrônicos.	68

Figura 13: Estrutura de modelo de características para serviços eletrônicos.	69
Figura 14: Meta-modelo de contrato eletrônico para serviços Web	80
Figura 15: Mapeamento entre características de Serviços Eletrônicos e Definições WSDL....	85
Figura 16: Mapeamento entre características de Atributos de QoS e Termos de WS-Agreement.	86
Figura 17: Exemplo de configuração de modelo de características para serviços eletrônicos..	92
Figura 18: Arquitetura do conjunto de ferramentas FeatureContract.	106
Figura 19: Seqüência de execução das ferramentas FeatureContract.	107
Figura 20: Ferramenta FeaturePlugin.	110
Figura 21: Painel de Restrições.	113
Figura 22: Ferramenta WSDL Editor.	117
Figura 23: Ferramenta XML Editor.	119
Figura 24: Ferramenta ActiveBPEL Designer.	120
Figura 25: Modelo de características para serviços eletrônicos do Sistema COB (características relacionadas aos serviços eletrônicos).	131
Figura 26: Modelo de características para serviços eletrônicos do Sistema COB (características relacionadas aos atributos de QoS).	132

Figura 27: Modelo de características para serviços eletrônicos do Sistema CRM (características relacionadas aos serviços eletrônicos).	133
Figura 28: Modelo de características para serviços eletrônicos do Sistema CRM (características relacionadas aos atributos de QoS).	134
Figura 29: Painel de propriedades para uma característica do tipo ‘atributo de QoS’.	137
Figura 30: Painel de restrições adicionais.	138
Figura 31: Molde de contrato – seção <wsdl:Definitions> (Sistema COB).	140
Figura 32: Molde de contrato – seção <wsdl:Definitions> (Sistema CRM).	141
Figura 33: Molde de contrato – seção <wsag:Terms>, Parte I (Sistema COB).	144
Figura 34: Molde de contrato – seção <wsag:Terms>, Parte II (Sistema COB).	145
Figura 35: Molde de contrato – seção <bpel:Process>.	148
Figura 36: Configuração do modelo de características para serviços eletrônicos do Sistema COB (características relacionadas aos serviços eletrônicos).	152
Figura 37: Configuração do modelo de características para serviços eletrônicos do Sistema COB (características relacionadas aos atributos de QoS).	153
Figura 38: Configuração do modelo de características para serviços eletrônicos do Sistema CRM.	154
Figura 39: Contrato eletrônico – seção <wsdl:Definitions> (Sistema COB).	157

Figura 40: Contrato eletrônico – seção <wsdl:Definitions> (Sistema CRM).	158
Figura 41: Contrato eletrônico – seção <wsag:Terms>, Parte I (Sistema COB).	159
Figura 42: Contrato eletrônico – seção <wsag:Terms>, Parte II (Sistema COB).	160
Figura 43: Contrato eletrônico – seção <bpel:Process>.	161
Figura 44: Modelo de Características – Sistema COB (Serviços Eletrônicos / Parte 1).	178
Figura 45: Modelo de Características – Sistema COB (Serviços Eletrônicos / Parte 2).	179
Figura 46: Modelo de Características – Sistema COB (Atributos de QoS / Parte 1).	180
Figura 47: Modelo de Características – Sistema COB (Atributos de QoS / Parte 2).	181
Figura 48: Modelo de Características – Sistema CRM (Serviços Eletrônicos).	182
Figura 49: Modelo de Características – Sistema CRM (Atributos de QoS).	183
Figura 50: Molde de Contrato Eletrônico – Seção WSDL (Sistema COB – Parte 1).	186
Figura 51: Molde de Contrato Eletrônico – Seção WSDL (Sistema COB – Parte 2).	187
Figura 52: Molde de Contrato Eletrônico – Seção WSDL (Sistema CRM).	188
Figura 53: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Service Properties (Sistema CRM).	192
Figura 54: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Guarantee Terms (Sistema CRM / Parte 1).	193

Figura 55: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Guarantee Terms (Sistema CRM / Parte 2).....	194
Figura 56: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Service Properties (Sistema COB).....	195
Figura 57: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Guarantee Terms (Sistema COB/ Parte 1).....	196
Figura 58: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Guarantee Terms (Sistema COB/ Parte 2).....	197
Figura 59: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Guarantee Terms (Sistema COB/ Parte 3).....	198
Figura 60: Molde de Contrato Eletrônico – Seção de processo de negócio WS-BPEL (visão geral do processo).	200
Figura 61: Molde de Contrato Eletrônico – Seção de processo de negócio WS-BPEL (visão expandida de Aplicar-AC e Reverter-AC).	201

Lista de Tabelas

Tabela 01: Papéis das organizações envolvidas.	65
Tabela 02: Opções de obrigatoriedade para cada tipo de característica.	73
Tabela 03: Sigla dos atributos para cada tipo de característica.	73
Tabela 04: Comparação entre as abordagens similares à proposta nesta tese.	100
Tabela 05: Medidas de Variabilidade.	164

Lista de Códigos

Código 01: Exemplo de especificação WSDL.....	27
Código 02: Exemplo de especificação WS-BPEL.....	30
Código 03: Exemplo de especificação WS-Agreement.....	33
Código 04: Exemplo de molde de contrato eletrônico para serviços Web – seção de definições WSDL (sistema de Cobrança).....	75
Código 05: Exemplo de molde de contrato eletrônico para serviços Web – seção de termos de WS-Agreement (sistema de Cobrança).....	76
Código 06: Exemplo de molde de contrato eletrônico para serviços Web – seção processo de negócio (entre os dois sistemas).....	77
Código 07: Exemplos de anotação no molde de contrato eletrônico.....	115
Código 08: Molde de contrato – seção <wsdl:Definitions>, elementos <partnerLinkType> (Sistema COB).....	143
Código 09: Molde de contrato – seção <bpel:Process>, elementos <partnerLink> e <variable>.....	150

Código 10: Contrato eletrônico – seção <wsdl:Definitions>, elementos <partnerLinkType> (Sistema COB)	159
Código 11: Contrato eletrônico – seção <bpel:Process>, elementos <partnerLink> e <variable>	162
Código 12: Molde de Contrato Eletrônico – Seção WSDL, Partner Link Type (Sistema COB).	189
Código 13: Molde de Contrato Eletrônico – Seção WSDL, Partner Link Type (Sistema CRM).	190
Código 14: Molde de Contrato Eletrônico – Seção WS-BPEL, Partner Links.....	202

Capítulo 1

Introdução

Organizações estão trabalhando cada vez mais de forma cooperativa para alcançar seus objetivos de negócio por meio da realização de processos de negócio interorganizacionais, normalmente usando um paradigma de subcontratação de serviços. O dinamismo existente no mercado requer que companhias ajam rapidamente para não perder parcerias ou oportunidades. O uso da Internet e de Sistemas Gerenciadores de Processos de Negócio (SGPN) são passos tecnológicos importantes para melhorar a cooperação entre as organizações [1, 2], principalmente por meio do comércio eletrônico entre organizações virtuais disponíveis em mercados eletrônicos [3]. Além disso, o recente paradigma de Computação Orientada por Serviços (COS) [4] oferece uma série de facilidades para a integração entre aplicações na Internet por meio do uso de serviços eletrônicos.

Dentre os diferentes tipos de serviços eletrônicos, os serviços Web [5] emergiram como a mais promissora tecnologia para a efetiva automação de interações interorganizacionais [6]. Eles são um tipo específico de serviços eletrônicos para serem usados principalmente na Internet. Os serviços Web são baseados em tecnologias padrão da indústria, tais como WSDL (*Web Services Description Language*) [7], UDDI (*Universal Description, Discovery, and Integration*) [8], e SOAP (*Simple Object Access Protocol*) [9] – todos eles baseados na linguagem XML (*eXtensible Markup Language*) [10]. O uso desses padrões torna mais fácil a publicação, a descoberta e a invocação automáticas de serviços eletrônicos na Internet. Os processos de negócio podem ser especificados em termos de

serviços Web utilizando linguagens tais como WS-BPEL (*Web Services Business Process Execution Language*) [11]. Além disso, linguagens, tais como WS-Agreement (*Web Services Agreement Specification*) [12, 13] e WS-Policy (*Web Services Policy Framework*) [14, 15], podem ser usadas para especificar atributos de qualidade de serviço (QoS¹) relacionados a serviços Web.

No contexto de negócios eletrônicos, contratos eletrônicos são usados para descrever detalhes sobre o fornecimento e o consumo de serviços eletrônicos em um processo de negócio [16, 17]. Eles podem também incluir atributos de QoS acordados entre as partes envolvidas [18, 19]. A atual complexidade envolvida no estabelecimento de contratos eletrônicos pode desencorajar a criação de novas parcerias de negócio. Os principais aspectos dessa complexidade são: a grande quantidade de informação necessária para o estabelecimento de contratos eletrônicos; o crescente número de parâmetros a serem considerados; a longa duração das negociações para o estabelecimento de contratos; e, o envolvimento de diferentes perfis (tanto equipes de negócio como equipes de desenvolvimento) de diferentes organizações. Um tipo específico de contrato eletrônico é aquele que envolve serviços Web, foco desta tese.

Para reduzir a complexidade do estabelecimento de contratos eletrônicos, é necessário melhorar sua estruturação e facilitar o reuso de informação nesse contexto. Muitas abordagens para o estabelecimento de contratos eletrônicos usam moldes de contrato [17, 20–30] para minimizar esses problemas. Os moldes de contrato eletrônico, em sua grande maioria, são tratados como simples documentos eletrônicos que possuem campos vazios a serem preenchidos com algum valor, normalmente a partir de uma lista pré-definida, durante o estabelecimento de contratos. Esses moldes normalmente não permitem uma representação estruturada, bem como a seleção de partes de contratos eletrônicos que sejam opcionais ou alternativas. De um modo geral, as abordagens existentes não oferecem mecanismos adequados para a gerência de elementos comuns e elementos variáveis de contratos eletrônicos. Embora elas tenham contribuído com uma série de avanços nos

¹ Do inglês: Quality of Service

aspectos de elaboração e de realização de contratos eletrônicos, ainda são limitadas no que diz respeito ao reuso de informação entre contratos eletrônicos similares.

Nesta tese, é proposta uma nova abordagem para reduzir a complexidade no estabelecimento de contratos eletrônicos para serviços Web e melhorar a estruturação e o reuso de informação. Tal abordagem é baseada em conceitos de linha de produto de software [31], principalmente em modelos de características² [32]. A maior contribuição da abordagem proposta é oferecer uma forma sistemática e eficiente para a estruturação e o reuso de informação, aperfeiçoando assim o processo de estabelecimento de contratos eletrônicos. A abordagem oferece um meio para representar informações de contratos eletrônicos por meio de modelos de características que podem ser transformados em moldes de contrato eletrônico. Assim, é possível gerenciar de forma eficiente as partes obrigatórias, opcionais e alternativas de contratos eletrônicos, não apenas relativas às ações envolvidas no processo de negócio, mas também aos atributos de QoS associados aos serviços Web contratados.

Linha de produto de software (LP) [31, 33-35] é uma abordagem da Engenharia de Software que promove a geração de produtos específicos, a partir de uma família de produtos, com base no reuso de uma infra-estrutura central bem definida. Nesse contexto, são explorados, de um modo sistemático, os pontos comuns e as variabilidades entre sistemas de um mesmo domínio. A infra-estrutura central é formada basicamente por uma arquitetura de software comum e um conjunto de componentes que podem ou não ser obrigatórios para os produtos a serem desenvolvidos. Com base nessa infra-estrutura, diferentes produtos de software, porém bastante similares – por serem parte de um mesmo domínio de software, são gerados. O processo de geração de produtos envolve o ajuste dos componentes às necessidades específicas de cada produto específico. Assim como o desenvolvimento de software, o estabelecimento de contratos eletrônicos também demanda um eficiente reuso de informação. Desse modo, ambas as áreas podem obter vantagens da

² Do inglês: features

abordagem de LP para alcançar uma melhor compreensão de domínio e reduzir o tempo para disponibilizar produtos para o mercado (o chamado tempo-para-mercado³).

A maioria das técnicas de LP usa modelos de características [32] para capturar e gerenciar pontos comuns e variabilidades [33–41]. Esses modelos descrevem propriedades de entidades em diferentes níveis de abstração, por meio de diagramas que representam características em uma estrutura hierárquica de nós. Eles constituem uma representação de informação simples, estruturada e de fácil entendimento, que pode ser usada em diferentes fases do ciclo de desenvolvimento de software. As facilidades oferecidas por modelos de características podem também ser úteis no estabelecimento de contratos eletrônicos para serviços Web, tornando os contratos eletrônicos mais fáceis para entender e melhorando o reuso de informação.

1.1 Objetivos da Tese

O objetivo desta tese é propor uma abordagem que visa reduzir a complexidade do estabelecimento de contratos eletrônicos para serviços Web. São propostas melhorias tanto na estruturação como nas facilidades de gerência e reuso das informações envolvidas nos contratos eletrônicos. A abordagem é inspirada na técnica de LP, usando principalmente os conceitos de modelos de características.

De um modo geral, a abordagem proposta consiste de um conjunto de estágios derivados de um processo de LP específico – o método FORM (*Feature-Oriented Reuse Method*) [37]. Modelos de características são usados para a representação genérica de serviços eletrônicos e possíveis níveis de atributos de QoS – os quais permitem a derivação de um molde de contrato eletrônico para serviços Web. O processo de negócio e os atributos de QoS integrantes dos moldes de contrato eletrônico são especificados, respectivamente, por meio das linguagens WS-BPEL e WS-Agreement.

³ Do inglês: time-to-market

As atividades do processo de estabelecimento de contratos eletrônicos, incluindo a negociação existente entre as partes envolvidas, são orientadas direta ou indiretamente por modelos de características e pelas suas possíveis configurações. Cada configuração possível dos modelos de características representa o conjunto de serviços eletrônicos que devem ser acordados entre ambas as partes envolvidas, incluindo os detalhes associados, e os níveis dos atributos de QoS também acordados. Os serviços eletrônicos genéricos que forem contratados são mapeados para serviços Web específicos que implementem suas operações, por meio de um relacionamento de um-para-um. Esses serviços Web são por fim referenciados nos contratos eletrônicos resultantes do processo de estabelecimento dos contratos.

É também objetivo desta tese, mostrar a possibilidade de automação da abordagem proposta e evidenciar a viabilidade de sua aplicação destacando seus benefícios. Para o primeiro caso, é apresentado o conjunto de ferramentas FeatureContract (*Feature Modeling based Web Services E-Contracts establishment toolkit*). Para o segundo caso, são usados resultados de um estudo de caso que mostram que a possibilidade de gerência de pontos comuns e de variabilidades oferecida pelos modelos de características torna mais fácil o estabelecimento de contratos eletrônicos, para ambos os lados envolvidos – fornecedores e consumidores de serviços eletrônicos.

Esta tese não possui o objetivo de tratar aspectos legais de contratos eletrônicos, tais como questões envolvendo assinaturas eletrônicas e segurança. São tratados apenas os aspectos técnico-funcionais, como os descritos nos parágrafos anteriores.

1.2 Estrutura da Tese

Esta tese contém seis capítulos (incluindo este capítulo de Introdução), dois apêndices e as referências bibliográficas. Nessa Introdução, foram apresentados o contexto geral em que o assunto específico desta tese está inserido, bem como as motivações e os objetivos da

abordagem proposta. Os assuntos apresentados em cada um dos outros capítulos e apêndices são descritos a seguir.

No Capítulo 2 é apresentada uma visão geral sobre contratos eletrônicos. Inicialmente são apresentados os conceitos básicos sobre contratos eletrônicos e moldes de contratos de uma forma geral, com ênfase nas duas partes principais: processos de negócio e atributos de QoS. Em seguida, são apresentados conceitos básicos de serviços Web para embasar a apresentação da abordagem proposta, incluindo duas das principais linguagens de especificação de processos de negócio e de atributos de QoS para serviços Web.

No Capítulo 3 são apresentadas visões gerais sobre a técnica de LP e sobre modelos de características. Primeiramente, são apresentados conceitos mais genéricos relacionados à abordagem de LP, contextualizando a área dentro da qual os modelos de características foram desenvolvidos. Em seguida, são apresentados os conceitos relacionados a modelos de características de uma forma geral, visando oferecer as motivações pelas quais esses modelos oferecem benefícios ao serem usados no contexto de contratos eletrônicos. Em seguida, é apresentado o meta-modelo específico de características, baseado em cardinalidade, que é usado na formalização da abordagem proposta nesta tese.

No Capítulo 4 é apresentada a abordagem para o estabelecimento de contratos eletrônicos para serviços Web, incluindo os atributos de QoS, que é baseada em modelos de características. A abordagem tem cinco estágios: 1) a elaboração dos modelos de características para serviços eletrônicos; 2) a criação de moldes de contratos eletrônicos para serviços Web; 3) o desenvolvimento de serviços Web a serem contratados e posteriormente usados na realização do contrato eletrônico; 4) a configuração dos modelos de características para serviços eletrônicos; e 5) a criação da instância de contrato eletrônico resultante. Todos esses estágios são detalhados nesse capítulo, com exceção do Estágio 3 – relacionado ao desenvolvimento de serviços Web, por estar fora do escopo desta tese. No final do capítulo, depois da apresentação da abordagem, são discutidos os principais trabalhos sobre contratos eletrônicos relacionados a tal abordagem.

No Capítulo 5 é apresentado o conjunto de ferramentas FeatureContract, desenvolvido para oferecer apoio automatizado à abordagem proposta. FeatureContract é composto por uma série de ferramentas, responsáveis por oferecer apoio a diferentes estágios do processo de estabelecimento de contratos eletrônicos para serviços Web.

No Capítulo 6 é apresentado um estudo de caso realizado para demonstrar a aplicabilidade da abordagem proposta. Ele é realizado dentro do contexto de sistemas para empresas operadoras de telecomunicações – abordando a integração de um sistema de atendimento a clientes e um sistema de cobrança. Exemplos de todos os artefatos produzidos pela aplicação da abordagem proposta são apresentados nesse capítulo. Além disso, em função do estudo de caso apresentado, é realizada uma análise sobre o grau de variabilidade existente no objeto do estudo de caso.

No Capítulo 7 são apresentadas as conclusões finais, incluindo uma visão geral da análise dos resultados com o desenvolvimento deste trabalho e com a realização do estudo de caso. Além disso, esse capítulo inclui trabalhos futuros relacionados a esta tese.

No Apêndice 1 e no Apêndice 2 são apresentados os artefatos completos desenvolvidos como parte da realização do estudo de caso apresentado nesta tese. Os modelos de características são apresentados no Apêndice 1. O molde de contrato eletrônico é apresentado no Apêndice 2.

No Apêndice 3 são apresentados os artigos relacionados a esta tese que já foram publicados ou que estão em processo de avaliação.

Capítulo 2

Contratos Eletrônicos para Serviços

Web

Contratos eletrônicos são usados para descrever detalhes sobre o fornecimento e o consumo de serviços eletrônicos em um processo de negócio, podendo incluir atributos de qualidade de serviço (QoS) acordados entre as partes envolvidas. Na abordagem proposta nesta tese, são considerados especificamente os contratos eletrônicos para serviços Web – um tipo específico de contrato eletrônico. Esse tipo de serviço faz parte de uma tecnologia promissora para a integração de sistemas executados na Internet. Neste capítulo é apresentada uma visão geral sobre contratos eletrônicos com o objetivo de contextualizar a abordagem proposta.

Para a ilustração dos conceitos deste capítulo, primeiramente é apresentado um exemplo de domínio de aplicação para contratos eletrônicos. Depois, são apresentados os conceitos básicos relacionados a contratos eletrônicos e moldes de contratos de uma forma geral, não necessariamente direcionados a serviços Web, com ênfase em: processo de negócio e atributos de QoS. Em seguida, são apresentados os conceitos básicos de serviços Web e informações relacionadas especificamente a contratos eletrônicos para serviços Web, incluindo a linguagem de especificação de serviços Web WSDL e duas das principais linguagens de especificação de processos de negócio e de atributos de QoS – WS-BPEL e WS-Agreement, respectivamente.

2.1 Exemplo de Domínio de Aplicação

Nesta seção é apresentado um exemplo básico de domínio de aplicação para o qual um contrato eletrônico pode ser estabelecido. Além disso, é apresentado um exemplo de contrato eletrônico para esse domínio de aplicação.

O exemplo trata do processo de negócio de intermediação para a compra de bilhetes aéreos realizada por uma agência de viagens. Ela interage com seus clientes e com uma companhia aérea com a qual a busca, a reserva e a compra são realizadas. Esse exemplo é baseado no apresentado por Leymann & Roller [48]. O processo de negócio “compra de bilhetes” que envolve essas duas organizações está representado, como um diagrama de seqüência da UML [54], na Figura 01.

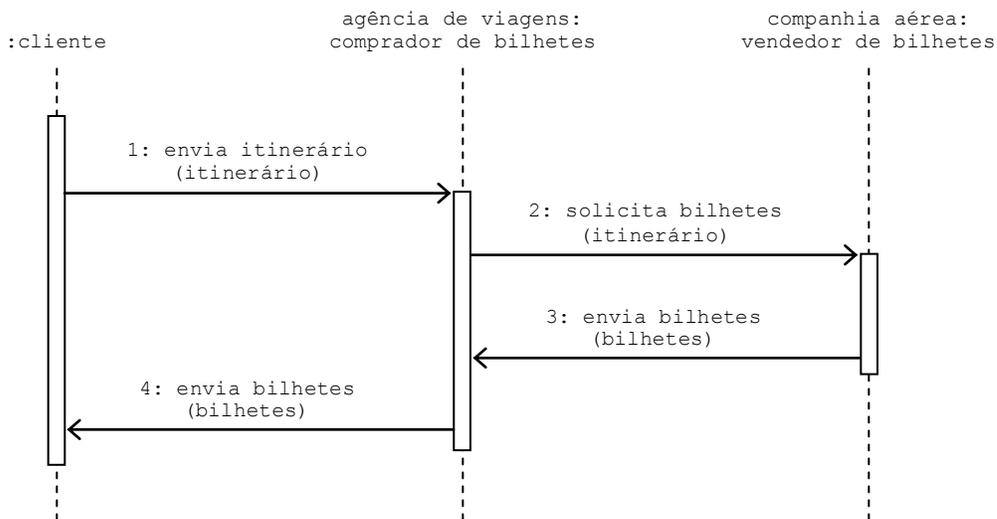


Figura 01: Processo de negócio “compra de bilhetes”.

Depois de receber de um **cliente** um **itinerário**, a **agência de viagens**, desempenhando o papel de **comprador de bilhetes**, **solicita bilhetes** à **companhia aérea**, repassando para ela o **itinerário** em questão. A **companhia aérea** por sua vez,

desempenhando o papel de **vendedor de bilhetes**, após a emissão **envia bilhetes** para a **agência de viagens**, que repassa os **bilhetes** ao seu **cliente**.

A parte do processo de negócio para a qual um contrato eletrônico pode ser necessário é a que engloba a interação entre a agência de viagens e a companhia aérea. A Figura 02 apresenta um exemplo de um contrato eletrônico estabelecido para o processo representado na Figura 01. O exemplo é apenas ilustrativo, sem detalhes de especificação. Os diferentes elementos envolvidos no processo, descritos acima, são apresentados em diferentes seções do contrato. Cada um desses diferentes tipos de elementos é abordado apropriadamente nas próximas seções deste capítulo.

```
contrato compra_de_bilhetes

papeis:
  comprador_de_bilhetes
  vendedor_de_bilhetes

partes:
  agencia_de_viagem : comprador_de_bilhetes
  companhia_aerea : vendedor_de_bilhetes

atividades:
  agencia_de_viagem.solicita_bilhetes
  agencia_de_viagem.recebe_bilhetes

  companhia_aerea.recebe_itinerario
  companhia_aerea.envia_bilhetes

mensagens:
  agencia_de_viagens.itinerario
  companhia_aerea.bilhetes

processo:
  agencia_de_viagem.solicita_bilhetes(itinerario)
  companhia_aerea.recebe_itinerario(itinerario)

  companhia_aerea.envia_bilhetes(bilhetes)
  agencia_de_viagem.recebe_bilhetes(bilhetes)

clausulas_contratuais:
  1. companhia_aerea.* disponivel_em 7X14
  2. companhia_aerea.solicita_bilhetes responde_em 60_segundos
  3. companhia_aerea.* exige_autenticacao_por usuario
  4. agencia_de_viagens.* exige_autenticacao_por grupo_do_usuario
```

Figura 02: Processo de negócio “compra de bilhetes”.

Apenas a última seção, `clausulas_contratuais`, não tem um mapeamento direto com o exemplo da Figura 01. Esses elementos definem algumas obrigações que as organizações envolvidas devem satisfazer. Essas obrigações definem que:

1. Todas as atividades (representadas pelo símbolo `*`) do parceiro `companhia_aerea` devem estar disponíveis para execução em uma taxa de 7X14 (14 horas por dia, em sete dias por semana);
2. A atividade `envia_bilhetes` do sistema do parceiro `companhia_area` deve ser executado em no máximo `60_segundos`;
3. Todas as atividades de ambos os parceiros devem exigir autenticação eletrônica ao ser invocado pelo outro parceiro, porém com diferentes níveis: a `companhia_area` deve `exige_autenticacao_por_usuario`, enquanto que a `agencia_de_viagens` apenas `por_grupo_do_usuario`.

2.2 Contratos Eletrônicos

Um contrato é um acordo entre duas ou mais partes interessadas em criar relacionamentos mútuos nos negócios ou obrigações legais. Ele define um conjunto de atividades a serem executadas por cada parte, as quais devem satisfazer um conjunto de termos e condições conhecidos como cláusulas contratuais. Contratos são necessários em muitos tipos de transações de negócio. Eles são fundamentais para que organizações comprometam-se em relações comerciais. Praticamente toda transação comercial que cruza as fronteiras de uma organização é acompanhada, implícita ou explicitamente, de um contrato.

Contratos são geralmente documentos volumosos preenchidos com jargões legais e sem facilidades para encontrar cláusulas contratuais relevantes cujo cumprimento precisa ser

considerado pelos parceiros de negócio. Primeiramente, um contrato deve especificar exatamente o produto ou serviço a ser comercializado de modo que ambos fornecedores e consumidores saibam o que eles podem esperar e o que é esperado deles. Além disso, um contrato deve estabelecer as regras de relacionamento de negócio, tais como obrigações e proibições. O contrato deve ainda conter informações para apoiar eventuais julgamentos em caso de discordâncias posteriores.

Um contrato eletrônico é um documento eletrônico usado para representar um acordo entre organizações parceiras que estão executando processos de negócio por meio da Internet, nos quais os serviços negociados são serviços eletrônicos. Contratos eletrônicos podem variar desde um simples pedido de compra para a venda de produtos pela Internet até documentos extremamente complexos para um acordo comercial entre parceiros de negócio multinacionais. Eles contêm detalhes a respeito do processo de negócio a ser realizado de forma cooperativa entre as organizações, e servem de base para a execução e o acompanhamento do processo definido. Entre os possíveis detalhes contidos em contratos eletrônicos estão informações sobre os serviços eletrônicos a serem executados, dados a serem trocados durante a execução dos serviços eletrônicos, atributos de QoS definidos para esses serviços, custos envolvidos e possíveis operações de controle e monitoramento.

O uso de contratos eletrônicos visa à melhoria da eficiência e da eficácia do processo de estabelecimento de contratos e o oferecimento de novas oportunidades para as organizações envolvidas. Uma abordagem eletrônica para contratos torna possível o estabelecimento de acordos a custos mais baixos, em um período de tempo menor e sem restrições geográficas [45]. De acordo com Angelov & Grefen [46], os benefícios que podem ser obtidos são de três tipos de valores: financeiros, estratégicos e de processo. Os valores estratégicos incluem: casamento de estratégias, vantagem competitiva, informação de gerência e arquitetura de tecnologia de informação estratégica. Os valores de processo incluem: qualidade, tempo do ciclo de vida, agilidade e adaptabilidade. Apesar desse grande conjunto de benefícios, em [46] são apresentados também alguns riscos associados

ao uso de contratos eletrônicos, tais como riscos políticos, de negócio, do ponto de vista legal, de padronização, de reestruturação interna e de segurança.

Além dos aspectos apresentados nos parágrafos anteriores, há ainda os aspectos de legalidade associados a contratos eletrônicos [17]. De acordo com esses aspectos, contratos devem definir, por exemplo, os procedimentos legais para o caso de quebra do acordo e para a necessidade de arbitragem como a definição de foro apropriado. Além disso, uma série de questões deve ser considerada com o objetivo de tornar os contratos eletrônicos válidos do ponto de vista legal. Essas questões normalmente dependem de aspectos envolvendo cada segmento específico de negócio. Do ponto de vista de tecnologia, devem ser considerados itens tais como assinaturas digitais, criptografia de chaves públicas e segurança. Esses aspectos não são focos desta tese. Alguns dos trabalhos que tratam de legalidade em contratos eletrônicos são apresentados em [45, 47].

2.2.1 Partes de Contratos Eletrônicos

Contratos eletrônicos se diferem amplamente em relação a tamanho, conteúdo e complexidade. Entretanto, normalmente existem elementos comuns a um mesmo domínio de contrato eletrônico. Os tipos de elementos mais comuns em contratos eletrônicos são [17]:

- **Partes:** que representam cada uma das organizações envolvidas em um processo de negócio. No exemplo da Seção 2.1, as partes envolvidas no processo “compra de bilhetes” são **agência de viagens** e **companhia aérea**;
- **Atividades:** que representam os serviços eletrônicos a serem executados por meio da realização do contrato eletrônico. Essas atividades são normalmente compostas para formar um processo de negócio. No exemplo da Seção 2.1, as atividades são

envia itinerário, recebe itinerário, envia bilhetes e recebe bilhetes, formando nessa ordem um processo de negócio;

- **Cláusulas contratuais:** que descrevem as restrições, em relação à execução das atividades, a serem satisfeitas durante a realização do contrato eletrônico. Diferentes tipos de cláusulas contratuais são apresentadas posteriormente nesta seção. No exemplo da Seção 2.1, existem quatro cláusulas contratuais, classificadas também posteriormente nesta seção;

Além desses tipos de elementos, os contratos eletrônicos podem também incluir papéis, pagamentos, artefatos de entrada e de saída, e parâmetros de entrada e de saída. No exemplo da Figura 01, existem os seguintes elementos adicionais: papéis – **comprador de bilhetes** e **vendedor de bilhetes**; e mensagens trocadas como parâmetros de entrada/saída – **itinerário** e **bilhetes**.

Podem existir vários tipos de cláusulas contratuais descrevendo diferentes tipos de restrições, dependendo das necessidades dos envolvidos no processo de negócio. Marjanovic et. al. [16] estabelecem que cláusulas de contrato eletrônico podem ser divididas em três tipos de restrições contratuais:

- **Obrigações:** que descrevem o que as partes deveriam fazer – como, por exemplo: “um cliente deve pagar por um serviço de acordo com os termos de pagamento”. No processo “compra de bilhetes”, exemplo da Seção 2.1, a **companhia aérea** pode ser obrigada a oferecer sempre a tarifa mais barata existente na venda de bilhetes para a **agência de viagens**;
- **Permissões (ou Direitos):** que descrevem o que as partes podem fazer – como, por exemplo: “um cliente pode opcionalmente solicitar informações sobre o estado atual do fornecimento de um determinado serviço”. No exemplo da Seção 2.1, a **agência**

de viagens pode ter o direito de cancelar a compra de bilhetes enquanto a emissão dos mesmos não tiver sido finalizada pela **companhia aérea**;

- **Proibições:** que descrevem o que as partes não podem fazer – como, por exemplo: “não é permitido que um cliente cancele um pedido depois que algum serviço já foi executado”. No exemplo da Seção 2.1, a **agência de viagens** não tem o direito de cancelar a compra de bilhetes depois que a emissão dos mesmos já tiver sido finalizada pela **companhia aérea**.

Restrições do tipo “obrigações” normalmente incluem cláusulas contratuais de **atributos de qualidade de serviço (QoS)** associadas aos serviços eletrônicos que definem atributos relacionados a propriedades não-funcionais. Eles afetam a definição e a execução de um serviço eletrônico. Exemplos de atributos de QoS são: disponibilidade, integridade, confiabilidade, segurança, desempenho e tempo de resposta [18, 19, 30]. Normalmente, o grupo de cláusulas contratuais relacionadas a atributos de QoS em um contrato eletrônico é chamado de SLA⁴. Embora SLAs possam ser entendidos como sendo uma parte de contratos eletrônicos, alguns autores se referem a esses dois termos como sinônimos [18, 81, 91, 92].

Em relação a atributos de QoS, um mesmo serviço eletrônico pode ser oferecido por uma organização com diferentes níveis de QoS para diferentes organizações consumidoras – em função do valor que a organização consumidora está disposta a pagar. Isso é similar ao oferecimento de serviços em diferentes categorias, tais como as classes prata, ouro e diamante de uma companhia aérea, por exemplo. Durante o estabelecimento de um contrato eletrônico, para cada atributo de QoS, um valor deve ser definido para ser usado como tolerável (por exemplo, um valor mínimo, máximo, ou exato) durante a realização de um contrato eletrônico.

⁴ Do inglês: Service Level Agreement

No exemplo de processo de negócio “compra de bilhetes”, da Seção 2.1, são definidas quatro cláusulas contratuais do tipo obrigação relacionados a atributos de QoS: um associado à disponibilidade, um associado ao tempo de resposta e dois deles associados à segurança – com diferentes níveis para cada um deles.

2.2.2 Ciclo de Vida de Contratos Eletrônicos

O ciclo de vida de contratos eletrônicos inclui fases relacionadas a: disponibilização de serviços eletrônicos que podem ser ofertados; negociação entre empresas; definição de acordos de negócio; e execução, controle e monitoramento do processo de negócio regulamentado pelos contratos eletrônicos. Embora existam abordagens distintas para descrever o ciclo de vida de contratos eletrônicos, tais como as apresentadas em [3, 20, 49-52], em geral pode-se resumir esse ciclo de vida nas seguintes fases:

1. **Implementação de serviços eletrônicos:** implementação de serviços por parte de organizações fornecedoras. Essa implementação pode ser realizada diretamente por iniciativa das próprias organizações fornecedoras que desejam disponibilizar um serviço eletrônico no mercado, ou então a partir de pedidos de potenciais organizações consumidoras que podem estar interessadas nos serviços;
2. **Disponibilização, busca e descoberta de serviços eletrônicos:** troca de informações entre organizações fornecedoras e organizações consumidoras em mercados virtuais. O anúncio de serviços é realizado pelos fornecedores, enquanto que os consumidores realizam buscas de serviços e comparações entre eles. Informações sobre as garantias de QoS podem ser, por exemplo, disponibilizadas pelas organizações fornecedoras. A descoberta de parcerias ocorre quando existe a

compatibilidade entre serviços publicados e serviços procurados. Essas atividades devem ser executadas com o apoio de facilidades de busca e descoberta⁵;

3. **Negociação e estabelecimento de contratos eletrônicos:** processo de decisão que estabelece como o processo de negócio deverá ser realizado entre a organização fornecedora e a organização consumidora. Durante a negociação são definidas as partes envolvidas, os serviços a serem prestados e as cláusulas contratuais que devem ser cumpridas durante a realização do contrato – incluindo possivelmente cláusulas de garantia de QoS. O processo de negociação pode ser realizado seguindo um protocolo em que os papéis e as responsabilidades são bem definidos por meio de atividades sistemáticas que cada organização envolvida deve seguir [53];

4. **Realização do processo de negócio:** execução e cumprimento dos termos estabelecidos no contrato eletrônico, por meio da execução dos serviços eletrônicos previstos e cumprimento das cláusulas contratuais estabelecidas. Para garantir o cumprimento das cláusulas, elas devem ser monitoradas durante a execução dos serviços eletrônicos. A organização consumidora também pode executar operações de monitoramento do processo conforme restrições estabelecidas no contrato eletrônico.

2.2.3 Meta-Modelos e Moldes de Contrato Eletrônico

Para facilitar o entendimento e o estabelecimento de contratos eletrônicos, existe a necessidade de se usar modelos que formalizem as regras a serem seguidas por um conjunto de contratos eletrônicos e artefatos que agilizem sua criação a partir de

⁵ Conhecidas como atividades de, em inglês, *matchmaking*.

informações pré-definidas. Para tal pode-se usar, respectivamente, meta-modelos de contrato eletrônico e de moldes de contrato eletrônico.

Meta-modelos de contrato eletrônico capturam os detalhes de nível conceitual sobre os possíveis elementos que podem estar envolvidos em um determinado contrato eletrônico. Eles prevêm que tipos de entidades e de relacionamentos entre as entidades podem fazer parte de contratos eletrônicos. Portanto, um meta-modelo de contrato eletrônico define as regras a partir das quais contratos eletrônicos devem ser definidos. Os meta-modelos são normalmente apresentados de modo formal ou semi-formal, por meio de representações gráficas – como, por exemplo, por meio de diagramas de classes da UML [54].

Exemplos de entidades a serem representadas por meta-modelos de contrato eletrônico são: partes, atividades e cláusulas contratuais, conforme apresentadas na Seção 2.2.1 desta tese. A Figura 03 apresenta um exemplo simples de um meta-modelo de contrato eletrônico, por meio de um diagrama de classes da UML. Esse meta-modelo de alto nível de abstração agrupa de forma estruturada os elementos apresentados na Seção 2.2.1, mantendo as regras apresentadas em tal seção.

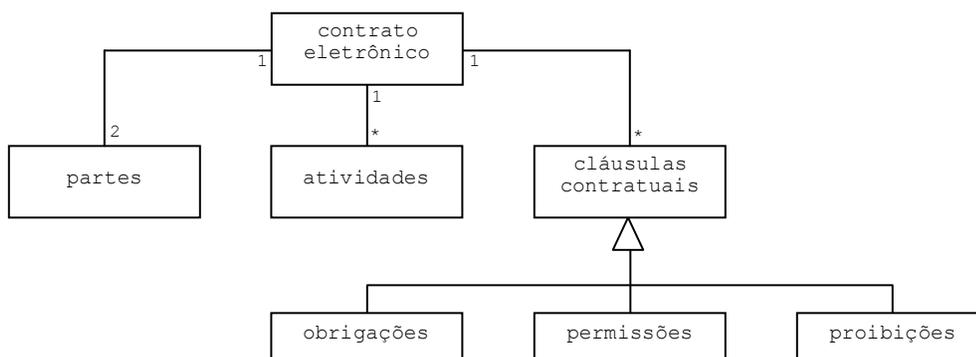


Figura 03: Exemplo de meta-modelo de contrato eletrônico.

Moldes de contrato eletrônico são contratos eletrônicos pré-elaborados que podem ser usados na criação de instâncias específicas de contrato eletrônico. Normalmente eles já são elaborados como um tipo de rascunho de contrato eletrônico, que precisa ser finalizado

para estar completo. Os moldes de contrato eletrônico mais simples são elaborados apenas como um documento eletrônico que possuem campos vazios a serem preenchidos com algum valor, normalmente a partir de uma lista pré-definida, durante o estabelecimento do contrato eletrônico. Tipos mais avançados de moldes de contrato eletrônico, porém mais raros, podem prever partes obrigatórias, opcionais e alternativas que podem ou não ser mantidas em uma determinada instância de contrato eletrônico, dependendo das escolhas tomadas.

Um exemplo de molde de contrato eletrônico seria o contrato apresentado na Figura 02 com alguns campos a serem ainda preenchidos. Exemplos de tais campos poderiam ser: ambas as partes envolvidas e os níveis definidos para os atributos de QoS da seção `clausulas_contratutais` (`7X24`, `60_segundos`, `usuario` e `grupo_do_usuario`).

Durante o estabelecimento de um contrato eletrônico específico, novos itens não previstos em um determinado molde de contrato eletrônico podem ser criados. Normalmente, nesse caso, o molde de contrato é atualizado para passar a conter esses novos itens de contrato – para estarem disponíveis como opções durante as próximas criações de instância de contrato eletrônico. O uso de moldes de contrato eletrônico possibilita que, a cada nova transação de negócio, não seja necessário estabelecer completamente um novo contrato eletrônico. O reuso de contratos eletrônicos pré-estabelecido leva a uma economia significativa.

2.2.4 Linguagem de Especificação de Contratos Eletrônicos

Um contrato eletrônico é normalmente um documento eletrônico altamente estruturado que pode ser empregado em diferentes fases da realização de processos de negócio interorganizacionais e transferido entre diferentes tipos de sistemas de informação, incluindo os Sistemas Gerenciadores de Processo de Negócio (SGPNs). Para facilitar a

especificação de contratos eletrônicos, é necessário uma linguagem de especificação, ou um conjunto de linguagens para diferentes partes de um contrato eletrônico, que permita que os contratos sejam definidos, executados, monitorados e transferidos entre diferentes SGPNs. É também importante que os contratos especificados por meio dessa linguagem de especificação sejam compreensíveis por seres humanos.

Uma das soluções mais usadas atualmente para a especificação de contratos eletrônicos, de uma forma geral, é o uso de uma linguagem de especificação baseada na linguagem XML [10]. XML é uma linguagem de marcação de dados que oferece um padrão para descrever dados estruturados, de modo a facilitar a declaração mais precisa de conteúdo e obter melhor desempenho nas buscas. Apesar da predominância de XML, ainda existem projetos que usam linguagens próprias [20, 29, 91], porém a tendência é que elas sejam substituídas também por XML.

É comum que um mesmo contrato eletrônico seja especificado por meio de duas ou mais diferentes linguagens – cada uma delas aplicadas para diferentes partes do contrato. Por exemplo, duas linguagens de especificação distintas podem ser usadas, uma para a especificação do processo de negócio embutido no contrato eletrônico, e outra para a especificação dos atributos e níveis de QoS acordados entre as partes para os serviços eletrônicos envolvidos.

2.3 Serviços Web

Um serviço Web [5] é um tipo específico de serviço eletrônico que usa padrões abertos da Internet para a sua descrição, busca e requisição de operações. Serviços Web podem ser vistos como aplicações, disponíveis via Internet, que podem ser usadas para executar determinados serviços eletrônicos. Eles podem ser desde simples serviços – como a realização de verificações de números de cartões de crédito, até serviços complexos – como manipulação de aplicações de hipotecas.

O benefício essencial associado à tecnologia de serviços Web é a ampla padronização que permite que eles sejam usados de forma simples em qualquer plataforma de software e hardware. Dessa forma, a integração de aplicações se torna mais fácil, já que todos os envolvidos usam o mesmo padrão.

A seguir é apresentada uma breve descrição dos principais padrões usados para apoio à tecnologia de serviços Web:

- **WSDL – *Web Services Description Language***: linguagem usada para descrever serviços Web. Uma descrição WSDL contém as informações relacionadas ao serviço Web necessárias para sua publicação, descoberta e invocação – incluindo capacidades, interface, comportamento e qualidade de serviço. Cada serviço Web é descrito de forma abstrata por meio de: um tipo de porta (*port type*); operações (*operations*) que o tipo de porta oferece; e a estrutura de mensagens (*message*) de entrada e de saída [7];
- **UDDI – *Universal Description, Discovery, and Integration***: padrão que define a estrutura e o conteúdo dos diretórios que contém descrições de serviços oferecidos. O padrão UDDI permite que provedores de serviços possam registrar seus serviços Web, usando as descrições WSDL, para que clientes de serviços possam descobri-los. Esse padrão permite também que clientes de serviços disponibilizem suas necessidades para que provedores de serviços busquem por usuários que precisam de um determinado serviço [8];
- **SOAP – *Simple Object Access Protocol***: protocolo que define um mecanismo para a comunicação geral entre serviços Web na Internet. Ele define o formato das mensagens (normalmente com conteúdo de dados XML) que são trocadas entre clientes de serviços, provedores de serviços e diretórios de serviços. De uma forma geral, o protocolo SOAP descreve como o protocolo HTTP (*HyperText Transfer*

Protocol) pode ser usado para chamada de procedimento remoto (RPC) na Internet, por meio de uma combinação de um cabeçalho HTTP com um corpo SOAP [9]. Porém, outros protocolos de transporte também podem ser usados.

De acordo com as definições anteriores, de um modo geral, um serviço Web pode ser entendido como uma aplicação que tem uma interface descrita em WSDL, registrada em um diretório de serviços via protocolo UDDI e interage com clientes usando troca de mensagens XML encapsuladas em envelopes SOAP [55].

2.4 Contratos Eletrônicos para Serviços Web

Contratos eletrônicos, de uma forma geral, envolvem o uso de serviços eletrônicos. Um tipo específico de contrato eletrônico é aquele que usa serviços Web. Do ponto de vista conceitual, praticamente não há diferenças entre as informações apresentadas na Seção 2.2, para contratos eletrônicos em geral, e as que se aplicam aos contratos eletrônicos para serviços Web. A diferença está nas tecnologias envolvidas em sua especificação e uso. No caso de contratos eletrônicos para serviços Web, as linguagens de especificação tanto para a especificação de processos de negócio quanto para a especificação de atributos de QoS precisam ser apropriadas para esse tipo de serviço eletrônico.

Ainda não existe uma linguagem padrão oficial para especificação de processos de negócio baseados em serviços Web. Diversas linguagens têm sido propostas; cada uma delas tem diferentes origens e possuem diferentes objetivos para tratar de particularidades específicas de processos de negócio. A maioria das linguagens existentes são extensões ou de linguagens de modelagem de propósito geral ou de linguagens de modelagem de processos de negócio intra-organizacionais.

Exemplos de linguagens de modelagem de processos baseados em serviços Web são:

- **WSFL – *Web Service Flow Language*** (proposta pela IBM) [56];
- **WSCL – *Web Service Conversation Language*** (proposta pela HP) [57];
- **WSCI – *Web Services Choreography Interface*** (proposta pela W3C) [58];
- **BPML – *Business Process Modeling Language*** (proposta pela *Business Process Management Initiative* – BPMI) [59];
- **BPSS – *Business Process Specification Schema*** (proposta pela ebXML) [60];
- **XLANG – *Web Services for Business Process Design*** (proposta pela Microsoft) [61];
- **WS-BPEL – *Web Services Business Process Execution Language*** (proposta pela IBM) [11].

Bernauer [62] apresenta uma comparação de diferentes propostas para linguagem de modelagem de processos de negócio (não restrita a linguagens baseadas em serviços Web), que indica os pontos fracos e os pontos fortes de cada uma. Essa comparação conclui que nenhuma linguagem preenche todos os requisitos identificados para a modelagem de processos de negócio. Apesar disso, a linguagem WS-BPEL tem sido apresentada como a linguagem que deve se tornar padrão para a modelagem de processos de negócio para serviços Web [6].

Situação semelhante acontece com as linguagens de especificação de atributos de QoS, porém com um grau ainda maior de indefinição do que no caso anterior. Três dos exemplos mais comuns atualmente para esse tipo de linguagem são:

- **WSLA - *Web Service Level Agreement*** (proposta pela IBM) [30, 52];
- **WS-Policy – *Web Services Policy Framework*** (proposta por um conjunto de organizações) [14, 15];

- **WS-Agreement** – *Web Services Agreement Specification* (proposta pela *Global Grid Forum* – GGF) [12, 13].

Assim como para linguagens de especificação de processos de negócio, também não há uma linguagem considerada uma linguagem padrão oficial para a especificação de atributos de QoS. Dentre as três linguagens anteriormente citadas, WS-Agreement é considerada como a mais abrangente, quando comparada com outras linguagens atuais como, por exemplo, WS-Policy. WS-Agreement é baseada na linguagem WSLA, uma linguagem considerada bastante consistente, adaptada exclusivamente para a tecnologia de serviços Web [30].

Nesta tese, as linguagens WS-BPEL e WS-Agreement foram escolhidas como base da abordagem proposta para as especificações, respectivamente, de processos de negócio e de atributos de QoS. Elas foram escolhidas por serem ambas bastante abrangentes e de grande disseminação. Essas duas linguagens são brevemente apresentadas nas subseções a seguir. Como essas duas linguagens são baseadas na linguagem de descrição de serviços Web WSDL, ela é também apresentada a seguir, antes das outras duas.

Exemplos destas três linguagens são apresentados nesta seção na forma de códigos, para um melhor entendimento de seus elementos. Exemplos gráficos são apresentados no estudo de caso desta tese.

2.4.1 WSDL

WSDL é uma linguagem projetada para a descrição de serviços Web. Ela possibilita a descrição de informações da interface de um serviço, tais como seus parâmetros de entrada e de saída, de modo que um usuário possa usá-lo sem necessariamente conhecer sua implementação. Essa linguagem é resultado de uma proposta conjunta da IBM e Microsoft. Atualmente, é considerada a linguagem padrão para a descrição de serviços

Web. Os elementos de WSDL são brevemente descritos a seguir. Dentre esses, os principais elementos são os tipos de porta (*port type*), operações (*operations*) e mensagens (*messages*).

- **Type**: define um tipo de dado complexo, baseado em tipos padrão de dados simples, para ser usado em situações específicas;
- **Operation**: define uma ação oferecida por um serviço Web;
- **Port Type**: define um conjunto de operações oferecido por um serviço Web;
- **Message**: define um tipo de dado a ser transmitido pelo sistema, durante a invocação a operações de serviços Web e o retorno deles;
- **Binding**: define um protocolo de comunicação e o formato de dados para as mensagens de um determinado tipo de porta (*port type*);
- **Port**: define um ponto final único, formado por uma combinação de uma ligação (*binding*) e um endereço de rede;
- **Service**: define um conjunto de pontos finais (*port*) relacionados.

No Código 01 é apresentado um exemplo de especificação de serviços Web na linguagem WSDL, em função do exemplo apresentado na Figura 01 (Seção 2.1). Primeiramente são definidos os tipos de mensagens a serem trocadas entre os sistemas, como – por exemplo, a mensagem **solicitaBilhetesReq**, composta por uma única parte – **itinerario**, do tipo *string*. Depois são definidos os tipos de porta, incluindo suas respectivas operações. Um exemplo de tipo de porta é **companhiaAereaPT**, composto por uma única operação – **solicitaBilhetes**, que possui como entrada o tipo de mensagem exemplificado acima **solicitaBilhetesReq**. Um tipo de porta representa um serviço Web composto que pode conter uma ou mais operações, a serem invocadas por outros sistemas.

Um elemento adicional apresentado no Código 01 é o tipo de ligação entre parceiros (*partner link type*). Ele é necessário apenas nas especificações de processo escritas em WS-BPEL, mas não faz parte da especificação original WSDL. Um tipo de ligação entre

parceiros define um par de serviços Web envolvidos na comunicação entre dois parceiros de negócio. O exemplo de tipo de ligação do Código 01 é **compraDeBilhetes**, que define dois papéis **compradorDeBilhetes** e **vendedorDeBilhetes** – cada um deles associado a um tipo de porta definido anteriormente. Os demais elementos WSDL não são apresentados nesse exemplo.

Código 01: Exemplo de especificação WSDL.

```
<wsdl:definitions name="agenciaDeViagens">
  <wsdl:message name="solicitaBilhetesReq">
    <wsdl:part name="itinerario" type="xsd:string"></wsdl:part>
  </wsdl:message>

  <wsdl:message name="enviaBilhetesRes">
    <wsdl:part name="bilhetes" type="xsd:string"></wsdl:part>
  </wsdl:message>

  <wsdl:portType name="companhiaAereaPT">
    <wsdl:operation name="solicitaBilhetes">
      <wsdl:input message="tns:solicitaBilhetesReq"></wsdl:input>
    </wsdl:operation>
  </wsdl:portType>

  <wsdl:portType name="agenciaDeViagensPT">
    <wsdl:operation name="enviaBilhetes">
      <wsdl:output message="tns:enviaBilhetesRes"></wsdl:output>
    </wsdl:operation>
  </wsdl:portType>

  <plnk:partnerLinkType name="compraDeBilhetes">
    <plnk:role name="compradorDeBilhetes">
      <plnk:portType name="tns:agenciaDeViagensPT" />
    </plnk:role>

    <plnk:role name="vendedorDeBilhetes">
      <plnk:portType name="tns:companhiaAereaPT" />
    </plnk:role>
  </plnk:partnerLinkType>
</wsdl:definitions>
```

2.4.2 WS-BPEL

WS-BPEL, anteriormente chamada BPEL4WS (*Business Process Execution Language for Web Services*), é uma linguagem projetada para a especificação formal tanto de processos de negócio como de protocolos de interação de negócio, usando serviços Web. Ela permite a composição de serviços Web com o objetivo de realizar funcionalidades mais complexas. A linguagem WS-BPEL combina e substitui as especificações WSFL (da IBM) [56] e a XLANG (da Microsoft) [61].

Existem alguns elementos fundamentais da linguagem WS-BPEL que podem ser usados tanto para a especificação de processos de negócio quanto para a especificação de protocolos de interação de negócio. Além desses conceitos, existem algumas extensões requeridas para cada um desses dois usos específicos. A estrutura geral de um processo de negócio especificado em WS-BPEL é formada por quatro seções: *PartnerLinks*, *Variables*, *FaultHandlers* e *Activities*, conforme descritas a seguir:

- ***PartnerLinks***: seção que define os diferentes parceiros (partes envolvidas) que interagem com o processo de negócio durante toda sua execução. Eles são usados para identificar a funcionalidade que deve ser oferecida por cada serviço parceiro. As ligações de parceiros (*partner link*) devem estar associadas a um tipo de ligação entre parceiros (*partner link type*) definido na especificação de serviços Web em WSDL;
- ***Variables***: seção que define as variáveis de dados usadas pelo processo de negócio. As definições são feitas em termos de tipos de mensagem WSDL, elementos ou tipos simples de esquemas XML. Elas são usadas para manter os dados de estado e o histórico do processo com base nas mensagens trocadas. As variáveis devem estar associadas a tipos de mensagens (*messages*) definidos na especificação WSDL;

- ***FaultHandlers***: seção que contém os tratadores de falhas que definem as atividades a ser executadas em resposta às falhas resultantes da invocação de serviços de avaliação e de aprovação;
- ***Activities***: seção que contém a descrição do comportamento normal para a execução do processo de negócio. Existem basicamente dois tipos de atividades:
 - ***Basic Activity***: tipo de atividade usado para executar alguma operação. Algumas dessas atividades básicas envolvem a interação com algum parceiro, as quais são: *invoke*, *receive* e *reply*. Outras dessas atividades básicas são executadas sem a interação com qualquer parceiro, as quais são: *wait*, *terminate*, *assign*, *empty*, *throw* e *compensate*;
 - ***Structured Activity***: tipo de atividade usado para agrupar atividades básicas (*basic activities*) dentro de algumas estruturas de fluxo. Tais atividades são: *while*, *pick*, *flow*, *sequence*, *switch* e *scope*.

Processos de negócio especificados em WS-BPEL definem a troca de mensagens entre serviços Web, especificados em WSDL. O próprio processo de negócio é considerado um parceiro de negócio – o parceiro de negócio principal, que interage com os outros parceiros.

No Código 02 é apresentado um exemplo de especificação WS-BPEL para um processo de negócio, em função do exemplo apresentado na Figura 01 (Seção 2.1). O processo apresentado envolve uma agência de viagens, considerada como o parceiro de negócio principal, e uma companhia de aviação – como um fornecedor de serviços eletrônicos. Na seção <partnerLinks>, o parceiro **aviacao** é definido. Na seção <variables>, são definidas as variáveis **itinerario** e **bilhetes**. A seção <faultHandlers> não é detalhada. Na seção de atividades, algumas atividades básicas (*receive*, *invoke* e

receive) são definidas dentro da atividade estruturada `flow` (que define atividades concorrentes). A atividade básica `invoke` é destinada a invocar o parceiro **aviacao**, para chamar sua operação (de um serviço Web) **solicitaBilhetes** e enviar-lhe o conteúdo da variável **itinerario**.

Código 02: Exemplo de especificação WS-BPEL.

```
<bpel:Process name="SolicitaçãoDeBilhetes">
  <partnerLinks>
    <partnerLink name="aviacao" partnerLinkType="compraDeBilhetes" (...)/>
  </partnerLinks>

  <variables>
    <variable name="itinerario" messageType="solicitaBilhetesReq"/>
    <variable name="bilhetes" messageType="enviaBilhetesRes"/>
  </variables>

  <faultHandler> (...)</faultHandler>

  <flow>
    <invoke partnerLink="aviacao"
      portType="CompanhiaAereaPT"
      operation="solicitaBilhetes"
      inputVariable="itinerario"
      (...)>
    </invoke>

    <receive partnerLink="aviacao" (...)> (...)</receive>
  </flow>
</bpel:Process>
```

Os símbolos “(...)” representam partes da especificação que foram suprimidos para diminuir o tamanho do exemplo. O mesmo se aplica aos demais códigos desta tese.

2.4.3 WS-Agreement

WS-Agreement é uma linguagem projetada para a especificação formal das capacidades de provedores de serviços Web (em particular aqueles relacionados a atributos de QoS). Além

disso, WS-Agreement também inclui um protocolo para a criação e monitoramento de acordos baseados nessas capacidades. Embora o nome dessa linguagem de especificação seja bastante genérico, o termo *Agreement*⁶ não se refere aqui a contratos eletrônicos de uma forma geral. Apenas uma parte comumente existente em contratos eletrônicos é coberta por essa linguagem: a especificação das capacidades não funcionais (como, por exemplo, atributos e níveis de QoS) associadas a serviços Web já definidos em algum outro lugar. Assim, não é definido por essa linguagem, por exemplo, o processo de negócio envolvendo os serviços Web cobertos por uma especificação WS-Agreement. Dependendo das necessidades, pode ser que uma especificação WS-Agreement seja o suficiente para representar todo o contrato eletrônico entre organizações, mas normalmente ele precisa ser usado com uma visão complementar do acordo entre elas.

A linguagem WS-Agreement é uma recomendação oferecida pelo grupo de trabalho GRAAP-WG (*Grid Resource Allocation Agreement Protocol Working Group*) do fórum GGF (*Global Grid Forum*) que não alcançou a situação de padrão ainda [12]. WS-Agreement é baseada na linguagem WSLA [30], que era considerada a mais precisa e detalhada das linguagens usadas na prática e foi descontinuada desde que foi substituída pela WS-Agreement.

Em relação à especificação de atributos de QoS, um acordo é criado com base principalmente em Objetivos de Níveis de Serviço (SLO⁷), que são condições que devem ser atingidas para que uma determinada necessidade pré-definida seja satisfeita. Um SLO representa o aspecto de QoS do acordo, tanto como requisitos do cliente quanto como garantias do fornecedor de serviços Web. Por exemplo, um acordo pode oferecer garantias sobre os limites (mínimo e/ou máximo) de tempo de resposta e de disponibilidade de serviço.

Um documento WS-Agreement consiste de três seções principais: *Name*, *Context* e *Terms*, conforme brevemente descritas a seguir:

⁶ Do inglês: acordo

⁷ Do inglês: Service Level Objectives

- **Name:** seção opcional que inclui um nome para o acordo WS-Agreement;
- **Context:** seção que descreve as partes envolvidas no acordo WS-Agreement e alguns meta-dados relacionados com o acordo, como, por exemplo, seu tempo de vida;
- **Terms:** seção que define o conteúdo do acordo WS-Agreement propriamente dito. Contém as seguintes três subseções:
 - **Service Description Terms (SDTs):** subseção que inclui um conjunto de Termos de Descrição de Serviços (SDT), que descrevem os serviços Web que serão fornecidos em função do acordo WS-Agreement, para os quais pode haver definições de garantias em relação a atributos de QoS. Os SDTs podem ser uma definição completa de um serviço Web ou então apenas uma referência a um serviço Web definido em outro lugar. Um acordo WS-Agreement pode conter qualquer quantidade de SDTs;
 - **Service Properties (SPs):** subseção que define um conjunto de propriedades mensuráveis e expostas, em termos de variáveis, associadas a operações de serviços Web, para as quais serão definidas garantias de QoS;
 - **Guarantee Terms (GTs):** subseção que define um conjunto de garantias de QoS (relacionadas tanto aos compromissos quanto às penalidades) associadas com os serviços Web descritos pelos STDs. Os GTs definem seu escopo, condições de qualificação, SLOs e listas de valores associados aos SLOs. Um acordo WS-Agreement pode conter qualquer quantidade de GTs.

No Código 03 é apresentado um exemplo da seção `<wsag:Terms>` de uma especificação WS-Agreement, em função do exemplo apresentado na Figura 01 (Seção 2.1). A primeira parte (`<wsag:ServiceDescription Term>`) define o serviço Web **solicitaBilhetes** que é um serviço a ser oferecido como parte do acordo WS-Agreement. A segunda parte (`<wsag:ServiceProperties>`) define **tempoDeRespostaEm Segundos** como uma propriedade mensurável e exposta, associada ao serviço Web **solicitaBilhetes**. A última parte (`<wsag:GuaranteeTerm>`) define um atributo de QoS a ser garantido durante a realização do acordo. A garantia de QoS é definida por um SLO que estabelece que o nível de tempo de resposta para o serviço Web relacionado deve ser de no máximo 60 segundos.

Código 03: Exemplo de especificação WS-Agreement.

```
<wsag:Terms>
  (...)
  <wsag:ServiceDescriptionTerm wsag:Name="reqBilhetesSDT"
    wsag:ServiceName="solicitaBilhetes">
  </wsag:ServiceDescriptionTerm>
  (...)
  <wsag:ServiceProperties wsag:Name="reqBilhetesProperties"
    wsag:ServiceName="solicitaBilhetes">
    <wsag:VariableSet>
      <wsag:Variable wsag:Name="tempoDeRespostaEmSegundos"
        wsag:Metric="time:duration">
        <wsag:Location> (...) </wsag:Location>
      </wsag:Variable>
    </wsag:VariableSet>
  </wsag:ServiceProperties>
  (...)
  <wsag:GuaranteeTerm Obligated="ServiceProvider">
    <wsag:ServiceScope ServiceName="solicitaBilhetes">
    </wsag:ServiceScope>
    <wsag:QualifyingCondition> (...) </wsag:QualifyingCondition>
    <wsag:ServiceLevelObjective>
      tempoDeRespostaEmSegundos IS-LESS-INCLUSIVE 60
    </wsag:ServiceLevelObjective>
    <wsag:BusinessValueList> (...) </wsag:BusinessValueList>
  </wsag:GuaranteeTerm>
  (...)
</wsag:Terms>
```

2.5 Considerações Finais

Neste capítulo foi apresentada uma visão geral sobre a área de contratos eletrônicos, inicialmente de um ponto de vista genérico e depois de um ponto de vista específico de contratos eletrônicos para serviços Web. A realização de processos de negócio interorganizacionais com base em contratos eletrônicos é apresentada como um conjunto de atividades bastante amplo e complexo.

Levando essa complexidade em consideração, Chiu et. al. [20, 63] compara o ciclo de vida de contratos eletrônicos ao ciclo de vida de sistemas de software, concluindo que eles possuem algumas características similares que podem ser usadas em benefício dos contratos eletrônicos. Entretanto, existem algumas propriedades que os diferenciam bastante, tais como: a natureza mais dinâmica de software que podem ser alterados com mais frequência do que contratos eletrônicos – depois de fechados entre os interessados, e a maior complexidade do desenvolvimento de software em relação ao estabelecimento de contratos eletrônicos. Apesar das diferenças existentes, contratos eletrônicos podem ser tratados por meio de atividades similares às de desenvolvimento de software, enquanto que a execução do contrato eletrônico pode ser considerada equivalente à execução do software depois de instalado ou implantando. Essa comparação pode facilitar o entendimento de um contrato eletrônico a partir de seus fundamentos até sua implementação. Com base nessas considerações é que nesta tese é proposta uma abordagem baseada em linha de produto de software (LP), associada a modelos de características também para software, para ser aplicada no estabelecimento de contratos eletrônicos para serviços Web.

Com o objetivo de oferecer um embasamento conceitual para a apresentação da abordagem proposta nesta tese, no próximo capítulo são apresentados os modelos de características – dentro do contexto de LP.

Capítulo 3

Linha de Produto de Software e Modelos de Características

Modelos de características são usados na engenharia de software para capturar e gerenciar pontos comuns e variabilidades em famílias de produtos de software. Características representam itens funcionais ou não funcionais de um produto que os usuários e clientes consideram importante na descrição e distinção de membros de uma família de produtos [41]. Esses modelos são amplamente usados no contexto de linha de produto de software (LP) – uma abordagem da engenharia de software que promove a geração de produtos de software específicos a partir de uma família de produtos.

A abordagem para o estabelecimento de contratos eletrônicos para serviços Web proposta nesta tese é inspirada no processo de desenvolvimento de LP. Em particular, usam-se modelos de características para representar aspectos comuns e variáveis dos contratos eletrônicos. Neste capítulo é apresentada uma visão geral tanto de modelos de características quanto da abordagem de LP, com ênfase nos primeiros.

Primeiramente, são apresentados os conceitos relacionados à abordagem de LP de uma forma geral. Em seguida, são apresentados os conceitos relacionados a modelos de características. É também apresentado o método de LP usado como base para a definição do processo contido na abordagem proposta nesta tese, método esse que usa modelos de características. Dentro do contexto de modelos de características, é apresentado o meta-

modelo de características baseado em cardinalidade, por se tratar do meta-modelo usado na abordagem proposta nesta tese.

3.1 Linha de Produto de Software

Linha de produto de software (LP) é uma abordagem que visa à construção sistemática de software baseada em uma família de produtos por meio do reuso de artefatos [34]. Uma família de produto de software é o conjunto de produtos de software com propriedades suficientemente similares para permitir a definição de uma infra-estrutura comum dos itens que os compõem e a parametrização das diferenças entre eles. Por exemplo, uma família de produtos para sistemas de gerência de fluxo de trabalho⁸ [64] tem uma arquitetura de software comum e um conjunto de componentes que povoam essa arquitetura, tais como escalonadores de tarefas, gerenciadores de recursos, interpretadores e gerenciadores de arquiteturas de fluxo de trabalho. Cada um desses componentes pode variar de um produto para outro, por exemplo, um produto pode requerer determinados tipos de recursos (exemplo: humano, equipamentos e salas). Cada produto pode ter uma política específica de escalonamento de tarefas (exemplo: prioridade de tarefas e classes de tarefas).

A abordagem de LP tem por objetivo identificar os aspectos comuns e as diferenças existentes entre os artefatos de software, ao longo do processo de desenvolvimento, de modo a explicitar os pontos de decisão em que a adaptação dos componentes para a geração de produtos específicos pode ser realizada. Para tal, durante o processo de desenvolvimento devem-se identificar as variabilidades dos produtos. Uma das formas de identificar e representar essas variabilidades é com o uso de modelos de características, descrito na próxima seção.

Variabilidades são diferenças tangíveis entre produtos que podem ser reveladas e distribuídas entre os artefatos da LP, sejam eles a arquitetura, os componentes, as

⁸ Do inglês: workflow

interfaces entre componentes ou as conexões entre componentes. As variações podem ser reveladas em qualquer etapa do desenvolvimento de uma LP, a começar da análise de requisitos. A variação mais conhecida em construção de sistemas é a existência de várias implementações para uma mesma especificação. Por exemplo, para uma especificação “classificar itens” que diz que itens entram desordenados e saem ordenados, pode-se implementar vários algoritmos como *Bubble sort*, *Insertion sort* ou *Quicksort*. A expressão de uma variação pode ser obtida pela introdução de parâmetros instanciáveis em tempo de construção associados a componentes, subsistemas ou coleção de subsistemas a partir dos quais um produto pode ser configurado atribuindo-se um conjunto de valores a esses parâmetros [35]. Um tipo de variação simples de ser representado é a escolha de componentes diferentes para uma mesma arquitetura. Assim, produtos podem ter maior ou menor capacidade, ou características diferentes dependendo do tipo de componente escolhido para a arquitetura [34].

Em geral, pode-se dizer que a abordagem de LP consiste de três atividades básicas, descritas a seguir [35]:

- **Engenharia de domínio:** atividades de desenvolvimento do núcleo de artefatos da LP. O artefato considerado mais importante é a arquitetura da LP, que representa a sua infra-estrutura central. Essa arquitetura é composta de componentes que são unidades de software independentes. Essas unidades encapsulam seu projeto e implementação e oferecem interfaces bem definidas para o meio externo. Os artefatos da LP são elaborados de modo a identificar quais são os pontos comuns e as variabilidades dos produtos representados por ela;
- **Engenharia de aplicação:** atividades de desenvolvimento de produtos específicos, por meio da criação de uma instância para a arquitetura da LP. Os pontos comuns dos artefatos são mantidos, enquanto que as variabilidades são resolvidas – em função dos requisitos do produto. Dependendo das decisões tomadas durante a

criação da instância para a arquitetura da LP, é selecionada a configuração de um produto específico;

- **Gerência da LP:** engloba as atividades relacionadas à construção e manutenção da LP, realizadas com base em planos que definem as estratégias a serem seguidas pela organização.

O processo de desenvolvimento de uma LP pode ser visto como dois modelos de ciclo de vida – um para a engenharia de domínio e outro para a engenharia de aplicação, conforme apresentado na Figura 04. A gerência da LP não é representada nessa figura.

Nas próximas subseções, são apresentados mais detalhes a respeito das três atividades básicas do processo de LP, previamente citadas.

3.1.1 Engenharia de Domínio

Na Figura 04, o primeiro modelo de ciclo de vida (Engenharia de Domínio) representa o desenvolvimento do núcleo de artefatos, que engloba a análise de domínio, o desenvolvimento da arquitetura e o desenvolvimento de componentes. Esse ciclo produz um modelo de domínio, a arquitetura da LP e um conjunto de componentes reusáveis.

Durante o desenvolvimento do núcleo de artefatos, na engenharia de domínio, três aspectos devem ser considerados: a definição do contexto da LP, o núcleo de artefatos e o plano de produção [34]. Um artefato é considerado um item de software reusável usado como bloco de construção de uma LP. Pode ser uma arquitetura de software, um componente, um *framework*, ou alguma documentação relativa à arquitetura de software ou a algum componente.

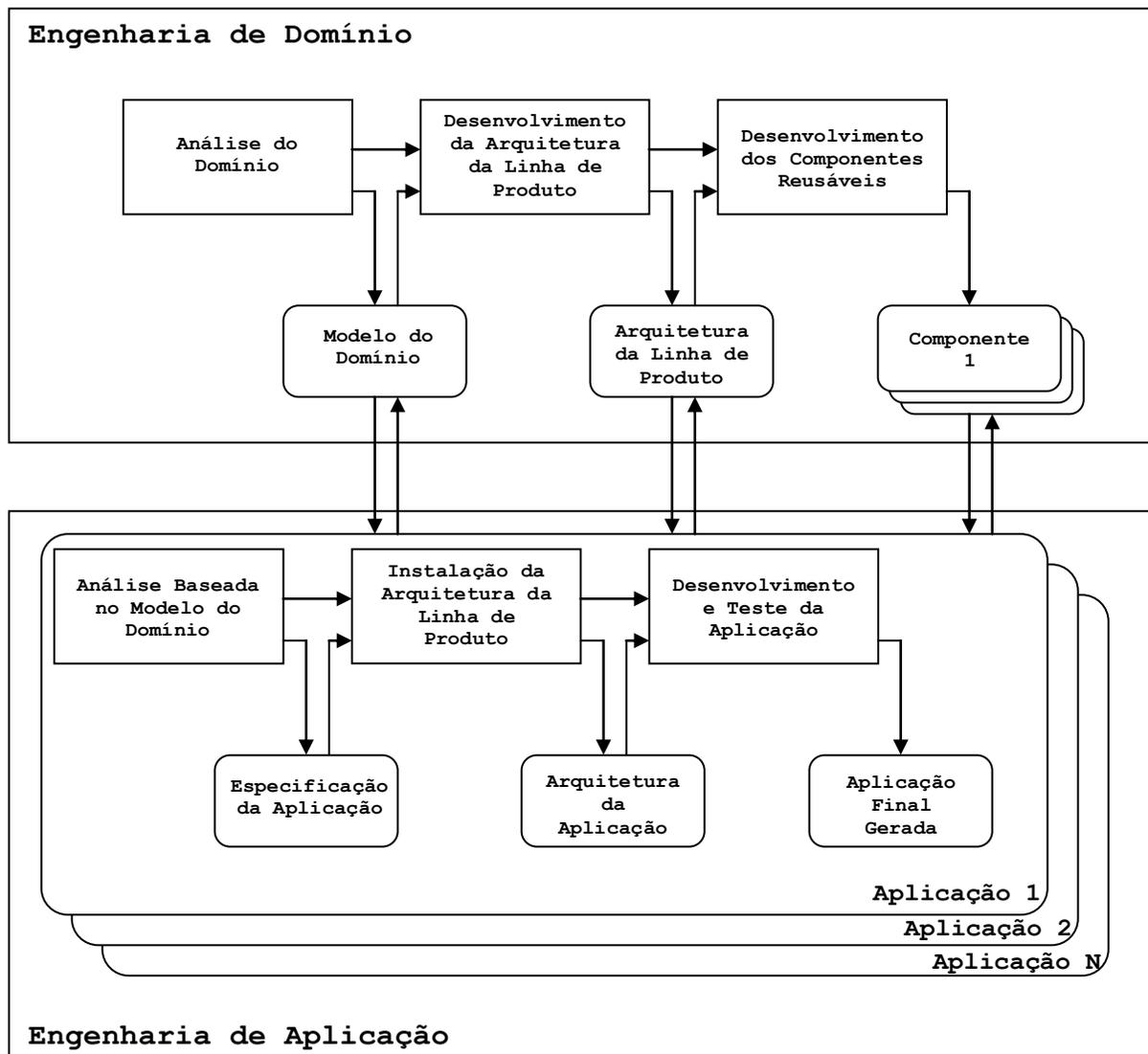


Figura 04: Processo de desenvolvimento de uma LP [34].

Arquitetura da linha de produto de software

A arquitetura da LP é um dos artefatos mais importantes. Ela representa a infra-estrutura central da LP. A arquitetura de um sistema de software engloba a definição de suas estruturas gerais, descrevendo os elementos que compõem os sistemas e as interações entre esses [36]. Além de descrever esses elementos e suas interações, uma arquitetura de software apóia também questões importantes de projeto, tais como: a organização do

sistema como composição de componentes, as estruturas de controle globais, os protocolos de comunicação, a composição dos elementos do projeto e a designação da funcionalidade dos componentes do projeto.

Uma arquitetura representa um investimento significativo para uma organização. Em função disso, é natural que as organizações desejem maximizar esse investimento por meio do reuso da arquitetura em vários sistemas. Quando uma organização está produzindo vários sistemas similares, reusando a mesma arquitetura e os componentes associados a ela, pode reduzir seus custos de produção e o tempo necessário para disponibilizar um produto no mercado.

A arquitetura de uma LP representa um meta-projeto das aplicações que constituem a família de produtos. Alguns requisitos importantes da arquitetura de LP são [35]:

- Ser estável, persistindo durante a vida da LP, sofrendo poucas alterações;
- Permitir integração flexível de novas funcionalidades durante o ciclo de vida da arquitetura;
- Permitir a representação explícita de variações de modo a maximizar o reuso;
- Fornecer mecanismos para implementar variações;
- Permitir a representação de variações inerentes da família de aplicação;
- Permitir a representação de variações impostas pelo mercado.

O projeto da arquitetura de software tem sido objeto de muitas técnicas recentes, que devem ser consideradas no projeto de uma arquitetura de LP, em particular dos métodos de desenvolvimento baseado em componentes [36]. Outros trabalhos importantes são: a padronização de arquiteturas de software com UML da OMG [54] e as investigações dedicadas a articular o relacionamento entre especificação de requisitos e o projeto da arquitetura de software [34].

Componentes da linha de produto de software

Um componente pode ser definido como uma unidade de software independente, que encapsula dentro de si seu projeto e implementação, e oferece interfaces bem definidas para o meio externo. Componentes têm pontos de interconexão chamados de Interfaces que concentram um conjunto de serviços relacionados [34].

As interfaces são chamadas de Interfaces Fornecidas (*Provided Interfaces*) ou Interfaces Requisitadas (*Required Interfaces*). A interface fornecida define as operações que o componente oferece a outros componentes. A interface requisitada define as operações que o componente requisitará de outros componentes. A interface requisitada de um componente conecta-se à interface fornecida pelo componente que oferece os serviços correspondentes.

Um componente é entendido por seus usuários por meio de sua especificação. A especificação de um componente deve estabelecer suas interfaces, argumentos, pré e pós-condições de cada serviço e invariantes. A conexão de componentes é realizada por protocolos de comunicação que vão desde simples chamadas de procedimentos até infra-estruturas de comunicação como DCOM [65] e CORBA [66].

3.1.2 Engenharia de Aplicação

O segundo modelo de ciclo de vida da Figura 04 (Engenharia de Aplicação) representa o desenvolvimento dos produtos, o qual engloba análise baseada no modelo do domínio, criação de uma instância para a arquitetura da LP e desenvolvimento e teste da aplicação. Esse ciclo produz, para cada produto sendo desenvolvido, uma especificação da aplicação, uma arquitetura da aplicação e uma aplicação final.

Durante a primeira atividade – a análise baseada no modelo do domínio, esse modelo é usado para extrair informações sobre a aplicação específica a ser desenvolvida e definir uma lista de características para a aplicação. Assim, o modelo do domínio representa as

necessidades do cliente em termos da arquitetura de LP e dos componentes disponíveis no repositório. Associado ao modelo de domínio, o modelo de decisão define os principais aspectos a serem considerados no desenvolvimento do produto. Esse modelo permite ao desenvolvedor da aplicação, por exemplo, identificar os ajustes necessários nos valores iniciais das características descritas pelos componentes tendo em vista as restrições arquiteturais impostas pelas variabilidades.

As necessidades não cobertas pelo modelo do domínio são denominadas novos requisitos. Os novos requisitos podem implicar na necessidade de estender a arquitetura da LP e adquirir novos componentes. Para os requisitos não satisfeitos pelo modelo existente, deve ser investigado o custo de desenvolver o software por encomenda, modificar a arquitetura ou os componentes existentes. A lista de características da aplicação recuperada do modelo de domínio mais aquelas definidas nos novos requisitos são integradas para formar a especificação do produto. Informações sobre os requisitos não antecipadas no modelo de domínio devem ser realimentadas para estender o modelo de domínio e/ou o repositório de componentes existentes.

A segunda atividade é a criação de uma instância para a arquitetura do produto. A estrutura da arquitetura é percorrida de cima para baixo, tomando-se decisões recursivamente até que a arquitetura genérica seja transformada na arquitetura específica do produto. O processo de construção da arquitetura é iterativo. A instância inicial provavelmente contém pendências como pontos de variação e decisões ainda não resolvidos. Essas pendências, geralmente, ocorrem devido aos requisitos específicos da aplicação. A implementação desses requisitos pode provocar modificações na estrutura da arquitetura da LP ou apenas afetar a instância da arquitetura da aplicação. A decisão depende da perspectiva de uso do requisito por outros clientes.

A terceira atividade da engenharia de aplicação refere-se ao povoamento da arquitetura com os componentes adequados. As técnicas usadas nesse processo são similares às aquelas usadas para o desenvolvimento baseado em componentes. Como nas atividades anteriores,

atenção especial deve ser dada à implementação dos requisitos específicos da aplicação. Os componentes podem ter várias origens, tais como:

- O repositório da LP pode conter o componente desejado;
- O requisito do cliente é implementado na própria aplicação;
- Um componente de prateleira (COTS) é adquirido;
- Um novo componente é implementado, encomendado ou adquirido de terceiros, pois os requisitos do usuário não foram satisfeitos pelos componentes existentes no repositório;
- Um componente é obtido de aplicações legadas.

Os testes da aplicação são feitos também de forma similar às técnicas de desenvolvimento baseado em componentes que incluem: teste de unidade e verificação de componentes, teste de integração e verificação geral das funcionalidades da aplicação.

3.1.3 Gerência da Linha de Produto de Software

Grande parte do sucesso de uma LP depende do compromisso da organização em construí-la e mantê-la. Assim, é importante que a organização estabeleça um plano de adoção da LP que descreva o estado desejado e as estratégias para atingir esse estado [35]. Para tal é necessário prover, à equipe responsável pela LP, os recursos e garantir que as atividades da LP sejam coordenadas e supervisionadas. Deve haver um sincronismo entre o grupo que desenvolve os artefatos e o grupo que gera os produtos. Deve também ser garantido o apoio ao desenvolvimento e à evolução dos artefatos da linha.

3.2 Modelos de Características

Modelos de características têm sido usados para capturar e gerenciar pontos comuns e variabilidades em LP [31]. Eles foram originalmente propostos no contexto de engenharia de domínio, como parte do método FODA (*Feature-Oriented Domain Analysis*) [32]. Desde então, eles têm sido aplicados em uma variedade de domínios incluindo sistemas de telecomunicações, bibliotecas de moldes, protocolos de rede e sistemas embarcados [33].

Um modelo de características é uma representação de particularidades relevantes de alguma entidade de interesse. Uma característica pode ser definida como uma propriedade de um sistema que é relevante para alguma entidade envolvida em seu desenvolvimento ou uso. Ela é usada para capturar pontos comuns ou estabelecer uma discriminação entre sistemas em uma família de sistemas. As características podem denotar qualquer propriedade funcional ou não funcional nos níveis de requisitos, de arquitetura, de componentes ou de plataformas computacionais [33]. De acordo com o método FODA original, características podem ser obrigatórias, opcionais ou alternativas.

Modelos de características são normalmente organizados em diagramas hierárquicos, na forma de árvore, em que cada nó representa uma característica e cada característica pode ser descrita por um conjunto de sub-características representadas como nós descendentes [38]. Modelos de características oferecem uma notação simples e intuitiva para representar pontos de variação independentemente de mecanismos de implementação tais como herança ou agregação.

Várias extensões e variantes têm sido propostas para as convenções de características do método FODA original, principalmente para se adequar a demandas da abordagem de LP [33, 38–41]. Por exemplo, várias notações têm sido desenvolvidas para distinguir entre características obrigatórias, opcionais e alternativas [38]. Outros exemplos de extensões estão relacionados a grupos de características, cardinalidade de características e de grupos de características, atributos, relacionamentos e modularização [33]. Algumas facilidades também foram propostas para tornar possível expressar restrições globais em relação às

possíveis combinações de características [38]. Normalmente, cada abordagem é definida por um meta-modelo, a partir do qual modelos de características podem ser definidos apropriadamente.

Na Figura 05 é apresentado um exemplo de modelo de características. A notação usada é a proposta por Czarnecki et. al. [33], cujo meta-modelo é apresentado na Seção 3.2.1. O modelo de características apresentado nessa figura representa os serviços oferecidos por uma loja eletrônica virtual. Dois serviços básicos são oferecidos: **pagamento** e **entrega**. O **pagamento** pode ser feito por **cartão de crédito**, **cartão de débito** ou **boleto bancário**, e um serviço de **deteção de fraude** pode ser opcionalmente usado. A entrega pode ser por **terra**, **ar**, **mar** ou qualquer combinação dessas opções.

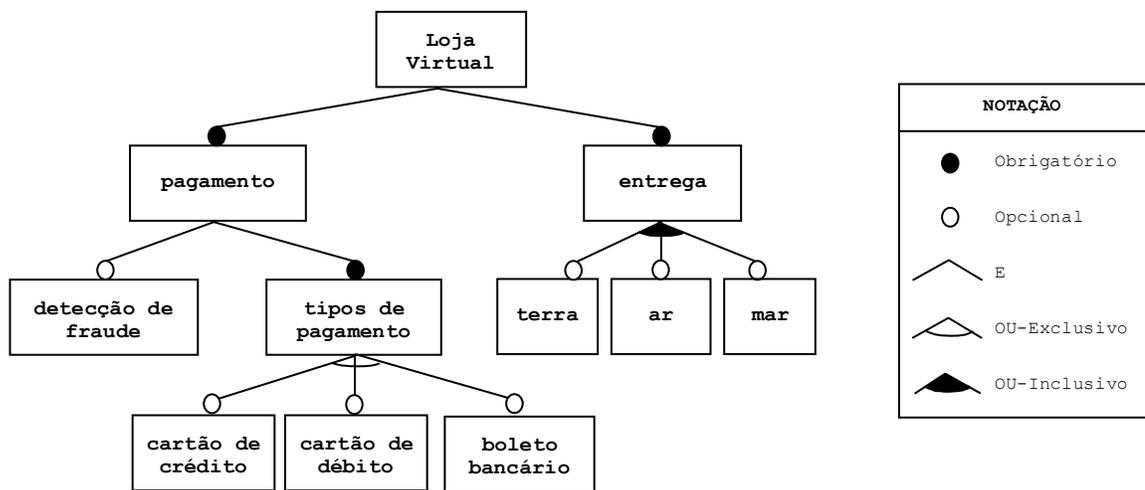


Figura 05: Exemplo de modelo de características [67].

Um modelo de características pode descrever o espaço de configuração de uma família de sistemas. Um membro da família pode ser especificado por meio da seleção das características desejadas a partir do modelo de características, dentro das restrições de variabilidade definidas pelo modelo (por exemplo, a escolha de exatamente uma característica a partir de um conjunto de características alternativas). Um exemplo de

configuração para o modelo de características apresentado na Figura 05 é: escolher o método de pagamento por cartão de crédito, sem o mecanismo de detecção de fraude, e escolher o método de entrega por meio de uma combinação de terra e mar. Esse processo é chamado de configuração de características [33].

Modelos de características possuem algumas propriedades oferecidas por ontologias computacionais [42]. Ontologias são frequentemente usadas na ciência da computação para capturar o conhecimento sobre algum domínio de interesse, possibilitando seu compartilhamento e seu reuso [43, 44] – por exemplo, em Web semântica [68]. Portanto, ambas as abordagens são usadas para representar conceitos em um domínio particular e relacionamentos entre eles. Entretanto, linguagens e ferramentas de ontologia normalmente oferecem facilidades de raciocínio para a verificação de consistência e completude, e máquinas de inferência que permitem o processamento de regras, o que não são oferecidas por modelos de características. Por outro lado, modelos de características oferecem facilidades para capturar e gerenciar conceitos comuns e variáveis, o que não é oferecido por ontologias. Apesar destas diferenças, cada uma delas poderia ser estendida para incorporar facilidades oferecidas pela outra.

Devido ao seu alto nível de abstração, a aplicação de modelos de características pode ser útil para outros objetivos além de, puramente, a abordagem de LP, tais como: encapsulamento de requisitos de sistema, apoio à tomada de decisão de mercados, planejamento futuro e comunicação entre os envolvidos no desenvolvimento de sistemas [39].

Algumas ferramentas de desenvolvimento de sistemas têm incorporado conceitos de modelos de características, o que facilita seu uso na indústria de software. Por exemplo, a ferramenta IBM Rational RequisitePro [69] oferece apoio para a modelagem de características, embora ainda primitiva – não permitindo representações gráficas. Outro exemplo de ferramenta é a Telelogic DOORS (*Dynamic Object Oriented Requirements System*) [70], que possuem algumas extensões que oferecem apoio também ainda limitado à modelagem de características.

O meta-modelo de características usado nesta tese é o proposto por Czarnecki et. al. [33], apresentado na próxima seção. Suas principais vantagens são: integração de outras extensões propostas anteriormente e balanço entre simplicidade e completude conceitual.

3.2.1 Meta-Modelo de Características Baseado em Cardinalidade

O meta-modelo de características baseado em cardinalidade, proposto por Czarnecki et. al. [33], envolve os conceitos de atributos, grupos de características, modularização de modelo e cardinalidade. O meta-modelo é apresentado como um diagrama de classes na Figura 06.

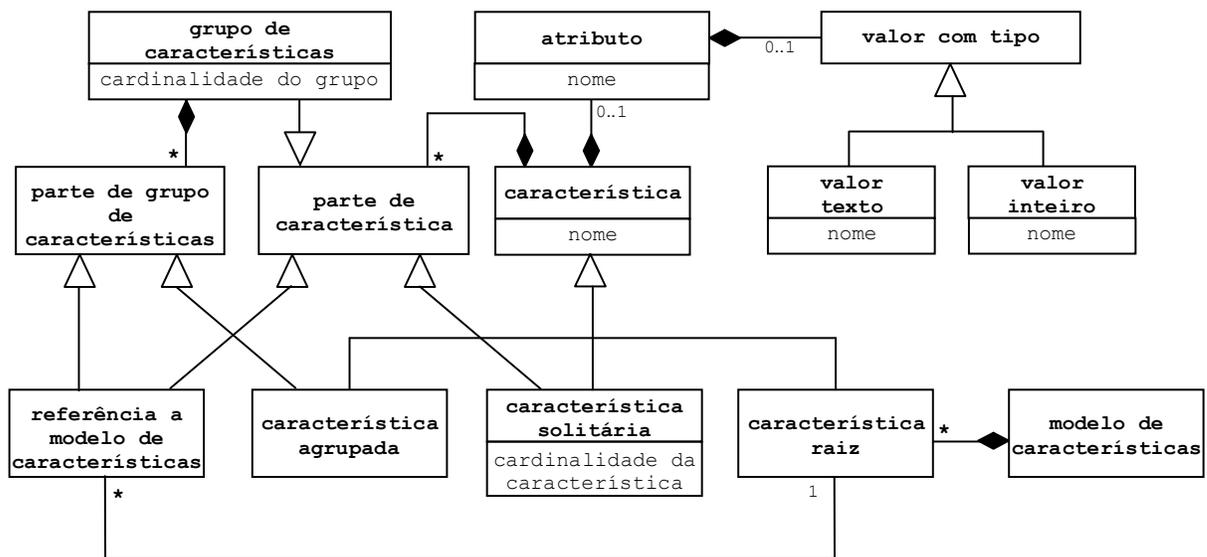


Figura 06: Meta-modelo de características baseado em cardinalidade [33].

Existem três tipos de características em um modelo de características:

- (i) Característica raiz: que forma a raiz do modelo de características;

- (ii) Característica agrupada: que pode ocorrer apenas em um grupo de características;
- (iii) Característica solitária: que, por definição, não é uma característica raiz e nem está agrupada em um grupo de características. Muitas características em um modelo de características típico são características solitárias.

Características podem ter um atributo com um valor com tipo (do tipo texto ou do tipo inteiro). As classes abstratas parte de grupo de características e parte de característica relacionam aqueles tipos de objeto que podem ser contidos por um grupo de características e por uma característica, respectivamente. Um grupo de características contém características agrupadas ou referências a outros modelos de características, enquanto que uma característica pode incluir características solitárias, grupos de características e referências a outros modelos. A modularização de modelos é obtida por meio do uso da classe referência a modelo de características. Ela pode se referir a uma única característica raiz, mas uma característica raiz pode ser referenciada por várias referências a modelo de características.

Cardinalidade de características e cardinalidade de grupos de características são representadas como atributos no meta-modelo de características. A cardinalidade da característica define quantas vezes uma sub-característica solitária (e possíveis sub-árvores abaixo dela) podem ser copiadas como uma filha de sua característica pai. Uma cardinalidade de característica é uma seqüência de intervalos da forma $[n1..n1'] [n1..n1']$. Por exemplo: $[0..3][6..6]$ requer que a característica associada seja tomada 0, 1, 2, 3 ou 6 vezes; e $[5..*]$ requer que a característica associada seja tomada 5 ou qualquer número maior do que 5 vezes. Similarmente, a cardinalidade do grupo é uma propriedade do relacionamento entre uma característica pai e um conjunto de sub-características. Um grupo de características expressa uma escolha sobre as características no grupo. Essa escolha é restringida pela cardinalidade do grupo $\{n-n'\}$, que especifica que se deve tomar pelo menos n e no máximo n' características agrupadas diferentes do grupo de

características. Por exemplo: [1..3] requer que pelo menos 1 característica agrupada e no máximo 3 características agrupadas sejam selecionadas para um determinado grupo de características.

3.2.2 Restrições Adicionais de Características

As restrições de características compõem uma parte importante dos modelos de características. Por meio de restrições de características é possível representar as regras que devem ser satisfeitas durante a configuração de um modelo de características específico. O meta-modelo de características baseado em cardinalidade já apresenta intrinsecamente um conjunto de restrições de características simples – aquelas relacionadas a tipos, cardinalidade e grupos de características e tipos de valores de atributo. Essas restrições são apenas restrições locais, portanto estão sempre relacionadas a uma única característica ou a um grupo de características.

Além desse tipo de restrição de características, a abordagem de Czarnecki inclui o uso de restrições adicionais mais complexas que não são cobertas pelo meta-modelo de características [67]. Essas restrições são definidas por meio de expressões formadas por lógica, aritmética, teoria dos conjuntos e operadores de texto⁹ sobre atributos de características e conjuntos de características. Para isso, a linguagem XPath [71] é usada. XPath é uma linguagem relacionada à linguagem XML, que oferece mecanismos flexíveis para referenciar (selecionar ou filtrar) dados contidos em diferentes partes de um documento XML. Ela usa uma notação de caminho (tal como URLs) para navegar através da estrutura hierárquica de documentos XML.

Essas restrições adicionais expressas por meio da linguagem XPath podem ser tanto locais quanto globais. Algumas restrições adicionais locais são relacionadas a restrições de valor de atributo (além de seu tipo, já oferecido pelo meta-modelo de características), tal

⁹ Do inglês: string.

como uma possível escala de valores (como, por exemplo, de 10 a 20). Outras restrições adicionais mais importantes são as restrições de implicação e restrições de exclusão, que podem ser tanto locais quanto globais. Essas restrições podem ser necessárias para declarar, por exemplo, que a escolha de uma característica requer obrigatoriamente a seleção (ou a não seleção) de uma outra determinada característica ou conjunto de características.

A partir do modelo de características apresentado na Figura 05, os seguintes exemplos de restrições adicionais poderiam ser definidos:

- **Restrição de implicação:** a seleção do serviço **detecção de fraude** apenas faz sentido se um dos seguintes métodos de pagamento for selecionado: **cartão de crédito** ou **cartão de débito**. A expressão XPath para essa restrição é: **if (//detecção de fraude) then (//cartão de crédito union //cartão de débito) else true();**
- **Restrição de exclusão:** a seleção do método de entrega por **ar** exclui obrigatoriamente a possibilidade de escolha, ao mesmo tempo, do método de entrega por **mar**. A expressão XPath para essa restrição é: **if (//mar) then not (//ar) else true().**

3.2.3 Apoio Automatizado a Modelos de Características

Czarnecki et. al. [33] desenvolveram uma ferramenta de apoio automatizado para modelos de características chamada FeaturePlugin [67, 72]. Ela oferece apoio à elaboração de modelos de características e suas configurações de acordo com as regras do meta-modelo de características apresentado anteriormente. FeaturePlugin oferece uma interface gráfica com o usuário e possibilita a exportação de modelos de características e de suas configurações como documentos XML. A interface gráfica permite que os modelos de

características sejam editados e visualizados como árvores hierárquicas, similares a visualização oferecida por sistemas de gerenciamento de arquivos, com pastas, subpastas e arquivos. A ferramenta não oferece edição e visualização no formato de grafos, como o apresentado na Figura 05, embora esteja planejado para futuras versões.

Na Figura 07 é apresentado o modelo de características, da Figura 05, elaborado com o uso da ferramenta FeaturePlugin. Os símbolos possuem os seguintes significados, de acordo com o meta-modelo de características baseado em cardinalidade, apresentado anteriormente:

- ▲ – característica raiz;
- – característica solitária obrigatória;
- – característica solitária opcional;
- ▲ – grupo de características (ou-inclusivo);
- ▲ – grupo de características (ou-exclusivo);
- – característica agrupada obrigatória;
- – característica agrupada opcional/alternativa;
- --> – referência para característica.

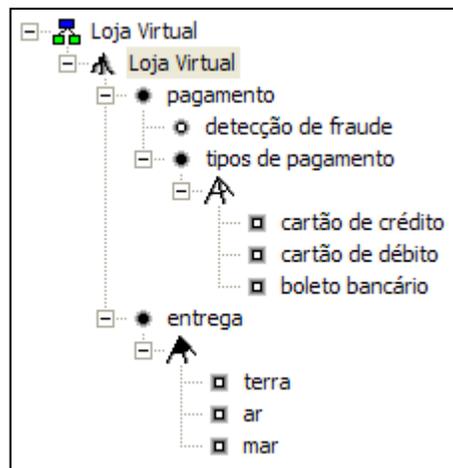


Figura 07: Exemplo de modelo de características – ferramenta FeaturePlugin.

Na Figura 08 é apresentado um exemplo de uma configuração do modelo de características da Figura 07. Os seguintes símbolos são usados em uma configuração de modelo de características (além dos usados em um modelo de características):

- característica opcional/alternativa não selecionada;
- característica opcional/alternativa selecionada;
- característica opcional/alternativa que não pode ser selecionada devido a restrições de cardinalidade.

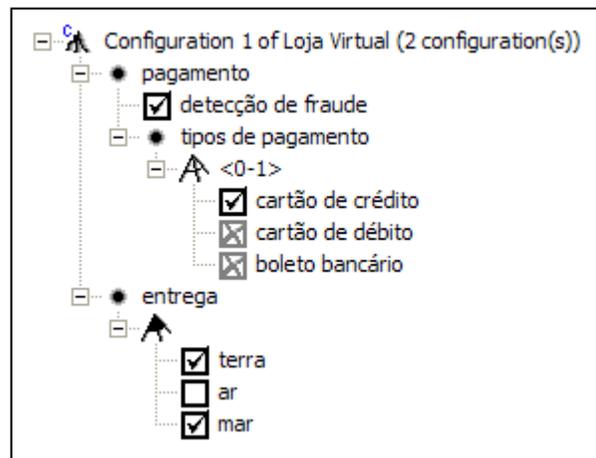


Figura 08: Exemplo de configuração de modelo de características – FeaturePlugin.

A ferramenta também oferece mecanismos para a avaliação de restrições adicionais expressas em XPath, apresentada na seção anterior. Por meio desse mecanismo, um determinado modelo de características configurado pode ter as restrições adicionais verificadas – como o objetivo de checar se elas são satisfeitas ou não.

A ferramenta FeaturePlugin é apresentada de forma mais detalhada no Capítulo 5, por ser parte do conjunto de ferramentas FeatureContract desenvolvido para automatizar a abordagem proposta nesta tese.

3.2.4 Método FORM

Um exemplo de método de LP que usa modelos de características é o método FORM (*A Feature-Oriented Reuse Method with Domain-Specific Reference Architectures*) [37]. Esse método é uma extensão do método FODA, o método precursor no uso de modelos de características para a análise de domínio de aplicação, citado no início da seção 3.2. O método FORM é o usado nesta tese como base para a definição do processo de estabelecimento de contratos.

Sendo um método de LP, as principais características do método FORM estão alinhadas com aquelas da Seção 3.1 deste capítulo, apresentadas do ponto de vista genérico. A diferença é que nesse caso é evidente o uso dos modelos de características como forma de capturar e gerenciar os pontos comuns e as variabilidades entre os produtos de software de um mesmo domínio.

A Figura 09 apresenta o processo associado ao método FORM. Esse processo é bastante parecido com o processo genérico para LP apresentado na Figura 04. Assim como no caso anterior, o processo apresenta dois modelos de ciclo de vida – um para a engenharia de domínio e outro para a engenharia de aplicação. Os tipos de atividades desempenhadas e os artefatos manipulados também são bastante parecidos. A principal diferença é que: enquanto no processo genérico o modelo principal de requisitos dos pontos comuns e das variabilidades do domínio é tratado genericamente como “modelo de domínio”, no processo FORM os modelos de características são usados especificamente para esse fim.

O processo de estabelecimento de contratos eletrônicos desta tese foi elaborado de modo a possuir atividades similares às existentes no método FORM, depois de realizadas as alterações necessárias. Assim, ao invés de informações relativas a software, são usadas informações relativas especificamente a contratos eletrônicos; por exemplo, a arquitetura de referência é substituída pelo molde de contrato, e os componentes reusáveis são substituídos pelos serviços Web. A contribuição principal do método FORM está mais

relacionada à fonte de inspiração para a definição da abordagem, por ter sido o primeiro método de LP avaliado dentro do contexto deste trabalho. Além da abordagem geral, seus detalhes mais específicos estão relacionadas aos modelos de características.

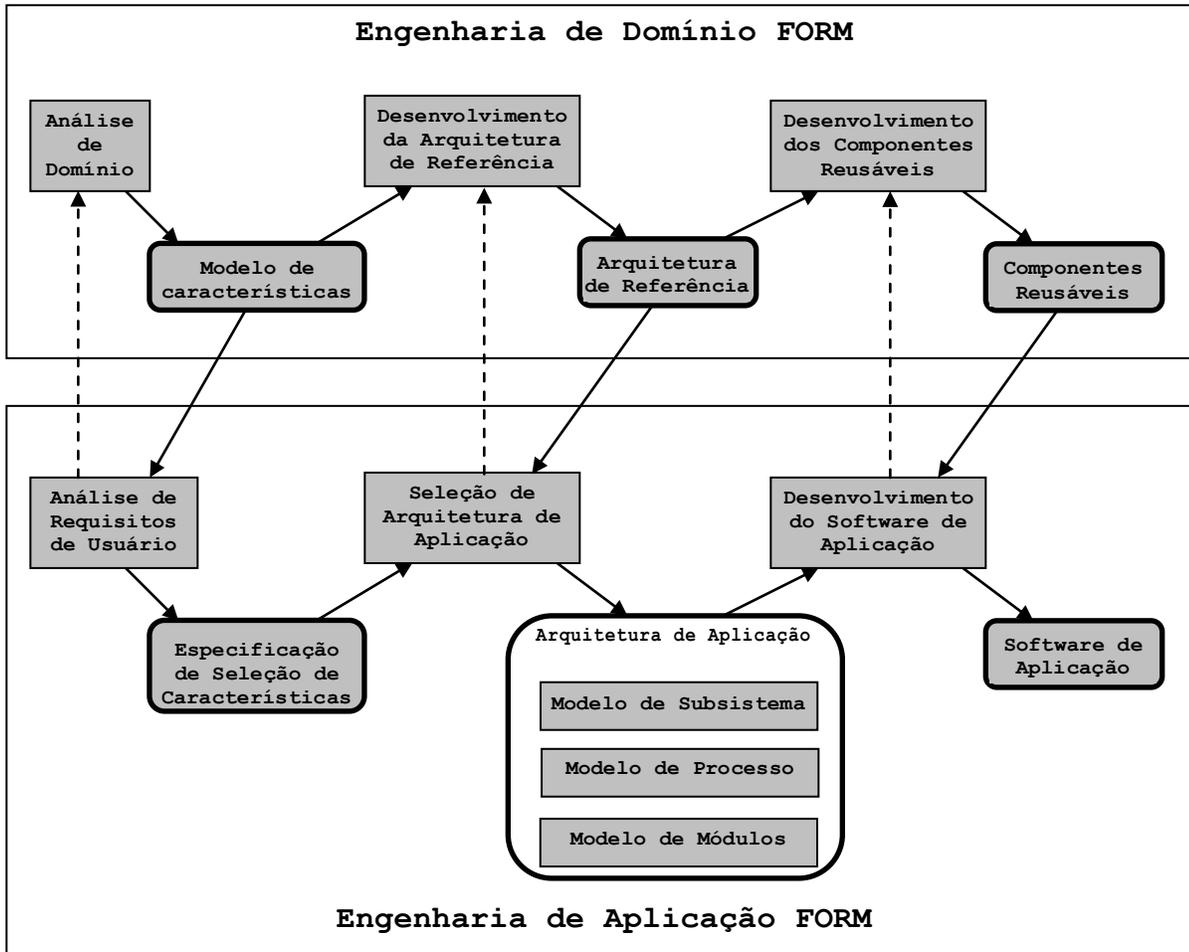


Figura 09: Processo de desenvolvimento do método FORM [37].

3.3 Considerações Finais

Neste capítulo foi apresentada uma visão geral sobre a abordagem de LP e sobre modelos de características. Ambas são amplamente usadas na definição da abordagem proposta

nesta tese para o estabelecimento de contratos eletrônicos para serviços Web. LP contribui como a inspiração conceitual para a abordagem proposta, enquanto que modelos de características contribuem para permitir a melhor gerência das variabilidades existentes entre possíveis e diferentes contratos eletrônicos a serem estabelecidos.

Embora a abordagem de LP tenha sido concebida especificamente para o processo de desenvolvimento de software, assim como modelos de características, a abordagem proposta nesta tese baseia-se na hipótese de que elas possam ser aplicadas para outras áreas similares ao desenvolvimento de software. O estabelecimento de contratos eletrônicos é um dos possíveis casos similares ao desenvolvimento de software que também pode se beneficiar da abordagem de LP. Ambas as áreas possuem necessidades semelhantes, tais como uma melhor compreensão de domínio e uma redução de tempo de disponibilização de produtos ao mercado – apesar de algumas diferenças significativas entre elas (conforme apresentado no Capítulo 2).

Em uma comparação entre o processo de desenvolvimento de software seguindo a abordagem de LP com o processo de estabelecimento de contratos eletrônicos para serviços Web, alguns mapeamentos entre os diferentes artefatos podem ser estabelecidos. Por exemplo, o molde de contrato eletrônico pode ser comparado com a arquitetura da LP, assim como os serviços Web que vão compor o contrato eletrônico final podem ser comparados com os componentes de software que vão compor a aplicação de software final. Da mesma forma que componentes de software devem ser selecionados para preencher a arquitetura da LP, formando a arquitetura da aplicação final; os serviços Web devem ser selecionados para compor o contrato eletrônico resultante de seu processo de estabelecimento.

Uma abordagem similar, porém para outra área de aplicação, foi proposta recentemente por Czarnecki & Antkiewicz [73]. De forma similar à proposta nesta tese, eles apresentam uma abordagem baseada em moldes para o mapeamento de modelos de características para representações concisas de variabilidade em diferentes tipos de outros modelos. Essa abordagem usa modelos de características para parametrizar qualquer tipo de modelo de

uma forma geral; enquanto que a abordagem apresentada nesta tese usa modelos de características para parametrizar particularmente contratos eletrônicos especificados usando as linguagens WS-BPEL e WS-Agreement. Contudo, a abordagem citada foi aplicada por enquanto apenas para diagramas de atividades e de classe da UML 2.0. Objetivando um alvo mais particular – o estabelecimento de contratos eletrônicos, a abordagem desta tese pode tratar alguns tipos de problema e soluções não tratadas nesse outro trabalho.

No próximo capítulo é apresentada então a abordagem para o estabelecimento de contratos eletrônicos para serviços Web, orientado pelo uso de moldes de contratos. Abordagem essa inspirada em conceitos de LP, com base principalmente no método FORM e na aplicação de modelos de características.

Capítulo 4

Abordagem para o Estabelecimento de Contratos Eletrônicos para Serviços Web

O estabelecimento de contratos eletrônicos para serviços Web pode ser desencorajador devido à complexidade envolvida nessa atividade. Visando superar inconvenientes existentes, algumas abordagens têm sido propostas para melhorar a estruturação e o reuso de informação nesse contexto [17, 20–30]. Essas abordagens são, normalmente, baseadas no uso de moldes de contratos eletrônicos. Embora elas tenham contribuído com uma série de avanços nas áreas de elaboração e de realização de contratos eletrônicos, oferecem um potencial limitado para o reuso de informação entre contratos eletrônicos similares.

Nesta tese, é proposta uma nova abordagem para o estabelecimento de contratos eletrônicos para serviços Web cuja maior contribuição é oferecer uma forma sistemática e eficiente para a estruturação e o reuso de informação. Tal abordagem possibilita esses avanços por usar conceitos de linha de produto de software e, principalmente, de modelos de características.

O conteúdo deste capítulo descreve a abordagem para o estabelecimento de contratos eletrônicos para serviços Web usando modelos de características. Primeiramente, são apresentadas algumas premissas para o uso da abordagem proposta. Depois é apresentada

uma visão geral da abordagem na forma de um processo para o estabelecimento de contratos eletrônicos. Em seguida, cada uma das etapas do processo é discutida de forma mais detalhada, com base em alguns exemplos de artefatos produzidos. Finalmente, os principais trabalhos da área de contratos eletrônicos relacionados à abordagem proposta são discutidos.

Desse ponto em diante, contrato eletrônico para serviços Web é referenciado, na maioria das vezes, apenas como contrato eletrônico – para fins de facilidade de leitura do texto.

4.1 Premissas da Abordagem

Existem diferentes abordagens possíveis para o estabelecimento de contratos eletrônicos, com diferentes objetivos e níveis de complexidade. Levando isso em consideração, as seguintes suposições foram seguidas para a concepção da abordagem proposta:

- **Quantidade de organizações envolvidas:** a abordagem é inicialmente desenvolvida para o estabelecimento de contratos eletrônicos entre apenas duas organizações. Esse é o cenário mais comum para o tipo de cooperação de negócio alvo da abordagem proposta. Apesar disso, dependendo do caso, acordos entre mais do que duas partes podem ser compostos por um agrupamento de vários acordos entre duas partes, por meio da aplicação dessa mesma abordagem – com os devidos cuidados e considerações;
- **Abordagem estática X abordagem dinâmica:** a abordagem é baseada em um modelo estático de negociação e estabelecimento do contrato eletrônico. Nesse modelo, em contraste com um modelo dinâmico, as organizações parceiras são conhecidas logo no início do processo de estabelecimento de contrato. Além disso,

os serviços Web a serem contratados são definidos durante o processo e já fixados nas instâncias de contrato eletrônico resultantes.

4.2 Visão Geral da Abordagem

A abordagem para o estabelecimento de contratos eletrônicos consiste principalmente de um processo de cinco estágios. Esses estágios são derivados do método FORM (*Feature-Oriented Reuse Method*) [37], apresentado no Capítulo 3 (Figura 09, Seção 3.2.4). Na abordagem proposta aqui, os modelos de características são usados para a representação genérica de serviços eletrônicos e níveis de atributos de QoS. Essa representação permite a derivação de um molde de contrato eletrônico.

As atividades do processo de estabelecimento de contratos eletrônicos, incluindo a negociação existente entre as partes envolvidas, são orientadas direta ou indiretamente pelos modelos de características e pelas possíveis configurações desses modelos, conforme ilustra a Figura 10. Cada configuração de um modelo de características apresenta o conjunto de serviços eletrônicos e informações associadas aos serviços, juntamente com o conjunto de níveis dos atributos de QoS, que devem todos ser acordados entre as partes envolvidas. Os serviços eletrônicos genéricos que forem selecionados para contratação são mapeados para serviços Web específicos que possuem suas operações, por meio de um relacionamento de um-para-um. Esses serviços Web são por fim referenciados nos contratos eletrônicos estabelecidos.

Nenhum protocolo é definido, como parte desta abordagem, especificamente para a negociação entre as organizações. Considera-se que elas devem entrar em acordos para o estabelecimento dos contratos, o que deve ser feito com o apoio das configurações de modelos de características. Entretanto, não há a definição de papéis e as responsabilidades que cada organização envolvida deve seguir. Apenas o processo global de estabelecimento de contratos eletrônicos é tratado, em que a negociação é uma dos sub-processos.

O processo proposto é apresentado graficamente na Figura 10. Assim como o processo de LP usado como base, o processo proposto aqui também pode ser visto como dois modelos de ciclo de vida – um para o desenvolvimento do molde e outro para o desenvolvimento do contrato. O termo “Engenharia”, usado no método FORM para cada um dos ciclos de vida, foi substituído pelo termo “Desenvolvimento”, visto que a área de estabelecimento de contratos eletrônicos pode ser considerada ainda em uma fase inicial de maturidade em termos de técnicas e métodos do que uma engenharia propriamente dita. Nas subseções seguintes são descritos os estágios, os artefatos e os papéis envolvidos nesse processo.

4.2.1 Estágios do Processo

Os cinco estágios parte do processo proposto são descritos a seguir, agrupados nos dois ciclos de vida apresentados na Figura 10:

- I) **Desenvolvimento do molde:** atividades relacionadas ao desenvolvimento de novos moldes de contrato ou a atualização de contratos pré-existentes:
 1. Elaboração dos modelos de características para serviços eletrônicos: dois modelos de características são elaborados para representar os serviços eletrônicos e os atributos de qualidade de serviço (QoS), uma para cada uma das organizações que desejam estabelecer o contrato eletrônico;
 2. Criação do molde de contrato eletrônico para serviços Web: tendo os dois modelos de características para serviços eletrônicos como base, um molde de contrato eletrônico é criado. Ele conterá informações necessárias que podem ser usadas em qualquer contrato eletrônico estabelecido a partir desses dois modelos de características;

3. Desenvolvimento e publicação dos serviços Web: serviços Web que implementam os serviços eletrônicos contratados devem ser desenvolvidos e publicados para estarem disponíveis durante a realização do processo de negócio contido no contrato eletrônico.

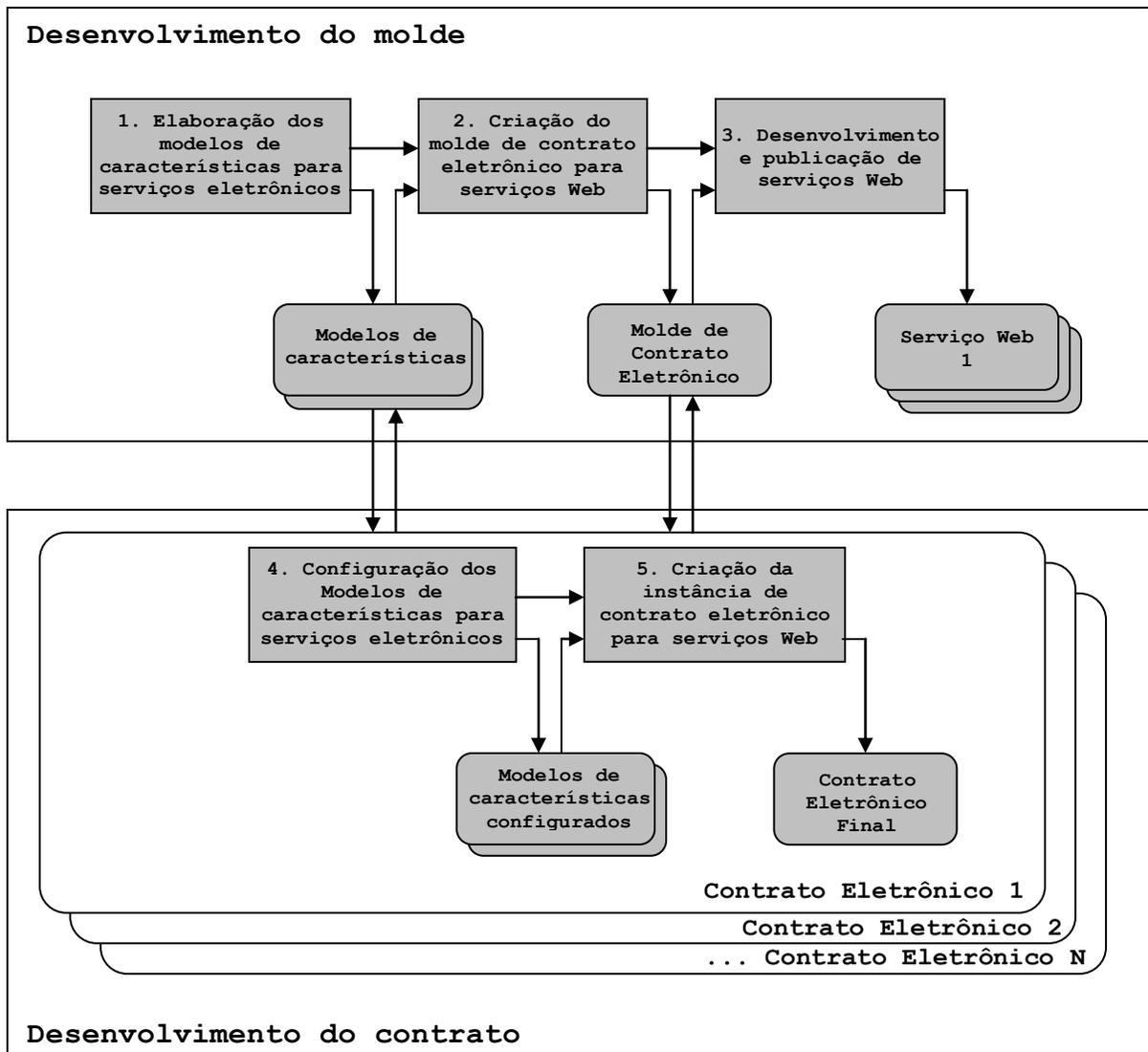


Figura 10: Processo de estabelecimento de contrato eletrônico com base em características.

II) **Desenvolvimento do contrato:** atividades relacionadas à criação de novas instâncias de contratos baseados no molde de contrato:

4. Configuração dos modelos de características para serviços eletrônicos: os dois modelos de características para serviços eletrônicos são então configurados para representar os serviços eletrônicos e níveis dos atributos de QoS para um determinado processo de negócio entre duas organizações;
5. Criação da instância de contrato eletrônico para serviços Web: um contrato eletrônico é produzido por meio do refinamento do molde de contrato eletrônico, com base no par de modelos de características para serviços eletrônicos configurados no Estágio 4.

Para a aplicação do processo proposto, existem dois cenários identificados para um determinado domínio de contrato eletrônico:

- (i) O mesmo par de organizações pode estabelecer variações de contratos eletrônicos similares;
- (ii) Uma organização, que já estabeleceu contratos eletrônicos com uma determinada organização anteriormente, pode estabelecer novos contratos eletrônicos similares com outra organização, usando pelo menos uma parte dos artefatos já usados nos contratos eletrônicos anteriores;

O processo de estabelecimento de um contrato inicia assumindo que as duas organizações envolvidas no processo de negócio são conhecidas. Todos os cinco estágios, exceto possivelmente o Estágio 1, são realizados já com uma determinada parceria de negócio como objetivo. Os estágios 1, 2 e 3 são inteiramente realizados apenas uma vez

para um domínio de contrato eletrônico envolvendo o mesmo par de organizações. Já os estágios 4 e 5 são inteiramente realizados para cada nova instância de contrato eletrônico para um mesmo domínio de contrato eletrônico. Durante a realização dos estágios 4 e 5, pode ser necessária a re-execução dos estágios 1, 2 ou 3 para gerenciar informações adicionais não tratadas previamente e que se tornaram necessárias posteriormente. Essas novas iterações dos estágios 1, 2 ou 3 ocorrem quando uma das organizações originalmente envolvida está para estabelecer uma nova instância de contrato eletrônico, para o mesmo domínio, porém com outra organização. Desse modo, ela precisa reusar, pelo menos parcialmente, os artefatos já produzidos.

Apesar de esta abordagem ter sido desenvolvida especificamente para os cenários descritos anteriormente, em que as duas organizações envolvidas no processo de negócio são conhecidas inicialmente, ela pode ser usada em um modelo de negócios do tipo “software de prateleira”. Uma organização independente, especializada na criação de contratos eletrônicos, pode elaborar os artefatos previstos nos Estágio 1 e Estágio 2 para que estes sejam comercializados. Estes artefatos seriam elaborados de forma genérica para determinados domínios de contrato eletrônico. Estando estes artefatos disponíveis, qualquer par de organizações relacionada a algum dos domínios cobertos poderiam então adquiri-los e usá-los na execução dos Estágio 4 e Estágio 5, criando uma instância específica de contrato. A discussão mais ampla desta aplicação da abordagem proposta está fora do escopo desta tese.

4.2.2 Artefatos Produzidos

A Figura 11 representa, como um diagrama de classes, os artefatos produzidos por meio da realização dos cinco estágios descritos anteriormente, bem como o relacionamento entre tais artefatos.

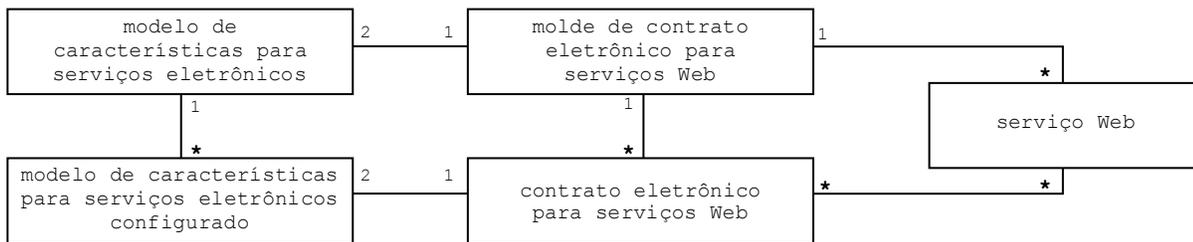


Figura 11: Relacionamento entre os artefatos produzidos pelo processo.

O modelo de características para serviços eletrônicos é o artefato base a partir do qual um único molde de contrato eletrônico para serviços Web é criado e uma ou mais modelos de características para serviços eletrônicos configurados são derivadas. Para cada modelo de características configurado, um contrato eletrônico para serviços Web particular é estabelecido. Os contratos eletrônicos são estabelecidos com base no mesmo molde de contrato eletrônicos para serviços Web. Cada serviço Web que implementa um determinado serviço eletrônico do modelo de características é referenciado por um molde de contrato eletrônico. Somente os serviços Web que implementam os serviços eletrônicos do modelo de características configurado são referenciados pelo contrato eletrônico correspondente.

4.2.3 Papéis Envolvidos

Esta abordagem é proposta para ser usada por diferentes tipos de grupos de cada organização envolvida, incluindo tanto grupos de negócio quanto grupos de desenvolvimento. Para grupos de negócio, os modelos de características para serviços eletrônicos são úteis durante o processo de negociação para o estabelecimento do contrato devido a seu alto nível de abstração. A visão estruturada dos serviços eletrônicos oferecidos pelos modelos de características melhora o entendimento de cada parte.

Em um baixo nível de abstração, os modelos de características e suas configurações serão úteis para os arquitetos de integração de sistemas durante o refinamento dos detalhes dos serviços eletrônicos a serem desenvolvidos. Além da integração de sistemas, provedores de serviços Web podem fazer uso da visão estruturada dos serviços eletrônicos para melhorar o planejamento, a análise e o projeto dos serviços Web contratados antes que o desenvolvimento seja realizado.

Em relação ao papel de cada organização envolvida no processo, elas podem participar de uma forma individualizada ou colaborativa, tanto em uma responsabilidade principal quanto em uma responsabilidade de apoio. A Tabela 01 apresenta um resumo do papel de cada organização no processo de estabelecimento de contrato, levando em consideração que cada uma delas normalmente atua principalmente como um fornecedor ou como um consumidor de serviços Web.

Tabela 01: Papéis das organizações envolvidas.

Estágio do Processo	Tipo de Participação	Responsabilidade da Organização	
		Consumidor	Fornecedor
1. Elaboração dos modelos de características	Individual	Apoio	Principal
2. Criação do molde de contrato eletrônico	Colaborativa	Principal	Apoio
3. Desenvolvimento/publicação de serviços Web	Individual	Apoio	Principal
4. Configuração dos modelos de características	Colaborativa	Principal	Apoio
5. Criação da instância de contrato eletrônico	Colaborativa	Principal	Principal

Para os Estágio 1 e Estágio 3, cada uma das duas organizações deve trabalhar de uma forma individualizada para oferecer os artefatos necessários. Em ambos os casos, a organização fornecedora precisa ter uma atuação mais forte do que a organização

consumidora. Para o Estágio 2, o Estágio 4 e o Estágio 5, ambas organizações devem trabalhar de uma forma colaborativa para oferecer os respectivos artefatos. Nesses três casos, a organização consumidora precisa ter a atuação mais forte do que a organização fornecedora.

A seguir, detalhes dos estágios do processo proposto são apresentados. O Estágio 3 não está incluído já que o desenvolvimento e a publicação de serviços Web estão fora do contexto da abordagem proposta nesta tese. Para um melhor entendimento de cada um desses estágios, são usados como exemplos partes de artefatos obtidos com a realização do estudo de caso apresentado no Capítulo 6.

4.3 Elaboração dos Modelos de Características para Serviços Eletrônicos

A elaboração dos modelos de características para serviços eletrônicos, denominada Estágio 1, é a primeira atividade para as organizações interessadas em realizar um determinado processo de negócio interorganizacional. Durante o Estágio 1, cada organização deve oferecer um modelo de características que represente as informações sobre os serviços a serem disponibilizados por cada uma delas. Esse modelo de características deve estar de acordo com o meta-modelo de características baseado em cardinalidade – apresentado na Seção 3.2.1.

O principal tipo de informação representado nesses modelos de características são os serviços eletrônicos oferecidos pelas organizações. Alguns desses serviços são obrigatórios para o processo de negócio a ser definido enquanto que outros são opcionais ou alternativos. Os modelos de características representam serviços eletrônicos apenas por meio de seu nome e de sua descrição, sem qualquer detalhe técnico, assim como ocorre com artefatos de requisitos para a engenharia de produto de software. Serviços Web específicos devem ser desenvolvidos e publicados por cada organização, como previsto no

Estágio 3. Durante a criação da instância de contrato eletrônico, os serviços eletrônicos representados em alto nível nos modelos de características são mapeados para os serviços Web desenvolvidos especificamente para implementá-los.

Além dos serviços eletrônicos, atributos de qualidade de serviço (QoS) também podem ser representados em um modelo de características. Para os atributos de QoS, além das propriedades comuns para as características, duas propriedades são acrescentadas – a métrica (por exemplo: “tempo” e “volume”/“quantidade”) e a unidade (por exemplo: a unidade “segundos” para a métrica “tempo”) usada em sua medida. Diferentes atributos de QoS podem ser definidos para representar informações relacionadas a: disponibilidade, integridade, confiabilidade, segurança, desempenho e tempo de resposta de serviços eletrônicos – dentre possíveis outros. Cada serviço eletrônico pode ter, associado a ele, um nível definido para algum atributo de QoS que deve ser garantido pelo provedor durante a realização do contrato eletrônico. Diferentes níveis para diversos atributos de QoS podem estar associados a cada serviço eletrônico.

Como exemplo, na Figura 12 é apresentada parte de um modelo de características para serviços eletrônicos. O modelo apresentado descreve as informações sobre serviços eletrônicos e atributos de QoS apresentadas por um sistema de Cobrança de uma organização que presta serviços de forma automatizada para outras organizações. O significado dos símbolos contidos nesse modelo está descrito na Seção 3.2.3. A parte superior do modelo apresenta informações sobre os serviços eletrônicos oferecidos, enquanto que a parte inferior apresenta informações sobre os atributos e níveis de QoS que podem ser associados a tais serviços eletrônicos.

Dentre os serviços eletrônicos disponibilizados pelo sistema de Cobrança encontram-se serviços relacionados a: aplicação, consulta, cancelamento e atualização de ações de cobrança e de reversões de ação de cobrança; além de outros serviços, não detalhados, relacionados a cheques irregulares, dívidas, encargos e descontos, e cadastros de cliente. Em relação aos atributos e níveis de QoS, são apresentados atributos relacionados a: segurança, disponibilidade, tempo de espera, acessos simultâneos, desempenho e tempo de

resposta. Para este último, alguns possíveis níveis de QoS são apresentados. Esses elementos são mais bem explorados como exemplos nas próximas seções deste capítulo. Além disso, esse modelo de características é apresentado com mais detalhes no capítulo em que o estudo de caso do qual ele faz parte é descrito.

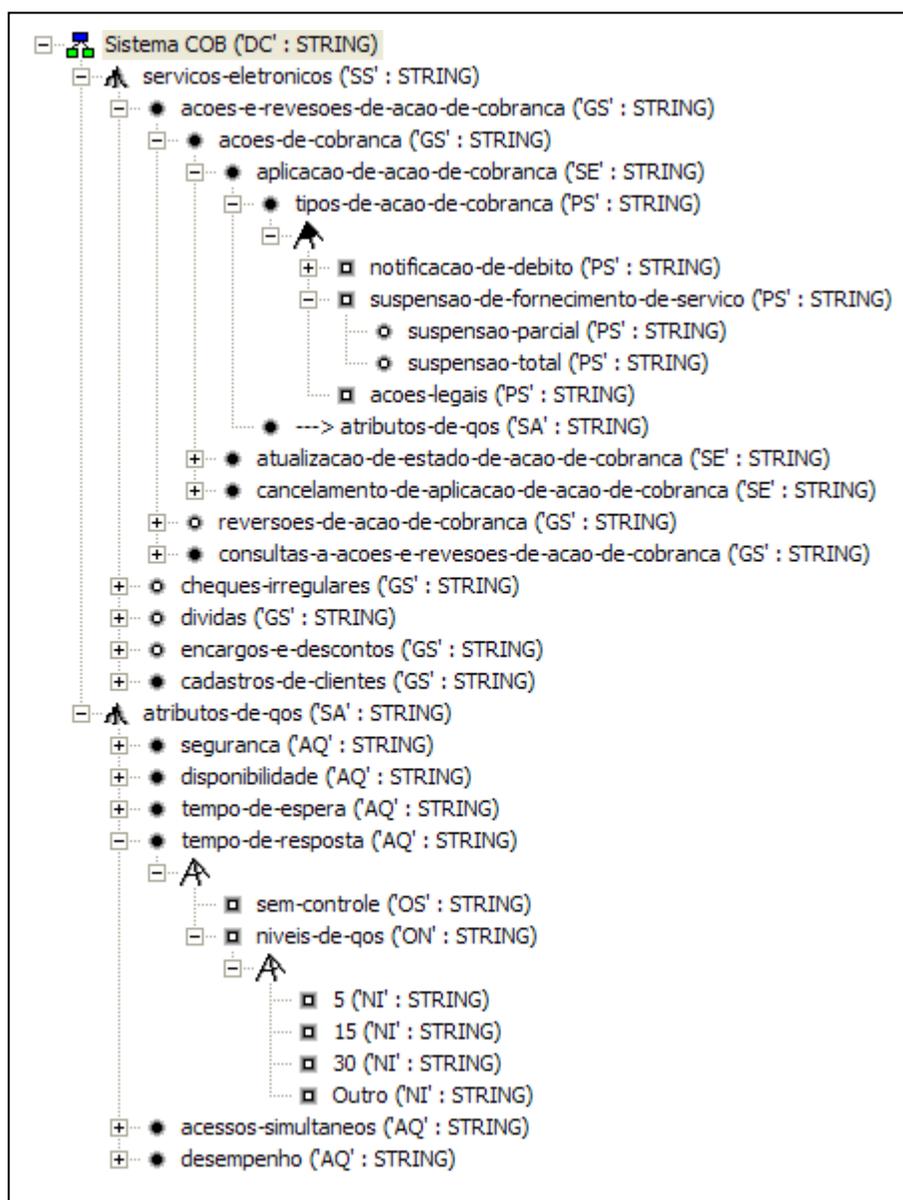


Figura 12: Exemplo de modelo de características para serviços eletrônicos.

4.3.1 Estrutura de Modelo de Características

Os serviços eletrônicos e os atributos de QoS são tratados como pontos comuns e variabilidades nos modelos de características. Eles podem ser especificados como características obrigatórias, opcionais e alternativas. A flexibilidade inerente ao meta-modelo de características permite a definição de serviços eletrônicos e atributos de QoS de diferentes formas. Desse modo, para possibilitar o uso sistemático dos modelos de características para esse fim, é definida uma estrutura padrão de modelos de características especificamente para a representação de serviços eletrônicos e atributos de QoS. Essa estrutura padrão é apresentada na Figura 13 e descrita na seqüência.

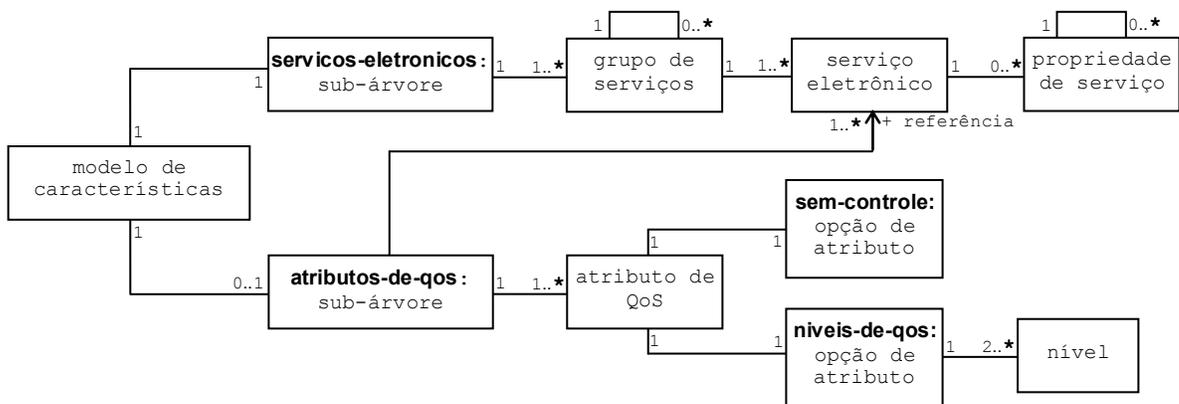


Figura 13: Estrutura de modelo de características para serviços eletrônicos.

A seguir, cada elemento da estrutura de modelo de características para serviços eletrônicos é descrito. O modelo de características cuja parte está apresentada na Figura 12, elaborado seguindo a estrutura padrão, é usado como exemplo.

- (i) Cada modelo de características para serviços eletrônicos consiste de duas sub-árvores, identificadas pelas seguintes características raiz pré-nomeadas:

- sub-árvore **servicos-eletronicos**: essa característica raiz é obrigatória. Ela contém as características que representam os serviços eletrônicos oferecidos por uma organização, incluindo agrupamento de serviços eletrônicos e detalhes funcionais sobre eles. A característica **servicos-eletronicos** é apresentada na parte superior da Figura 12;
- sub-árvore **atributos-de-qos**: essa característica raiz é opcional. Ela contém as características que representam os atributos de qualidade de serviço (QoS) que podem ser associados aos serviços eletrônicos definidos na sub-árvore **servicos-eletronicos**, incluindo as possíveis opções de atributos e níveis. A característica **atributos-de-qos** é apresentada na parte central da Figura 12.

(ii) As características sob a sub-árvore **servicos-eletronicos** podem ser:

- grupo de serviços: essa característica representa um grupo ou um subgrupo de serviços eletrônicos na hierarquia de serviços. Esse tipo de característica é usado para estruturar informações sobre serviços. A sub-árvore **servicos-eletronicos** deve ter pelo menos uma característica grupo de serviços. Exemplo: característica **acoes-de-cobranca**;
- serviço eletrônico: essa característica representa um serviço eletrônico que corresponde a ações a serem realizadas durante a realização de um contrato eletrônico, tais como: regras de negócio, gerência de processo ou busca de informações. Cada característica grupo de serviços deve possuir pelo menos uma característica serviço eletrônico. Exemplo: característica **aplicacao-de-acao-de-cobranca**;

- propriedade de serviço: essa característica representa informações adicionais sobre um serviço eletrônico, necessárias para melhor descrever seus aspectos. Essas características são opcionais e não possuem nenhum tipo de restrição. Exemplo: características **tipos-de-acao-de-cobranca** e **suspensao-parcial**.

(iii) As características sob a sub-árvore **atributos-de-qos** podem ser:

- atributo de QoS: essa característica representa um atributo de QoS que deve ser atribuído à característica serviço eletrônico. A sub-árvore **atributos-de-qos** deve ter pelo menos uma característica atributo de QoS. Características atributo de QoS devem possuir um grupo de características que contenha somente duas características agrupadas pré-nomeadas (**sem-controle** ou **niveis-de-qos**). Exemplo: característica **tempo-de-resposta**;
- opção de atributo **sem-controle**: essa característica agrupada representa a opção em que um atributo de QoS pode ser indefinido para uma determinada característica serviço eletrônico. Exemplo: característica **sem-controle** sob a característica **tempo-de-resposta**;
- opção de atributo **niveis-de-qos**: essa característica agrupada contém os níveis que podem ser definidos para um atributo de QoS para uma determinada característica serviço eletrônico. Uma opção de atributo **niveis-de-qos** deve possuir um grupo de características que contenha seus níveis de QoS. Exemplo: característica **niveis-de-qos** sob a característica **tempo-de-resposta**;

- `nível`: essa característica agrupada representa um determinado nível de QoS que pode ser definido para um atributo de QoS. Uma opção de atributo **niveis-de-qos** deve possuir pelo menos duas características `nível`. Uma das características `nível` pode possuir nenhum valor pré-definido – sendo representado por um `Valor com Tipo`. Exemplo: característica **15**.

(iv) A característica representada na sub-árvore **atributos-de-qos** deve ser referenciada por todas as características do tipo de característica `serviço eletrônico` da sub-árvore **servicos-eletronicos**, por meio do mecanismo de modularização de modelos de características. Exemplo: a referência **ref: atributos-de-qos** da característica **aplicacao-de-acao-de-cobranca**.

Em função das regras apresentadas acima, alguns elementos relacionados a serviços eletrônicos devem ser descritos por características necessariamente obrigatórias, opcionais ou alternativas; enquanto que outros elementos não possuem essa restrição. A Tabela 02 apresenta as opções de obrigatoriedade que devem ser usadas para cada tipo de característica apresentado na Figura 13.

O meta-modelo de características baseado em cardinalidade foi estendido para incluir um atributo que identifica esses tipos de características apresentados – chamado de “tipo de característica”. Eles não são representados graficamente no modelo de características pela ferramenta FeaturePlugin [67, 72], mas representados em um painel de propriedades adicionais (apresentado no próximo capítulo), juntamente com outras propriedades adicionais já existentes. Para uma posterior compreensão rápida de um modelo de características, cada uma das características pode ser nomeada de modo que seus nomes sugerem seu tipo. A Tabela 03 apresenta os possíveis valores usados para o atributo “tipo de característica”.

Tabela 02: Opções de obrigatoriedade para cada tipo de característica.

Tipo de Característica	Opção de Obrigatoriedade
servicos-eletronicos: sub-árvore	Obrigatório
grupo de serviços	Obrigatório / Opcional / Alternativo
serviço eletrônico	Obrigatório / Opcional / Alternativo
propriedade de serviço	Obrigatório / Opcional / Alternativo (quando presente)
atributos-de-qos: sub-árvore	Obrigatório (quando presente)
atributo de QoS	Obrigatório
sem-controle: opção de atributo	Alternativo (com a outra opção de atributo)
niveis-de-qos: opção de atributo	Alternativo (com a outra opção de atributo)
Nível	Alternativo (todos os níveis entre si)

Tabela 03: Sigla dos atributos para cada tipo de característica.

Sigla do Atributo	Tipo de Característica Associado ao Atributo
MC	Modelo de características;
SS	Sub-árvore "serviços eletrônicos";
GS	Grupo de serviços;
SE	Serviço eletrônico;
PS	Propriedade de serviço;
SA	Sub-árvore "atributos de QoS";
AQ	Atributo de QoS;
OS	Opção de atributo "sem controle";
ON	Opção de atributo "níveis de QoS";
NI	Nível de atributo de QoS.

4.3.2 Restrições Adicionais de Características

As restrições adicionais de características oferecidas pela abordagem de modelos de características de Czarniecki et. al. [33] (apresentadas na Seção 3.2.2) também podem ser usadas em modelos de características para serviços eletrônicos. Nenhuma alteração ou adaptação do mecanismo para a abordagem apresentada nesta tese foi necessária. Exemplos de tais restrições adicionais são apresentados juntamente com os demais exemplos de artefatos produzidos pelo estudo de caso que é apresentado no próximo capítulo. Essas restrições adicionais de características são úteis para se forçar uma determinada combinação de valores durante a configuração do modelo de características e a respectiva criação da instância de contrato eletrônico de serviços Web, durante os estágios 4 e 5 respectivamente.

4.4 Criação do Molde de Contrato Eletrônico para Serviços Web

O Estágio 2 do processo proposto consiste em criar o molde de um contrato eletrônico entre as duas organizações envolvidas no processo de negócio. Nesse estágio são usados os dois modelos de características criados no estágio anterior. O molde de contrato eletrônico produzido deve abranger toda a informação que pode possivelmente estar em um contrato eletrônico resultante no final desse processo de cinco estágios. Informações obrigatórias do molde de contrato são diretamente incorporadas no contrato eletrônico, enquanto que a decisão sobre a presença de informações opcionais ou alternativas é adiada até o momento da negociação dos serviços eletrônicos e níveis de QoS a serem contratados (Estágio 4).

O Código 04, o Código 05 e o Código 06 apresentam partes de um molde para contrato eletrônico – a ser usado como exemplo nesta seção. Os exemplos são apresentados na forma de códigos para um melhor entendimento dos elementos. Exemplos gráficos são

apresentados no estudo de caso desta tese bem como os códigos referentes ao molde de contrato eletrônico são apresentados com mais detalhes em capítulo futuro que descreve o estudo de caso. O molde de contrato apresentado se refere a um acordo para a realização de um processo de negócio entre duas organizações – uma que oferece eletronicamente serviços eletrônicos relacionados à cobrança (conforme parte do modelo de características da Figura 12) e outra que usa esses serviços.

Código 04: Exemplo de molde de contrato eletrônico para serviços Web – seção de definições WSDL (sistema de Cobrança).

```
<wsdl:definitions>
  <wsdl:types></wsdl:types>

  <!--FEATURE_ID = aplicacaodeacaodecobranca-->
  <wsdl:message name="aplicacao-de-acao-de-cobranca-Req-MSG">
    <wsdl:part name="aplicacao-de-acao-de-cobranca-Req-MSG-PART"
      type="xsd:string" />

    <!--FEATURE_ID = tiposdeacaodecobranca-->
    <wsdl:part name="tipos-de-acao-de-cobranca-MSG-PART"
      type="xsd:string" />
  </wsdl:message>

  <!--FEATURE_ID = aplicacaodeacaodecobranca-->
  <wsdl:message name="aplicacao-de-acao-de-cobranca-Res-MSG">
    <wsdl:part name="aplicacao-de-acao-de-cobranca-Res-MSG-PART"
      type="xsd:string" />
  </wsdl:message>
  (...)
  <!--FEATURE_ID = acoesdecobranca-->
  <wsdl:portType name="acoes-de-cobrancaPT">
    <!--FEATURE_ID = aplicacaodeacaodecobranca-->
    <wsdl:operation name="aplicacao-de-acao-de-cobrancaOP">
      <wsdl:input message="tns:aplicacao-de-acao-de-cobranca-Req-MSG" />
      <wsdl:output message="tns:aplicacao-de-acao-de-cobranca-Res-MSG" />
    </wsdl:operation>
    (...)
  </wsdl:portType>

  <!--FEATURE_ID = reversoesdeacaodecobranca-->
  <wsdl:portType name="reversoes-de-acao-de-cobrancaPT">
    <!--FEATURE_ID = aplicacaodeacaodecobranca0-->
    <wsdl:operation name="aplic-de-rev-de-acao-de-cobrancaOP">
```

```

        <wsdl:input message="tns:ap-de-rev-de-acao-de-cobranca-Req-MSG" />
        <wsdl:output message="tns:ap-de-rev-de-acao-de-cobranca-Res-MSG" />
    </wsdl:operation>
    (...)
</wsdl:portType>
(...)
<!--FEATURE_ID = acoesdecobranca-->
<plnk:partnerLinkType name="acoes-de-cobrancaPLT">
    <plnk:role name="acoes-de-cobrancaROLE">
        <plnk:portType name="tns:acoes-de-cobrancaPT" />
    </plnk:role>
</plnk:partnerLinkType>

<!--FEATURE_ID = reversoesdeacaodecobranca-->
<plnk:partnerLinkType name="reversoes-de-acao-de-cobrancaPLT">
    <plnk:role name="reversoes-de-acao-de-cobrancaROLE">
        <plnk:portType name="tns:reversoes-de-acao-de-cobrancaPT" />
    </plnk:role>
</plnk:partnerLinkType>
(...)
</wsdl:definitions>

```

Código 05: Exemplo de molde de contrato eletrônico para serviços Web – seção de termos de WS-Agreement (sistema de Cobrança).

```

<wsag:Terms>
    <!--FEATURE_ID = aplicacaodeacaodecobranca-->
    <wsag:ServiceProperties wsag:Name="aplicacao-de-acao-de-cobrancaSP"
        wsag:ServiceName="aplicacao-de-acao-de-cobrancaOP">
        <wsag:VariableSet>
            (...)
            <!--FEATURE_ID = reference19IXItempoderesposta-->
            <wsag:Variable wsag:Name="tempo-de-respostaVAR"
                wsag:Metric="time:duration">
                <wsag:Location><!--A ser completado--></wsag:Location>
            </wsag:Variable>
            (...)
        </wsag:VariableSet>
    </wsag:ServiceProperties>
    (...)
    <!--FEATURE_ID = aplicacaodeacaodecobranca-->
    <wsag:GuaranteeTerm Obligated="fornecedorDeServico">
        <wsag:ServiceScope ServiceName="aplicacao-de-acao-de-cobrancaOP" />

        <wsag:QualifyingCondition><!--A ser comp--></wsag:QualifyingCondition>

        <wsag:ServiceLevelObjective> tempo-de-respostaVAR IS-LESS-INCLUSIVE

```

```

        <!--FEATURE_ID = referencel9IXIsemcontrole2--> sem-controle
        <!--FEATURE_ID = referencel9IXIa5--> 5
        <!--FEATURE_ID = referencel9IXIa16--> 15
        <!--FEATURE_ID = referencel9IXIa31--> 30
        <!--FEATURE_ID = referencel9IXIoutrol--> Outro
    </wsag:ServiceLevelObjective>

    <wsag:BusinessValueList><!--A ser comp--></wsag:BusinessValueList>
</wsag:GuaranteeTerm>
    (...)
</wsag:Terms>

```

Código 06: Exemplo de molde de contrato eletrônico para serviços Web – seção processo de negócio (entre os dois sistemas).

```

<bpel:process>
    (...)
    <bpel:partnerLinks>
        <!--FEATURE_ID = acoesdecobranca-->
        <bpel:partnerLink name="acoes-de-cobranca"
            partnerLinkType="nsl:acoes-de-cobrancaPLT"
            partnerRole="acoes-de-cobrancaROLE"/>

        <!--FEATURE_ID = reversoesdeacaodecobranca-->
        <bpel:partnerLink name="reversoes-de-acao-de-cobranca" (...) />
        (...)
        <!--FEATURE_ID = osparaexecutaracaodecobranca-->
        <bpel:partnerLink myRole="OS-para-executar-acao-de-cobrancaROLE"
            name="OS-para-executar-acao-de-cobranca"
            partnerLinkType="nsl:OS-para-executar-acao-de- (...) />
        <!--FEATURE_ID = osparaexecutarreversaodeacaodecobranca-->
        <bpel:partnerLink myRole="OS-para-executar-reversao-de-acao-de- (...) />
        (...)
    </bpel:partnerLinks>

    <bpel:variables>
        <!--FEATURE_ID = aplicacaodeacaodecobranca-->
        <bpel:variable messageType="nsl:aplicacao-de-acao-de-cobranca-Req-MSG"
            name="aplicacao-de-acao-de-cobranca-Request-MSG"/>

        <!--FEATURE_ID = aplicacaodeacaodecobranca-->
        <bpel:variable messageType="nsl:aplicacao-de-acao-de-cobranca- (...) />
        (...)
        <!--FEATURE_ID = aplicacaodereversaodeacaodecobranca-->
        <bpel:variable messageType="nsl:aplicacao-de-reversao-de-acao-de (...) />

        <!--FEATURE_ID = aplicacaodereversaodeacaodecobranca-->

```

```

<bpel:variable messageType="ns1:aplicacao-de-reversao-de-acao-de (...) />
(...)
<!--FEATURE_ID = registrodeOSparaexecutaracaodecobranca-->
<bpel:variable messageType="ns1:registro-de-OS-para-executar- (...) />

<!--FEATURE_ID = registrodeOSparaexecutaracaodecobranca-->
<bpel:variable messageType="ns1:registro-de-OS-para-executar- (...) />
(...)
</bpel:variables>

<bpel:flow>
  <bpel:sequence>
    <!--FEATURE_ID = aplicacaodeacaodecobranca-->
    <bpel:invoke inputVariable="aplicacao-de-acao-de-cobranca-Req-MSG"
      name="Aplicar-AC"
      operation="aplicacao-de-acao-de-cobrancaOP"
      outputVariable="aplicacao-de-acao-de-cobranca-Res-MSG"
      partnerLink="acoes-de-cobranca"
      portType="ns1:acoes-de-cobrancaPT"/>
    <bpel:flow>
      <bpel:sequence>
        <!--FEATURE_ID = registrodeOSparaexecutaracaodecobranca-->
        <bpel:receive name="Registrar-OS-para-AC"
          operation="registro-de-OS-para-executar- (...) "
          partnerLink="OS-para-executar-acao-de-cobranca"
          portType="ns1:OS-para-executar-acao-de- (...) "
          variable="registro-de-OS-para-executar- (...) "/>

          <!--FEATURE_ID = registrodeOSparaexecutaracaodecobranca-->
          <bpel:reply name="Retornar-ID-da-OS" (...) />
        </bpel:sequence>
        (...)
      </bpel:flow>
      (...)
    </bpel:sequence>
  </bpel:sequence>
  <bpel:sequence>
    <!--FEATURE_ID = aplicacaodereversaodeacaodecobranca-->
    <bpel:invoke inputVariable="aplicacao-de-reversao-de-acao-de- (...) />
    <bpel:flow>
      <bpel:sequence>
        <!--FEATURE_ID = registrodeOSparaexecutarreversaodeaca (...) -->
        <bpel:receive name="Registrar-OS-para-Reverter-AC" (...) />

        <!--FEATURE_ID = registrodeOSparaexecutarreversaodeaca (...) -->
        <bpel:reply name="Retornar-ID-da-OS" (...) />
      </bpel:sequence>
      (...)
    </bpel:flow>
  
```

```
(...)  
</bpel:sequence>  
</bpel:flow>  
</bpel:process>
```

O Código 04 e o Código 05 se referem às partes do molde de contrato que contém, respectivamente, a definição dos serviços eletrônicos e dos atributos de QoS associados a eles, para o sistema de Cobrança. Partes similares do molde de contrato existem para o outro sistema que usa os serviços eletrônicos oferecidos pelo sistema de Cobrança e que também oferece alguns serviços para o primeiro. O Código 06 se refere à parte do molde de contrato que contém a definição do processo de negócio existente entre as duas organizações envolvidas, em função dos serviços eletrônicos contratados.

Os elementos que aparecem nesses três exemplos são os descritos na Seção 2.4.1, na Seção 2.4.2 e na Seção 2.4.3. Esses elementos são mais bem explorados como exemplos nas próximas seções deste capítulo.

4.4.1 Meta-modelo de Contrato Eletrônico para serviços Web

Um meta-modelo de contrato eletrônico foi desenvolvido para representar as regras a serem seguidas durante a criação tanto de contratos eletrônicos quanto para moldes de contratos eletrônicos. O meta-modelo, apresentado na Figura 14, foi criado por meio da unificação dos principais conceitos relacionados a:

- (i) **Serviços Web**: descritos por meio da linguagem WSDL;
- (ii) **Atributos de QoS para serviços Web**: descritos por meio da linguagem WS-Agreement; e
- (iii) **Processos de negócio envolvendo serviços Web**: descritos por meio da linguagem WS-BPEL.

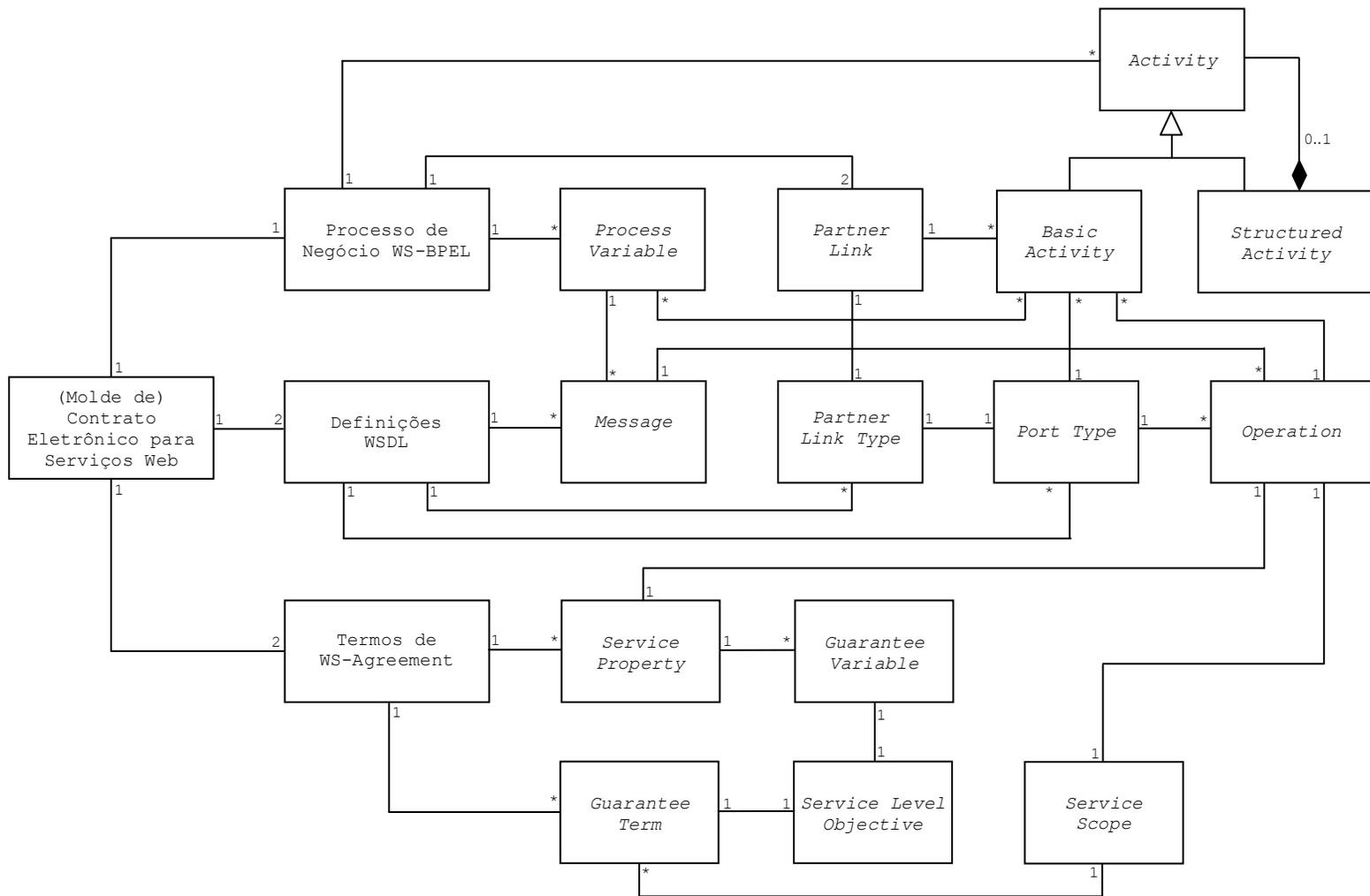


Figura 14: Meta-modelo de contrato eletrônico para serviços Web

A cardinalidade entre elementos do meta-modelo de contrato está limitada para satisfazer apenas às necessidades da abordagem proposta nesta tese. O meta-modelo de contrato eletrônico é uma visão de alto nível dessas três linguagens (WSDL, WS-Agreement e WS-BPEL) e, desse modo, apenas seus elementos significativos são representados. Mesmo alguns elementos de alto nível da linguagem WS-Agreement não são representados, visto que existe uma sobreposição com alguns elementos das linguagens WSDL e WS-BPEL. Os elementos excluídos devido à sobreposição são citados juntamente com a descrição dos elementos do meta-modelo.

Alguns elementos do meta-modelo estão em português, aqueles referentes aos elementos incorporados no meta-modelo para identificar cada uma das seções do contrato eletrônico; enquanto que outros estão em inglês, aqueles referentes aos elementos obtidos a partir das três linguagens de especificação usadas como base para a definição do meta-modelo.

A seguir, os elementos do meta-modelo são descritos brevemente. Dada a alta complexidade das três linguagens usadas, seus elementos não são descritos em detalhes. Apenas uma visão geral de cada agrupamento de elementos é apresentada. Um pouco mais de detalhes a respeito de alguns desses elementos é apresentado na Seção 2.4.1, na Seção 2.4.2 e na Seção 2.4.3. Detalhamentos completos são encontrados em [7, 11, 12]. Os exemplos citados na descrição abaixo se referem a elementos do molde de contrato eletrônico apresentado no Código 04, no Código 05 e no Código 06, elaborado seguindo essas regras.

Um (Molde de) Contrato Eletrônico para Serviços Web é composto por três seções:

- (i) **Seção de definições WSDL:** essa seção contém os elementos básicos da linguagem WSDL: Messages (exemplo: **aplicacao-de-acao-de-cobranca-Req-MSG**), Partner Link Types (exemplo: **acoes-de-cobrancaPLT**), Port Types (exemplo: **acoes-de-cobrancaPT**) e Operations (exemplo: **aplicacao-**

de-acao-de-cobrancaOP). Esses últimos dois descrevem em conjunto os serviços Web propriamente ditos. Esses elementos são usados nas duas outras seções. Os elementos *Type*, *Binding*, *Port* e *Service* não são incluídos nesse meta-modelo, para mantê-lo mais simples. Existem duas seções de definições WSDL no molde de contrato eletrônico – uma para cada uma das duas organizações envolvidas no acordo de negócio interorganizacional. Essa seção é identificada no molde de contrato eletrônico pelos identificadores `<wsdl:Definitions>`;

(ii) **Seção de termos de WS-Agreement**: essa seção contém a descrição dos atributos de QoS, e seus respectivos níveis, relacionados com os serviços Web envolvidos. Os atributos e níveis de QoS são descritos em termos dos seguintes elementos: *Service Properties* (exemplo: **aplicacao-de-acao-de-cobrancaSP**) – incluindo *Variables* (exemplo: **tempo-de-respostaVAR**); e *Guarantee Terms* (exemplo: **fornecedorDeServico**) – incluindo *Service Scope* (exemplo: **aplicacao-de-acao-de-cobrancaOP**) e *Service Level Objectives* (exemplo: **tempo-de-respostaVAR IS-LESS-INCLUSIVE**). Os elementos *Name*, *Context* e *Service Description Terms* não são incluídos nesse meta-modelo visto que já existem elementos similares das duas outras seções representando essas informações. Similarmente ao item anterior, existem duas seções de termos de WS-Agreement no molde de contrato eletrônico – uma para cada uma das duas organizações envolvidas no acordo de negócio interorganizacional. Essa seção é identificada no molde de contrato eletrônico pelos identificadores `<wsag:Terms>`;

(iii) **Seção de processo de negócio WS-BPEL**: essa seção contém a descrição do processo de negócio que faz a composição de serviços Web. O processo de negócio é descrito em termos dos seguintes elementos: *Partner Links* (exemplo: **acoes-de-cobranca**), *Variables* (exemplo: **aplicacao-de-acao-de-**

cobranca-Request-MSG) e *Activities* – tanto do tipo *Basic Activities* (exemplo: **invoke**) quanto do tipo *Structured Activities* (exemplo: **flow**). Os elementos da linguagem WS-BPEL responsáveis pelo tratamento de falhas (*FaultHandler*) não são incluídos nesse meta-modelo para deixá-lo mais simples. Porém, devido sua importância para determinados domínios de contratos, sua inclusão no meta-modelo pode ser necessária, o que pode ser considerada uma extensão simples. Essa seção é identificada no molde de contrato eletrônico pelo identificador `<bpel:Process>`.

Fisicamente, o artefato molde de contrato eletrônico é armazenado em cinco arquivos:

- (i) Dois arquivos, cada um contendo a respectiva seção `<wsdl:Definitions>`, uma para cada uma das duas organizações envolvidas;
- (ii) Dois arquivos, cada um contendo uma seção `<wsag:Terms>`, uma para cada uma das duas organizações envolvidas;
- (iii) Um arquivo contendo a seção `<bpel:Process>`, único para as duas organizações envolvidas.

Na Seção 2.2.1 foram apresentadas como partes comuns de contratos eletrônicos os seguintes tipos de cláusulas contratuais: Obrigações, Permissões e Proibições. Porém, dada a natureza das linguagens de especificação usadas como base para a definição deste meta-modelo de contrato eletrônico, essas cláusulas não estão definidas explicitamente nele. Esse tipo de cláusula contratual é incorporado nos contratos eletrônicos por meio dos elementos existentes no meta-modelo apresentado. Os serviços Web que devem ser obrigatoriamente executados em uma determinada parte do processo de negócio podem ser entendidos como uma obrigação; enquanto que serviços Web que podem ser opcionalmente executados em diferentes estados do processo de negócio, como permissões. Os atributos de QoS, definidos na seção de termos de WS-Agreement, podem

ser entendidos como obrigações ou proibições, dependendo do caso. Visando manter compatibilidade com outros artefatos especificados por meio das linguagens utilizadas aqui, nenhuma alteração foi realizada para deixar explícita a existência de tais cláusulas contratuais.

4.4.2 Mapeamento entre Meta-modelos de Características e de Contrato Eletrônico

A criação de um molde de contrato eletrônico representa o Estágio 2 do processo e é realizado em dois passos:

1. As definições WSDL e os termos de WS-Agreement são criados diretamente a partir dos dois modelos de características para serviços eletrônicos. Para esse primeiro passo, existe um mapeamento de elementos do meta-modelo de características para elementos do meta-modelo de contrato eletrônico;
2. O processo de negócio escrito em WS-BPEL é criado a partir das definições WSDL e informações adicionais definidas durante esse estágio do processo. Esse passo pode ser apoiado por uma ferramenta de especificação de processos na linguagem WS-BPEL, tais como: Oracle BPEL Process Manager [74], Eclipse BPEL Designer [75] e Active BPEL [76].

Para tornar possível o mapeamento entre o meta-modelo de características e o meta-modelo de contrato eletrônico, referente ao primeiro item apresentado acima, foi definido um conjunto de regras descritas a seguir. A Figura 15 e a Figura 16 representam graficamente essas regras.

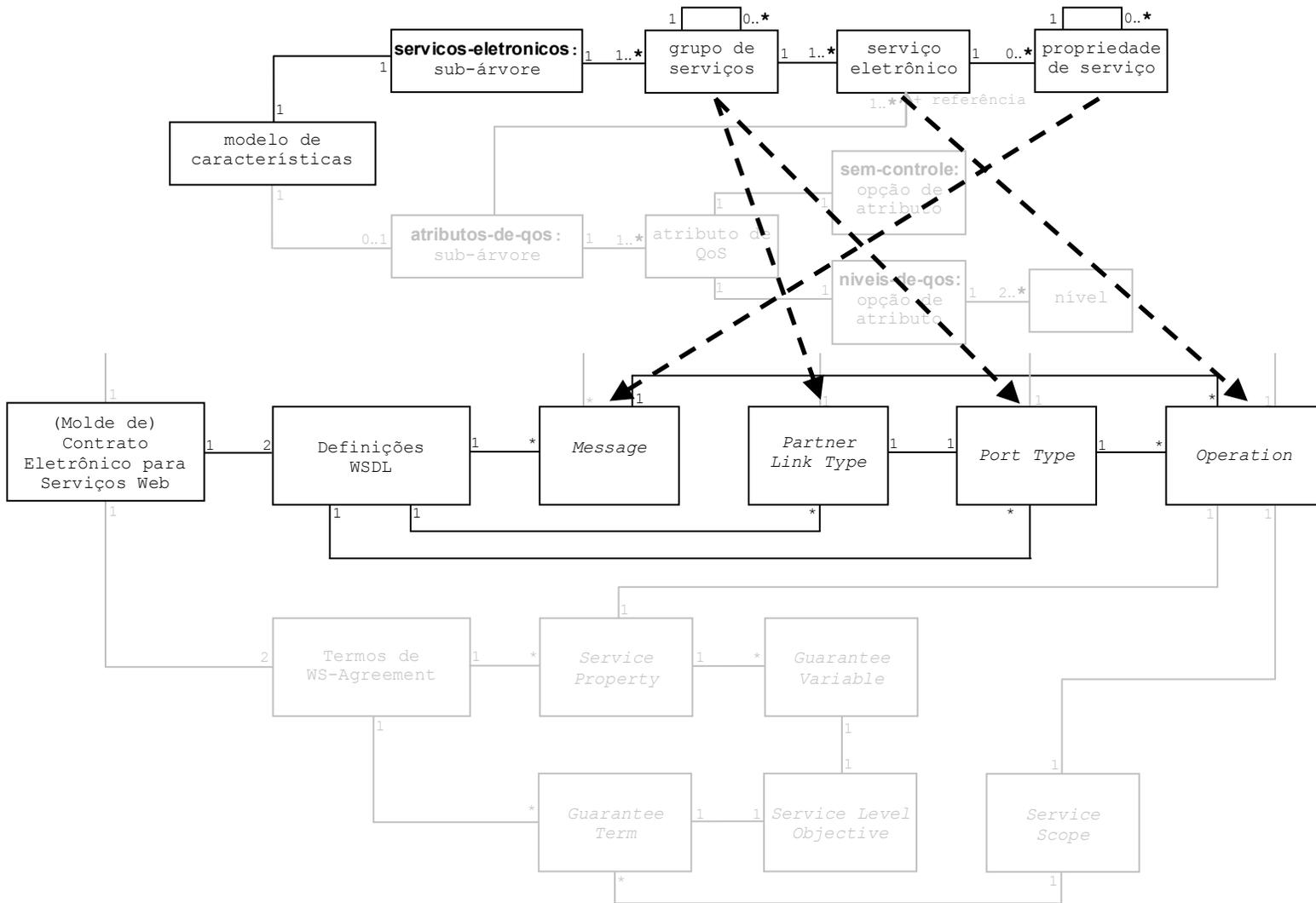


Figura 15: Mapeamento entre características de Serviços Eletrônicos e Definições WSDL.

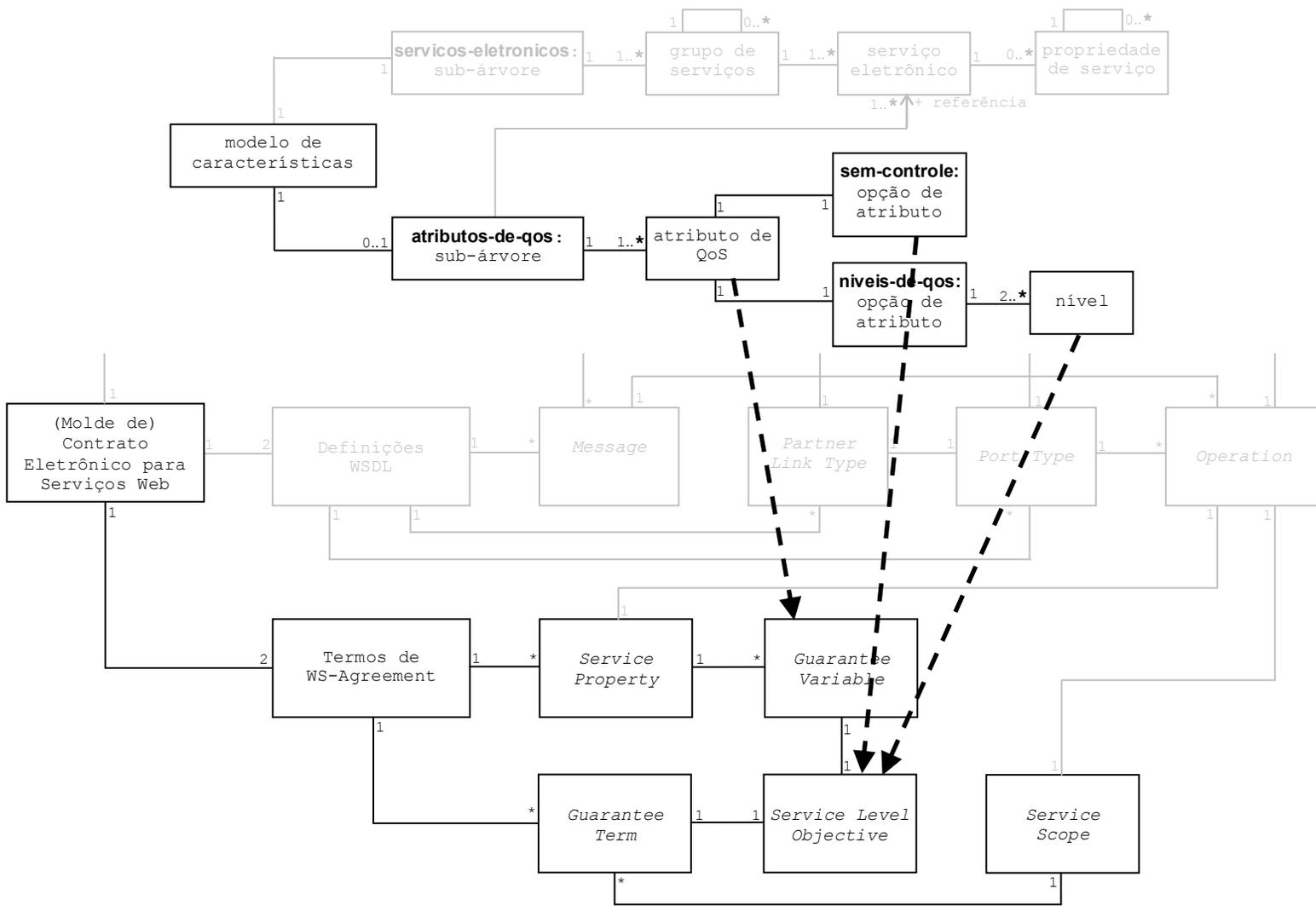


Figura 16: Mapeamento entre características de Atributos de QoS e Termos de WS-Agreement.

- **Definições WSDL:** é a seção do (molde de) contrato eletrônico que inclui os elementos de serviços Web (apresentadas na Figura 15). Essa seção corresponde às características da sub-árvore **servicos-eletronicos**, contendo:
 - **Port Types (Tipos de Porta):** um tipo de porta é criado para cada característica do tipo grupo de serviços;
 - **Operations (Operações):** uma operação (para um tipo de porta) é criada para cada característica do tipo serviço eletrônico existente a partir da hierarquia de características de grupo de serviços. Para cada operação, é criada pelo menos uma variável de entrada e uma variável de saída, com base nos tipos de mensagens definidos a seguir;
 - **Messages (Mensagens):** dois tipos de mensagem (uma de entrada e uma de saída), a serem usadas como variáveis pelas operações, são criados para cada característica do tipo serviço eletrônico. Além disso, um tipo de mensagem (de entrada) é criado para cada característica do tipo propriedades de serviços existente a partir da hierarquia de características de serviço eletrônico. Todos os tipos de mensagem são criados sendo do tipo `xsd:string`;
 - **Partner Link Types (Tipos de ligação entre parceiros):** um tipo de ligação entre parceiros é criado para cada característica do tipo grupo de serviços;
 - **Types (Tipos):** nenhum tipo de dado complexo é gerado automaticamente para o molde de contrato. Apenas o tipo de dado simples `xsd:string` é usado na geração dos outros elementos do molde. Tipos de dados complexos, quando necessários, devem ser criados após a geração automática do molde.

- **Termos de WS-Agreement:** a parte do (molde de) contrato eletrônico que inclui as informações de QoS associados aos serviços Web (apresentadas na Figura 16). Essa seção corresponde às características da sub-árvore **atributos-de-qos**, contendo:
 - ***Service Properties/Guarantee Variables (Propriedades de Serviço/Variáveis de Garantia):*** uma variável de garantia (associada a uma propriedade de serviço) é criada para cada característica do tipo `atributo` de QoS, para cada característica do tipo `serviço eletrônico`;
 - ***Guarantee Terms/Service Level Objectives (Termos de Garantia/Objetivos de Nível de Serviço):*** um objetivo de nível de serviço (associado a um termo de garantia) é criado para a característica agrupada que representa a opção de `atributo sem-controle` e para cada uma das sub-características do tipo `nível` da característica agrupada que representa a opção de `atributo niveis-de-qos`, para cada característica do tipo `serviço eletrônico`.

Os elementos das seções **Definições WSDL** e **Termos de WS-Agreement** são criados usando as estruturas mais simples da respectiva linguagem de especificação. Considerando as regras estabelecidas, esses elementos podem ser gerados de forma automática. Os elementos gerados inicialmente podem ser expandidos para estruturas mais complexas, caso seja necessário. As formas básicas em que esses elementos são criados são detalhadas no capítulo relativo ao estudo de caso, em que os exemplos mais completos produzidos pelo estudo de caso são apresentados.

Como o tipo de ambos os artefatos, modelos de características para serviços eletrônicos e molde de contrato eletrônico, são documentos na linguagem XML, o mapeamento entre eles pode ser realizado automaticamente com o uso de um processador de documentos XSLT (*XSL Transformations*) [77]. XSLT é uma linguagem para a transformação de

documentos XML – que seguem as regras de um determinado esquema XML, em outros documentos XML – que seguem as regras de outro esquema XML.

As propriedades das características em relação à sua obrigatoriedade são levadas em consideração durante a criação de um molde de contrato eletrônico. Conseqüentemente, algumas partes do molde de contrato também serão identificadas como partes obrigatórias, opcionais ou alternativas. Mesmo os elementos do molde de contrato eletrônico não relacionados diretamente com os elementos dos modelos de características (como apresentado anteriormente) são indiretamente relacionados a elas, de acordo com os relacionamentos existentes no meta-modelo de contratos eletrônicos. Por exemplo, uma atividade básica, elemento de processo de negócio *Basic Activity*, é indiretamente relacionado a alguma característica devido a seu relacionamento com uma operação (elemento de processo de negócio *Operation*) e, portanto, ambos serão identificados como obrigatórios, opcionais ou alternativos da mesma forma.

Desse modo, um molde genérico de contrato eletrônico é criado para diversos tipos de integração entre as duas organizações. Esse molde genérico é especializado durante cada criação de uma nova instância de contrato, de acordo com os respectivos modelos de características configurados. É possível que o processo se inicie com uma versão básica do molde de contrato eletrônico que pode evoluir à medida que ele é usado e reusado.

Ambos os artefatos – modelo de características para serviços eletrônicos e molde de contrato eletrônico para serviços Web são estruturados de forma hierárquica. O mapeamento existente entre os elementos desses dois artefatos replica no segundo a estrutura hierárquica existente no primeiro. Um exemplo dessa regra: da mesma forma que, em um modelo de características, uma característica do tipo *grupo de serviços* agrupa várias características do tipo *serviço eletrônico*; em um molde de contrato eletrônico, o elemento *tipo de porta* relacionado a essa característica do tipo *grupo de serviços* agrupa vários elementos *operações* referentes às mesmas características do tipo *serviço eletrônico*. A mesma regra se aplica às demais partes do molde de contrato que estão relacionadas com outras respectivas partes do modelo de características.

Anotações no Molde de Contrato

Para manter a rastreabilidade entre os documentos fonte e destino, anotações são criadas no molde de contrato eletrônico. As anotações correspondem a comentários na linguagem XML, ligando os elementos dos modelos de características para serviços eletrônicos aos elementos do molde de contrato eletrônico derivado por meio de um identificador. Um exemplo, no Código 04, é a anotação `<!--FEATURE_ID = acoesdecobranca-->` que está ligando o tipo de porta **acoes-de-cobrancaPT** do molde de contrato eletrônico com a característica **acoes-de-cobranca** do modelo de características para serviços eletrônicos – uma característica obrigatória, nesse caso específico.

Mesmo os elementos do molde de contrato eletrônico criados na sequência da transformação XSLT automática devem ser anotados. As anotações são necessárias para apontar para os elementos dos modelos de características para serviços eletrônicos com os quais eles estão indiretamente ligados. Portanto, no Código 06, há elementos também anotados com o mesmo comentário citado no parágrafo anterior `<!--FEATURE_ID = acoesdecobranca-->`, por estarem indiretamente ligados à mesma característica **acoes-de-cobranca** do modelo de características para serviços eletrônicos. Um exemplo é o `<bpel:partnerLink name="acoes-de-cobranca" (...)/>` que está ligado indiretamente à característica citada por meio de um elemento `Partner Link Type` da especificação WSDL que está ligada diretamente à tal característica.

4.5 Configuração dos Modelos de Características para Serviços Eletrônicos

Considerando que os artefatos referentes aos estágios 1, 2 e 3 que correspondem ao desenvolvimento do molde de contrato eletrônico já estão elaborados, é possível iniciar os estágios 4 e 5 correspondentes à criação de uma instância de contrato eletrônico. Os

estágios 4 e 5 são realizados para cada instância de contrato eletrônico que se deseja criar. As seções anteriores apresentaram a maior parte das informações necessárias para a estruturação dos artefatos produzidos pelo processo proposto nesta tese. Nesta seção é apresentado o Estágio 4 e na próxima o Estágio 5.

O Estágio 4 do processo proposto consiste em configurar os dois modelos de características para serviços eletrônicos, elaborados no Estágio 2. Essa configuração corresponde à principal fase de negociação em que os serviços eletrônicos e os atributos e níveis de QoS são escolhidos. O par de modelos de características configurados resultante desse estágio representa a informação a ser realmente usada no processo de negócio sendo contratado.

A negociação entre as partes envolvidas é realizada completamente de acordo com as técnicas de configuração para modelos de características [33]. Nenhuma alteração do procedimento de configuração de modelos de características, conforme proposto por Czarnecki et. al. [33], precisou ser realizada.

Portanto, a configuração dos modelos de características para serviços eletrônicos mantém as características obrigatórias que representam os serviços eletrônicos e/ou atributos de QoS; enquanto que as características opcionais e alternativas são escolhidas de acordo com a negociação realizada entre as partes. Além disso, as restrições de cardinalidade e restrições adicionais de características devem ser obedecidas.

Desse modo, as atividades de negociação entre duas organizações, visando estabelecer um processo de negócio interorganizacional, são baseadas nos modelos de características para serviços eletrônicos oferecidos por cada uma delas. A negociação é realizada de uma forma sistemática, em que as partes devem convergir para se decidir quais características opcionais e/ou alternativas devem ser selecionadas.

Na Figura 17, é apresentada parte de uma configuração do modelo de características para serviços eletrônicos do sistema de cobrança correspondente ao modelo de característica apresentado na Figura 12.

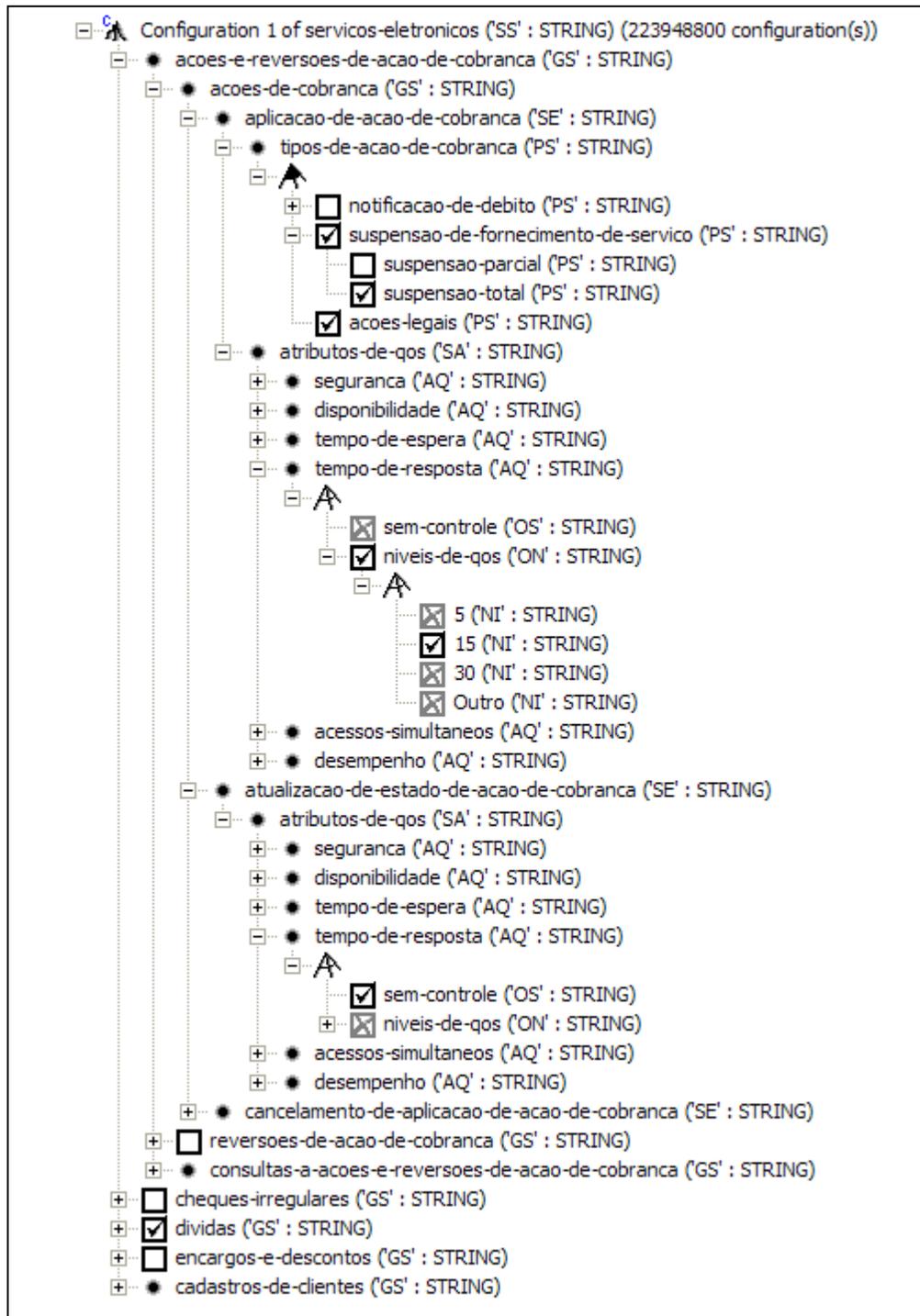


Figura 17: Exemplo de configuração de modelo de características para serviços eletrônicos.

As características obrigatórias são apresentadas como pré-selecionadas; algumas características opcionais/alternativas são apresentadas selecionadas enquanto outras não. Por exemplo, apenas algumas sub-características da característica **tipos-de-acao-de-cobranca**, todas opcionais/alternativas do tipo propriedade de serviço, foram selecionadas.

Em relação aos atributos de QoS, um mesmo atributo, definido apenas uma vez no modelo de características, está associado a diferentes características do tipo serviço eletrônico, e assim podem apresentar diferentes escolhas para cada uma delas. Na Figura 17, um exemplo é a característica **tempo-de-resposta** do tipo atributo de QoS. Como a característica **tempo-de-resposta** está associada a, pelo menos, duas características diferentes do tipo serviço eletrônico – **aplicacao-de-acao-de-cobranca** e **atualizacao-de-estado-de-acao-de-cobranca**, ela apresenta diferentes escolhas para diferentes serviços eletrônicos: nível de QoS **15** e **sem-controle**, respectivamente.

4.6 Criação da Instância de Contrato Eletrônico para Serviços Web

O Estágio 5 corresponde à criação da instância de contrato eletrônico. Nesse estágio, o molde de contrato eletrônico é refinado levando-se em consideração os dois modelos de características para serviços eletrônicos configurados no estágio anterior. Após a finalização da fase de negociação no referido estágio, detalhes pendentes são definidos e informações não requeridas são removidas do contrato. Em um primeiro passo, as alterações são realizadas automaticamente no molde de contrato eletrônico, em função das decisões tomadas durante a negociação dos serviços eletrônicos e níveis de QoS a serem contratados (seleção ou não de algumas características). Em um segundo passo, os responsáveis pela criação da instância de contrato eletrônico podem realizar, de forma manual, a inclusão de elementos cuja necessidade não foi detectada nos estágios

anteriores, e assim não estão previstos nos artefatos usados; ou mesmo alguns ajustes finos no contrato em função de alguma necessidade específica também não prevista anteriormente.

Como todos os elementos do molde de contrato estão relacionados direta ou indiretamente às características por meio de anotações (comentários), os modelos de características configurados no estágio anterior podem ser usados por um analisador em um processo automático de remoção. Nesse processo, são mantidos no contrato eletrônico resultante todos os elementos do molde de contrato eletrônico associados a características obrigatórias ou a características opcionais e/ou alternativas que tiverem sido selecionadas durante a configuração dos modelos de características. Por outro lado, são removidos do contrato eletrônico resultante todos os elementos do molde de contrato associados a características, opcionais e/ou alternativas, não selecionadas durante a configuração dos modelos de características. Além da remoção desses elementos, as anotações do contrato eletrônico, na forma de comentários XML, são também removidas por não serem mais necessárias.

Como exemplo, no Código 04, no Código 05 e no Código 06, que representam um molde de contrato eletrônico, alguns elementos são mantidos em função do modelo de características configurado apresentada na Figura 17, enquanto que outros não. Por exemplo, todos os elementos do molde de contrato anotados com o comentário **<!--FEATURE_ID = acoesdecobranca-->** são mantidos, por estarem associados a uma característica obrigatória. Já os elementos anotados com o comentário **<!--FEATURE_ID = reversoesdeacaodecobranca-->** são removidos, por estarem associados a uma característica opcional não selecionada.

Durante as criações de possíveis instâncias de contrato eletrônico, informações não previstas podem ser requeridas – tais como: novos serviços eletrônicos, novos níveis e/ou atributos de QoS, e novas atividades no processo de negócio. Nesse caso, os artefatos básicos usados para o estabelecimento do contrato eletrônico (modelos de características para serviços eletrônicos e molde de contrato eletrônico) podem ser atualizados para

incluir essas novas informações. Essa facilidade permite o reuso de informações durante o futuro estabelecimento de novos contratos eletrônicos similares, a partir do mesmo conjunto básico de artefatos. Depois da realização desse estágio, o contrato eletrônico pode ser disponibilizado para uso por algum SGPN.

4.7 Trabalhos Relacionados

Nesta seção são apresentados os principais trabalhos relacionados à abordagem apresentada neste capítulo. No final da seção é apresentado um quadro resumo comparando os vários trabalhos apresentados e a abordagem proposta aqui. Considerando que, do ponto de vista conceitual, praticamente não há diferenças entre os contratos eletrônicos para serviços eletrônicos de forma geral e os contratos eletrônicos específicos para serviços Web, ambos os tipos são considerados nesta seção.

Enquanto a abordagem proposta nesta tese está relacionada ao apoio para o estabelecimento de contratos eletrônicos, a grande maioria dos projetos existentes na área de contratos eletrônicos está relacionada às fases posteriores a esta. Portanto, são normalmente abordados assuntos tais como a realização de contratos eletrônicos, englobando tanto sua execução quanto seu monitoramento. Além disso, muitos trabalhos estão relacionados à especificação de processos de negócio, de modo a torná-lo bem entendido tanto por humanos quanto pelas máquinas. Esses trabalhos, por não estarem diretamente relacionados ao foco dessa tese, não são cobertos aqui.

Dos trabalhos que compem também o estabelecimento de contratos eletrônicos, muitos deles não apresentam facilidades para reuso de informações entre contratos eletrônicos similares, foco da proposta nesta tese. Eles estão normalmente relacionados ao processo de negociação a ser seguido e às ferramentas a serem usadas durante a negociação entre as partes envolvidas [53, 78-81].

Os trabalhos que apresentam abordagens sistemáticas para o estabelecimento de contratos eletrônicos normalmente o fazem com base na aplicação de meta-modelos e/ou moldes de contrato eletrônico [17, 20-30, 82-89]. Em ambos os casos, estas facilidades permitem um reuso e estruturação de informações de forma básica e limitada, visto não terem essas questões como principais objetivos.

O projeto COSMOS [82] é um das primeiras iniciativas a tratar de forma sistemática o uso de contratos eletrônicos, por meio de um meta-modelo de contrato que permite o uso de moldes de contrato. Esse projeto apresentou conceitos básicos relacionados a meta-modelos e moldes de contrato que são usados ainda atualmente nos projetos mais recentes. O meta-modelo de contrato proposto por esse projeto dividia um contrato em quatro seções principais: “Quem”, “Quando”, “O que” e “Legal”. Um dos projetos mais recentes fortemente baseados nesses conceitos é a infra-estrutura¹⁰ 4W [28], que disponibiliza um meta-modelo de contrato eletrônico constituído por cláusulas contratuais que descrevem diferentes condições ou situações envolvidas. Estas cláusulas podem ser de quatro tipos, representados por quatro grupos de conceitos: “Quem” (*Who*), “Onde” (*Where*), “O que” (*What*) e “Como” (*How*). Enquanto o projeto COSMOS cobre o estabelecimento de contratos entre apenas duas partes, a infra-estrutura 4W cobre contratos entre duas ou mais partes.

O projeto CrossFlow [17] é também um dos projetos precursores que lidam de forma sistemática com contratos eletrônicos por meio de moldes de contrato, porém mais moderno e completo do que o projeto COSMOS. Entretanto, similarmente ao projeto COSMOS, o projeto CrossFlow oferece apoio a contratos envolvendo apenas duas partes. Esse projeto propõe um sistema de gerência de processos de negócio interorganizacionais para controlar a execução de serviços eletrônicos oferecidos por diferentes organizações. Todo o ciclo de vida dos processos de negócio gerenciados pelo sistema é baseado em contratos eletrônicos que especificam as interações existentes entre fornecedores e consumidores de serviços eletrônicos em empresas virtuais. A seleção dos serviços a

¹⁰ Do inglês: framework

serem executados durante a realização do contrato é feita de forma dinâmica, em que os serviços são escolhidos em tempo de execução. Os contratos eletrônicos são estabelecidos com base em um meta-modelo de contrato [3]. A estratégia do projeto CrossFlow para reuso de informações relacionadas a contratos eletrônicos é o uso de moldes genéricos de contrato eletrônico [83]. Esses são criados em função de contratos similares já estabelecidos e freqüentemente usados em um determinado segmento de mercado. Durante a criação de moldes, os campos que possuem valores variáveis para cada contrato eletrônico específico são deixados em branco para serem preenchidos posteriormente. Os últimos trabalhos publicados do projeto CrossFlow indicavam a necessidade de formas mais apropriadas para lidar com moldes de contrato [17].

Outros trabalhos de pesquisa avançaram na abordagem de uso de moldes de contrato eletrônico. Chiu et. al. [20] propuseram uma abordagem que usa variáveis controladas para moldes de contrato que podem envolver várias organizações. Além disso, tanto os moldes de contrato quanto as variáveis de molde de contrato são tratados como novas entidades em um meta-modelo de contrato eletrônico. Um molde de contrato pode conter variáveis cujos valores são definidos e alterados de forma controlada, culminando na escolha de um dos possíveis valores durante o estabelecimento de contrato eletrônico. Essa abordagem apresenta duas infra-estruturas para a realização de contratos eletrônicos. A primeira é usada para a execução das atividades de contratos eletrônicos [84, 85], enquanto que a segunda é usada para garantir o cumprimento das cláusulas de contratos eletrônicos [20]. Associados a essas duas infra-estruturas, existem dois meta-modelos. O primeiro deles é o meta-modelo de contrato eletrônico, em que está previsto o uso de moldes de contrato e variáveis de molde. O segundo meta-modelo é o meta-modelo de cumprimento de contrato eletrônico, em que as cláusulas contratuais são definidas em termos de regras de ECA (Evento-Condição-Ação).

Mais recentemente, Berry & Milosevic [25] exploraram o conceito de molde de comunidade de contrato eletrônico, uma forma de molde de domínio de contrato – em uma abordagem similar à apresentada no parágrafo anterior. Um molde de comunidade é um

contrato eletrônico parcialmente especificado, envolvendo duas ou mais organizações, nos quais valores parametrizados são associados durante o tempo de ativação do contrato. Nessa abordagem, os moldes de contrato eletrônico são apresentados como formulários padrão de contrato eletrônico, cujos parâmetros são associados a valores concretos apenas durante o estabelecimento (ou ativação) de contratos, quando uma instância específica do formulário é criada.

Em outra linha de pesquisa, Farrell et. al. [26] usam o ‘cálculo de eventos’¹¹ para rastrear o estado normativo de contratos eletrônicos entre duas ou mais partes envolvidas, também fazendo uso de moldes de contrato. Neste trabalho, de forma similar aos apresentados anteriormente, múltiplas instâncias de contrato eletrônico podem ser criadas com base em um molde de contrato. Diferentes parâmetros podem ser usados para permitir a adaptação do molde para uma instância particular de contrato eletrônico.

Hoffner e Field [27] desenvolvem um trabalho de transformação de acordos textuais em contratos eletrônicos com base principalmente em moldes de contrato. Um processo de busca e descoberta¹², proposto neste trabalho, consiste em selecionar um molde correto entre os existentes e então preencher os campos variáveis desse molde com a informação da descrição do acordo textual. Alguns outros trabalhos relacionados a técnicas de busca e descoberta de atributos e níveis de QoS estão apresentados em [86] e [87].

A infra-estrutura UCM (*Unified Contract Management*) oferece apoio a todas as fases do ciclo de vida de contratos eletrônicos, com base em ontologias [29, 88, 89]. Os contratos UCM cobrem apenas duas organizações envolvidas. Essa infra-estrutura inclui uma abordagem para a modelagem conceitual de contratos eletrônicos formada por três camadas hierárquicas de ontologias de contrato eletrônico. A camada de ontologia de mais baixo nível representa a ontologia de molde de contrato eletrônico. Essa camada é constituída por uma biblioteca de moldes projetados ou implementados com base nas camadas de ontologia de níveis superiores – que contém as informações genéricas sobre

¹¹ Do inglês: event calculus

¹² Do inglês: matchmaking

contratos eletrônicos. Apesar da abordagem ser apresentada como baseada em ontologias, não são apresentados porém mecanismos para inferências, comuns ao conceito de ontologias [43, 44].

Existem também alguns trabalhos relacionados especificamente à descrição de propriedades de serviços Web no contexto de contratos eletrônicos. Um desses trabalhos está relacionado com a infra-estrutura WSLA (*Web Service Level Agreement*) [30], que oferece apoio à criação e ao uso de atributos de QoS para serviços Web em moldes de contrato eletrônico envolvendo duas ou mais partes. Essa abordagem facilita o processo automático dinâmico de busca de informações na Internet. O uso dessa infra-estrutura permite que fornecedores de serviços ofereçam serviços Web em diferentes níveis de qualidade, dependendo das necessidades dos clientes, podendo gerar custos também diferenciados. Uma implementação de WSLA é disponibilizada como parte do IBM Web Services Toolkit [90].

Outro trabalho bastante amplo no tratamento exclusivamente de cláusulas de QoS e apresentado em [91, 92]. Trata-se de uma infra-estrutura para a gerência inteligente de SLAs entre duas ou mais organizações envolvidas, aplicável quando serviços são encapsulados e disponibilizados como serviços Web. O principal objetivo da plataforma proposta é oferecer aos fornecedores mecanismos para: o monitoramento de violações de SLAs; a previsão de violações de SLAs antes que elas ocorram para que se seja possível a tentativa de ações corretivas; e a análise de violações de SLAs visando o entendimento de suas causas, ajudando a identificar como melhorar as operações para cumprir o SLA.

Ainda na linha de atributos de QoS, um trabalho mais recente apresenta uma infra-estrutura unificada para a descrição contratual compreensiva de serviços Web [93]. Além disso, dois padrões complementares são propostos para lidar com políticas de garantia para serviços Web: WS-Agreement [12, 13] e WS-Policy [14, 15], incluindo moldes de contratos eletrônicos da mesma forma que as abordagens mencionadas anteriormente.

A Tabela 04 apresenta um resumo das principais abordagens apresentadas nesta seção. Os seguintes aspectos foram considerados como estando presente ou não em cada uma das

abordagens: 1) abordagem específica para serviços Web; 2) tratamento de atributos e níveis de QoS; 3) uso de meta-modelo de contrato eletrônico; 4) uso de moldes de contrato eletrônico; 5) uso de linguagens baseadas em XML para a especificação de contratos; 6) abordagem dinâmica para a seleção de serviços eletrônicos; 7) apoio a contratos eletrônicos envolvendo mais do que duas partes; 8) apoio ao uso de regras ECA; 9) tratamento de aspectos legais relacionados a contratos eletrônicos; e 10) oferecimento de algum mecanismo sistemático para reuso de informações entre diferentes contratos, além de moldes de contratos – quando presentes.

Tabela 04: Comparação entre as abordagens similares à proposta nesta tese.

	Serviços Web	QoS	Meta modelo	Moldes	XML	Ligação dinâmica	Multi partes	Regras ECA	Aspectos legais	Reuso sistemático
COSMOS [82]		X	X	X	X				X	
CrossFlow [17]		X	X	X	X	X			X	
Milosevic [25]			X		X		X	X	X	
Chiu [20]	X		X	X			X	X		
ER^{EC} [22]	X		X		X		X	X		
4W [28]			X	X	X		X			
UCM [29]			X	X						
SLA [91]	X	X	X				X	X		
WSLA [30]	X	X	X	X	X	X	X			
Farrell [26]		X	X		X		X		X	
Esta proposta	X	X	X	X	X					X

Os critérios usados aqui para a comparação entre as diferentes abordagens não representam nenhuma lista oficial ou pré-definida por alguma organização independente para esta área de pesquisa. Eles foram definidos durante o desenvolvimento da abordagem apresentada aqui, em função das principais características apresentadas pelos trabalhos relacionados que foram analisados. Desta forma, é possível que haja outros critérios que poderiam ser usados nesta comparação, mostrando outras limitações da abordagem proposta aqui, mas que não foram destacados nos trabalhos relacionados.

Dez projetos considerados representativos nesta área de pesquisa são apresentados na tabela anterior. Na última linha são apresentadas propriedades relacionadas à abordagem proposta nesta tese. Considerando os aspectos relacionados especificamente ao estabelecimento de contratos eletrônicos, essa abordagem se apresenta como um misto de várias outras abordagens, com um diferencial não apresentados em nenhuma delas – o uso de modelos de características para apoiar tal processo como um mecanismo sistemático para o reuso de informações. Entretanto, várias das abordagens comparadas possuem propriedades que visam a facilitar as fases seguintes ao estabelecimento do contrato: execução e monitoramento do contrato, não contempladas nesse trabalho. Um exemplo é o apoio à especificação de regras ECA, propriedade essa não apresentada pela abordagem proposta aqui. Outras duas propriedades apresentadas por outras abordagens e não apresentadas aqui são: uso de uma abordagem dinâmica para a seleção de serviços eletrônicos, o que é oferecido por poucas das abordagens sendo comparadas; e apoio a contratos eletrônicos envolvendo mais do que duas partes, sendo oferecido pela grande maioria das outras abordagens na comparação.

4.8 Considerações Finais

Neste capítulo foi apresentada uma nova abordagem para o estabelecimento de contratos eletrônicos, baseada em modelos de características e inspirada na abordagem de linha de produto de software (LP). Além do uso de modelos de características, a abordagem proposta é baseada no uso de moldes de contratos eletrônicos para agilizar o processo de estabelecimento de novos contratos como criação de instâncias de moldes existentes.

Esta abordagem consiste principalmente em um processo de cinco estágios – os quais são divididos em duas fases: desenvolvimento do contrato e criação da instância de contrato. Os três primeiros estágios, que compreendem a primeira fase, estão relacionados a: a elaboração de modelos de características para serviços eletrônicos; a criação de molde

de contrato eletrônico; e o desenvolvimento e a publicação de serviços Web. Os dois últimos estágios, que compreendem a segunda fase, estão relacionados a: configuração de modelos de características para serviços eletrônicos; e criação de instância de contrato eletrônico.

A maior contribuição desta abordagem é oferecer uma forma sistemática e eficiente para a estruturação e o reuso de informação, aperfeiçoando assim o processo de estabelecimento de contratos eletrônicos. A abordagem oferece um meio para representar informações de contratos eletrônicos por meio de modelos de características que podem ser transformados em moldes de contrato eletrônico. Além disso, ela permite a gerência eficiente de partes obrigatórias, opcionais e alternativas de contratos eletrônicos, não apenas do processo de negócio, mas também dos atributos de QoS associados aos serviços Web contratados.

Na grande maioria dos trabalhos relacionados apresentados na Seção 4.7, existem diferentes limitações relacionadas ao uso de moldes de contratos eletrônicos, tais como: moldes de contrato são normalmente tratados como apenas contratos eletrônicos parcialmente preenchidos que possuem campos vazios a serem preenchidos com algum valor, possivelmente de uma lista de valores predefinidos; e normalmente, não há apoio algum a representação e seleção de parte de contrato eletrônico que sejam opcionais ou alternativas. É esperado que com o uso de modelos de características – em um processo similar a LP, como proposto nesta tese, seja possível se obter um melhor reuso de informações nos moldes de contrato eletrônico. Essa nova abordagem pode ser entendida como uma forma de parametrização de contrato eletrônico que deve ser melhorada durante a criação de diversas instâncias de contrato que podem ocorrer para um mesmo tipo de contrato eletrônico.

Embora a abordagem proposta nesta tese possua uma grande parte das características existentes nas outras abordagens, existem também algumas limitações associadas a elas. Três limitações já apresentadas estão relacionadas a: a falta de apoio a acordos entre mais do que duas partes; o uso de um modelo estático de negociação; e a não cobertura de um

protocolo de negociação bem definido entre as partes envolvidas. Algumas limitações adicionais são apresentadas na avaliação da abordagem apresentada no capítulo referente ao estudo de caso.

No próximo capítulo é apresentado o conjunto de ferramentas FeatureContract, desenvolvido para oferecer apoio automatizado à abordagem proposta nesta tese. FeatureContract é composto por uma série de ferramentas, responsáveis por oferecer apoio a diferentes estágios do processo de estabelecimento de contratos eletrônicos para serviços Web.

Capítulo 5

Conjunto de Ferramentas

FeatureContract

No Capítulo 4 foi apresentada uma nova abordagem para o estabelecimento de contratos eletrônicos para serviços Web, com base em modelos de características. O embasamento conceitual e técnico usado para o desenvolvimento de tal abordagem indicam a possibilidade de redução da complexidade desse processo, por meio de melhorias tanto na estruturação como no reuso das informações envolvidas. Para que tal redução seja efetiva, é imprescindível o apoio automatizado de ferramentas para o estabelecimento de contratos eletrônicos.

Neste capítulo é apresentado o conjunto de ferramentas¹³ FeatureContract (*Feature Modeling based Web Services E-Contracts establishment toolkit*), desenvolvido para esse fim. O conjunto de ferramentas é considerado um protótipo por ainda não possuir todo o conjunto desejado de funcionalidades implementado; não apresentando, por exemplo, uma completa integração entre as ferramentas e um mecanismo muito adequado para a interação com o usuário. Primeiramente, é apresentada uma visão geral de sua estrutura e da seqüência de execução de suas ferramentas, destacando a integração necessária entre elas. Em seguida, cada uma das ferramentas partes de sua arquitetura é apresentada com mais detalhes.

¹³ Do inglês: toolkit

5.1 Arquitetura e Interações do Conjunto de Ferramentas

O conjunto de ferramentas FeatureContract é composto por seis ferramentas. Cada uma dessas ferramentas é responsável por oferecer apoio automatizado a diferentes estágios do processo de estabelecimento de contratos eletrônicos. A Figura 18 e a Figura 19 apresentam, respectivamente, a arquitetura do conjunto de ferramentas e a seqüência de execução de seus diferentes componentes.

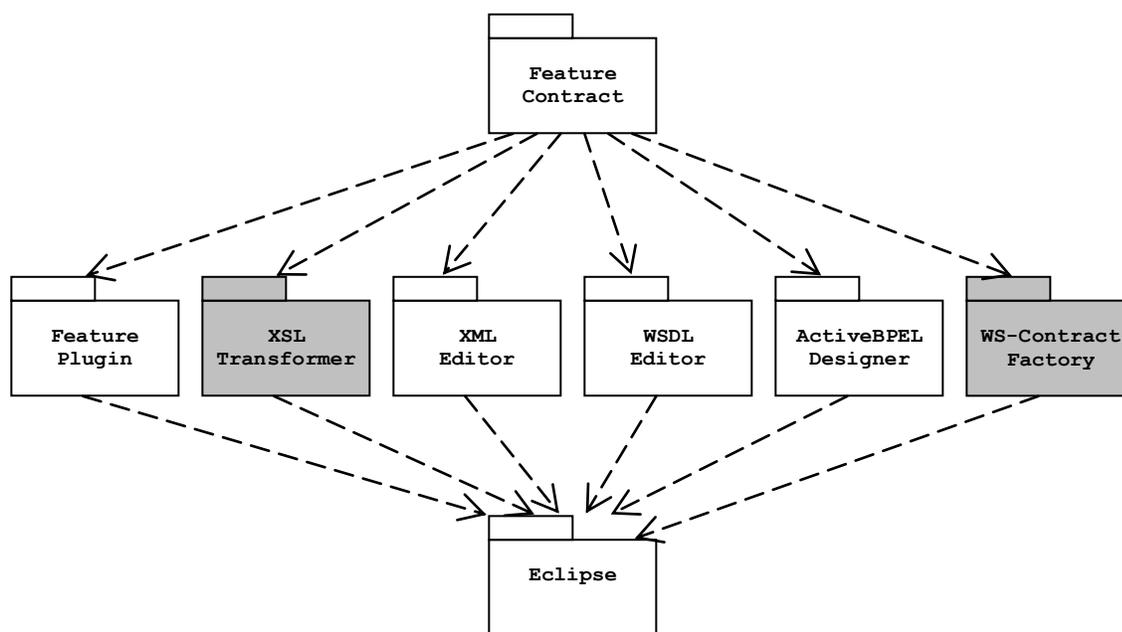


Figura 18: Arquitetura do conjunto de ferramentas FeatureContract.

A Figura 18 apresenta a arquitetura do conjunto de ferramentas FeatureContract por meio de um diagrama de pacotes da UML. FeatureContract atua como um ambiente para o estabelecimento de contratos eletrônicos, tendo como base a plataforma Eclipse (*Eclipse integrated development environment*) [94]. Cada um dos seis pacotes intermediários representa uma das ferramentas existentes. Algumas delas foram desenvolvidas por outros

grupos de pesquisa e desenvolvimento (FeaturePlugin [72], WSDL Editor [95], XML Editor [95] e ActiveBPEL Designer [76]); enquanto outras foram desenvolvidas especificamente como parte deste trabalho de tese: XSLTransformer e WS-Contract Factory (destacados em cinza na figura). A maioria delas foi desenvolvida como *plug-in*¹⁴ para a plataforma Eclipse.

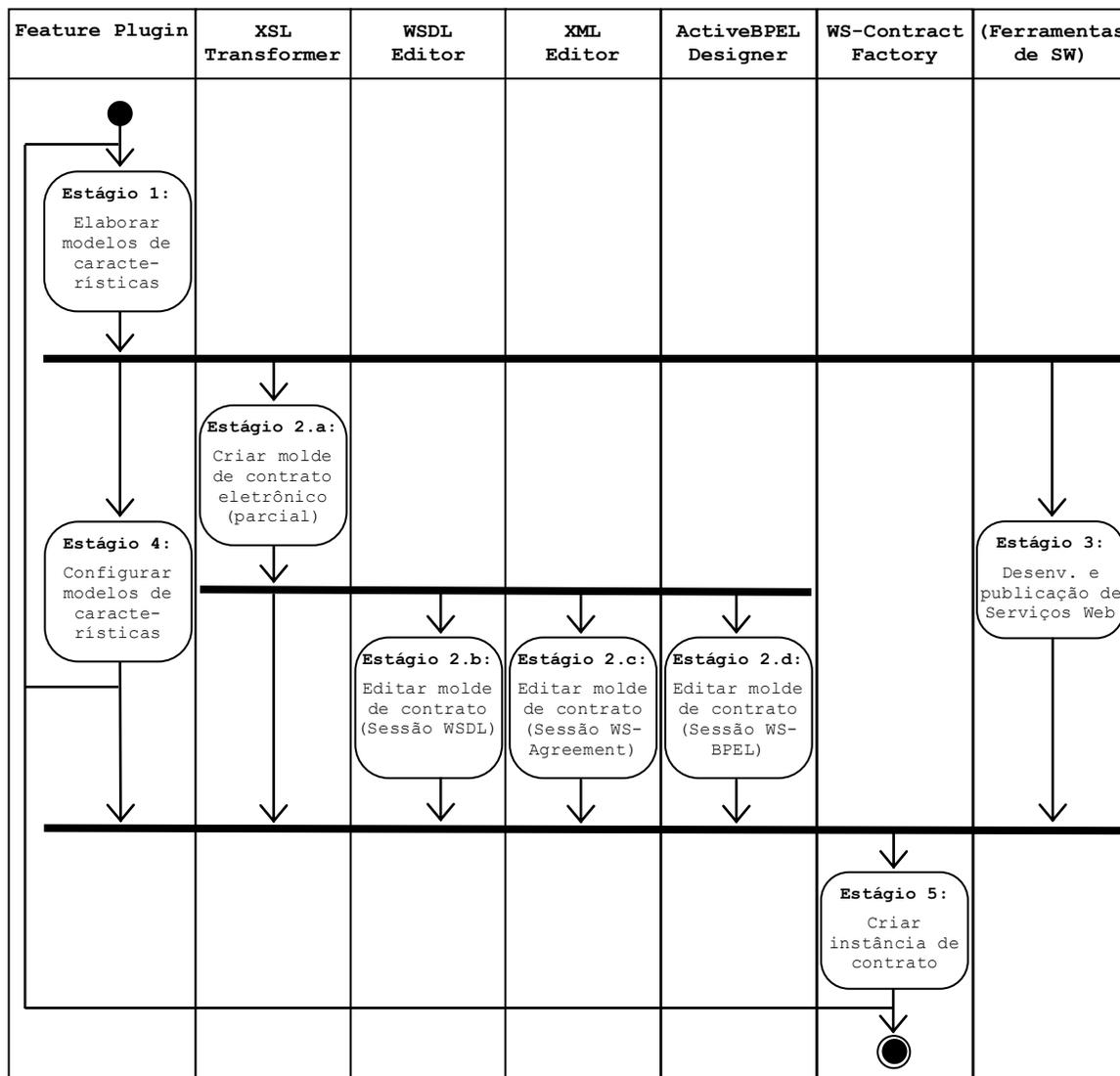


Figura 19: Seqüência de execução das ferramentas FeatureContract.

¹⁴ Termo usado na língua inglesa para representar uma ferramenta adicional a ser incorporada por uma outra.

A Figura 19 apresenta a seqüência de execução das ferramentas do conjunto FeatureContract por meio de um diagrama de atividades da UML. Cada raia do diagrama representa as atividades realizadas com o apoio de uma das ferramentas existentes no conjunto. A última raia representa as atividades relacionadas ao desenvolvimento e publicação de serviços Web, cujo detalhamento está fora do escopo desta tese, para as quais quaisquer ferramentas apropriadas podem ser usadas.

As atividades estão agrupadas em macro-atividades, correspondentes àquelas definidas no processo de estabelecimento de contratos no capítulo anterior. Apenas o Estágio 2 é dividido em 4 atividades (com as três últimas opcionais), por existirem quatro ferramentas envolvidas em sua completa execução. Cada uma das atividades produz um novo artefato usando como entrada os artefatos produzidos nas atividades imediatamente anteriores (caso das atividades associadas aos estágios 1, 2.a, 4 e 5), ou então apenas os altera (caso das atividades associadas aos estágios 2.b, 2.c e 2.d). Alguns fluxos de retorno são também mapeados, considerando que é previsto no processo que os artefatos produzidos pelos estágios 1 e 2 podem ser estendidos durante ou depois da execução dos estágios 4 e 5.

A comunicação entre as ferramentas é do tipo “integração por dados”. Deste modo, conforme descrito no parágrafo anterior, os artefatos produzidos por uma ferramenta são usados como entrada para uma outra ferramenta. Esta integração deve ser realizada de forma totalmente automática, porém por se tratar de um protótipo, algumas destas integrações ainda precisam de intervenção humana. Tais intervenções não impedem o uso do conjunto de ferramentas com o propósito de validação da automação da abordagem.

5.2 Ferramentas de FeatureContract

Nesta seção, é apresentada uma visão geral de cada uma das ferramentas partes do conjunto de ferramentas FeatureContract. Por simplicidade, exemplos genéricos são usados, ao invés dos exemplos específicos apresentados no capítulo do estudo de caso.

5.2.1 FeaturePlugin

A ferramenta FeaturePlugin oferece apoio à elaboração de modelos de características e suas configurações de acordo com as regras do meta-modelo de características apresentado na Seção 4.3.1. Desse modo, ele é usado em dois estágios do processo de estabelecimento de contratos eletrônicos. Essa ferramenta foi desenvolvida pelo grupo de pesquisa de Czarnecki et. al. [72], para o contexto genérico de engenharia de domínio baseado no modelo de características. FeaturePlugin foi desenvolvida como um *plug-in* para a plataforma Eclipse [94]. Aqui, ela foi incorporada no conjunto de ferramentas FeatureContract para ser aplicada para o contexto específico de contratos eletrônicos. Uma facilidade importante oferecida por essa ferramenta é a propriedade de exportar os modelos criados, bem como suas configurações, em especificações XML.

Nenhuma alteração no código e/ou na estrutura interna da ferramenta foi necessária. Apenas algumas extensões foram realizadas nas propriedades cobertas pelo meta-modelo de características usado por ela (descritos na Seção 4.3.1). A ferramenta já foi desenvolvida de forma a facilitar a realização desse tipo de adaptações.

Essa ferramenta oferece uma interface gráfica com o usuário para auxiliar na criação e manipulação de modelos de características e suas configurações. Esta interface gráfica permite que os modelos de características sejam editados e visualizados como árvores hierárquicas, similares a visualização oferecida por sistemas de gerenciamento de arquivos, com pastas, subpastas e arquivos. A ferramenta não oferece edição e visualização no formato de grafos, como o apresentado na Figura 05 (Capítulo 3), embora esteja planejado para futuras versões.

A Figura 20 apresenta os painéis principais da ferramenta. O painel principal apresenta o modelo de características de uma loja virtual (marcado entre as chaves superiores), bem como uma de suas configurações (marcado entre chaves na metade inferior). Embora o exemplo apresente apenas uma configuração, as diversas outras configurações também podem ser visualizadas nesse mesmo painel.

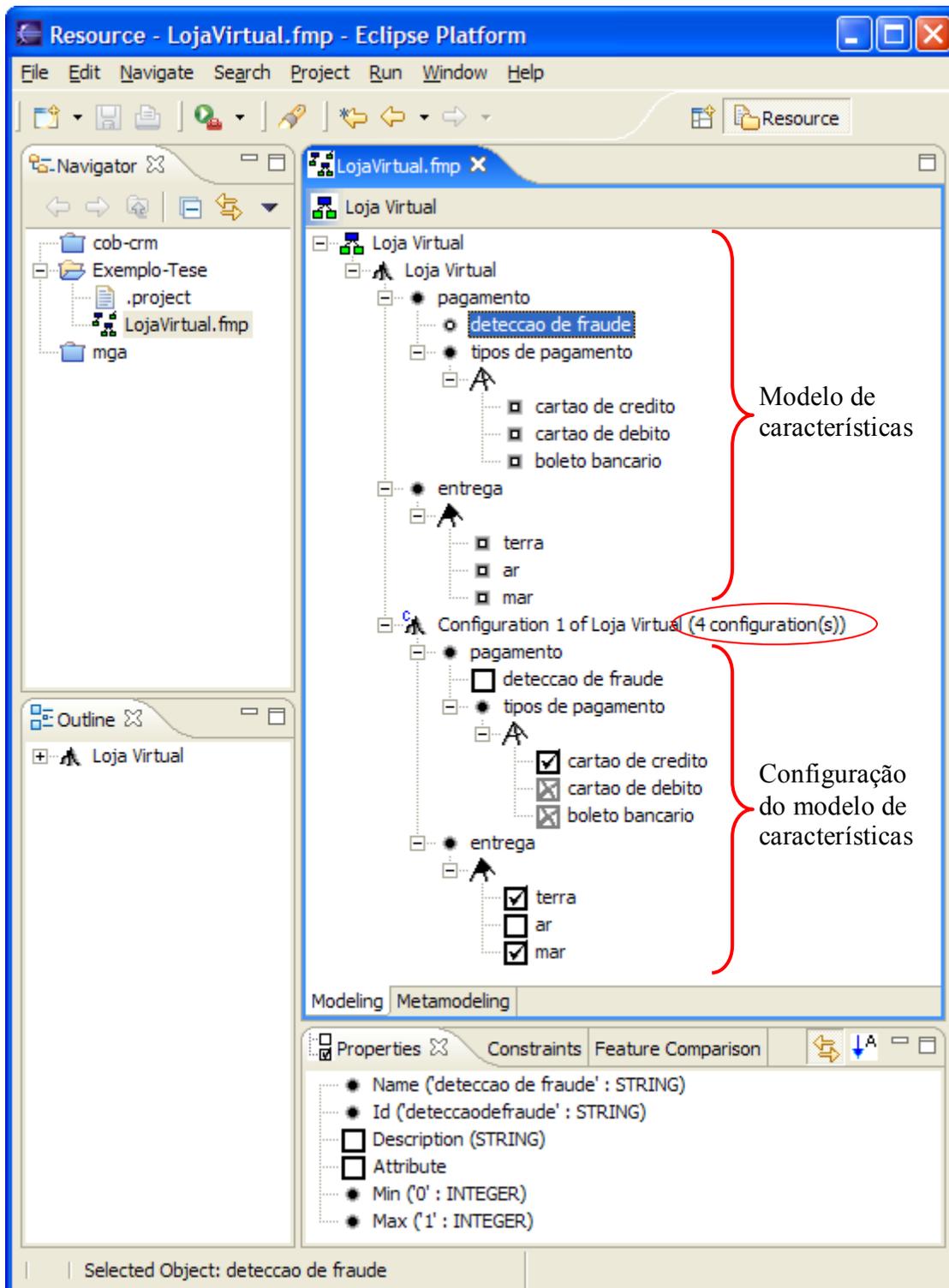


Figura 20: Ferramenta FeaturePlugin.

Os símbolos usados pela ferramenta possuem os seguintes significados, de acordo com o meta-modelo de características baseado em cardinalidade, apresentado na Seção 3.2.1:

- ▲ – característica raiz;
- – característica solitária obrigatória;
- – característica solitária opcional;
- ▲ – grupo de características (ou - inclusivo);
- ▲ – grupo de características (ou - exclusivo);
- – característica agrupada obrigatória;
- – característica agrupada opcional/alternativa;
- --> – referência para característica.

Os seguintes símbolos são usados na configuração de modelos de características:

- característica opcional/alternativa não selecionada;
- característica opcional/alternativa selecionada;
- característica opcional/alternativa que não pode ser selecionada devido a restrições de cardinalidade.

Na Figura 20, também é apresentado o painel Propriedades (*Properties*) no qual são apresentados os valores das propriedades relacionadas à característica **deteccao de fraude**, que é a selecionado no momento. Dependendo do tipo de característica selecionada, diferentes propriedades são apresentadas. A maioria dessas propriedades está relacionada aos elementos do meta-modelo de características baseado em cardinalidade. Para o exemplo dado, são apresentadas as seguintes propriedades:

- **Name:** contém o nome da característica;
- **Id:** contém um identificador único para a característica, gerado automaticamente;

- **Description:** contém uma descrição opcional para a característica, em complemento a seu nome;
- **Attribute:** contém um atributo, com tipo, para a característica;
- **Min:** contém o valor mínimo para ocorrências da respectiva característica em modelos de características configurados. O Valor “0” indica que a característica é opcional; qualquer valor maior que “0” indica que ela é obrigatória;
- **Max:** contém o valor máximo para ocorrências da respectiva característica em modelos de características configurados.

Além dessas propriedades que são padrão da ferramenta, outras propriedades podem ser definidas pelo usuário. Para a incorporação da FeaturePlugin como parte do conjunto de ferramentas FeatureContract, as seguintes propriedades foram incluídas:

- **Metric:** propriedade adicional para representar a métrica usada para uma característica do tipo atributo de QoS, como, por exemplo: tempo e quantidade. O uso dessa propriedade é apresentado na Seção 4.3;
- **Unit:** propriedade adicional para representar a unidade usada para uma característica do tipo atributo de QoS, como, por exemplo: segundos e horas (para a métrica “tempo”) e acessos (para a métrica “quantidade”). O uso dessa propriedade é apresentado na Seção 4.3.

Para cada configuração existente, a ferramenta FeaturePlugin apresenta as possíveis combinações para a configuração das características opcionais/alternativas. Por exemplo, na configuração apresentada na Figura 20, existem ainda 4 combinações possíveis para a configuração das características opcionais/alternativas ainda não selecionadas.

Na Figura 21, é apresentado um outro painel de apoio – o painel Restrições (*Constraints*). Esse painel é usado para a edição de restrições adicionais de características,

em XPath [71], conforme apresentado na Seção 3.2.2. Na figura, são apresentados, como ilustração, os dois exemplos de restrições adicionais de características que foram apresentados também na Seção 3.2.2.

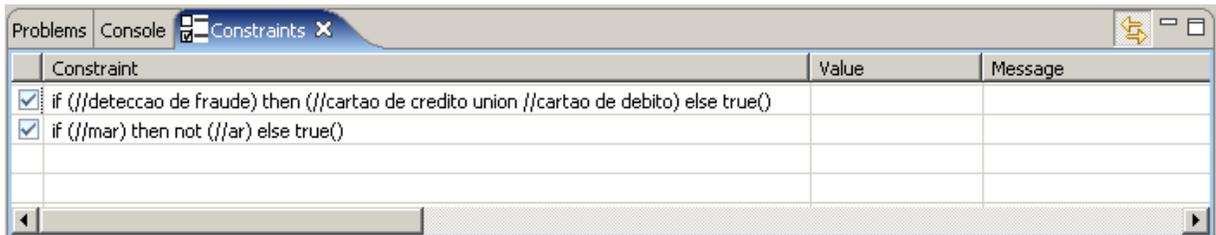


Figura 21: Painel de Restrições.

A ferramenta oferece ainda mecanismos para a avaliação de restrições, tanto das que já fazem parte do próprio meta-modelo – tais como restrições de cardinalidade de grupos, como das restrições adicionais expressas em XPath. Por meio desses mecanismos, um determinado modelo de características configurado pode ter as restrições adicionais verificadas. Esse mecanismo de validação de restrições leva em consideração apenas modelos de características configurados e não os modelos de características.

5.2.2 XSLTransformer

A ferramenta XSLTransformer realiza automaticamente a transformação de um par de modelos de características para serviços eletrônicos em uma versão inicial de molde de contrato eletrônico, que precisa ser completado posteriormente. Essa ferramenta foi desenvolvida pelo grupo de pesquisa do qual o autor desta tese faz parte, especificamente para fazer parte do conjunto de ferramentas FeatureContract.

XSLTransformer foi desenvolvido na linguagem XSLT [77]. Essa é uma linguagem usada especificamente para a transformação de documentos XML em documentos XML

os quais seguem regras de determinados esquemas XML. Qualquer interpretador XML pode ser usado para a execução de um código escrito em XSLT, e conseqüentemente a ferramenta XSLTransformer.

O XSLTransformer não possui interação com o usuário. Ele atua como um processador de documentos, tendo os seguintes artefatos de entrada e saída:

- **Entrada:** um arquivo no formato XML, produzido pela ferramenta FeaturePlugin, que representa os dois modelos de características das duas organizações;
- **Saída:** quatro arquivos também no formato XML, representando a versão inicial de um molde de contrato eletrônico para serviços Web. São criados dois arquivos para cada organização envolvida no contrato – um contendo a seção de definições WSDL e outro contendo a seção de termos de WS-Agreement.

Essa transformação é realizada seguindo as regras definidas na Seção 4.4.2. Conforme as regras de mapeamento apresentadas naquela seção, essa transformação gera um molde de contrato eletrônico que mantém entre seus elementos a mesma estrutura hierárquica existente entre as características no modelo de características.

Durante a transformação, XSLTransformer cria ainda anotações que permitem a rastreabilidade entre as diferentes características, usadas como entrada, e os elementos no molde de contrato produzido. Essa rastreabilidade é uma forma de, indiretamente, definir quais partes do molde de contrato são obrigatórias e/ou opcionais/ /alternativas, dependendo das características com a qual ela está ligada por meio do mecanismo de rastreabilidade. Ela é necessária para que o Estágio 5 do processo de estabelecimento de contratos possa ser realizado posteriormente (ver Seção 4.4 e Seção 4.6).

As anotações são feitas usando comentários da linguagem XML que ligam os elementos dos modelos de características para serviços eletrônicos aos elementos do molde de contrato eletrônico derivado. A ligação com uma determinada característica é feita por

meio de seu identificador único, uma de suas propriedades – gerada automaticamente pela ferramenta FeaturePlugin, como apresentado na seção anterior.

No Código 07 é apresentado, como exemplo, um trecho de um molde de contrato eletrônico gerado pela ferramenta XSLTransformer. As anotações são criadas com o formato `<!--FEATURE_ID = (ID da característica)-->`, onde (ID da característica) é o identificador único da característica. Existem três comandos aninhados nesse trecho de molde de contrato, cada um deles relacionado a uma característica diferente.

Código 07: Exemplos de anotação no molde de contrato eletrônico.

```
<!--FEATURE_ID = acoesdecobranca-->
<portType name="acoes-de-cobrancaPT">
  <!--FEATURE_ID = aplicacaodeacaodecobranca-->
  <operation name="aplicacao-de-acao-de-cobranca">
    <!--FEATURE_ID = tiposdeacaodecobranca-->
    <input message="tns:tipos-de-acao-de-cobranca" />
  </operation>

  <!--FEATURE_ID = atualizacaodeestadodeacaodecobranca-->
  <operation name="atualizacao-de-estado-de-acao-de-cobranca">
  </operation>

  <!--FEATURE_ID = cancelamentodeaplicacaodeacaodecobranca-->
  <operation name="cancelamento-de-aplicacao-de-acao-de-cobranca">
  </operation>
</portType>
```

Esse trecho exemplo de molde contrato está associado às características do modelo de características apresentado como exemplo na Seção 4.3 (Figura 12). Por exemplo, o comentário `<!--FEATURE_ID = acoesdecobranca-->` associa o elemento do molde `<portType name="acoes-de-cobrancaPT">` (um elemento tipo de porta) à característica **acoes-de-cobranca** (uma característica do tipo grupo de serviços), cujo identificador gerado automaticamente pela ferramenta FeaturePlugin é `acoesdecobranca`. O mesmo ocorre com os demais elementos desse exemplo, tais como:

- O comentário `<!--FEATURE_ID = aplicacaodeacaodecobranca-->` que associa `<operation name="aplicacao-de-acao-de-cobranca">` (um elemento operação), a **aplicacao-de-acoes-de-cobranca** (uma característica do tipo serviço eletrônico), cujo identificador gerado automaticamente pela ferramenta FeaturePlugin é `aplicacaodeacaodecobranca`; e
- O comentário `<input message="tns:tipos-de-acao-de-cobranca" />` que associa `<!--FEATURE_ID = tiposdeacaodecobranca-->` (um elemento tipo de mensagem), a **tipos-de-acao-de-cobranca** (uma característica do tipo propriedade de serviço), cujo identificador gerado automaticamente pela ferramenta FeaturePlugin é `tiposdeacaodecobranca`.

5.2.3 WSDL Editor

A ferramenta WSDL Editor oferece apoio à criação de novas especificações WSDL ou à edição de arquivos já existentes com conteúdo especificado na linguagem WSDL. WSDL Editor é um *plug-in* desenvolvido para a plataforma Eclipse. Essa ferramenta é fornecida como parte do conjunto de ferramentas chamado de Eclipse Web Tools Platform (WTP) [95]. A edição de especificações WSDL pode ser realizada tanto na forma gráfica (estruturada em tabelas) como na forma textual.

O WSDL Editor é usado nesse contexto para permitir a edição dos arquivos que contém as seções de definições WSDL do molde de contrato. A ferramenta XSLTransformer gera, parcialmente, as seções de definições WSDL do molde de contrato. Apenas as partes que dependem das informações contidas nos modelos de características são geradas automaticamente. Dependendo do contexto, pode ser necessário complementar os arquivos gerados com dados adicionais ou alterar os elementos gerados automaticamente para expandi-los. Exemplos de elementos da seção de definições WSDL, do molde de contrato,

que não são gerados automaticamente são: tipos de dados complexos, partes adicionais de tipos de mensagens, mensagens adicionais de entrada e saídas para operações.

Na Figura 22 é apresentado o painel principal da ferramenta WSDL Editor em que um arquivo WSDL com dois tipos de porta (*Port Types*), o primeiro deles com uma operação e o segundo com duas: **customPT – receiveCustomOrder**, e **ubiPT – receiveUBLOrder** e **receiveUBLOrder Custom**; respectivamente. Cada uma das três operações possui duas mensagens – uma de entrada e uma de saída.

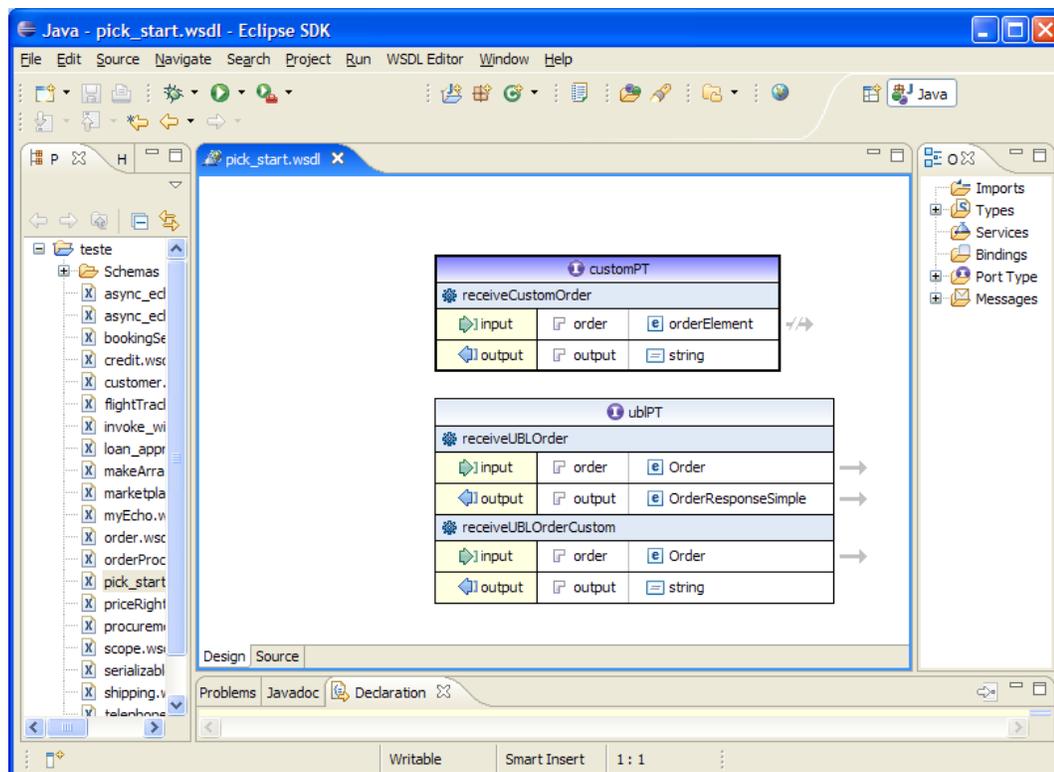


Figura 22: Ferramenta WSDL Editor.

5.2.4 XML Editor

A ferramenta XML Editor oferece apoio à criação de novas especificações XML ou à edição de arquivos já existentes. XML Editor é outro *plug-in* para a plataforma Eclipse,

também fornecido como parte do conjunto de ferramentas Eclipse Web Tools Platform (WTP) [95]. Ele é um editor genérico para especificações XML, usado para oferecer apoio na criação de novas especificações ou para editar arquivos já existentes com conteúdo especificado na linguagem XML. A edição de especificações XML pode ser realizada tanto na forma gráfica (estruturada em árvores hierárquicas) como na forma textual.

Ele é usado nesse contexto para permitir a edição dos arquivos que contém as seções de termos de WS-Agreement do molde de contrato. Similarmente ao caso anterior, as seções de termos de WS-Agreement do molde de contrato são geradas parcialmente. Como os arquivos relativos às seções de definições WSDL, esses arquivos podem ser editados para que novas informações sejam acrescentadas ou informações existentes sejam alteradas.

A Figura 23 apresenta o painel principal da ferramenta XML Editor. O exemplo refere-se a um arquivo WS-Agreement com um elemento relacionado a propriedades de serviço (`wsag:ServiceProperties`) e a termos de garantia (`wsag:GuaranteeTerm`). O segundo desses possui um objetivo de nível de serviço (`wsag:ServiceLevelObjective`) atribuído com **tempo-de-resposta IS_LESS-INCLUSIVE 5**.

5.2.5 ActiveBPEL Designer

A ferramenta ActiveBPEL Designer oferece apoio à criação de novos processos de negócio especificados na linguagem WS-BPEL ou à edição de arquivos já existentes com conteúdo especificado nessa linguagem. Ele é usado nesse contexto para permitir a criação da seção relativa ao processo de negócio do molde de contrato eletrônico.

Durante a geração do arquivo inicial referente ao molde de contrato, nada é gerado em relação ao processo de negócio envolvendo os sistemas das duas organizações. Conforme exposto na apresentação da abordagem, as informações necessárias para a elaboração do processo de negócio não fazem parte do contexto dos modelos de características.

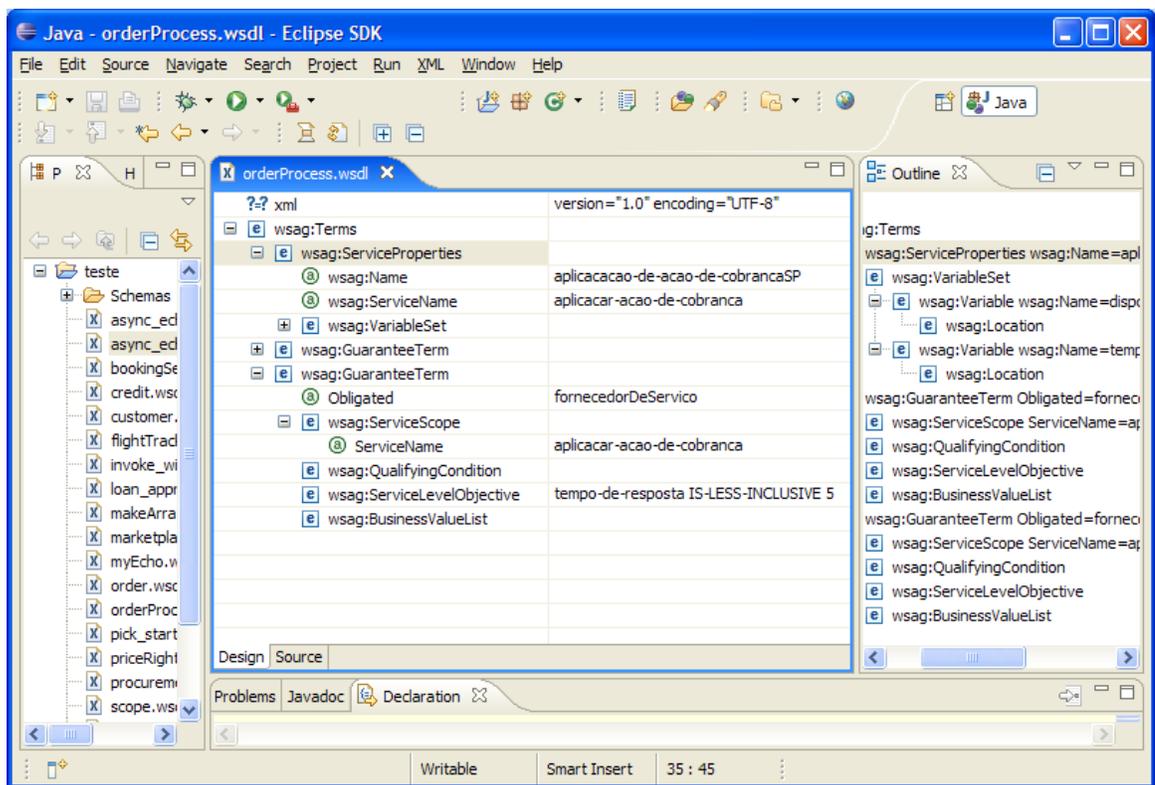


Figura 23: Ferramenta XML Editor.

A ferramenta ActiveBPEL Designer é também um *plug-in* desenvolvido para a plataforma Eclipse. Ele permite a edição de especificações WS-BPEL tanto na forma gráfica (estruturada na forma de grafos) como na forma textual. Como um processo de negócio WS-BPEL é baseado em serviços Web especificados por meio da linguagem WSDL, essa ferramenta usa como entrada os arquivos produzidos pela ferramenta XSLTransformer. Apenas o conteúdo dos arquivos referentes às seções de definições WSDL são usados e não os arquivos contendo termos de WS-Agreement.

Na Figura 24 é apresentado o painel principal da ferramenta ActiveBPEL Designer que contém um exemplo de processo de negócio WS-BPEL consistindo de uma seqüência de atividades, composta por: primeiramente, duas atividades paralelas do tipo *receive*; seguidas por duas atividades alternativas do tipo *assign*; e finalizadas com uma subseqüência de duas atividades do tipo *reply*.

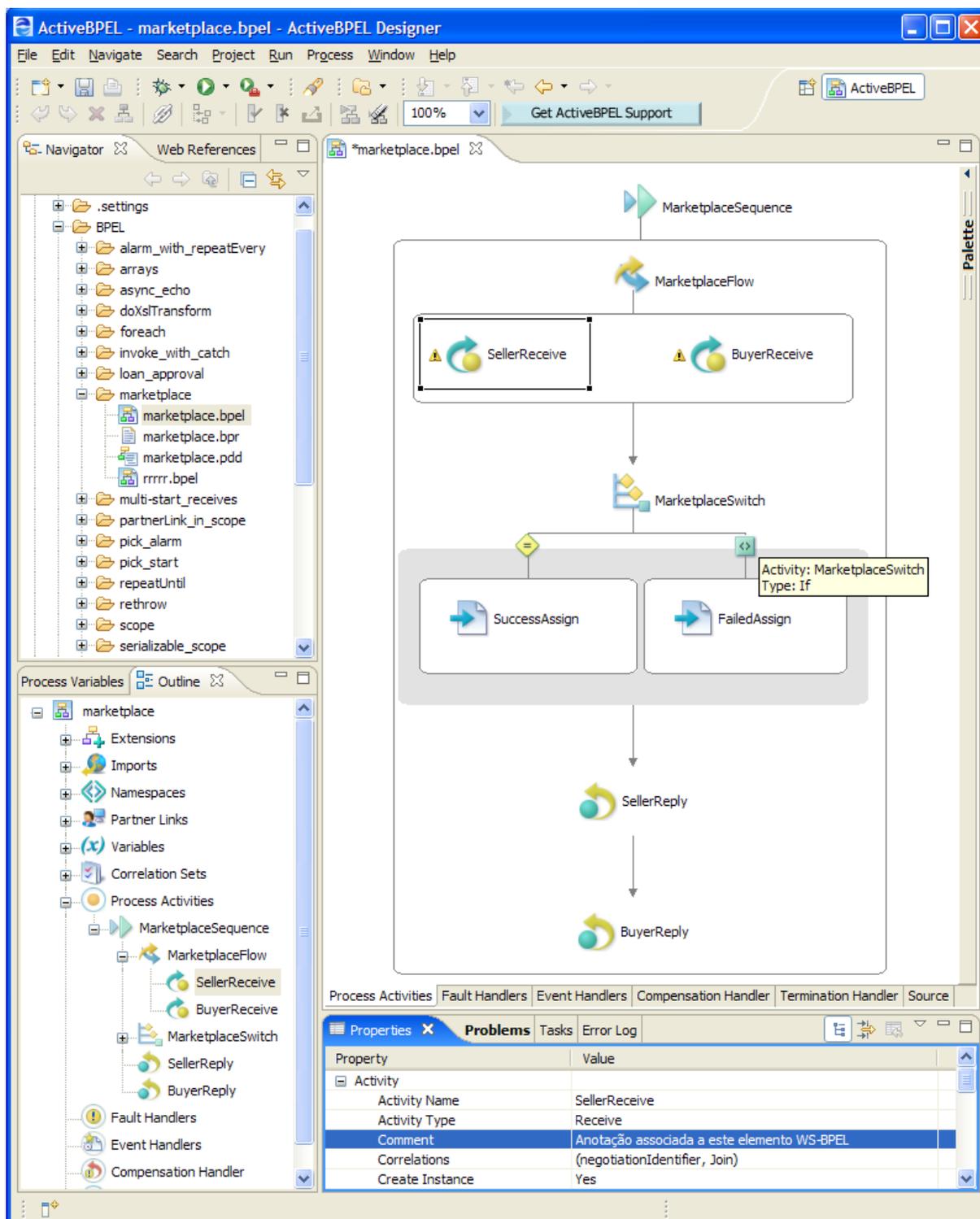


Figura 24: Ferramenta ActiveBPEL Designer.

Além da representação visual, a ferramenta disponibiliza também a especificação WS-BPEL (chamada de código-fonte) a qual é gerada automaticamente em função das ações do usuário. Além do painel principal, existem painéis de apoio à elaboração dos processos de negócio.

Além da elaboração de processos de negócio, essa ferramenta possibilita que o código WS-BPEL receba anotações na forma de comentários XML. Elas são necessárias para que cada elemento da especificação possua sua rastreabilidade definida com as respectivas características dos modelos de características para serviços eletrônicos. Trata-se do mesmo tipo de anotações existente para os demais arquivos que compõem o molde de contrato eletrônico. A ferramenta ActiveBPEL Designer disponibiliza um mecanismo para que as anotações sejam realizadas sem necessidade de edição diretamente no código-fonte.

Ao invés disso, as anotações são realizadas pelo usuário por meio do painel de propriedades (parte inferior da Figura 24). No exemplo, como a atividade **SellerReceive** está selecionada no painel principal, são apresentadas suas propriedades na parte inferior, com o item comentário destacado. Para cada nova atividade criada no processo de negócio, é necessário que o usuário insira um comentário associando-o com a respectiva característica do tipo *serviço eletrônico* – a mesma que já está associada com a operação da especificação WSDL usada na criação da atividade. Similarmente à especificação WSDL, todas as anotações devem ser criadas usando o formato **<!-- FEATURE_ID = (ID da característica)-->**, onde (ID da característica) é o identificador único da característica.

5.2.6 WS-Contract Factory

A ferramenta WS-Contract Factory realiza automaticamente a criação de uma instância de contrato eletrônico para serviços Web, tendo como entrada o molde de contrato e o par de modelos de características para serviços eletrônicos configurados. Essa ferramenta foi

desenvolvida pelo grupo de pesquisa do qual o autor dessa tese faz parte, especificamente para fazer parte do conjunto de ferramentas FeatureContract.

Essa é uma ferramenta que não possui interação com o usuário. Ele atua como um processador de documentos, tendo os seguintes artefatos de entrada e saída:

- **Entrada:** cinco arquivos no formato XML, representando o molde completo de contrato, conforme segue: um arquivo contendo a seção de definições WSDL e um contendo a seção de termos de WS-Agreement para uma das organizações; um arquivo contendo a seção de definições WSDL e um contendo a seção de termos de WS-Agreement para a outra organização; e um arquivo contendo a seção de processo de negócio WS-BPEL para as duas organizações envolvidas no contrato;
- **Saída:** cinco arquivos no formato XML – equivalentes aos arquivos de entrada, porém representando o contrato eletrônico resultante. Cada um desses arquivos possui um subconjunto das informações contidas nos cinco arquivos de entrada.

A criação da instância de contrato eletrônico é realizada com base na análise das anotações existentes em todos os arquivos do molde de contrato eletrônico. Cada elemento do molde de contrato associado, via anotação, a alguma característica que não tenha sido selecionada durante a configuração dos modelos de características deve ser removida do artefato em que ela se encontra. Além disso, todas as anotações são também removidas do molde de contrato, mesmo aquelas relacionadas a elementos que foram mantidas no contrato eletrônico resultante.

WS-Contract Factory foi desenvolvido na linguagem Java. Seu algoritmo principal faz a manipulação do arquivo de molde de contrato com o apoio de uma pilha como estrutura de dados. A pilha é usada para manter controle do nível de indentação dentro de elementos de especificação XML durante a varredura do arquivo. Dependendo das características associadas, tendo sido selecionadas ou não, a decisão de repassar para o arquivo de contrato final é tomada. As informações empilhadas na estrutura de dados são usadas

durante seu desempilhamento para se decidir se um trecho de especificação XML em um nível anterior de indentação está associado a uma característica selecionada ou não, cuja decisão foi tomada ao se passar nesse mesmo nível pela primeira vez antes de descer um nível na estrutura aninhada.

Abrangência de remoção

Para simplificar o processo de remoção de elementos no molde de contrato eletrônico, e assumindo que a estrutura hierárquica do molde de contrato é a mesma que a existente nos modelos de características, então uma regra relacionada à abrangência de remoção é utilizada.

A abrangência de remoção para cada elemento do molde de contrato eletrônico é definida levando-se em consideração a estrutura hierárquica dos elementos no molde de contrato. Para os elementos do molde de contrato que agrupa outros elementos abaixo dele, dentro de sua estrutura hierárquica, se ele tiver que ser removido, então todos os elementos dentro de sua estrutura hierárquica também o são. O contrário não é necessariamente verdadeiro.

Como uma característica não selecionada implica que todas suas sub-características também necessariamente não o são, então o mesmo se aplica aos elementos e sub-elementos do molde de contrato. Se essa regra de abrangência de remoção não fosse aplicada, os respectivos elementos acabariam sendo removidos da mesma forma, visto que nos graus mais baixos da hierarquia, eles estariam associados a características também não selecionadas. Porém, essa regra é aplicada para simplificar o processo de remoção.

Como um exemplo, na Figura 07 apresentada anteriormente: se o elemento `<portType name="acoes-de-cobrancaPT">` tiver que ser removido, dado que sua respectiva característica, cujo identificador está na anotação `<!--FEATURE_ID = acoesdecobranca-->`, não foi selecionada; então todos os elementos aninhados a esse elemento mais externo, incluindo – por exemplo, os elementos `<operation`

`name="aplicacao-de-acao-de-cobranca">` e `<input message="tns:tipos-de-acao-de-cobranca"/>` devem ser automaticamente removidos. Certamente as características associadas a esses elementos mais internos, cujos identificadores estão respectivamente nas anotações `<!--FEATURE_ID = aplicacao deacaodecobranca-->` e `<!--FEATURE_ID = tiposdeacaodecobranca-->`, também não foram selecionadas, dada que a característica pai não o foi, o que justifica a remoção adiantada de tais elementos internos.

Especificamente no caso do processo de negócio especificado em WS-BPEL é possível que todas as atividades básicas, dentro de uma atividade, estrutura sejam removidas. Nesse caso, esses elementos mais externos são removidos também – em qualquer nível da hierarquia de atividades. Por exemplo, se todas as atividades básicas (tais como *invoke*, *receive*, *reply*) dentro uma atividade estruturada (tal como *flow*) forem removidas, essa também o é.

5.3 Considerações Finais

Neste capítulo foi apresentado o conjunto de ferramentas FeatureContract, desenvolvida pra oferecer apoio automatizado à abordagem proposta nesta tese. FeatureContract consiste de um conjunto de ferramentas que juntas cobrem todos os estágios definidos para o processo de estabelecimento de contratos eletrônicos para serviços Web com base em modelos de características. Algumas dessas ferramentas foram desenvolvidas especificamente para serem parte do conjunto de ferramentas FeatureContract, enquanto que as demais já existiam anteriormente e foram apenas incorporadas (com algumas adaptações) no conjunto de ferramentas. Dessas ferramentas pré-existentes, todas são *plugins* do ambiente Eclipse.

A aplicação da abordagem não é uma atividade completamente automática. Grande parte dos passos corresponde a atividades realizadas por humanos, apenas com o apoio de

ferramentas apropriadas. Entretanto, o uso do conjunto de ferramentas disponibilizado, em conjunto com as regras definidas dentro da abordagem proposta, garante a aplicação sistemática de técnicas de estruturação e reuso de informações no referido contexto. As ferramentas partes do conjunto de ferramentas FeatureContract podem ainda ter suas funcionalidades expandidas para que o apoio automatizado dado à realização da abordagem proposta seja ainda maior.

O conjunto de ferramentas desenvolvido foi exercitado na realização de um estudo de caso com o objetivo de avaliar a abordagem proposta. Esse estudo de caso é apresentado no próximo capítulo. Exemplos parciais de todos os artefatos consumidos/produzidos pelo conjunto de ferramentas FeatureContract são apresentados como resultados do estudo caso.

Capítulo 6

Estudo de Caso

Neste capítulo é apresentado um estudo de caso realizado com o objetivo principal de demonstrar a aplicabilidade da abordagem apresentada no Capítulo 4. A demonstração de tal aplicabilidade é realizada por meio da execução da abordagem proposta aqui, em que foram gerados os artefatos previstos de acordo com as definições apresentadas. O estudo de caso tem também o objetivo de apresentar exemplos mais detalhados dos artefatos produzidos durante a execução do processo de estabelecimento de contrato, em relação àqueles apresentados junto com a abordagem proposta.

Primeiramente é apresentado o domínio da aplicação para o qual o estudo de caso foi realizado. Em seguida, são apresentados exemplos detalhados dos artefatos produzidos com o apoio do conjunto de ferramentas FeatureContract. Os artefatos completos são apresentados nos Apêndice 1 e Apêndice 2. Na seqüência, é realizada uma análise do grau de variabilidade existente nos modelos de características desse estudo de caso. Por fim, é apresentada uma análise dos resultados obtidos com a realização do estudo de caso, destacando vantagens e benefícios e também incluindo algumas desvantagens e limitações.

6.1 Domínio da Aplicação

O estudo de caso teve como objeto de estudo a integração entre dois sistemas de apoio a negócios e operações, conhecidos como sistemas BOSS (*Business and Operation Support*

Systems) [96], no contexto de empresas operadoras de telecomunicações. Esse objeto de estudo foi escolhido devido ao grande potencial para seu tratamento como processos de negócio interorganizacionais, em que uma empresa operadora de telecomunicações subcontrata outras empresas para oferecerem determinados serviços especializados de negócios ou operações. Essa abordagem pode ser usada pelas empresas operadoras por não terem domínio de conhecimento ou, então, para a diminuição de custos.

Dentro do contexto de sistemas BOSS para telecomunicações, o estudo de caso considera a integração entre os seguintes sistemas:

- **Sistema de Atendimento a Clientes (CRM – *Customer Relationship Management*)**: um Sistema CRM gerencia o relacionamento entre uma empresa operadora de telecomunicações e seus clientes. Esse tipo de sistema deve oferecer um conjunto de serviços eletrônicos, incluindo, por exemplo: venda de produtos e serviços para clientes; criação de contratos entre a empresa e clientes; registro, consulta e cancelamento de ordens de serviço; criação e atualização de cadastro de clientes; e, envio de comunicações a clientes. A execução de um Sistema CRM requer a integração com outros sistemas BOSS para telecomunicações, tais como: gerência de recursos, gerência de planta externa, gerência de força de trabalho, tarifação de serviços (usados pelos clientes), faturamento de serviços (usados pelos clientes), arrecadação, cobrança e contabilização.
- **Sistema de Cobrança (COB)**: um Sistema COB oferece apoio computacional à cobrança de débitos de clientes de uma empresa operadora de telecomunicações. Esse tipo de sistema deve oferecer um conjunto de serviços eletrônicos, incluindo, por exemplo: aplicação, atualização e cancelamento de ações de cobrança; aplicação, atualização e cancelamento de reversões de ação de cobrança; inclusão e exclusão de registros de cheques irregulares; parcelamento e cancelamento de débitos de clientes; cancelamento de encargos (tais como multas e juros); e

concessão de descontos para pagamento de dívidas. Com relação às ações de cobrança, existem diferentes tipos de ação que podem ser aplicadas, tais como: notificação de débito (por correspondência normal, por correspondência eletrônica ou por telefone), suspensão do fornecimento de serviços contratados (podendo ser uma suspensão parcial ou total) e ações legais contra o cliente em débito.

O estudo de caso considerou que uma empresa operadora de telecomunicações, que opera um Sistema CRM, contrata os serviços de cobrança de débitos de outra empresa especializada em cobrança, que opera um Sistema COB, criando assim um processo de negócio interorganizacional. Cada sistema oferece um conjunto de serviços eletrônicos – implementados e disponibilizados como serviços Web, para serem usados pelo outro sistema. Um contrato eletrônico é estabelecido para definir os detalhes relacionados ao acordo de negócio entre ambas as organizações.

Nesse processo de negócio, ambos os sistemas precisam disponibilizar serviços eletrônicos um para o outro – e não apenas o Sistema COB para o Sistema CRM. Para que determinadas tarefas a serem executadas pelo Sistema COB possam ser finalizadas, pode ser necessária a execução de serviços pelo Sistema CRM, além do próprio Sistema COB. A aplicação de uma ação de cobrança do tipo suspensão de fornecimento de serviço, por exemplo, deve ser feita do lado da empresa operadora de telecomunicações, por meio de uma ordem de serviço do Sistema CRM. Outros tipos de ação de cobrança, como a notificação de débito, por exemplo, podem ser executados inteiramente pelo Sistema COB.

A realização do estudo de caso ocorreu em um cenário fictício, e não em um conjunto de configurações usado por organizações reais. Isso se deve à complexidade do estudo de caso e a não identificação de empresas disponíveis para esse objetivo.

O estudo de caso realizado pode ser considerado de médio porte, por meio de uma análise subjetiva, devido ao número de serviços envolvidos e amplitude do processo de negócio envolvendo as duas organizações. Essa complexidade média pode ser verificada por meio dos artefatos gerados, apresentados na seqüência. Não foram encontrados

suficientes relatos de outros estudos de casos similares para que uma comparação mais objetiva pudesse ser realizada. A complexidade do estudo de caso poderia ter sido ainda maior, porém nem todos os possíveis serviços eletrônicos foram nele tratados, para limitar sua complexidade.

Este domínio de aplicação foi escolhido para este estudo de caso dado que o autor desta tese possui amplo conhecimento de informações relacionadas a ele. O autor trabalhou durante 4 anos em uma empresa de desenvolvimento de software para sistemas de telecomunicações, que desenvolvia uma série de sub-sistemas incluindo os dois usados neste estudo de caso, porém para uma abordagem de execução intra-organizacional. Em função do conhecimento existente pelo autor, este atuou no estudo de caso como o próprio especialista de domínio, fazendo uso de bibliografia especialista como apoio [96].

6.2 Artefatos Produzidos

Nesta seção são apresentados, como exemplos, partes dos artefatos produzidos pela aplicação do processo de estabelecimento de contratos eletrônicos. Os exemplos de artefatos são apresentados na ordem em que os estágios do processo foram realizados.

6.2.1 Modelos de Características para Serviços Eletrônicos

Nesta seção são apresentados os modelos de características produzidos no Estágio 1 do processo de estabelecimento de contratos, com o apoio da ferramenta FeaturePlugin. Os modelos de características representam informações das duas organizações. O modelo de características mostrado na Figura 25 e na Figura 26 contém, respectivamente, informações sobre os serviços eletrônicos e informações sobre os atributos de QoS oferecidos pelo Sistema CRM para o Sistema COB. O modelo de características mostrado

na Figura 27 e na Figura 28 contém os mesmos tipos de informações para o sentido inverso da cooperação interorganizacional – do Sistema COB para o Sistema CRM.

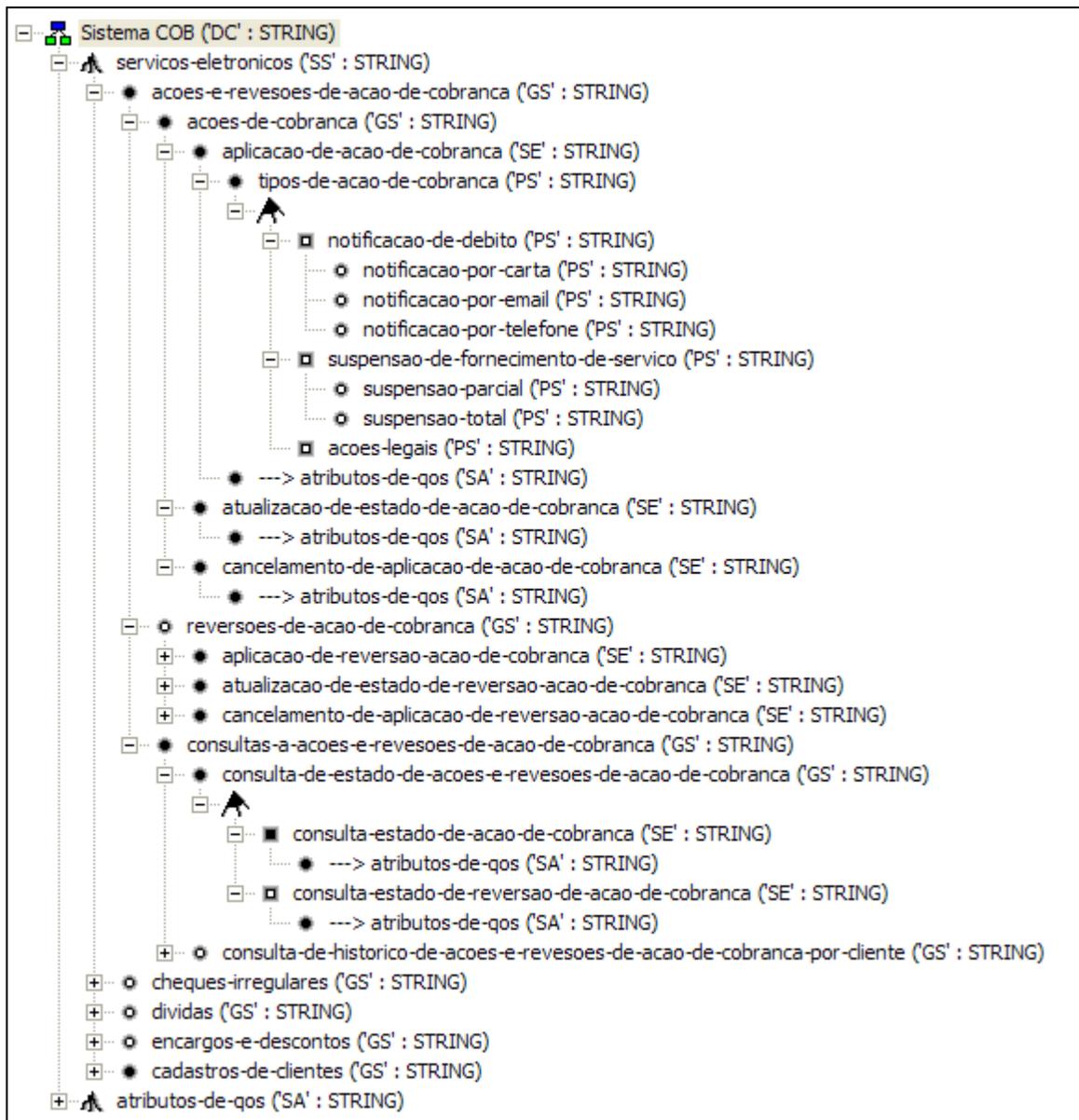


Figura 25: Modelo de características para serviços eletrônicos do Sistema COB (características relacionadas aos serviços eletrônicos).

Os modelos de características para serviços eletrônicos foram elaborados seguindo as regras definidas no meta-modelo de características baseado em cardinalidade, apresentado na Figura 06, Seção 3.2.1, e de acordo com os diferentes tipos de características e os relacionamentos hierárquicos existentes entre esses tipos, conforme definido no capítulo anterior (Seção 4.3.1, Figura 13).

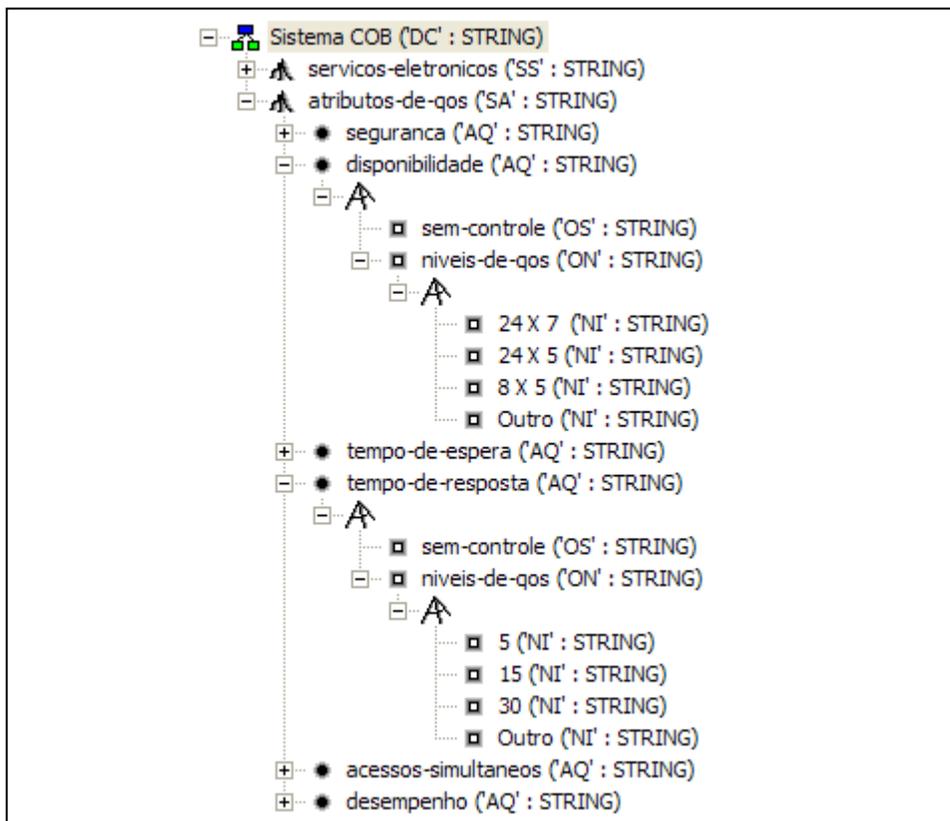


Figura 26: Modelo de características para serviços eletrônicos do Sistema COB (características relacionadas aos atributos de QoS).

Os dois modelos de características são: **Sistema COB** (Figura 25 e Figura 26) e **Sistema CRM** (Figura 27 e Figura 28). Os modelos de características são apresentados aqui apenas parcialmente, dada a inviabilidade da apresentação completa desses artefatos. O tamanho e/ou a complexidade desses dois modelos podem ser entendidas em função dos

seguintes exemplos de métricas associadas a eles: quantidade total de características para cada um dos modelos igual a 83 e 35, respectivamente; e profundidade máxima para cada um dos modelos igual a 12 e 10 sub-características, respectivamente. Ambos os modelos são apresentados por completo nos apêndices desta tese.

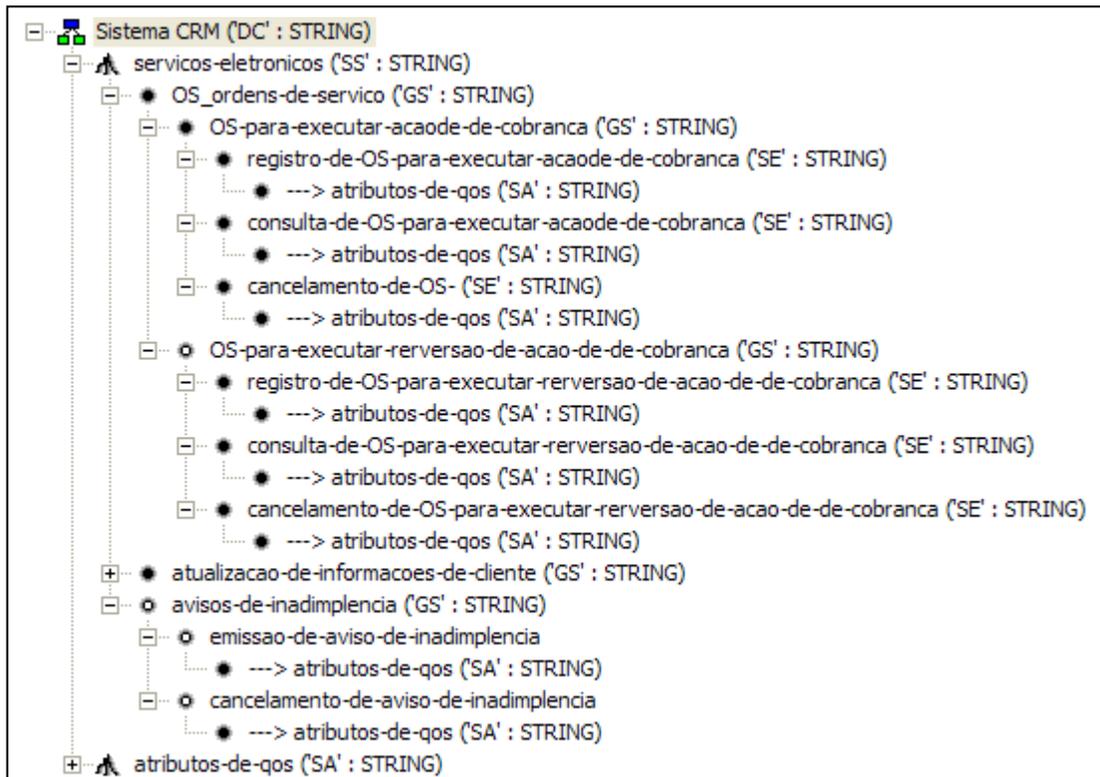


Figura 27: Modelo de características para serviços eletrônicos do Sistema CRM (características relacionadas aos serviços eletrônicos).

Cada um desses modelos possui um conjunto de sub-características que representam de uma forma estruturada os serviços eletrônicos e os atributos de QoS disponibilizados por cada um deles. Ambos os modelos possuem como raiz as características (pré-nomeadas) **servicos-eletronicos** e **atributos-de-qos** (do tipo sub-árvore). Essas características aparecem apenas uma vez em cada um dos modelos de características. Cada uma das

características existentes nesses modelos possui uma propriedade que representa o seu tipo no contexto específico de serviços eletrônicos e atributos de QoS, em função da estrutura apresentada na Figura 13. O conteúdo do atributo para cada característica é apresentado, no modelo, logo após o seu nome, no seguinte formato: ('XX': STRING), em que XX é a sigla do atributo de acordo com a Tabela 03. Exemplos de tipos de característica e siglas são: grupo de serviços (GS), serviço eletrônico (SE), atributo de QoS (AQ) e nível (NI).

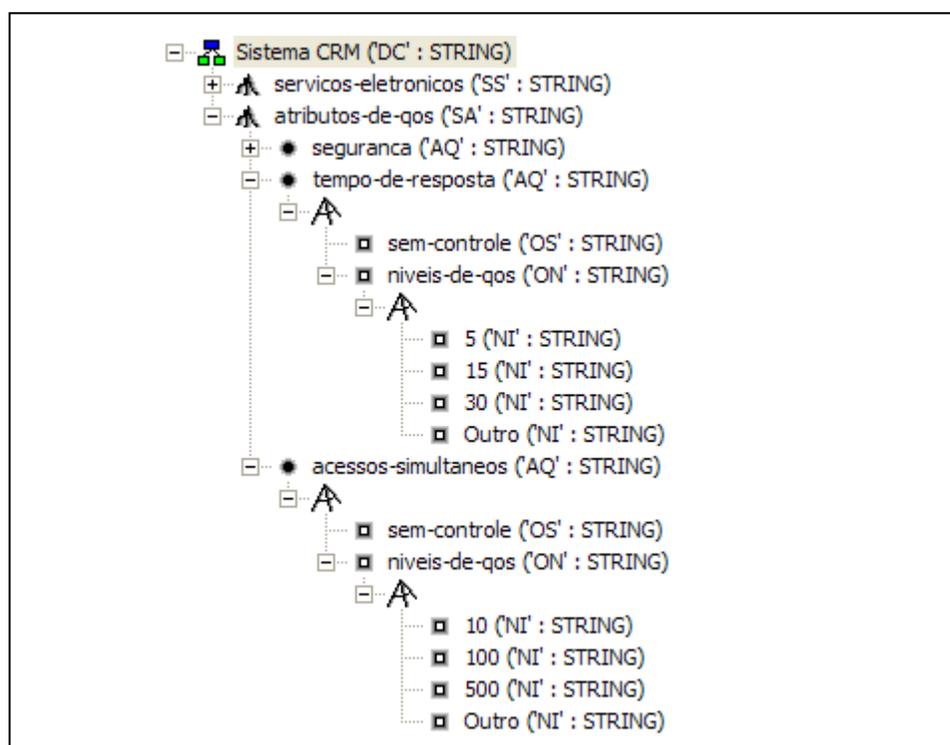


Figura 28: Modelo de características para serviços eletrônicos do Sistema CRM (características relacionadas aos atributos de QoS).

A seguir são apresentados exemplos dos demais tipos de características, em função do modelo de características apresentado na Figura 25 e na Figura 26 para o Sistema COB. Exemplos similares existem na Figura 27 e na Figura 28 para o Sistema CRM.

A característica raiz **servicos-eletronicos** contém as sub-características que representam agrupamentos estruturados de serviços eletrônicos oferecidos pelo sistema COB. Exemplos desse tipo de características são: **acoes-e-reversoes-de-acoes-de-cobranca**, **acoes-de-cobranca**, **reversoes-de-acao-de-cobranca**, **cheques-irregulares** e **encargos-e-descontos** (todas do tipo grupo de serviços).

De forma opcional, cada uma das características do tipo serviço eletrônico pode conter sub-características que representam detalhes funcionais associados ao respectivo serviço eletrônico. A característica **aplicacao-de-acao-de-cobranca**, por exemplo, contém as seguintes sub-características que representam detalhes funcionais relacionados à aplicação de ações de cobrança: **notificacao-de-debito** (incluindo **notificacao-por-carta**, **notificacao-por-email** e **notificacao-por-telefone**), **suspensao-de-fornecimento-de-servico** (incluindo **suspensao-parcial** e **suspensao-total**) e **acoes-legais** (todas do tipo propriedade de serviço).

Dentre todas as características existentes nos modelos apresentados anteriormente, algumas delas são definidas como características obrigatórias, por serem consideradas críticas para a execução do Sistema COB, ou por serem parte fundamental do negócio de cobrança ou por serem atividades de apoio necessárias. Desse modo, serviços eletrônicos relacionados diretamente à aplicação de ações de cobrança são definidos como obrigatórios, pois devem ser contratados. Já os serviços relacionados à reversão de aplicação de ações de cobrança não são definidos como obrigatórios, visto que a empresa operadora de telecomunicações pode ou não contratar esses serviços. Outros serviços relacionados a cheques irregulares, encargos e descontos, e dívidas são também definidos como opcionais. Existem grupos de serviços eletrônicos definidos como obrigatórios, enquanto que os serviços eletrônicos que fazem partes desses são opcionais. Além disso, existem grupos de serviços opcionais, enquanto todos os seus serviços componentes são obrigatórios. Nesse caso, se o grupo for selecionado, todos seus serviços também o são.

A característica raiz **atributos-de-qos** (Figura 26) contém uma série de sub-características representando atributos de QoS que podem ser associados a serviços

eletrônicos. Exemplos desse tipo de características são: **disponibilidade**, **tempo-de-resposta**, **acessos-simultaneos** e **seguranca** (todas do tipo atributo de QoS).

Cada uma dessas características do tipo atributo de QoS contém, por sua vez, duas sub-características (pré-nomeadas) **sem-controle** e **niveis-de-qos** (do tipo opção de atributo) representando possíveis opções de atributos de QoS. Para cada característica do tipo atributo de QoS, existe sempre um par dessas duas características do tipo atributo de QoS.

Cada uma das características **niveis-de-qos** contém, por sua vez, sub-características representando os possíveis níveis de QoS que podem ser escolhidos para cada serviço eletrônico associado a um atributo de QoS. Para a característica **tempo-de-resposta**, por exemplo, as seguintes características representam os possíveis níveis de QoS: **5**, **15**, **30** e **outro** (todas do tipo nível).

Embora o conjunto de características relacionadas aos atributos e níveis de QoS seja bastante similar para os dois modelos de características, o Sistema COB apresenta um conjunto maior de atributos de QoS do que o Sistema CRM. Além disso, diferentes níveis são previstos para um mesmo atributo de QoS – **tempo-de-resposta**, por exemplo. Essa diferença se deve ao fato de que a organização responsável pelo sistema CRM está subcontratando serviços da organização que disponibiliza o Sistema COB. Assim, é natural que ela exija um conjunto maior de atributos de QoS ou mesmo níveis mais precisos para determinados atributos.

As características do tipo atributo de QoS contém, além de seu nome e da propriedade que representa seu tipo, dois atributos opcionais – métrica e unidade, a serem medidos. Essas informações não são apresentadas no painel principal da ferramenta FeaturePlugin (visão gráfica do modelo), mas sim no painel de propriedades. Na Figura 29 é apresentado o painel de propriedades para a característica **tempo-de-resposta**. De acordo com a propriedade *Metric&Unit*, esse atributo de QoS deve ser medido como um atributo cuja métrica é “tempo” e cuja unidade é “segundo”. Desse modo, as opções **5**, **15**

e **30** se referem a **5** segundos, **15** segundos e **30** segundos como possíveis tempo de resposta máximo que podem estar associados a diferentes serviços eletrônicos.



Figura 29: Painel de propriedades para uma característica do tipo ‘atributo de QoS’.

Cada sub-característica do tipo serviço eletrônico da característica raiz **servicos-eletronicos** deve referenciar a característica raiz **atributos-de-qos** e conseqüentemente todas as sub-características representando os atributos e níveis de QoS que podem ser definidos para cada serviço eletrônico. Assim, embora o conjunto de características relacionado a atributos e níveis de QoS seja definido apenas uma vez, ele pode ser usado várias vezes, uma vez para cada serviço eletrônico existente devido ao mecanismo de modularidade e à possibilidade de configuração do modelo de características. Como exemplo, no modelo da Figura 25, a característica **aplicacao-de-acao-de-cobranca** possui a referência **• --> atributos-de-qos**, que será expandida durante a configuração do modelo de características – apresentado posteriormente.

Um conjunto de restrições adicionais de características foi definido para esses dois modelos de características, além das restrições já previstas nas regras do próprio meta-modelo de características usado. Essas restrições adicionais também precisaram ser validadas durante a configuração do modelo de características. Na Figura 30 é apresentado o painel de restrições adicionais, da ferramenta FeaturePlugin, com duas restrições

identificadas como necessárias para o Sistema COB. As restrições são especificadas na linguagem XPATH [71] e descritas as seguir:

- **if (//consulta-estado-de-reversao-de-acao-cobranca) then (//reversoes-de-acao-de-cobranca) else true()**:estabelece que se a primeira característica, que representa o serviço eletrônico responsável pela consulta do estado de uma reversão de ação de cobrança, é selecionada, então a segunda característica que representa o grupo de serviços eletrônicos básicos associados a ações de cobrança, deve ser obrigatoriamente selecionada também. Isso é feito porque o uso da primeira característica apenas faz sentido se a segunda também puder ser utilizada.
- **if (//acoes-legais) then (//notificacao-por-carta union //notificacao-por-email union //notificacao-por-telefone) else true()**:estabelece que se a primeira característica que representa o tipo de ação de cobrança “ações legais”, é selecionada, então pelo menos uma das três seguintes características que representam formas de notificação prévia de débito, devem ser obrigatoriamente selecionadas também. Isso se deve ao fato de que uma ação legal pode ser iniciada apenas se algum tipo de notificação prévia ao cliente devedor tiver sido realizado.

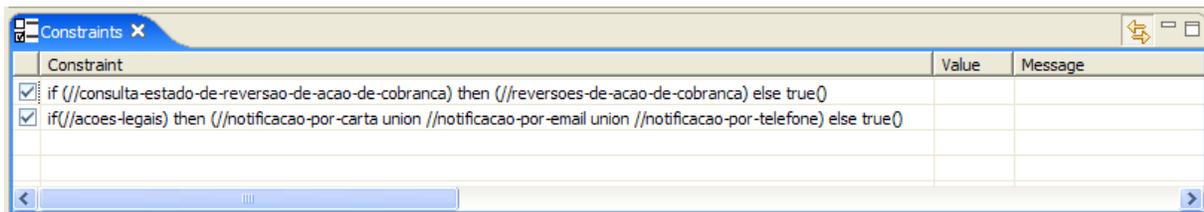


Figura 30: Painel de restrições adicionais.

6.2.2 Molde de Contrato Eletrônico para Serviços Web

Nesta seção é apresentado o molde de contrato produzido como resultado do Estágio 2 do processo de estabelecimento de contratos. O molde de contrato é apresentado em duas etapas: primeiramente são apresentadas partes dos arquivos relativos às seções `<wsdl:Definitions>` e `<wsag:Terms>`, gerados com o apoio da ferramenta XSLTransformer; em seguida é apresentada parte do arquivo contendo a seção `<bpel:Process>`, elaborado com o apoio da ferramenta ActiveBPEL Designer. Os artefatos são apresentados em sua forma gráfica pelas ferramentas que compõem o conjunto FeatureContract.

Os elementos básicos das seções `<wsdl:Definitions>` e `<wsag:Terms>` são gerados de forma automática a partir dos respectivos elementos nos modelos de características. Para a seção `<wsdl:Definitions>`, foram usadas as características de serviços eletrônicos apresentadas na Figura 25 e na Figura 27; para `<wsag:Terms>`, as características dos atributos de QoS contidas na Figura 26 e na Figura 28. A geração automática foi realizada com base nas regras definidas para esse fim (Seção 4.4.2). Após a geração dessas seções do molde, algumas inclusões ou alterações foram realizadas para a incorporação de informações não previstas nos modelos de características.

Na Figura 31 e na Figura 32 são apresentadas partes da seção `<wsdl:Definitions>` para o Sistema COB e para o Sistema CRM, respectivamente. Os serviços Web descritos em linguagem WSDL são apresentados graficamente por meio da ferramenta WSDL Editor, integrante do conjunto de ferramentas FeatureContract. Essa seção contém os elementos básicos da linguagem WSDL usados para descrever os serviços Web do processo de negócio. As características usadas para a geração dessa seção do molde são aquelas localizadas abaixo da sub-árvore **servicos-eletronicos**. A seguir são apresentados exemplos de elementos da Figura 31, em função do modelo de características apresentado na Figura 25 para o Sistema COB.

acoes-de-cobrancaPT		
aplicacao-de-acao-de-cobrancaOP		
input	aplicacao-de-acao-de-cobranca-Request-MSG-PART	string
	tipos-de-acao-de-cobranca-MSG-PART	string
output	aplicacao-de-acao-de-cobranca-Response-MSG-PART	string
atualizacao-de-estado-de-acao-de-cobrancaOP		
input	atualizacao-de-estado-de-acao-de-cobranca-Request-MSG-PART	string
output	atualizacao-de-estado-de-acao-de-cobranca-Response-MSG-PART	string
cancelamento-de-aplicacao-de-acao-de-cobrancaOP		
input	cancelamento-de-aplicacao-de-acao-de-cobranca-Request-MSG-PART	string
output	cancelamento-de-aplicacao-de-acao-de-cobranca-Response-MSG-PART	string

reversoes-de-acao-de-cobrancaPT		
aplicacao-de-reversao-de-acao-de-cobrancaOP		
input	aplicacao-de-reversao-de-acao-de-cobranca-Request-MSG-PART	string
output	aplicacao-de-reversao-de-acao-de-cobranca-Response-MSG-PART	string
atualizacao-de-estado-de-reversao-de-acao-de-cobrancaOP		
input	atualizacao-de-estado-de-reversao-de-acao-de-cobranca-Request-MSG-PART	string
output	atualizacao-de-estado-de-reversao-de-acao-de-cobranca-Response-MSG-PART	string
cancelamento-de-aplicacao-de-reversao-de-acao-de-cobrancaOP		
input	cancelamento-de-aplicacao-de-reversao-de-acao-de-cobranca-Request-MSG-PART	string
output	cancelamento-de-aplicacao-de-reversao-de-acao-de-cobranca-Response-MSG-PART	string

encargos-e-descontosPT		
cancelamento-de-encargoOP		
input	cancelamento-de-encargo-Request-MSG-PART	string
	tipo-de-encargo-MSG-PART	string
output	cancelamento-de-encargo-Response-MSG-PART	string
concessao-de-descontoOP		
input	concessao-de-desconto-Request-MSG-PART	string
	tipos-de-desconto-MSG-PART	string
output	concessao-de-desconto-Response-MSG-PART	string

dividasPT		
parcelamento-de-dividaOP		
input	parcelamento-de-divida-Request-MSG-PART	string
output	parcelamento-de-divida-Response-MSG-PART	string
cancelamento-de-dividaOP		
input	cancelamento-de-divida-Request-MSG-PART	string
output	cancelamento-de-divida-Response-MSG-PART	string

Figura 31: Molde de contrato – seção <wsdl:Definitions> (Sistema COB).

OS-para-executar-acao-de-de-cobrancaPT		
registro-de-OS-para-executar-acao-de-de-cobrancaOP		
input	registro-de-OS-para-executar-acao-de-de-cobranca-Request-MSG-PART	string
output	registro-de-OS-para-executar-acao-de-de-cobranca-Response-MSG-PART	string
consulta-de-OS-para-executar-acao-de-de-cobrancaOP		
input	consulta-de-OS-para-executar-acao-de-de-cobranca-Request-MSG-PART	string
output	consulta-de-OS-para-executar-acao-de-de-cobranca-Response-MSG-PART	string
cancelamento-de-OS-para-executar-acao-de-cobrancaOP		
input	cancelamento-de-OS-para-executar-acao-de-cobranca-Request-MSG-PART	string
output	cancelamento-de-OS-para-executar-acao-de-cobranca-Response-MSG-PART	string

OS-para-executar-rerversao-de-acao-de-de-cobrancaPT		
registro-de-OS-para-executar-rerversao-de-acao-de-de-cobrancaOP		
input	registro-de-OS-para-executar-rerversao-de-acao-de-de-cobranca-Request-MSG-PART	string
output	registro-de-OS-para-executar-rerversao-de-acao-de-de-cobranca-Response-MSG-PART	string
consulta-de-OS-para-executar-rerversao-de-acao-de-de-cobrancaOP		
input	consulta-de-OS-para-executar-rerversao-de-acao-de-de-cobranca-Request-MSG-PART	string
output	consulta-de-OS-para-executar-rerversao-de-acao-de-de-cobranca-Response-MSG-PART	string
cancelamento-de-OS-para-executar-rerversao-de-acao-de-de-cobrancaOP		
input	cancelamento-de-OS-para-executar-rerversao-de-acao-de-de-cobranca-Request-MSG-PART	string
output	cancelamento-de-OS-para-executar-rerversao-de-acao-de-de-cobranca-Response-MSG-PART	string

atualizacao-de-informacoes-de-clientePT		
atualizacao-de-estado-de-conta-devedora-de-clienteOP		
input	atualizacao-de-estado-de-conta-devedora-de-cliente-Request-MSG-PART	string
output	atualizacao-de-estado-de-conta-devedora-de-cliente-Response-MSG-PART	string
atualizacao-de-estado-de-cheque-irregular-de-clienteOP		
input	atualizacao-de-estado-de-cheque-irregular-de-cliente-Request-MSG-PART	string
output	atualizacao-de-estado-de-cheque-irregular-de-cliente-Response-MSG-PART	string

avisos-de-inadimplenciaPT		
emissao-de-aviso-de-inadimplenciaOP		
input	emissao-de-aviso-de-inadimplencia-Request-MSG-PART	string
output	emissao-de-aviso-de-inadimplencia-Response-MSG-PART	string
cancelamento-de-aviso-de-inadimplenciaOP		
input	cancelamento-de-aviso-de-inadimplencia-Request-MSG-PART	string
output	cancelamento-de-aviso-de-inadimplencia-Response-MSG-PART	string

Figura 32: Molde de contrato – seção <wsdl:Definitions> (Sistema CRM).

Exemplos similares aos apresentados a seguir existem também para a Figura 32, em função do modelo do modelo de características da Figura 27 para o Sistema CRM.

Os principais elementos da seção `<wsdl:Definitions>` são os tipos de porta `<portType>`, criados a partir das características do tipo grupo de serviços. Como exemplo, o tipo de porta **acoes-de-cobrancaPT** é gerado a partir da característica **acoes-de-cobranca**. Cada tipo de porta representa um serviço Web composto por um conjunto de operações `<operation>`, criadas a partir das sub-características do tipo serviço eletrônico. Um exemplo de operação para o tipo de porta **acoes-de-cobrancaPT** é **aplicacao-de-acao-de-cobrancaOP**. Esse tipo de porta é gerado a partir da característica chamada **aplicacao-de-acao-de-cobranca**.

Cada operação possui um conjunto de pelo menos dois tipos de mensagem associados, uma de entrada e uma de saída, a serem usadas como variáveis pelas operações. Além desses dois tipos, algumas operações possuem tipos de mensagem adicionais criados a partir de sub-característica do tipo propriedades de serviços existente a partir da hierarquia de características. Por exemplo, para a operação **aplicacao-de-acao-de-cobrancaOP** há os dois tipos de mensagem básicos: tipo de mensagem de entrada – **aplicacao-de-acao-de-cobranca-Request-MSG-PART** (`input`); e tipo de mensagem de saída – **aplicacao-de-acao-de-cobranca-Request-MSG-PART** (`output`). Além desses dois, há um tipo adicional de mensagem de entrada – **tipos-de-acao-de-cobranca-MSG-PART**, criado a partir da característica chamada **tipos-de-acao-de-cobranca**.

Além dos elementos apresentados na Figura 31, existe outro tipo de elemento da seção `<wsdl:Definitions>` que é o tipo de ligação entre parceiros `<partnerLinkType>`. Esse tipo de elemento não faz parte do conjunto padrão de elementos de uma especificação WSDL. Como a ferramenta WSDL Editor oferece visualização gráfica apenas aos tipos de elemento padrão, ele não é apresentado na Figura 31. Sendo assim, como exemplo, parte dos elementos `<partnerLinkType>` para o Sistema COB é apresentado, na forma textual, no Código 08. Por serem similares, os elementos para o Sistema CRM não são apresentados. Um exemplo de tipo de ligação entre parceiros criado é **acoes-de-**

cobrancaPLT, gerado a partir da característica **acoes-de-cobranca**, do tipo grupo de serviços e associado ao tipo de porta **acoes-de-cobrancaPT**.

Código 08: Molde de contrato – seção <wsdl:Definitions>, elementos <partnerLinkType> (Sistema COB).

```
<!--FEATURE_ID = acoesdecobranca-->
<plnk:partnerLinkType name="acoes-de-cobrancaPLT">
  <plnk:role name="acoes-de-cobrancaROLE">
    <plnk:portType name="tns:acoes-de-cobrancaPT" />
  </plnk:role>
</plnk:partnerLinkType>

<!--FEATURE_ID = reversoesdeacaodecobranca-->
<plnk:partnerLinkType name="reversoes-de-acao-de-cobrancaPLT">
  <plnk:role name="reversoes-de-acao-de-cobrancaROLE">
    <plnk:portType name="tns:reversoes-de-acao-de-cobrancaPT" />
  </plnk:role>
</plnk:partnerLinkType>
(...)
<!--FEATURE_ID = cadastrosdeclientes-->
<plnk:partnerLinkType name="cadastros-de-clientesPLT">
  <plnk:role name="cadastros-de-clientesROLE">
    <plnk:portType name="tns:cadastros-de-clientesPT" />
  </plnk:role>
</plnk:partnerLinkType>
```

Como parte da estratégia de anotação do molde de contrato, todos os elementos apresentados anteriormente são precedidos por comentários no arquivo-fonte (especificação WSDL). Estas anotações não são visíveis graficamente na Figura 31 e na Figura 32. No Código 08 elas são visualizadas, como o caso do comentário **<!--FEATURE_ID = acoesdecobranca-->** que precede o elemento tipo de ligação entre parceiros **<plnk:partnerLinkType name="acoes-de-cobrancaPLT">**, ligando-o à característica usada em sua criação.

Na Figura 33 e na Figura 34 são apresentadas partes da seção **<wsag:Terms>** para o Sistema COB. Os atributos de QoS descritos em linguagem WS-Agreement são apresentados graficamente por meio da ferramenta XML Editor, integrante do conjunto de

ferramentas FeatureContract. Essa seção contém a descrição dos atributos de QoS, e seus respectivos níveis, relacionados com os serviços Web envolvidos. As características usadas para a geração dessa seção do molde de contrato são todas as existentes abaixo da sub-árvore **atributos-de-qos**. A seguir são apresentados exemplos de elementos do molde de contrato gerados em função do modelo de características apresentado na Figura 26. Exemplos de elementos similares são encontrados no outro arquivo, produzido para o Sistema CRM.

!--	FEATURE_ID = aplicacaodeacaodecobranca
[-] [e] wsag:ServiceProperties	
[a] wsag:Name	aplicacao-de-acao-de-cobrancaSP
[a] wsag:ServiceName	aplicacao-de-acao-de-cobrancaOP
[-] [e] wsag:VariableSet	
!--	FEATURE_ID = reference20IXIseguranca
+ [e] wsag:Variable	
!--	FEATURE_ID = reference20IXIdisponibilidade
[-] [e] wsag:Variable	
[a] wsag:Name	disponibilidadeVAR
[a] wsag:Metric	amount:frequency
+ [e] wsag:Location	
!--	FEATURE_ID = reference20IXItempodeespera
+ [e] wsag:Variable	
!--	FEATURE_ID = reference20IXItempoderesposta
[-] [e] wsag:Variable	
[a] wsag:Name	tempo-de-respostaVAR
[a] wsag:Metric	time:duration
+ [e] wsag:Location	
!--	FEATURE_ID = reference20IXIacessosimultaneos
+ [e] wsag:Variable	
!--	FEATURE_ID = reference20IXIdesempenho
+ [e] wsag:Variable	
!--	FEATURE_ID = atualizacaodeestadodeacaodecobranca
+ [e] wsag:ServiceProperties	
!--	FEATURE_ID = cancelamentodeaplicacaodeacaodecobranca
+ [e] wsag:ServiceProperties	

Figura 33: Molde de contrato – seção <wsag:Terms>, Parte I (Sistema COB).

!--	FEATURE_ID = aplicacaodeacaodecobranca
[-] e wsag:GuaranteeTerm	
a Obligated	fornecedorDeServico
[-] e wsag:ServiceScope	
a ServiceName	aplicacao-de-acao-de-cobrancaOP
+ e wsag:QualifyingCondition	
[-] e wsag:ServiceLevelObjective	
[]	disponibilidadeVAR IS-LESS-INCLUSIVE
!--	FEATURE_ID = reference20IXIsemcontrole0
[]	sem-controle
!--	FEATURE_ID = reference20IXIa24X7
[]	24X7
!--	FEATURE_ID = reference20IXIa24X5
[]	24X5
!--	FEATURE_ID = reference20IXIa8X5
[]	8X5
!--	FEATURE_ID = reference20IXIoutro
[]	Outro
+ e wsag:BusinessValueList	
!--	FEATURE_ID = aplicacaodeacaodecobranca
+ e wsag:GuaranteeTerm	
!--	FEATURE_ID = aplicacaodeacaodecobranca
[-] e wsag:GuaranteeTerm	
a Obligated	fornecedorDeServico
[-] e wsag:ServiceScope	
a ServiceName	aplicacao-de-acao-de-cobrancaOP
+ e wsag:QualifyingCondition	
[-] e wsag:ServiceLevelObjective	
[]	tempo-de-respostaVAR IS-LESS-INCLUSIVE
!--	FEATURE_ID = reference20IXIsemcontrole2
[]	sem-controle
!--	FEATURE_ID = reference20IXIa5
[]	5
!--	FEATURE_ID = reference20IXIa16
[]	15
!--	FEATURE_ID = reference20IXIa31
[]	30
!--	FEATURE_ID = reference20IXIoutro1
[]	Outro
+ e wsag:BusinessValueList	
!--	FEATURE_ID = aplicacaodeacaodecobranca
+ e wsag:GuaranteeTerm	

Figura 34: Molde de contrato – seção <wsag:Terms>, Parte II (Sistema COB).

Na Figura 33 é apresentada uma parte da seção `<wsag:Terms>` associada à definição de propriedades de serviço `<wsag:ServiceProperties>` criadas para cada operação de serviço Web na seção `<wsdl:Definitions>`. Para cada propriedade de serviço, um conjunto de variáveis de garantia `<wsag:Variable>` é também criado. As variáveis de garantia são geradas a partir de todas as características do tipo atributo de QoS. A subseção `<wsag:Location>` é criada vazia, visto que informações para seu preenchimento não são definidas no modelo de características, mas sim posteriormente. O exemplo de propriedade de serviço apresentado na figura é **aplicacao-de-acao-de-cobrancaSP**, associada à operação **aplicacao-de-acao-de-cobrancaOP**. Uma das variáveis de garantia para essa propriedade de serviço é **tempo-de-respostaVAR**, gerada da característica **tempo-de-resposta**. Como todas as características do tipo atributo de QoS são obrigatórias, então todos os elementos propriedade de serviço também o são.

Na Figura 34 é apresentada uma parte da seção `<wsag:Terms>` associada à definição de termos de garantia `<wsag:GuaranteeTerm>` também criados para cada operação de serviço Web; um para cada variável de garantia `<wsag:Variable>` de uma determinada propriedade de serviço. Para cada termo de garantia, um objetivo de nível de serviço `<wsag:ServiceLevelObjective>` é associado à respectiva variável de garantia. O objetivo de nível de serviço é criado a partir de: características agrupadas para a opção de atributo **sem-controle**; e sub-características do tipo nível das características agrupadas para a opção de atributo **niveis-de-qos**.

O exemplo de termo de garantia apresentado na Figura 34 é o associado à operação **aplicacao-de-acao-de-cobrancaOP** e à variável de garantia **tempo-de-respostaVAR**. As opções de atributo criadas para esse termo de garantia são: **sem-controle**, **5**, **15**, **30** e **Outro**. Como no caso anterior, as seções `<wsag:QualifyingCondition>` e `<wsag:BusinessValueList>` são criadas vazias, visto que informações para seu preenchimento não são definidas no modelo de características, mas sim posteriormente. Como todas as características do tipo opção de atributo **sem-controle** e **niveis-de-**

qos são mutuamente exclusivas, então as respectivas partes do molde de contrato também o são. O mesmo ocorre para as características do tipo `nível` e para as respectivas partes do molde de contrato.

Ainda como parte da estratégia de anotação do molde de contrato eletrônico, a estratégia de uso de comentários é também seguida para os elementos descritos na seção `<wsag:Terms>`. Desse modo, cada um desses elementos é precedido por comentários que fazem a ligação entre eles e as respectivas características usadas em suas criações, gerados automaticamente durante a geração dessa seção do molde de contrato. Tais comentários são visíveis graficamente, como – por exemplo – o comentário `<!-- FEATURE_ID = reference20IXIdisponibilidade-->` que precede o elemento variável de garantia `<wsag:Variable wsag:Name="disponibilidadeVAR" wsag:Metric="amount:frequency">`, ligando-o à característica usada em sua criação.

Na Figura 35 é apresentada parte da seção `<bpel:Process>` para o processo de negócio envolvendo o Sistema COB e o Sistema CRM. O processo descrito em linguagem WS-BPEL é apresentado graficamente por meio da ferramenta ActiveBPEL Designer, integrante do conjunto FeatureContract. Essa seção contém a descrição do processo de negócio que faz a composição de serviços Web das seções `<wsdl:Definitions>` de cada um dos sistemas envolvidos. Por ser uma descrição de processo no nível de molde de contrato eletrônico, trata-se de uma descrição genérica de processo. Portanto, diversas possibilidades de acordo de negócio entre as organizações responsáveis pelo Sistema COB e pelo Sistema CRM estão representadas. Dessas possibilidades, apenas uma delas é usada em cada instância de contrato a ser criada em uma etapa futura.

Como apenas parte do processo de negócio é apresentada aqui, não estão representados todos os serviços Web que foram criados na seção `<wsdl:Definitions>` e que foram usados na seção `<bpel:Process>`. Está apresentada apenas a parte do processo de negócio envolvendo os serviços Web relativos a: aplicação de ações de cobrança e a reversão da aplicação de tais ações, por parte do Sistema COB; e a gerência de ordens de serviços para apoiar tais aplicações e reversões, por parte do Sistema CRM.

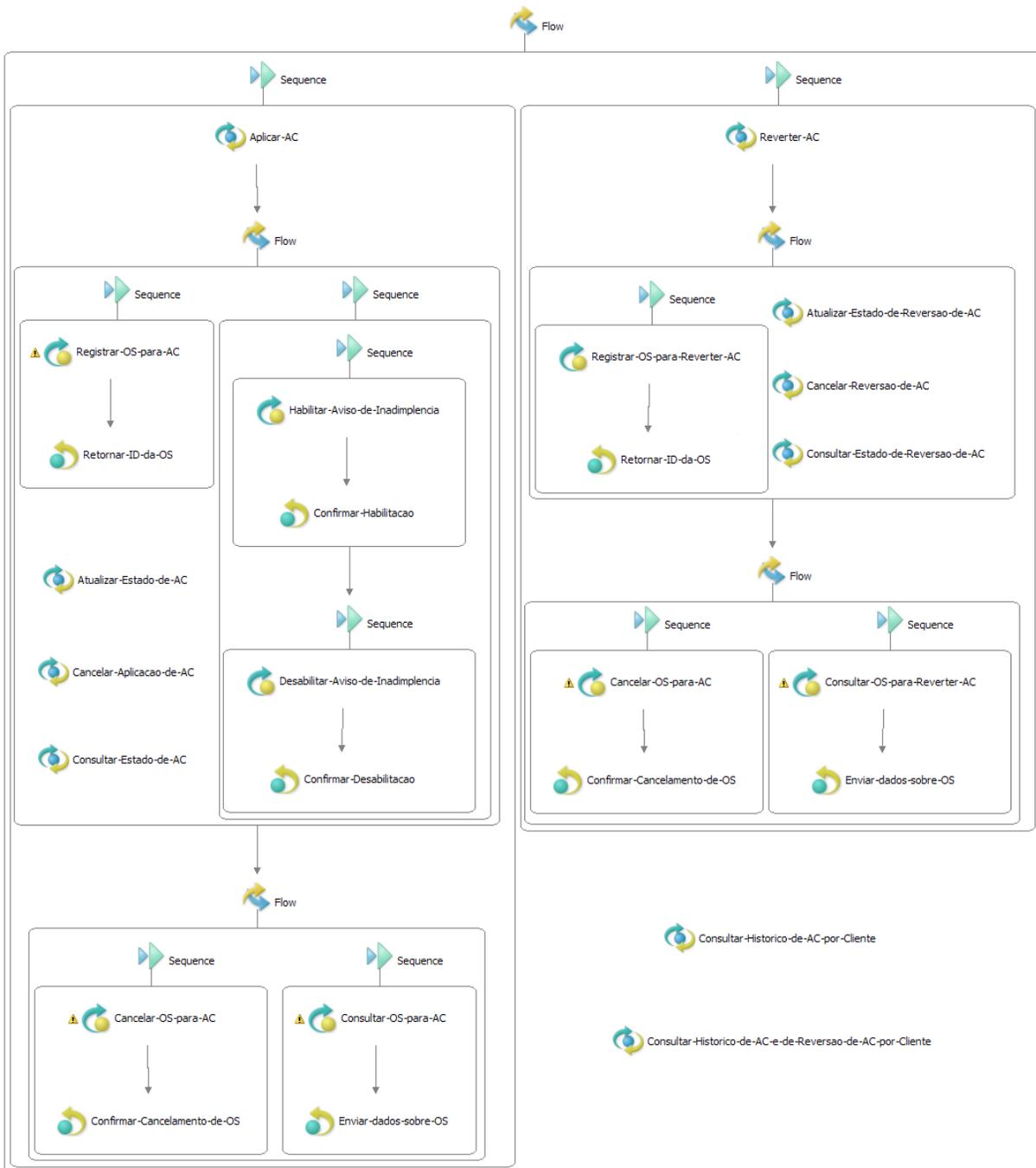


Figura 35: Molde de contrato – seção <bpel:Process>.

O processo apresentado na Figura 35 é composto por uma série de seqüências de fluxos paralelos de serviços Web sendo invocados entre os dois sistemas envolvidos. O processo

é especificado tendo o Sistema CRM como ator principal na cooperação entre os dois sistemas. Desse modo, a comunicação entre os sistemas para o uso de serviços Web é realizado de duas formas:

1. **Sistema CRM invoca serviços Web oferecidos pelo Sistema COB:** esse tipo de operação é representado por apenas um elemento gráfico no processo (*invoke* – ). O recebimento da invocação do serviço Web por parte do sistema COB e a sua resposta ao sistema CRM não são apresentados, por estarem no escopo de operação do sistema COB. Exemplo: a ação **Aplicar-AC** invoca a operação **aplicacao-de-acao-de-cobrancaOP** do tipo de porta **acoes-de-cobrancaPT** do sistema COB;
2. **Sistema CRM recebe uma invocação do Sistema COB / Sistema CRM responde à invocação realizada pelo Sistema COB:** estas operações são representadas por dois elementos gráficos no processo (*receive* –  e *reply* – ). Ao contrário do caso anterior, a invocação do serviço Web por parte do sistema COB não é apresentada por estar no escopo de operação do sistema COB. Exemplo: As ações **Registrar-OS-para-AC** e **Retornar-ID-da-OS** que, respectivamente, recebe e responde à invocação **registrar-OS-para-executar-acao-de-cobrancaPT** do tipo de porta **OS-para-executar-acao-de-cobrancaPT** do sistema CRM.

Além dos elementos apresentados na Figura 35, existem outros dois tipos de elemento da seção `<bpel:Process>` – ligação entre parceiros `<partnerLink>` e variável do processo `<variable>`. Esses tipos de elemento não são apresentados graficamente por meio da ferramenta ActiveBPEL Designer. Parte dos elementos `<partnerLinkType>` e `<variable>` para o processo é apresentada, na forma textual, no Código 09. Exemplos de ligações entre parceiros são: **acoes-de-cobranca** – em que as operações dos serviços Web são executadas pelo sistema “parceiro” (COB) do sistema principal (CRM); e **OS-para-executar-acao-de-cobranca** – em que as operações dos serviços Web são

executadas pelo sistema “principal”. Exemplos de variável de processo são: **aplicacao-de-acao-de-cobranca-Request** e **aplicacao-de-acao-de-cobranca-Response**.

Código 09: Molde de contrato – seção <bpel:Process>, elementos <partnerLink> e <variable>.

```
<bpel:partnerLinks>
  <!-- FEATURE_ID = acoesdecobranca -->
  <bpel:partnerLink name="acoes-de-cobranca"
    partnerLinkType="ns1:acoes-de-cobrancaPLT"
    partnerRole="acoes-de-cobrancaROLE"/>
  (...)
  <!-- FEATURE_ID = oSparaexecutaracaodedecobranca -->
  <bpel:partnerLink myRole="OS-para-executar-acao-de-cobrancaROLE"
    name="OS-para-executar-acao-de-cobranca"
    partnerLinkType="ns1:OS-para-executar-acao-de-cob (...) "/>
  (...)
</bpel:partnerLinks>

<bpel:variables>
  <!--FEATURE_ID = aplicacaodeacaodecobranca-->
  <bpel:variable messageType="ns1:aplicacao-de-acao-de-cobranca-Req-MSG"
    name="aplicacao-de-acao-de-cobranca-Request"/>

  <!--FEATURE_ID = aplicacaodeacaodecobranca-->
  <bpel:variable messageType="ns1:aplicacao-de-acao-de-cobranca-Res-MSG"
    name="aplicacao-de-acao-de-cobranca-Response"/>
  (...)
  <!--FEATURE_ID = actionApplication-->
  <bpel:variable messageType="ns1:registro-de-OS-para-executar-acao-de (...) "
    name="registro-de-OS-para-executar-acao-de-cobranca (...) "/>

  <!--FEATURE_ID = actionApplication-->
  <bpel:variable messageType="ns1:registro-de-OS-para-executar-acao-de (...) "
    name="registro-de-OS-para-executar-acao-de-cobranca (...) "/>
  (...)
</bpel:variables>
```

Como parte da estratégia de anotação do molde de contrato, todos os elementos apresentados anteriormente são precedidos por comentários no arquivo-fonte (especificação WSDL). Estas anotações não são visíveis graficamente na Figura 35. No Código 09 elas são visualizadas, como o caso do comentário <!--FEATURE_ID =

OSparaexecutaracaodedecobranca--> que precede o elemento ligação entre parceiros **<bpel:partnerLink (...) name="OS-para-executar-acao-de-cobranca" (...)/>**, ligando-o à característica usada em sua criação.

6.2.3 Configurações dos Modelos de Características para Serviços Eletrônicos

Nesta seção são apresentados os modelos de características configurados no Estágio 4 do processo de estabelecimento de contratos usando a ferramenta FeaturePlugin, integrante do conjunto de ferramentas FeatureContract. Os modelos de características configurados são os apresentados na Figura 25 e na Figura 26. Partes dos modelos de características configurados são apresentadas na Figura 36, na Figura 37 e na Figura 38. Várias outras configurações poderiam ser facilmente derivadas para eles, caso diferentes processos de negócio fossem desejados.

Primeiramente é apresentada a configuração para o Sistema COB, cujo modelo de características foi mostrado na Figura 25. A Figura 36 apresenta parte da configuração das características relativas a serviços eletrônicos. Diversas características obrigatórias aparecem pré-selecionadas, como a **aplicacao-de-acao-de-cobranca**. Algumas características opcionais/alternativas não foram selecionadas – como por exemplo **reversoes-de-acao-de-cobranca**, do tipo `grupo de serviços`. Nesse caso, a empresa operadora de telecomunicações não deseja contratar esse serviço. Conseqüentemente, todas suas sub-características, do tipo `serviço eletrônico`, também não são selecionadas – embora sejam características obrigatórias.

Por outro lado, algumas características opcionais/alternativas foram selecionadas. Por exemplo, a característica **consulta-de-historico-de-acoes-de-cobranca-por-cliente**, do tipo `serviço eletrônico`. A seleção dessa característica faz com que a característica **consulta-de-historico-de-acoes-e-reversoes-de-acao-de-cobranca**, também opcional,

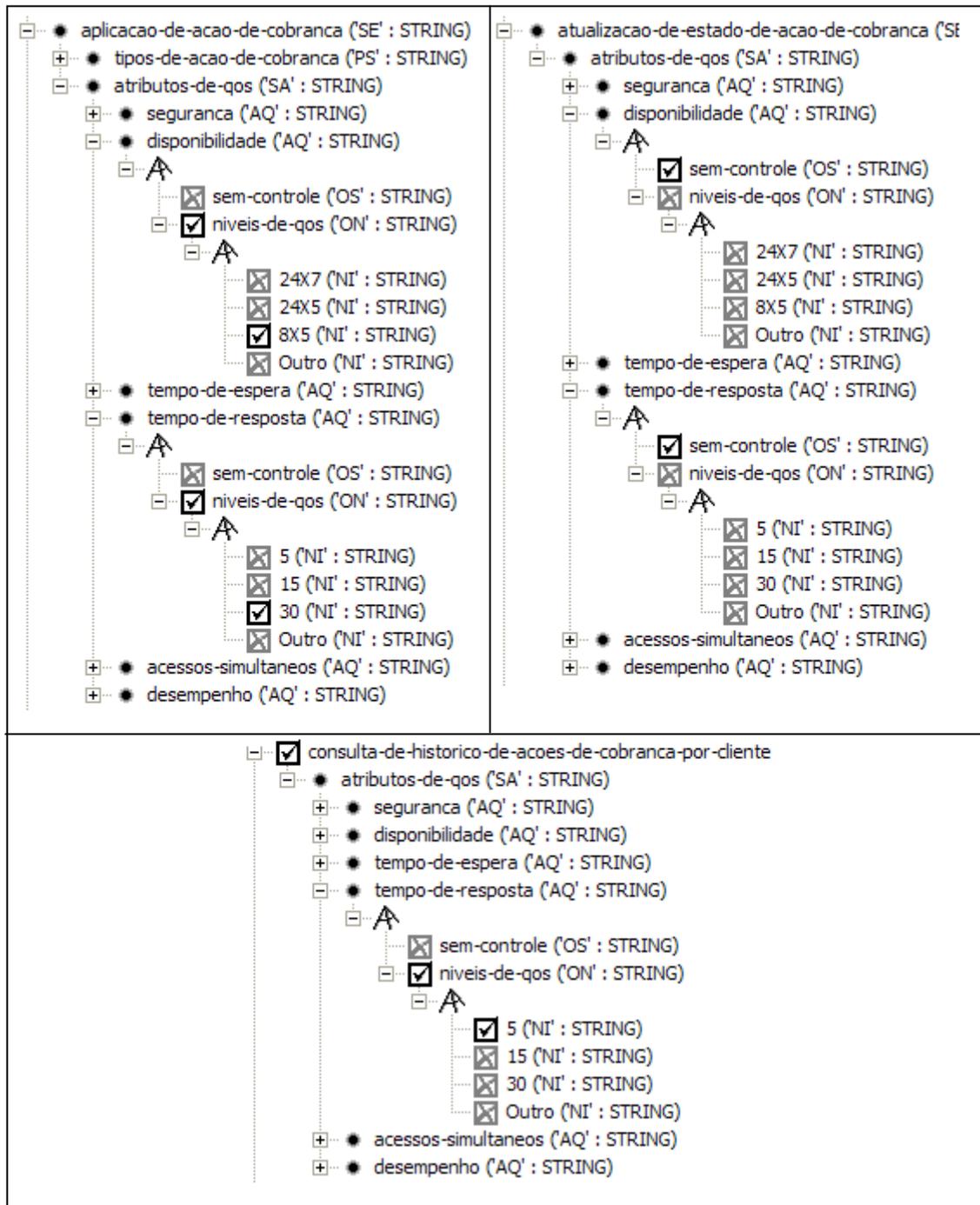


Figura 37: Configuração do modelo de características para serviços eletrônicos do Sistema COB (características relacionadas aos atributos de QoS).

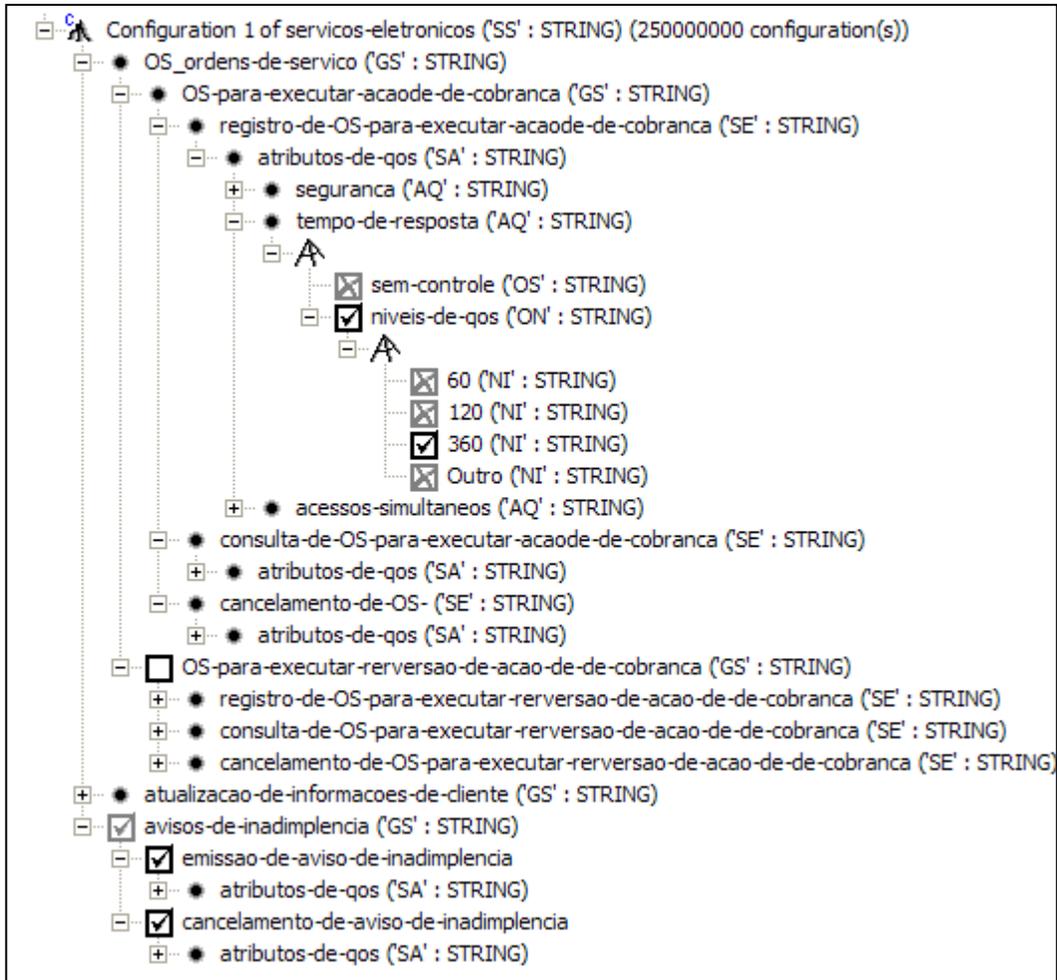


Figura 38: Configuração do modelo de características para serviços eletrônicos do Sistema CRM.

Partes da configuração das características relativas aos atributos de QoS são apresentadas na Figura 37. Como todo o conjunto de características da sub-árvore **atributos-de-qos** está associado às características do tipo serviço eletrônico da sub-árvore **servicos-eletronicos**, aquelas precisam ser configuradas para todas essas. Nessa figura, são apresentadas três partes da configuração para três diferentes características do tipo serviço eletrônico: **aplicacao-de-acao-de-cobranca**,

atualizacao-de-estado-de-acao-de-cobranca e **consulta-de-historico-de-acoes-de-cobranca-por-cliente**.

Para todas essas características do tipo serviço eletrônico, todas as características do tipo atributo de QoS foram configuradas – por serem obrigatórias, embora, somente para algumas, a configuração é apresentada. Apesar de serem todas obrigatórias, a característica **sem-controle**, do tipo opção de atributo, pode ser selecionada – como, por exemplo, para o atributo **disponibilidade** para o serviço eletrônico **atualizacao-de-estado-de-acao-de-cobranca**. Para esse mesmo atributo, as características **niveis-de-qos**, do tipo opção de atributo, e **8X5**, do tipo nível, foram selecionadas para o serviço eletrônico **aplicacao-de-acao-de-cobranca**.

As duas restrições adicionais de características que foram definidas para esse modelo de características (apresentadas na Seção 6.2.1) foram validadas pela ferramenta FeaturePlugin durante a configuração desse modelo de características:

- **if (//consulta-estado-de-reversao-de-acao-cobranca) then (//reversoes-de-acao-de-cobranca) else true()**: ambas as características não foram selecionadas;
- **if (//acoes-legais) then (//notificacao-por-carta union //notificacao-por-email union //notificacao-por-telefone) else true()**: ambas as características notificacao-por-email e notificacao-por-telefone foram selecionadas, permitindo a seleção realizada da característica acoes-legais.

A configuração do modelo de características para o Sistema CRM também foi realizada. Parte dessa configuração é apresentada na Figura 38. Dado que, durante a negociação entre as duas empresas, os serviços eletrônicos relativos a reversões de ação de cobrança não foram selecionados, então a característica **OS-para-executar-rerversao-de-acao-de-cobranca** do Sistema CRM, do tipo grupo de serviços, também não o foi.

Como o Sistema COB não executará reversões de ação de cobrança, não é necessário que o Sistema CRM disponibilize a ele esses serviços.

6.2.4 Instância de Contrato Eletrônico para Serviços Web

Nesta seção é apresentada a instância de contrato resultante do Estágio 5 para o estabelecimento de contratos. Ele é produzido pelo componente WS-Contract Factory, do conjunto de ferramentas FeatureContract. São incluídos elementos associados às características obrigatórias e às características opcionais/alternativas que tenham sido selecionadas durante o Estágio 5; os elementos associados às características opcionais/alternativas que não foram selecionadas são removidos do molde para formar o contrato resultante. O molde de contrato eletrônico usado para esse estágio é o conjunto final de arquivos, incluindo todas as alterações/adições realizadas, caso necessário, depois de sua geração automática pelo componente XSLTransformer.

O conjunto de arquivos resultantes desse estágio é apresentado na Figura 39, na Figura 40, na Figura 41, na Figura 42, na Figura 43, no Código 10 e no Código 11. Esses arquivos foram criados com base naqueles apresentados na Seção 6.2.2, e possuem um subconjunto das informações apresentadas neles.

Na Figura 39 e na Figura 40 são apresentadas partes da seção `<wsdl:Definitions>` do contrato eletrônico para o Sistema COB e para o Sistema CRM, respectivamente. Esses artefatos são referentes às mesmas partes do molde de contrato apresentados na Figura 31 e na Figura 32. Na Figura 39 não é apresentado, por exemplo, o tipo de porta **reversoes-de-acao-de-cobrancaPT** e suas operações, dado que o conjunto de serviços eletrônicos associado a ele não foi selecionado no modelo de características configurado (Figura 25). Já na Figura 40 não é apresentado por motivo similar, por exemplo, o tipo de porta **OS-para-executar-reversao-de-acao-de-cobrancaPT**.

No Código 10 é apresentada outra parte da seção <wsdl:Definitions> do contrato eletrônico, em que os tipos de ligação entre parceiros são definidos textualmente. Esse artefato é referente à mesma parte do molde de contrato apresentado no Código 08. Nessa parte do contrato não é apresentado, por exemplo, o tipo de ligação entre parceiros **reversoes-de-acao-de-cobrancaPLT**, dado que a característica associada a ele não foi selecionada no modelo de características configurado (Figura 25). Por se tratar de uma instância final do contrato eletrônico, todas as anotações foram removidas por não serem mais necessárias e para simplificar os artefatos resultantes do processo. As anotações existentes nas demais partes do contrato, apresentadas graficamente nas figuras anteriores também foram removidas do arquivo-fonte.

aco-es-de-cobrancaPT		
aplicacao-de-acao-de-cobrancaOP		
input	aplicacao-de-acao-de-cobranca-Request-MSG-PART	string
	tipos-de-acao-de-cobranca-MSG-PART	string
output	aplicacao-de-acao-de-cobranca-Response-MSG-PART	string
atualizacao-de-estado-de-acao-de-cobrancaOP		
input	atualizacao-de-estado-de-acao-de-cobranca-Request-MSG-PART	string
output	atualizacao-de-estado-de-acao-de-cobranca-Response-MSG-PART	string
cancelamento-de-aplicacao-de-acao-de-cobrancaOP		
input	cancelamento-de-aplicacao-de-acao-de-cobranca-Request-MSG-PART	string
output	cancelamento-de-aplicacao-de-acao-de-cobranca-Response-MSG-PART	string
encargos-e-descontosPT		
cancelamento-de-encargoOP		
input	cancelamento-de-encargo-Request-MSG-PART	string
	tipo-de-encargo-MSG-PART	string
output	cancelamento-de-encargo-Response-MSG-PART	string
concessao-de-descontoOP		
input	concessao-de-desconto-Request-MSG-PART	string
	tipos-de-desconto-MSG-PART	string
output	concessao-de-desconto-Response-MSG-PART	string

Figura 39: Contrato eletrônico – seção <wsdl:Definitions> (Sistema COB).

OS-para-executar-acao-de-de-cobrancaPT		
registro-de-OS-para-executar-acao-de-de-cobrancaOP		
input	registro-de-OS-para-executar-acao-de-de-cobranca-Request-MSG-PART	string
output	registro-de-OS-para-executar-acao-de-de-cobranca-Response-MSG-PART	string
consulta-de-OS-para-executar-acao-de-de-cobrancaOP		
input	consulta-de-OS-para-executar-acao-de-de-cobranca-Request-MSG-PART	string
output	consulta-de-OS-para-executar-acao-de-de-cobranca-Response-MSG-PART	string
cancelamento-de-OS-para-executar-acao-de-cobrancaOP		
input	cancelamento-de-OS-para-executar-acao-de-cobranca-Request-MSG-PART	string
output	cancelamento-de-OS-para-executar-acao-de-cobranca-Response-MSG-PART	string

avisos-de-inadimplenciaPT		
emissao-de-aviso-de-inadimplenciaOP		
input	emissao-de-aviso-de-inadimplencia-Request-MSG-PART	string
output	emissao-de-aviso-de-inadimplencia-Response-MSG-PART	string
cancelamento-de-aviso-de-inadimplenciaOP		
input	cancelamento-de-aviso-de-inadimplencia-Request-MSG-PART	string
output	cancelamento-de-aviso-de-inadimplencia-Response-MSG-PART	string

Figura 40: Contrato eletrônico – seção <wsdl:Definitions> (Sistema CRM).

Na Figura 41 e na Figura 42 são apresentadas partes da seção <wsag:Terms> do contrato eletrônico para o Sistema COB. Esses artefatos são referentes às mesmas partes do molde de contrato apresentados na Figura 33 e na Figura 34. Na Figura 41 todos os elementos, associados às propriedades de serviço, apresentados na Figura 33 foram mantidos. Nenhum deles foi removido visto que todos eles estavam relacionados às características opcionais que foram selecionadas no modelo de características configurado (Figura 26). Apenas as anotações do contrato foram removidas. No restante desse artefato, houve elementos removidos, porém não os apresentados aqui.

Já na Figura 42 alguns dos elementos, associados aos termos de garantia, apresentados na Figura 34 foram removidos. Quatro dos cinco possíveis níveis de QoS para o atributo **tempo-de-respostaVAR**, por exemplo, foram removidos. Foram removidas as opções **sem-controle**, **5**, **15** e **outro**; foi mantida a opção **30**, dado que a característica associada a

ela foi a única selecionada para o respectivo atributo de QoS no modelo de características configurado (Figura 26). As anotações do contrato para esta seção também foram removidas.

Código 10: Contrato eletrônico – seção <wsdl:Definitions>, elementos <partnerLinkType> (Sistema COB).

```

<plnk:partnerLinkType name="acoes-de-cobrancaPLT">
  <plnk:role name="acoes-de-cobrancaROLE">
    <plnk:portType name="tns:acoes-de-cobrancaPT" />
  </plnk:role>
</plnk:partnerLinkType>
(...)
<plnk:partnerLinkType name="cadastros-de-clientesPLT">
  <plnk:role name="cadastros-de-clientesROLE">
    <plnk:portType name="tns:cadastros-de-clientesPT" />
  </plnk:role>
</plnk:partnerLinkType>

```

[-] [e] wsag:ServiceProperties	
[a] wsag:Name	aplicacao-de-acao-de-cobrancaSP
[a] wsag:ServiceName	aplicacao-de-acao-de-cobrancaOP
[-] [e] wsag:VariableSet	
+ [e] wsag:Variable	
[-] [e] wsag:Variable	
[a] wsag:Name	disponibilidadeVAR
[a] wsag:Metric	amount:frequency
+ [e] wsag:Location	
+ [e] wsag:Variable	
[-] [e] wsag:Variable	
[a] wsag:Name	tempo-de-respostaVAR
[a] wsag:Metric	time:duration
+ [e] wsag:Location	
+ [e] wsag:Variable	
+ [e] wsag:Variable	
+ [e] wsag:ServiceProperties	
+ [e] wsag:ServiceProperties	

Figura 41: Contrato eletrônico – seção <wsag:Terms>, Parte I (Sistema COB).

[-] [e] wsag:GuaranteeTerm	
[a] Obligated	forneecedorDeServico
[-] [e] wsag:ServiceScope	
[a] ServiceName	aplicacao-de-acao-de-cobrancaOP
[+] [e] wsag:QualifyingCondition	
[-] [e] wsag:ServiceLevelObjective	
[document icon]	disponibilidadeVAR IS-LESS-INCLUSIVE
[document icon]	8X5
[+] [e] wsag:BusinessValueList	
[+] [e] wsag:GuaranteeTerm	
[-] [e] wsag:GuaranteeTerm	
[a] Obligated	forneecedorDeServico
[-] [e] wsag:ServiceScope	
[a] ServiceName	aplicacao-de-acao-de-cobrancaOP
[+] [e] wsag:QualifyingCondition	
[-] [e] wsag:ServiceLevelObjective	
[document icon]	tempo-de-respostaVAR IS-LESS-INCLUSIVE
[document icon]	30
[+] [e] wsag:BusinessValueList	
[+] [e] wsag:GuaranteeTerm	

Figura 42: Contrato eletrônico – seção <wsag:Terms>, Parte II (Sistema COB).

Na Figura 43 é apresentada partes da seção <bpel:Process> do contrato eletrônico. Esse artefato é referente à mesma parte do molde de contrato apresentado na Figura 35. Na Figura 43 não são, portanto, apresentadas as atividades associadas aos serviços Web de reversão de ação de cobrança para o Sistema COB (exemplo: atividade `invoke` **Reverter-AC**) e de OS para reversão de ação de cobrança para o Sistema CRM (exemplo: atividade `receive` **Registrar-OS-para-Reverter-AC** e atividade `reply` **Retornar-ID-da-OS**). Estas atividades foram removidas do contrato resultante dado que todo o conjunto de serviços eletrônicos associado a eles não foi selecionado no modelo de características configurado (Figura 25 e Figura 27).

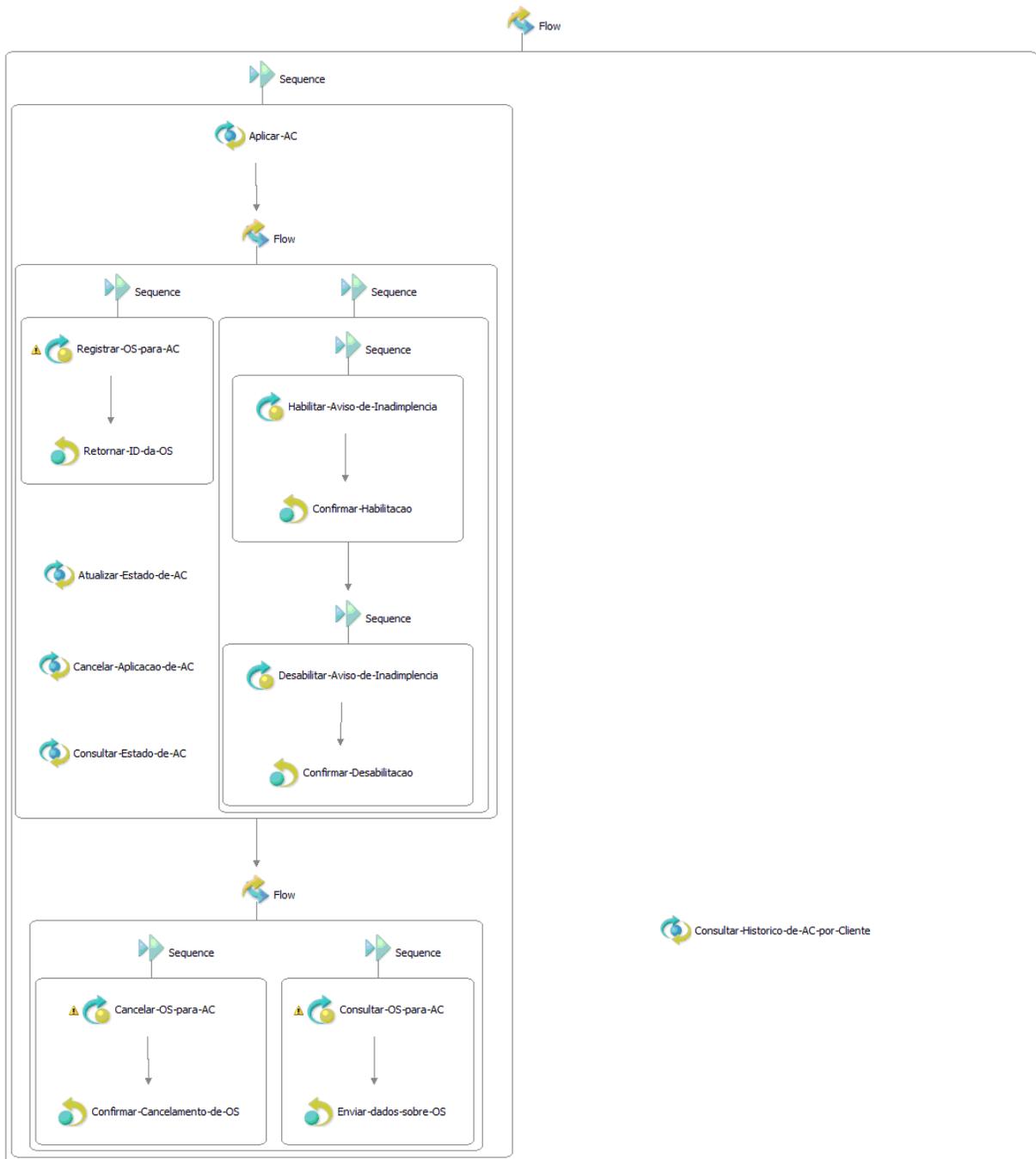


Figura 43: Contrato eletrônico – seção <bpel:Process>.

No Código 11 é apresentada outra parte da seção `<bpel:Process>` do contrato eletrônico, em que as ligações entre parceiros e as variáveis são definidas textualmente. Esse artefato é referente à mesma parte do molde de contrato apresentado no Código 09. No Código 11 todos os elementos, associados a propriedades de serviço, foram mantidos. Nenhum deles foi removido visto que todos os apresentados no Algoritmo 09 estavam relacionados a características obrigatórias no modelo de características (Figura 25 e Figura 27). Entretanto, houve outros elementos removidos no restante desse artefato, que não os apresentados aqui, por serem opcionais/alternativos e não terem sido selecionados durante a configuração dos modelos. Assim como nos casos anteriores, as anotações do contrato foram removidas.

Código 11: Contrato eletrônico – seção `<bpel:Process>`, elementos `<partnerLink>` e `<variable>`.

```
<bpel:partnerLinks>
  <bpel:partnerLink name="acoes-de-cobranca"
    partnerLinkType="ns1:acoes-de-cobrancaPLT"
    partnerRole="acoes-de-cobrancaROLE"/>
  (...)
  <bpel:partnerLink myRole="OS-para-executar-acao-de-cobrancaROLE"
    name="OS-para-executar-acao-de-cobranca"
    partnerLinkType="ns1:OS-para-executar-acao-de-cob (...) "/>
  (...)
</bpel:partnerLinks>

<bpel:variables>
  <bpel:variable messageType="ns1:aplicacao-de-acao-de-cobranca-Req-MSG"
    name="aplicacao-de-acao-de-cobranca-Request"/>

  <bpel:variable messageType="ns1:aplicacao-de-acao-de-cobranca-Res-MSG"
    name="aplicacao-de-acao-de-cobranca-Response"/>
  (...)
  <bpel:variable messageType="ns1:registro-de-OS-para-executar-acao-de (...) "
    name="registro-de-OS-para-executar-acao-de-cobranca (...) "/>

  <bpel:variable messageType="ns1:registro-de-OS-para-executar-acao-de (...) "
    name="registro-de-OS-para-executar-acao-de-cobranca (...) "/>
  (...)
</bpel:variables>
```

6.3 Análise do Grau de Variabilidade

Nesta seção é apresentada uma métrica relacionada ao grau de variabilidade no contexto desse estudo de caso. Grau de variabilidade é usado aqui como a quantidade possível de combinações de características relacionadas a serviços eletrônicos existentes em um modelo de características, para um determinado domínio de contrato. Ela serve para estimar a potencial efetividade da abordagem baseada em modelos de características para o estabelecimento de contratos eletrônicos. Quanto maior a quantidade de pontos variáveis em um determinado domínio e, conseqüentemente, de combinações possíveis de tais pontos, maior a complexidade associada a sua gerência para o estabelecimento de contratos similares. Sendo assim, quanto maior essas quantidades, maior a necessidade de uso de uma abordagem sistemática para sua gerência, tal como a proposta nesta tese.

Duas medidas relacionadas são apresentadas na Tabela 05: as quantidades de características obrigatórias e opcionais/alternativas e as quantidades de possíveis configurações que podem ser derivadas durante as configurações dos modelos de características. As quantidades são agrupadas tanto por serviços eletrônicos e quanto por atributos de QoS, para ambos o Sistema COB e o Sistema CRM. Os valores apresentados na tabela são relativos aos artefatos gerados para o estudo de caso apresentado neste capítulo (Figura 25, Figura 26, Figura 27 e Figura 28). A quantidade de configurações possíveis é obtida com o apoio da ferramenta FeaturePlugin, conforme apresentado na Seção 5.2.1, Figura 20.

As medidas associadas aos atributos de QoS são apresentadas de forma isolada para o conjunto de características definidas abaixo da característica raiz **atributos-de-qos**. Esse conjunto de características é associado a cada uma das características do tipo `serviço eletrônico` e em cada um desses casos uma configuração diferente é possível. Portanto, o valor apresentado é exponencialmente maior se considerado todas as possíveis combinações dessas características para cada uma das características do tipo `serviço eletrônico` às quais elas estão associadas.

Tabela 05: Medidas de Variabilidade.

Modelo de características para Serviços Eletrônicos	Tipos de Características		Configurações Possíveis
	Obrigatórias	Opcionais/Alternativas	
Sistema COB (Serviços eletrônicos)	19	23	288.360
Sistema CRM (Serviços eletrônicos)	11	4	20
Sistema COB (atributos de QoS)	6	35	12.500
Sistema CRM (atributos de QoS)	3	17	100
Total	39	79	300.980

Embora essas medidas não possam ser vistas como dados estatísticos, já que foram obtidas de um único espaço amostral, elas oferecem bons indicativos da potencial efetividade da abordagem proposta. As quantidades de possíveis combinações podem ser consideradas bastante altas para serem gerenciadas sem o apoio da abordagem proposta aqui. Se tais quantidades fossem pequenas, a aplicação da abordagem não seria justificada do ponto de vista custo/benefício. Sendo assim, a métrica apresentada mostra o potencial de tal abordagem para um domínio específico de contrato eletrônico que possua vários pontos de variabilidade e, conseqüentemente, leve a uma grande quantidade de diferentes configurações possíveis. Considera-se razoável extrapolar esse entendimento para outros domínios de contrato eletrônico, embora possam provavelmente existir outros domínios de contrato para os quais a aplicação dessa abordagem poderia não ser vantajosa.

Como os dois sistemas do estudo de caso possuem papéis diferentes, e bem definidos, dentro da cooperação interorganizacional (fornecedor e consumidor de serviços eletrônicos), as métricas apresentadas para cada caso possuem considerável diferença. Como o Sistema CRM está atuando com o papel de consumidor de serviços, é natural que ele tenha um número menor de características opcionais/alternativas – em relação à

quantidade total de características – que o Sistema COB que está atuando com o papel de fornecedor de serviços.

Apesar das diferenças existentes nas métricas individuais, a quantidade acumulada mostra que esse tipo de domínio de contrato eletrônico apresenta um grande grau de variabilidade. Conseqüentemente, a abordagem proposta pode ser considerada útil e efetiva já que explora os pontos de variabilidade com o objetivo de reuso de informações entre diferentes contratos.

6.4 Análise dos Resultados do Estudo de Caso

O principal resultado da realização do estudo de caso foi demonstrar que a abordagem proposta é realmente possível de ser utilizada, inclusive com o apoio automatizado do conjunto de ferramentas desenvolvido. Além desse resultado direto, outros resultados podem ser obtidos por meio de uma análise de sua completa execução. Alguns desses resultados já tinham sido obtidos inicialmente durante a definição da abordagem proposta, sendo confirmados com a realização do estudo de caso. Certamente, a própria abordagem foi em alguns pontos ajustada durante a execução do estudo de caso que mostrou determinadas fragilidades nela, embora a abordagem aqui apresentada seja resultante das avaliações dos resultados do estudo de caso.

Esta seção apresenta uma análise dos principais resultados obtidos com a realização do estudo de caso, além da própria demonstração de aplicabilidade da abordagem.

- **Representação adequada de serviços eletrônicos por modelos de características:** os modelos de características foram considerados adequados para a representação de informações relativas a serviços eletrônicos e a atributos e níveis de QoS. A flexibilidade do meta-modelo de características baseado em cardinalidade permitiu o uso de regras particulares para a especificação dessas informações. As facilidades de modularização foram particularmente úteis para a

especificação de atributos de QoS, que puderam ser definidos apenas uma vez, mas reaproveitado para todos os serviços eletrônicos durante a configuração dos modelos. Mesmo se os artefatos produzidos não forem reusados, a aplicação de modelos de características a serviços eletrônicos já se mostra útil pela representação estruturada de elementos de contratos obrigatórios e opcionais/alternativos. Com o apoio automatizado da ferramenta usada, foi possível acrescentar novas propriedades no modelo de características e estender o meta-modelo usado nesta abordagem.

- **Completitude do meta-modelo de contrato eletrônico para serviços Web:** o meta-modelo definido inclui os itens mais relevantes requeridos em um contrato eletrônico para serviços Web. Sua estrutura bem definida mostra o potencial reuso para o domínio de aplicação relativo a contratos eletrônicos. As linguagens WS-BPEL e WS-Agreement foram consideradas bastante completas e consistentes, simplificando a criação de moldes de contrato eletrônico. Um benefício intermediário bastante importante desta tese é a apresentação de um único meta-modelo de contrato eletrônico para serviços Web que integra conceitos existentes em ambas as linguagens WS-BEPL e WS-Agreement, além de WSDL. Esse tipo de meta-modelo pode ser considerado original visto que não foi identificado nenhum outro similar já apresentado na literatura relacionada.
- **Facilidade de mapeamento entre meta-modelos:** foi possível identificar um mapeamento direto e natural entre elementos no meta-modelo de características e no meta-modelo de contrato eletrônico para serviços Web. A definição desse mapeamento foi facilitada dado que ambos os meta-modelos usam linguagens baseadas em XML. Algumas regras foram definidas para mapear os diferentes tipos de características, representando informações relativas a serviços eletrônicos e atributos de QoS, a elementos nas seções do contrato eletrônico. Essas regras foram

avaliadas como adequadas durante o desenvolvimento do conjunto de ferramentas e a realização do estudo de caso.

- **Melhor estruturação e reuso de informações envolvidas:** a realização do estudo de caso demonstrou que o uso de modelos de características durante o estabelecimento de contratos eletrônicos, os torna mais simples, sistemáticos e de fácil compreensão. A abordagem proposta melhora o reuso de informações e artefatos, permitindo o estabelecimento mais rápido de novos contratos. Além disso, pontos comuns e variabilidades oferecidas por modelos de características representam serviços eletrônicos e atributos de QoS de um modo estruturado e controlado. Diferentes envolvidos no processo de estabelecimento de contratos, de diferentes níveis, podem se beneficiar da abordagem proposta. Os modelos de características para serviços eletrônicos podem ser entendidos como um espaço de configuração de contratos eletrônicos; dependendo nas escolhas feitas durante as configurações desses modelos, contratos eletrônicos diferentes e particulares são estabelecidos.
- **Aplicabilidade para contratos eletrônicos complexos:** a abordagem proposta é útil mesmo no estabelecimento de contratos eletrônicos complexos, visto que o meta-modelo de características usado para representar serviços eletrônicos pode ser facilmente particionado. Portanto, um contrato eletrônico para serviços Web complexo pode ser naturalmente estabelecido a partir de uma série de pequenos modelos de características para serviços eletrônicos – agrupados em apenas dois modelos de características, por meio do mecanismo de modularização.

Além dos pontos fortes da abordagem proposta apresentados anteriormente, existem também algumas desvantagens e limitações associadas à sua aplicação. Esses pontos fracos são apresentados a seguir.

- **Necessidade de conhecimento de modelagem de características:** para que a abordagem proposta nesta tese possa ser aplicada, é imprescindível que a grande maioria de seus usuários conheça modelos de características. Embora ela não seja considerada uma técnica complexa e sua aplicação possua um conjunto de benefícios, ainda assim é um conceito adicional que precisa ser dominado, além dos conceitos relacionados diretamente a contratos eletrônicos.
- **Cobertura não extensiva de tipos de elementos de contratos eletrônicos:** embora os principais tipos de elementos básicos de contratos eletrônicos estejam cobertos na abordagem proposta aqui, existem alguns tipos de elementos adicionais que não estão cobertos. Alguns exemplos são: informações associadas ao pagamento realizado pela organização consumidora para a organização fornecedora; validade do contrato; e aspectos legais.
- **Falta de apoio a acordos entre mais do que duas partes:** a abordagem oferece apoio ao estabelecimento de contratos eletrônicos entre apenas duas organizações. Embora esse seja o cenário mais comum e acordos entre mais do que duas partes podem ser compostos por um agrupamento de vários acordos entre duas partes, a abordagem não pode ser aplicada diretamente sem ajustes para acordos entre três ou mais organizações.
- **Modelo estático de negociação:** a abordagem proposta cobre apenas as situações em que as organizações parceiras são conhecidas logo no início do processo de estabelecimento de contrato assim como os serviços Web a serem contratados. Embora o modelo estático possa ser considerado bastante comum para um amplo conjunto de domínios de contrato, essa abordagem não cobre as situações em que as organizações fornecedoras de serviços e os próprios serviços Web são selecionados apenas durante a execução do processo de negócio interorganizacional.

- **Ausência de protocolo de negociação:** a abordagem apresenta nesta tese não inclui a especificação de um protocolo de negociação a ser seguido entre as organizações envolvidas. Considera-se que elas devem entrar em acordos para o estabelecimento dos contratos, o que deve ser feito com o apoio das configurações de modelos de características. Entretanto, não há a definição de papéis e as responsabilidades que cada organização envolvida deve seguir. Apenas o processo global de estabelecimento de contratos eletrônicos é tratado, em que a negociação é uma dos sub-processos;
- **Heterogeneidade das ferramentas que englobam o conjunto FeatureContract:** embora cada uma das ferramentas que fazem parte do conjunto de ferramentas desenvolvido para esta abordagem tenha apoiado adequadamente diferentes estágios do processo de estabelecimento de contratos, elas são bastante diferentes entre si o que dificulta a integração automatizada entre elas. Algumas diferenças existem em termos de forma de manipulação de artefato de entrada e armazenamento de artefato de saída e de interação com o usuário.

Algumas destas limitações serão discutidas na seção de trabalhos futuros do capítulo de conclusões. Porém, algumas das desvantagens apresentadas são intrínsecas à abordagem proposta e devem ser consideradas como o custo a ser pago para a obtenção dos benefícios oferecidos.

6.5 Considerações Finais

Neste capítulo, um estudo de caso foi apresentado com o objetivo de demonstrar a aplicabilidade da abordagem proposta nesta tese. Tal aplicabilidade foi demonstrada por meio da completa execução da abordagem proposta. Neste capítulo foram apresentados

exemplos de todos os artefatos produzidos por meio do estudo de caso realizado. Os exemplos relacionados a um tipo de integração factível de existir no mundo real mostram que a abordagem proposta pode ser aplicada. O uso do conjunto de ferramentas FeatureContract, no estudo de caso, demonstrou que seu uso realmente apóia a abordagem proposta.

Além da aplicação direta da abordagem definida, foi apresentada também uma análise do grau de variabilidade existente nesse estudo de caso. Essa métrica indica que existe um grande potencial de uso para a abordagem proposta, o que poderia ser confirmado com uma série de aplicações da abordagem em outros estudos de casos similares. Visto que não foram identificadas outras abordagens que fazem uso de contatos eletrônicos de uma forma sistemática como a proposta nesta tese, a comparação com outros trabalhos é limitada. Além disso, mecanismos avançados de estruturação e reuso de informações também não são encontrados em outras abordagens similares.

Por fim, foi apresentado um conjunto de análise dos resultados obtidos com a execução do estudo de caso. A análise dos resultados obtidos possibilitou a conclusão dos seguintes itens tais como: adequação da representação de serviços eletrônicos por modelos de características; proposição de um meta-modelo completo de contrato eletrônico para serviços Web; adequação do mapeamento entre meta-modelos de características e de contrato eletrônico; e demonstração de vantagens e benefícios da abordagem proposta para diferentes grupos de usuários.

No próximo capítulo são apresentadas as conclusões gerais desta tese, incluindo um conjunto das principais lições aprendidas bem como uma lista de trabalhos futuros a serem desenvolvidos.

Capítulo 7

Conclusões

Com o objetivo de facilitar a realização de processos de negócio interorganizacionais, esta tese apresenta uma abordagem para o estabelecimento de contratos eletrônicos. O uso de contratos é imprescindível para a definição formal dos direitos e deveres de cada uma das organizações envolvidas em uma cooperação eletrônica, porém sua elaboração pode ser complexa e apresentar um custo muito elevado. Dessa forma, é importante que haja apoio técnico e metodológico para que novos contratos eletrônicos sejam estabelecidos de forma mais ágil, com menores custos e prazos envolvidos.

Atualmente, a cooperação eletrônica entre diversas organizações é realizada por meio de um paradigma de subcontratação de serviços eletrônicos. Nesse paradigma, existem organizações fornecedoras e organizações consumidoras de serviços eletrônicos, que devem satisfazer determinados níveis de qualidade de serviço. As parcerias de negócio são criadas em mercados eletrônicos virtuais, similares a mercados tradicionais que lidam com serviços não eletrônicos. No caso dos mercados eletrônicos, os processos de negócio são realizados com o apoio de tecnologias tais como Internet, Sistemas Gerenciadores de Processos de Negócio (SGPN) e Computação Orientada por Serviços (COS).

Nesse contexto tecnológico, a abordagem apresentada nesta tese enfatiza o estabelecimento de contratos eletrônicos para serviços Web. Apesar disso, ela pode ser entendida como uma abordagem genérica, podendo ser aplicada a qualquer tipo de serviço eletrônico. O direcionamento a serviços Web foi considerado por se tratar do tipo mais promissor de serviços eletrônicos na atualidade. Uma série de linguagens e ferramentas já

disponíveis para serviços Web facilitou a definição e a experimentação da abordagem proposta. Em termos de linguagens, foram usadas WS-BPEL e WS-Agreement para a especificação de processos de negócio e atributos de QoS, respectivamente.

A abordagem proposta consiste basicamente de um processo de cinco estágios, envolvendo a criação sistemática de moldes de contratos eletrônicos e a criação de suas instâncias. Ambas as criação de molde e a criação de suas instâncias são orientadas por características representando serviços eletrônicos, atributos e níveis de QoS e outras informações relacionadas. Essas características são organizadas em modelos de características, por meio de uma abordagem de engenharia de software usada para a representação de propriedades obrigatórias, opcionais e alternativas para um determinado domínio de aplicação. É por meio do uso desses modelos, que a abordagem proposta visa oferecer um melhor nível de reuso e estruturação de informações comparando com outras abordagens existentes para o mesmo fim.

Diferentes tipos de grupo das organizações envolvidas recebem benefício da aplicação desta abordagem, incluindo tanto grupos de negócio quanto grupos de desenvolvimento. A utilidade para grupos de negócio existe durante o processo de negociação para o estabelecimento do contrato devido ao seu alto nível de abstração. A visão estruturada dos serviços eletrônicos oferecidos pelos modelos de características melhora o entendimento de cada parte. Para os arquitetos de integração de sistemas, a utilidade da abordagem existe durante o refinamento dos detalhes dos serviços eletrônicos a serem desenvolvidos. Provedores de serviços Web podem também se beneficiar da visão estruturada dos serviços eletrônicos para melhorar o planejamento, a análise e o projeto dos serviços Web contratados antes que o desenvolvimento seja realizado.

Os cinco estágios do processo de estabelecimento de contratos eletrônicos foram definidos com base em um processo específico para linha de produto de software. LP é uma abordagem que objetiva o desenvolvimento de diversos produtos de software diferentes, porém com uma estrutura básica comum. A estrutura básica é identificada por pontos comuns entre os diversos produtos, enquanto que as diferenças são identificadas

por variabilidades. Os modelos de características são usados para representar tais pontos comuns e variabilidades, tanto no processo de LP como na abordagem apresentada aqui.

Os três primeiros estágios estão relacionados ao desenvolvimento genérico do molde de contrato; enquanto que os dois últimos estão relacionados ao desenvolvimento específico de diferentes contratos finais únicos. Dois modelos de características, representando as informações sobre serviços eletrônicos e os atributos de QoS, são produzidos no Estágio 1. Esses modelos orientam a elaboração, parcialmente automática, do molde de contrato no Estágio 2. Similarmente, vários pares de configuração dos mesmos modelos de características são criados no Estágio 4, representando subconjuntos de tais modelos. Esses pares de configuração orientam a criação automática de instâncias de contrato eletrônico resultantes no Estágio 5. Durante o Estágio 3, serviços Web são desenvolvidos e publicados para serem referenciados nos moldes e nos contratos eletrônicos produzidos.

A principal contribuição da abordagem proposta é permitir uma melhor gerência dos pontos comuns e das variabilidades existentes em contratos eletrônicos para serviços Web similares. Incluem-se, nesse aspecto, a estruturação e o reuso sistemáticos de informações. A execução de alguns estágios do processo pode ser vista como parte de uma abordagem mais abrangente de negociação para o estabelecimento de contrato eletrônico, no qual as partes envolvidas estão contratando serviços eletrônicos, de acordo com determinados limites para atributos de QoS. Esta abordagem pode ser vista como uma LP de contrato eletrônico.

Além da abordagem em si, foi proposto um apoio computacional para sua aplicação. Para isso, o conjunto de ferramentas, chamado FeatureContract, oferece apoio a diferentes estágios do processo definido. Para demonstrar a aplicabilidade da abordagem, incluindo o uso do conjunto de ferramentas desenvolvido, foi realizado um estudo de caso cujos resultados também foram apresentados nesta tese. Uma análise sobre o grau de variabilidade apresentado nos modelos de características desenvolvidos no estudo de caso indica uma boa aplicabilidade dessa abordagem.

Com a realização deste trabalho e o estudo de caso, uma série de resultados foi obtida. A análise desses resultados apresentou um conjunto de vantagens e benefícios associados à abordagem proposta como também algumas desvantagens e limitações associadas à mesma. Como vantagens e benefícios destacam-se: representação adequada de serviços eletrônicos por modelos de características; proposição de um meta-modelo completo de contrato eletrônico para serviços Web; facilidade de mapeamento entre meta-modelos; melhor estruturação e reuso de informações envolvidas; e aplicabilidade para contratos eletrônicos complexos. As desvantagens e limitações são as seguintes: necessidade de conhecimento de modelagem de características; cobertura não extensiva de tipos de elementos de contratos eletrônicos; falta de apoio a acordos entre mais do que duas partes; modelo estático de negociação; ausência de um protocolo de negociação bem definido entre as partes envolvidas; e heterogeneidade das ferramentas que englobam o conjunto FeatureContract.

7.1 Trabalhos Futuros

Esta seção apresenta os principais trabalhos futuros a serem desenvolvidos como continuação do trabalho apresentado nesta tese.

- **Extensão da abordagem proposta:** o domínio de aplicação relacionado a contratos eletrônicos é bastante amplo, sendo que uma parte relativamente grande desse domínio foi coberta na abordagem proposta nesta tese. Ainda assim, essa abordagem pode ser estendida para incorporar outras propriedades ou funcionalidades relacionadas a contratos eletrônicos para serviços Web ainda não cobertas. Exemplos são:

- Representação, pelos modelos de características, de outras informações associadas a serviços eletrônicos e atributos de QoS que podem ser úteis ao processo – como detalhes sobre os parâmetros de entrada/saída dos serviços eletrônicos;
 - Mecanismos associados a: publicação, busca e descoberta de serviços Web;
 - Aplicação do modelo dinâmico ao processo de negociação e estabelecimento de contrato eletrônico – além do modelo estático;
 - Extensão da abordagem para ser aplicável a mais do que duas organizações envolvidas por vez;
 - Aplicação da abordagem na geração de modelos de características e moldes de contratos a serem comercializados como “software de prateleira”;
 - Definição de um protocolo de negociação em que os papéis e as responsabilidades são bem definidos por meio de atividades sistemáticas que cada organização envolvida deve seguir.
- **Ferramenta FeatureContract:** o conjunto de ferramentas apresentado nesta tese foi desenvolvido de forma limitada com o objetivo de atender às necessidades pontuais para a realização do estudo de caso proposto. Para esse objetivo, ela foi considerada adequada. Porém, a ferramenta pode ser complementada de forma a deixá-la mais operacional e contendo mais funcionalidades. Em relação à intervenção do usuário, três pontos principais deveriam ser tratados com mais prioridade: 1) possibilitar a integração completamente automática entre as diversas ferramentas existentes; 2) melhorar os mecanismos disponibilizados para a interação do usuário com essas ferramentas; e 3) permitir que pelo menos parte da seção relativa ao processo de negócio (as partes relacionadas a variáveis e a ligações entre parceiros) já seja também criada automaticamente. Particularmente, as duas ferramentas desenvolvidas especificamente para esta tese – XSLTransformer e WS-Contract Factory – precisam ser aprimoradas; porém mesmo as outras ferramentas

incorporadas por FeatureContract precisam ser ajustadas para permitir melhor adequação ao contexto da abordagem proposta. Um exemplo de tal adequação é diferenciar nas representações gráficas elementos que são obrigatórios e opcionais/alternativas, possivelmente por meio de um esquema de cores.

- **Análise da abordagem proposta:** embora a aplicabilidade da abordagem proposta tenha sido avaliada por meio da realização de um estudo de caso, novas análises podem ser realizadas visando ampliar essa avaliação do ponto de vista de viabilidade e efetividade. Particularmente, a comparação com outras abordagens similares para contratos eletrônicos poderia ser realizada, apesar da dificuldade existente visto que elas não oferecem mecanismos sistemáticos de reuso e estruturação de informações.
- **Ontologias computacionais:** considerando que ontologias computacionais e modelos de características possuem propriedades similares, é importante realizar uma avaliação detalhada dos complementos existentes entre as duas abordagens. O objetivo de pesquisa aqui é avaliar se existem reais possibilidades e vantagens em se estender os modelos de características usados nesta tese, para incorporar propriedades adicionais existentes em ontologias computacionais, tais como mecanismos de inferência. Essa análise pode ser tanto direcionada a aplicação de ambas no contexto de estabelecimento de contratos eletrônicos, como em um contexto mais geral – levando a outra linha de pesquisa independente.

Apêndice A

Modelos de Características

Este apêndice apresenta os dois modelos de características completos que foram elaborados durante o desenvolvimento do estudo de caso apresentado nesta tese. Primeiramente, é apresentado o modelo de características referente ao Sistema COB; em seguida, o modelo de características referente ao sistema CRM.

A.1 Sistema COB

Nas figuras a seguir (Figura 44 a Figura 47) são apresentadas as partes que juntas integram o modelo de características do sistema COB. As duas primeiras figuras apresentam as partes relativas à sub-árvore **servicos-eletronicos**; enquanto que as duas últimas apresentam as partes relativas à sub-árvore **atributos-de-qos**. As partes do modelo são apresentadas sempre completas, porém com algumas características não expandidas, alternando-se entre as figuras. As características referentes aos atributos de QoS são apresentadas de forma expandida apenas na parte relativa a sua definição principal (Figura 46 e Figura 47); nos seus usos por meio de modularização associadas aos serviços eletrônicos (Figura 44 e Figura 46), elas são apresentadas de forma contraída.



Figura 44: Modelo de Características – Sistema COB (Serviços Eletrônicos / Parte 1).

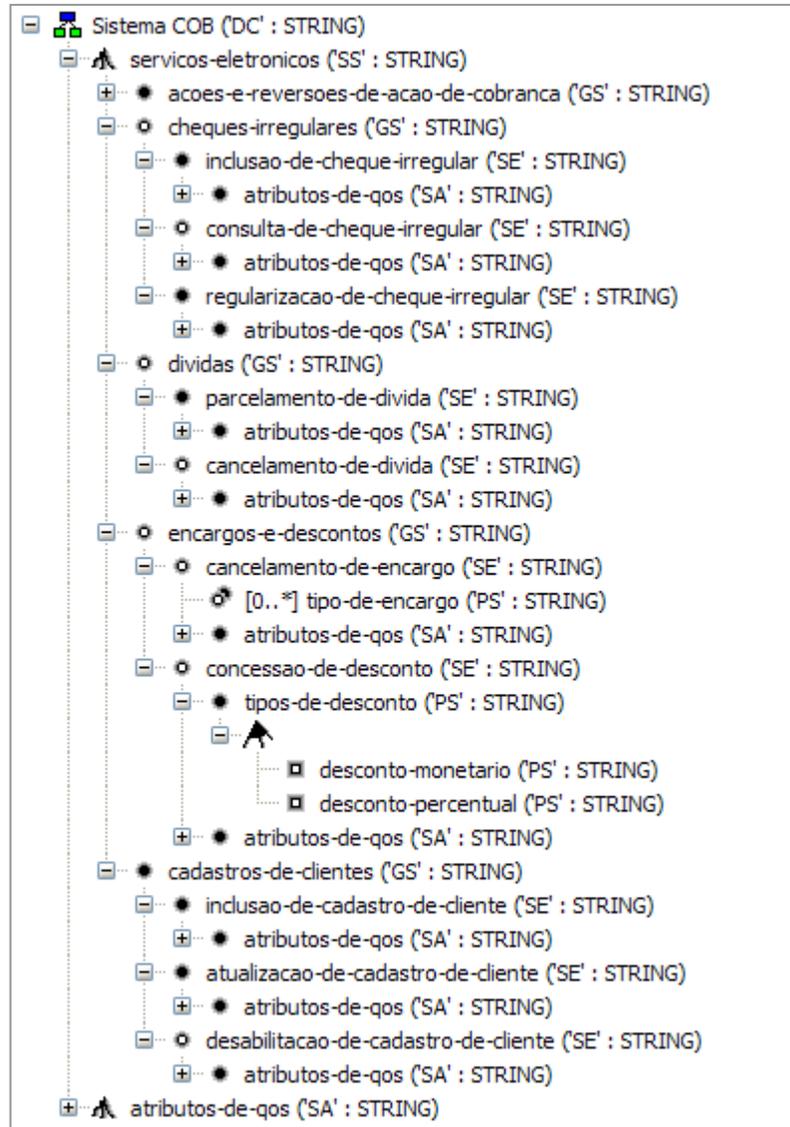


Figura 45: Modelo de Características – Sistema COB (Serviços Eletrônicos / Parte 2).

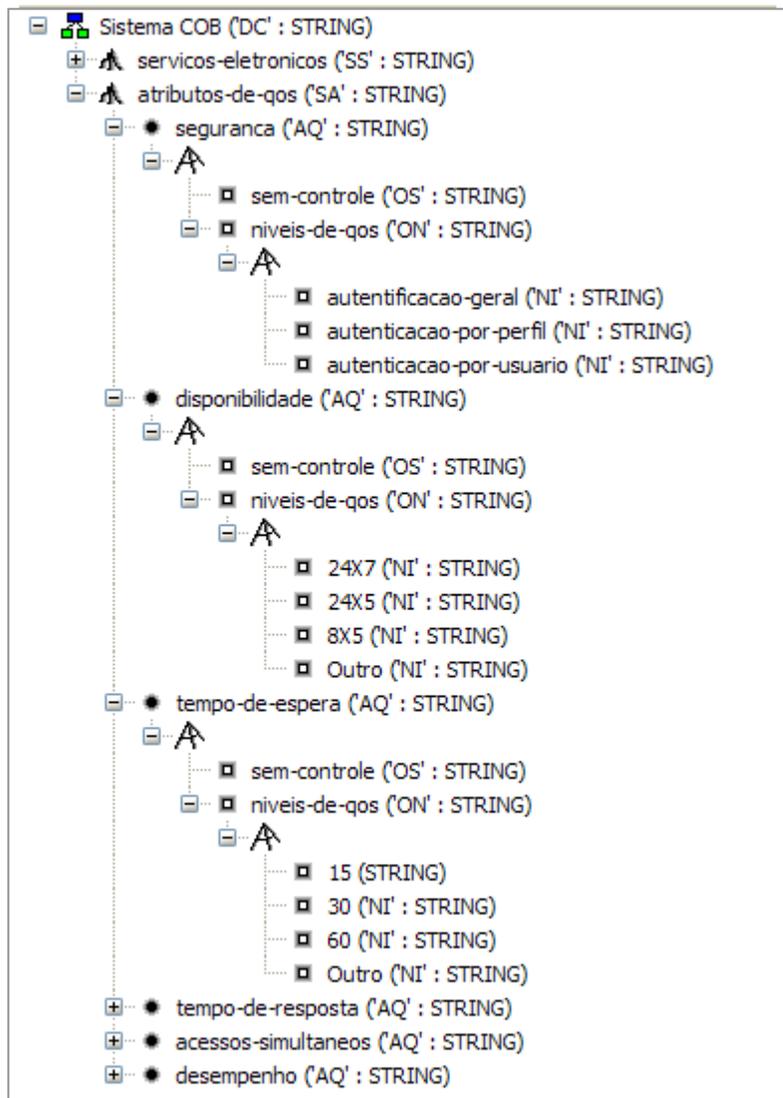


Figura 46: Modelo de Características – Sistema COB (Atributos de QoS / Parte 1).

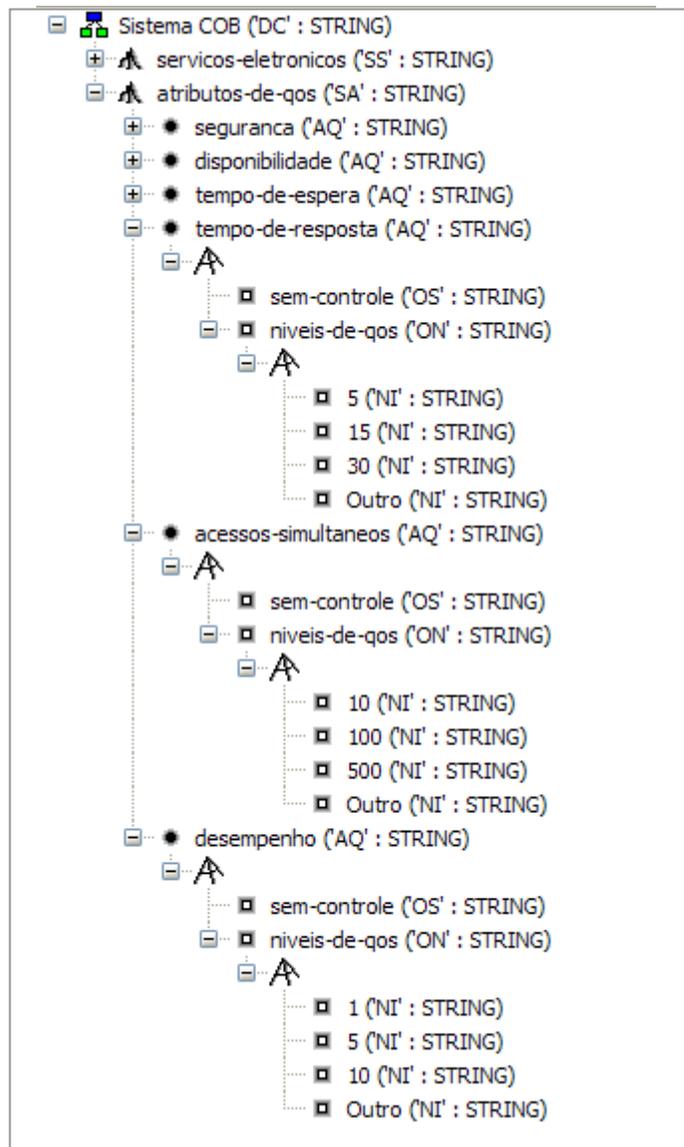


Figura 47: Modelo de Características – Sistema COB (Atributos de QoS / Parte 2).

A.2 Sistema CRM

Nas figuras a seguir (Figura 48 e Figura 49) são apresentadas as partes que juntas integram o modelo de características do sistema CRM. Similarmente à seção anterior, a primeira figura apresenta a parte relativa à sub-árvore **servicos-eletronicos**; enquanto que a segunda apresenta as partes relativa à sub-árvore **atributos-de-qos**.

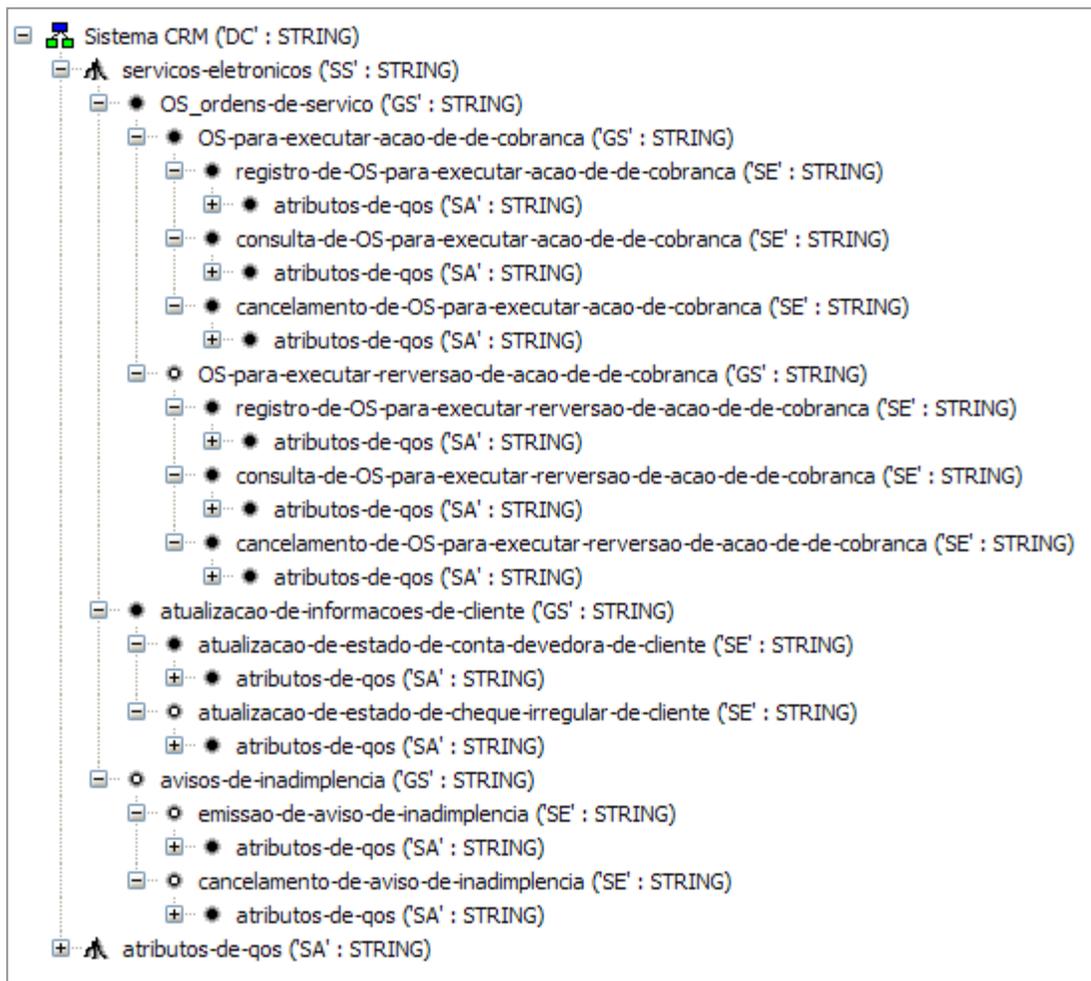


Figura 48: Modelo de Características – Sistema CRM (Serviços Eletrônicos).

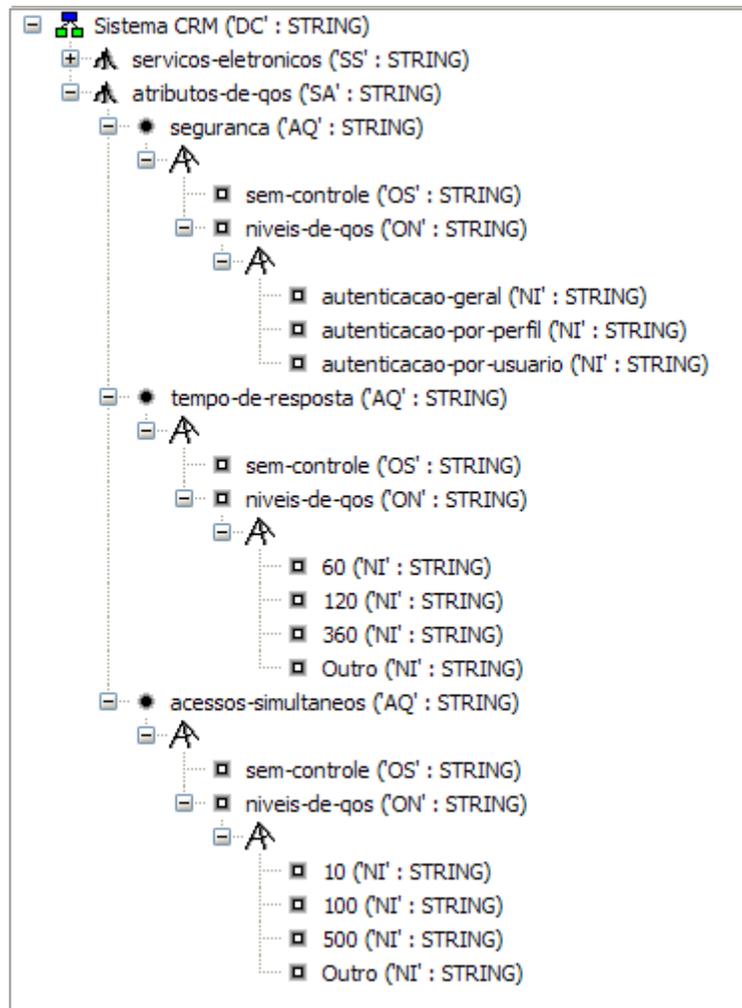


Figura 49: Modelo de Características – Sistema CRM (Atributos de QoS).

Uma Abordagem Baseada em Características para o Estabelecimento de Contratos Eletrônicos para Serviços Web

Este exemplar corresponde à redação final da Tese devidamente corrigida e defendida por Marcelo Fantinato e aprovada pela Banca Examinadora.

Campinas, 10 de dezembro de 2007.

Maria Beatriz Felgar de Toledo (Orientadora)

Itana Maria de Souza Gimenes (Co-orientadora)

Tese apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de doutor em Ciência da Computação.

Apêndice B

Molde de Contrato Eletrônico

Este apêndice apresenta o molde de contrato eletrônico completo que foi gerado durante o desenvolvimento do estudo de caso apresentado nesta tese. O molde de contrato é apresentado dividido em suas três seções principais: Seção de definições WSDL, Seção de termos de WS-Agreement e Seção de processo de negócio WS-BPEL. Quando possível, essas informações são apresentadas apenas por meio de visão gráfica, visto que as especificações XML ocupariam espaço excessivo.

B.1 Seção de Definições WSDL

Nas figuras a seguir (Figura 50 a Figura 52) são apresentadas as partes que juntas integram a seção de definições WSDL do molde de contrato eletrônico. A Figura 50 e a Figura 51 apresentam as partes relativas ao sistema COB, enquanto que a Figura 52 apresenta a parte relativa ao sistema CRM. As anotações ligando os elementos com as características apresentadas no apêndice anterior não são, portanto, apresentadas aqui. O Código 12 e o Código 13 apresentam os trechos dessas seções referentes à especificação dos elementos Partner Link Types, que não podem ser vistos pela visão gráfica.

acoes-de-cobrancaPT		
aplicacao-de-acao-de-cobrancaOP		
input	aplicacao-de-acao-de-cobranca-Request-MSG-PART	string
	tipos-de-acao-de-cobranca-MSG-PART	string
output	aplicacao-de-acao-de-cobranca-Response-MSG-PART	string
atualizacao-de-estado-de-acao-de-cobrancaOP		
input	atualizacao-de-estado-de-acao-de-cobranca-Request-MSG-PART	string
output	atualizacao-de-estado-de-acao-de-cobranca-Response-MSG-PART	string
cancelamento-de-aplicacao-de-acao-de-cobrancaOP		
input	cancelamento-de-aplicacao-de-acao-de-cobranca-Request-MSG-PART	string
output	cancelamento-de-aplicacao-de-acao-de-cobranca-Response-MSG-PART	string

reversoes-de-acao-de-cobrancaPT		
aplicacao-de-reversao-de-acao-de-cobrancaOP		
input	aplicacao-de-reversao-de-acao-de-cobranca-Request-MSG-PART	string
output	aplicacao-de-reversao-de-acao-de-cobranca-Response-MSG-PART	string
atualizacao-de-estado-de-reversao-de-acao-de-cobrancaOP		
input	atualizacao-de-estado-de-reversao-de-acao-de-cobranca-Request-MSG-PART	string
output	atualizacao-de-estado-de-reversao-de-acao-de-cobranca-Response-MSG-PART	string
cancelamento-de-aplicacao-de-reversao-de-acao-de-cobrancaOP		
input	cancelamento-de-aplicacao-de-reversao-de-acao-de-cobranca-Request-MSG-PART	string
output	cancelamento-de-aplicacao-de-reversao-de-acao-de-cobranca-Response-MSG-PART	string

consulta-de-historico-de-acoes-e-reversoes-de-acao-de-cobrancaPT		
consulta-de-historico-de-acoes-de-cobranca-por-clienteOP		
input	consulta-de-historico-de-acoes-de-cobranca-por-cliente-Request-MSG-PART	string
output	consulta-de-historico-de-acoes-de-cobranca-por-cliente-Response-MSG-PART	string
consulta-de-historico-de-acoes-e-reversoes-de-acao-de-cobranca-por-clienteOP		
input	consulta-de-historico-de-acoes-e-reversoes-de-acao-de-cobranca-por-cliente-Request-MSG-PART	string
output	consulta-de-historico-de-acoes-e-reversoes-de-acao-de-cobranca-por-cliente-Response-MSG-PART	string

Figura 50: Molde de Contrato Eletrônico – Seção WSDL (Sistema COB – Parte 1).

cheques-irregularesPT		
inclusao-de-cheque-irregularOP		
input	inclusao-de-cheque-irregular-Request-MSG-PART	string
output	inclusao-de-cheque-irregular-Response-MSG-PART	string
consulta-de-cheque-irregularOP		
input	consulta-de-cheque-irregular-Request-MSG-PART	string
output	consulta-de-cheque-irregular-Response-MSG-PART	string
regularizacao-de-cheque-irregularOP		
input	regularizacao-de-cheque-irregular-Request-MSG-PART	string
output	regularizacao-de-cheque-irregular-Response-MSG-PART	string

dividasPT		
parcelamento-de-dividaOP		
input	parcelamento-de-divida-Request-MSG-PART	string
output	parcelamento-de-divida-Response-MSG-PART	string
cancelamento-de-dividaOP		
input	cancelamento-de-divida-Request-MSG-PART	string
output	cancelamento-de-divida-Response-MSG-PART	string

encargos-e-descontosPT		
cancelamento-de-encargoOP		
input	cancelamento-de-encargo-Request-MSG-PART	string
	tipo-de-encargo-MSG-PART	string
output	cancelamento-de-encargo-Response-MSG-PART	string
concessao-de-descontoOP		
input	concessao-de-desconto-Request-MSG-PART	string
	tipos-de-desconto-MSG-PART	string
output	concessao-de-desconto-Response-MSG-PART	string

cadastros-de-clientesPT		
inclusao-de-cadastro-de-clienteOP		
input	inclusao-de-cadastro-de-cliente-Request-MSG-PART	string
output	inclusao-de-cadastro-de-cliente-Response-MSG-PART	string
atualizacao-de-cadastro-de-clienteOP		
input	atualizacao-de-cadastro-de-cliente-Request-MSG-PART	string
output	atualizacao-de-cadastro-de-cliente-Response-MSG-PART	string
desabilitacao-de-cadastro-de-clienteOP		
input	desabilitacao-de-cadastro-de-cliente-Request-MSG-PART	string
output	desabilitacao-de-cadastro-de-cliente-Response-MSG-PART	string

Figura 51: Molde de Contrato Eletrônico – Seção WSDL (Sistema COB – Parte 2).

OS-para-executar-acao-de-de-cobrancaPT		
registro-de-OS-para-executar-acao-de-de-cobrancaOP		
input	registro-de-OS-para-executar-acao-de-de-cobranca-Request-MSG-PART	string
output	registro-de-OS-para-executar-acao-de-de-cobranca-Response-MSG-PART	string
consulta-de-OS-para-executar-acao-de-de-cobrancaOP		
input	consulta-de-OS-para-executar-acao-de-de-cobranca-Request-MSG-PART	string
output	consulta-de-OS-para-executar-acao-de-de-cobranca-Response-MSG-PART	string
cancelamento-de-OS-para-executar-acao-de-cobrancaOP		
input	cancelamento-de-OS-para-executar-acao-de-cobranca-Request-MSG-PART	string
output	cancelamento-de-OS-para-executar-acao-de-cobranca-Response-MSG-PART	string

OS-para-executar-rerversao-de-acao-de-de-cobrancaPT		
registro-de-OS-para-executar-rerversao-de-acao-de-de-cobrancaOP		
input	registro-de-OS-para-executar-rerversao-de-acao-de-de-cobranca-Request-MSG-PART	string
output	registro-de-OS-para-executar-rerversao-de-acao-de-de-cobranca-Response-MSG-PART	string
consulta-de-OS-para-executar-rerversao-de-acao-de-de-cobrancaOP		
input	consulta-de-OS-para-executar-rerversao-de-acao-de-de-cobranca-Request-MSG-PART	string
output	consulta-de-OS-para-executar-rerversao-de-acao-de-de-cobranca-Response-MSG-PART	string
cancelamento-de-OS-para-executar-rerversao-de-acao-de-de-cobrancaOP		
input	cancelamento-de-OS-para-executar-rerversao-de-acao-de-de-cobranca-Request-MSG-PART	string
output	cancelamento-de-OS-para-executar-rerversao-de-acao-de-de-cobranca-Response-MSG-PART	string

atualizacao-de-informacoes-de-clientePT		
atualizacao-de-estado-de-conta-devedora-de-clienteOP		
input	atualizacao-de-estado-de-conta-devedora-de-cliente-Request-MSG-PART	string
output	atualizacao-de-estado-de-conta-devedora-de-cliente-Response-MSG-PART	string
atualizacao-de-estado-de-cheque-irregular-de-clienteOP		
input	atualizacao-de-estado-de-cheque-irregular-de-cliente-Request-MSG-PART	string
output	atualizacao-de-estado-de-cheque-irregular-de-cliente-Response-MSG-PART	string

avisos-de-inadimplenciaPT		
emissao-de-aviso-de-inadimplenciaOP		
input	emissao-de-aviso-de-inadimplencia-Request-MSG-PART	string
output	emissao-de-aviso-de-inadimplencia-Response-MSG-PART	string
cancelamento-de-aviso-de-inadimplenciaOP		
input	cancelamento-de-aviso-de-inadimplencia-Request-MSG-PART	string
output	cancelamento-de-aviso-de-inadimplencia-Response-MSG-PART	string

Figura 52: Molde de Contrato Eletrônico – Seção WSDL (Sistema CRM).

Código 12: Molde de Contrato Eletrônico – Seção WSDL, Partner Link Type (Sistema COB).

```
<!--FEATURE_ID = acoesdecobranca-->
  <plnk:partnerLinkType name="acoes-de-cobrancaPLT">
    <plnk:role name="acoes-de-cobrancaROLE">
      <plnk:portType name="tns:acoes-de-cobrancaPT" />
    </plnk:role>
  </plnk:partnerLinkType>

<!--FEATURE_ID = reversoesdeacaodecobranca-->
  <plnk:partnerLinkType name="reversoes-de-acao-de-cobrancaPLT">
    <plnk:role name="reversoes-de-acao-de-cobrancaROLE">
      <plnk:portType name="tns:reversoes-de-acao-de-cobrancaPT" />
    </plnk:role>
  </plnk:partnerLinkType>

<!--FEATURE_ID = consultadeestadodeacoesdecobranca-->
  <plnk:partnerLinkType name="consulta-de-estado-de-acoes-e-reversoes-de-acao-de-cobrancaPLT">
    <plnk:role name="consulta-de-estado-de-acoes-e-reversoes-de-acao-de-cobrancaROLE">
      <plnk:portType name="tns:consulta-de-estado-de-acoes-e-reversoes-de-acao-de-cobrancaPT" />
    </plnk:role>
  </plnk:partnerLinkType>

<!--FEATURE_ID = consultadehistoricodeacoesereversoesdeacaodecobranca-->
  <plnk:partnerLinkType name="consulta-de-historico-de-acoes-e-reversoes-de-acao-de-cobrancaPLT">
    <plnk:role name="consulta-de-historico-de-acoes-e-reversoes-de-acao-de-cobrancaROLE">
      <plnk:portType name="tns:consulta-de-historico-de-acoes-e-reversoes-de-acao-de-cobrancaPT" />
    </plnk:role>
  </plnk:partnerLinkType>

<!--FEATURE_ID = chequesirregulares-->
  <plnk:partnerLinkType name="cheques-irregularesPLT">
    <plnk:role name="cheques-irregularesROLE">
      <plnk:portType name="tns:cheques-irregularesPT" />
    </plnk:role>
  </plnk:partnerLinkType>

<!--FEATURE_ID = dividas-->
  <plnk:partnerLinkType name="dividasPLT">
    <plnk:role name="dividasROLE">
      <plnk:portType name="tns:dividasPT" />
    </plnk:role>
  </plnk:partnerLinkType>

<!--FEATURE_ID = cadastrosdeclientes-->
  <plnk:partnerLinkType name="encargos-e-descontosPLT">
```

```

        <plnk:role name="encargos-e-descontosROLE">
            <plnk:portType name="tns:encargos-e-descontosPT" />
        </plnk:role>
    </plnk:partnerLinkType>

<!--FEATURE_ID = cadastrrosdeclientes0-->
    <plnk:partnerLinkType name="cadastrros-de-clientesPLT">
        <plnk:role name="cadastrros-de-clientesROLE">
            <plnk:portType name="tns:cadastrros-de-clientesPT" />
        </plnk:role>
    </plnk:partnerLinkType>
</wsdl:definitions>

```

Código 13: Molde de Contrato Eletrônico – Seção WSDL, Partner Link Type (Sistema CRM).

```

<!--FEATURE_ID = oSparaexecutaracaodedecobranca-->
    <plnk:partnerLinkType name="OS-para-executar-acao-de-de-cobrancaPLT">
        <plnk:role name="OS-para-executar-acao-de-de-cobrancaROLE">
            <plnk:portType name="tns:OS-para-executar-acao-de-de-cobrancaPT" />
        </plnk:role>
    </plnk:partnerLinkType>

<!--FEATURE_ID = chargeActionReversions-->
    <plnk:partnerLinkType name="OS-para-executar-rerversao-de-acao-de-de-
cobrancaPLT">
        <plnk:role name="OS-para-executar-rerversao-de-acao-de-de-
cobrancaROLE">
            <plnk:portType name="tns:OS-para-executar-rerversao-de-acao-de-de-
cobrancaPT" />
        </plnk:role>
    </plnk:partnerLinkType>

<!--FEATURE_ID = encargosedescontos-->
    <plnk:partnerLinkType name="atualizacao-de-informacoes-de-clientePLT">
        <plnk:role name="atualizacao-de-informacoes-de-clienteROLE">
            <plnk:portType name="tns:atualizacao-de-informacoes-de-clientePT"
/>
        </plnk:role>
    </plnk:partnerLinkType>

<!--FEATURE_ID = cadastrrosdeclientes-->
    <plnk:partnerLinkType name="avisos-de-inadimplenciaPLT">
        <plnk:role name="avisos-de-inadimplenciaROLE">
            <plnk:portType name="tns:avisos-de-inadimplenciaPT" />
        </plnk:role>
    </plnk:partnerLinkType>
</wsdl:definitions>

```

B.2 Seção de Termos de WS-Agreement

Nas figuras a seguir (Figura 53 a Figura 59) são apresentadas partes da seção de definições de termos WS-Agreement. Dada a extensão bastante grande destes artefatos, devido ao alto número de combinações entre serviços eletrônicos e atributos de QoS, apenas algumas partes representativas deles são apresentadas. Primeiramente, são apresentadas partes dessa seção referentes ao Sistema CRM (*Service Properties*, na Figura 53; e *Guarantee Terms*, na Figura 54 e na Figura 55). Na sequência, são apresentadas partes referentes ao Sistema COB (*Service Properties*, na Figura 56; e *Guarantee Terms*, na Figura 57, na Figura 58 e na Figura 59).

[-] [e] wsag:Terms	
[a] xmlns:wsag	http://schemas.ggf.org/graap/2005/09/ws-agreement/
!--	FEATURE_ID = actionApplication
[-] [e] wsag:ServiceProperties	
[a] wsag:Name	registro-de-OS-para-executar-acao-de-de-cobrancaSP
[a] wsag:ServiceName	registro-de-OS-para-executar-acao-de-de-cobrancaOP
[-] [e] wsag:VariableSet	
!--	FEATURE_ID = reference19IXIseguranca
[-] [e] wsag:Variable	
[a] wsag:Name	segurancaVAR
[a] wsag:Metric	type:constraint
[-] [e] wsag:Location	
!--	A ser completado
!--	FEATURE_ID = reference19IXItempoderesposta
[-] [e] wsag:Variable	
[a] wsag:Name	tempo-de-respostaVAR
[a] wsag:Metric	time:duranton
[-] [e] wsag:Location	
!--	A ser completado
!--	FEATURE_ID = reference19IXIdisponibilidade
[-] [e] wsag:Variable	
[a] wsag:Name	acessos-simultaneosVAR
[a] wsag:Metric	users:amount
[-] [e] wsag:Location	
!--	A ser completado
!--	FEATURE_ID = actionUpdate
[-] [e] wsag:ServiceProperties	

Figura 53: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Service Properties (Sistema CRM).

+	e	wsag:ServiceProperties	
		!--	FEATURE_ID = actionApplication
-	e	wsag:GuaranteeTerm	
		a	Obligated
			fornecedorDeServico
	-	e	wsag:ServiceScope
		a	ServiceName
			registro-de-OS-para-executar-acao-de-de-cobrancaOP
	-	e	wsag:QualifyingCondition
			!--
			A ser completado
	-	e	wsag:ServiceLevelObjective
			segurancaVAR IS-LESS-INCLUSIVE
			!--
			FEATURE_ID = reference 19IXIsemcontrole
			sem-controle
			!--
			FEATURE_ID = reference 19IXIautenticacaogeral
			autenticacao-geral
			!--
			FEATURE_ID = reference 19IXIautenticacaoporperfil
			autenticacao-por-perfil
			!--
			FEATURE_ID = reference 19IXIautenticacaoporusuario
			autenticacao-por-usuario
	-	e	wsag:BusinessValueList
			!--
			A ser completado
			!--
			FEATURE_ID = actionApplication
	-	e	wsag:GuaranteeTerm
		a	Obligated
			fornecedorDeServico
	-	e	wsag:ServiceScope
		a	ServiceName
			registro-de-OS-para-executar-acao-de-de-cobrancaOP
	-	e	wsag:QualifyingCondition
			!--
			A ser completado
	-	e	wsag:ServiceLevelObjective
			tempo-de-respostaVAR IS-LESS-INCLUSIVE
			!--
			FEATURE_ID = reference 19IXIsemcontrole0
			sem-controle
			!--
			FEATURE_ID = reference 19IXIa60
			60
			!--
			FEATURE_ID = reference 19IXIa120
			120
			!--
			FEATURE_ID = reference 19IXIa360
			360
			!--
			FEATURE_ID = reference 19IXIoutro0
			Outro
	-	e	wsag:BusinessValueList
			!--
			A ser completado
			!--
			FEATURE_ID = actionApplication
+	e	wsag:GuaranteeTerm	

Figura 54: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Guarantee Terms (Sistema CRM / Parte 1).

+	e	wsag:GuaranteeTerm	
	!	--	FEATURE_ID = actionApplication
-	e	wsag:GuaranteeTerm	
	a	Obligated	fornecedorDeServico
-	e	wsag:ServiceScope	
	a	ServiceName	registro-de-OS-para-executar-acao-de-de-cobrancaOP
-	e	wsag:QualifyingCondition	
	!	--	A ser completado
-	e	wsag:ServiceLevelObjective	
			acessos-simultaneosVAR IS-LESS-INCLUSIVE
	!	--	FEATURE_ID = reference 19IXIsemcontrole 1
			sem-controle
	!	--	FEATURE_ID = reference 19IXIa 10
			10
	!	--	FEATURE_ID = reference 19IXIa 100
			100
	!	--	FEATURE_ID = reference 19IXIa 500
			500
	!	--	FEATURE_ID = reference 19IXIoutro 1
			Outro
-	e	wsag:BusinessValueList	
	!	--	A ser completado
	!	--	FEATURE_ID = actionUpdate
+	e	wsag:GuaranteeTerm	

Figura 55: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Guarantee Terms (Sistema CRM / Parte 2).

[-] [e] wsag:Terms	
[a] xmlns:wsag	http://schemas.ggf.org/graap/2005/09/ws-agreement/
!--	FEATURE_ID = aplicacaodeacaodecobranca
[-] [e] wsag:ServiceProperties	
[a] wsag:Name	aplicacao-de-acao-de-cobrancaSP
[a] wsag:ServiceName	aplicacao-de-acao-de-cobrancaOP
[-] [e] wsag:VariableSet	
!--	FEATURE_ID = reference20IXIseguranca
[-] [e] wsag:Variable	
[a] wsag:Name	segurancaVAR
[a] wsag:Metric	type:constraint
[-] [e] wsag:Location	
!--	A ser completado
!--	FEATURE_ID = reference20IXIdisponibilidade
[-] [e] wsag:Variable	
[a] wsag:Name	disponibilidadeVAR
[a] wsag:Metric	hours_days:amount
[-] [e] wsag:Location	
!--	A ser completado
!--	FEATURE_ID = reference20IXItempodeespera
[-] [e] wsag:Variable	
[a] wsag:Name	tempo-de-esperaVAR
[a] wsag:Metric	time:duration
[-] [e] wsag:Location	
!--	A ser completado
!--	FEATURE_ID = reference20IXItempoderesposta
[-] [e] wsag:Variable	
[a] wsag:Name	tempo-de-respostaVAR
[a] wsag:Metric	time:duration
[-] [e] wsag:Location	
!--	A ser completado
!--	FEATURE_ID = reference20IXIacessosimultaneos
[-] [e] wsag:Variable	
[a] wsag:Name	acessos-simultaneosVAR
[a] wsag:Metric	users:amount
[-] [e] wsag:Location	
!--	A ser completado
!--	FEATURE_ID = reference20IXIdesempenho
[-] [e] wsag:Variable	
[a] wsag:Name	desempenhoVAR
[a] wsag:Metric	time:duration
[-] [e] wsag:Location	
!--	A ser completado
!--	FEATURE_ID = atualizacaodeestadoeacaodecobranca
!--	
[-] [e] wsag:ServiceProperties	

Figura 56: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Service Properties (Sistema COB).

+	e	wsag:ServiceProperties	
		!--	FEATURE_ID = aplicacaodeacaodecobranca
-	e	wsag:GuaranteeTerm	
		a	Obligated
			fornecedorDeServico
	-	e	wsag:ServiceScope
		a	ServiceName
			aplicacao-de-acao-de-cobrancaOP
	-	e	wsag:QualifyingCondition
			!--
			A ser completado
	-	e	wsag:ServiceLevelObjective
			segurancaVAR IS-LESS-INCLUSIVE
			!--
			FEATURE_ID = reference20IXIsemcontrole
			sem-controle
			!--
			FEATURE_ID = reference20IXIautenticacaogeral
			autenticacao-geral
			!--
			FEATURE_ID = reference20IXIautenticacaoporperfil
			autenticacao-por-perfil
			!--
			FEATURE_ID = reference20IXIautenticacaoporusuario
			autenticacao-por-usuario
	-	e	wsag:BusinessValueList
			!--
			A ser completado
			!--
			FEATURE_ID = aplicacaodeacaodecobranca
-	e	wsag:GuaranteeTerm	
		a	Obligated
			fornecedorDeServico
	-	e	wsag:ServiceScope
		a	ServiceName
			aplicacao-de-acao-de-cobrancaOP
	-	e	wsag:QualifyingCondition
			!--
			A ser completado
	-	e	wsag:ServiceLevelObjective
			disponibilidadeVAR IS-LESS-INCLUSIVE
			!--
			FEATURE_ID = reference20IXIsemcontrole0
			sem-controle
			!--
			FEATURE_ID = reference20IXIa24X7
			24X7
			!--
			FEATURE_ID = reference20IXIa24X5
			24X5
			!--
			FEATURE_ID = reference20IXIa8X5
			8X5
			!--
			FEATURE_ID = reference20IXIoutro
			Outro
	-	e	wsag:BusinessValueList
			!--
			A ser completado
			!--
			FEATURE_ID = aplicacaodeacaodecobranca
+	e	wsag:GuaranteeTerm	

Figura 57: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Guarantee Terms (Sistema COB/ Parte 1).

+	e	wsag:GuaranteeTerm	FEATURE_ID = aplicacaodeacaodecobranca
		!--	
-	e	wsag:GuaranteeTerm	
		a	Obligated
			fornecedorDeServico
		-	e
		wsag:ServiceScope	
		a	ServiceName
			aplicacao-de-acao-de-cobrancaOP
		-	e
		wsag:QualifyingCondition	
			!--
			A ser completado
		-	e
		wsag:ServiceLevelObjective	
			tempo-de-esperaVAR IS-LESS-INCLUSIVE
			!--
			FEATURE_ID = reference20IXIsemcontrole 1
			sem-controle
			!--
			FEATURE_ID = reference20IXIa 15
			15
			!--
			FEATURE_ID = reference20IXIa30
			30
			!--
			FEATURE_ID = reference20IXIa60
			60
			!--
			FEATURE_ID = reference20IXIoutro0
			Outro
		-	e
		wsag:BusinessValueList	
			!--
			A ser completado
			!--
			FEATURE_ID = aplicacaodeacaodecobranca
		-	e
		wsag:GuaranteeTerm	
		a	Obligated
			fornecedorDeServico
		-	e
		wsag:ServiceScope	
		a	ServiceName
			aplicacao-de-acao-de-cobrancaOP
		-	e
		wsag:QualifyingCondition	
			!--
			A ser completado
		-	e
		wsag:ServiceLevelObjective	
			tempo-de-respostaVAR IS-LESS-INCLUSIVE
			!--
			FEATURE_ID = reference20IXIsemcontrole2
			sem-controle
			!--
			FEATURE_ID = reference20IXIa5
			5
			!--
			FEATURE_ID = reference20IXIa16
			15
			!--
			FEATURE_ID = reference20IXIa31
			30
			!--
			FEATURE_ID = reference20IXIoutro 1
			Outro
		-	e
		wsag:BusinessValueList	
			!--
			A ser completado
			!--
			FEATURE_ID = aplicacaodeacaodecobranca
		+	e
		wsag:GuaranteeTerm	

Figura 58: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Guarantee Terms (Sistema COB/ Parte 2).

+	e	wsag:GuaranteeTerm	!	--	FEATURE_ID = aplicacaodeacaodecobranca
-	e	wsag:GuaranteeTerm			
		ⓐ			fornecedorDeServico
-	e	wsag:ServiceScope			
		ⓐ			aplicacao-de-acao-de-cobrancaOP
-	e	wsag:QualifyingCondition			
		!		--	A ser completado
-	e	wsag:ServiceLevelObjective			
					acessos-simultaneosVAR IS-LESS-INCLUSIVE
		!		--	FEATURE_ID = reference20IXIsemcontrole3
					sem-controle
		!		--	FEATURE_ID = reference20IXIa 10
					10
		!		--	FEATURE_ID = reference20IXIa 100
					100
		!		--	FEATURE_ID = reference20IXIa500
					500
		!		--	FEATURE_ID = reference20IXIoutro2
					Outro
-	e	wsag:BusinessValueList			
		!		--	A ser completado
!		--			FEATURE_ID = aplicacaodeacaodecobranca
-	e	wsag:GuaranteeTerm			
		ⓐ			fornecedorDeServico
-	e	wsag:ServiceScope			
		ⓐ			aplicacao-de-acao-de-cobrancaOP
-	e	wsag:QualifyingCondition			
		!		--	A ser completado
-	e	wsag:ServiceLevelObjective			
					desempenhoVAR IS-LESS-INCLUSIVE
		!		--	FEATURE_ID = reference20IXIsemcontrole4
					sem-controle
		!		--	FEATURE_ID = reference20IXIa 1
					1
		!		--	FEATURE_ID = reference20IXIa6
					5
		!		--	FEATURE_ID = reference20IXIa 11
					10
		!		--	FEATURE_ID = reference20IXIoutro3
					Outro
-	e	wsag:BusinessValueList			
		!		--	A ser completado
!		--			FEATURE_ID = atualizacaoodeestadodeacaodecobranca
+	e	wsag:GuaranteeTerm			

Figura 59: Molde de Contrato Eletrônico – Seção de termos WS-Agreement, Guarantee Terms (Sistema COB/ Parte 3).

B.3 Seção de Processo de Negócio WS-BPEL

Nas figuras a seguir (Figura 60 e Figura 61) são apresentadas partes da seção de processo de negócio WS-BPEL do molde de contrato eletrônico. Todas as possíveis formas de comunicação entre as duas organizações, visumbradas durante a realização do estudo de caso, estão cobertas nesse processo. As comunicações são realizadas por meio do uso dos serviços Web disponibilizados por cada um deles e apresentados nos modelos de características. Dependendo do tipo de interação necessária, um conjunto menor ou maior de serviços Web é encadeado.

A Figura 60 apresenta o processo de negócio geral, em que duas partes dele estão minimizadas. As partes minimizadas são apresentadas de forma expandida na Figura 61. As anotações ligando os elementos com as características apresentadas no apêndice anterior não são também apresentadas aqui. O Código 14 apresenta os trechos dessa seção referente à especificação dos elementos Partner Links, que não podem ser vistos pela visão gráfica.

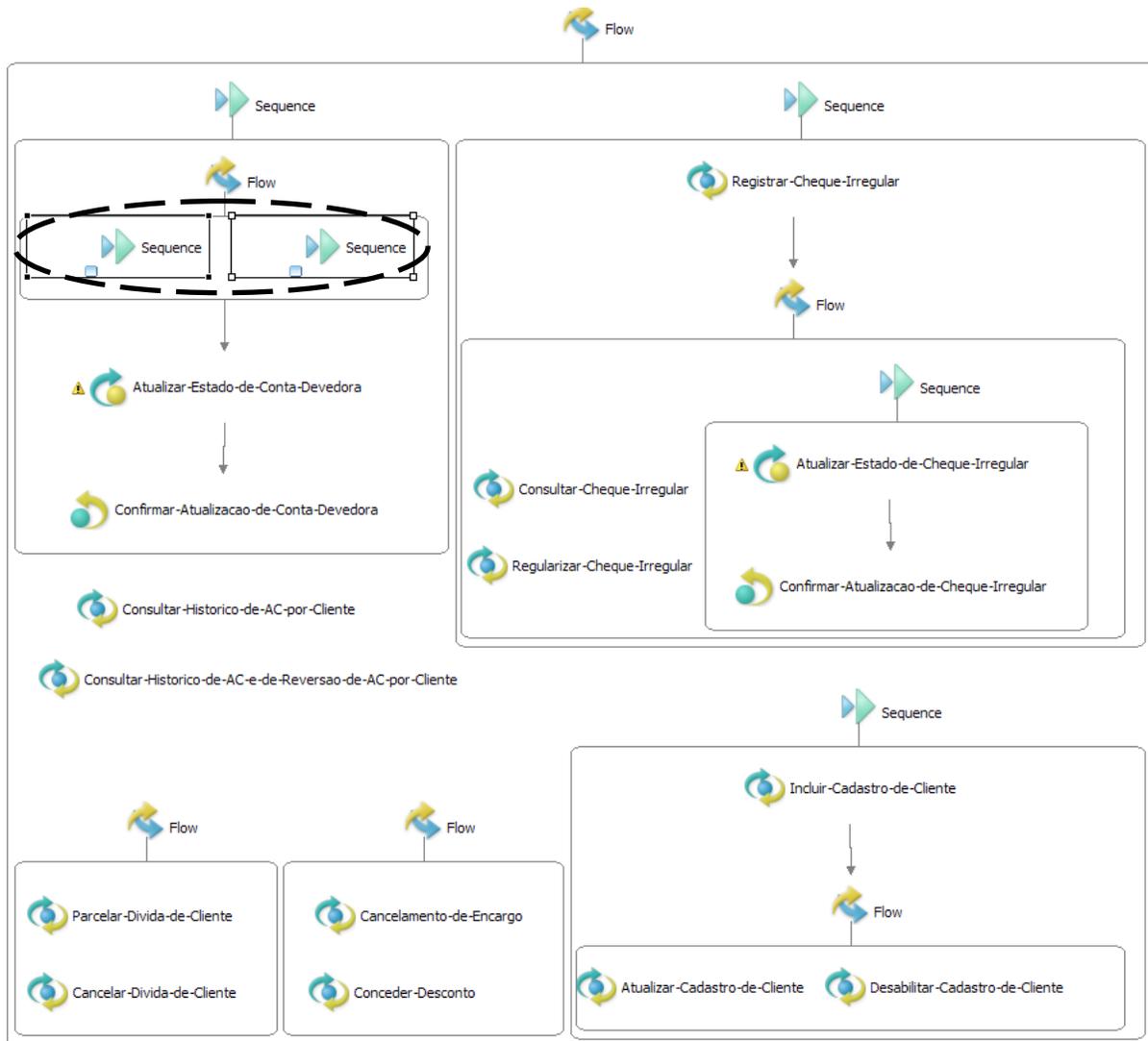


Figura 60: Molde de Contrato Eletrônico – Seção de processo de negócio WS-BPEL (visão geral do processo).

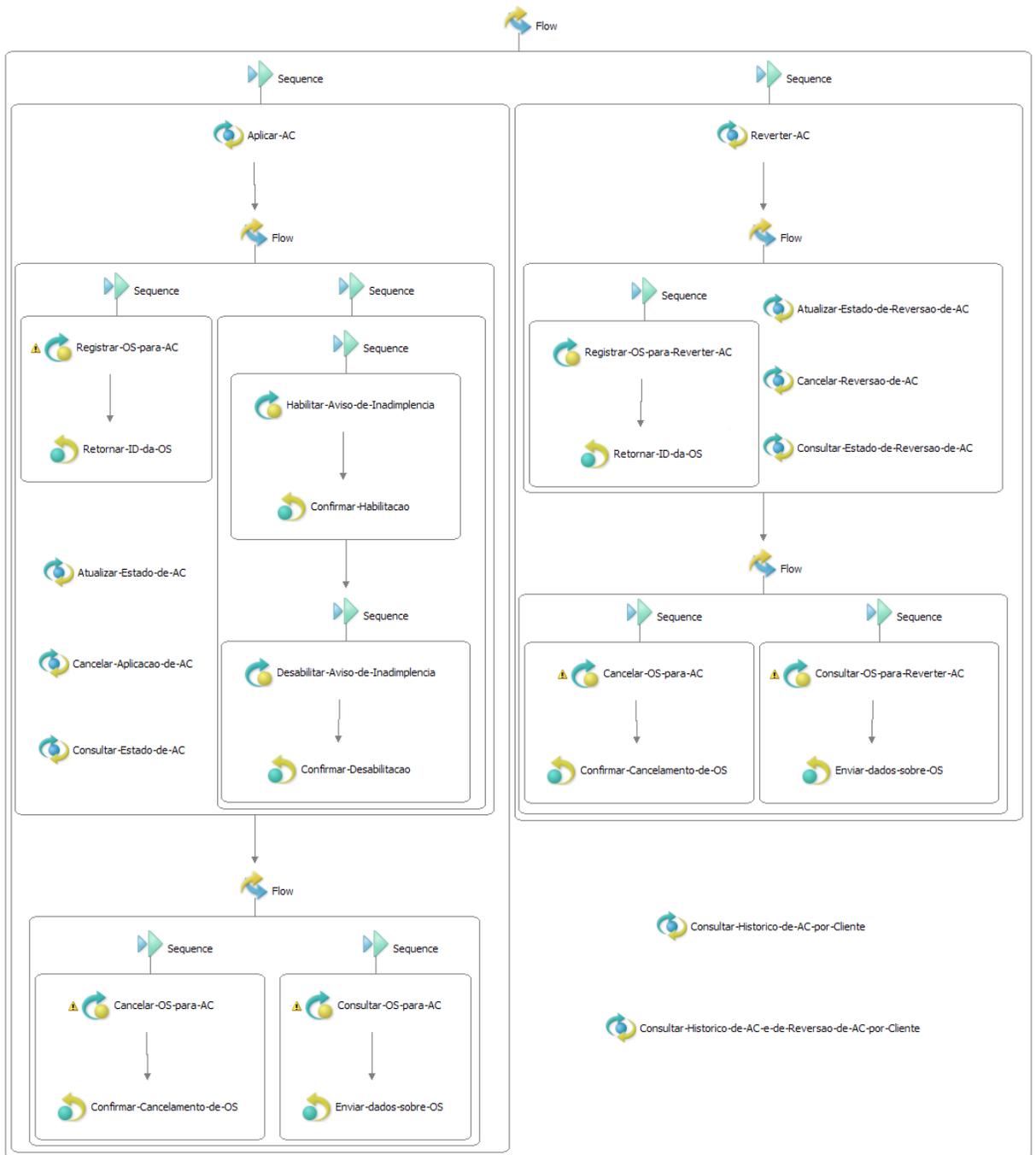


Figura 61: Molde de Contrato Eletrônico – Seção de processo de negócio WS-BPEL (visão expandida de Aplicar-AC e Reverter-AC).

Código 14: Molde de Contrato Eletrônico – Seção WS-BPEL, Partner Links.

```
<bpel:partnerLinks>
  <!-- FEATURE_ID = acoesdecobranca -->
  <bpel:partnerLink name="acoes-de-cobranca" partnerLinkType="ns1:acoes-de-
cobrancaPLT" partnerRole="acoes-de-cobrancaROLE"/>
  <!-- FEATURE_ID = reversoesdeacaodecobranca -->
  <bpel:partnerLink name="reversoes-de-acao-de-cobranca"
partnerLinkType="ns1:reversoes-de-acao-de-cobrancaPLT"
partnerRole="reversoes-de-acao-de-cobrancaROLE"/>
  <!-- FEATURE_ID = chequesirregulares -->
  <bpel:partnerLink name="cheques-irregulares"
partnerLinkType="ns1:cheques-irregularesPLT" partnerRole="cheques-
irregularesROLE"/>
  <!-- FEATURE_ID = dividas -->
  <bpel:partnerLink name="dividas" partnerLinkType="ns1:dividasPLT"
partnerRole="dividasROLE"/>
  <!-- FEATURE_ID = encargosedescontos -->
  <bpel:partnerLink name="encargos-e-descontos"
partnerLinkType="ns1:encargos-e-descontosPLT" partnerRole="encargos-e-
descontosROLE"/>
  <!-- FEATURE_ID = cadastrosdeclientes -->
  <bpel:partnerLink name="cadastros-de-clientes"
partnerLinkType="ns1:cadastros-de-clientesPLT" partnerRole="cadastros-de-
clientesROLE"/>
  <!-- FEATURE_ID = OSparaexecutaracaodecobranca -->
  <bpel:partnerLink myRole="OS-para-executar-acao-de-cobrancaROLE"
name="OS-para-executar-acao-de-cobranca" partnerLinkType="ns1:OS-para-
executar-acao-de-cobrancaPLT"/>
  <!-- FEATURE_ID = OSparaexecutarrerversaodeacaodecobranca -->
  <bpel:partnerLink myRole="OS-para-executar-rerversao-de-acao-de-
cobrancaROLE" name="OS-para-executar-rerversao-de-acao-de-cobranca"
partnerLinkType="ns1:OS-para-executar-rerversao-de-acao-de-de-cobrancaPLT"/>
  <!-- FEATURE_ID = atualizacaodeinformacoesdecliente -->
  <bpel:partnerLink myRole="atualizacao-de-informacoes-de-clienteROLE"
name="atualizacao-de-informacoes-de-cliente"
partnerLinkType="ns1:atualizacao-de-informacoes-de-clientePLT"/>
  <!-- FEATURE_ID = avisosdeinadimplencia -->
  <bpel:partnerLink myRole="avisos-de-inadimplenciaROLE" name="avisos-de-
inadimplencia" partnerLinkType="ns1:avisos-de-inadimplenciaPLT"/>
  <!-- FEATURE_ID = consultadeestadodeacoesereversoesdeacaodecobranca -->
  <bpel:partnerLink name="consulta-de-estado-de-acoes-e-reversoes-de-acao-
de-cobranca" partnerLinkType="ns1:consulta-de-estado-de-acoes-e-reversoes-
de-acao-de-cobrancaPLT" partnerRole="consulta-de-estado-de-acoes-e-
reversoes-de-acao-de-cobrancaROLE"/>
  <!-- FEATURE_ID = consultadehistoricodeacoesereversoesdeacaodecobranca -->
  <bpel:partnerLink name="consulta-de-historico-de-acoes-e-reversoes-de-
acao-de-cobranca" partnerLinkType="ns1:consulta-de-historico-de-acoes-e-
reversoes-de-acao-de-cobrancaPLT" partnerRole="consulta-de-historico-de-
acoes-e-reversoes-de-acao-de-cobrancaROLE"/>
</bpel:partnerLinks>
```

Apêndice C

Artigos Publicados e em Avaliação

Este apêndice apresenta os artigos relacionados à proposta apresentada nesta tese que foram publicados ou que estão em processo de avaliação para posterior publicação.

C.1 Artigos Publicados

Título	A Feature-based Approach to Electronic Contracts
Autores	Marcelo Fantinato, Maria Beatriz F. de Toledo, Itana Maria de S. Gimenes
Evento	IEEE CEC/EEE (8th IEEE Int. Conference on E-Commerce Technology / 3rd IEEE Int. Conference on Enterprise Computing, E-Commerce and E-Services)
Local	Palo Alto, California, EUA
Data	junho de 2006
Publicação	IEEE Computer Society

Título	Web Service E-Contract Establishment Using Features
Autores	Marcelo Fantinato, Itana Maria de S. Gimenes, Maria Beatriz F. de Toledo
Evento	BPM (4th International Conference on Business Process Management)
Local	Vienna, Austria
Data	Setembro de 2006
Publicação	Springer – Lecture Notes in Computer Science

Título	Supporting QoS Negotiation with Feature Modeling
Autores	Marcelo Fantinato, Itana Maria de S. Gimenes, Maria Beatriz F. de Toledo
Evento	ICSOC (5th International Conference on Service-Oriented Computing)
Local	Vienna, Austria
Data	Setembro de 2007
Publicação	Springer – Lecture Notes in Computer Science

Título	Estabelecimento de Contratos Eletrônicos para Serviços Web com a Ferramenta FeatureContract
Autores	Marcelo Fantinato, Maria Beatriz F. de Toledo, Itana Maria de S. Gimenes, Anderson Rodrigues de Oliveira, Eduardo Pezutti B. dos Santos
Evento	WBPM (I Brazilian Workshop on Business Process Management)
Local	Gramado – RS – Brasil
Data	Outubro de 2007

C.2 Artigo em Avaliação

Título	WS-Contract Establishment with QoS: an Approach Based on Feature Modeling
Autores	Marcelo Fantinato, Maria Beatriz F. de Toledo, Itana Maria de S. Gimenes
Revista	IJCIS (International Journal of Cooperative Information Systems)
Publicação	World Scientific Publishing Company
Situação	Revisado – submetido para segunda avaliação

Referências Bibliográficas

- [1] U. Dayal, M. Hsu and R. Ladin, Business Process Coordination: State of the Art, Trends, and Open Issues, in *Proceedings of VLBD*, Roma, Morgan-Kaufmann, 2001, pp. 3-13.
- [2] F. Leymann, D. Roller and M.-T. Schmidt, Web Services and Business Process Management, *IBM Systems Journal* 41(2), 2002, pp. 198-211.
- [3] P. W. P. J. Grefen et. al., CrossFlow: Cross-Organizational Workflow Management for Service Outsourcing in Dynamic Virtual Enterprises, *IEEE Data Engineering Bulletin*, 24(1), IEEE Computer Society, 2001, pp. 52-57.
- [4] M. Papazoglou and D. Georgakopoulos, Service-Oriented Computing, *Communications of the ACM* 46(10), 2003, pp. 24-28.
- [5] Web Services, <http://www.w3.org/2002/ws/>, 2007.
- [6] G. Alonso, F. Casati, H. Kuno and V. Machiraju, *Web services: Concepts, Architectures, and Applications*, Springer Verlag, Berlin, 2003.
- [7] Web Services Description Language (WSDL), <http://www.w3.org/TR/wsdl>, 2007.
- [8] UDDI.org, <http://www.uddi.org>, 2007.
- [9] SOAP Specifications, <http://www.w3.org/TR/SOAP>, 2007.

- [10] Extensible Markup Language (XML), <http://www.w3.org/XML/>, 2007.
- [11] OASIS Web Services Business Process Execution Language (WS-BPEL) TC, http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel, 2007.
- [12] A. Andrieux et. al., Web Services Agreement Specification (WS-Agreement), http://www.ogf.org/Public_Comment_Docs/Documents/Oct-2006/WS-AgreementSpecificationDraftFinal_sp_tn_jpver_v2.pdf, 2007.
- [13] N. Oldham, K. Verma, A. Sheth and F. Hakimpour, Semantic WS-Agreement Partner Selection, in *Proceedings of WWW*, Edinburgh, ACM Press, 2006, pp. 697-706.
- [14] S. Bajaj et. al., Web Services Policy Framework (WS-Policy), <http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-polfram/ws-policy-2006-03-01.pdf>, 2007.
- [15] L. Baresi, S. Guinea and P. Plebani, WS-Policy for Service Monitoring, in *Proceedings of VLDB-TESS*, Trondheim, Springer, 2005, pp. 72-83.
- [16] O. Marjanovic and Z. Milosevic, Towards Formal Modeling of e-Contracts, in *Proceedings of EDOC*, Seattle, IEEE Computer Society, 2001, pp. 59-68.
- [17] Y. Hoffner, S. Field, P. Grefen and H. Ludwig, Contract-Driven Creation and Operation of Virtual Enterprises, *Computer Networks* 37, 2001, pp. 111-136.
- [18] A. Sahai, V. Machiraju, M. Sayal, A. van Moorsel and F. Casati, Automated SLA Monitoring for Web Services, in *Proceedings of DSON*, Montreal, Springer, 2002, pp. 28-41.
- [19] D. A. Menasce, QoS Issues in Web Services, *IEEE Internet Computing* 6(6), 2002, pp. 72-75.

- [20] D. K. W. Chiu, S-C Cheung and S. Till, A Three Layer Architecture for E-Contract Enforcement in an E-Service Environment, in *Proceedings of HICSS*, Big Island, IEEE Computer Society, 2003, pp. 74.
- [21] M. Rouached, O. Perring and C. Godart, A Contract Layered Architecture for Regulating Cross-Organisational Business Processes, in *Proceedings of BPM*, Nancy, 2005, pp. 410-415.
- [22] P. R. Krishna, K. Karlapalem and A. R. Dani, From Contract to E-Contracts: Modeling and Enactment, *Information Technology and Management* 6(4), 2005, pp. 363-387.
- [23] P. F. Linington, Automating Support for E-Business Contracts, *IJCIS journal* 14(2/3), 2005, pp. 77-98.
- [24] O. von Susani and P. Dugerdil, Contract-Based Cross-Organizational Automated Processes, in *Proceedings of IEEE CEC*, Munich, IEEE Computer Society, 2005, pp. 540-543.
- [25] A. Berry and Z. Milosevic, Extending Choreography with Business Contract Constraints, *IJCIS journal* 14(2/3), 2005, pp. 131-179.
- [26] A. D. H. Farrell, M. Sergot, M. Salle and C. Bartolini, Using the Event Calculus for Tracking the Normative State of Contracts, *IJCIS journal* 14(2/3), 2005, pp. 99-129.
- [27] Y. Hoffner and S. Field, Transforming Agreements into Contracts, *IJCIS journal* 14(2/3), 2005, pp. 217-244.
- [28] S. Angelov, P. Grefen; The 4W framework for B2B e-contracting, *IJNVO journal*, 2(1), 2003, pp.78-97.

- [29] V. Kabilan e P. Johannesson, Semantic Representation of Contract Knowledge using Multi Tier Ontology, in *Proceedings of SWDB*, Berlin, 2003, pp. 395-414.
- [30] A. Keller and H. Ludwig, The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services, *JNSM journal* 11(1), 2003, pp. 57-81.
- [31] SEI – Software Engineering Institute, A Framework for Software Product Line Practice - Version 4.2, <http://www.sei.cmu.edu/productlines/framework.html>, 2007.
- [32] K. Kang, S. Cohen, J. Hess, W. Novak and A. Peterson, *Feature-Oriented Domain Analysis (FODA) Feasibility Study*, Technical Report CMU/SEI-90-TR-021, SEI/CMU, 1990.
- [33] K. Czarnecki, S. Helsen and U. Eisenecker, Staged Configuration through Specialization and Multi-Level Configuration of Feature Models, *Software Process: Improvement and Practice* 10(2), 2005, pp. 143-169.
- [34] I. M. S. Gimenes and G. H. Travassos, O Enfoque de linha de produto para Desenvolvimento de Software, *Anais da XXI Jornada de Atualização em Informática (JAI)*, Porto Alegre, SBC, 2002, pp. 01-31.
- [35] P. Clements and L. Northrop, *Software Product Lines: Practices and Patterns*, SEI Series in Software Engineering, Addison-Wesley, 2001.
- [36] L. Bass, P. Clements, R. Kazman, *Software Architecture in Practice*, Addison Wesley Longman, 1998.
- [37] K. C. Kang, S. Kim, J. Lee, K. Kim, E. Shin and M. Huh, FORM: A Feature-Oriented Reuse Method with Domain-Specific Reference Architectures, *Annals of Software Engineering* 5, 1998, pp. 143-168.

- [38] V. Cechticky, A. Pasetti, O. Rohlik and W. Schaufelberger, XML-Based Feature Modelling, in *Proceedings of ICSR*, Madrid, Springer, 2004, pp. 101-114.
- [39] P. Sochos, I. Philippow and M. Riebisch, Feature-Oriented Development of Software Product Lines: Mapping Feature Models to the Architecture, in *Proceedings of Net.ObjectDays*, Erfurt, Springer, 2004, pp. 138-152.
- [40] H. Gomaa, *Designing software product lines with UML: from use cases to pattern-based software architectures*, Addison-Wesley, 2005.
- [41] M. Griss, Implementing Product-Line Features with Component Reuse, in *Proceedings of ICSR*, Vienna, Springer, 2000, pp. 137-152.
- [42] K. Czarnecki, C. H. P. Kim and K. T. Kalleberg, Feature Models are Views on Ontologies, in *Proceedings of SPLC*, Baltimore, IEEE Computer Society, 2006, pp. 41-51.
- [43] T. R. Gruber, A Translation Approach to Portable Ontologies, *Knowledge Acquisition* 5(2), 1993, pp. 199-220.
- [44] OWL Web Ontology Language Overview, <http://www.w3.org/TR/owl-features/>, 2007.
- [45] S. Angelov, P. Grefen; A Conceptual Framework for B2B Electronic Contracting, in *Proceedings of IFIP PROVE*, Portugal, 2002, pp. 143-150.
- [46] S. Angelov, P. Grefen; The Business Case for B2B E-Contracting, in *Proceedings of ICEC*, The Netherlands, 2004, pp. 31-40.
- [47] G. Michael, S.-S. Katarina, G. Markus. Legal Aspects of Electronic Contracts, In *Proceedings of IDSO*, CEUR-WS.org, Stockholm, 2000.

- [48] F. Leymann, D. Roller. Business Processes in a Web Services World. <http://www.ibm.com/developerworks/library/ws-bpelwp/>, 2007.
- [49] A. Goodchild, C. Herring, Z. Milosevic. Business Contracts for B2B, in *Proceedings of CAISE*, Stockholm, CEUR-WS.org, 2000.
- [50] P. R. Krishna, K. Karlapalem, and D.K.W. Chiu. An ER^{EC} Framework for e-Contract Modeling, Enactment and Monitoring, *Data and Knowledge Engineering* 51(1), 2004, pp. 31-58.
- [51] S. Angelov, P. Grefen. Support for B2B e-Contracting - The Process Perspective, in *Proceedings of BASYS*, Cancun, 2002, pp. 87-96.
- [52] A. Dan et. al., Web Services on demand: WSLA-driven Automated Management, *IBM Systems Journal*, 43(1), 2004, pp. 136-158.
- [53] G. Governatori et. al., A formal approach to protocols and strategies for (legal) negotiation, in *11th International Conference on Artificial Intelligence and Law*, 2001, pp. 168-177.
- [54] Object Management Group – Unified Modeling Language, <http://www.uml.org/>, 2007.
- [55] F. Casati, A Conversation on Web Services: What's New, What's True, What's Hot, and What's Not. (Invited paper), in *Proceedings of ECAI*, França, 2002.
- [56] F. Laymann, Web Services Flow Language (WFSL 1.0), <http://www-3.ibm.com/software/solutions/webservices/pdf/WSFL.pdf>, 2001.
- [57] Web Services Conversation Language (WSCL), W3C Note, 2002.

- [58] Web Services Choreography Interface (WSCCI), <http://www.w3.org/TR/wsci>, 2007.
- [59] Business Process Modeling Language (BPML), <http://www.bpmi.org>, 2007.
- [60] Business Process Specification Schema (BPSS). http://www.ebxml.org/specdrafts/cc_and_bp_document_overview_ver_1.01.pdf, 2006,
- [61] S. Thatte, XLANG - Web Services for Business Process Design, http://www.gotdotnet.com/team/xml_wsspecs/xlang-c/default.htm, 2001.
- [62] M. Bernauer, G. Kappel, G. Kramler and W. Retschitzegger, Specification of Interorganizational Workflows - A Comparison of Approaches, in *Proceedings of SCI*, Orlando, 2003, pp. 30-36.
- [63] S. C. Cheung, D. K. W. Chiu, and S. Till, A three-layer framework for cross-organizational e-contract enactment, in *Proceedings of WES*, Lecture Notes in Computer Science, 2512, Springer, 2002, pp. 78-92.
- [64] Workflow Management Coalition (WfMC), <http://www.wfmc.org>, 2007.
- [65] DCOM Technical Overview (DCOM), <http://msdn2.microsoft.com/en-us/library/ms809340.aspx>, 2007.
- [66] OMG's CORBA Website – Common Object Request Broker Architecture, <http://www.corba.org>, 2007.
- [67] M. Antkiewicz and K. Czarnecki, FeaturePlugin: Feature Modeling Plug-in for Eclipse, in *Proceedings of eTX workshop*, Vancouver, ACM Press, 2004, pp. 67-72.

- [68] H. Wang, L. Y. Fang, J. Sun, H. Zhang and J. Z. Pan, A Semantic Web Approach to Feature Modeling and Verification, in *Proceedings of SWESE*, Galway, Springer-Verlag, 2005.
- [69] IBM – Requirements Management – Rational RequisitePro – Software, <http://www-306.ibm.com/software/awdtools/reqpro/>, 2007.
- [70] Telelogic DOORS: A market-leading solution for Requirements Management, <http://www.telelogic.com/products/doors/index.cfm>, 2007.
- [71] W3C XML Path Language (XPath), <http://www.w3.org/TR/xpath/>, 2007.
- [72] Feature Modeling Plug-in, <http://sourceforge.net/projects/fmp/>, 2007.
- [73] K. Czarnecki and M. Antkiewicz, Mapping Features to Models: A Template Approach Based on Superimposed Variants, in *Proceedings of GPCE*, Tallinn, Springer, 2005, pp. 422-437.
- [74] Oracle BPEL Process Manager, <http://www.oracle.com/technology/products/ias/bpel/index.html>, 2007.
- [75] BPEL Project home, <http://www.eclipse.org/bpel/>, 2007.
- [76] Active Endpoints – ActiveBPEL Designer for SOA Orchestration, <http://www.active-endpoints.com/>, 2007.
- [77] XSL Transformations (XSLT), <http://www.w3.org/TR/xslt>, 2007.
- [78] W. Streitberger, Framework for the Negotiation of Electronic Contracts in E-Business on Demand, in *Proceedings of IEEE CEC*, Munich, IEEE Computer Society, 2005, pp. 370-373.

- [79] S. Rinderle and M. Benyoucef, Towards the Automation of E-Negotiation Processes Based on Web Services – A Modeling Approach, in *Proceedings of WISE*, New York, Springer, 2005, pp. 443-453.
- [80] A. Jertila and M. Schoop, Electronic Contracts in Negotiation Support Systems: Challenges, Design and Implementation, in *Proceedings of IEEE CEC*, Munich, IEEE Computer Society, 2005, pp. 396-399.
- [81] H. Kaminski and M. Perry, SLA Automated Negotiation Manager for Computing Services, in *Proceedings of IEEE CEC EEE*, San Francisco, IEEE Computer Society, 2006, pp. 47-54.
- [82] F. Griffel, et. al., Electronic Contracting with COSMOS – How to Establish, Negotiate and Execute Electronic Contracts on the Internet, in *Proceedings of EDOC*, IEEE Computer Society, 1998, pp. 46-55.
- [83] M. Koetsier, P. Grefen, J. Vonk, Contracts for Cross-Organizational Workflow Management, in *Proceedings of EC-WEB*, Londres-Greenwich, 2000, pp. 110-121.
- [84] D. K. W. Chiu, S. C. Cheung, e P. C. K. Hung, A meta-model for contract template driven e-negotiation processes, in *Proceedings of PACIS*, Tokyo, 2002, pp. 854-868.
- [85] S. C. Cheung, P. C. K. Hung, e D. K. W. Chiu, A meta-model for e-contract template variable dependencies facilitating e-negotiation, in *Proceedings of ER*, IEEE Computer Society Press, Tampere, 2002, pp. 50-64.
- [86] A. Wombacher, B. Mahleko e E. Neuhold, IPSI-PF: A Business Process Matchmaking Engine based on annotated Finite State Automata, *ISEBM Journal*, 3(2), 2005, pp. 127-150.

- [87] M. Dumas, B. Benatallah, N. Russell and M. Spork, A configurable matchmaking framework for electronic marketplace, *ECRA journal*, 3(1), 2004, pp. 95-106.
- [88] V. Kabilan, P. Johannesson e D. M. Rugaimukamu, Business Contract Obligation Monitoring through Use of Multi Tier Contract Ontology, in *Proceedings of OTM*, Springer-Verlag, 2003, pp. 690-702.
- [89] V. Kabilan, P. Johannesson e D. M. Rugaimukamu, An Ontological Approach to Unified Contract Management, in *Proceedings of EJC*, Japan, 2003, pp. 106-110.
- [90] alphaWorks : Web Services Toolkit : Overview, <http://www.alphaworks.ibm.com/tech/webservicestoolkit>, 2007.
- [91] A. Sahai, et. al., Automated SLA Monitoring for Web Services, in *Proceedings of DSON*, Springer-Verlag, 2002, pp. 28-41.
- [92] M. Castellanos, F. Casati, U. Dayal, M.-C. Shan, Intelligent management of SLA for composite Web services, in *Proceedings of DNIS*, Japan, 2003, pp. 158-171.
- [93] V. Tasic, On Comprehensive Contractual Descriptions of Web Services, in *Proceedings of IEEE EEE*, Hong Kong, IEEE Computer Society, 2005, pp. 444-449.
- [94] Eclipse.org home, <http://www.eclipse.org/>.
- [95] Web Tools Platform (WTP), <http://www.eclipse.org/webtools/main.php>, 2007.
- [96] K. Terplan, *OSS Essentials: Support System Solutions for Service Providers*, John Wiley & Sons, New York, 2001.
- [97] J. Hunger and M. Thiebaud, *Telecommunications Billing Systems – Implementing and Upgrading for Profitability*, McGraw-Hill, New York, 2003.