

Martim Filipe de Cazula e Constantino

Proposta de um modelo de rastreamento entre requisitos de software  
e o contexto organizacional

Dissertação apresentada ao  
Instituto de Pesquisas Tecnológicas do Estado  
de São Paulo - IPT, para obtenção do título de  
Mestre em Engenharia de Computação.  
Área de Concentração: Engenharia de Software

Orientador: Prof. Dr. Paulo Sérgio Muniz Silva

São Paulo

Junho/2008

## **Agradecimentos**

Ao meu orientador professor Dr. Paulo Sérgio Muniz, pela inestimável ajuda ao longo de todo o trabalho e pela atenção e dedicação sem as quais este trabalho não seria viável.

À minha namorada Denise, que me acompanhou durante todo o período de elaboração deste trabalho.

Aos amigos de cujo convívio me ausentei em diversos momentos, convívio esse que espero retomar com urgência.

## Resumo

Muitos projetos não atingem o esperado grau de sucesso por falhas no entendimento entre quem tem um problema e quem pretende resolver esse problema. A engenharia de requisitos é apontada como a disciplina capaz de solucionar esse problema de entendimento, tendo como importante atividade o rastreamento de requisitos. Entretanto, a descrição de uma solução proposta na forma de requisitos de *software* não contempla o contexto organizacional, que representa o ambiente no qual se tem o problema. Sem esta conexão com o contexto organizacional, não é possível resgatar a real origem dos requisitos, o que torna falho o rastreamento. Este trabalho propõe a criação de um modelo de rastreamento de requisitos que relaciona os requisitos de *software* com o contexto organizacional em que foram criados. O modelo é baseado em um meta-modelo que relaciona os requisitos ao contexto organizacional e à base racional que lhes serve de justificativa. O quadro de referência de Zachman é utilizado como guia para a construção do modelo do contexto organizacional, e o BPDM (*Business Process Definition Metamodel*), o IDEF3 (*Integrated DEFINition Method 3*), o BMM (*Business Motivation Model*) e o OSM (*Organization Structure Metamodel*) são utilizados como referências para a criação do modelo do contexto organizacional. O modelo de base racional é adaptado de diferentes propostas de base racional e o modelo de requisitos inclui os conceitos de necessidade dos envolvidos, características de *software* e requisitos de *software*. Por fim, é apresentado um exemplo de utilização do modelo proposto.

Palavras-chave: Rastreamento de Requisitos, Contexto Organizacional, Base Racional, Requisitos de *Software*, *Business Process Definition Metamodel* (BPDM), *Integrated DEFINition Method 3* (IDEF3), *Business Motivation Model* (BMM), *Organization Structure Metamodel* (OSM), Quadro de Referência de Zachman

## **Abstract**

### **Proposal of a model of traceability between software requirements and the organizational context**

Many projects don't meet the expected degree of success due to failures in the understanding between those who have a problem and those who intend to solve the problem. Requirements engineering is considered the best suited discipline to deal with such problems, having the requirements traceability as an important activity. However, the description of a proposed solution in the form of software requirements doesn't consider the organizational context, which represents the environment in which there is a problem. Without the connection with the organizational context, it is not possible to assess the real origins of the requirements, thus making the traceability weaker. The current work proposes to create a requirements traceability model that relates the software requirements to the organization context that gave birth to those requirements. The model is based on a proposed meta-model that relates the requirements to the organizational context and to the design rationale that acts as a justification for the requirements. The Zachman Framework is used as a guide to the specification of an organizational context model, and BPDM (Business Process Definition Metamodel), IDEF3 (Integrated DEFinition Method 3), BMM (Business Motivation Model) and OSM (Organization Structure Metamodel) are used as references for the creation of the model. The design rationale model is adapted from different proposals of design rationale and the requirements model include the concepts of stakeholder needs, features and software requirements. Finally, an example of the proposed model's utilization is presented.

Keywords: Requirements Traceability, Organizational Context, Design Rationale, Software Requirements, Business Process Definition Metamodel (BPDM), Integrated DEFinition Method 3 (IDEF3), Business Motivation Model (BMM), Organization Structure Metamodel (OSM), Zachman Framework

## Índice de Figuras

Figura 1 - Pirâmide de requisitos de <i>software</i> (Leffingwell, 2003) .....	20
Figura 2 - Meta-modelo de rastreamento de requisitos (Ramesh, Jarke, 2001).....	22
Figura 3 - Modelo de baixo nível de rastreamento (Ramesh, Jarke, 2001) .....	23
Figura 4 - Modelo de gerenciamento de requisitos (Ramesh, Jarke, 2001).....	24
Figura 5 - Modelo de base racional (Ramesh, Jarke, 2001) .....	25
Figura 6 - Modelo de sistema (Ramesh, Jarke, 2001).....	26
Figura 7 - Modelo de teste (Ramesh, Jarke, 2001) .....	27
Figura 8 - Estratégia sem utilização de modelo de caso de uso (Spence, Probasco, 2000).....	28
Figura 9 - Estratégia utilizando apenas modelo de caso de uso (Spence, Probasco, 2000).....	29
Figura 10 - Estratégia com caso de uso definindo as características do sistema (Spence, Probasco, 2000).....	29
Figura 11 - Estratégia com as características guiando o modelo de caso de uso (Spence, Probasco, 2000).....	30
Figura 12 - Estratégia com casos de uso interpretando a ERS (Spence, Probasco, 2000).....	31
Figura 13 - Meta-modelo de rastreamento de requisitos (Letelier, 2002) .....	32
Figura 14 - Adaptação através de extensão do meta-modelo da UML.....	35
Figura 15 - Representação da adaptação através de estereótipo .....	36
Figura 16 - Relacionamento de rastreamento segundo a UML (OMG, 2007).....	36
Figura 17 - Definição dos relacionamento de rastreamento .....	38
Figura 18 - Proposta de meta-modelo de rastreamento de requisitos pré-ER.....	41
Figura 19 - Exemplo de instância válida do meta-modelo .....	43
Figura 20 - Exemplo de arquitetura corporativa (Morgan, 2002) .....	58
Figura 21 - Exemplo de meta-modelo para três aspectos do QRAC de Zachman (Sowa, Zachman, 1992).....	59
Figura 22 - Relacionamento entre os aspectos de estruturação e motivação (Chapin et al, 2005).....	60
Figura 23 - Relacionamento entre os aspectos dos processos e de motivação (Chapin et al, 2005) .....	61
Figura 24 - Núcleo do modelo de composição no BPDM (Rivet et al, 2007).....	63
Figura 25 - Modelo de condições no BPDM (Rivet et al, 2007) .....	65
Figura 26 - Modelo de cursos do BPDM (Rivet et al, 2007).....	66
Figura 27 - Conceitos básicos do modelo de acontecimento e mudança do BPDM (Rivet et al, 2007).....	67
Figura 28 - Modelo de acontecimento comportamentais do BDPM (Rivet et al, 2007) .....	68
Figura 29 - Condição de mudança no BPDM (Rivet et al, 2007).....	69
Figura 30 - Modelo de comportamento de processo do BPDM (Rivet et al, 2007).....	70
Figura 31 - Modelo de interação simples do BPDM (Rivet et al, 2007).....	71
Figura 32 - Modelo de atividades do BPDM (Rivet et al, 2007).....	72
Figura 33 - Modelo de Protocolo de Interação do BPDM (Rivet et al, 2007) .....	74
Figura 34 - Exemplo de esquema centrado em processo (Mayer et al, 1995) .....	75
Figura 35 - Exemplo de esquema centrado em objetos (Mayer et al, 1995).....	76
Figura 36 – Meta-modelo de fins no BMM (Chapin et al, 2005) .....	83
Figura 37 – Meta-modelo de meios no BMM (Chapin et al, 2005).....	84
Figura 38 – Meta-modelo de Influenciadores no BMM (Chapin et al, 2005) .....	86
Figura 39 - Relacionamentos entre fins, meios e avaliações (Chapin et al, 2005).....	87

Figura 40 - Modelo núcleo do OSM (Cummins et al, 2006).....	88
Figura 41 - Modelo de entidade legal do OSM (Cummins et al, 2006).....	90
Figura 42 - Modelo de posição do OSM (Cummins et al, 2006).....	91
Figura 43 - Modelo de unidade organizacional do OSM (Cummins et al, 2006).....	92
Figura 44 - Modelo do aspecto dos processos.....	93
Figura 45 - Modelo do aspecto de motivação organizacional.....	94
Figura 46 - Modelo do aspecto de estruturação organizacional.....	95
Figura 47 - Modelo do contexto organizacional.....	97
Figura 48 - Meta-modelo IBIS (Conklin, Begeman, 1988).....	99
Figura 49 - Modelo de Potts e Bruns (Lee, 1991).....	100
Figura 50 - Modelo do Laço de Racionalização (Louridas, Loucopoulos, 2000).....	101
Figura 51 - Modelo da base racional.....	104
Figura 52 - Modelo de requisitos.....	106
Figura 53 - Rastreamento entre contexto com problema e contexto desejado.....	107
Figura 54 - Rastreamento do tipo referência entre ContextoOrganizacional e BaseRacional.....	107
Figura 55 - Rastreamento do tipo referência entre ContextoOrganizacional e Requisito.....	107
Figura 56 - Rastreamento do tipo motivação entre ContextoOrganizacional e Requisito.....	108
Figura 57 - Rastreamento do tipo objetivo entre ContextoOrganizacional e Requisito.....	108
Figura 58 - Relacionamento do tipo justificativa entre BaseRacional e Requisito.....	108
Figura 59 - Rastreamento do tipo motivação entre Requisito e BaseRacional.....	109
Figura 60 - Modelo de estrutura organizacional da ABC.....	113
Figura 61 - Modelo de motivação organizacional da ABC.....	114
Figura 62 - Modelo de processos da ABC.....	115
Figura 63 - Modelo de sub-atividades de AT2.....	115
Figura 64 - Modelo de sub-atividades de AT3.....	116
Figura 65 - Relacionamentos entre os aspectos do contexto organizacional.....	117
Figura 66 - Modelo de motivação para alteração do contexto organizacional.....	118
Figura 67 - Modelo desejado de processos de negócio.....	119
Figura 68 - Modelo de necessidades do projeto PR.....	120
Figura 69 - Modelo de características do projeto PR.....	121
Figura 70 - Modelo de base racional que justifica uma característica de <i>software</i> .....	122
Figura 71 - Modelo de requisitos de <i>software</i> do projeto PR.....	124
Figura 72 - Modelo de base racional que justifica um requisito de <i>software</i> .....	124
Figura 73 - Exemplo de necessidade como diferença do contexto organizacional.....	125
Figura 74 - Exemplo de base racional como rastreamento entre dois requisitos.....	125

## Índice de Tabelas

Tabela 1 - Comparação de QRACs em termos de aspectos corporativos (Urbaczewski, Mrdalj, 2006).....	49
Tabela 2 - Comparação dos QRACs em termos de elementos constituintes (Leist, Zellner, 2006).....	50
Tabela 3 - Comparação dos QRACs em termos dos critérios adotados.....	51
Tabela 4 - Mapeamento das perspectivas da construção civil para sistemas de informação.....	53
Tabela 5 - Mapeamento entre IDEF3 e BPDM.....	82
Tabela 6 - Resumo da estrutura organizacional da ABC.....	113
Tabela 7 - Resumo das necessidades do projeto PR.....	120
Tabela 8 - Resumo das características do projeto PR.....	121
Tabela 9 - Resumo das etapas do caso de uso CA1.....	123

## **Lista de Siglas e Abreviações**

**BMM** – *Business Motivation Model*

**BPDM** – *Business Process Definition Metamodel*

**BPML** – *Business Process Modeling Language*

**BPMN** – *Business Process Modeling Notation*

**EPC** – *Event Driven Process Chain*

**ERS** – *Especificação dos Requisitos de Software*

**FCS** – *Fator Crítico de Sucesso*

**FEAF** – *Federal Enterprise Architecture Framework*

**IDEF3** – *Integrated DEFinition Method 3*

**OMG** – *Object Management Group*

**OSM** – *Organization Structure Metamodel*

**QRAC** – *Quadro de Referência para Arquitetura Corporativa*

**RAD** – *Role Activity Diagram*

**TEAF** – *Treasury Enterprise Architecture Framework*

**TISAF** – *Treasury Information Systems Architecture Framework*

**TOGAF** – *The Open Group Architecture Framework*

**UML** – *Unified Modeling Language*

**UOB** – *Unit of Behavior*



## Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>11</b>
1.1	Motivação	11
1.2	Objetivo	13
1.3	Resultados Esperados e Contribuições	14
1.4	Organização do Trabalho	15
<b>2</b>	<b>REQUISITO DE SOFTWARE</b>	<b>17</b>
2.1	Definição de Requisito de <i>Software</i>	17
2.2	Levantamento dos Requisitos de <i>Software</i>	19
2.3	Conclusão	20
<b>3</b>	<b>RASTREAMENTO DE REQUISITOS DE SOFTWARE</b>	<b>21</b>
3.1	Modelos de Referência	21
3.2	Estratégias de Rastreamento com Casos de Uso	27
3.3	Rastreamento em Projetos baseados em UML	31
3.4	Conclusão	33
<b>4</b>	<b>META-MODELO DE RASTREAMENTO DE REQUISITOS</b>	<b>34</b>
4.1	Considerações para a Criação do Meta-modelo de Rastreamento	34
4.2	Relacionamentos no Meta-modelo de Rastreamento	36
4.3	Elementos do Meta-modelo de Rastreamento	39
4.4	Meta-modelo de rastreamento de requisitos	41
4.5	Conclusão	44
<b>5</b>	<b>QUADRO DE REFERÊNCIA PARA ARQUITETURA CORPORATIVA</b>	<b>45</b>
5.1	Arquitetura Corporativa da Informação	45
5.2	Quadros de Referência para Arquitetura Corporativa (QRAC)	46
5.2.1	TOGAF – The Open Group Architecture Framework	46
5.2.2	Quadro de Referência de Zachman	47
5.2.3	TEAF – <i>Treasury Enterprise Architecture Framework</i>	48
5.2.4	CrITÉrios de Escolha para o Quadro de Referência	49
5.3	Quadro de Referência de Zachman	51
5.4	Rastreamento de Requisitos no Quadro de Referência de Zachman	55
5.5	Conclusão	56
<b>6</b>	<b>MODELO DO CONTEXTO ORGANIZACIONAL</b>	<b>57</b>
6.1	Hipóteses Adotadas	57
6.2	BPDM	62
6.2.1	Modelo de Composição	63
6.2.2	Modelo de Cursos	65
6.2.3	Modelo de Acontecimento e Mudança	67
6.2.4	Modelo de Comportamento de Processo	69
6.2.5	Modelo de Interação Simples	71
6.2.6	Modelo de Atividades	72
6.2.7	Extensões para BPMN	73
6.2.8	Modelo de Protocolo de Interação	74
6.3	IDEF3	75
6.3.1	Notação do IDEF3	76
6.3.2	Procedimento do IDEF3	78

6.4	Aplicação do método IDEF3 ao meta-modelo BPDM.....	79
6.5	BMM.....	82
6.5.1	Conceitos da área de Fins e Meios.....	83
6.5.2	Conceitos da área de Influenciadores.....	86
6.5.3	Relacionamento entre meios e fins.....	87
6.6	OSM.....	87
6.6.1	Modelo do Núcleo.....	88
6.6.2	Modelo de Entidade Legal.....	89
6.6.3	Modelo de Posição.....	91
6.6.4	Modelo de Unidade Organizacional.....	92
6.7	Representação do Modelo de Contexto Organizacional.....	92
6.8	Conclusão.....	97
<b>7</b>	<b>MODELO DE RASTREAMENTO DE REQUISITOS.....</b>	<b>98</b>
7.1	Base Racional.....	98
7.1.1	Meta-modelo IBIS.....	99
7.1.2	Modelo de Potts e Bruns.....	100
7.1.3	Modelo de Referência para Base Racional.....	100
7.1.4	Modelo do Laço de Racionalização.....	101
7.1.5	Modelo da Base Racional.....	102
7.2	Requisito.....	104
7.3	Integração entre os modelos.....	107
7.4	Conclusão.....	109
<b>8</b>	<b>EXEMPLO DE APLICAÇÃO DO MODELO.....</b>	<b>110</b>
8.1	Descrição do Exemplo.....	110
8.2	Contexto Organizacional do Problema.....	112
8.3	Contexto Organizacional Desejado.....	117
8.4	Necessidades dos Envolvidos.....	119
8.5	Características do Sistema.....	121
8.6	Requisitos de <i>Software</i> .....	122
8.7	Exemplos de Rastreamento.....	125
8.8	Conclusão.....	125
<b>9</b>	<b>CONCLUSÃO.....</b>	<b>126</b>
9.1	Resultados Obtidos.....	126
9.2	Trabalhos Futuros.....	129
	<b>REFERÊNCIAS.....</b>	<b>131</b>

# 1 Introdução

## 1.1 Motivação

O rastreamento de requisitos de *software* pode ser definido, segundo Gotel e Finkelstein (1994), como a habilidade de descrever e seguir o ciclo de vida de um requisito em ambas as direções (para frente e para trás). Isto é, permitir acompanhar um requisito desde sua origem, passando pelo seu desenvolvimento e especificação até sua subsequente implantação e uso, considerando todos os refinamentos e iterações ocorridos ao longo das diversas fases do desenvolvimento de um *software*.

Matthias Jarke (1998) divide o rastreamento de requisitos em quatro grupos:

- Rastreamento para frente a partir dos requisitos: a responsabilidade por implementar um requisito é atribuída aos componentes de software que serão desenvolvidos;
- Rastreamento para trás até os requisitos: verifica o sistema desenvolvido contra o que foi especificado, permitindo evitar desenvolvimento além do necessário;
- Rastreamento para frente até os requisitos: mudanças nas necessidades dos envolvidos ou nas decisões técnicas assumidas no início do processo de desenvolvimento refletem em alterações em toda a estrutura de requisitos e;
- Rastreamento para trás a partir dos requisitos: as decisões técnicas tomadas no levantamento dos requisitos podem influenciar a maneira como os envolvidos vêem suas próprias necessidades.

Gotel e Finkelstein (1994) identificam dois grupos distintos de rastreamento: pré e pós-especificação de requisitos (pré-ER e pós-ER, respectivamente). O rastreamento pós-ER disponibiliza informações relativas ao uso do requisito, como os produtos de *software* originados dele. O rastreamento pré-ER disponibiliza informações relativas ao processo de elaboração do requisito, como o responsável por ele, as premissas e decisões adotadas e o contexto em que o requisito foi criado (Pinheiro, 2003).

O rastreamento pré-ER é mais complexo e menos apoiado por métodos e ferramentas, se comparado ao rastreamento pós-ER (Gotel, Finkelstein, 1994). Entretanto, a falta ou a baixa qualidade do rastreamento pré-ER são reconhecidas como as principais causas da maioria dos problemas relacionados ao rastreamento de requisitos.

Os problemas originados pela falta ou baixa qualidade do rastreamento pré-ER aparecem em diversas formas. Especificar requisitos sem possuir uma noção concreta do contexto

organizacional em que eles se encaixam pode levar à criação de requisitos mal formulados e que não estejam de acordo com as estratégias e processos da organização. O contexto organizacional é a visão de negócios da organização, com seus processos, regras, políticas e intenções (Morgan, 2002).

Se a especificação de um requisito se dá em um contexto organizacional, e se a criação de um sistema tende a alterar este contexto organizacional, não registrar este contexto após a elaboração do requisito pode gerar os seguintes problemas:

- Dificuldade no entendimento do requisito e da razão de ele ter sido criado (rastreamento para trás a partir dos requisitos). Sem o contexto organizacional, resta apenas uma referência a quem o criou, o que sujeita o requisito às limitações humanas (de memória, por exemplo).
- Dificuldade na atualização de requisitos após mudanças no contexto organizacional (rastreamento para frente até os requisitos). Sem o conhecimento do relacionamento entre o contexto e os requisitos, alterações no contexto demandam uma revisão geral de todos os requisitos, em vez de somente aqueles relacionados à parte do contexto que foi alterada.

Diversos autores compartilham a preocupação de Gotel e Finkelstein (1994) com o rastreamento pré-ER. Ramesh e Jarke (2001) propõem modelos de referência de rastreabilidade de requisitos, incluindo modelos que forneçam suporte aos conceitos de suposições e decisões tomadas na elaboração dos requisitos. Patrício Letelier (2002) propõe meta-modelos de rastreabilidade em que os conceitos de responsabilidade pelos requisitos e de razões para suas elaborações são considerados. Bleistein, Cox e Verner (2005) propõem a criação de um modelo único de rastreabilidade, que inclua os rastreamentos pré e pós-ER, e que se baseiem em um modelo de intenções organizacionais.

Há, entretanto, a necessidade de se considerar o contexto organizacional por completo na especificação dos requisitos, e não apenas parte dele, como ocorre nos trabalhos citados (o enfoque é dado ao modelo de intenções da organização). Além disso, é preciso considerar a variável tempo no rastreamento, pois os contextos organizacionais sofrem constantes mudanças e é importante registrá-las.

## 1.2 Objetivo

O objetivo do trabalho é criar um modelo de rastreamento que permita relacionar requisitos de *software* e entidades de modelos de negócio, permitindo identificar o contexto organizacional em que os requisitos foram criados. Como parte da criação do modelo de rastreamento, é criado um meta-modelo de rastreamento relacionando requisitos, o contexto organizacional e a base racional por trás dos requisitos.

As entidades que representam os requisitos seguem o conceito apresentado por Leffingwell e Widrig (2003), em que as necessidades dos envolvidos dão origem a características de *software* e estas, por sua vez, dão origem às especificações dos requisitos de *software*. Embora Leffingwell dê enfoque ao rastreamento pós-ER, este tipo de rastreamento não está no escopo do presente trabalho.

As entidades que representam a base racional seguem os conceitos apresentados nos trabalhos de Ramesh e Jarke (2001), Conklin e Begeman (1998), Lee (1991) e Louridas e Loucopoulos (2000) e incluem questões, alternativas, argumentos e decisões.

As entidades que representam o contexto organizacional são extraídas de diferentes modelos adequados aos diversos aspectos da visão de negócio apresentados no quadro de referência de Zachman (Sowa, Zachman, 1992). Embora extraídas de diferentes modelos, as entidades de negócio formam uma arquitetura de negócio que por si só pode representar um contexto organizacional. O escopo do trabalho é limitado a apenas três dos seis aspectos previstos por Sowa e Zachman (1992). O aspecto dos processos é considerado o aspecto central do modelo de contexto organizacional e a ele são adicionados os aspectos de motivação e estruturação organizacional.

A utilidade do modelo de rastreamento proposto é permitir responder às seguintes perguntas sobre um requisito:

- Quais são as justificativas para a elaboração do requisito de *software*?
- Em que contexto organizacional o requisito de *software* foi elaborado?
- Como uma mudança no contexto organizacional pode influenciar o requisito de *software*?

### 1.3 Resultados Esperados e Contribuições

O resultado esperado para este trabalho é a criação de um modelo de rastreamento de requisitos de *software* pré-ER que situe os requisitos em um contexto organizacional, além de permitir o registro das discussões e decisões tomadas na especificação dos requisitos.

Tal modelo agrega os benefícios de duas abordagens relativas ao rastreamento pré-ER: os conceitos de necessidades dos envolvidos e características de *software* e o conceito de base racional. Desta forma, o modelo permite registrar as necessidades que motivaram os requisitos e também as discussões, alternativas, argumentos e decisões que ocorreram durante a especificação dos requisitos.

Entretanto, o modelo proposto no presente trabalho vai além e acrescenta o conceito do contexto organizacional como forma de se registrar mais profundamente as razões por detrás dos requisitos de *software*.

A criação do modelo do contexto organizacional não só traz informações mais precisas para a elaboração de requisitos como facilita o entendimento do domínio do problema por parte dos responsáveis pelo levantamento dos requisitos. Além disso, a própria atividade de modelagem do contexto organizacional pode revelar falhas nos processos da organização ou gerar um melhor entendimento dela, permitindo aos envolvidos resolver problemas até então não notados ou efetuar melhorias.

O presente trabalho aproveita esses conceitos e apresenta um meta-modelo de rastreamento que introduz o conceito do contexto organizacional como base dos requisitos e como referência para a base racional.

O quadro de referência de Zachman é escolhido como guia para a criação do modelo de contexto organizacional. Dos seis aspectos que compõem a visão de negócios do quadro de referência de Zachman, o trabalho limita-se a utilizar apenas três. O aspecto dos processos é considerado o mais importante aspecto de um contexto organizacional. A modelagem deste aspecto é gerada a partir do mapeamento entre um meta-modelo de processos (BPDm) e o método IDEF3. Completam o modelo do contexto organizacional outros dois aspectos: estruturação e motivação organizacional. Além de estarem intimamente ligados ao aspecto dos processos, esses dois aspectos possuem importantes características para o rastreamento de requisitos pré-ER. Alguns autores, como Bleistein, Cox e Verner (2005) e Morgan (2002), consideram a motivação organizacional como a principal fonte da especificação dos requisitos. Quanto à estruturação organizacional, tanto Letelier (2002) quanto Ramesh e Jarke

(2001) consideram fundamental a presença dos envolvidos no projeto no rastreamento de requisitos.

#### **1.4 Organização do Trabalho**

O trabalho está dividido nas seguintes partes:

A seção 2, Requisito de *Software*, apresenta o conceito de requisito de *software* utilizado neste trabalho. Além dos requisitos de software, são apresentados os conceitos de necessidades dos envolvidos no projeto e de características de *software*.

A seção 3, Rastreamento de Requisitos de Software, apresenta três abordagens, encontradas na literatura, para o rastreamento de requisitos. Estas abordagens apresentam fundamentos que são utilizados para a elaboração de um meta-modelo de rastreamento de requisitos.

A seção 4, Meta-modelo de Rastreamento de Requisitos, apresenta uma proposta de meta-modelo para o rastreamento de requisitos que situa os requisitos em um contexto organizacional. Além disso, posiciona a base racional como justificativa dos requisitos.

A seção 5, Quadro de Referência para Arquitetura Corporativa, introduz o conceito de quadros de referência para arquitetura corporativa como forma de guiar o processo de definição dos elementos que compõem o modelo do contexto organizacional.

A seção 6, Modelo do Contexto Organizacional, apresenta a escolha de modelos e meta-modelos que representam diferentes aspectos do contexto organizacional. O meta-modelo BPDM (*Business Process Definition Metamodel*) é utilizado, em conjunto com o método IDEF3, para a geração de um modelo que represente o aspecto dos processos da organização. O modelo BMM (*Business Motivation Model*) é utilizado como modelo que representa o aspecto motivacional da organização. O meta-modelo OSM (*Organization Structure Metamodel*) é adaptado para gerar um modelo estrutural da organização. Por fim, estes três modelos são agrupados em um único modelo que representa o modelo do contexto organizacional.

A seção 7, Modelo de Rastreamento de Requisitos, descreve os demais elementos apresentados no meta-modelo de rastreamento de requisitos proposto, complementando a proposta de criação de um modelo de rastreamento.

A seção 8, Exemplo de Aplicação do Modelo, apresenta um exemplo de aplicação do modelo de rastreamento.

A seção 9, Conclusão, apresenta as conclusões finais do trabalho, incluindo recomendações sobre aspectos que podem ser melhorados ou acrescentados ao trabalho.



## 2 Requisito de Software

Os requisitos de *software* são o objetivo final da atividade de análise em um projeto de *software*. Além disso, são elementos-chave em um modelo de rastreamento pré-ER, enfoque da proposta apresentada neste trabalho. Portanto, é necessário defini-lo com clareza, pois o termo será bastante utilizado no restante do trabalho.

Este capítulo apresenta uma definição de requisito de *software*, quais são suas características desejadas e qual é o seu papel no desenvolvimento de um *software*. Além disso, apresenta os conceitos de necessidade dos envolvidos no projeto e de característica de *software*, que fazem parte do processo de levantamento dos requisitos.

### 2.1 Definição de Requisito de *Software*

No presente trabalho adota-se a definição de requisito de *software* apresentada por Dorfman e Thayer (1990 apud Leffingwell e Widrig, 2003), segundo os quais um requisito de *software* é:

- a. Uma funcionalidade de *software*, desejada por um usuário, para resolver um problema ou atingir um objetivo.
- b. Uma capacidade de *software* que deve estar presente em um sistema para satisfazer um contrato, padrão, especificação ou outro tipo de documentação imposta ao projeto.

Faulk (1995) complementa a definição de Dorfman e Thayer afirmando que a especificação de um requisito de *software* deve descrever precisamente o que um *software* deve fazer, sem, entretanto, mencionar como.

Faulk afirma que, tão importante quanto a análise e levantamento dos requisitos, é a sua documentação formal. É comum formalizar os requisitos em um documento chamado Especificação dos Requisitos de *Software* (ERS). Parnas (1986 apud Faulk, 1995) apresenta os diversos papéis da ERS em um projeto:

- Para o cliente, a ERS registra exatamente o que deve ser entregue e pode ser usada como a base para um contrato de desenvolvimento de *software*.
- A ERS registra o resultado da atividade de análise do problema. Esta documentação, anterior ao início da codificação, permite que as dúvidas sejam sanadas na fase inicial do projeto, quando o custo para correção de erros é menor.
- Para o gerente de projeto, a ERS pode ser utilizada para a elaboração de cronogramas e verificação do andamento do projeto.

- Para a equipe de desenvolvimento, a ERS contém quais funcionalidades devem fazer parte do sistema e quais restrições se aplicam à solução.
- Para a equipe de testes, a ERS é a base para a criação dos planos de teste e é a documentação mais adequada contra a qual o sistema será validado.

Faulk ainda apresenta as características desejadas para uma ERS:

- **Completa:** a ERS define o conjunto de implementações aceitáveis e deve conter toda a informação necessária para o desenvolvimento do *software*. Qualquer implementação que esteja de acordo com a ERS é uma implementação válida. Portanto, uma ERS incompleta pode dar origem a um *software* também incompleto ou que não atenda de maneira adequada os anseios do cliente.
- **Independente da implementação:** a ERS não deve favorecer ou indicar uma tecnologia para a criação do *software*, a não ser que isso seja em si um requisito do projeto.
- **Consistente e não ambígua:** a ERS não pode estar sujeita a interpretações conflitantes e seus requisitos só podem ter uma possível interpretação.
- **Precisa:** os requisitos devem ser concisos e objetivos, definindo claramente as funcionalidades desejadas.
- **Verificável:** um requisito é verificável se é possível determinar se uma dada implementação satisfaz ou não este requisito.
- **Compreensível:** a ERS deve ser escrita em termos do problema em questão e deve ser compreendida tanto pelo cliente quanto pelos desenvolvedores. Desta forma, facilita-se a comunicação e o entendimento entre as partes.

Os requisitos de *software* também podem ser categorizados em funcionais e não-funcionais, como mencionam Faulk (1995), Leffingwell e Widrig (2003) e Spence e Probasco (2000). Um requisito funcional se refere a um comportamento desejado do sistema, com os dados de entrada e saída bem definidos. Um requisito não-funcional se refere aos outros tipos de restrições, como usabilidade, desempenho, facilidade de manutenção entre outros.

A ERS agrupa ambas as categorias de requisitos em um só documento. Outras propostas, como a do RUP, apresentadas nos trabalhos de Leffingwell e Widrig (2003) e Spence e Probasco (2000), mantêm as categorias de requisitos em documentos distintos. Os requisitos

funcionais são representados por casos de uso, e os não-funcionais, em um documento chamado Especificação Suplementar (*Supplementary Specification*).

## 2.2 Levantamento dos Requisitos de *Software*

Em um processo de levantamento de requisitos, dois domínios devem ser analisados, segundo Leffingwell (2003): o domínio do problema e o domínio da solução. No domínio do problema se encontram os envolvidos no projeto, cujas necessidades devem ser atendidas pela criação de um *software*. No processo de levantamento dos requisitos, este domínio deve ser bem compreendido e os requisitos de *software* devem ser escritos em uma linguagem compreensível para aqueles que pertencem a este domínio. O domínio da solução envolve os requisitos de *software* e o próprio sistema que será construído.

O primeiro passo no processo de levantamento de requisitos é compreender o domínio do problema e determinar quais são as necessidades dos envolvidos no projeto. Estas necessidades podem ser problemas que precisam ser resolvidos, melhorias desejadas ou até obrigações legais que devem ser cumpridas.

Com o conjunto de necessidades dos envolvidos determinado, pode-se adentrar o domínio da solução. Entretanto, nesta próxima etapa ainda não se determinam os requisitos de *software*. Antes disso, é preciso definir uma solução para o problema, em um alto nível e em linguagem de fácil compreensão para os envolvidos no projeto. Nesta etapa, portanto, são definidas as características que estarão presentes no *software*. Uma característica de *software* é uma descrição simples e objetiva do que o *software* fará ou conterà, e a sua determinação é muito útil para validar o entendimento do problema e para apresentar uma visão geral da solução.

Após o entendimento do problema e a aprovação da solução adotada, passa-se à elaboração dos requisitos de *software*. Os requisitos são gerados a partir do detalhamento das características, até que atinjam granularidade suficiente para conter toda a informação necessária para a codificação do *software*. Esta informação abrange tanto os requisitos funcionais como os não-funcionais.

Leffingwell considera que o número de necessidades dos envolvidos no projeto não deve ser muito grande. As características derivadas destas necessidades aumentam um pouco em número, mas ainda assim mantendo-se em quantidade suficientemente baixa para permitir uma rápida leitura da solução. Por fim, o número de requisitos de *software* cresce bastante em função do detalhamento das características. Leffingwell representa estes conceitos na forma de uma pirâmide, como ilustra a Figura 1.

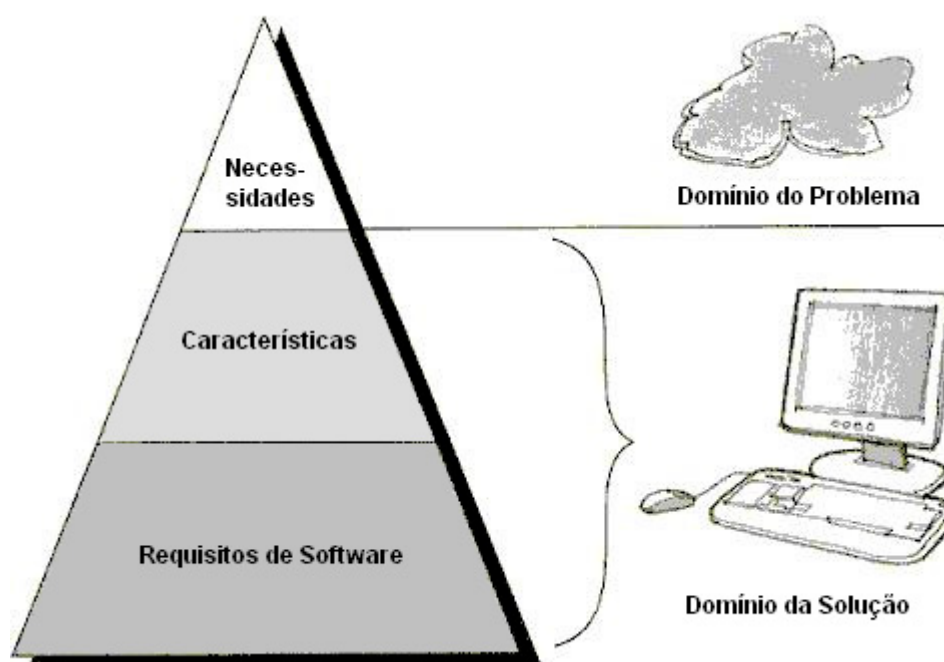


Figura 1 - Pirâmide de requisitos de *software* (Leffingwell, 2003)

Além da definição de requisito de *software* apresentada por Dorfman e Thayer e estendida por Faulk, no presente trabalho adota-se também o relacionamento entre requisitos de *software* e características do sistema e necessidades dos envolvidos, conforme os conceitos apresentados por Leffingwell.

### 2.3 Conclusão

Este capítulo apresentou o que é um requisito de *software*, qual é o seu papel em um projeto e quais são suas características desejadas. Apresentou também, de forma sucinta, as etapas do processo de levantamento dos requisitos.

Embora este capítulo ainda não tenha discutido o rastreamento dos requisitos, a pirâmide dos requisitos apresentada denota a importância de elementos especificados em etapas anteriores à etapa de especificação dos requisitos em si. A existência e relevância destes elementos, que são as necessidades dos envolvidos e as características de *software*, indicam a importância do rastreamento de requisitos pré-ER.

### 3 Rastreamento de Requisitos de Software

O capítulo 1 apresentou as diversas categorias de rastreamento de requisitos, assim como as vantagens de sua utilização. Este capítulo apresenta diferentes abordagens para o rastreamento de requisitos encontradas na literatura.

Ramesh e Jarke (2001) propõem a criação de modelos de referência para o rastreamento de requisitos, permitindo ao responsável pelo rastreamento escolher um modelo que se ajuste melhor à realidade de sua organização.

Spence e Probasco (2000) apresentam diversas estratégias para o rastreamento de requisitos em projetos nos quais os requisitos são representados em forma de casos de uso.

Letelier (2002) apresenta um meta-modelo de rastreamento de requisitos para projetos que utilizam UML como notação para representação dos requisitos.

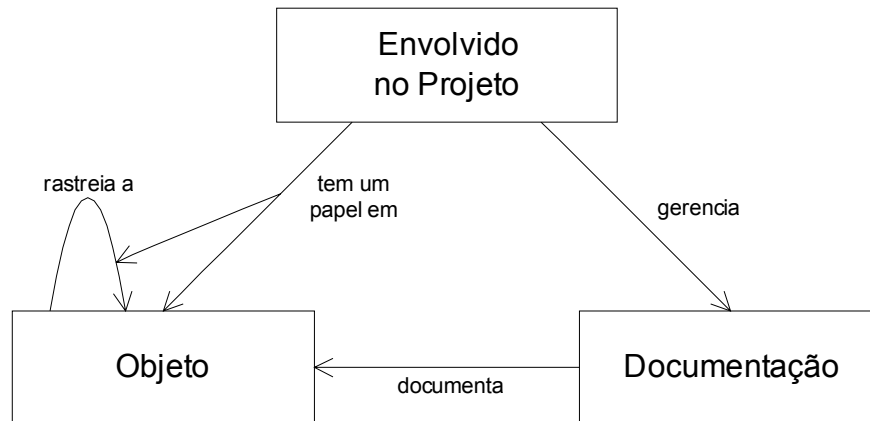
#### 3.1 Modelos de Referência

Ramesh e Jarke (2001) definem modelos de referência como protótipos de modelos adequados a diversos domínios e seguindo as regras de um único meta-modelo. A criação de modelos de referência tem como propósito reduzir o esforço de se criar modelos para cada nova aplicação ou sistema. Com estes modelos, o usuário não precisa criar um modelo completamente novo, basta selecionar as partes relevantes de um dos modelos de referência e adaptá-las para o problema em questão.

Em seu estudo, Ramesh e Jarke consideram três aspectos importantes para o rastreamento de requisitos:

- **Documentação:** representa os artefatos físicos nos quais a informação dos requisitos e do rastreamento é armazenada.
- **Envolvido no Projeto:** é o agente responsável por gerenciar o rastreamento de requisitos, além de possuir um papel em relação a alguns objetos presentes no rastreamento.
- **Objeto:** representa todos os tipos de objetos e informações, incluindo os requisitos de *software*, que poderão ser rastreados.

A Figura 2 ilustra o agrupamento destes três aspectos na forma de um meta-modelo de rastreamento de requisitos.



**Figura 2 - Meta-modelo de rastreamento de requisitos (Ramesh, Jarke, 2001)**

Segundo Ramesh e Jarke, este meta-modelo pode representar as seis dimensões do rastreamento de requisitos:

- **O quê?:** representa qual tipo de informação, com seus atributos e características, deve fazer parte do modelo de rastreamento.

No meta-modelo, um objeto representa as entradas e saídas do processo de desenvolvimento de *software*. Exemplos de objetos são requisitos, decisões, hipóteses, alternativas, componentes do sistema, entre outros.

- **Quem?:** representa os agentes que possuem diversos papéis na criação e manutenção dos objetos e do rastreamento entre eles.

No meta-modelo, um envolvido no projeto representa os agentes envolvidos nas diversas etapas do processo de desenvolvimento de *software*. Exemplos de envolvidos no projeto são os gerentes, analistas de sistemas, entre outros.

- **Onde?:** representa o local onde as informações do rastreamento são armazenadas.

Todos os objetos relevantes ao rastreamento devem estar presentes na documentação, que pode ser uma mídia física, como um texto, ou algo intangível, como a cultura de uma organização. Exemplos de documentação são a ERS, minuta de reunião, chamada telefônica, entre outros.

- **Como?:** representa como as informações do rastreamento são apresentadas.

A documentação do rastreamento de requisitos pode variar em grau de formalidade e em tipo de apresentação. A ERS, por exemplo, pode ser um documento de texto, mas outras especificações de requisitos podem estar na forma de casos de uso.

- **Por quê?:** representa a razão da criação ou alteração de um objeto do rastreamento.

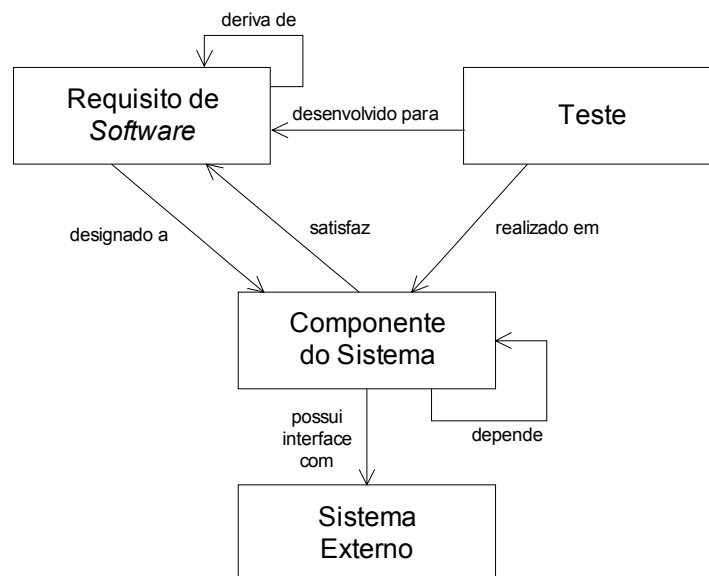
A motivação para a criação ou modificação de um objeto no modelo de rastreamento pode ser registrada por um outro objeto, como uma hipótese, um argumento ou uma alternativa.

- **Quando?:** representa o momento em que a informação sobre o rastreamento foi gerada.

As informações temporais das entidades e relacionamentos do modelo de rastreamento podem ser registradas como atributos destes elementos.

A partir deste meta-modelo, Ramesh e Jarke adotaram uma abordagem empírica, incluindo visitas e entrevistas em diversas empresas, para a criação dos modelos de referência. O enfoque adotado, entretanto, foi apenas da entidade objeto. As entidades documentação e envolvido no projeto não constam nos modelos de referência. O resultado final foi separado em duas categorias, dependendo do nível de rastreamento desejado.

Os usuários de baixo nível de rastreamento de requisitos não se interessam ou não estão dispostos a despende esforços para determinar a motivação por trás dos requisitos. Desta forma, o modelo de referência se torna mais simples, como ilustra a Figura 3.



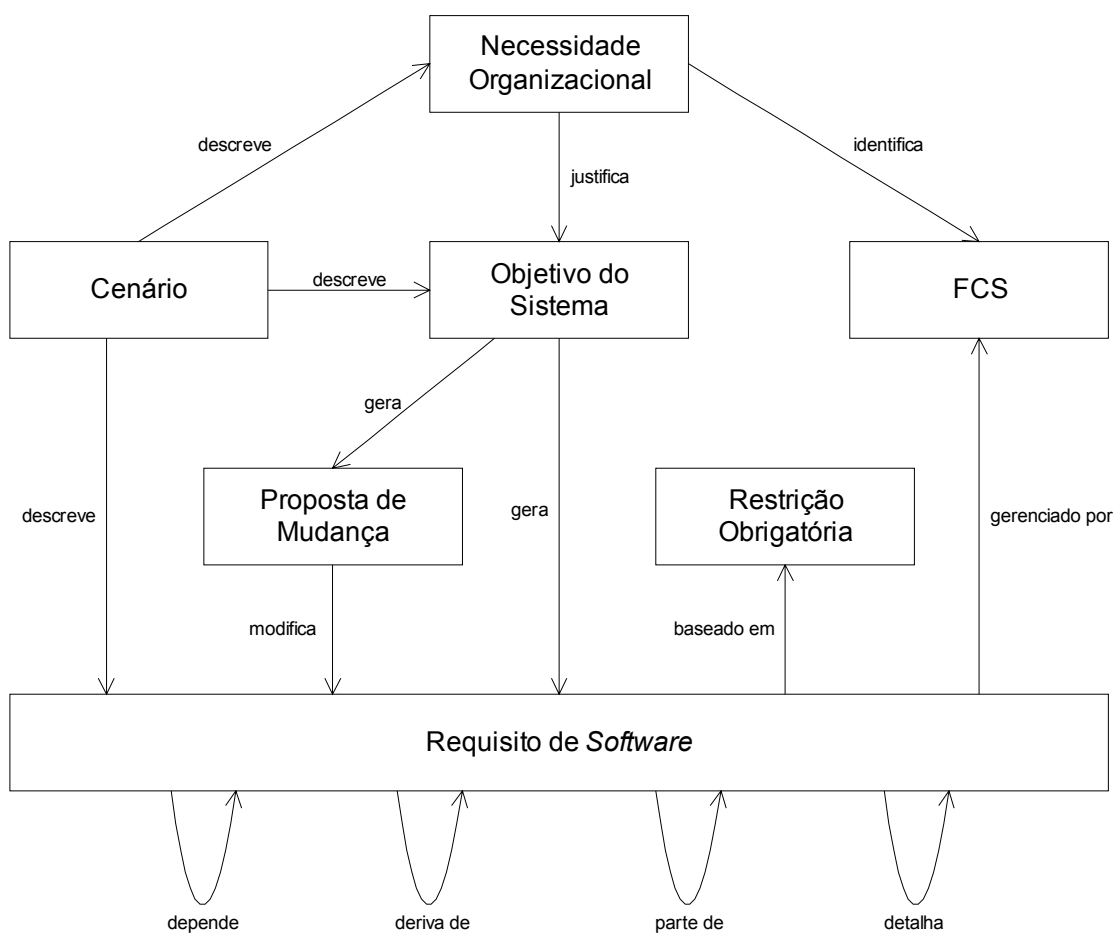
**Figura 3 - Modelo de baixo nível de rastreamento (Ramesh, Jarke, 2001)**

Para os usuários de baixo nível de rastreamento, o mais importante é designar cada requisito de *software* a um componente do sistema. Antes disso, entretanto, é preciso refinar e detalhar os requisitos de alto nível. Portanto, os requisitos de baixo nível são derivados dos de alto nível. Além disso, deve sempre haver um componente de sistema que satisfaça cada requisito de *software*.

Os testes são desenvolvidos para cada requisito de *software*, mas são realizados nos componentes do sistema. Por fim, os componentes do sistema apresentam dependências entre si e podem possuir interfaces com sistemas externos.

Para os usuários de alto nível de rastreamento de requisitos, há mais elementos para se modelar. Portanto, em prol da clareza de leitura do modelo, Ramesh e Jarke agruparam os elementos em quatro modelos: gerenciamento de requisitos, base racional, sistema e teste.

A Figura 4 ilustra o modelo de gerenciamento de requisitos.



**Figura 4 - Modelo de gerenciamento de requisitos (Ramesh, Jarke, 2001)**

Na visão dos usuários de alto nível de rastreamento de requisitos, um *software* é construído para satisfazer as necessidades da organização. Portanto, um objetivo do sistema deve ser justificado por uma necessidade organizacional. Além de gerar os requisitos de *software*, o objetivo do sistema dá origem às propostas de mudança, que por sua vez resultam em alterações nos requisitos. Embora a terminologia seja diferente, os termos necessidade organizacional e objetivo do sistema equivalem, respectivamente, aos termos necessidade dos envolvidos e característica do sistema, definidos no capítulo 2. Ramesh e Jarke ainda definem

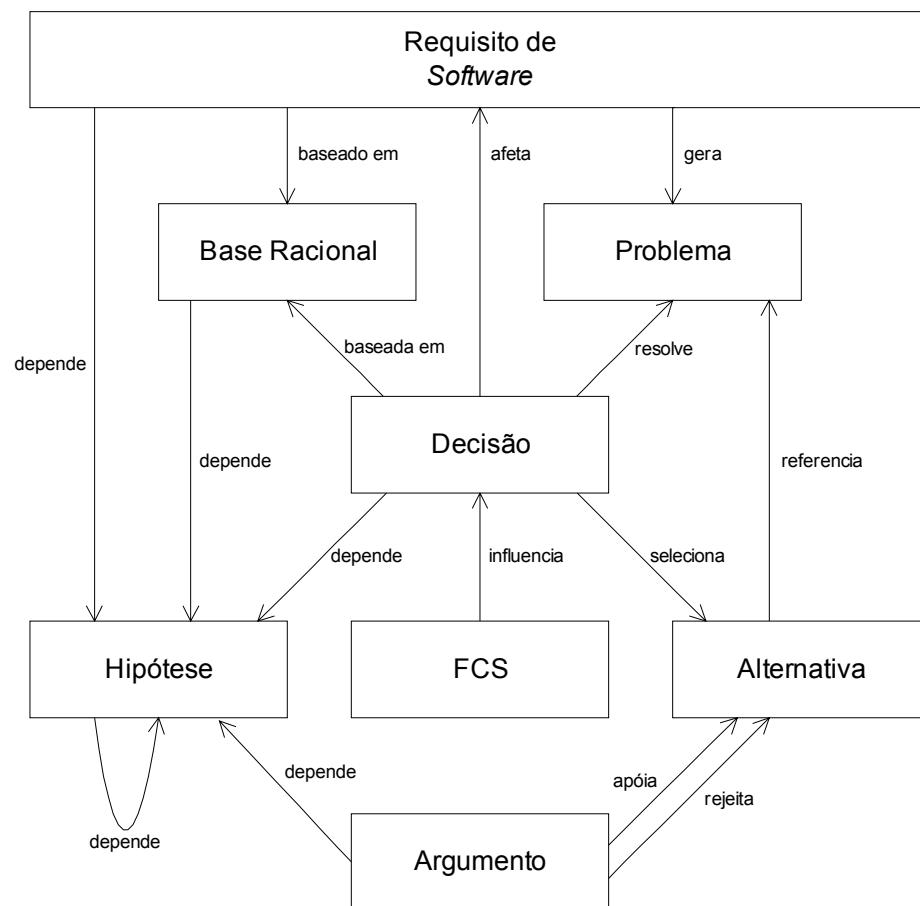


o elemento cenário, que descreve necessidades organizacionais, objetivos do sistema ou requisitos de *software*.

Devido ao elevado número de requisitos em sistemas complexos, é importante determinar os fatores críticos de sucesso (FCS) para o projeto, e gerenciar os requisitos com base nestes fatores. Exemplos de FCSs são custo e prazo, entre outros.

Por fim, alguns requisitos de *software* são baseados em restrições obrigatórias, como padrões e políticas da organização. Além disso, os requisitos possuem relação de dependência, derivação, composição e detalhamento entre si.

A Figura 5 ilustra o modelo de base racional.



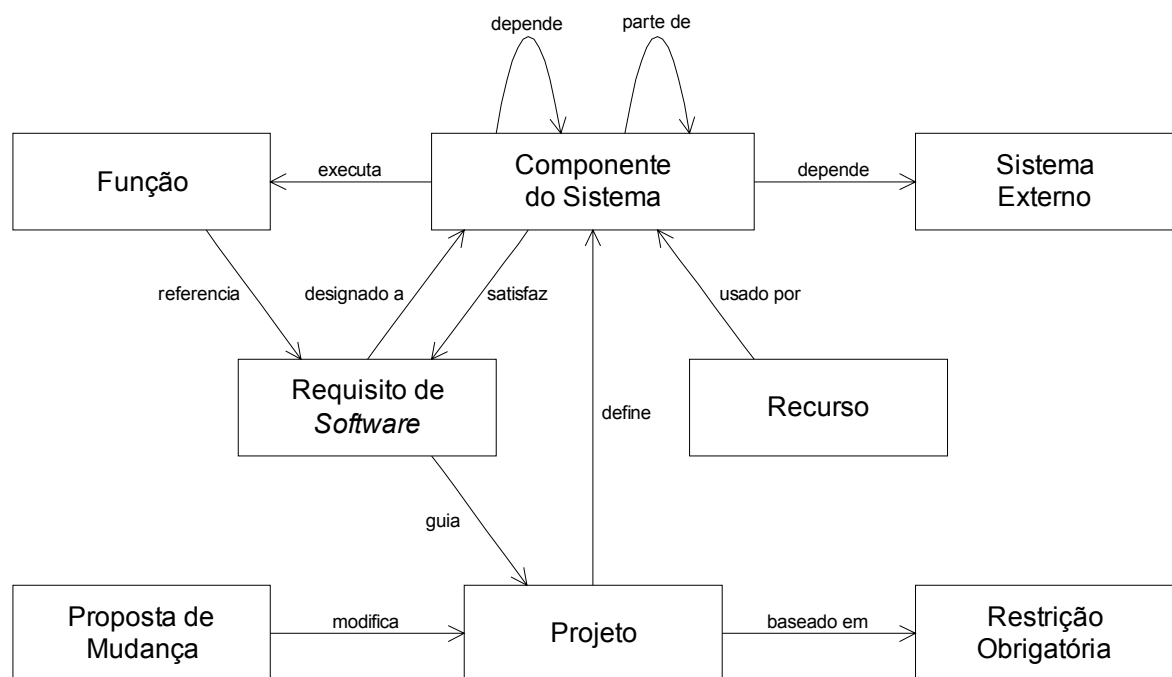
**Figura 5 - Modelo de base racional (Ramesh, Jarke, 2001)**

Segundo Ramesh e Jarke, a especificação ou modificação de um requisito de *software* pode gerar um novo problema, devido ao conflito de interesses ou interpretações dos envolvidos no projeto. Ramesh e Jarke advogam que é importante registrar as decisões tomadas em relação aos problemas que surgem, pois estas decisões podem afetar os requisitos de *software*.

Deve-se também modelar as alternativas apresentadas para os problemas, assim como os argumentos favoráveis e contrários. Normalmente, a decisão de escolher uma das alternativas é influenciada pelos FCSs.

Outro elemento importante do modelo são as hipóteses, pois delas dependem outras hipóteses, os requisitos de *software*, as decisões tomadas, os argumentos utilizados e a base racional por detrás de outros elementos. Por fim, o elemento denominado base racional é a base de criação dos requisitos de *software* e das tomadas de decisão.

A Figura 6 ilustra o modelo de sistema.



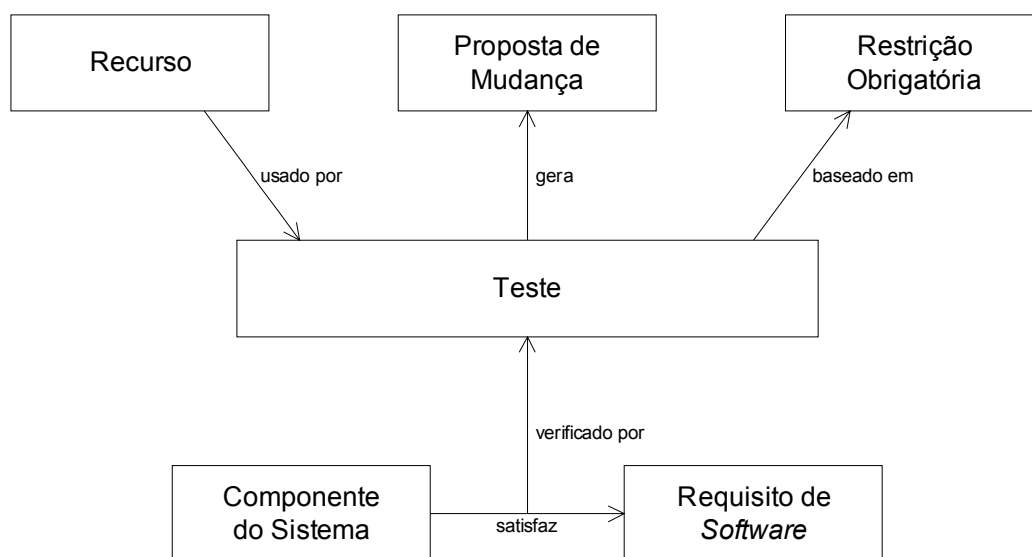
**Figura 6 - Modelo de sistema (Ramesh, Jarke, 2001)**

No modelo de sistema, o termo projeto se refere a qualquer atividade que produza um artefato de *software*, incluindo a codificação. O projeto é guiado pelos requisitos de *software* e baseado em restrições obrigatórias, como padrões e políticas da organização. Assim como as propostas de mudança podem alterar um requisito, elas podem modificar o projeto.

Os componentes podem depender de outros componentes e sistemas externos ou fazer parte de um componente maior. São definidos pelo projeto e satisfazem os requisitos de *software*. Ramesh e Jarke identificaram a importância de se modelar as funções executadas pelos componentes, pois muitas delas referenciam importantes requisitos funcionais.

Por fim, os recursos da organização, como dinheiro ou pessoal, são aplicados na construção ou no uso dos componentes.

A Figura 7 ilustra o modelo de teste.



**Figura 7 - Modelo de teste (Ramesh, Jarke, 2001)**

Segundo Ramesh e Jarke, os procedimentos de teste são escritos e executados para verificar se os componentes do sistema satisfazem os requisitos de *software*. Além disso, os testes demandam recursos da organização e são baseados nas restrições obrigatórias. Por fim, os resultados obtidos nos testes podem gerar novas propostas de mudança.

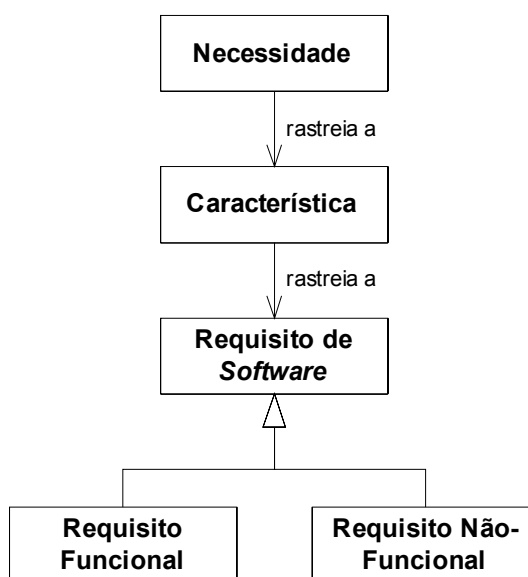
### 3.2 Estratégias de Rastreamento com Casos de Uso

Para Spence e Probasco (2000), um item rastreável é qualquer texto ou elemento de modelo que precisa ser explicitamente rastreado até outro texto ou item de modelo, de forma a se manter um registro da dependência entre eles.

Spence e Probasco dividem o rastreamento em dois grupos: rastreamento implícito e explícito. O rastreamento implícito se dá entre artefatos com uma relação formal bem estabelecida entre si, como, por exemplo, uma classe em um modelo de sistema e sua respectiva implementação. O rastreamento explícito, por outro lado, é aquele que é registrado em uma ferramenta ou modelo.

Ao optar pelo uso de rastreamento de requisitos, a organização deve decidir qual o nível de rastreamento explícito desejado. Esta decisão reflete a estratégia de rastreamento a se adotar. Em seu trabalho, Spence e Probasco apresentam diversas estratégias para o rastreamento de requisitos de *software*.

- **Não utilizando modelo de caso de uso**



**Figura 8 - Estratégia sem utilização de modelo de caso de uso (Spence, Probasco, 2000)**

Nesta estratégia, representada na Figura 8, as necessidades dos envolvidos são a base das características que, por sua vez, são detalhadas em requisitos de *software*, funcionais ou não-funcionais. Como não utilizam casos de uso, os requisitos de *software* são normalmente documentados em uma ERS.

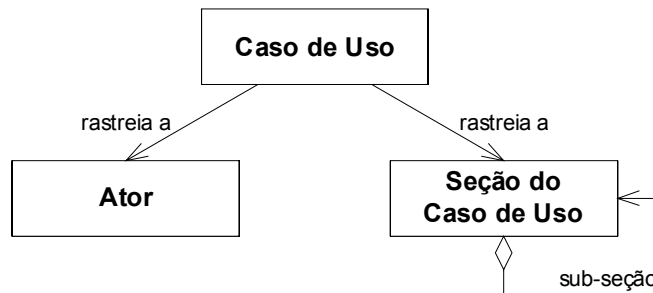
O nível de rastreamento explícito é alto, pois, segundo Spence e Probasco, uma ERS pode conter muitos registros que devem ser rastreados individualmente. O nível de completude desta estratégia é baixa, pois é difícil garantir que uma ERS, com seus muitos registros, aborde todos os requisitos necessários.

Como vantagens, esta estratégia tem o fato de ser bem conhecida e recomendada por diversos padrões. Além disso, permite um rastreamento muito detalhado e formal, o que a torna útil na elaboração de contratos. Como desvantagens, é difícil de terminar a análise dos requisitos e de entender o resultado desta análise. Pela própria característica da ERS, os requisitos não pertencem a um contexto, o que dificulta agrupá-los por funcionalidade e dificulta efetuar uma análise de impacto em caso de mudança em alguns requisitos.

- **Utilizando apenas modelo de caso de uso**

Nesta estratégia, representada na Figura 9, o modelo de caso de uso e a especificação suplementar representam o resultado completo da análise de requisitos. Um caso de uso identifica os atores atrelados a ele e é dividido em seções, como os caminhos de sucesso,

alternativos e de exceção. Devido à sua simplicidade, esta estratégia é comumente utilizada em projetos internos ou com parceiros de confiança.

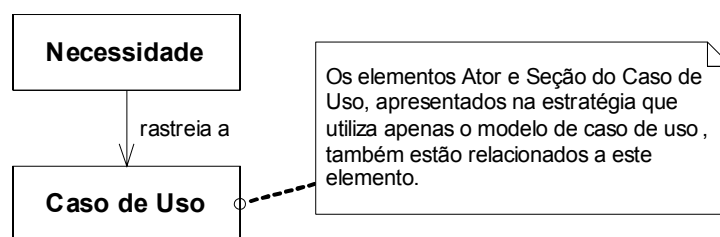


**Figura 9 - Estratégia utilizando apenas modelo de caso de uso (Spence, Probasco, 2000)**

Não há necessidade de rastreamento explícito neste caso e a completude do modelo é baixa. Embora a utilização de casos de uso facilite a construção de modelos de requisitos completos, a falta de rastreamento para as necessidades dos envolvidos pode levar à criação de um sistema incompleto, ou que não atenda adequadamente aos interesses do cliente.

As vantagens desta estratégia são o número reduzido de documentações e a facilidade de entendimento dos modelos. Como desvantagens, por não possuir uma relação com as necessidades dos envolvidos, é difícil garantir que o modelo represente o sistema que se quer construir. E, pelo baixo grau de formalidade, não é normalmente aceito como base para um contrato.

- **Modelo de caso de uso definindo as características do sistema**



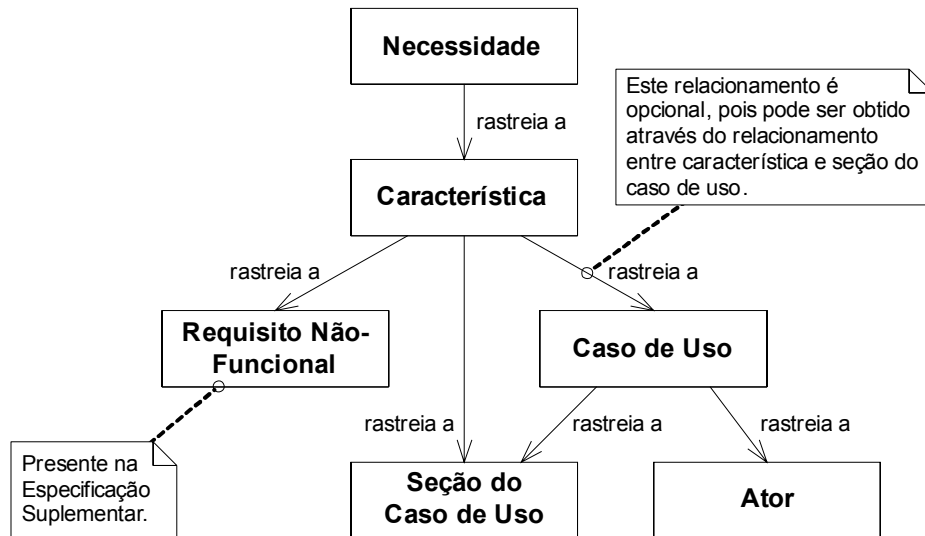
**Figura 10 - Estratégia com caso de uso definindo as características do sistema (Spence, Probasco, 2000)**

Nesta estratégia, representada na Figura 10, o modelo de caso de uso representa tanto as características de *software* como os requisitos de *software*. É uma variação da estratégia com apenas o modelo de caso de uso, acrescentando-se as necessidades dos envolvidos.

O nível de rastreamento explícito é baixo, pois existe apenas entre as necessidades, que geralmente estão em pequeno número, e os modelos de caso de uso. O nível de completude é médio, pois a presença das necessidades ajuda a garantir a validade do modelo.

Como vantagens, além das facilidades apresentadas na estratégia com apenas modelo de caso de uso, há ainda as necessidades, tornando o sistema mais próximo aos interesses do cliente. As desvantagens estão na utilização de casos de uso como características de *software*. Segundo Spence e Probasco, esta idéia parece razoável no início dos projetos, mas com o aumento ou modificação das características, pode se tornar catastrófica.

- **Características guiando o modelo de caso de uso**



**Figura 11 - Estratégia com as características guiando o modelo de caso de uso (Spence, Probasco, 2000)**

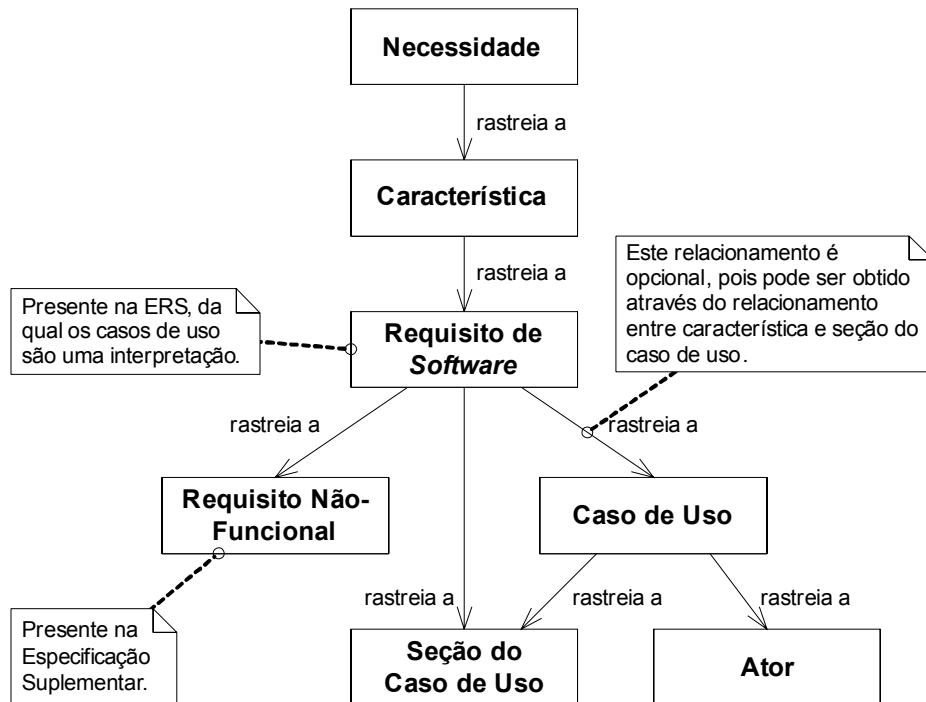
Nesta estratégia, representada na Figura 11 e recomendada por Spence e Probasco, os casos de uso e a especificação suplementar cumprem o papel da ERS. As necessidades e características, descritas em um artefato chamado de Documento Visão, são rastreadas até os casos de uso ou até a especificação suplementar.

O nível de rastreamento explícito é médio, pois é preciso registrar os relacionamentos entre necessidades, características e casos de uso ou requisitos não-funcionais. A completeza é alta pois, além dos casos de uso, há ainda as necessidades e características, representando os interesses do cliente e a proposta de solução.

Como vantagens, acrescenta escalabilidade à estratégia com os casos de uso no papel das características. A principal desvantagem é não ser sempre aceito como base para contratos.

- **Modelo de caso de uso sendo uma interpretação da ERS**

Estratégia utilizada em situações nas quais é obrigatória a produção de uma ERS, como representado na Figura 12, mas se quer modelar os requisitos na forma de casos de uso para facilitar o entendimento.



**Figura 12 - Estratégia com casos de uso interpretando a ERS (Spence, Probasco, 2000)**

O nível de rastreamento explícito é muito alto, pois agrega todo o rastreamento da estratégia sem utilização de casos de uso com o mapeamento entre ERS e o modelo de casos de uso. O nível de completudeza é muito alto, principalmente devido aos casos de uso.

A principal vantagem está em tornar formal, e conseqüentemente aceitável em contratos, um modelo baseado em casos de uso. A desvantagem está no excesso de documentação gerada e a confusão em se misturar duas abordagens distintas.

- **Modelo de caso de uso agrupando diversas ERS**

Esta estratégia é análoga à estratégia com o modelo de casos de uso interpretando a ERS, Neste caso, entretanto, há múltiplas SRSs, originadas de grupos distintos de envolvidos no projeto. Assim como a estratégia com uma única ERS, os níveis de rastreamento explícito e de completudeza são muito altos.

### 3.3 Rastreamento em Projetos baseados em UML

Patrício Letelier (2002) propõe um meta-modelo de rastreamento de requisitos para projetos que utilizam a notação UML para a especificação dos requisitos. Em seu meta-modelo, Letelier inclui o que considera os três importantes tipos de informação para o rastreamento de requisitos:

- Vínculos de rastreamento entre especificações de tipos distintos permitem verificar se o sistema atende às expectativas dos envolvidos no projeto e se não há implementação desnecessária. Além disso, facilitam a análise de impacto no projeto quando os requisitos sofrem alteração.
- Os rastreamentos entre os envolvidos no projeto e as especificações melhoram a comunicação entre as pessoas, garantindo que a contribuição de cada um seja devidamente analisada e registrada.
- A elaboração de uma base racional para as especificações, incluindo alternativas, decisões e hipóteses, contribui para melhorar o entendimento do sistema por parte dos envolvidos no projeto e para facilitar possíveis mudanças nos requisitos sem a necessidade de reavaliar alternativas já descartadas.

A Figura 13 ilustra o meta-modelo de rastreamento proposto por Letelier.

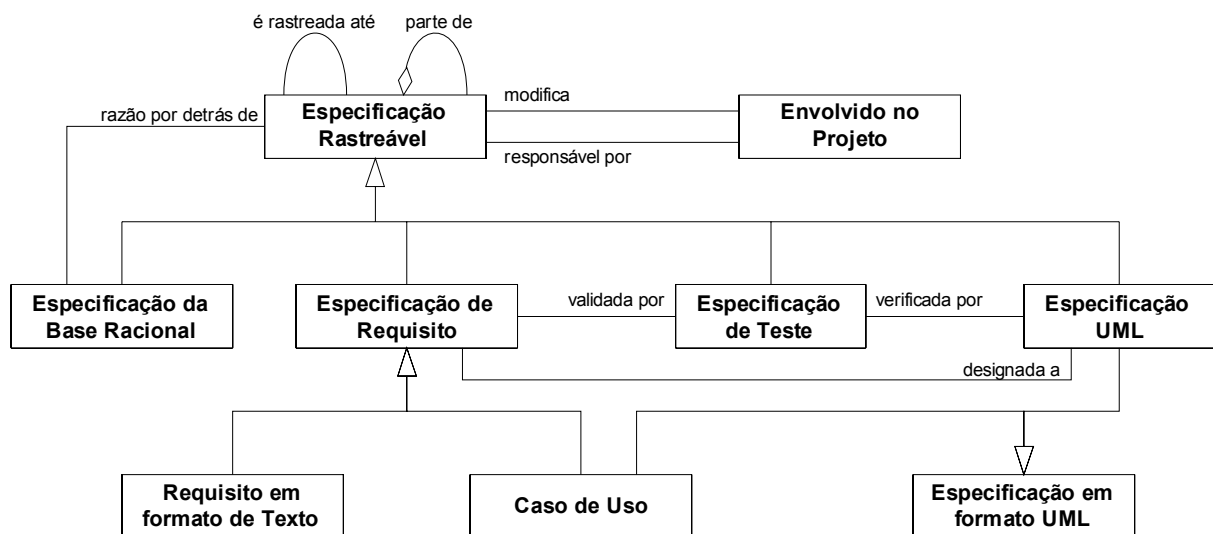


Figura 13 - Meta-modelo de rastreamento de requisitos (Letelier, 2002)

Duas entidades têm maior relevância no meta-modelo: envolvido no projeto e especificação rastreável. Um envolvido no projeto é responsável por criar e modificar especificações. Uma especificação rastreável é qualquer especificação de *software*, como um documento, um modelo, um caso de uso, uma classe, entre outros. Uma especificação rastreável pode apresentar diversas formas, como uma base racional, um requisito, um teste ou uma especificação qualquer em formato UML.

A especificação de requisito pode representar um único requisito ou um grupo de requisitos. Dependendo da forma como é representado, um requisito pode ser classificado como um caso de uso ou como um requisito em formato de texto. A especificação da base racional



representa os fundamentos, alternativas ou hipóteses associados a uma especificação rastreável. Por fim, a especificação de teste define os testes que serão realizados para validar os requisitos e verificar as demais especificações em UML.

### **3.4 Conclusão**

Este capítulo apresentou três abordagens para o rastreamento de requisitos de *software*. Os modelos de referência de Ramesh e Jarke (2001) denotam a preocupação dos autores com o rastreamento pré-ER, por meio do modelo de base racional. Spence e Probasco (2000) consideram o rastreamento pré-ER importante, mas os únicos elementos destinados a eles são as necessidades dos envolvidos e as características de *software*. Patrício Letelier (2002) se preocupa mais com o rastreamento pós-ER e com a adequação de seu meta-modelo à notação UML, embora reconheça a importância do rastreamento pré-ER pela utilização da entidade de especificação da base racional.

Embora os autores supra-citados denotem um interesse, em maior ou menor grau, ao rastreamento pré-ER, nenhum deles considera o contexto organizacional em que os requisitos de *software* são elaborados. As necessidades dos envolvidos, as hipóteses, alternativas, argumentos e outros elementos da base racional são registrados, mas não os processos, atividades, estruturação e motivação organizacional.

Desta forma, as necessidades dos envolvidos e a base racional considerada ficam desconectadas da organização. Se o intuito do rastreamento pré-ER é garantir que os requisitos de *software* representem adequadamente os interesses da organização, tal desconexão deve ser evitada.

## 4 Meta-modelo de Rastreamento de Requisitos

Os modelos e meta-modelos apresentados no Capítulo 3 não posicionam os requisitos de *software* em um contexto organizacional. Dessa forma, não é possível rastrear um requisito de *software* até a entidade organizacional que deu origem a ele.

Este capítulo apresenta uma proposta de meta-modelo de rastreamento de requisitos pré-ER que situa requisitos no contexto organizacional. O meta-modelo é a base do modelo proposto nos capítulos 6 e 7 do presente trabalho.

### 4.1 Considerações para a Criação do Meta-modelo de Rastreamento

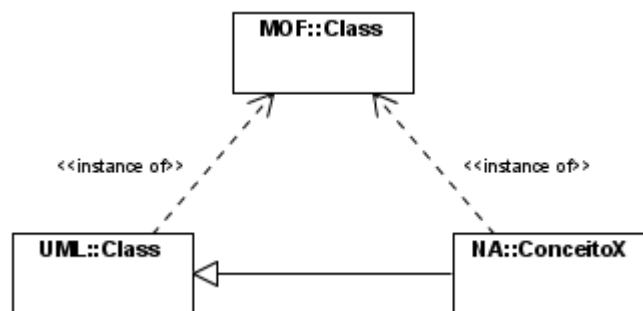
A principal razão para a criação de um meta-modelo de rastreamento de requisitos, como um passo anterior à criação de um modelo particular de rastreamento, é de ordem prática. Por exemplo, na implantação de um processo de rastreamento de requisitos em uma empresa é razoável ter-se facilidades de configuração para o processo, baseadas em um determinado modelo de rastreamento de requisitos mais apropriado para as características da empresa. Mais apropriado aqui significa, em termos pragmáticos, o emprego de conceitos e suas relações essenciais existentes no domínio de conhecimento e de experiência da empresa. Desse modo, a obtenção de um modelo de rastreamento de requisitos apropriado deve resultar de uma ampla pesquisa apoiada em estudos de caso, cujo objetivo principal é o levantamento das características comuns de empresas de um certo domínio de conhecimento da perspectiva do rastreamento de requisitos.

No presente trabalho, investiga-se e propõe-se um conjunto básico de conceitos e de seus relacionamentos que supõe-se estarem presentes nos mais variados modelos de rastreamento de requisitos. Em outras palavras, opera-se em um nível de abstração mais elevado, em que se abstraem características específicas dos vários domínios de conhecimento subjacentes aos elementos de requisitos rastreáveis no mundo concreto das empresas. Como, de modo informal, um meta-modelo permite a definição dos conceitos com os quais pode-se criar modelos apropriados, o nível de abstração adotado é o da meta-modelagem. Propõe-se, portanto, um meta-modelo de rastreamento de requisitos que se acredita ter força expressiva capaz de contribuir para a elaboração dos modelos mencionados.

Como o presente trabalho está ancorado na UML, adota-se na definição do meta-modelo de rastreamento de requisitos uma extensão da UML evitando-se a definição de uma nova linguagem para expressar o rastreamento. Tal extensão deve ser realizada no nível de

abstração do meta-modelo da UML. O OMG (2007a, 2007b) preconiza dois modos principais para a extensão da UML: extensão baseada no próprio meta-modelo da UML e extensão baseada no uso de perfis da UML. Como se trata de um trabalho inicial e, de certa maneira, informal, adota-se o primeiro modo de extensão, uma vez que a definição de um perfil exige um grau de formalismo que requer o amadurecimento dos conceitos aqui elaborados.

De modo prático, os conceitos essenciais de um modelo de rastreamento de requisitos, isto é, de uma instância do meta-modelo de rastreamento de requisitos, serão expressos por classes da UML. No meta-modelo da UML, um elemento *UML::Class* é uma instância do elemento *MOF::Class*<sup>1</sup>. A Figura 14 ilustra o modo básico recomendado para a extensão (ou adaptação) do meta-modelo da UML com uma meta-classe específica, com seus conceitos particulares de uma determinada adaptação de interesse, por exemplo identificada pelo espaço de nomes *NA* (para *Nova Adaptação*). Na adaptação ilustrada, define-se um novo constructo da linguagem, o *NA::ConceitoX*, no caso uma especialização (uma sub-classe) do elemento *UML::Class*. Utiliza-se aqui o mecanismo de herança definido no MOF (OMG, 2003).



**Figura 14 - Adaptação através de extensão do meta-modelo da UML**

A representação do novo constructo *NA::ConceitoX*, ou seja, a sua sintaxe concreta, pode ser variada. No presente trabalho adota-se o estilo apresentado na Figura 15. A representação utiliza um dos modos permitidos no MOF, em que se representa o novo constructo através de um estereótipo, cujo nome adota uma abreviação de acordo com uma convenção acordada entre os usuários do constructo. Em vez de se utilizar o estereótipo «*NA::ConceitoX*», adota-se o estereótipo «*ConceitoX*» convencioneando-se que ele denota o mesmo elemento cujo estereótipo é «*NA::ConceitoX*». *ConceitoX*, apesar de ser uma *UML::Class*, pode ter definições específicas para um determinado propósito, com novos atributos e restrições no nível da meta-classe. No entanto, no presente trabalho não serão elaboradas tais

<sup>1</sup> MOF é uma abreviatura de *Meta Object Facility* (OMG,2006). O fato de tanto o meta-modelo da UML, como o meta-meta-modelo MOF, terem o mesmo nome *Class* para um de seus elementos, pode eventualmente causar alguma ambigüidade, mesmo que suas definições sejam relativamente distintas, pertencendo a níveis de abstração distintos.

especializações nos constructos propostos no meta-modelo de requisitos. Adota-se uma interpretação de mais alto nível de abstração, em que são ignoradas as características específicas de tais meta-classes.

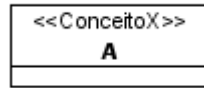


Figura 15 - Representação da adaptação através de estereótipo

#### 4.2 Relacionamentos no Meta-modelo de Rastreamento

Por se tratar de um meta-modelo de rastreamento, seus elementos devem ser considerados rastreáveis e os relacionamentos entre eles devem ser relacionamentos de rastreamento. No presente trabalho, um relacionamento de rastreamento é um relacionamento entre dois elementos no qual um dos elementos exerce uma influência na criação ou existência do outro.

A especificação da UML (OMG, 2007a) possui um elemento denominado *Trace*, com funções de rastreamento, conforme ilustrado na Figura 16.

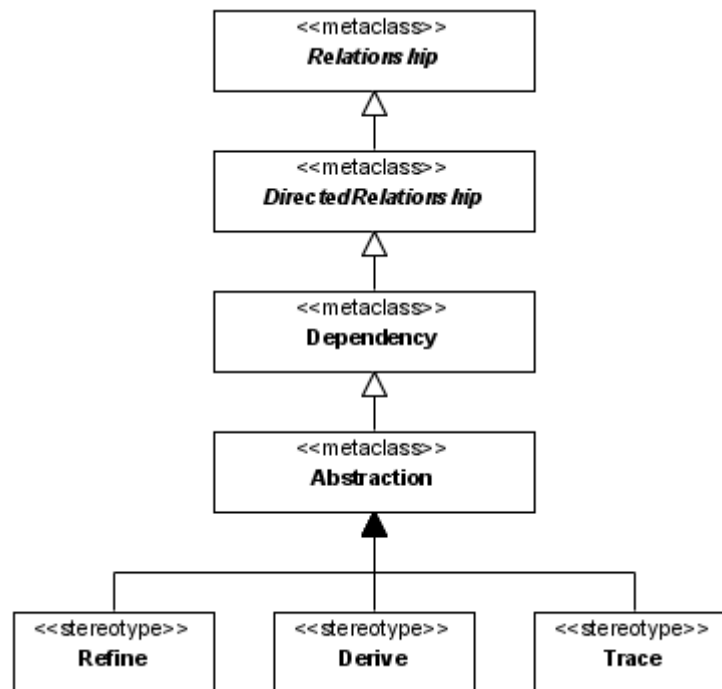


Figura 16 - Relacionamento de rastreamento segundo a UML (OMG, 2007)

Na definição da UML, um *Relationship* (relacionamento) é um elemento que referencia elementos relacionados. E um *DirectedRelationship* (relacionamento direcionado) é um relacionamento que define elementos como origem e como destino do relacionamento. Uma *Dependency* (dependência) é um relacionamento direcionado no qual os elementos que cumprem o papel de origem são chamados de fornecedores e os elementos de destino são

chamados de clientes. Em uma dependência, a especificação ou implementação dos elementos clientes depende dos elementos fornecedores. Por fim, uma *Abstraction* (abstração) é uma dependência na qual os elementos fornecedores e clientes representam um mesmo conceito em diferentes níveis de abstração ou diferentes visões. Uma abstração define ainda um mapeamento entre os elementos relacionados, que define de que maneira os elementos clientes abstraem os elementos fornecedores.

A UML define três estereótipos do elemento abstração que são relevantes para o presente trabalho: *Refine*, *Derive* e *Trace*. O *Refine* (refinamento) define um relacionamento entre elementos no qual os elementos clientes apresentam um refinamento (melhoria ou adaptação) em relação aos elementos fornecedores. Normalmente, os elementos clientes e fornecedores estão em diferentes níveis conceituais, como os níveis de análise e projeto.

A *Derive* (derivação) define um relacionamento no qual os elementos clientes detalham os elementos fornecedores. Neste tipo de relacionamento, há uma forte dependência entre os elementos clientes e os elementos fornecedores.

O *Trace* (rastreamento) indica um relacionamento entre elementos de mesmo conceito mas presentes em modelos diferentes. Normalmente, o rastreamento é utilizado para registrar requisitos e mudanças em modelos distintos. Como mudanças podem ocorrer em qualquer modelo, ou seja, em qualquer lado do rastreamento, a direção da dependência (normalmente do cliente para o fornecedor) frequentemente pode ser ignorada.

A semântica dos elementos *Refine* e *Derive* pode ser aproveitada no presente trabalho sem alterações na sua definição. Para o elemento *Trace*, entretanto, sua definição entra em conflito com a definição de relacionamento de rastreamento proposta anteriormente. Na definição utilizada no presente trabalho não há restrições quanto aos elementos rastreáveis possuírem o mesmo conceito.

Esta pequena mas importante diferença gera a necessidade de se redefinir a semântica do elemento *Trace*. Entretanto, não é permitido alterar a definição de elementos presentes na UML, mas novos elementos podem ser adicionados por meio de extensões à UML. Como os nomes dos novos elementos não podem entrar em conflito com nomes já existentes na UML, um nome como *Trace* não poderia ser utilizado. A solução adotada, mesmo que simplista e informal, é utilizar a tradução de termos. Por exemplo, embora já exista o termo *Trace* na UML, pode-se usar o termo *Rastreamento* para um novo elemento. Letelier (2002) adota

estratégia semelhante, mas usa sinônimos em vez de traduções. Ele redefine o elemento *Trace* usando o nome *TraceTo*.

Como a definição de rastreamento utilizada no presente trabalho não exige que os elementos rastreáveis possuam o mesmo conceito ou possuam uma forte dependência entre si, um relacionamento de rastreamento é considerado uma extensão de uma *Association* (associação).

A Figura 17 ilustra os relacionamentos de rastreamento utilizados no presente trabalho.

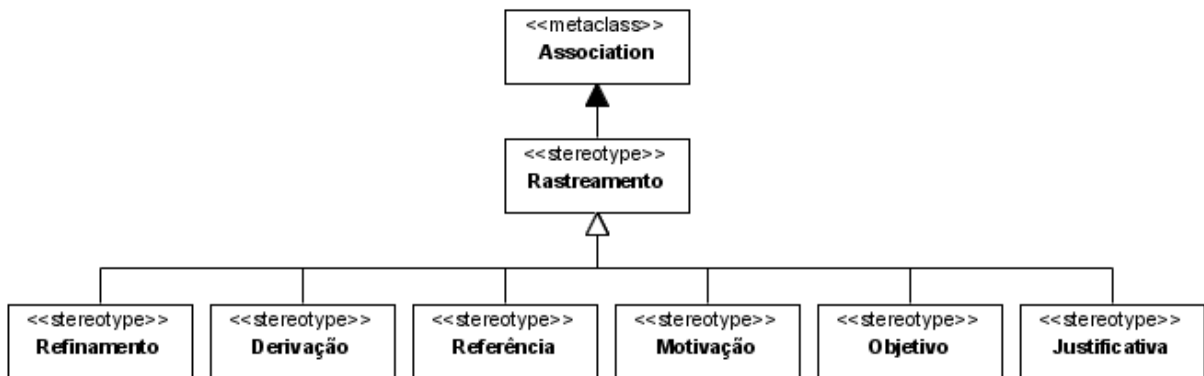


Figura 17 - Definição dos relacionamentos de rastreamento

### Rastreamento

É uma relação entre dois grupos de elementos pela qual um dos grupos, denominado origem, exerce ou exerceu uma certa influência na criação ou existência do outro grupo, denominado destino. Por exemplo, duas versões de um mesmo elemento podem ser relacionadas por um rastreamento.

### Refinamento

De semântica similar ao elemento *Refine*, é um tipo de rastreamento no qual um grupo de elementos (destino) apresenta um refinamento (melhoria ou adaptação) em relação a outro grupo de elementos (origem). Normalmente, os elementos origem e destino estão em diferentes níveis conceituais, como os níveis de análise e projeto.

### Derivação

De semântica similar ao elemento *Derive*, é um tipo de rastreamento no qual um grupo de elementos (destino) detalha outro grupo de elementos (origem). Neste tipo de relacionamento, há uma forte dependência entre os elementos origem e destino.

**Referência**

É um tipo de rastreamento no qual um grupo de elementos (destino) se refere, em sua definição, a outro grupo de elementos (origem). Por exemplo, uma argumentação pode se referir a uma regra de negócio.

**Motivação**

É um tipo de rastreamento no qual um grupo de elementos (origem) motivou a criação de outro grupo de elementos (destino), embora os elementos destino possam existir sem os elementos origem. Este tipo de rastreamento registra a causa ou origem de um grupo de elementos. Por exemplo, uma regra de negócio pode motivar a criação de um requisito de *software*.

**Objetivo**

É um tipo de rastreamento no qual um grupo de elementos (origem) é o objetivo da existência de outro grupo de elementos (destino). Por exemplo, uma necessidade organizacional pode ter como objetivo automatizar um processo da organização.

**Justificativa**

É um tipo de rastreamento no qual um grupo de elementos (origem) representa uma justificativa ou razão para a existência de outro grupo de elementos (destino). Por exemplo, uma argumentação pode representar a justificativa de um requisito de *software*.

**4.3 Elementos do Meta-modelo de Rastreamento**

Ramesh e Jarke (2001) apresentam um meta-modelo de rastreamento de requisitos (Figura 2 do Capítulo 3) que relaciona três elementos: os envolvidos no projeto, as documentações e os objetos rastreáveis. Em seguida, apresentam uma série de modelos de referência relativos apenas ao elemento que representa os objetos rastreáveis. Portanto, não se encontra nenhuma referência aos envolvidos no projeto ou às documentações nos modelos de referência apresentados. O mesmo conceito é utilizado no presente trabalho para definir um meta-modelo de rastreamento de requisitos, significando que os envolvidos no projeto e as documentações não fazem parte do meta-modelo.

Por se tratar de um meta-modelo para rastreamento de requisitos de *software* pré-ER, nenhum elemento que pertença exclusivamente à análise pós-ER é contemplado no meta-modelo.

O primeiro passo para a elaboração do meta-modelo é incluir o elemento que represente os requisitos de *software*. No presente trabalho, é utilizado o termo **requisito** para designar o conjunto de conceitos que representam tudo o que é requerido ou necessário para se solucionar um problema da organização. Note-se que há uma diferença entre dois termos similares: requisito e requisito de *software*. O primeiro representa os desejos e necessidades de uma organização em sanar seus problemas ou efetuar melhorias. O segundo é um tipo de requisito que especifica o que é necessário para se criar um *software* que resolva um problema da organização ou efetue uma melhoria.

O conceito de requisito adotado no presente trabalho está relacionado aos conceitos apresentados por Leffingwell (2003), e presentes também nos trabalhos de Spence e Probasco (2000) e Ramesh e Jarke (2001). De acordo com esses autores, as necessidades dos envolvidos são registradas e representam a base para a elaboração dos requisitos. Características de *software* são elaboradas como forma de solucionar as necessidades levantadas. Por fim, aceitas as características propostas, elas são refinadas para formar o conjunto de requisitos de *software*, funcionais ou não. Este conjunto de informações forma o conceito de requisito adotado no trabalho.

Para Spence e Probasco (2000) e Leffingwell (2003), o rastreamento dos requisitos às características de *software* e às necessidades dos envolvidos representa completamente o rastreamento pré-ER. Outros autores, como Ramesh e Jarke (2001) e Letelier (2002), vão além. Eles consideram importante registrar a base racional por trás das soluções adotadas, de forma que não seja necessário retomar as discussões a respeito de uma solução sempre que houver uma mudança nos fatos que serviram como base para a solução adotada. No presente trabalho, o termo **base racional** é utilizado como o conjunto de questões, alternativas, argumentos e decisões utilizadas na elaboração de um ou mais requisitos.

Bleistein, Cox e Verner (2005) reconhecem a fragilidade de se limitar o rastreamento de requisitos pré-ER apenas às necessidades dos envolvidos, às características de *software* e à base racional. Por isso, propõem um modelo que integre os requisitos de *software* aos elementos que representam o modelo de motivação organizacional. O presente trabalho estende esse conceito para além do aspecto da motivação organizacional, introduzindo o contexto organizacional por completo como a real origem dos requisitos de *software*.

No meta-modelo proposto, o termo **contexto organizacional** cumpre diversos papéis. Em relação à base racional, representa os fatos aos quais ela faz referência. Quando membros da organização desejam melhorar um aspecto da organização, resolver um problema ou se



adequar a um fator externo, como uma nova lei, surge a necessidade de se alterar o contexto organizacional. Portanto, as necessidades dos envolvidos no projeto podem ser interpretadas como a vontade de se alterar um contexto organizacional com problemas, criando um novo (desejado) contexto organizacional.

#### 4.4 Meta-modelo de rastreamento de requisitos

A Figura 18 ilustra o meta-modelo de rastreamento de requisitos proposto neste trabalho.

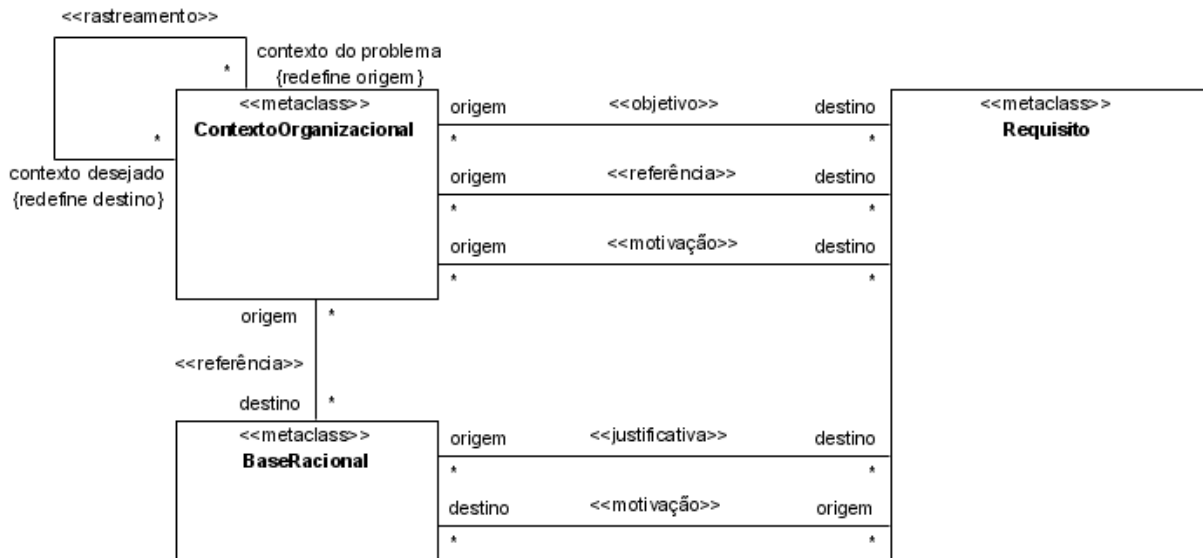


Figura 18 - Proposta de meta-modelo de rastreamento de requisitos pré-ER

Os constructos **ContextoOrganizacional**, **BaseRacional** e **Requisito** são meta-classes, cujos estereótipos são abreviações convencionadas de um espaço de nomes referenciando o domínio de rastreamento de requisitos, como por exemplo, uma abreviação de **Rastreamento de Requisitos::ContextoOrganizacional**. Assim, uma classe que seja uma instância de **ContextoOrganizacional**, por exemplo, a classe *Atividade*, terá o estereótipo «Contexto Organizacional»<sup>2</sup>.

Em um modelo de rastreamento condizente com o meta-modelo proposto, duas versões do contexto organizacional podem ser registradas. A versão inicial ou atual é denominada de **contexto do problema**, pois é nela que se encontram os problemas ou oportunidades de melhoria que motivam a criação de um *software*. Para se resolver o problema, é usual imaginar como o *software* que será criado modificará o contexto organizacional, criando uma segunda versão denominada de **contexto desejado**.

<sup>2</sup> Seria mais rigoroso definir um nome totalmente qualificado para a classe de um modelo gerado do meta-modelo proposto, de tal modo que denotasse sua pertinência a um espaço de nomes específico para cada domínio representado pelas meta-classes. Isso não será adotado no presente trabalho, pois acredita-se que as convenções informalmente introduzidas não devem provocar problemas significativos de interpretação.

A diferença entre os dois contextos representa a necessidade organizacional, que é um dos tipos de requisito. O requisito, em si, representa o meio pelo qual o contexto desejado será atingido.

Em algumas situações, um elemento do contexto do problema é alterado, dando origem a uma nova versão do mesmo elemento no contexto desejado. Nestes casos, é possível manter um relacionamento de rastreamento entre estes elementos, conforme ilustrado no meta-modelo. A manutenção destes rastreamentos é uma importante ferramenta para acompanhar a evolução do contexto organizacional ao longo do tempo e ao longo das diversas criações ou manutenções de *software*.

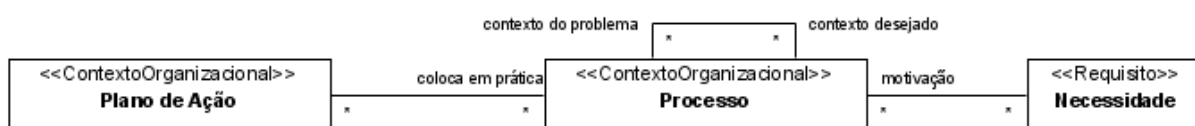
Em outros casos, entretanto, não é possível manter um relacionamento entre elementos do contexto do problema e do contexto desejado. Isso ocorre pois nem sempre um elemento existe em ambos os contextos. Quando uma organização é criada ou estabelece novos procedimentos, não há um contexto anterior com problemas, e sim apenas um novo contexto desejado. Em outros casos, elementos presentes no contexto do problema não mais existirão, e é isso que motiva a alteração de um *software*.

O meta-modelo proposto dispõe de outras maneiras para representar como um requisito é influenciado por alterações no contexto organizacional. Normalmente, problemas ou oportunidades de melhoria notados no contexto do problema motivam a criação de requisitos. É possível, entretanto, que as próprias alterações desejadas para o contexto organizacional motivem também a criação de outros requisitos. Por outro lado, elementos presentes no contexto desejado representam o objetivo por trás de um requisito.

Embora não esteja no escopo do meta-modelo, os trabalhos de Leffingwell (2003), Spence e Probasco (2000) e Ramesh e Jarke (2001) indicam uma série de relacionamentos de rastreamento direto entre os elementos de um requisito. O meta-modelo proposto, entretanto, oferece uma maneira indireta de rastreamento entre os elementos do tipo requisito. Em alguns casos, um requisito de alto nível, como uma necessidade ou característica de *software* pode suscitar, ou motivar, uma discussão sobre como tal requisito será implementado. Após discussões sobre o assunto chega-se a uma conclusão, que pode levar à criação de outros requisitos, tais como características ou requisitos de *software*, normalmente mais detalhados em relação ao requisito que lhes motivou a origem. O registro destas discussões e da decisão tomada representa a base racional que justifica os requisitos.

Por fim, tanto a base racional como os requisitos podem referenciar diversos elementos de um contexto organizacional. A relação entre os elementos da base racional ou dos requisitos com o contexto organizacional é similar à proposta de Tony Morgan (2002) em relação às regras de negócio e o modelo de fatos da organização. Os fatos representam os elementos presentes nas diversas atividades da organização, seus atributos e os relacionamentos entre esses elementos. Segundo Morgan (2002), as regras de negócio devem utilizar a terminologia definida no modelo de fatos, o que remove possíveis ambigüidades. Morgan (2002) cita o seguinte exemplo de regra de negócio: “*Uma conta pode ser fechada somente se seu saldo for zero*”. Neste exemplo, conta é um elemento do modelo de fatos e saldo é um atributo deste elemento.

A Figura 19 ilustra um modelo válido instanciado a partir do meta-modelo.



**Figura 19 - Exemplo de instância válida do meta-modelo**

Observa-se o uso de notações que simplificam a visualização dos relacionamentos no modelo. No exemplo, o relacionamento entre Processo e Necessidade é um rastreamento do tipo motivação, no qual o elemento Processo cumpre o papel de origem. A mesma forma de notação é utilizada para os rastreamentos do tipo justificativa, objetivo e referência. Os rastreamentos de refinamento e derivação utilizam notação similar, mas os termos refinamento e derivação são colocados próximos aos elementos que cumprem o papel de destino. A Figura 19 ilustra ainda o relacionamento de rastreamento entre elementos do contexto organizacional, identificado pelos papéis de contexto do problema e contexto desejado. Estas notações indicam, portanto, todos os relacionamentos de rastreamento explícitos utilizados no presente trabalho. É possível, entretanto, que elementos possuam relacionamentos entre si que não sejam do tipo rastreamento, como é o caso do relacionamento entre Plano de Ação e Processo. Nestes casos, a semântica do relacionamento é dada pela natureza dos elementos, que devem ser do mesmo tipo (não é possível, por exemplo, utilizar um relacionamento que não seja de rastreamento entre um elemento do tipo BaseRacional e outro do tipo Requisito). Qualquer relacionamento deste tipo é considerado, neste trabalho, uma forma de rastreamento implícito.

#### **4.5 Conclusão**

O meta-modelo apresentado relaciona os requisitos, nos quais se encontram os requisitos de *software*, aos conceitos de base racional e contexto organizacional, permitindo um melhor entendimento dos elementos que deram origem a um requisito.

O próximo passo para a criação de um modelo de rastreamento de requisitos é definir o que pode representar cada elemento deste meta-modelo. A definição do modelo do contexto organizacional, por sua importância no presente trabalho, é apresentada nos dois seguintes capítulos. O primeiro apresenta os quadros de referência para arquitetura corporativa como guia para a criação do modelo e o segundo apresenta a criação do modelo em si. Por fim, o capítulo 7 detalha os demais elementos do meta-modelo.

## 5 Quadro de Referência para Arquitetura Corporativa

Uma tarefa essencial para criar um modelo de rastreamento de requisitos de *software* condizente com o meta-modelo proposto é definir os elementos que constituem um contexto organizacional.

Este capítulo apresenta o que é um quadro de referência para arquitetura corporativa (QRAC) e como pode ser útil para guiar o processo de levantamento dos elementos que compõem um modelo do contexto organizacional. Complementam este capítulo a apresentação de três QRACs (TOGAF, QRAC de Zachman e TEAF) e a escolha do QRAC de Zachman como o guia para o modelo de contexto organizacional.

### 5.1 Arquitetura Corporativa da Informação

Alguns trabalhos na área de rastreamento pré-ER, como os publicados por Ramesh e Jarke (2001), Letelier (2002) e Bleistein, Cox e Verner (2005), fazem referência a aspectos que podem ser considerados como parte do contexto organizacional. Entretanto, conforme mencionado anteriormente, a abrangência dos aspectos considerados não é muito grande. De certa forma, apenas são considerados os aspectos relacionados às intenções e motivações organizacionais.

Se os modelos de rastreamento pré-ER existentes não são suficientes para a determinação de um conjunto completo de aspectos corporativos, é preciso utilizar outra fonte para a obtenção destas informações. A Arquitetura Corporativa da Informação pode exercer este papel. Na definição de Leist e Zellner (2006), uma arquitetura corporativa da informação se refere a um sistema de informação, e é responsável pela organização de seus componentes e dos relacionamentos entre eles. Pereira e Sousa (2004) citam algumas vantagens no uso deste tipo de arquitetura:

- Auxilia organizar e estruturar um ambiente corporativo muitas vezes complexo.
- Permite uma visão global dos recursos de informação da corporação.
- Permite detectar redundâncias e caminhos críticos nos processos de negócio.
- Serve como uma ponte entre o domínio de negócio e o domínio de sistemas.

A obtenção de uma arquitetura corporativa da informação é uma tarefa específica de cada organização. Como o intuito deste trabalho é propor um modelo genérico de rastreamento, uma dada arquitetura de informação não poderia ser usada em detrimento de outra. Em seu

lugar, é preciso utilizar um quadro de referência para arquitetura corporativa (QRAC), que possa servir como ferramenta para a elaboração das arquiteturas.

## 5.2 Quadros de Referência para Arquitetura Corporativa (QRAC)

De acordo com a definição do *The Open Group* (2007), um QRAC é:

*“Uma ferramenta que pode ser usada para desenvolver uma ampla gama de arquiteturas. Esta ferramenta deve descrever um método para projetar um sistema de informação com base em uma série de componentes e deve mostrar como os componentes se relacionam. Deve conter uma série de ferramentas auxiliares e prover um vocabulário comum. Deve também incluir uma lista de padrões recomendados e de produtos que podem ser usados na construção dos componentes.”*

Ainda de acordo com o *The Open Group*, são dois os elementos-chave que um QRAC deve prover:

- Definição dos produtos que devem ser gerados na elaboração de uma arquitetura.
- Descrição de um método a ser utilizado na elaboração de uma arquitetura.

Este trabalho apresenta três quadros de referência, com o intuito de escolher um deles para que seja a base para obtenção dos elementos que representem o contexto organizacional.

### 5.2.1 TOGAF – The Open Group Architecture Framework

O TOGAF consiste de um método e um conjunto de ferramentas de suporte para a elaboração de arquiteturas corporativas (*The Open Group*, 2007). A primeira versão do TOGAF foi publicada em 1995 e o trabalho foi baseado no TAFIM (*Technical Architecture Framework for Information Management*), do Departamento de Defesa norte-americano.

O TOGAF possui três partes:

- **TOGAF ADM (*Architecture Development Method*)**: método utilizado para criar descrições de arquitetura.
- ***Enterprise Continuum***: repositório de modelos de referência e padrões, utilizados pela indústria em geral e pelas corporações em particular.
- ***Resource Base***: conjunto de regras e instruções para auxiliar o arquiteto na utilização do TOGAF ADM.

O TOGAF ADM é a parte mais importante do TOGAF, constituindo a maior parte dele. É um método iterativo, aplicado durante todo o processo de desenvolvimento de um sistema e executado em ciclos de 8 passos:

- Visão da arquitetura
- Arquitetura corporativa
- Arquitetura de sistemas de informação
- Arquitetura tecnológica
- Oportunidades e soluções
- Plano de migração
- Governança de implementação
- Gerenciamento de mudança de arquitetura

O TOGAF apresenta regras para bons princípios na criação de arquiteturas corporativas, embora não especifique em detalhes estes princípios. Entretanto, categoriza estes princípios em 3 níveis:

- Suporte à tomada de decisão
- Instruções para gerência de recursos de TI
- Suporte a princípios de arquitetura para desenvolvimento e implementação

### **5.2.2 Quadro de Referência de Zachman**

O QRAC de Zachman foi baseado nos princípios da arquitetura clássica, relacionada à construção civil, estabelecendo um vocabulário comum e um conjunto de perspectivas que podem descrever qualquer sistema de informação (Zachman, 1987).

As perspectivas definidas pelo quadro de referência são (em termos da construção civil e em termos de sistemas de informação):

- Representação Conceitual – Escopo do Sistema
- Desenhos do Arquiteto – Modelo de Negócio
- Planos do Arquiteto – Modelo de Sistema
- Planos do Construtor – Modelo de Tecnologia
- Planos dos Sub-Empreiteiros – Descrição detalhada de Componentes

- Construção – Sistema de Informação

Além das perspectivas, o QRAC de Zachman apresenta uma segunda dimensão, relacionada aos diversos aspectos de uma corporação. Estes aspectos são representados por seis interrogativas básicas: o quê (informação), como (processos), onde (localização), quem (pessoas), quando (tempo) e por quê (motivação) (Sowa, Zachman, 1992).

O QRAC de Zachman não estabelece uma ordem ou mesmo um processo para que cada célula (resultado da combinação das duas dimensões em uma matriz) seja preenchida. O enfoque está em garantir que todas as perspectivas e aspectos estejam presentes, visando a uma arquitetura completa de um sistema.

### 5.2.3 TEAF – *Treasury Enterprise Architecture Framework*

O Departamento do Tesouro norte-americano publicou, em 2000, o TEAF, um QRAC para utilização em seus diversos escritórios (*Department of Treasury*, 2000). O TEAF foi derivado de outros trabalhos, como o TISAF (*Treasury Information Systems Architecture Framework*), o FEAF (*Federal Enterprise Architecture Framework*) e o QRAC de Zachman.

Para o Departamento do Tesouro, seus múltiplos escritórios devem funcionar como empresas independentes e a arquitetura dos sistemas deve representar os inter-relacionamentos entre os escritórios para melhorar a gestão dos recursos de TI.

Assim como o QRAC de Zachman, o TEAF apresenta duas dimensões para a arquitetura: perspectivas e visões.

As perspectivas são 4:

- Planejador
- Dono
- Arquiteto
- Construtor

São 4 também as visões:

- Visão Funcional
- Visão da Informação
- Visão Organizacional
- Visão de Infra-Estrutura



### 5.2.4 Critérios de Escolha para o Quadro de Referência

Os seguintes critérios serão utilizados na escolha do QRAC utilizado neste trabalho:

- Maior número de aspectos corporativos considerados

Um aspecto corporativo é uma abstração da organização com enfoque em determinada característica relacionada a uma interrogativa básica – o quê faz uma organização, quando, como e por quê ela faz isso, onde as coisas acontecem e quem as faz acontecer.

O QRAC escolhido será utilizado como única base para a obtenção dos elementos que representarão o contexto organizacional no modelo de rastreamento proposto. Isso significa que elementos não considerados no QRAC também não estarão presentes no modelo de rastreamento. Portanto, basear-se em um QRAC que desconsidere alguns aspectos corporativos pode resultar em um modelo incompleto, que não contenha importantes elementos para o rastreamento de requisitos pré-ER.

- Menor especificidade de aplicação

Se um QRAC é específico para determinada indústria (tipo de corporação), ele pode ter um enfoque em aspectos corporativos mais relevantes para esta indústria, em detrimento a outros aspectos corporativos. Para a construção de um modelo de rastreamento, o único critério válido para classificar um aspecto corporativo em termos de relevância é o seu papel no rastreamento, e não a sua aplicabilidade a uma dada indústria ou modelo de negócio.

- Enfoque nos produtos em comparação com os métodos utilizados para obtê-los

O objetivo do trabalho é criar um modelo de rastreamento, e não um método completo para criação de modelos deste tipo. Portanto, há apenas o interesse nos tipos de elementos corporativos que representariam o contexto organizacional, e não em métodos e técnicas utilizados para determinar estes elementos em uma organização.

Urbaczewski e Mrdalj (2006) e Leist e Zellner (2006) realizaram comparações entre diversos QRACs. A Tabela 1 sintetiza a comparação em termos dos aspectos corporativos considerados, utilizando como base os aspectos considerados no QRAC de Zachman.

**Tabela 1 - Comparação de QRACs em termos de aspectos corporativos (Urbaczewski, Mrdalj, 2006)**

QRAC	O Quê?	Como?	Onde?	Quem?	Quando?	Por quê?
Zachman	Dados	Funções	Relacionamentos	Pessoas	Tempo	Motivação
TEAF	Visão da Informação	Visão Funcional	Visão de Infra-estrutura	Visão Organizacional		
TOGAF		Instruções para tomada de decisões		Instruções para gerenciamento de recursos de TI		

Leist e Zellner (2006) comparam os QRACs de acordo com a presença de 5 diferentes elementos que podem constituir um QRAC:

- **Meta-modelo:** indica que tipos de elementos podem estar presentes em uma arquitetura e como estes elementos podem ser relacionados.
- **Modelo de procedimento:** método pelo qual a arquitetura é especificada, incluindo ordem dos passos, regras e procedimentos para determinação da arquitetura.
- **Técnicas:** técnicas específicas para determinadas partes da arquitetura.
- **Papéis:** indicam os responsáveis por especificar determinadas partes da arquitetura.
- **Documentos de especificação:** sugestões de documentos que contêm as especificações de cada parte da arquitetura.

Leist e Zellner (2006) apresentam uma tabela (Tabela 2) com suas conclusões sobre a presença destes 5 elementos nos QRACs apresentados nas seções anteriores.

**Tabela 2 - Comparação dos QRACs em termos de elementos constituintes (Leist, Zellner, 2006)**

	<b>Zachman</b>	<b>TEAF</b>	<b>TOGAF</b>
<b>Documentos de especificação</b>	Contemplado	Contemplado	Não Contemplado
<b>Meta-modelo</b>	Parcialmente Contemplado	Parcialmente Contemplado	Não Contemplado
<b>Papéis</b>	Não Contemplado	Contemplado	Não Contemplado
<b>Técnicas</b>	Não Contemplado	Não Contemplado	Parcialmente Contemplado
<b>Modelo de procedimento</b>	Parcialmente Contemplado	Contemplado	Contemplado

Em relação ao critério de número de aspectos considerados, apenas o QRAC de Zachman considera os aspectos de tempo e motivação. O aspecto da motivação organizacional, em particular, é apontado como de grande importância para o rastreamento, conforme os trabalhos de Ramesh e Jarke (2001), Letelier (2002) e Bleistein, Cox e Verner (2005).

Em termos da especificidade de aplicação do QRAC, os QRACs de Zachman e o TOGAF são bastante genéricos, enquanto o TEAF tem forte enfoque no departamento de tesouro americano.

Quanto ao enfoque dado pelo QRAC, tanto o de Zachman quanto o TEAF dão importância aos produtos de arquitetura gerados. O TOGAF, por sua vez, tem seu ponto forte no método para obtenção dos produtos, recomendando até mesmo o uso de outros QRACs para suprir a ausência de produtos mais bem definidos.

A Tabela 3 resume a compatibilidade dos QRACs em relação aos critérios adotados.

**Tabela 3 - Comparação dos QRACs em termos dos critérios adotados**

<b>Crítérios</b>	<b>Zachman</b>	<b>TEAF</b>	<b>TOGAF</b>
<b>Número de aspectos corporativos</b>	Forte – todos os seis aspectos são considerados	Média – os aspectos Quando? e Por que? não são considerados	Fraca – apenas os aspectos Como? E Quem? são considerados
<b>Especificidade de aplicação</b>	Alto	Baixo – enfoque no departamento de tesouro americano	Alto
<b>Produtos x métodos</b>	Forte em produtos Fraco em métodos	Forte em produtos Médio em métodos	Fraco em produtos Forte em métodos

Portanto, de acordo com os critérios adotados, escolhe-se o QRAC de Zachman.

### **5.3 Quadro de Referência de Zachman**

Em 1987, John Zachman publicou um artigo no qual definia um quadro de referência para arquiteturas de sistemas de informação (Zachman, 1987). Em 1992, em parceria com John Sowa, publicou uma extensão deste trabalho, formalizando o quadro de referência apresentado anteriormente (Sowa, Zachman, 1992).

De acordo com Zachman, o aumento no escopo e na complexidade dos sistemas de informação exigia uma estrutura lógica para definir e controlar as interfaces e a integração de todos os componentes de um sistema. Além disso, considerava que a então crescente descentralização dos componentes dos sistemas de informação levaria ao caos se não houvesse alguma ordem (arquitetura).

Na criação do seu quadro de referência, Zachman optou por uma visão independente dos sistemas de informação, para que seu trabalho não pudesse ser maculado por práticas já existentes. Para garantir a abrangência de seu quadro de referência, Zachman baseou seu trabalho na arquitetura mais convencional, a arquitetura da construção civil.

Durante o projeto de uma construção civil, diversos planos ou representações são elaborados, de acordo com a fase do projeto.

#### **a) Representação Conceitual**

O dono (contratante) do projeto expressa suas idéias, que servem de base para o arquiteto. O esboço criado pelo arquiteto garante ao dono do projeto que suas idéias foram compreendidas.

#### **b) Desenhos do Arquiteto**

Os desenhos do arquiteto representam uma visão mais detalhada do projeto, com enfoque no que é interessante para o dono do projeto.

Estes desenhos servem como uma forma de comunicação entre o dono do projeto e o arquiteto, e como representam os interesses do primeiro, são normalmente representados em uma linguagem que o dono do projeto entenda com facilidade.

### **c) Planos do Arquiteto**

Os planos do arquiteto representam uma visão mais elaborada do projeto, contendo informações mais específicas dele, como, por exemplo, os materiais e acabamentos desejados.

Estes planos servem como forma de comunicação entre o arquiteto e o construtor.

### **d) Planos do Construtor**

O construtor revisa os planos do arquiteto, alterando-os com enfoque em como o projeto será construído. Os planos sofrem adaptações de acordo com limitações de tecnologia e ferramentas.

### **e) Planos dos Sub-Empreiteiros**

Os planos dos sub-empreiteiros, diferentemente dos outros planos ou representações, não se referem ao projeto por inteiro, e sim a partes dele. A especificação destas partes, fora do contexto geral do projeto, pode representar padrões de construção, que podem ser utilizados em outras partes do projeto ou até mesmo em outros projetos.

### **f) Construção**

É o produto final do projeto, a construção em si. Os planos e representações apresentados podem ser divididos em três categorias, uma para cada envolvido no projeto. Estas categorias são a representação arquitetônica do dono do projeto, do arquiteto (projetista) e do construtor. Cada representação arquitetônica é denominada de **perspectiva**.

Cada perspectiva não é simplesmente uma extensão da perspectiva anterior, com o acréscimo de maiores detalhes. A natureza, o conteúdo e a semântica das perspectivas são diferentes.

Como o interesse de Zachman era criar um quadro de referência para arquiteturas de sistemas de informação, e não de arquiteturas de construção civil, ele apresentou um mapeamento dos conceitos de perspectivas para a realidade dos sistemas de informação (Tabela 4).

**Tabela 4 - Mapeamento das perspectivas da construção civil para sistemas de informação**

<b>Termos Genéricos</b>	<b>Construção Civil</b>	<b>Sistemas de Informação</b>
Estimativa	Representação conceitual – esboço da construção	Escopo do sistema
Perspectiva do dono do projeto	Desenhos do arquiteto	Modelo do negócio
Perspectiva do Projetista	Planos do arquiteto	Modelo do sistema
Perspectiva do Construtor	Planos do Construtor	Modelo de tecnologia
Representação fora do contexto	Planos dos sub-empregados	Detalhamento de componentes
Produto	Construção	Sistema de informação

Além das diversas perspectivas, Zachman menciona que há diferentes tipos de descrição que se pode fazer de um projeto. Cada tipo de descrição é denominado de **abstração**.

As abstrações são representações de um projeto com enfoque em uma de suas características. Zachman relacionou as abstrações a seis interrogativas básicas:

**a) O quê? (aspecto de dados)**

O enfoque da abstração está nos materiais, informações e entidades que compõem um sistema, e o modelo básico que pode representá-la é o modelo de entidade-relacionamento.

**b) Como? (aspecto dos processos)**

O enfoque da abstração está nos processos e funções que fazem parte de um sistema, e o modelo básico que pode representá-la é o modelo de função-argumento.

**c) Onde? (aspecto de localização)**

O enfoque da abstração está na localização dos componentes de um sistema e o modelo básico que pode representá-la é o modelo de nó-ligação.

**d) Quem? (aspecto de estruturação)**

O enfoque da abstração está nos participantes de um sistema e o modelo básico que pode representá-la é o modelo de papel-funcionalidade.

**e) Quando? (aspecto de tempo)**

O enfoque da abstração está nos momentos e eventos relevantes para um sistema, e o modelo básico que pode representá-la é o modelo de evento-ciclo.

**f) Por quê? (aspecto de motivação)**

O enfoque da abstração está nas razões que motivaram a criação de um sistema e o modelo básico que pode representá-la é o modelo de meio-fim.

Em seu trabalho, Zachman organiza os conceitos de perspectivas e abstrações em uma matriz, contendo seis linhas (perspectivas) e seis colunas (abstrações). Para cada célula das três primeiras linhas desta matriz, Zachman sugere um meta-modelo que possa representá-la. O conteúdo de cada uma das células não está, todavia, no escopo do presente trabalho.

Por fim, Zachman menciona algumas regras que governam o uso do seu quadro de referência.

### **Regra 1. As colunas não possuem ordem**

A existência de uma ordem entre as colunas implica a existência de prioridade entre elas, e nenhum aspecto das corporações é sempre mais importante do que outro.

### **Regra 2. Cada coluna possui um modelo básico simples**

Há um único meta-modelo genérico para cada coluna, definindo que tipos de entidades e seus relacionamentos são importantes em uma dada abstração.

### **Regra 3. O modelo básico de cada coluna deve ser único**

Cada abstração tem o enfoque em um dado aspecto da corporação. Portanto, as entidades e seus relacionamentos devem ser diferentes em cada coluna, para evitar repetição de conceitos.

### **Regra 4. Cada linha representa uma perspectiva única**

A diferença entre as perspectivas é bastante clara quando se examina as linhas 2, 3 e 4, fortemente relacionadas às visões do dono do projeto, do projetista e do construtor, respectivamente. O primeiro se preocupa com a usabilidade do projeto, o segundo, com o desenho de uma solução e o terceiro, em como implementar esta solução.

### **Regra 5. Cada célula é única**

Se há um meta-modelo diferente para cada coluna, e se as diferentes perspectivas mudam o significado destes meta-modelos em cada linha, é natural que cada célula possua componentes distintos das demais.

### **Regra 6. A composição dos modelos de todas as células de uma linha constitui um modelo completo para a perspectiva daquela linha**

Uma célula em uma coluna representa uma única abstração da perspectiva representada pela linha desta célula. A soma de todas as abstrações de uma linha resulta na mais completa descrição da perspectiva desta linha.

#### 5.4 Rastreamento de Requisitos no Quadro de Referência de Zachman

Embora o QRAC de Zachman tenha sido escolhido devido à sua abrangência, nem todas as suas 36 células serão consideradas neste trabalho. Somente serão utilizadas as células que possuem elementos considerados como partes do contexto organizacional.

Os elementos do contexto organizacional são aspectos da organização que se quer aprimorar. Estes aspectos fazem parte da perspectiva do dono do negócio, pois é ele quem enxerga quais são as melhorias que se pretende fazer. Estes aspectos e a necessidade de aprimorá-los fazem parte do domínio do problema. No QRAC de Zachman, esta perspectiva é representada pela linha 2, correspondente aos modelos de negócio da organização.

As características de *software* e os requisitos de software fazem parte do domínio da solução. Este domínio se estende da linha 3 à linha 6 do QRAC de Zachman. No presente trabalho, assume-se que a maior parte dos requisitos está na linha 3, que representa os modelos de sistema, uma vez que o sistema é projetado com base nos requisitos.

A linha 1, que representa o escopo do sistema, apesar de poder tomar parte no rastreamento pré-ER, possui informações pouco detalhadas e em nível muito elevado. Por não ser possível extrair desta linha elementos que fariam parte do modelo proposto neste trabalho, ela não será considerada.

As linhas 4, 5 e 6, representando, respectivamente, os modelos de tecnologia, as descrições detalhadas de partes do sistema e o sistema em si, trazem elementos gerados após a especificação de requisitos (e como resultado dela). Portanto, são relevantes apenas para o rastreamento de requisitos pós-ER.

Considerando que contexto organizacional se encontra na linha 2 do QRAC de Zachman, é preciso definir também quais colunas representam este contexto. Como o QRAC de Zachman foi escolhido pela sua abrangência, a princípio todas as colunas serão consideradas como possíveis fontes para a determinação dos elementos que representam o contexto organizacional.

Para este trabalho, é preciso definir, para cada coluna, qual modelo é o mais adequado, do ponto de vista do rastreamento de requisitos, para fornecer os elementos que definirão o contexto organizacional no modelo proposto.

## **5.5 Conclusão**

Como forma de definir o contexto organizacional, o capítulo definiu o que são os QRACs e como podem ser úteis ao trabalho. O capítulo apresentou três QRACs encontrados na literatura e justificou a escolha do QRAC de Zachman sob o ponto de vista do rastreamento de requisitos proposto no presente trabalho. Mostrou ainda que apenas parte do QRAC de Zachman, a linha que representa os modelos de negócio, é importante para o trabalho.

Por fim, concluiu que o primeiro passo para a criação do modelo proposto é definir que tipo de modelo de contexto organizacional é o mais adequado, em termos de rastreamento de requisitos, para cada coluna do QRAC de Zachman.



## 6 Modelo do Contexto Organizacional

O conjunto de modelos dos seis aspectos da linha 2 do QRAC de Zachman forma um modelo completo do contexto organizacional. Neste trabalho, entretanto, uma abordagem prática é adotada, considerando o aspecto dos processos como o mais relevante para a especificação de um *software*. Outros dois aspectos são considerados, devido à sua forte relação com o aspecto dos processos: estruturação e motivação organizacional (Chapin et al, 2005; Morgan, 2002).

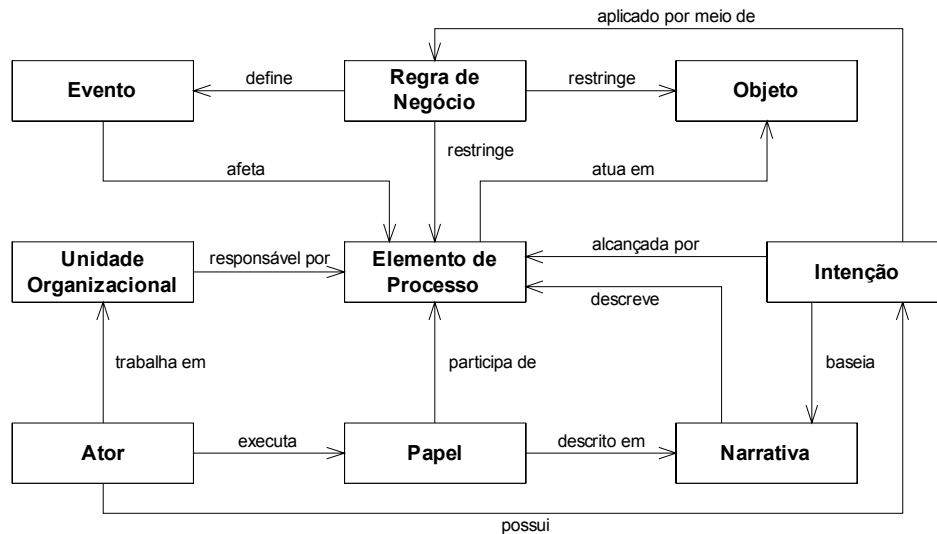
O capítulo apresenta um meta-modelo para definição de processos de negócio chamado BPDM (*Business Process Definition Metamodel*). Dada a sua importância no trabalho, ele é descrito com detalhes. O método IDEF3 é utilizado como um método-guia para a elaboração de um modelo do aspecto dos processos a partir do meta-modelo BPDM. Para o aspecto de motivação é utilizado um modelo de motivação organizacional conhecido como BMM (*Business Motivation Model*) e, para o aspecto de estruturação, um meta-modelo de estrutura organizacional chamado OSM (*Organization Structure Metamodel*). Por fim, o capítulo apresenta o modelo de contexto organizacional resultante da junção dos modelos dos três aspectos considerados.

### 6.1 Hipóteses Adotadas

A respeito do aspecto dos processos, Morgan (2002) afirma:

*“Se você entrar em uma organização e pedir a uma pessoa que descreva como a organização funciona, essa pessoa muito provavelmente irá lhe descrever um processo de negócio.”*

Esta afirmação denota a importância dos processos de negócio na representação do contexto organizacional. Morgan (2002) ainda apresenta um exemplo de arquitetura corporativa (Figura 20) no qual os processos de negócio têm papel central. Embora se trate de um exemplo entre tantos outros possíveis, é relevante notar que o trabalho de Morgan (2002) tem o enfoque nas regras de negócio, e ainda assim o **Elemento de Processo** é a entidade com o maior número de relacionamentos com outras entidades. Apenas a entidade **Ator** não possui ligação direta com o **Elemento de Processo**, embora tal relacionamento possa ser facilmente inferido a partir da entidade **Papel**.



**Figura 20 - Exemplo de arquitetura corporativa (Morgan, 2002)**

Sowa e Zachman (1992) também destacam a importância dos processos de negócio em seu exemplo de meta-modelo para os aspectos de dados, processos e localização (Figura 21). Embora não escrevam explicitamente sobre a relevância do aspecto dos processos, pois isso iria contra os conceitos de seu trabalho, ela pode ser percebida na disposição dos aspectos e pela quantidade de relacionamentos entre as entidades dos três aspectos. Observa-se que não está no escopo deste trabalho uma descrição detalhada do exemplo de Sowa e Zachman.

O aspecto dos processos é apresentado ao centro, facilitando a representação gráfica do meta-modelo. O aspecto dos processos participa do maior número de relacionamentos inter-aspectos. Com o aspecto de dados, são quatro: Evento – Regra de Negócio, Evento – Entidade de Negócio, Processo de Negócio – Entidade de Negócio e Recurso de Negócio – Entidade de Negócio. Com o aspecto de localização, mais quatro relacionamentos: Evento – Unidade de Trabalho, Função de Negócio – Local de Negócio, Recurso de Negócio – Local de Negócio e Recurso de Negócio – Troca de Negócio. Por outro lado, há apenas um relacionamento entre os aspectos de dados e localização: Entidade de Negócio – Local de Negócio.

Com base nestes conceitos, neste trabalho utiliza-se a hipótese de que o aspecto dos processos é o aspecto mais relevante para a modelagem do contexto organizacional. A existência de diversos métodos e notações para a modelagem de processos de negócio constitui mais um fator de apoio à hipótese adotada. List e Korherr (2006) citam sete linguagens para modelagem de processos de negócio (BPML – *Business Process Modelling Languages*): Diagramas de atividade da UML, BPD (Business Process Definition Metamodel), BPMN (*Business Process Modelling Notation*), EPC (*Event Driven Process Chain*), IDEF3 (*Integrated DEFINITION Method 3*), Redes de Petri e RAD (*Role Activity Diagram*).

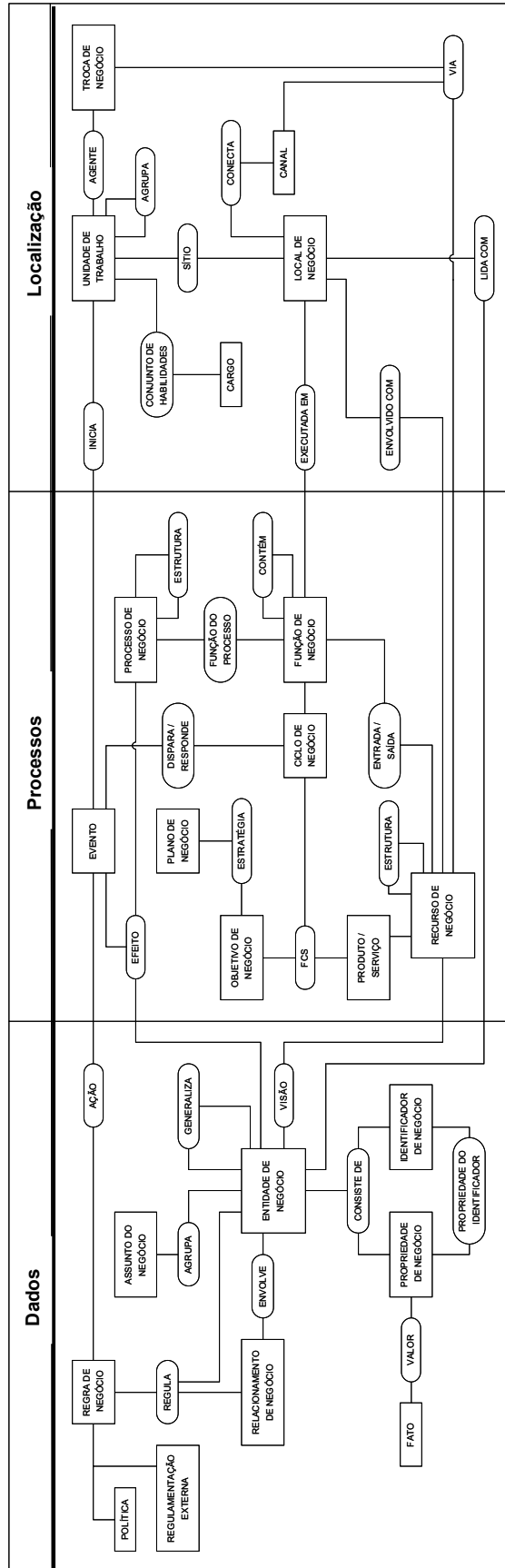


Figura 21 - Exemplo de meta-modelo para três aspectos do QRAC de Zachman (Sowa, Zachman, 1992)

Como resultado da adoção desta hipótese, o primeiro passo para a construção do modelo de contexto organizacional é modelar o aspecto dos processos. Entre as sete BPML apresentadas por List e Korherr (2006), apenas o BPDM, o EPC e os diagramas de atividade apresentam um meta-modelo de processos, que poderia ser usado como base do modelo do aspecto de processos. O BPDM e o EPC pertencem ao domínio do processo em si, enquanto os diagramas de atividade pertencem ao domínio do *software*. Como o objetivo desta escolha é determinar um modelo que represente o contexto organizacional, dá-se preferência às BPML que pertençam ao domínio do processo. Neste trabalho, será utilizado o BPDM em detrimento do EPC, pois o primeiro tem claro enfoque no aspecto dos processos, enquanto o segundo baseia os processos em eventos, que seriam melhor definidos no aspecto de tempo.

Como o BPDM é um meta-modelo e o objetivo deste capítulo é definir um modelo para o contexto organizacional, o método IDEF3 será utilizado como um método-guia para a instanciação de um modelo de processos a partir do BPDM. O IDEF3 foi escolhido por ser um método e notação bem estabelecidos (List, Korherr, 2006), o que facilitaria o uso do modelo proposto neste trabalho.

Em seu trabalho, Chapin et al (2005) posicionam o BMM (*Business Motivation Model*) na coluna do aspecto de motivação do QRAC de Zachman e denotam o relacionamento deste aspecto com os aspectos de estruturação e de processos. A Figura 22 ilustra o relacionamento entre os aspectos de estruturação e motivação e a Figura 23 ilustra o relacionamento entre os aspectos dos processos e de motivação.

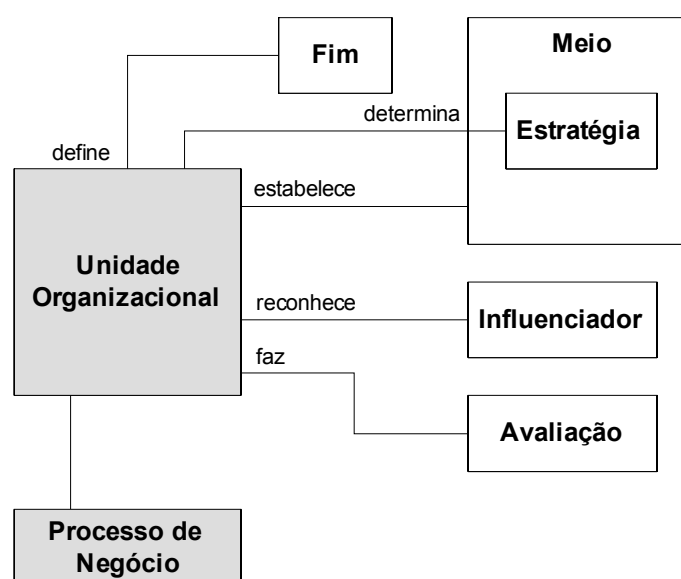
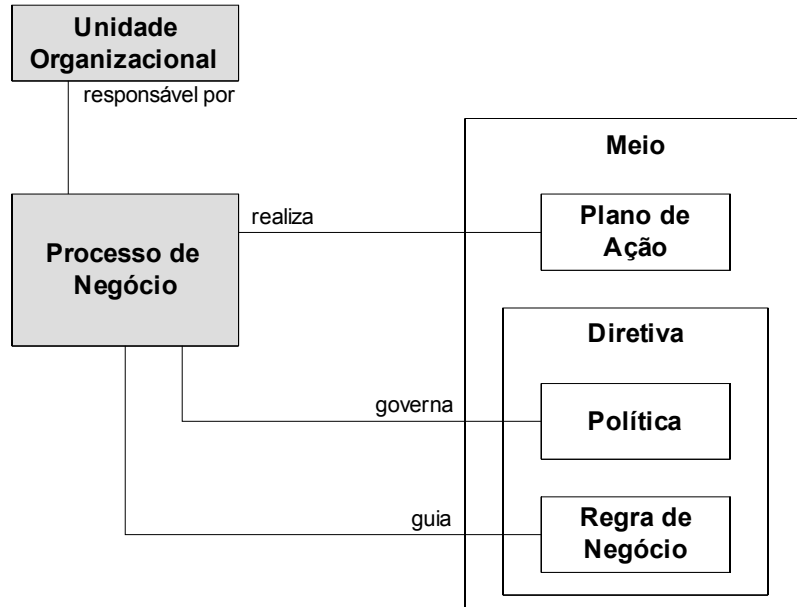


Figura 22 - Relacionamento entre os aspectos de estruturação e motivação (Chapin et al, 2005)



**Figura 23 - Relacionamento entre os aspectos dos processos e de motivação (Chapin et al, 2005)**

Os elementos **Unidade Organizacional** e **Processo de Negócio** representam modelos que não estão no escopo do BMM. Entretanto, Chapin et al (2005) afirmam que esses elementos são apenas uma representação do papel de outros trabalhos, especificamente do OSM para a **Unidade Organizacional** e do BPDM para o **Processo de Negócio**.

Neste trabalho, a ligação entre BPDM, OSM e BMM será mantida, o que garante ao contexto organizacional a abrangência de três aspectos: processos, estruturação e motivação. Além da ligação com o BPDM, o OSM fornece os elementos que representam os envolvidos no projeto, fator importante ao rastreamento conforme o meta-modelo de Ramesh e Jarke (2001), apresentado na Figura 2 do capítulo 2. O BMM, por sua vez, representa os elementos de motivação organizacional presentes nos trabalhos de diversos autores a respeito do rastreamento de requisitos (Ramesh e Jarke, 2001; Bleistein, Cox e Verner, 2005).

Embora o OSM seja um meta-modelo, não se utilizará neste trabalho um método específico para a instanciação de um modelo de estrutura organizacional, como realizado com o BPDM. Para este caso, serão utilizados apenas os elementos com maior afinidade com o BPDM e BMM.

A fim de delimitar o escopo do presente trabalho, os demais aspectos – dados, tempo e localização – não farão parte do modelo de contexto organizacional proposto.

## 6.2 BPDM

O BPDM (*Business Process Definition Metamodel*) é um meta-modelo para a definição de processos de negócio, provendo recursos para a criação de modelos de processo independentemente de notação ou metodologia (Rivet et al, 2007). O BPDM considera duas importantes visões sobre processos:

- Orquestração: possui enfoque no sequenciamento de atividades, incluindo ramificações e sincronização. Descreve o quê acontece em um processo supervisionado por uma entidade.
- Coreografia: possui enfoque nas interações entre diversas funções relacionadas a um processo. Descreve como entidades distintas se relacionam na execução de um processo.

Por se tratar de um meta-modelo extenso, o BPDM define 8 pacotes de modelos agrupados em 4 categorias:

### 1) Abstrações comuns

Possui elementos não orientados a processo que são comuns aos demais modelos. Contém 2 pacotes:

- a) Modelo de composição
- b) Modelo de cursos

### 2) Modelo comum de comportamento

Possui elementos orientados a dinâmica de um processo, e que são comuns aos demais modelos. Contém 3 pacotes:

- a) Modelo de acontecimento e mudança
- b) Modelo de comportamento de processo
- c) Modelo de interação simples

### 3) Modelo de atividades

Possui elementos exclusivos da visão de orquestração. Contém 2 pacotes:

- a) Modelo de atividades
- b) Extensões para BPMN (*Business Process Modeling Notation*)

#### 4) Modelo de protocolo de interação

Possui elementos exclusivos da visão de coreografia. Contém apenas um pacote, de mesmo nome da categoria.

#### 6.2.1 Modelo de Composição

Permite conectar elementos devido às suas relações com um outro elemento em comum. Por exemplo, atividades e seus sequenciamentos estão conectados pelo processo do qual fazem parte.

A Figura 24 ilustra o núcleo do modelo de composição, contendo os principais elementos deste modelo (composição, parte e conexão).

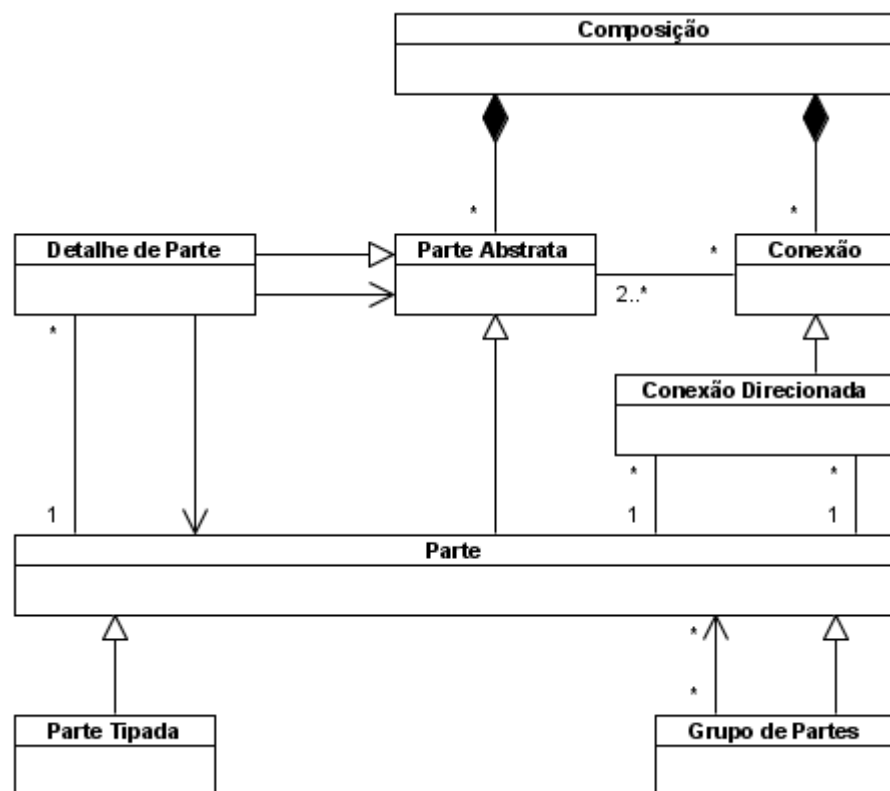


Figura 24 - Núcleo do modelo de composição no BPDM (Rivet et al, 2007)

#### Composição

Representa um conjunto de partes conectadas entre si. Cada parte e cada conexão possui uma relação com a composição.

#### Parte Abstrata

Representação generalizada de qualquer parte de uma composição. Permite que partes sejam conectadas com outras partes ou com partes de outras partes.

**Conexão**

Conexão genérica entre um número qualquer de partes de uma composição.

**Detalhe de Parte**

Elemento abstrato que relaciona uma parte a um pedaço desta parte. Por exemplo, se uma atividade é considerada uma parte de uma composição, o início ou fim desta atividade é um pedaço desta parte. Portanto, uma conexão pode ter em uma de suas pontes um pedaço de uma parte. Isso permite, por exemplo, que um sequenciamento de atividades especifique que uma atividade se inicia após o início ou fim de uma outra atividade.

**Conexão Direcionada**

Conexão entre apenas duas partes, na qual se define uma parte como a origem e a outra parte como o destino da conexão.

**Parte**

Um elemento qualquer de uma composição, que se relaciona a outras partes (ou detalhes de parte) por meio de conexões.

**Parte Tipada**

É um tipo de parte que especifica que indivíduos que fazem o papel desta parte devem ser de um tipo específico. Por exemplo, se uma parte em um modelo representa o presidente de uma empresa, um indivíduo, para poder cumprir este papel, deve ser do tipo pessoa.

**Grupo de Partes**

Coleção de outras partes

A Figura 25 ilustra o papel das condições no modelo de composição.

**Condição**

Determina se um dado elemento a qual a condição está atrelada deve ocorrer ou não. Durante a execução de um processo, as condições podem ser avaliadas como verdadeiras ou falsas. No BPDM, as condições estão atreladas às conexões. Se o sequenciamento entre duas atividades possuir uma condição qualquer, a segunda atividade só ocorrerá quando tal condição for verdadeira.

**Condição Opaca**

Condição simples cuja lógica pode ser expressa em linguagem natural.



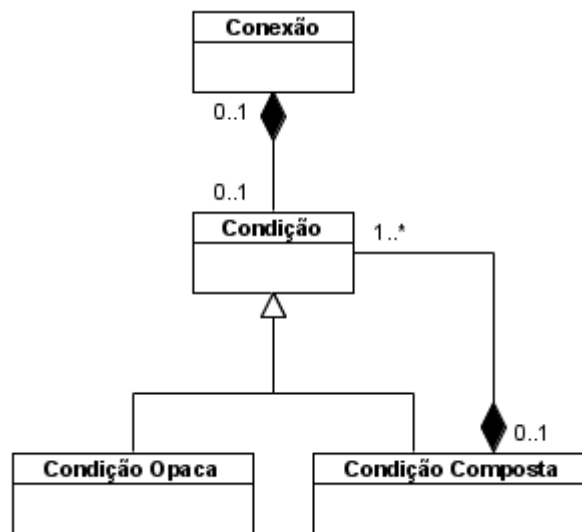


Figura 25 - Modelo de condições no BPDM (Rivet et al, 2007)

### Condição Composta

Condição composta por outras condições, ligadas entre si por operadores lógicos como E, OU e NÃO.

### 6.2.2 Modelo de Cursos

Extende o modelo de composição de forma a seqüenciar partes no tempo, por meio de sucessões.

A Figura 26 ilustra os elementos do modelo de cursos. As classes representadas em fundo cinza e sem a divisão entre nome e atributos e métodos já foram definidas previamente.

#### Curso

Representa uma composição na qual as partes se conectam por sucessões no tempo.

#### Parte de Curso

Um tipo de parte utilizada em cursos e conectada por sucessões. Possui dois tipos importantes: as partes de curso tipadas e os controles de curso.

#### Parte de Curso Tipada

É um tipo de parte de curso que representa uma etapa do curso. Em uma execução real de um curso, devem haver elementos de um certo tipo cumprindo o papel das partes de curso tipada.

#### Controle de Curso

Elementos abstratos que permitem a ramificação de cursos em diversos caminhos, introduzindo lógicas que indicam como as partes de curso tipadas são ordenadas no tempo.

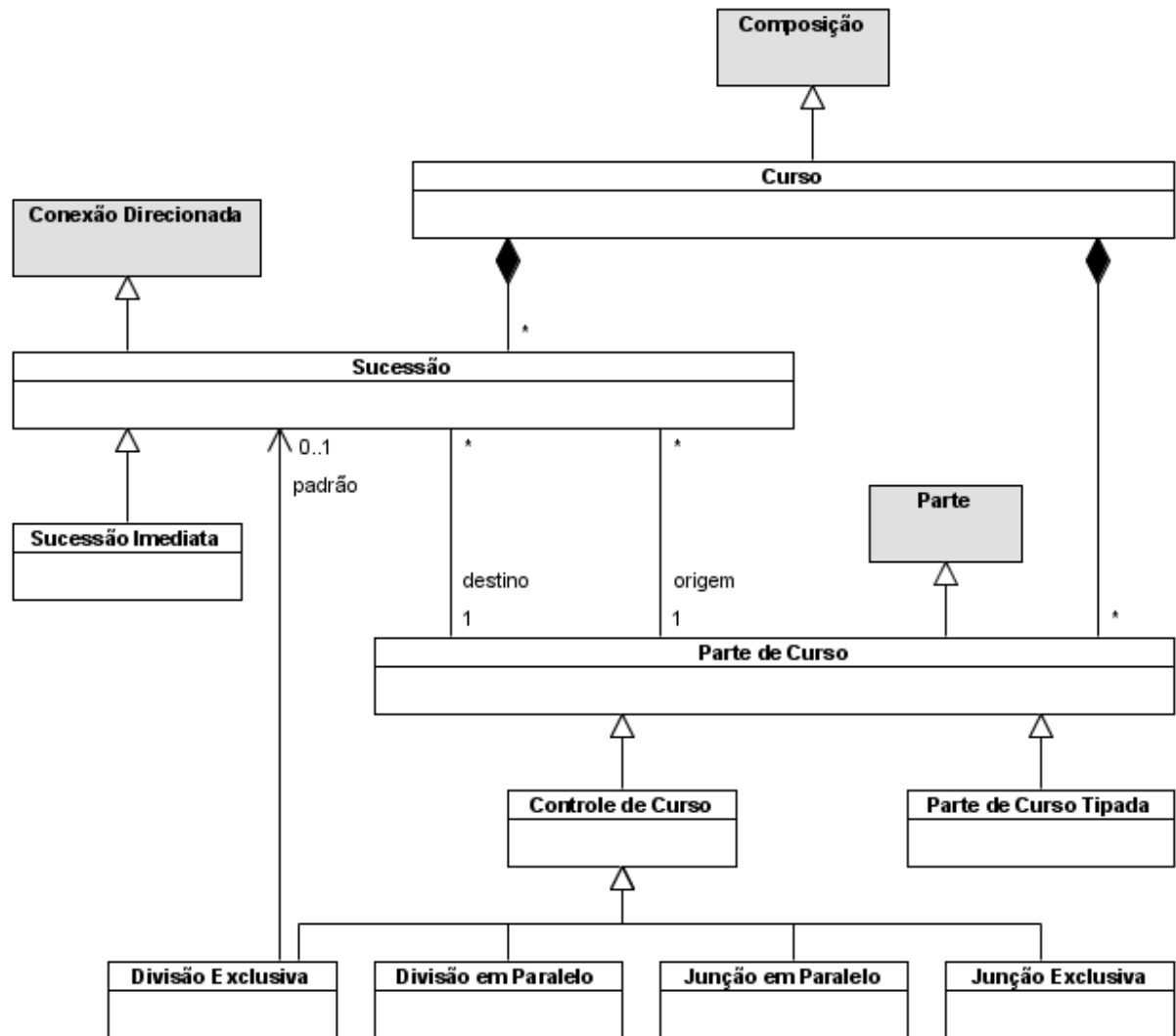


Figura 26 - Modelo de cursos do BPDM (Rivet et al, 2007)

### Divisão em Paralelo

Controle de curso que ramifica um caminho de entrada em diversos caminhos de saída. Em tempo de execução do curso, todos os caminhos de saída são percorridos.

### Junção em Paralelo

Controle de curso que agrupa uma série de caminhos em um só. O caminho de saída do controle só é percorrido após todos os caminhos de entrada terem sido executados.

### Divisão Exclusiva

Controle de curso que ramifica um caminho de entrada em diversos caminhos de saída. Em tempo de execução do curso, apenas um dos caminhos de saída é percorrido. Pode-se estabelecer um caminho padrão, que será adotado caso a lógica do controle de curso não consiga definir o caminho adequado.

### **Junção Exclusiva**

Controle de curso que agrupa uma série de caminhos em um só. O caminho de saída do controle é percorrido após a execução de pelo menos um caminho de entrada.

### **Sucessão**

Tipo de conexão direcionada utilizada em cursos para indicar que uma parte (destino) sucede a outra (origem) no tempo.

### **Sucessão Imediata**

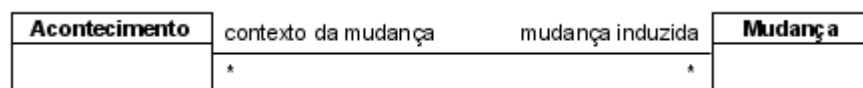
Tipo de sucessão na qual a parte de destino se inicia ao mesmo tempo da parte de origem.

## **6.2.3 Modelo de Acontecimento e Mudança**

Extende os modelos de curso e composição, introduzindo o conceito de dinâmica. Um acontecimento é uma entidade dinâmica que possui um certa duração no tempo. Uma mudança é também uma entidade dinâmica, mas que possui uma duração atômica.

Um acontecimento induz ou produz uma mudança. Uma mudança, por sua vez, ocorre no contexto de um acontecimento. Por exemplo, uma prova de atletismo, como uma corrida com revezamento, pode ser modelada como um acontecimento. A largada, os revezamentos e a chegada são mudanças que ocorrem no decorrer deste acontecimento.

A Figura 27 ilustra os conceitos básicos do modelo de acontecimento e mudança.



**Figura 27 - Conceitos básicos do modelo de acontecimento e mudança do BPDM (Rivet et al, 2007)**

### **Acontecimento**

É uma entidade dinâmica que possui certa duração no tempo, sendo também o contexto de certas mudanças.

### **Mudança**

É uma entidade dinâmica que ocorre em um ponto do tempo, tendo duração atômica.

A principal função do modelo de acontecimento e mudança é permitir a ordenação no tempo de eventos relacionados ao ciclo de vida de um acontecimento, como o início e o fim de uma atividade. Para este fim, o BPDM introduz os acontecimentos e mudanças comportamentais.

A Figura 28 ilustra o modelo de acontecimentos comportamentais.

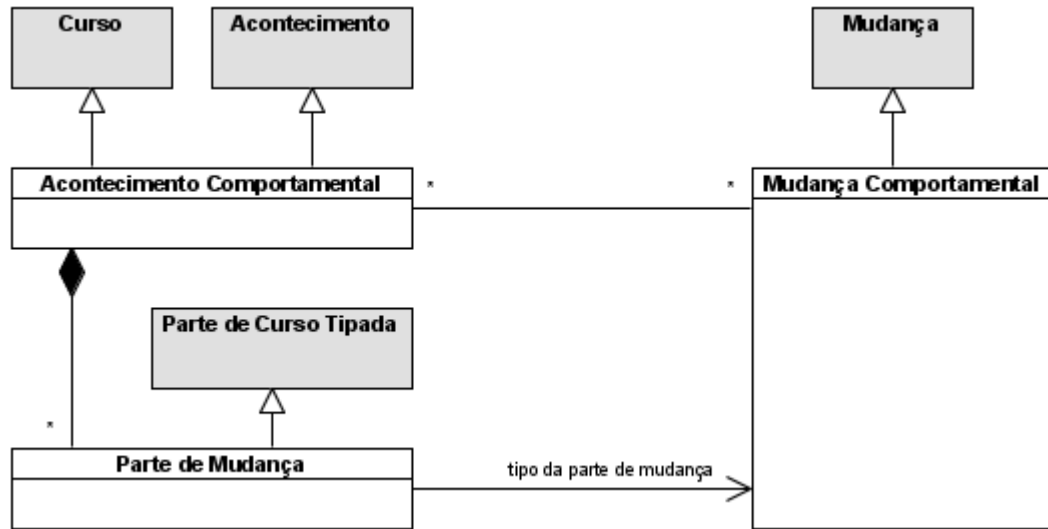


Figura 28 - Modelo de acontecimento comportamentais do BDPM (Rivet et al, 2007)

### Acontecimento Comportamental

É um tipo de acontecimento que induz ou produz mudanças comportamentais. Por ser um curso, possui partes que podem ser ordenadas no tempo, por meio de sucessões.

### Mudança Comportamental

É um tipo de mudança representado por eventos relacionados ao ciclo de vida de um acontecimento.

### Parte de Mudança

É uma parte de curso tipada cujo tipo é uma mudança comportamental. É um artifício do BDPM para permitir que eventos do ciclo de vida de um acontecimento possam ser ordenados no tempo por sucessões. Para um acontecimento comportamental, é comum definir duas mudanças comportamentais: o evento de início e o evento de término do acontecimento. Cada um destes eventos cumpre o papel de uma parte de mudança deste acontecimento. Assim sendo, estes eventos podem estar relacionados por uma sucessão, indicando que o evento de término ocorre após o evento de início.

O modelo de acontecimento e mudança introduz também o conceito de condição de mudança, conforme ilustrado na Figura 29.

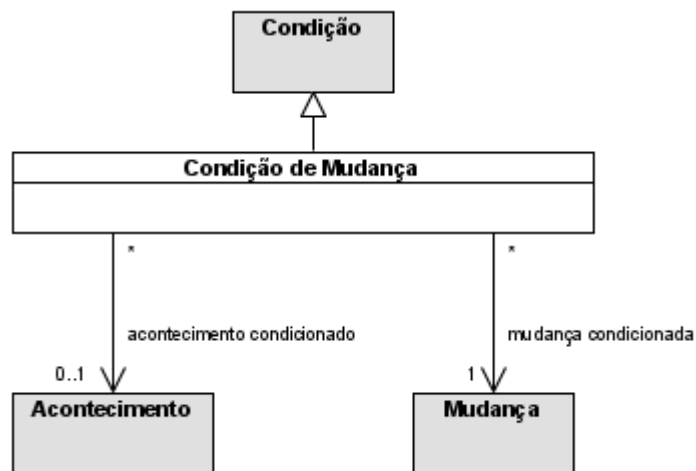


Figura 29 - Condição de mudança no BPDM (Rivet et al, 2007)

### Condição de Mudança

É um tipo de condição que se torna verdadeira quando ocorre uma mudança no contexto de um acontecimento. Por exemplo, a liberação de um pagamento pode estar atrelada a confirmação de recebimento de uma mercadoria. Neste exemplo, a compra da mercadoria é um acontecimento e a confirmação de seu recebimento é uma mudança, à qual está atrelada uma condição.

#### 6.2.4 Modelo de Comportamento de Processo

Permite que acontecimentos comportamentais sejam ordenados no tempo como partes de um outro acontecimento comportamental.

A Figura 30 ilustra o modelo de comportamento de processo.

### Comportamento de Processo

É um tipo de curso cujas partes podem ser do tipo de um acontecimento comportamental.

### Etapa de Acontecimento

É uma parte de curso tipada cujo tipo é um acontecimento comportamental. As diversas etapas de um processo, como por exemplo o empacotamento ou envio de uma mercadoria, são acontecimentos que pertencem ao curso deste processo.

### Sucessão de Processo

É um tipo de sucessão que permite referenciar as partes de mudança das etapas de acontecimento. Em um processo de empacotamento e envio de mercadoria, uma sucessão de processo pode ter como origem o acontecimento de empacotamento, referenciando a parte de

mudança atrelada ao evento de término deste acontecimento. Esta sucessão de processo pode ter como destino o acontecimento de envio da mercadoria, referenciando a parte de mudança atrelada ao evento de início do acontecimento. Desta forma, o modelo garante que o envio da mercadoria só pode começar após o término do seu empacotamento.

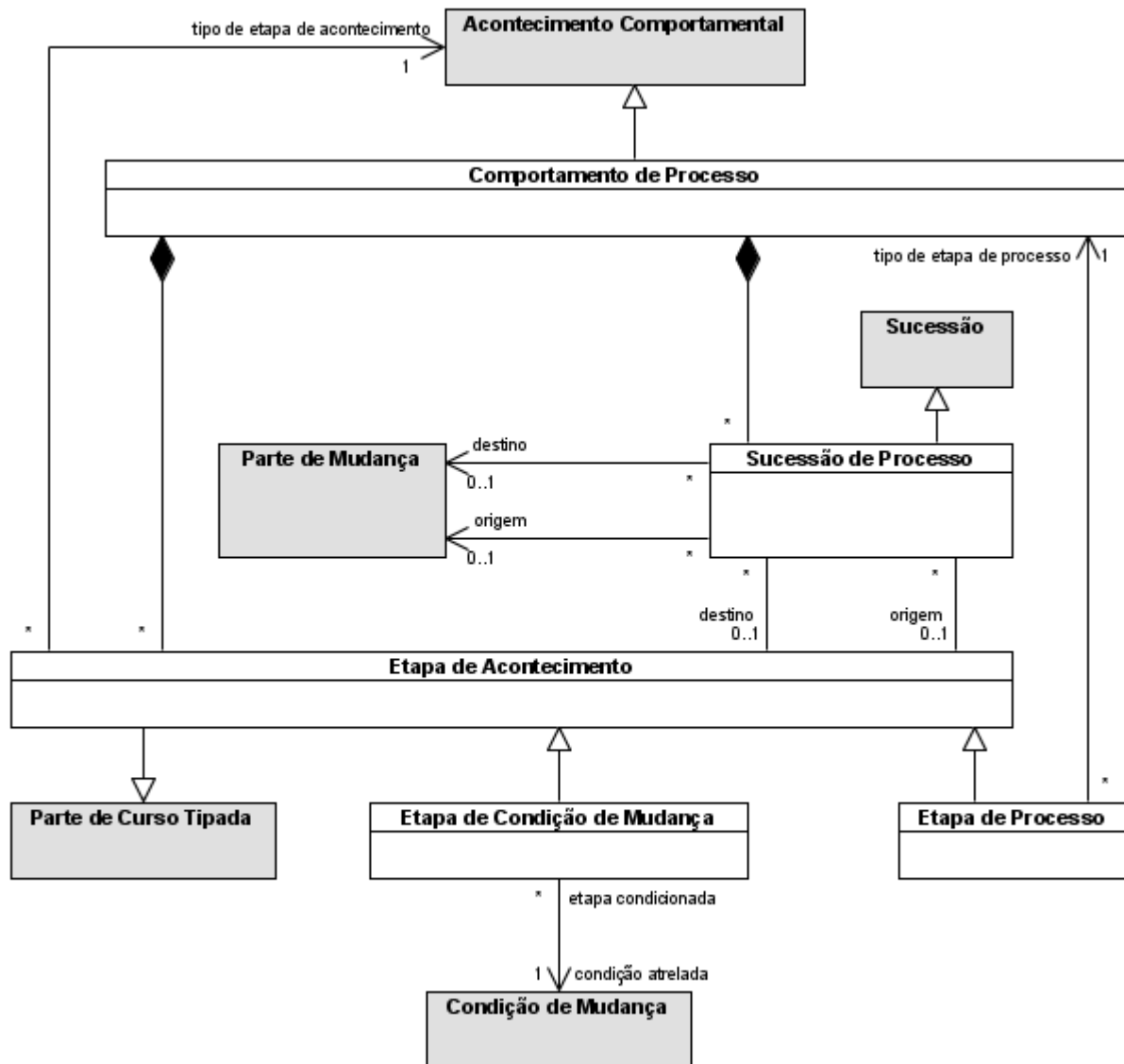


Figura 30 - Modelo de comportamento de processo do BPDM (Rivet et al, 2007)

### Etapa de Processo

É uma etapa de acontecimento cujo tipo é um comportamento de processo. Este elemento permite modelar um processo como uma etapa de um outro processo.

### Etapa de Condição de Mudança

É uma etapa de acontecimento atrelada a uma condição de mudança. Em um processo de pintura de uma peça, uma etapa de processo é a própria pintura da peça. Uma etapa seguinte

seria o empacotamento desta peça. Entretanto, entre estas duas etapas pode haver uma etapa de condição de mudança, responsável pelo secamento da tinta. Neste caso, a condição atrelada à etapa pode ser o próprio secamento da tinta ou um tempo pré-definido.

### 6.2.5 Modelo de Interação Simples

Permite que interações sejam utilizadas como uma etapa de acontecimento, podendo ser ordenadas no tempo e tendo eventos como início e fim.

A Figura 31 ilustra o modelo de interação simples.

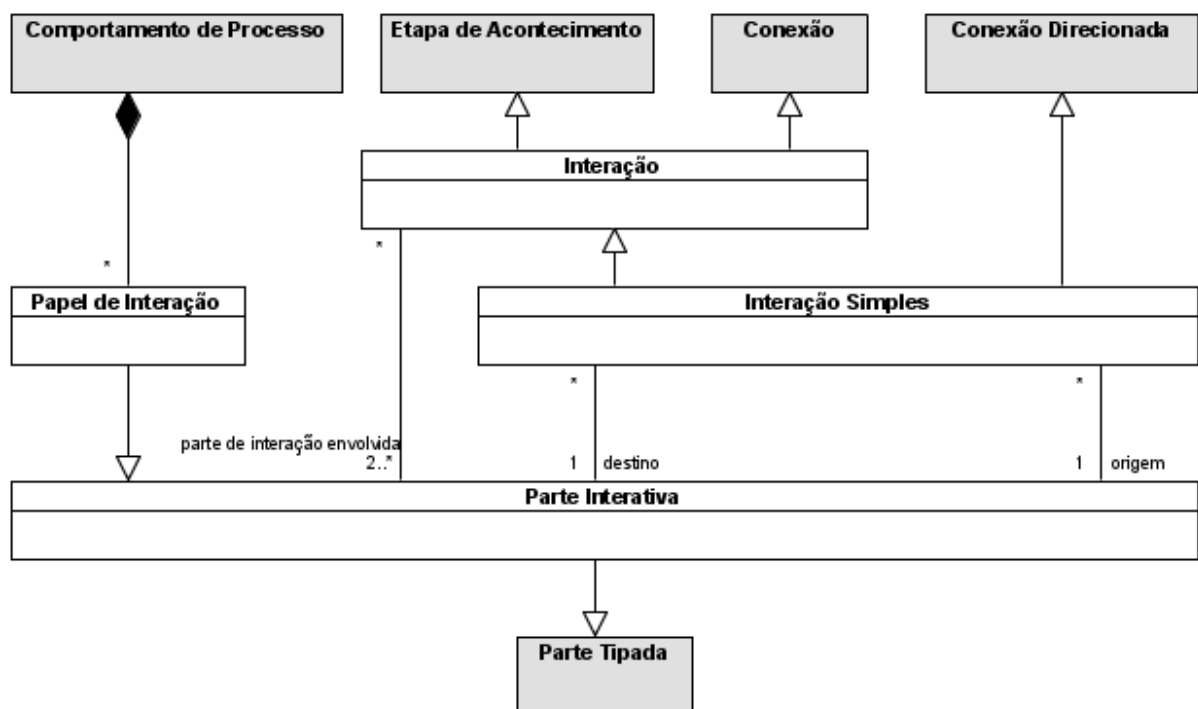


Figura 31 - Modelo de interação simples do BPDM (Rivet et al, 2007)

#### Interação

É um tipo de etapa de acontecimento, o que lhe permite ser ordenada no tempo e ter os eventos de início e término. É também uma conexão entre partes que podem interagir entre si.

#### Parte Interativa

É uma parte tipada que pode interagir com outras partes. Uma reunião entre cliente e fornecedor pode ser uma etapa em um processo. Neste exemplo, a reunião é uma interação, e o cliente e o fornecedor são partes interativas.

#### Interação Simples

Uma interação simples é uma interação entre apenas duas partes, na qual algum item ou informação é transferido entre a parte interativa de origem para a parte interativa de destino.

Em uma entrega de material, o entregador e o receptor são partes interativas de uma interação simples, que é o próprio ato da entrega. Neste caso, o material sendo entregue é o item transferido na interação.

### Papel de Interação

É um tipo de parte interativa que pertence ao contexto do comportamento de processo no qual se dá uma interação. Em uma entrega de produto ao cliente, o cliente é um papel de interação, pois é parte integrante do processo.

### 6.2.6 Modelo de Atividades

É um modelo com enfoque na visão de orquestração, cujos elementos são facilmente relacionados aos conceitos mais comuns de um processo, como o processo em si, suas atividades e os papéis responsáveis pela execução destas atividades.

A Figura 32 ilustra o modelo de atividades.

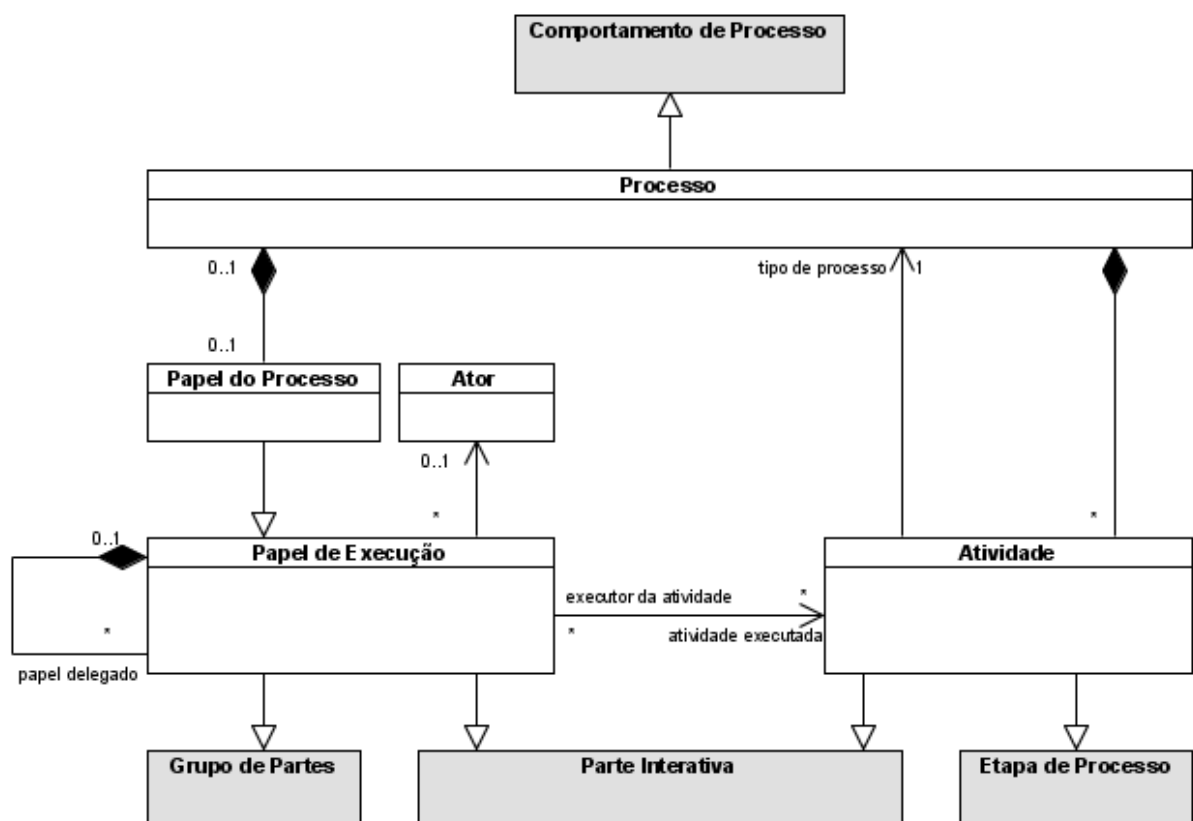


Figura 32 - Modelo de atividades do BPDM (Rivet et al, 2007)

### Processo

Um processo possui um série de atividades e interações que são executadas sob a autoridade de um indivíduo cumprindo um papel de processo.



### **Atividade**

É um tipo de etapa do processo com as seguintes características:

- Pode ser ordenada no tempo por sucessões de processo.
- Opera sob a responsabilidade de um indivíduo cumprindo um papel de execução.
- Pode ativar um sub-processo ou ser uma simples tarefa, como início e término.

Por ser um parte interativa, pode se comunicar com outras partes interativas do processo, incluindo outras atividades.

### **Papel de Execução**

Os indivíduos que cumprem um papel de execução são responsáveis pelo andamento das atividades de um processo. Além disso, podem interagir com outras partes do processo, como outros indivíduos cumprindo papéis de execução ou de processo e outras atividades.

É também um grupo de partes, permitindo que diversos indivíduos cumpram o papel de execução de uma atividade.

O modelo permite também que um papel delegue responsabilidade para outros papéis de execução.

### **Papel de Processo**

Tipo especial de papel de execução que delega ao indivíduo que o está cumprindo a responsabilidade pelo processo como um todo.

### **Ator**

É uma entidade responsável por executar as tarefas sob responsabilidade de um papel de execução.

#### **6.2.7 Extensões para BPMN**

O BPMN (*Business Process Modeling Notation*) é uma notação para modelagem de processos de negócio utilizada como prova de conceito para o BPDM. Por isso, o BPDM define alguns elementos extras para atender elementos da notação BPMN não contemplados nos demais modelos.

Por serem específicos da notação BPMN, tais elementos não fazem parte do escopo deste trabalho.

### 6.2.8 Modelo de Protocolo de Interação

É um modelo com enfoque na visão de coreografia e que permite agrupar interações formando interações mais complexas e reutilizáveis.

A Figura 33 ilustra o modelo de protocolo de interação.

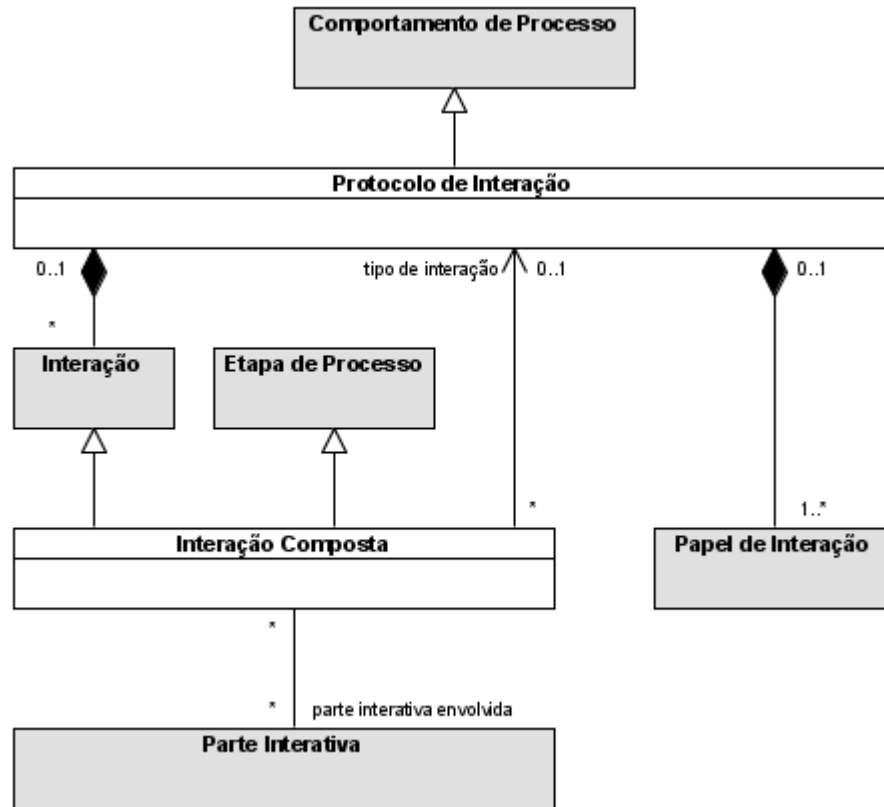


Figura 33 - Modelo de Protocolo de Interação do BPDM (Rivet et al, 2007)

#### Protocolo de Interação

É um comportamento de processo cujas etapas são interações. Por exemplo, um protocolo de interação entre duas empresas pode começar com um delas fazendo um pedido de compra, prosseguir com a outra empresa entregando o pedido, a empresa compradora enviando um pagamento e a empresa vendedora entregando um recibo. Neste exemplo, há quatro interações simples ordenadas no tempo, pois uma só deve acontecer após a outra estar concluída. As duas empresas cumprem papéis de interação neste exemplo.

#### Interação Composta

É uma etapa de processo cujo tipo é um protocolo de interação. O exemplo de duas empresas envolvendo um pedido é um protocolo de interação, mas este protocolo pode cumprir o papel de uma interação em um outro protocolo maior, que represente toda a interação entre as duas empresas. Este protocolo pode, ainda, ser reutilizado na interação com outras empresas.

### 6.3 IDEF3

De acordo com Mayer et al (1995), um dos modos mais comuns de se descrever uma situação ou um processo é contar uma estória na forma de atividades e eventos que ocorrem em uma determinada seqüência. O método IDEF3 foi elaborado para capturar este tipo de descrição.

O método IDEF3 é composto por duas partes que são usadas em conjunto para a geração de modelos de processo:

- Uma notação para expressar a descrição dos processos e cenários.
- Um procedimento para o levantamento dos elementos relevantes, incorporando técnicas testadas e aprovadas na prática.

Além das duas partes do método IDEF3, há ainda duas estratégias adotadas ao se criar um modelo de processo:

- Esquema centralizado no processo (exemplificado na Figura 34).
- Esquema centralizado nos objetos (exemplificado na Figura 35).

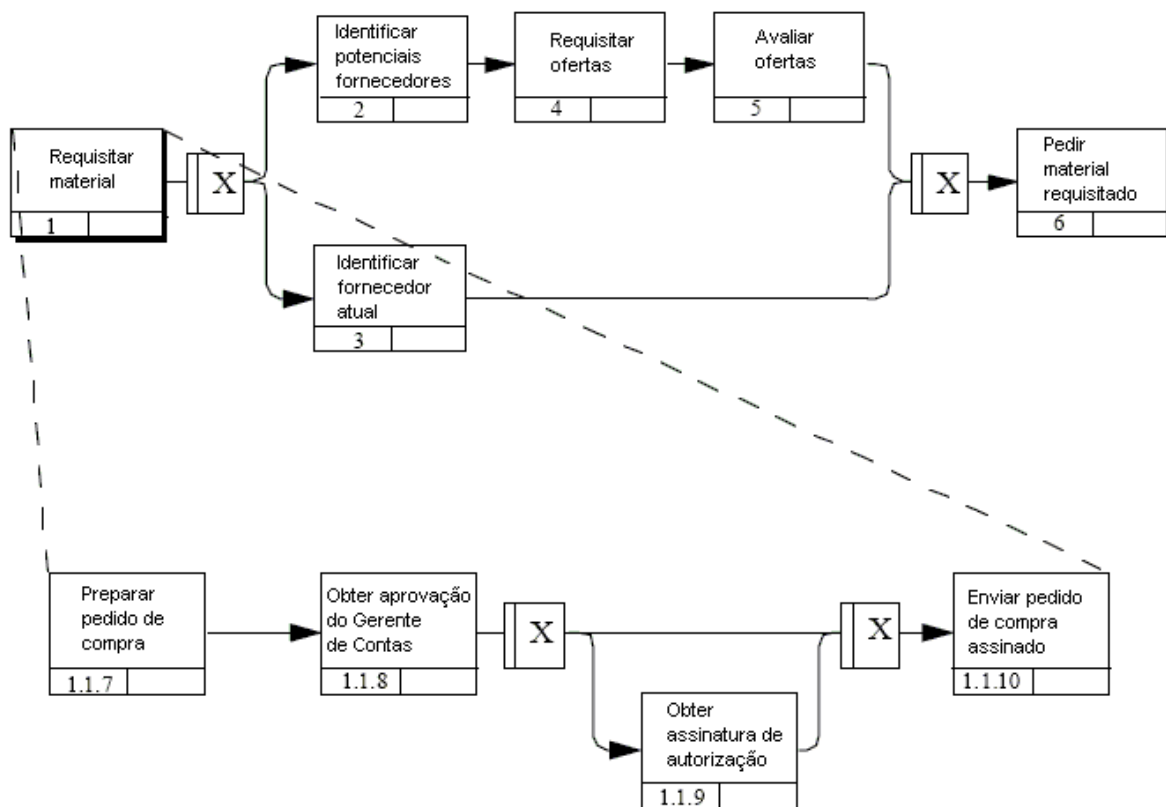


Figura 34 - Exemplo de esquema centrado em processo (Mayer et al, 1995)

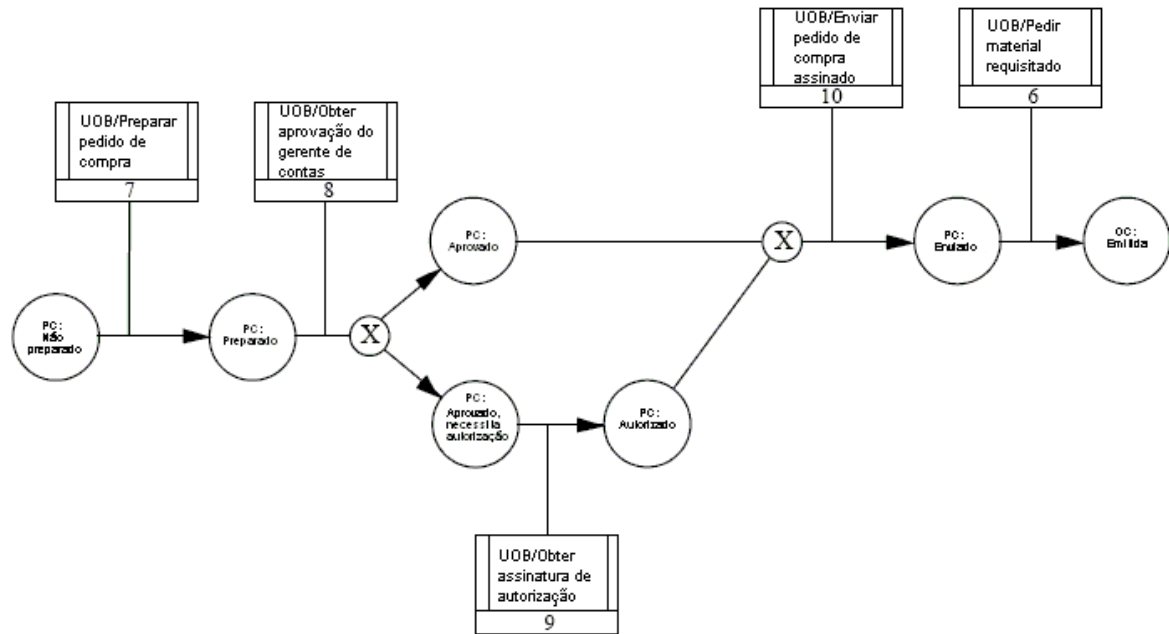


Figura 35 - Exemplo de esquema centrado em objetos (Mayer et al, 1995)

### 6.3.1 Notação do IDEF3

A notação do IDEF3 define uma série de elementos representativos de um processo, para ambas as estratégias. Os tópicos a seguir apresentam uma breve descrição dos elementos mais relevantes.

#### Unidade de Comportamento – UOB (*Units of Behavior*)

É o elemento mais comum do esquema centrado em processos. Representa um tipo qualquer de acontecimento, como, por exemplo, uma atividade ou um processo.

Uma UOB pode ser decomposta em diversas UOBs, de tal forma que uma única UOB possa representar um cenário próprio.

As UOBs podem ser conectadas entre si por meio de vínculos, que indicam o seqüenciamento destas UOBs. Além disso, as UOBs podem ser referenciadas nas transições de estados de um objeto.

#### Vínculo (*Link*)

Os vínculos conectam UOBs de forma a representar a dinâmica do processo, isto é, de forma a seqüenciar os acontecimentos no tempo.

A linguagem do IDEF3 define uma série de tipos de vínculos:

- Vínculo de precedência simples: expressa a precedência temporal entre duas instâncias de UOBs.

- Vínculo de precedência com restrição: similar ao vínculo de precedência simples, mas, ao contrário deste, imputa uma ou mais restrições sobre a ocorrência das instâncias das UOBs. As restrições mais simples obrigam que uma instância da UOB deva sempre ocorrer após uma outra, ou que uma instância não pode existir se outra ainda não tiver ocorrido. Há, entretanto, outros tipos de restrições, não necessariamente relacionados à existência ou não de uma instância de uma UOB.
- Vínculo relacional: expressa uma relação qualquer entre duas UOBs, cuja natureza é definida pelo usuário. Esta relação não é temporal, pois tal papel é cumprido pelos vínculos de precedência.
- Vínculo de transição: expressa a transformação de um objeto em outro ou a mudança de estado de um objeto.

### **Junção (*Junction*)**

As junções permitem que um processo possua seqüências de atividades (caminhos) que ocorram em paralelo. O IDEF3 define 3 tipos básicos de junção, associados a operadores binários conhecidos:

- E: todos os caminhos que saem ou chegam a esta junção devem ocorrer em paralelo.
- OU: pelo menos um dos caminhos que saem ou chegam a esta junção deve ocorrer. Os caminhos relacionados a esta junção são alternativos.
- OU exclusivo: apenas um dos caminhos que saem ou chegam a esta junção pode ocorrer.

As junções também são classificadas por outros dois critérios. Um dos critérios classifica a junção como junção de divergência (divide um caminho em múltiplos caminhos) ou junção de convergência (agrupa diversos caminhos em um só). O outro critério classifica a junção como síncrona (todos os caminhos iniciam ou encerram ao mesmo tempo) ou assíncrona (os caminhos podem ter início e fim distintos dos outros caminhos).

### **Referente (*Referent*)**

Não representam elementos reais de um processo, sendo apenas um artifício da linguagem para a construção dos modelos. Os referentes cumprem o papel de outros elementos, como as UOBs, permitem a representação de laços em um processo e possibilitam o relacionamento de elementos dos esquemas centrados em processo e objetos.

### **Objeto e Estados de Objetos (*Object and Object States*)**

Representam tipos de objetos e seus respectivos estados, que podem sofrer alteração ao longo da execução de um processo. São os principais elementos dos esquemas centrados em objetos, cujo enfoque é registrar as diversas transformações sofridas pelos objetos relevantes a um processo.

#### **6.3.2 Procedimento do IDEF3**

O procedimento recomendado pelo IDEF3 visa à construção dos dois tipos de esquema, e é composto por uma série de passos:

- 1) Definição do escopo, o que inclui a definição do contexto e do propósito para a utilização do IDEF3
- 2) Coleta de dados
  - a) Entrevista com os especialistas da área, que resulta nas primeiras descrições dos cenários. Estas descrições devem ilustrar como o sistema funciona naquele momento, e não como deveria funcionar.
  - b) Análise das primeiras descrições, resultando na criação de três listas: objetos (incluindo funções exercidas no processo), atividades e fatos sobre os objetos e restrições.
- 3) Elaboração dos esquemas de processo
  - a) Posicionamento inicial das UOBs, baseado na lista de atividades. Este posicionamento já inclui os vínculos de precedência e as junções necessárias.
  - b) Desenvolvimento das elaborações. Para cada UOB, junção ou vínculo, é detalhado quais são os fatos ou restrições relevantes, e sua lógica é explicada.
  - c) Revisão dos esquemas de processo com os especialistas.
- 4) Elaboração dos esquemas de objetos
  - a) Seleção dos objetos relevantes ao processo, a partir da lista de objetos.
  - b) Determinação dos possíveis estados destes objetos.
  - c) Posicionamento inicial dos objetos e estados, incluindo os vínculos de transição e as junções.
  - d) Inclusão de referências às UOBs responsáveis pela transformação dos objetos.
  - e) Desenvolvimento das elaborações para os objetos, vínculos de transição e junções.

- f) Revisão dos esquemas de objetos com os especialistas.

#### **6.4 Aplicação do método IDEF3 ao meta-modelo BPDM**

Neste trabalho, o IDEF3 será utilizado como método-guia para a determinação dos elementos que representam o aspecto dos processos no contexto organizacional. Isso significa que nem todos os elementos do BPDM serão aproveitados, e sim apenas aqueles que são diretamente referenciados pelo IDEF3.

A utilização do método IDEF3 como método-guia não implica que o modelo de contexto organizacional proposto só possa ser utilizado por quem também utiliza este método. Para os usuários de outros métodos de modelagem de processos, é possível que o modelo precise ser adaptado ou estendido, mas este tipo de análise está fora do escopo deste trabalho.

Os tópicos a seguir apresentam o mapeamento entre elementos do IDEF3 e do BPDM. O mapeamento está representado pelo símbolo “↔”.

##### **UOB ↔ Atividade**

Uma UOB possui as seguintes características:

- a) É um tipo qualquer de acontecimento.
- b) Pode ser decomposta em um sub-processo.
- c) Pode ser referenciada na transição de estados de um objeto
- d) É conectada por vínculos de precedência ou relacionais.

O elemento do BPDM mais adequado para representar uma UOB é uma atividade, por possuir as seguintes características:

- i) É conectado por sucessões de processo e, mais genericamente, por um tipo qualquer de conexão entre partes (d).
- ii) É uma parte de curso tipada cujo tipo é um acontecimento comportamental (a).
- iii) É um tipo de comportamento de processo, e um dos tipos de comportamento de processo é um processo em si (b).

No presente trabalho, o aspecto “O quê?” não será considerado. Como consequência, a característica (c) também será descartada.

## **Vínculo**

Um vínculo conecta dois elementos (UOB ou junção) entre si. Existem quatro tipos de vínculo no método IDEF3.

### **Vínculo de Precedência Simples ↔ Sucessão**

- a) Conecta dois elementos com uma relação temporal.
- b) Pode conectar UOBs e junções.

O elemento do BDPM mais adequado para representar um vínculo de precedência simples é uma sucessão, por possuir as seguintes características:

- i) Ordena duas partes de curso no tempo (a).
- ii) Permite conectar dois tipos de parte de curso: as partes tipadas (que se especializam em atividades) e os controles de curso (que representam junções) (b).

### **Vínculo de Precedência com Restrição ↔ Sucessão de Processo**

- a) Possui as mesmas características de um vínculo de precedência simples.
- b) Define restrições quanto à forma em que os elementos relacionados são ordenados no tempo.
- c) Define restrições genéricas entre os elementos relacionados.

O elemento do BDPM mais adequado para representar um vínculo de precedência com restrição é uma sucessão de processo, por possuir as seguintes características:

- i) É um tipo de sucessão (a).
- ii) Permite referenciar os eventos de ciclo de vida de uma atividade, o que possibilita uma ordenação mais complexa das atividades no tempo (b).
- iii) Por ser uma conexão, pode possuir uma condição atrelada (b) (c).

### **Vínculo Relacional ↔ Conexão**

- a) Não define uma relação temporal ou seqüencial entre os elementos.
- b) Define restrições genéricas entre os elementos relacionados.

O elemento do BDPM mais adequado para representar um vínculo relacional é uma conexão, por possuir as seguintes características:

- i) Pode possuir uma condição atrelada (b).



- ii) É um elemento mais alto na hierarquia das sucessões que não define relação de seqüenciamento ou ordenação no tempo (a).

Embora uma conexão possa relacionar diversos elementos, para o IDEF3 este relacionamento deve se restringir a apenas dois elementos. Uma conexão direcionada poderia ser utilizada neste caso, pois limita as partes conectadas em duas. Entretanto, optou-se pela conexão pois uma conexão direcionada implica em uma relação de origem e destino que não existe para os vínculos relacionais do IDEF3.

### **Vínculo de Transição**

- a) Relaciona dois objetos e/ou estados de um objeto.
- b) Pode estar relacionado a uma UOB.

Os vínculos de transição relacionam elementos do aspecto “O quê?”. Como este aspecto não será considerado neste trabalho, tais vínculos também não o serão.

### **Junção ↔ Controle de Curso**

A característica principal de uma junção é permitir a ramificação de caminhos em um processo. O elemento do BDPM mais adequado para representar uma junção é um controle de curso, que permite conectar uma atividade a uma série de outras atividades.

No IDEF3, uma mesma junção pode cumprir dois papéis: divergência ou convergência. No BPDM, há elementos distintos de acordo com este critério. As junções de divergência são denominadas de divisão, e as de convergência simplesmente de junção. São três os tipos de junção definidos no IDEF3:

#### **Junção E ↔ Junção ou Divisão em Paralelo**

- a) No papel de divergência, ramifica um processo em diversos caminhos, e todos eles devem acontecer.
- b) No papel de convergência, aguarda a conclusão de todos os caminhos que chegam até a junção para então dar prosseguimento ao processo.

Uma divisão em paralelo cria múltiplos caminhos de saída e exige que todos sejam executados (a). Uma junção em paralelo aguarda o encerramento de todos os caminhos de entrada para dar procedimento ao processo (b).

### Junção OU

- a) No papel de divergência, ramifica um processo em diversos caminhos alternativos, e pelo menos um deles deve acontecer.
- b) No papel de convergência, aguarda a conclusão de todos os caminhos alternativos em execução, que não são necessariamente todos os caminhos conectados à junção, para então dar prosseguimento ao processo.

Não há no BPDM um controle de curso com estas características.

### Junção OU Exclusivo ↔ Junção ou Divisão Exclusiva

- a) No papel de divergência, ramifica um processo em diversos caminhos alternativos, e apenas um deles deve acontecer.
- b) No papel de convergência, aguarda a conclusão do caminho alternativo em execução, para então dar prosseguimento ao processo.

Uma divisão exclusiva cria múltiplos caminhos de saída e exige que apenas um deles seja executado (a). Uma junção exclusiva aguarda o encerramento do único caminho de entrada em execução para dar procedimento ao processo (b).

### Objeto e Estados dos Objetos

Conforme mencionado anteriormente, os objetos não estão no escopo deste trabalho, que não está considerando o aspecto “O quê?” na construção do modelo de contexto organizacional.

A Tabela 5 resume o mapeamento entre IDEF3 e BPDM.

**Tabela 5 - Mapeamento entre IDEF3 e BPDM**

IDEF3	BPDM
UOB	Atividade
Vínculo de Precedência Simples	Sucessão
Vínculo de Precedência com Restrição	Sucessão de Processo
Vínculo Relacional	Conexão
Junção	Controle de Curso
Junção E	Junção ou Divisão em Paralelo
Junção OU Exclusivo	Junção ou Divisão Exclusiva

## 6.5 BMM

De acordo com Chapin et al (2005), se uma organização executa uma atividade, ela deve saber o porquê de fazê-lo. O BMM é um modelo utilizado para registrar estes porquês, na forma de planos de negócio. O BMM apresenta as seguintes definições:

- Elementos pertencentes ao plano de negócio.

- Fatores que influenciam um plano de negócio.
- Relacionamentos entre os elementos do plano e os fatores de influência

O BMM divide seu modelo em duas grandes áreas:

- Meios e Fins: contém os elementos do plano de negócio, ou seja, o que é preciso para atingir os objetivos esperados
- Influenciadores: contém os fatores de influência, ou seja, a razão para a existência de cada elemento do plano de negócio.

### 6.5.1 Conceitos da área de Fins e Meios

A Figura 36 ilustra os elementos que representam um fim no BMM.

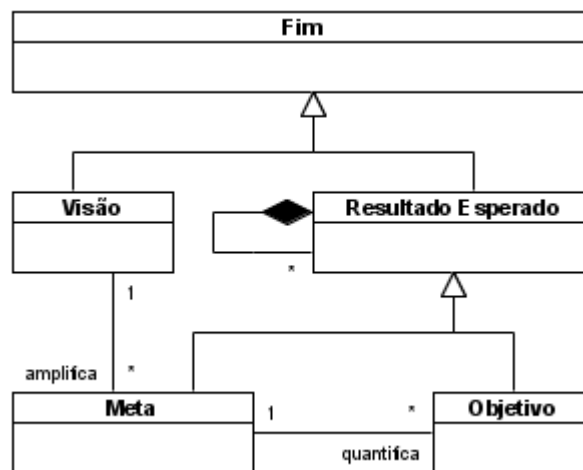


Figura 36 – Meta-modelo de fins no BMM (Chapin et al, 2005)

#### **Fim**

Representa algo ou situação que a organização deseja alcançar, sem especificar como.

#### **Visão**

Tipo de fim que descreve o estado futuro que uma organização deseja ter, de maneira pouco detalhada. Por exemplo, ser a melhor consultoria em um dado setor.

#### **Resultado Esperado**

Tipo de fim que descreve um estado ou situação que uma organização deseja atingir e manter. É um fim menos abstrato do que a visão. Um resultado esperado pode ser decomposto em outros resultados, trazendo maior detalhamento a um fim.

## Meta

Tipo de resultado esperado que amplifica uma visão, ou seja, que define o que deve ser satisfeito para que o estado futuro definido na visão seja alcançado. Por exemplo, aumentar a satisfação dos clientes.

## Objetivo

É o tipo de resultado esperado menos abstrato. A definição de um objetivo deve ser simples, representando um fim atingível, mensurável e com prazo bem definido. Por ser mensurável, um objetivo quantifica uma meta. Por exemplo, disponibilizar um *call center* para os clientes até o final do ano.

A Figura 37 ilustra os elementos que representam um meio no BMM.

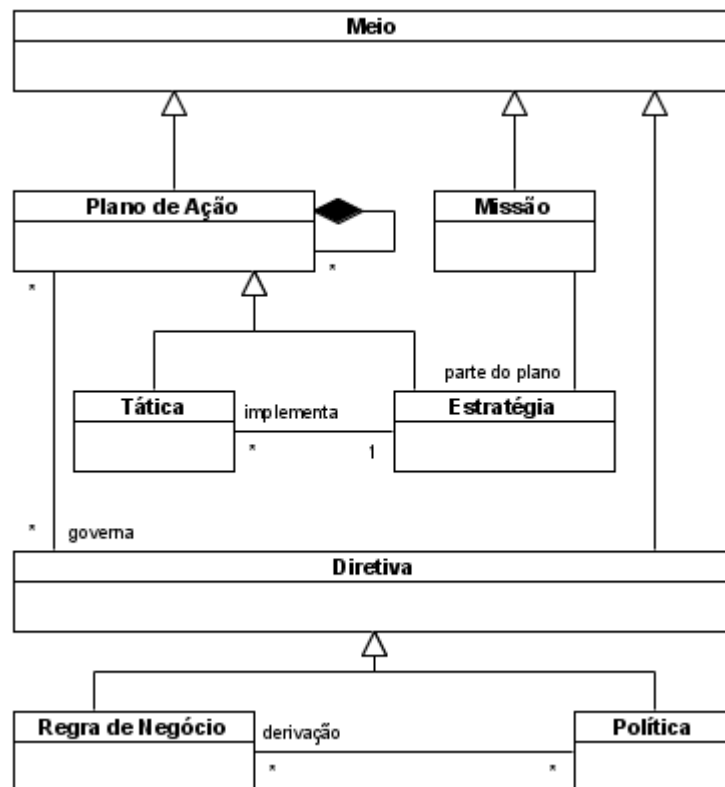


Figura 37 – Meta-modelo de meios no BMM (Chapin et al, 2005)

## Meio

Representa qualquer tipo de ação ou instrumento voltado para a realização de um fim. Um meio não especifica quais passos devem ser executados, e sim o tipo de ação necessária para se atingir um fim.

**Missão**

Tipo de meio que representa uma abordagem não detalhada, e de longo prazo, para se atingir uma visão. Por exemplo, prover consultoria e alocação de recursos para qualquer empresa na América do Sul.

**Plano de Ação**

É uma abordagem que canaliza esforços para atingir um resultado esperado. Um processo, por exemplo, é utilizado para realizar um plano de ação. Assim como um resultado esperado, o plano de ação pode ser decomposto em planos menores.

**Estratégia**

É uma parte mais específica da missão cujo objetivo é permitir que uma meta seja alcançada. Por exemplo, implantar um sistema de relacionamento com clientes.

**Tática**

É uma implementação da estratégia, visando a atingir um objetivo. Por exemplo, equipar a equipe de vendas com computadores pessoais.

**Diretiva**

Indica como um plano de ação deve ou não ser colocado em prática. Uma diretiva pode governar um plano de ação. Por exemplo, a diretiva de não permitir entregas de pizzas fora de um raio de 30 km governa a estratégia de entregar pizzas no local especificado pelo cliente. Uma diretiva pode também ser a origem de um plano de ação. Por exemplo, a diretiva de reduzir a depreciação de veículos dá origem à tática de efetuar um rodízio dos veículos entre diversas franquias de uma locadora de automóveis. Por fim, uma diretiva pode dar suporte para atingir um resultado. A diretiva de fornecer o número de telefone dos vendedores aos clientes dá suporte à meta de prover um atendimento de alta qualidade.

**Política**

É um tipo de diretiva cujo propósito é controlar uma organização e seus processos. Possui um nível maior de abstração se comparado a uma regra de negócio. A diretiva de reduzir a depreciação dos veículos de aluguel é uma política da empresa. Esta política rege a maneira como os aluguéis são efetuados, mas não implica em nenhuma aplicação prática direta.

## Regra de Negócio

É um tipo de diretiva com uma definição bem simples e direta, que pode ser posta em prática com facilidade. Normalmente, é criada a partir de uma política. Por exemplo, a política de redução da depreciação dos veículos dá origem a regra de não permitir a renovação de aluguel por telefone para os veículos cuja quilometragem ultrapasse 50.000 km.

### 6.5.2 Conceitos da área de Influenciadores

A Figura 38 ilustra os elementos que representam influenciadores no BMM.

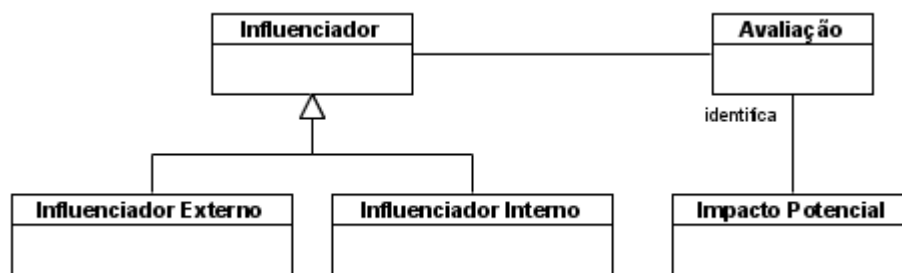


Figura 38 – Meta-modelo de Influenciadores no BMM (Chapin et al, 2005)

#### Influenciador

Um influenciador é qualquer coisa que possa produzir um efeito em uma organização. Entretanto, a definição de um influenciador não o classifica como um efeito positivo ou negativo para a organização, e sim neutro.

#### Influenciador Externo

O influenciador externo não faz parte da organização e, portanto, foge ao seu controle. Exemplos de influenciadores externos são competidores, clientes, fornecedores, leis, tecnologia, etc...

#### Influenciador Interno

O influenciador interno faz parte ou é uma característica da organização. Embora sob seu controle, não é fácil modificar estes influenciadores. Exemplos de influenciadores internos são hábitos, valores, qualidade dos recursos disponíveis, infra-estrutura, etc...

#### Avaliação

É um julgamento que se faz sobre um influenciador e que identifica como ele pode afetar o uso de um meio ou o alcance de um fim. É a avaliação sobre um determinado influenciador que o torna um fator positivo ou negativo para a organização.

## Impacto Potencial

A caracterização de um influenciador como um fator positivo ou negativo é quantificada ou qualificada por meio da declaração de impactos potenciais à organização. Uma avaliação positiva identifica possíveis recompensas e uma avaliação negativa identifica riscos ao negócio.

### 6.5.3 Relacionamento entre meios e fins

No BMM, os conceitos de fins, meios e influenciadores possuem alguns relacionamentos entre si, conforme ilustra a Figura 39.

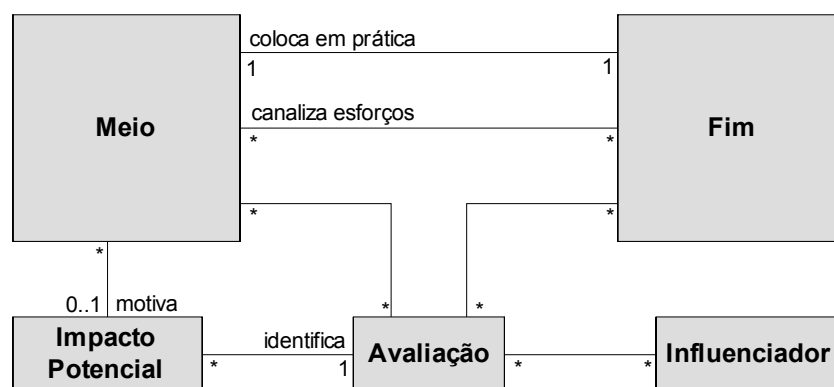


Figura 39 - Relacionamentos entre fins, meios e avaliações (Chapin et al, 2005).

Fins e meios possuem ligações diretas entre si. O objetivo de uma missão é colocar em prática a visão da organização. Um plano de ação visa a canalizar os esforços necessários para que a organização possa atingir os resultados esperados.

Uma avaliação sobre um influenciador é realizada considerando-se a maneira como um meio ou fim pode ser afetado por este influenciador. E os impactos potenciais motivam a criação de diretivas que auxiliam a organização a lidar com os riscos identificados ou a se beneficiar das possíveis recompensas.

## 6.6 OSM

De acordo com os autores do OSM (Cummins et al, 2006), a tradicional estrutura única de organização hierárquica não é mais válida na maioria das organizações modernas. Nestas novas organizações, suas unidades podem se relacionar entre si por meio de relações variadas: hierárquicas, de parceria, relações temporárias de colaboração no desenvolvimento de um produto, etc...

O OSM é apresentado em quatro partes, cada uma com um enfoque distinto do meta-modelo:

- Núcleo
- Entidade Legal
- Posição
- Unidade Organizacional

### 6.6.1 Modelo do Núcleo

A Figura 40 ilustra o núcleo do OSM. As entidades com nome em itálico são abstratas, não podendo ser instanciadas em um modelo.

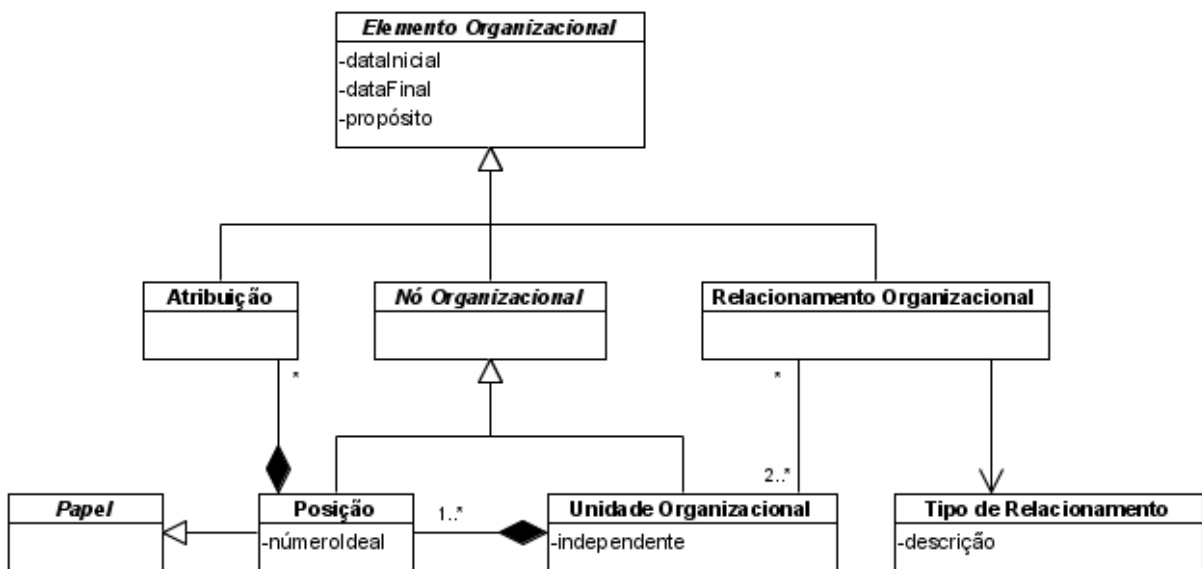


Figura 40 - Modelo núcleo do OSM (Cummins et al, 2006)

#### *Elemento Organizacional*

Um elemento organizacional é todo elemento que possui uma declaração de propósito e uma faixa de datas dentro da qual o elemento está efetivo na organização.

#### *Nó Organizacional*

Representa um nó da estrutura da organização.

#### **Unidade Organizacional**

É o elemento principal da estrutura organizacional. Representa qualquer associação de pessoas no contexto da organização e reconhecida por ela. Uma unidade organizacional pode ser a própria organização, um departamento, uma equipe, um comitê ou grupo de estudos, por exemplo. Uma unidade organizacional é considerada independente quando não possui uma unidade organizacional pai.



## **Relacionamento Organizacional**

Relaciona unidades organizacionais, normalmente com base em níveis de autoridade entre elas.

### **Tipo de Relacionamento**

Característica do relacionamento entre unidades organizacionais, contemplando desde a tradicional relação hierárquica como uma relação temporária de colaboração entre organizações parceiras.

### ***Papel***

Elemento que representa tanto um papel dentro da estrutura organizacional como na execução de atividades de um processo.

### **Posição**

Define um papel na organização. Uma posição pode ter um número ideal de pessoas atribuídas a ela. Uma posição do tipo gerente, por exemplo, geralmente tem o número ideal igual a um.

### **Atribuição**

Relaciona uma pessoa a uma posição em um dado intervalo de tempo. Uma mesma pessoa pode ser atribuída a diversas posições, ao mesmo tempo ou em momentos distintos. E uma posição também pode possuir diversas pessoas atribuídas a ela, inclusive ao mesmo tempo.

## **6.6.2 Modelo de Entidade Legal**

A Figura 41 ilustra o modelo de entidade legal do OSM. As entidades representadas em fundo cinza já foram definidas em modelos do OSM apresentados anteriormente.

### **Entidade Legal**

É uma entidade com procedência legal para participar de contratos e possuir bens e propriedades. Existem dois tipos de entidades legais: as pessoas que trabalham na organização e a organização em si.

### **Informação de Contato**

Representa o endereço, número de telefone e outras informações que permitem que uma entidade legal seja contatada.

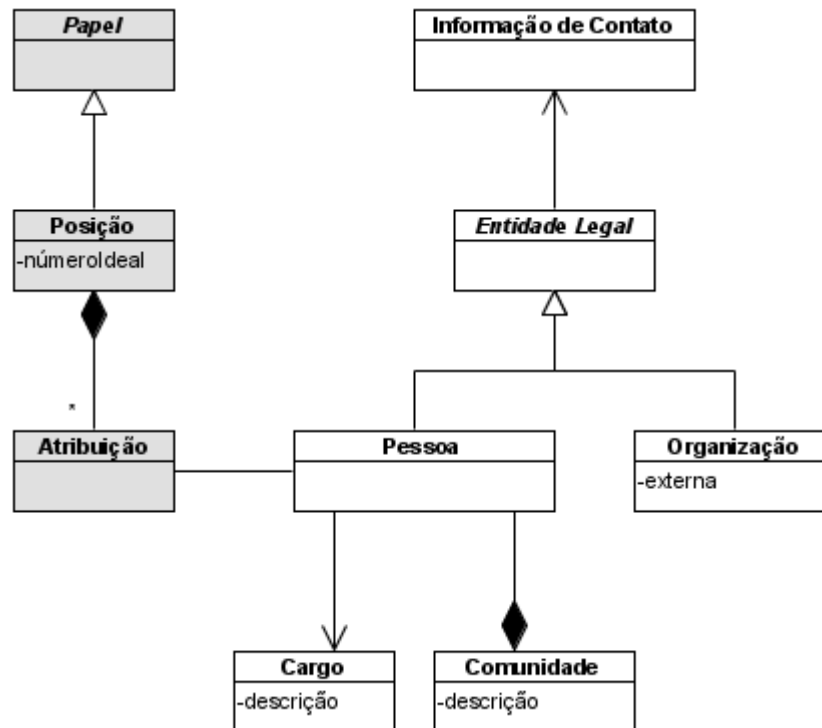


Figura 41 - Modelo de entidade legal do OSM (Cummins et al, 2006)

## Organização

É a unidade organizacional no topo da hierarquia. Todas as unidades organizacionais estão diretamente ligadas a ela por serem parte da organização. Uma organização pode ser externa, como um parceiro de negócios, por exemplo.

## Pessoa

Representa uma pessoa associada com a organização, seja ela um funcionário ou um prestador de serviço. Somente uma pessoa pode ocupar uma posição e, embora seja comum atribuir uma decisão a uma unidade organizacional, há sempre uma pessoa diretamente responsável por ela.

## Cargo

Representa o cargo de uma pessoa na organização. Diferentes pessoas podem possuir o mesmo cargo e atuar em diferentes unidades organizacionais e posições.

## Comunidade

Um grupo de pessoas com um interesse comum. Pessoas só possuem relacionamentos entre si por meio de posições na unidade organizacional ou por serem membros de uma mesma comunidade.

### 6.6.3 Modelo de Posição

A Figura 42 ilustra o modelo de posição do OSM.

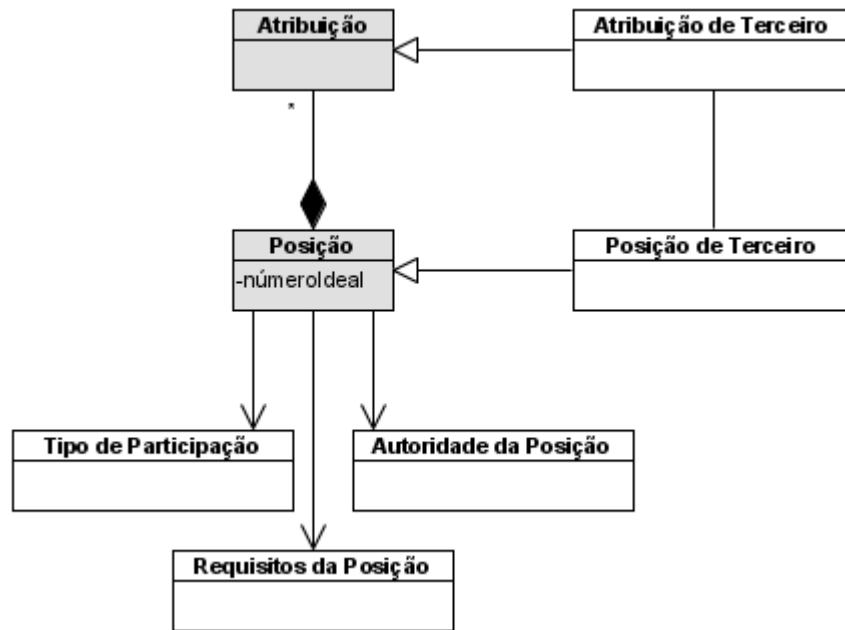


Figura 42 - Modelo de posição do OSM (Cummins et al, 2006)

#### Tipo de Participação

Caracteriza a natureza da participação de uma posição na organização, como a gerência, por exemplo.

#### Autoridade da Posição

Indica o grau de autoridade de uma posição. Pode incluir, por exemplo, nível de gasto permitido, autorização para contratações e demissões, entre outros.

#### Requisitos da Posição

Representa os requisitos para uma pessoa ocupar uma posição.

#### Atribuição de Terceiro

Similar a uma atribuição, mas referente aos prestadores de serviço terceirizados.

#### Posição de Terceiro

Similar a uma posição, mas referente aos prestadores de serviço terceirizados.

### 6.6.4 Modelo de Unidade Organizacional

A Figura 43 ilustra o modelo de unidade organizacional.

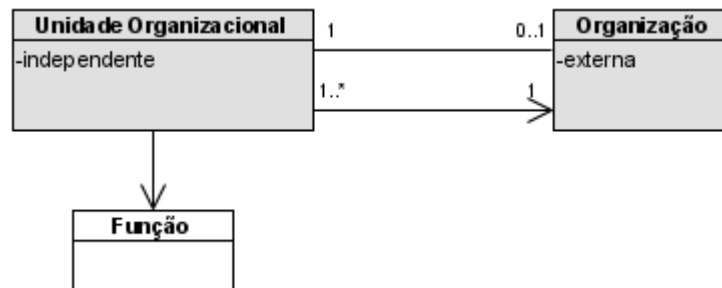


Figura 43 - Modelo de unidade organizacional do OSM (Cummins et al, 2006)

#### Organização

Embora a organização já tenha sido apresentada no modelo de entidade legal, seu relacionamento com a unidade organizacional requer maiores informações. Uma organização é uma unidade organizacional, posicionada no topo da hierarquia. Além disso, todas as unidades organizacionais referenciam diretamente a organização, permitindo que atuem como entidades legais.

#### Função

Descreve a função ou responsabilidade de uma unidade organizacional.

### 6.7 Representação do Modelo de Contexto Organizacional

Dois passos são necessários para a criação do modelo de contexto organizacional:

- Criação dos modelos individuais de cada aspecto organizacional considerado.
- Junção dos modelos individuais em um único modelo.

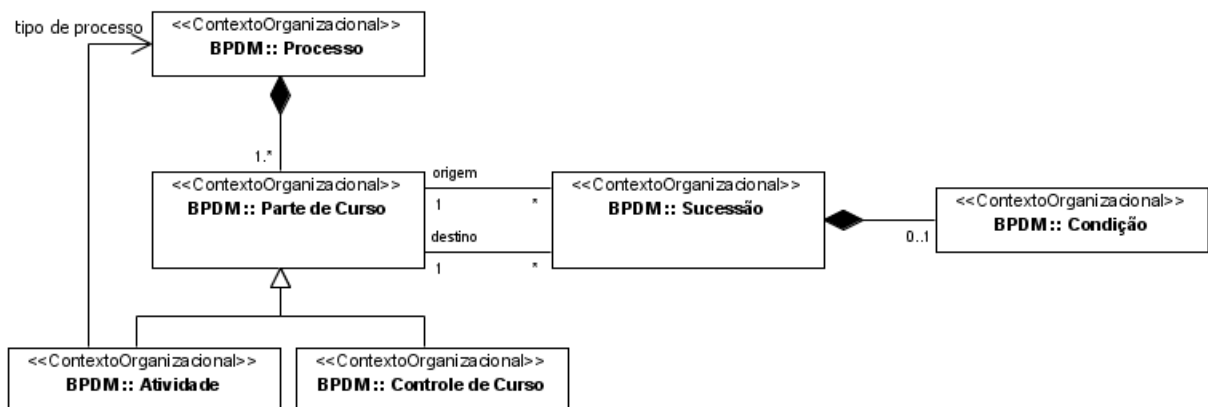
O modelo do aspecto dos processos é criado a partir do mapeamento entre BPDM e IDEF3, resumido na Tabela 5 apresentada anteriormente.

Três importantes elementos são extraídos deste mapeamento: Atividade, Controle de Curso e Sucessão. O elemento UOB é diretamente mapeado ao elemento Atividade. As junções do IDEF3 são representadas pelos Controles de Curso. Embora o IDEF3 defina dois tipos de junção (E e OU), apenas a generalização destes elementos será apresentada no modelo. Isso não significa que as junções ou divisões em paralelo e as junções ou divisões exclusivas não façam parte do modelo do aspecto dos processos. Elas apenas são omitidas da representação gráfica do modelo para efeito de simplificação do diagrama, sem perda do conceito, uma vez que sua generalização (Controle de Curso) está presente.

Três tipos de vínculos do IDEF3 são mapeados a três elementos do BPDM: Vínculo Relacional é mapeado a Conexão, Vínculo de Precedência Simples a Sucessão e Vínculo de Precedência com Restrição a Sucessão de Processo. Sucessão de Processo estende Sucessão que por sua vez estende Conexão. Desta forma, o elemento Sucessão de Processo une as características de si próprio, da Sucessão e da Conexão. Portanto, apenas este elemento será representado no modelo. Entretanto, para simplificação da leitura do modelo, o termo Sucessão de Processo será substituído pelo termo Sucessão.

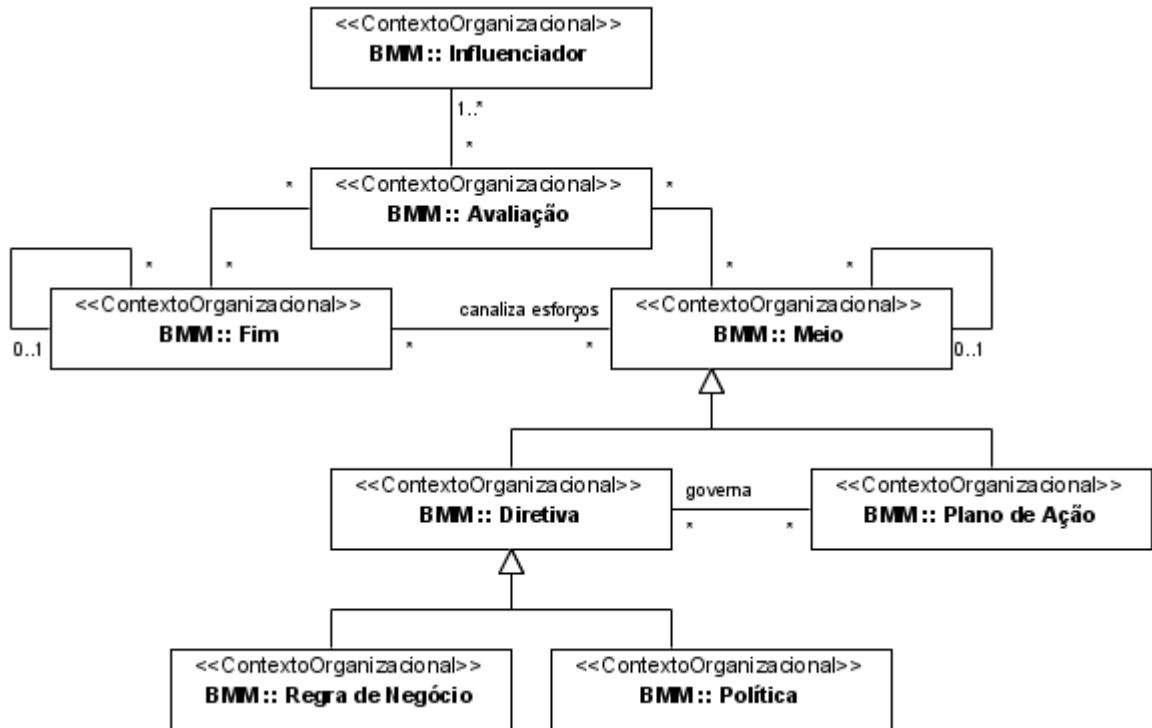
Além dos três elementos extraídos do mapeamento entre BPDM e IDEF3, outros três elementos completam o modelo: Processo, Parte de Curso e Condição. O Processo é fundamental no BPDM, pois é o elemento que agrupa os demais elementos, como Atividades, Controles de Curso e Sucessões. A Parte de Curso é uma generalização dos elementos Atividade e Controle de Curso. Por fim, o elemento Condição aparece no modelo por representar um ponto de conexão entre os aspectos dos processos e de motivação.

A Figura 44 ilustra o modelo do aspecto dos processos.



**Figura 44 - Modelo do aspecto dos processos**

O modelo do aspecto dos processos é criado a partir do BMM. Uma importante colaboração do BMM é apresentar os elementos que possuem vínculo com os aspectos dos processos e de estruturação, conforme ilustrado anteriormente na Figura 22 e Figura 23. Como no presente trabalho o aspecto dos processos é considerado o mais relevante, apenas os elementos do BMM que possuem relacionamentos com os outros dois aspectos serão considerados. O resultado da aplicação deste critério está ilustrado na Figura 45.



**Figura 45 - Modelo do aspecto de motivação organizacional**

Embora o BMM aponte um relacionamento entre Estratégia e Unidade Organizacional, no qual a primeira determina a segunda, o elemento Estratégia não está presente no modelo do aspecto de motivação, contrariando o critério adotado. A razão da remoção deste elemento está no fato de a representação da Estratégia no modelo, como um sub-tipo de Plano de Ação, demandaria a representação da Tática também, sob o risco da Estratégia ser interpretada como único sub-tipo de Plano de Ação. A representação de Estratégia e Tática incluiria todos os sub-tipos de Meio, em detrimento a nenhum sub-tipo de Fim representado no modelo. Novamente, isso poderia gerar uma má interpretação quanto à importância dos Fins em relação aos meios.

Por fim, o aspecto de estruturação organizacional é baseado nos elementos presentes no OSM, como ilustra a Figura 46.

Do modelo do núcleo, todos os elementos são aproveitados para o modelo do aspecto de estruturação, com exceção dos elementos abstratos e do Relacionamento Organizacional, que é representado no modelo como um simples relacionamento da Unidade Organizacional. A estes elementos é acrescentado o elemento Pessoa, extraído do modelo de entidade legal.



**Figura 46 - Modelo do aspecto de estruturação organizacional**

O elemento Organização pode ser removido do modelo, uma vez que uma Unidade Organizacional pode representá-lo, conforme mostrado no modelo de unidade organizacional (Figura 43). Sem o elemento Organização, é possível descartar o elemento Entidade Legal. As propriedades deste elemento podem ser incorporadas na única especialização restante de Entidade Legal: Pessoa.

Os atributos dos elementos incluídos no modelo, como o Cargo para Pessoa, Função para Unidade Organizacional e Tipo de Participação, Requisitos da Posição e Autoridade da Posição para Posição, não são representados no modelo para evitar a poluição visual do mesmo.

Os relacionamentos entre os modelos dos três aspectos são baseados nos relacionamentos apresentados no BMM. Entre os aspectos de motivação e dos processos, Chapin et al (2005) apresentam os seguintes relacionamentos:

- Um processo realiza um plano de ação.
- Uma regra de negócio guia um processo.
- Uma política governa um processo.

Embora o BMM defina estas relações, elas são muito genéricas, pois o BMM não especifica quais elementos do meta-modelo de processos fazem parte destas relações.

De acordo com o BMM, um processo é criado para pôr em prática um plano de ação. Portanto, é razoável assumir que há uma ligação direta entre plano de ação e processo.

Uma política pode não ser um elemento forte o suficiente para demandar a criação de um processo completo, mas pode certamente influenciá-lo, sendo responsável pela existência de algumas atividades e controles de curso.

As regras de negócio, por sua definição simples e prática, estão relacionadas as diversas condições que regem um processo. Esta relação entre condição e regra de negócio é especificada pelo próprio BPDM.

Entre os aspectos de motivação e estruturação, Chapin et al (2005) apresentam os seguintes relacionamentos:

- Uma unidade organizacional define um fim.
- Uma unidade organizacional estabelece um meio.
- Uma estratégia determina a criação de uma unidade organizacional.
- Uma unidade organizacional reconhece um influenciador.
- Uma unidade organizacional faz uma avaliação.

De acordo com o OSM, uma unidade organizacional é um agrupamento de posições, e, conseqüentemente, de pessoas. E, embora a unidade organizacional seja o elemento mais importante do OSM, nenhuma decisão pode ser atribuída a ela, uma vez que ela em si não é uma pessoa. Por isso, quem define um fim, estabelece um meio, reconhece um influenciador e faz avaliações é uma ou mais posições dentro de uma unidade organizacional.

Uma estratégia (plano de ação) determina a criação de uma unidade organizacional por completo, com suas posições e relacionamentos com as demais unidades organizacionais. Portanto, há uma ligação direta entre estes elementos.

Embora os relacionamentos entre os aspectos dos processos e de estruturação não estejam no escopo do BMM, Chapin et al (2005) sugerem que uma unidade organizacional seja responsável por um processo.

Esta ligação de responsabilidade é tão forte que o próprio BPDM inclui um elemento, o papel de execução, que é mais apropriado ao OSM. Para o BPDM, um papel de execução é uma especialização de um papel, e o OSM enxerga uma posição da mesma maneira. Portanto, ao invés de estabelecer uma ligação entre um papel de execução e uma posição, eles serão considerados como um mesmo elemento. Por este elemento ser mais forte no escopo do OSM, a terminologia do OSM será utilizada (Posição).



A Figura 47 representa o modelo do contexto organizacional.

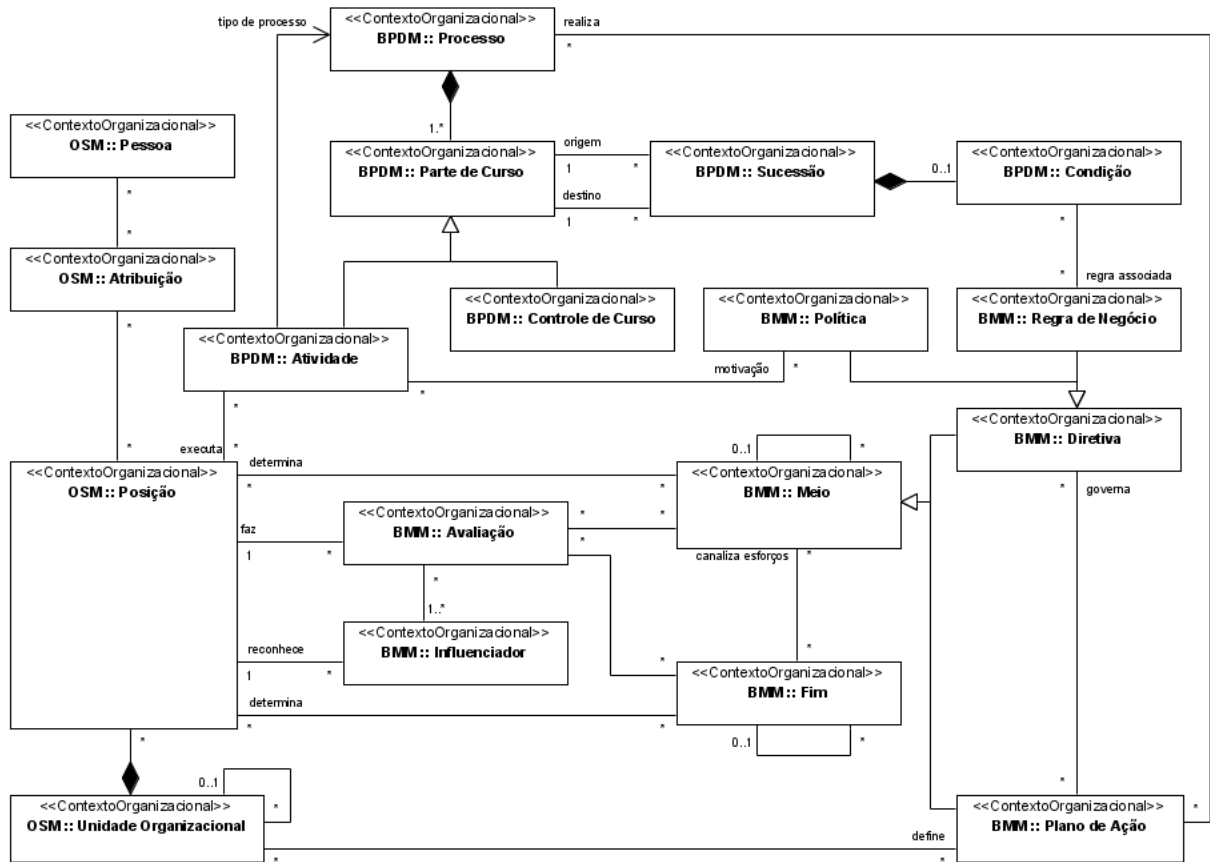


Figura 47 - Modelo do contexto organizacional

## 6.8 Conclusão

Este capítulo apresentou o modelo de contexto organizacional proposto neste trabalho. Embora seja reconhecida a importância dos seis aspectos organizacionais mencionados no QRAC de Zachman, por questões práticas apenas os aspectos dos processos, de motivação e de estruturação organizacional fazem parte do modelo proposto.

O aspecto dos processos foi considerado o aspecto mais relevante, e foi modelado a partir do meta-modelo BPDM e do método-guia IDEF3. O aspecto de motivação organizacional foi modelado a partir do BMM e o aspecto de estruturação organizacional a partir do OSM. Os relacionamentos entre os três aspectos foram criados a partir de indicações contidas no BMM.

## 7 Modelo de Rastreamento de Requisitos

O modelo do contexto organizacional, apresentado no capítulo anterior, representa apenas parte do meta-modelo de rastreamento de requisitos proposto. Para complementá-lo, é necessário definir modelos para a base racional e para o requisito.

Este capítulo apresenta a definição dos modelos que, em conjunto com o modelo do contexto organizacional, formam o modelo de rastreamento de requisitos proposto no presente trabalho.

### 7.1 Base Racional

Moran e Carroll (1996 apud Louridas e Loucopoulos, 2000) definem a base racional para a criação de um artefato como:

- Uma expressão do relacionamento entre o artefato a ser criado, seu propósito, a conceitualização do criador do artefato e os empecilhos em se atingir o propósito.
- O conjunto de razões lógicas que justificam a criação do artefato.
- Uma notação para expressar as razões lógicas para criação do artefato.
- Um método para criação do artefato no qual as razões para criação são apresentadas de forma explícita.
- Documentação das razões para a criação do artefato, das etapas do processo de criação e da história completa da criação do artefato e do contexto em que o artefato foi criado.
- Uma explicação do porquê o artefato é do jeito que é.

O objetivo de registrar a base racional durante o processo de levantamento dos requisitos de *software* é facilitar discussões em grupo, auxiliando projetos cooperativos, prover explicações sobre decisões tomadas e disponibilizar alternativas e argumentos utilizados poupando re-trabalho de análise em futuras revisões dos requisitos.

Diversos autores propõem modelos e métodos para o registro da base racional em um projeto, não necessariamente um projeto de *software*. Entretanto, a natureza dos elementos que representam uma tomada de decisão é suficientemente genérica para ser utilizada como base racional para os requisitos de *software*.

Os tópicos a seguir apresentam o trabalho de quatro autores sobre os modelos de base racional. Esses trabalhos servirão como base para a criação do modelo utilizado no presente trabalho.

### 7.1.1 Meta-modelo IBIS

O IBIS (*Issue-Based Information System*) é considerado o primeiro trabalho na área de registro da base racional, e um dos mais influentes neste campo de pesquisa. A Figura 48 ilustra o modelo proposto por Conklin e Begeman (1998).

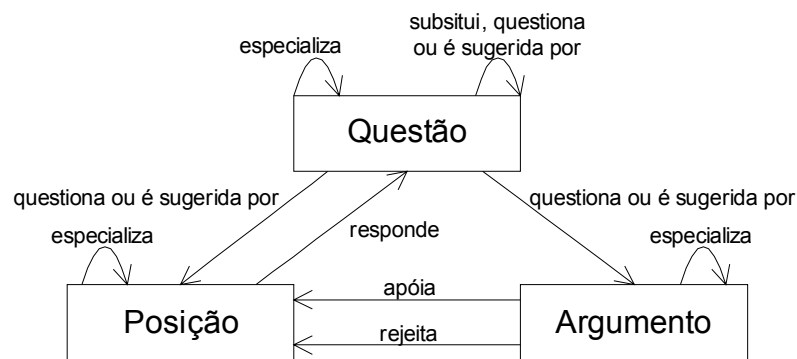


Figura 48 - Meta-modelo IBIS (Conklin, Begeman, 1988)

#### Questão

Representa qualquer coisa que precisa ser discutida durante a criação de um projeto.

Uma questão pode especializar outra questão, detalhando um pouco mais o assunto. Além disso, uma questão pode substituir, questionar a necessidade ou sugerir outra questão.

#### Posição

Representa uma maneira de se resolver uma questão.

Uma posição responde uma questão no sentido de resolvê-la. Além disso, uma posição pode ser originada de outra (especializada) e sua existência pode levar à criação de outras questões.

#### Argumento

Representa uma declaração que apóia ou rejeita uma posição.

Assim como uma posição, um argumento pode dar origem a outro argumento ou levar à criação de uma questão.

### 7.1.2 Modelo de Potts e Bruns

O modelo de Potts e Bruns foi criado para permitir o rastreamento da evolução de um artefato no desenvolvimento de um projeto. A Figura 49 ilustra o modelo de Potts e Bruns (Lee, 1991).

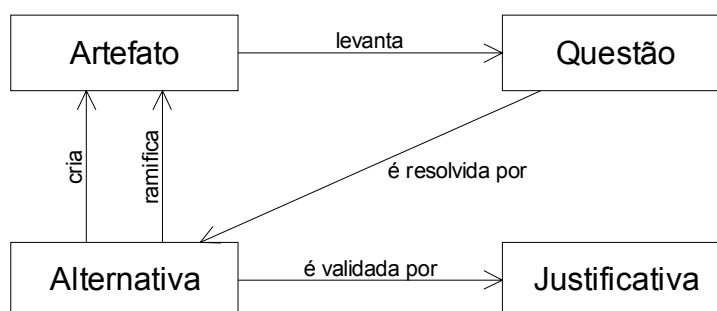


Figura 49 - Modelo de Potts e Bruns (Lee, 1991)

#### Artefato

Um objeto qualquer produzido ao longo do desenvolvimento de um projeto.

#### Questão

Conceito análogo ao introduzido pelo IBIS. Para Potts e Bruns, entretanto, uma questão é sempre levantada a partir de um artefato.

#### Alternativa

Conceito análogo ao de posição introduzido pelo IBIS.

Uma alternativa pode sugerir a criação de um novo artefato, ou a alteração de um artefato existente.

#### Justificativa

Conceito análogo ao de argumento introduzido pelo IBIS.

Uma justificativa valida (ou invalida) uma alternativa proposta.

### 7.1.3 Modelo de Referência para Base Racional

O modelo de referência para a base racional, de Ramesh e Jarke (2001), está ilustrado na Figura 5 do capítulo 3.

O modelo de Ramesh e Jarke representa uma evolução em relação ao IBIS e ao modelo de Potts e Bruns. O conceito de argumento ou justificativa é dividido em dois conceitos, denominados argumento e hipótese. Uma hipótese é uma espécie de argumento mais

genérico, que não está diretamente relacionado a uma alternativa. Ela serve de base para argumentos e também para a tomada de decisões.

A especificação de um elemento para registrar a decisão tomada é o que mais diferencia este modelo dos outros apresentados. Para Ramesh e Jarke, este é o elemento central do modelo, pois é quem resolve os problemas (questões), seleciona alternativas, se baseia em hipóteses, é influenciado por FCSs e afeta os requisitos (artefatos).

#### 7.1.4 Modelo do Laço de Racionalização

O modelo do laço de racionalização não é um modelo estático, como os demais apresentados. Ele é uma mistura de elementos estáticos e elementos dinâmicos, o que o torna mais um processo do que um modelo em si. A Figura 50 ilustra o modelo do laço de racionalização (Louridas, Loucopoulos, 2000).

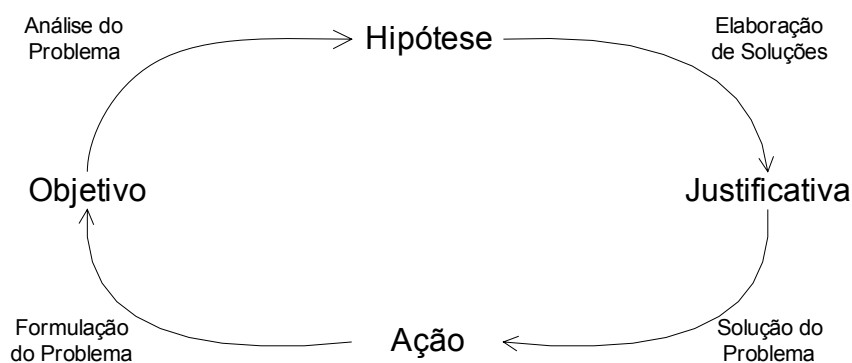


Figura 50 - Modelo do Laço de Racionalização (Louridas, Loucopoulos, 2000)

#### Objetivo

Gerado a partir do processo de formulação do problema, representa um objetivo que deve ser alcançado, uma demanda que deve ser satisfeita, um problema que deve ser resolvido ou uma questão que deve ser discutida.

#### Hipótese

Gerada a partir do processo de análise do problema, representa uma sugestão, proposta ou idéia a respeito da resolução dos problemas levantados.

#### Justificativa

Gerada a partir do processo de elaboração de soluções, representa um argumento para apoiar ou refutar uma hipótese.

## **Ação**

Gerada a partir do processo de solução do problema, representa a ação tomada para atingir o objetivo definido. Esse tipo de ação pode levar à criação de novos artefatos, à solução de problemas ou ao levantamento de novas questões que precisam ser discutidas.

Louridas e Loucopoulos (2000) acreditam que os relacionamentos entre os elementos de seu modelo podem assumir diversas formas, e que não faria sentido limitar as possibilidades em seu modelo. Portanto, não especificam como os elementos se relacionam entre si, apresentando um modelo de relacionamentos livres. Isso permite, entre outras coisas, que elementos de um laço dêem origem a outros laços. Por exemplo, uma hipótese apresentada em um laço pode levantar uma questão, cujo objetivo é determinar se tal hipótese é ou não válida. Um novo laço pode ser criado para aferir a validade da hipótese.

### **7.1.5 Modelo da Base Racional**

Os modelos de base racional apresentados anteriormente possuem muitas similaridades entre si, principalmente em respeito aos elementos que constituem os modelos. Entretanto, nenhum desses modelos é específico para o uso como forma de justificar um requisito. Portanto, é proposto um modelo de base racional para esse fim, utilizando como base os conceitos extraídos dos modelos apresentados.

O primeiro passo para a construção do modelo é determinar seus elementos. Os elementos do modelo da base racional são adaptações dos elementos presentes nos modelos apresentados: questão, alternativa, argumento e decisão.

O elemento questão é utilizado diretamente no IBIS e no modelo de Potts e Bruns, e é utilizado, com outra nomenclatura, nos modelos de Ramesh e Jarke (problema) e do laço de racionalização (objetivo). A função do elemento, portanto, é a mesma utilizada nos modelos apresentados: representar algo que precisa ser discutido e solucionado. Uma questão também pode dar origem a outras questões, conforme argumenta Louridas e Loucopoulos (2000) ou como ilustrado no meta-modelo IBIS (Conklin, Begeman, 1988).

O elemento alternativa cumpre o papel da posição no meta-modelo IBIS, da hipótese no modelo do laço de racionalização e da própria alternativa, nos modelos de Potts e Bruns e de Ramesh e Jarke. A sua função, como nos modelos apresentados, é representar uma alternativa, ou possibilidade de resposta, a uma questão proposta.

Conklin e Begeman (1988) advogam que uma alternativa pode dar origem a outras alternativas. Entretanto, o objetivo do modelo de base racional proposto é servir como uma justificativa para a criação de um requisito. Não há a necessidade de se registrar se uma alternativa surgiu naturalmente em razão de uma questão ou se foi derivada de outra alternativa. Para o fim desejado, basta conhecer quais foram as alternativas propostas para uma questão. Uma questão permite tal derivação pois é importante saber qual a origem de uma questão. Por exemplo, ao rastrear a justificativa para um requisito, pode-se chegar a uma questão que não foi originada por um requisito, e sim por uma outra questão. Mantendo o rastreamento entre as questões, pode-se chegar ao requisito que motivou a questão inicial. Desta forma, fecha-se um ciclo de rastreamento entre dois requisitos, no qual um requisito motivou a criação do outro.

O elemento argumento é o mesmo utilizado no IBIS e, com outra nomenclatura (justificativa), nos modelos de Potts e Bruns e do laço de racionalização. Ramesh e Jarke, além de utilizarem o elemento argumento, adicionaram o elemento hipótese, que funciona como um argumento global que pode afetar alternativas de diversas questões. Novamente, não é o objetivo do modelo proposto representar de forma completa a base racional para qualquer decisão, e sim servir como justificativa para um requisito. Desta forma, o elemento hipótese do modelo de Ramesh e Jarke não é considerado no modelo proposto. Se houver necessidade de utilizá-lo, um argumento pode cumprir esse papel.

Por fim, o elemento decisão é baseado no elemento ação do modelo de laço de racionalização e no próprio elemento decisão presente no modelo de Ramesh e Jarke. A sua função é registrar qual foi a alternativa escolhida e baseada em quais argumentos. Representa, portanto, qual a decisão tomada e qual curso de ação será seguido, solucionando uma questão proposta.

O segundo passo para a criação do modelo é determinar quais são os relacionamentos entre os elementos definidos. Louridas e Loucopoulos (2000) sugerem o uso de relacionamentos livres, devido à natureza mais complexa das relações entre os elementos da base racional. Embora isso permita a representação mais completa de uma base racional, uma vez que quaisquer relacionamentos são permitidos, o modelo perde em organização, e a possível existência de qualquer tipo de relacionamento pode tornar confusas a criação e leitura de um modelo. Como o objetivo não é a criação de uma base racional genérica, e sim de uma específica para o rastreamento de requisitos, os relacionamentos entre os elementos serão pré-determinados.

Os critérios para definição dos relacionamentos utilizados no modelo seguem os métodos de criação da base racional sugeridos por Louridas e Loucopoulos (2000) e os interesses de rastreamento de um requisito até ela. Na criação de uma base racional, um requisito pode motivar uma questão, à qual são apresentadas alternativas. Cada alternativa é apoiada ou rejeitada por argumentos, e uma decisão final resolve a questão e pode levar à criação de outro requisito. Em termos do interesse do rastreamento, um requisito é rastreado a uma decisão tomada, que deve representar a solução para uma questão e referenciar a alternativa adotada (ou alternativas, quando se complementam) e os argumentos que corroboram a decisão.

A Figura 51 ilustra a proposta do presente trabalho para o modelo da base racional

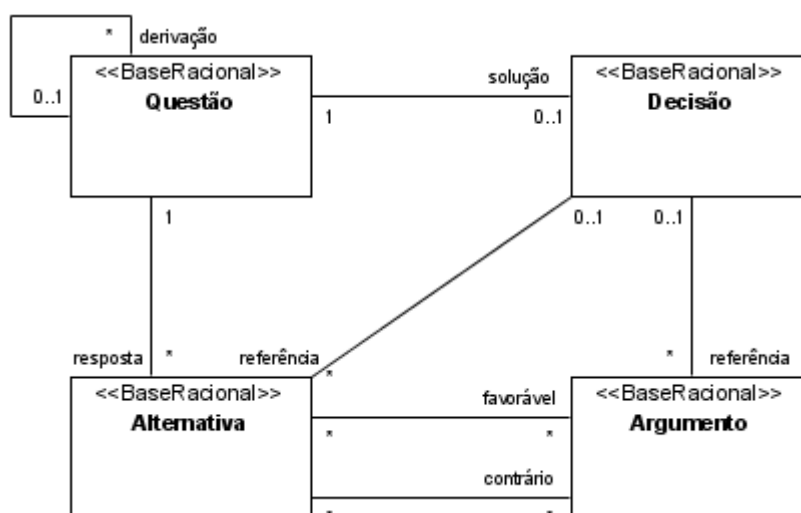


Figura 51 - Modelo da base racional

## 7.2 Requisito

Três elementos são importantes para representar os requisitos, de acordo com Leffingwell e Widrig (2003): necessidades dos envolvidos, características de *software* e requisitos de *software*. Cada um destes elementos representa um tipo de requisito, em domínios ou níveis de detalhamento diferentes.

Uma **necessidade** representa um desejo, por parte de membros da organização, em se resolver um problema ou aprimorar um aspecto da organização. Nesta etapa do processo, o requisito se encontra no domínio do problema e seu detalhamento é mínimo, se comparado ao detalhamento necessário para se desenvolver um *software*. Em alguns casos, a necessidade está muito relacionada a um problema, e não há, a princípio, uma idéia muito clara de como



resolver o problema. Em outros casos, a necessidade já está melhor formulada, e já se possui um esboço de como seria um *software* que resolvesse o problema.

Leffingwell e Widrig (2003) apresentam alguns passos da etapa de análise de problema, que resulta no levantamento das necessidades organizacionais. As duas primeiras etapas são: (i) chegar em um acordo sobre a definição do problema e (ii) identificar as causas do problema. Esta preocupação em determinar as causas de um problema é refletida no presente trabalho por meio da criação de um quarto elemento do tipo requisito, denominado de **problema**. Um problema contém uma declaração de algo que é percebido como ruim pelos envolvidos no projeto, e que pode ser melhorado ou solucionado com a implantação de um *software*. Além disso, um problema representa a motivação para a especificação de uma necessidade.

O conjunto de problemas e necessidades possui fortes relações com o contexto organizacional, uma vez que estes elementos pertencem ao mesmo domínio, o domínio do problema. Normalmente, um problema é motivado por elementos do contexto organizacional que possuem deficiências, e a definição do problema explicita tais deficiências. Por sua vez, uma necessidade, motivada por um problema, tem como objetivo produzir um novo e melhorado contexto organizacional. Com o uso destes três relacionamentos, pode-se considerar o conjunto de necessidades como uma representação das diferenças entre o contexto organizacional desejado e o contexto organizacional com problemas.

Uma **característica de *software*** possui um nível de detalhamento apenas um pouco maior que o de uma necessidade. A diferença entre eles está no domínio em que a característica de *software* se encontra. Enquanto uma necessidade é elaborada por um membro da organização e está atrelada ao domínio do problema, uma característica de *software* é elaborada pelos responsáveis pela especificação final do *software*, e está atrelada ao domínio da solução.

Os **requisitos de *software***, além de pertencerem exclusivamente ao domínio da solução, possuem um nível de detalhamento muito maior que o dos outros elementos. Desta forma, os requisitos de *software* possuem informação suficiente para o desenvolvimento do *software*.

Spence e Probasco (2000) dividem os requisitos de *software* em dois grupos: os requisitos funcionais e os não-funcionais. Os não-funcionais são normalmente registrados em forma de texto, mas os funcionais podem ser registrados, além da forma textual, na forma de casos de uso. Spence e Probasco denotam ainda dois elementos importantes de um caso de uso: seção do caso de uso e ator. Uma seção de caso de uso pode cumprir diversos papéis em um caso de uso. Pode representar tópicos do caso de uso, como a seção de pré-condições,

funcionalidades, cenários (sucesso, alternativo ou exceção) ou etapas de uma funcionalidade. Da mesma forma que um caso de uso pode possuir diversas seções, uma seção pode também possuir sub-seções. Por fim, dependendo da granularidade da seção de caso de uso, um ator pode ser o responsável por executá-la.

Cada um dos três principais elementos do tipo requisito pode apresentar derivações que pertencem ao mesmo domínio e nível de detalhamento. Por exemplo, uma necessidade mais genérica pode ser detalhada por outras necessidades, facilitando o seu entendimento. E como afirmam Leffingwell e Widrig (2003), Spence e Probasco (2000) e Ramesh e Jarke (2001), os três elementos possuem relacionamentos entre si. Uma necessidade é refinada em características de *software*, que por sua vez são refinadas em requisitos de *software*. Uma seção de caso de uso também pode cumprir o papel de um requisito de *software*, e por esta razão, também pode representar o refinamento de uma característica. Em uma de suas estratégias de rastreamento, Spence e Probasco (2000) mostram um relacionamento direto de refinamento entre necessidades e características de *software*. Entretanto, assim como o fazem Spence e Probasco, neste trabalho tal relacionamento não será recomendado.

A Figura 52 ilustra o modelo proposto para os requisitos.

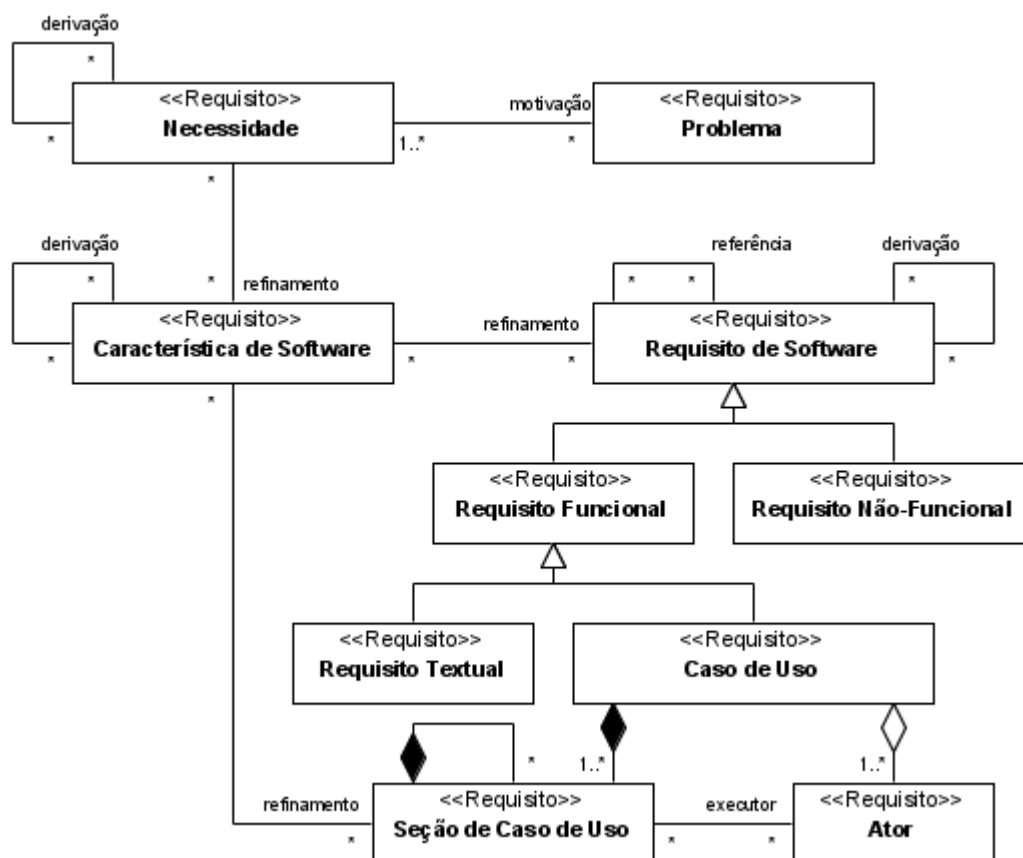


Figura 52 - Modelo de requisitos

### 7.3 Integração entre os modelos

O meta-modelo proposto prevê 7 relacionamentos de rastreamento envolvendo as metaclasses ContextoOrganizacional, BaseRacional e Requisito. Ao definir um modelo instanciado a partir desse meta-modelo, é preciso identificar como esses relacionamentos se aplicam.

- Rastreamento entre contexto do problema e contexto desejado

O meta-modelo permite que elementos do tipo ContextoOrganizacional possam ser rastreados a outros elementos também do tipo ContextoOrganizacional. Embora o modelo do contexto organizacional não apresente estes relacionamentos, pois a representação visual ficaria poluída, o modelo proposto os inclui. Entretanto, há uma restrição: os elementos conectados por este tipo de rastreamento devem ser sempre do mesmo tipo. Por exemplo, uma atividade desejada pode ser rastreada até uma atividade com problema, mas não a uma regra de negócio. A Figura 53 ilustra este tipo de rastreamento.

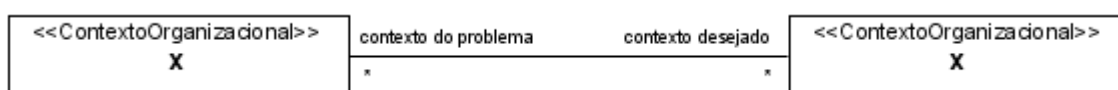


Figura 53 - Rastreamento entre contexto com problema e contexto desejado

- Contexto organizacional como referência para a base racional

A relação é mantida para todos os elementos do contexto organizacional e da base racional, como ilustra a Figura 54.

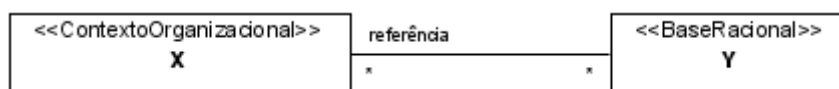


Figura 54 - Rastreamento do tipo referência entre ContextoOrganizacional e BaseRacional

- Contexto organizacional como referência para o requisito

A relação é mantida para todos os elementos do contexto organizacional e do requisito, como ilustra a Figura 55.

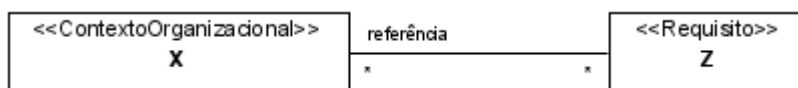
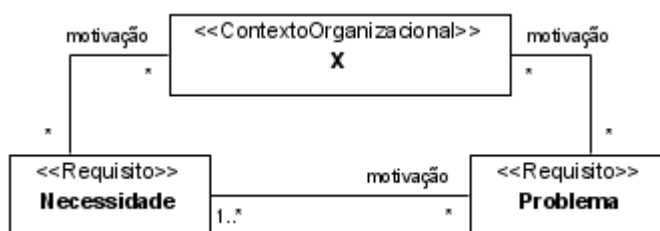


Figura 55 - Rastreamento do tipo referência entre ContextoOrganizacional e Requisito

- Contexto organizacional como motivação para o requisito

Para manter as necessidades como fontes das características de *software* e dos requisitos de *software*, apenas os problemas e necessidades podem ser motivadas por um elemento do tipo

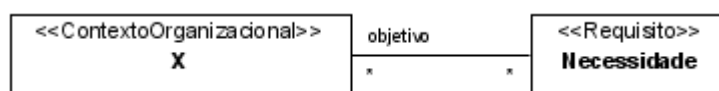
ContextoOrganizacional. A Figura 56 ilustra os relacionamentos de motivação permitidos entre contexto organizacional e requisito.



**Figura 56 - Rastreamento do tipo motivação entre ContextoOrganizacional e Requisito**

- Contexto organizacional como objetivo do requisito

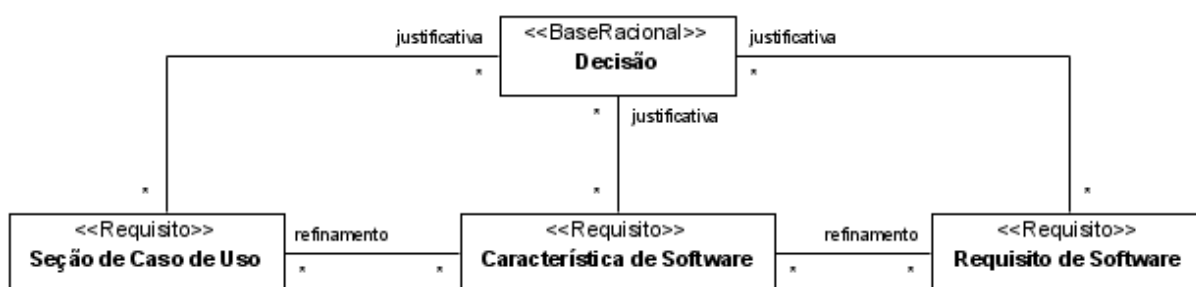
Para manter as necessidades como uma representação da diferença entre o contexto desejado e o contexto com problema, apenas as necessidades terão como objetivo um elemento do tipo ContextoOrganizacional. A Figura 57 ilustra os relacionamentos de objetivo entre contexto organizacional e requisito.



**Figura 57 - Rastreamento do tipo objetivo entre ContextoOrganizacional e Requisito**

- Base Racional como justificativa do requisito

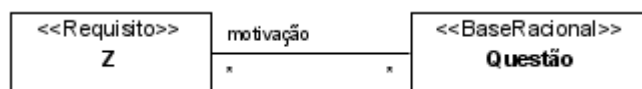
Uma decisão tomada após a discussão de uma questão pode levar à criação de novos requisitos. Entretanto, as discussões têm como objetivo determinar soluções, e não problemas e necessidades. Desta forma, apenas os tipos de requisito que representam uma solução são relacionados à base racional, conforme ilustra a Figura 58.



**Figura 58 - Relacionamento do tipo justificativa entre BaseRacional e Requisito**

- Requisito como motivação para a base racional

O processo de elaboração de uma base racional deve começar com o surgimento de uma questão, normalmente motivada por um requisito. Portanto, apenas essa relação será incluída no modelo de rastreamento, conforme ilustra a Figura 59.



**Figura 59 - Rastreamento do tipo motivação entre Requisito e BaseRacional**

#### **7.4 Conclusão**

Este capítulo apresentou os modelos propostos para a base racional e para os requisitos, que, em conjunto com o modelo do contexto organizacional, representam o modelo completo de rastreamento de requisitos proposto no presente trabalho. Além dos modelos apresentados, os relacionamentos de rastreamento entre elementos de diferentes modelos foram especificados.

## **8 Exemplo de Aplicação do Modelo**

O conjunto dos modelos definidos nos capítulos 7 (modelo do contexto organizacional) e 8 (modelos de base racional e de requisito) forma o modelo completo de rastreamento de requisitos proposto no presente trabalho. Este capítulo apresenta um exemplo de utilização deste modelo de rastreamento.

### **8.1 Descrição do Exemplo**

O exemplo apresentado se baseia em um projeto denominado PR (Previsão de Reembolso) realizado em 2004 para a organização ABC, que atua no ramo da saúde. Mais especificamente, a organização possui uma rede de farmácias e também vende planos de saúde, principalmente para funcionários de outras organizações.

A ABC notou um aumento significativo em seus gastos com hospitais e médicos, proveniente da utilização dos planos de saúde que vende. Percebeu-se que muitos dos problemas que acarretavam em intervenções médicas mais custosas, como operações ou longas estadias em hospitais, surgiam do desleixo dos pacientes em tomar suas medicações corretamente. Esses pacientes, ao sinal de problemas de saúde, utilizavam seus planos de saúde em consultas médicas mas não seguiam a receita indicada pelo médico, considerando o custo dos medicamentos muito alto. A doença se agravava e o paciente requeria cuidados extras.

Uma das formas de reduzir esses gastos foi subsidiar a compra de medicamentos por parte dos clientes, incentivando-os a cumprir a receita do médico e evitando que as doenças se agravassem. Esse subsídio para compra de medicamentos se deu pela criação de planos de farmácia.

O tipo mais comum de plano de farmácia envolvia a divisão dos medicamentos em três grupos: remédios genéricos, medicamentos presentes na lista de medicamentos preferenciais e medicamentos não presentes na lista. A lista de medicamentos preferenciais era criada por um setor da ABC e contava com pelo menos um medicamento para cada classe de doença (controle de colesterol, dores musculares, etc...). Desta forma, um cliente poderia utilizar apenas os remédios presentes na lista sem correr riscos à sua saúde.

Ao preencher uma receita médica utilizando o plano de farmácia, o usuário do plano pagava apenas uma pequena taxa, denominada franquia, que variava de acordo com a presença do medicamento comprado em um dos três grupos definidos. Medicamentos genéricos possuíam

a menor franquia, seguidos pelos medicamentos presentes na lista preferencial e por fim, pelos remédios fora da lista

A farmácia que efetuava a venda com o desconto (pois só recebia o valor da franquia), comunicava diariamente a ABC, que consolidava os valores informados e reembolsava a farmácia mensalmente. Para auxiliar nos gastos extras com reembolso às farmácias, a ABC criava contratos de reembolso com os fabricantes dos medicamentos. Como a maioria dos usuários optava pelos medicamentos de marca presentes na lista preferencial, era vantajoso para os fabricantes de medicamentos incluir os seus medicamentos nessa lista. Portanto, os contratos estipulavam que os fabricantes deveriam reembolsar a ABC pelas vendas de seus medicamentos atrelados aos planos de farmácia.

Os contratos possuíam diversas cláusulas, que atrelavam o reembolso à venda dos medicamentos. O reembolso podia estar relacionado à venda absoluta de medicamentos (por exemplo, \$1 por medicamento vendido) ou à participação de mercado do medicamento em sua classe de doença. Outro fator importante era a exclusividade do medicamento na lista preferencial. Colocar na lista preferencial mais de um medicamento da mesma classe de doença permitia criar contratos com mais de um fabricante, embora os reembolsos seriam menores por contrato.

Em resumo, a ABC gastava dinheiro ao reembolsar as farmácias e ganhava dinheiro com a venda dos planos, com a redução dos gastos com os planos de saúde, com descontos negociados com as farmácias e com os reembolsos dos fabricantes.

Para controlar todas estas informações, a ABC criou um sistema de informação denominado GPF (Gerenciamento de Planos de Farmácia). Por meio desse *software*, a ABC manipulava sua lista de medicamentos preferenciais, armazenava os registros de venda de medicamentos, cadastrava os contratos de reembolso com os fabricantes e efetuava os cálculos de pagamento às farmácias e de reembolso dos fabricantes.

Outras organizações passaram a atuar nesse mercado, limitando os ganhos da ABC. Neste cenário, a ABC concluiu ser necessário reduzir mais os seus custos operacionais. Uma das maneiras encontradas foi aumentar o reembolso dos fabricantes de medicamento. Para isso, bastava melhorar o processo de criação dos contratos com os fabricantes, maximizando o reembolso. O setor responsável pela criação dos contratos possuía a experiência de mercado necessária para escrever bons contratos, mas dependiam de análises efetuadas com base em

dados históricos e que exigiam muito esforço manual. Desta forma, os contratos elaborados não eram os mais eficientes, prejudicando a maximização do reembolso.

Para solucionar esta questão, a ABC encomendou o projeto PR que permitiria cadastrar contratos experimentais, projetar registros de venda no futuro e calcular de forma automatizada os reembolsos dos fabricantes utilizando estas informações. Isso permitiria à ABC simular diversos contratos e situações do mercado no futuro para prever qual seria o contrato ideal.

## **8.2 Contexto Organizacional do Problema**

Ao iniciar o projeto, e como meio de garantir a rastreabilidade dos requisitos aos elementos que lhes deram origem, é preciso registrar o contexto organizacional no qual há um problema ou oportunidade de melhoria.

Os tópicos a seguir descrevem a estrutura organizacional, os meios e fins relevantes e os processos que pertencem ao contexto do problema. Por fim, esses elementos são organizados em um modelo conforme o modelo de contexto organizacional proposto.

### **Estrutura Organizacional**

Três unidades organizacionais apresentam alguma relevância para o problema. O departamento denominado *Industry Relations* (IR) é responsável pela negociação dos contratos de reembolso com os fabricantes. O diretor deste departamento é o principal responsável e principal usuário do sistema a ser criado. Na época da elaboração dos requisitos, quem atuava na posição de diretor de IR era John Smith, a partir do qual grande parte dos requisitos foi obtida.

Outro departamento importante é o *Health Outcomes* (HO), responsável pela venda de planos, negociações com as farmácias e utilização do sistema GPF. Duas posições são relevantes no departamento HO: o diretor responsável pelas negociações com as farmácias e pela criação do GPF e os operadores do GPF, que recebem os registros de venda e processam os cálculos de reembolso. O diretor de HO, na pessoa de David Lee, auxiliou o processo de levantamento de requisitos por sua experiência na criação do GPF.

Por último, estão as farmácias que aceitam os planos de farmácia da ABC. Os gerentes dessas farmácias são os responsáveis por consolidar os dados de venda de medicamentos por meio dos planos de farmácia e de enviá-los a ABC.

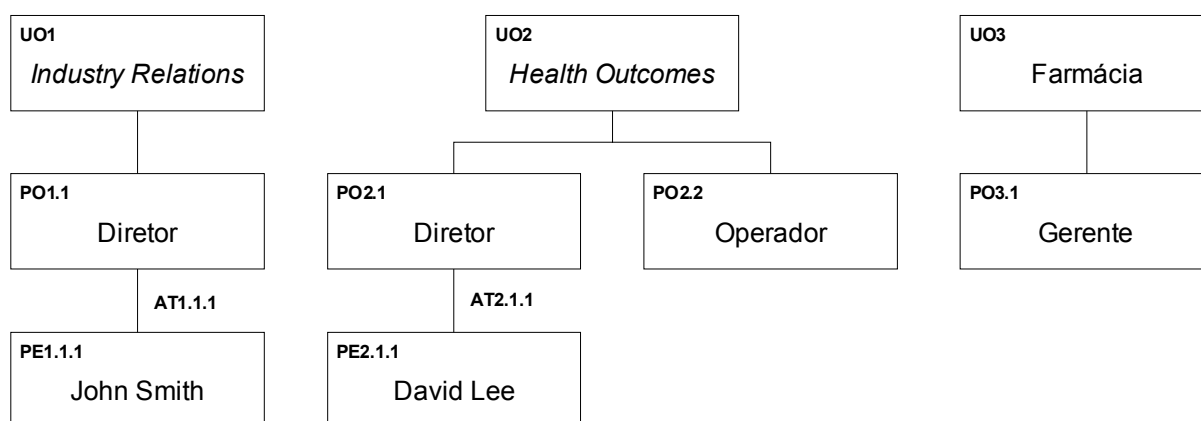


A Tabela 6 resume a estrutura organizacional da ABC, relevante ao projeto PR. A sigla UO identifica unidade organizacional, PO, posição, AT, atribuição e PE, pessoa.

**Tabela 6 - Resumo da estrutura organizacional da ABC**

<b>Unidade Organizacional</b>	<b>Posição</b>	<b>Atribuição</b>	<b>Pessoa</b>
UO1 - <i>Industry Relations</i>	PO1.1 - Diretor	AT1.1.1 - 2004	PE1.1.1 - John Smith
UO2 - <i>Health Outcomes</i>	PO2.1 - Diretor	AT2.1.1 - 2004	PE2.1.1 - David Lee
	PO2.2 - Operador		
UO3 - Farmácia	PO3.1 - Gerente		

A Figura 60 ilustra o modelo da estrutura organizacional. Devido à natureza do elemento atribuição, ele é representado no modelo por um relacionamento, e não por uma entidade. Isso torna a leitura do modelo mais simples e evita a poluição gráfica do mesmo.



**Figura 60 - Modelo de estrutura organizacional da ABC**

### **Motivação Organizacional**

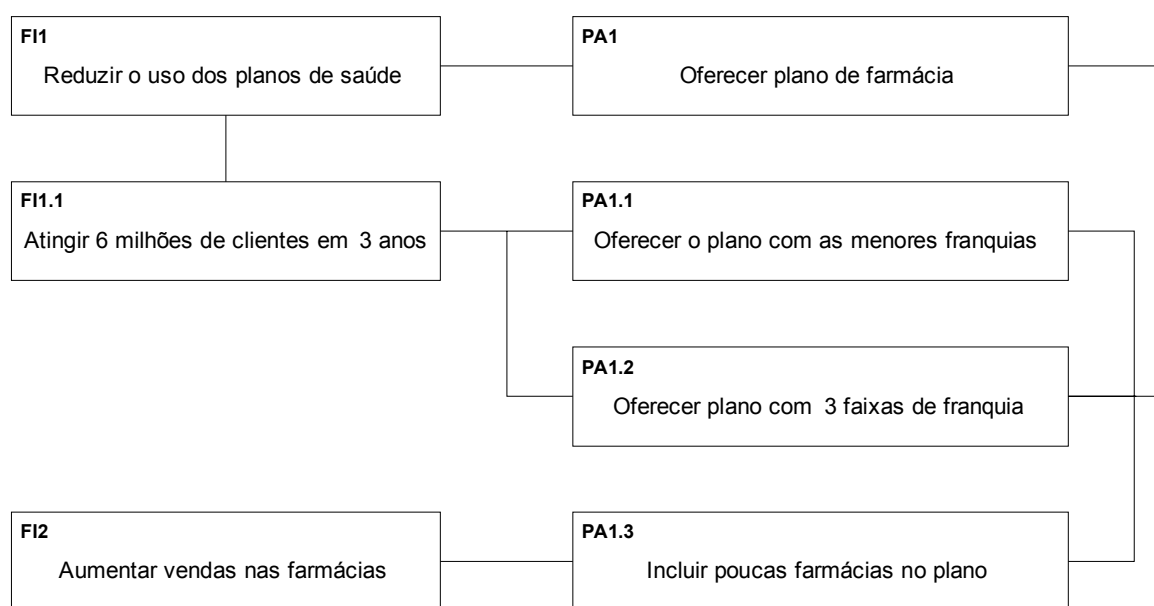
O que motivou a criação do sistema GPF e da contratação do projeto PR foi, em última análise, o objetivo da ABC de reduzir o uso dos planos de saúde pelos clientes. Para atingir este fim, a ABC passou a comercializar planos de farmácia. Para que a redução no uso dos planos de saúde fosse significativa, era importante que a maioria dos clientes de plano de saúde optasse também pelo plano de farmácia. Portanto, a ABC estipulou uma meta interna de atingir 6 milhões de clientes em um prazo de 3 anos.

Para conquistar tantos clientes, a ABC optou por oferecer um produto com o melhor benefício para o cliente. Desta forma, optou por oferecer planos com as franquias mais baixas do mercado e com as três faixas de medicamentos (genérico, presente na lista e ausente da lista), de forma que o cliente pudesse economizar ao máximo em suas compras, tornando o plano muito interessante.

Como a ABC possui uma forte rede de farmácias, aproveitou para estimular o aumento das vendas nessa rede. Para isso, evitou negociar com muitas outras redes para a aceitação dos

planos. Embora a maior exclusividade de farmácias não seja boa para o cliente, que tem opções reduzidas para a compra de seus medicamentos, ela fortalece a própria rede de farmácias e permite negociar maiores descontos com as outras redes de farmácia incluídas no plano. Além disso, por contar com um produto de ótimo benefício, a ABC conclui que a exclusividade de farmácias não prejudicaria o plano de atingir 6 milhões de clientes.

A Figura 61 ilustra o modelo de motivação organizacional da ABC, que resultou na criação dos planos de farmácia. O modelo relaciona os planos de ação com os fins para os quais canalizam esforços. Além disso, mostra a decomposição de fins e meios em outros fins e meios mais específicos. A sigla FI significa fim e PA, plano de ação.



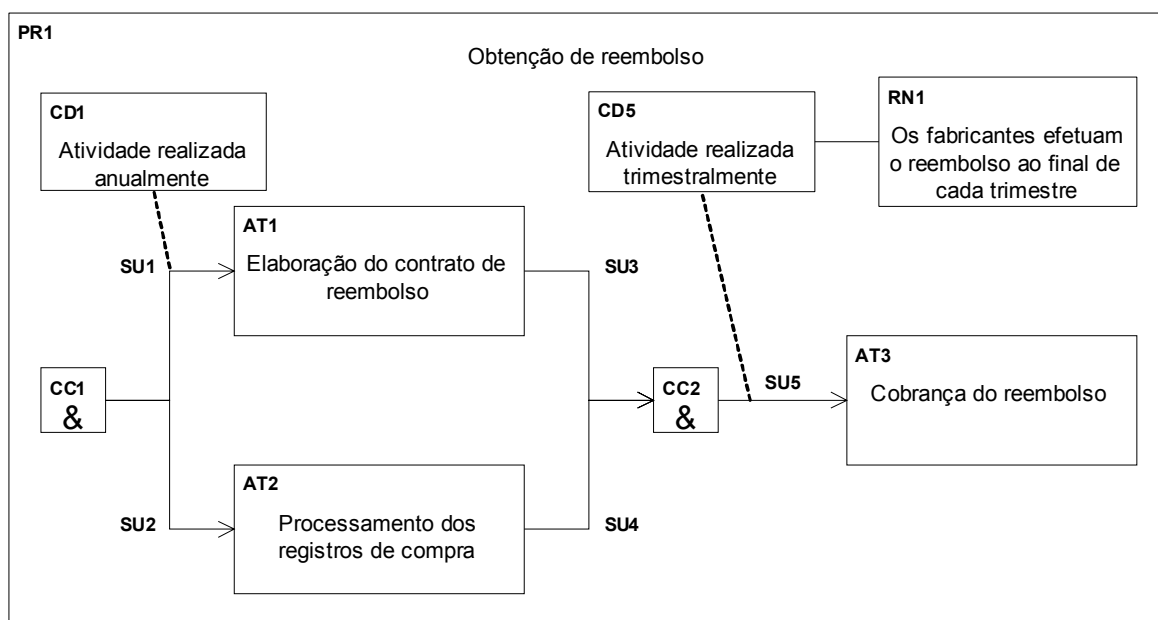
**Figura 61 - Modelo de motivação organizacional da ABC**

### **Processos de negócio**

É possível destacar três importantes atividades executadas na ABC em relação ao processo de obtenção de reembolso do fabricante de medicamentos: realizar a atividade de negociação com o fabricante, resultando na criação de um contrato de reembolso, receber e processar os registros de vendas de medicamentos nas farmácias participantes e calcular e emitir a fatura para recebimento do reembolso.

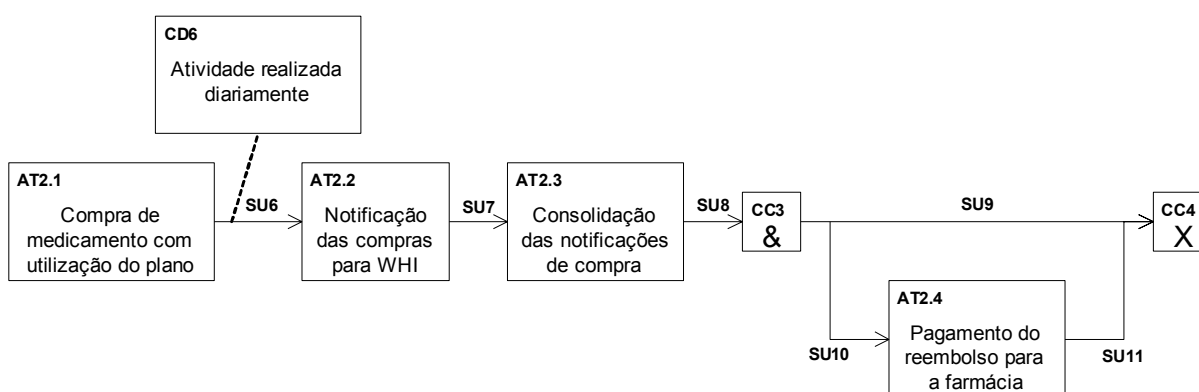
A Figura 62 ilustra o modelo de processos relevantes para o projeto PR. As atividades são identificadas no modelo pela sigla AT e os processos por PR. Os controles de curso (CC) do tipo junção E são indicados pelo símbolo &. Pela natureza do elemento sucessão (SU), ele é representado por um relacionamento navegável, o que indica qual é o elemento predecessor e sucessor. Arelado às sucessões, podem haver condições (CD), cujo relacionamento é

representado por uma linha tracejada. Por fim, o modelo traz uma regra de negócio (RN), ilustrando sua relação com uma condição. Embora pertença ao modelo de motivação organizacional, a RN1 foi ilustrada no modelo de processos devido a sua forte ligação com a condição CD5.



**Figura 62 - Modelo de processos da ABC**

Como uma atividade pode cumprir o papel de um processo, é possível detalhar uma atividade em suas sub-atividades. O detalhamento de AT2 está ilustrado na Figura 63.

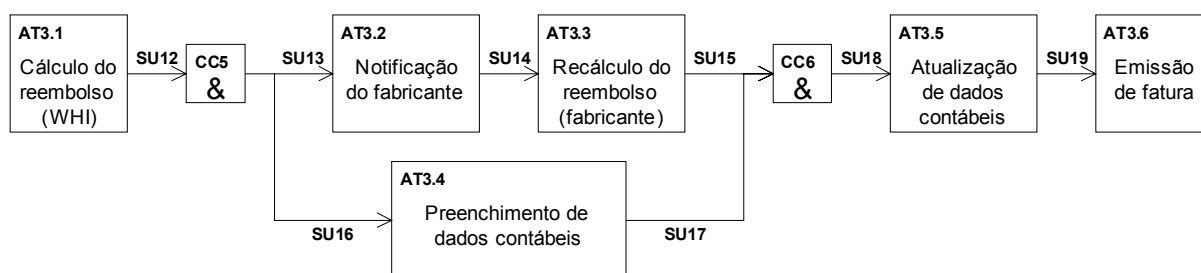


**Figura 63 - Modelo de sub-atividades de AT2**

Após pelo menos uma compra de medicamento (AT2.1), a farmácia consolida as informações e notifica a ABC no final de cada dia (AT2.2). A ABC consolida as informações recebidas das diversas farmácias (AT2.3) e efetua o pagamento de reembolso (AT2.4). O controle de curso CC3 é uma junção do tipo E, que dispara as sucessões SU9 e SU10 em paralelo. O controle de curso CC4 é uma junção do tipo OU, que permite a continuidade do processo assim que pelo menos uma das sucessões (SU9 ou SU11) ocorra. Na prática, esta construção

do modelo permite que a atividade AT2.4 seja executada em paralelo e de forma assíncrona em relação ao restante do processo de reembolso (AT3).

A Figura 64 ilustra o detalhamento da atividade AT3. Em um primeiro momento, a ABC efetua um cálculo do valor do reembolso (AT3.1), utilizando o sistema GPF. Os valores são preenchidos no sistema de contabilidade (AT3.4). Em paralelo a esta atividade, a ABC notifica o fabricante (AT3.2), que recalcula o valor do reembolso, informa a ABC e autoriza a emissão de fatura (AT3.3). Com os dados recebidos do fabricante, a ABC reconcilia as informações contábeis (AT3.5) e emite a fatura para o reembolso (AT3.6).

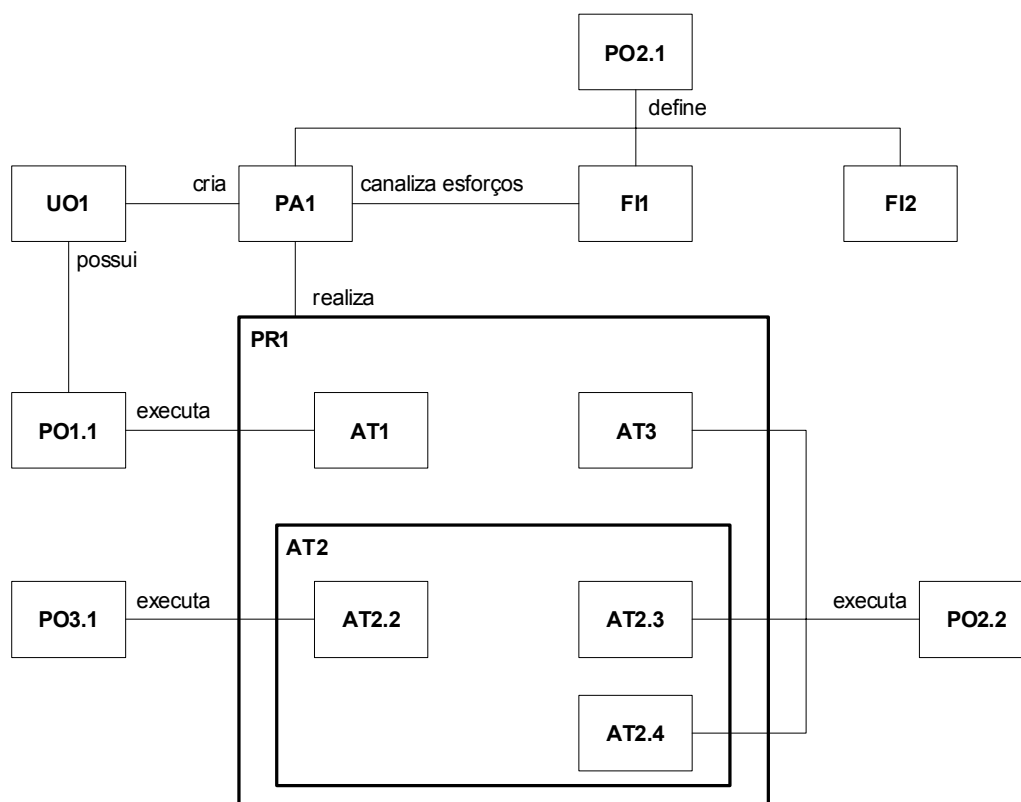


**Figura 64 - Modelo de sub-atividades de AT3**

### **Modelo do Contexto Organizacional**

A representação completa do contexto organizacional com problemas demanda a junção dos modelos dos três diferentes aspectos (estrutura organizacional, motivação organizacional e processos de negócio). A representação gráfica dos três aspectos em um único modelo resultaria em um modelo de leitura complexa. Entretanto, é preciso identificar os relacionamentos entre os elementos dos diferentes aspectos. A Figura 65 ilustra estes relacionamentos. Nesta representação, estão incluídos apenas os elementos que possuem relacionamentos com elementos de outros aspectos. Alguns elementos de um mesmo aspecto e que possuem relacionamentos diretos entre si estão presentes na representação gráfica. Nestes casos, os seus relacionamentos, embora intra-aspectos, também são ilustrados, de forma que o modelo não induza a nenhuma dúvida ou ambigüidade.

Os fins FI1 e FI2 e o plano de ação PA1 são definidos pela posição PO2.1, que representa o diretor do departamento HO. O plano de ação PA1 induz à criação do departamento IR, representado pela unidade organizacional UO1 e é realizado, em parte, pelo processo de obtenção de reembolso, PR1.



**Figura 65 - Relacionamentos entre os aspectos do contexto organizacional**

O diretor do departamento IR, PO1.1, executa a atividade de elaboração do contrato de reembolso, AT1. O gerente de farmácia, PO3.1, executa a atividade de notificação das compras para ABC, AT2.2. Por fim, o operador do departamento HO, PO2.2, executa as atividades de cobrança do reembolso, AT3, consolidação das notificações de compra, AT2.3, e pagamento do reembolso à farmácia, AT2.4.

### 8.3 Contexto Organizacional Desejado

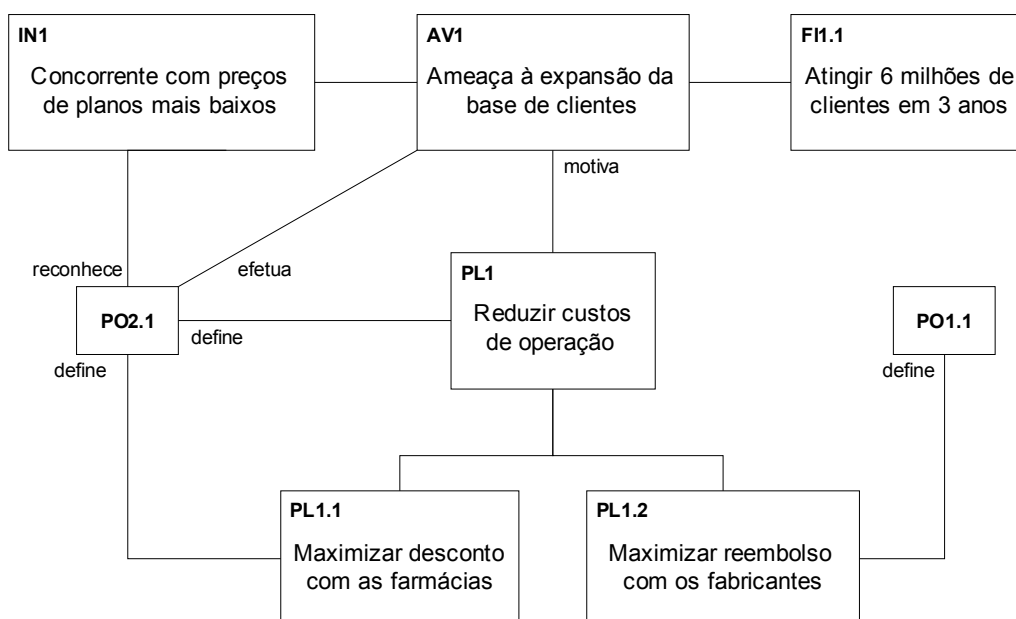
Após um tempo de atuação da ABC como vendedora de planos de farmácia, surgiu um concorrente no setor. O então diretor de HO, David Lee, avaliou esse novo concorrente como uma ameaça à meta de atingir 6 milhões de clientes em 3 anos, uma vez que o concorrente vendia planos mais baratos.

Como resposta a esta ameaça, o diretor de HO instituiu uma política de redução dos custos de operação. Como parte desta política, determinou uma outra política, que visava a maximizar os descontos obtidos pela ABC nas redes de farmácia participantes. Como a ABC controla uma dessas redes, ela poderia forçar as outras a se adequarem, sob o medo de perder uma fatia potencial de 6 milhões de clientes.

O diretor de IR, John Smith, adotou a política de redução de custos por meio de uma outra política, que visava a maximização dos reembolsos com fabricantes. Notando que os maiores pagadores eram os fabricantes mais fortes no mercado, o diretor de IR se viu limitado na sua tentativa de exigir maiores valores por contrato. Portanto, optou por melhorar a eficiência dos contratos que criava.

Para obter o melhor contrato possível, John Smith imaginou um processo no qual poderia cadastrar em um sistema contratos experimentais e poderia, usando seus conhecimentos do mercado, prever os registros de utilização do plano (vendas de medicamento). Com estas duas informações, poderia rodar simulações de reembolso que o ajudariam a decidir pelo melhor contrato.

A Figura 66 ilustra as novas motivações da ABC em resposta a uma ameaça às suas metas. A sigla IN identifica um influenciador, AV uma avaliação e PL uma política. As posições responsáveis por reconhecer um influenciador, efetuar uma avaliação e definir novas políticas também estão presentes no modelo.



**Figura 66 - Modelo de motivação para alteração do contexto organizacional**

A Figura 67 ilustra o modelo desejado para os novos processos da ABC. A atividade AT1, que consistia na elaboração de um contrato de reembolso, foi substituída pela AT1\*, que visa a elaborar o melhor contrato possível. O modelo ainda ilustra o fato de a política PL1.2 (maximizar reembolso com os fabricantes) motivar a criação da atividade AT1\* (o símbolo \* denota uma nova versão do elemento). Por fim, mostra também o relacionamento de

rastreamento entre um elemento do contexto do problema (AT1) e um elemento do contexto desejado (AT1\*).

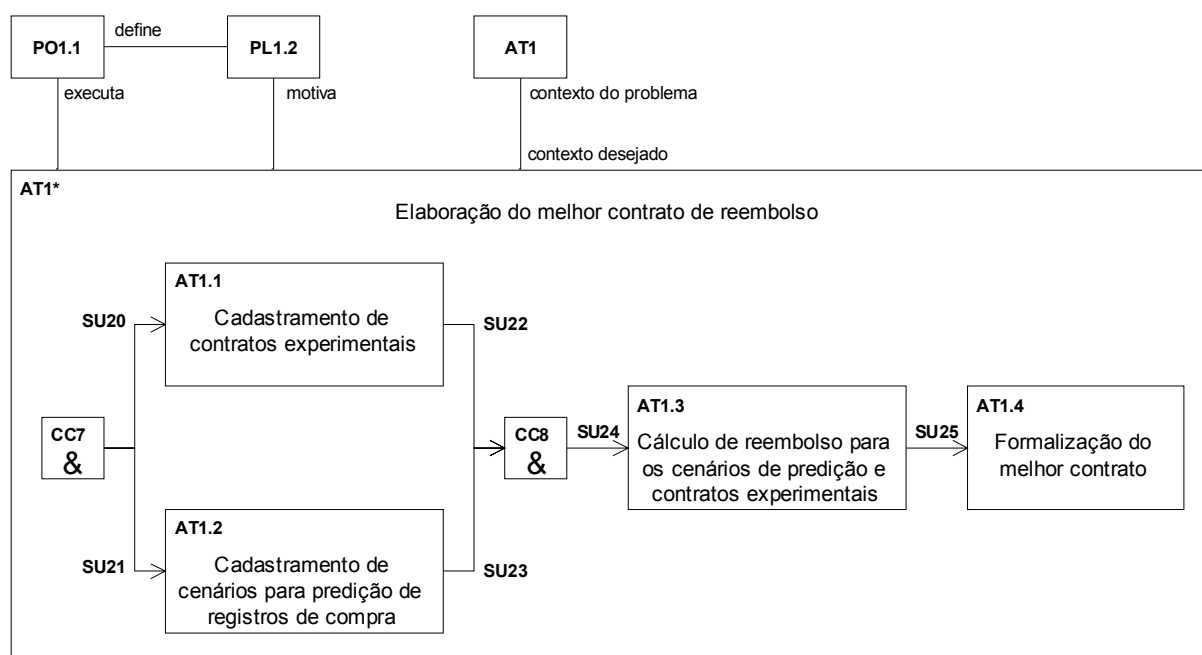


Figura 67 - Modelo desejado de processos de negócio

#### 8.4 Necessidades dos Envolvidos

O principal envolvido no projeto PR é John Smith, diretor de IR, e sua principal necessidade é aprimorar o processo de elaboração dos contratos de reembolso. A razão desta necessidade está no fato de o processo atual de elaboração de contratos (AT1) não produzir os melhores contratos, resultando na perda de potenciais valores de reembolso. Para atender à sua necessidade, John Smith espera que seja criado um sistema que lhe permita efetuar cálculos de reembolso para períodos no futuro (AT1\*).

Para aprimorar a elaboração de contratos, o diretor de IR precisa compreender melhor o comportamento dos reembolsos no tempo. Para isso, John Smith tem a necessidade de simular reembolsos para cenários de mercado no futuro de forma rápida e automatizada, uma vez que a atividade atual de simulação demanda muito tempo e esforço

Como forma de simular reembolsos futuros, John Smith espera a criação de um sistema que lhe permita criar contratos experimentais (AT1.1), cadastrar cenários para predição de registros de compra de medicamento (AT1.2) e calcular os reembolsos usando os contratos experimentais e os cenários de predição (AT1.3).

Por fim, John Smith detalha os fatores de interesse para um cenário de predição de mercado futuro. É preciso modelar um fator de acréscimo da utilização dos planos (devido à entrada de

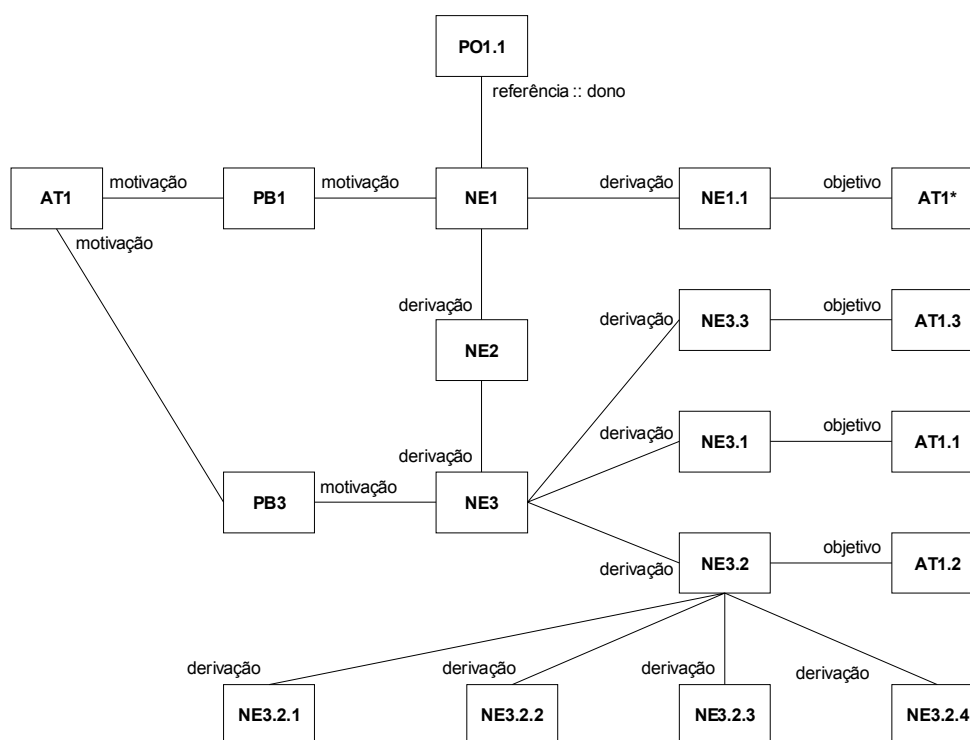
novos clientes, por exemplo), modelar a participação de mercado das drogas de uma mesma classe de doença, permitir a inclusão de novos medicamentos e modelar os índices de inflação, que alteram os valores dos remédios e, conseqüentemente, os valores de reembolso.

A Tabela 7 resume as necessidades e problemas do projeto PR. A sigla NE identifica necessidades e PB, problemas.

**Tabela 7 - Resumo das necessidades do projeto PR**

<b>Necessidades</b>	
NE1	Aprimorar a elaboração de contratos de reembolso
NE1.1	Sistema que efetue cálculos de reembolso para períodos futuros
NE2	Compreender o comportamento dos reembolsos no tempo
NE3	Simular reembolsos futuros de forma rápida e automatizada
NE3.1	Cadastro de contratos experimentais
NE3.2	Cadastro de cenários de predição de registros de compra
NE3.2.1	Modelagem do fator de acréscimo na utilização dos planos
NE3.2.2	Modelagem da participação de mercado das drogas de uma mesma classe
NE3.2.3	Inclusão de novos medicamentos
NE3.2.4	Modelagem dos índices de inflação dos preços de medicamentos
NE3.3	Cálculo automático de reembolsos com base em cenários de predição e contratos experimentais
<b>Problemas</b>	
PB1	A atividade atual não produz contratos otimizados
PB3	A atividade atual demanda muito esforço e tempo

A Figura 68 representa o modelo de necessidades do projeto PR.



**Figura 68 - Modelo de necessidades do projeto PR**



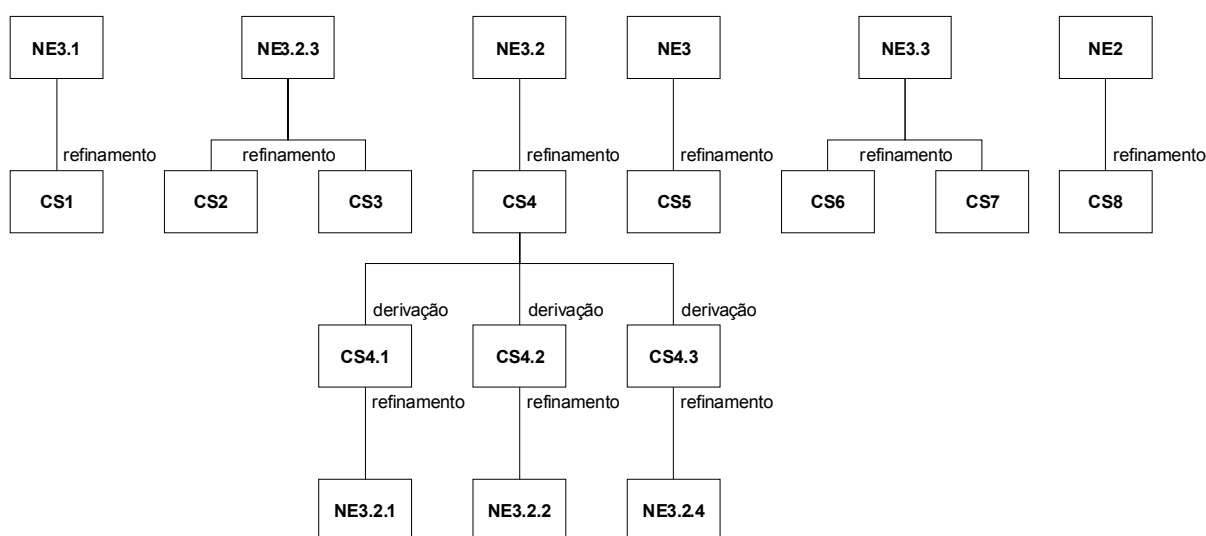
## 8.5 Características do Sistema

A maioria das características propostas para o sistema reflete os necessidades dos envolvidos, conforme apresenta a Tabela 8.

**Tabela 8 - Resumo das características do projeto PR**

Características do Sistema	
CS1	Alteração do GPF para permitir a inclusão de contratos experimentais
CS2	Alteração do GPF para permitir a inclusão de medicamentos experimentais
CS3	Alteração do GPF para permitir o uso de medicamentos experimentais nas cestas de produtos presentes nos contratos de reembolso
CS4	Nova funcionalidade para cadastro de cenários de predição de mercados futuros
CS4.1	Entrada de fator de acréscimo do número de usuários dos planos
CS4.2	Entrada de porcentagem de participação de mercado para os medicamentos de uma cesta de medicamentos
CS4.3	Entrada de índice anual de inflação
CS5	Ferramenta para geração de registros de venda no futuro utilizando como base os registros de compra do passado e os cenários de predição de mercados futuros
CS6	Alteração da ferramenta de cálculo de reembolso do GPF
CS7	Funcionalidade de simulação de cálculo de reembolso
CS8	Geração de gráficos com as previsões de reembolso no tempo

A Figura 69 ilustra o modelo de características e suas relações com o modelo de necessidades.



**Figura 69 - Modelo de características do projeto PR**

A necessidade de efetuar cálculos de reembolso no projeto PR suscitou a seguinte questão: qual ferramenta utilizar para efetuar o cálculo de reembolso? Três alternativas foram propostas:

- Utilizar a ferramenta de cálculo existente no GPF, sem alterações
- Reformular a ferramenta de cálculo do GPF
- Criar uma nova ferramenta de cálculo

O fato de a ferramenta de cálculo do GPF já estar pronta, testada e funcionando, é um argumento que favorece a utilização desta ferramenta. Entretanto, a maneira como os cálculos são efetuados, considerando apenas dados históricos, impede o uso da ferramenta do GPF, sem alterações, no projeto PR.

Aproveitar todas as regras de negócio já implementadas na ferramenta de cálculo é um fato que favorece a opção pela reformulação da ferramenta. Por fim, a criação de uma nova ferramenta traria o ônus de manutenção de duas ferramentas distintas para o mesmo fim.

A decisão tomada foi reformular a ferramenta de cálculo do GPF de modo a ser utilizada tanto pelo GPF quanto pelo PR. A Figura 70 ilustra o modelo da base racional que resultou na criação da característica de sistema CS6. A sigla QU se refere ao elemento questão, AL à alternativa, AG ao argumento e DE à decisão.

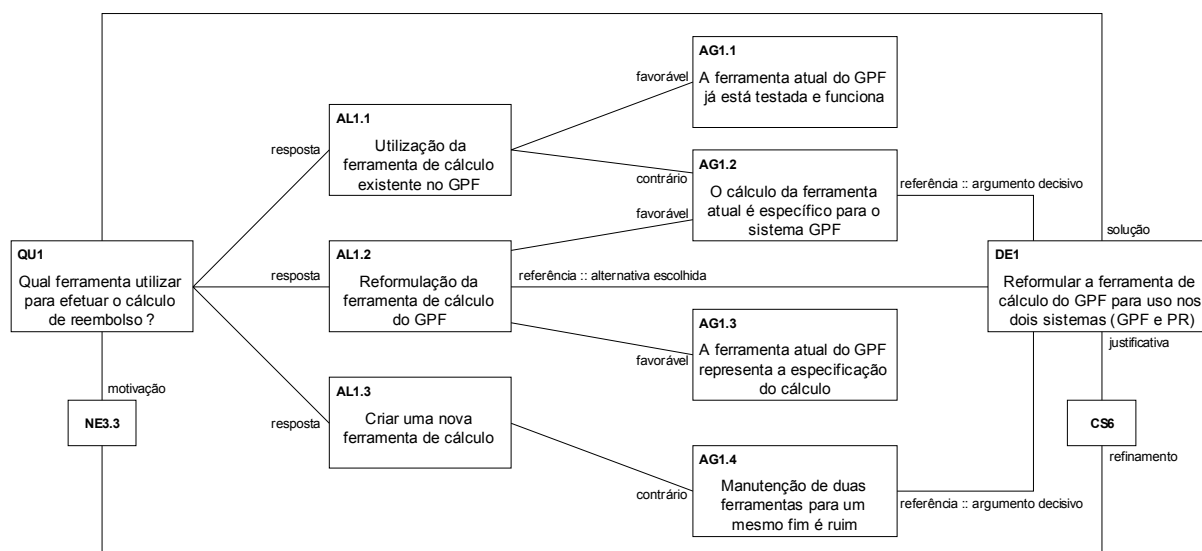


Figura 70 - Modelo de base racional que justifica uma característica de *software*

## 8.6 Requisitos de *Software*

Os requisitos de *software* do projeto PR foram formalizados em casos de uso. Entretanto, não é o escopo deste exemplo e do presente trabalho apresentar um caso de uso completo, em sua forma correta, e sim identificar como os elementos constituintes de um caso de uso se encaixam no modelo de rastreamento de requisitos proposto.

Dois casos de uso fazem parte do exemplo. O caso de uso CA1, que define o processo de simulação de reembolso, é descrito em seguida com maiores detalhes, apresentando os atores participantes e as etapas do processo (na forma de seções do caso de uso). O caso de uso CA2, referente ao processo de elaboração dos cenários de predição, é usado como referência para o caso de uso CA1.

Três atores fazem parte do caso de uso CA1:

- AO1 – administrador de contrato
- AO2 – ferramenta de geração de registros de venda no futuro
- AO3 – ferramenta de cálculo do reembolso

A funcionalidade principal do caso de uso CA1 é a simulação de reembolso com sucesso. Em uma simulação, o administrador de contrato escolhe um cenário de predição e informa o período da simulação. A ferramenta de geração de registros de venda consulta a base de dados em busca dos registros relevantes, e agrega esses registros por meio de critérios estabelecidos. Por exemplo, o nome do cliente que efetuou a compra de um medicamento não é relevante, e é descartado. O número de compras por medicamento, entretanto, é de vital importância, e faz parte da agregação. Após agregar os registros, a ferramenta aplica os fatores de predição.

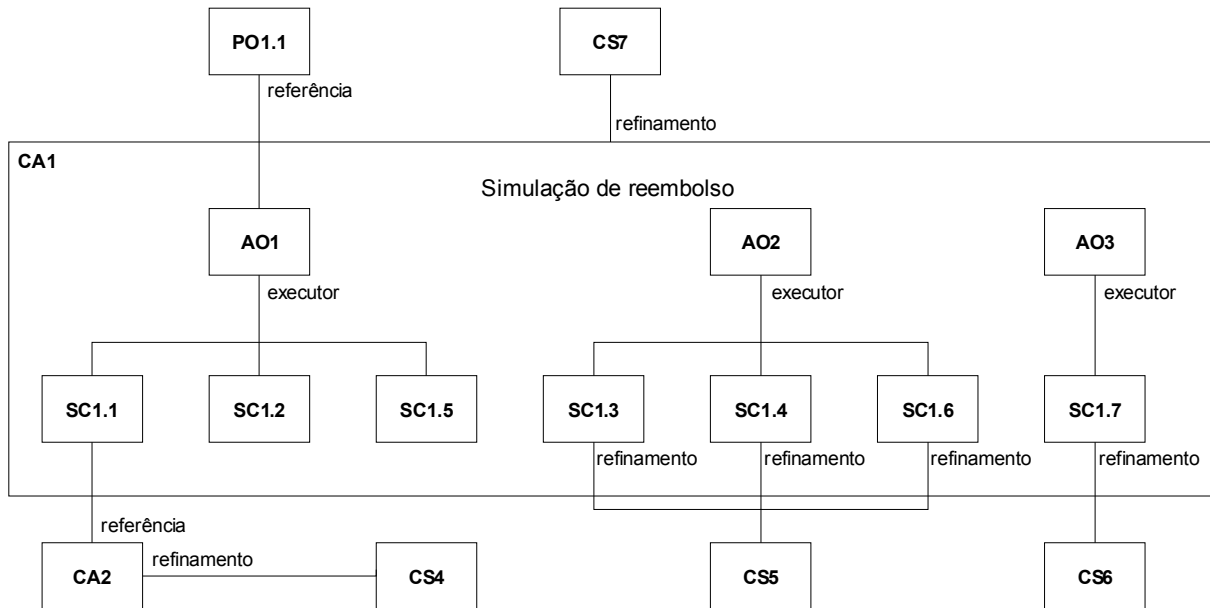
O administrador de contrato recebe em sua tela uma estimativa de participação de mercado dos medicamentos baseada nos registros previstos pela ferramenta de geração de registros. O administrador de contrato pode alterar as porcentagens de participação, o que leva a ferramenta de geração de registros a alterar alguns valores destes registros. Por fim, a ferramenta de cálculo de reembolso realiza o cálculo da simulação e apresenta o relatório ao usuário.

As sete etapas que compõem a funcionalidade principal do caso de uso CA1 são resumidas na Tabela 9.

**Tabela 9 - Resumo das etapas do caso de uso CA1**

<b>Etapa</b>	<b>Descrição</b>	<b>Ator</b>
SC1.1	Escolha do cenário de predição	AO1
SC1.2	Escolha do período de simulação	AO1
SC1.3	Agregação dos registros de venda	AO2
SC1.4	Aplicação dos fatores de predição aos registros de venda agregados	AO2
SC1.5	Ajuste na participação de mercado dos medicamentos	AO1
SC1.6	Alteração nos registros de venda agregados de acordo com a nova participação de mercado dos medicamentos	AO2
SC1.7	Cálculo do reembolso	AO3

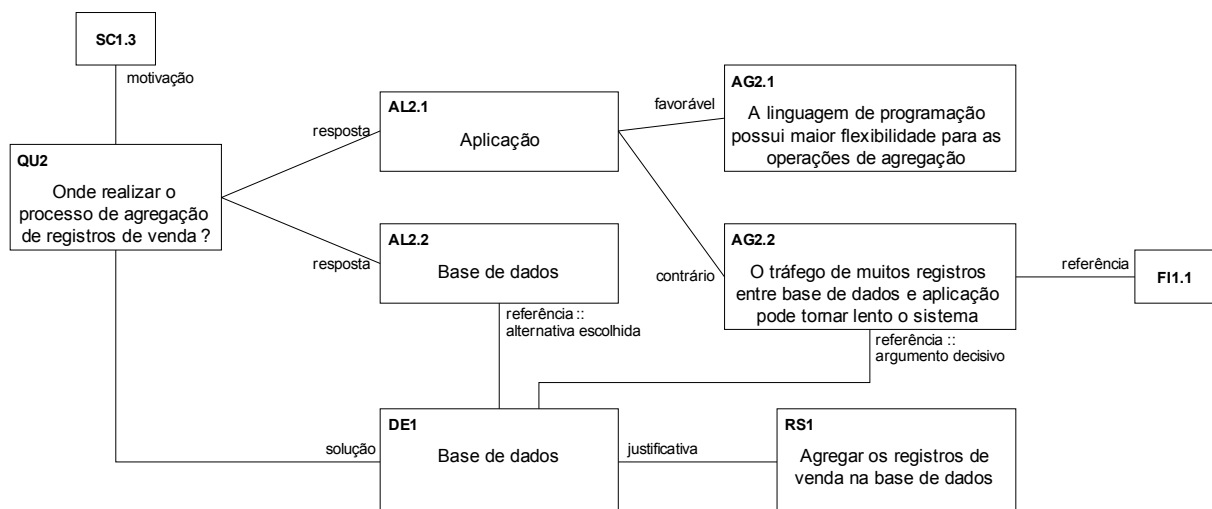
A Figura 71 ilustra o modelo de requisitos de *software* do projeto PR, incluindo os relacionamentos de refinamento com as características de *software* e a referência ao contexto organizacional.



**Figura 71 - Modelo de requisitos de *software* do projeto PR**

A etapa SC1.3 do caso de uso CA1 levantou uma questão importante: onde seria realizado o processo de agregação dos registros de venda? As alternativas eram realizar a agregação na própria aplicação ou na base de dados. Um argumento favorável a aplicação indicava a maior flexibilidade disponível em uma linguagem de programação para transformar (agregar) os registros de venda. Por outro lado, o número muito elevado de registros de venda, de acordo com a meta de atingir 6 milhões de clientes, poderia tornar muito lenta a comunicação entre a base de dados e a aplicação. Este último argumento foi decisivo na escolha da base de dados como local da agregação dos registros, dando origem a um requisito não-funcional do projeto.

A Figura 72 ilustra a base racional motivada por um requisito funcional e que justifica um requisito não-funcional.



**Figura 72 - Modelo de base racional que justifica um requisito de *software***

### 8.7 Exemplos de Rastreamento

Um dos conceitos adotados no presente trabalho considera que as necessidades dos envolvidos advêm da diferença entre o contexto desejado e o contexto com problemas. A Figura 73 ilustra o rastreamento de um requisito de *software* até os elementos do contexto organizacional que lhes deu origem, destacando a diferença dos contextos como origem das necessidades. As linhas cheias denotam rastreamento explícito e as linhas pontilhadas, rastreamento implícito.

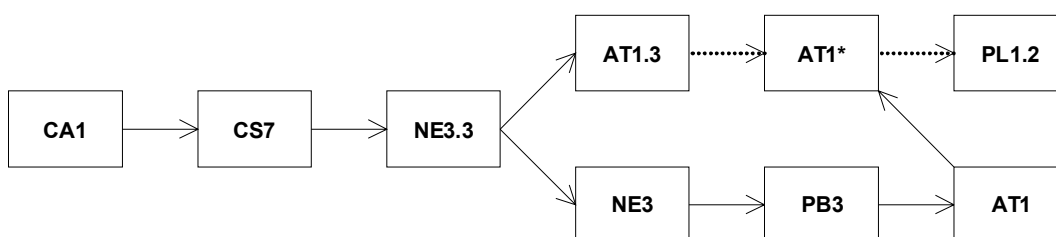


Figura 73 - Exemplo de necessidade como diferença do contexto organizacional

Outro importante conceito adotado no presente trabalho é a existência de uma base racional que justifica um requisito. Além disso, por um requisito poder motivar uma base racional, é possível relacionar dois requisitos indiretamente por meio da base racional, como ilustra a Figura 74.

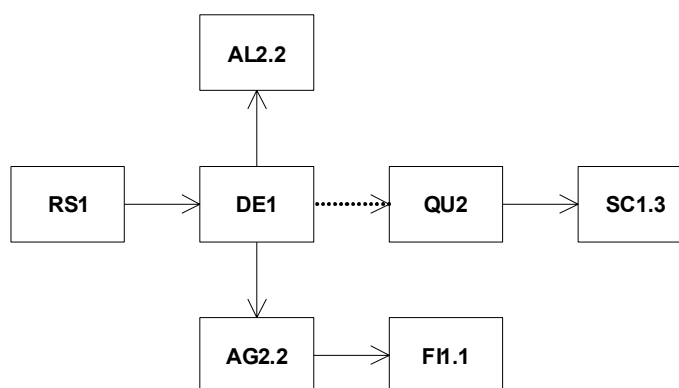


Figura 74 - Exemplo de base racional como rastreamento entre dois requisitos

### 8.8 Conclusão

Este capítulo apresentou um exemplo de utilização do modelo de rastreamento de requisitos proposto no presente trabalho e destacou exemplos de rastreamentos possíveis utilizando o modelo gerado. Além de apresentar os modelos de cada um dos três elementos do meta-modelo (contexto organizacional, base racional e requisito), mostrou exemplos de todos os relacionamentos de rastreamento entre estes três elementos.

## 9 Conclusão

Este capítulo apresenta as conclusões sobre os resultados obtidos pelo presente trabalho e também apresenta recomendações para trabalhos futuros.

### 9.1 Resultados Obtidos

O objetivo do presente trabalho era criar um modelo de rastreamento de requisitos que permitiria relacionar os requisitos de *software* ao contexto organizacional que lhes deu origem. O trabalho alcançou tal objetivo em duas etapas: (a) construção de um meta-modelo de rastreamento e (b) instanciação deste meta-modelo por meio da definição de modelos para cada um dos elementos do meta-modelo.

A criação do meta-modelo é, de certa forma, informal. Não se baseia em estudos empíricos em organizações, mas sim na abstração de conceitos que julga-se serem importantes para o rastreamento de requisitos. Dos três elementos do meta-modelo, o contexto organizacional e os requisitos são obrigatórios do ponto de vista do objetivo traçado para o trabalho. A base racional é um complemento freqüentemente utilizado em rastreamentos pré-ER.

Embora sejam sete os relacionamentos previstos no meta-modelo, eles cumprem quatro funções distintas: registrar a evolução do contexto organizacional, relacionar os requisitos à diferença entre contexto desejado e contexto do problema, registrar a base racional por trás das soluções propostas e tornar o contexto organizacional uma espécie de domínio de termos referenciados pelos requisitos e pela base racional.

O modelo proposto baseia-se no meta-modelo de rastreamento e utiliza-se de outros trabalhos para a definição de seus elementos e relacionamentos. A base racional e os requisitos são conceitos utilizados com muita freqüência em modelos de rastreamento de requisitos pré-ER, como atestam os trabalhos de Letelier (2002), Leffingwell e Widrig (2003), Ramesh e Jarke (2001), Spence e Probasco (2000), Louridas e Loucopoulos (2000) e Pinheiro (2003), entre outros. Por esta razão, esses conceitos já estão mais amadurecidos e, provavelmente, já tiveram utilização prática suficiente para provar sua validade. Assim, podem ser utilizados no modelo de rastreamento sem maiores problemas.

O conceito do contexto organizacional também aparece em diversos trabalhos, na forma de modelos de negócio, com a função de aprimorar o entendimento de uma organização. Entretanto, há poucos estudos que efetivamente relacionam o contexto organizacional com as especificações de *software*, como o faz Bleinstein, Cox e Verner (2005). Como o

rastreamento entre requisitos e o contexto organizacional não é um conceito tão amadurecido quanto os conceitos de requisito ou base racional, foi necessário buscar uma maneira para guiar o processo de construção do contexto organizacional, e o QRAC de Zachman cumpriu este papel.

Por fim, foi apresentado um exemplo que auxilia o entendimento do modelo proposto, embora não tenha a função de validá-lo. Para isso, seria fundamental a elaboração de um conjunto de estudos de caso.

Há um público-alvo para o uso de rastreamento de requisitos de *software* pré-ER, em especial os que se interessam por registrar a real origem dos requisitos e que têm interesse em rastreá-los ao contexto organizacional. O modelo de rastreamento proposto atende a um sub-grupo desse público-alvo, que tem nos processos de negócio o principal aspecto de seu contexto organizacional.

Entretanto, o modelo proposto, assim como qualquer modelo de rastreamento de requisitos de *software*, demanda o controle de muitas informações, o que se torna inviável sem o uso de uma ferramenta adequada. E não somente para a construção do modelo em si, como foi realizado no exemplo apresentado, mas para a futura utilização do modelo para obtenção das informações de rastreamento.

Portanto, o modelo apresentado pode ser considerado um trabalho inicial, que pode motivar a elaboração de métodos para a obtenção das informações presentes no modelo e também a criação de modelos mais completos, o que, em última análise, pode dar origem a uma ferramenta de rastreamento de requisitos de *software* pré-ER.

No estágio em que se encontra, o modelo proposto pode ser útil e trazer alguns benefícios. Em primeiro lugar, por incentivar a modelagem do contexto organizacional, atividade que por si só melhora o processo de elaboração dos requisitos de *software* ao dar ao projetista um melhor entendimento do problema. Em segundo lugar, o modelo pode ser construído para pequenos projetos de *software* nos quais há muita informação a respeito do contexto organizacional mas não muitos requisitos de *software* e também pouca necessidade de registro da base racional. Nesses casos, o volume de informação que deve ser modelada não é muito grande, pois são poucos os requisitos, o que torna viável a utilização manual do modelo. Por outro lado, a riqueza do contexto organizacional, quando modelada, traz os benefícios apreçados ao longo do presente trabalho.

Conforme mencionado no capítulo 1, a utilidade do modelo proposto é responder a três perguntas:

- Em que contexto organizacional o requisito de *software* foi elaborado?

O modelo proposto registra duas versões do contexto, denominadas de contexto do problema e contexto desejado. Uma das vantagens de se determinar o contexto organizacional antes de especificar um *software* é que esse processo melhora a compreensão do problema e, conseqüentemente, melhora o processo de elaboração da solução.

Registrar o contexto organizacional, ao invés de apenas obtê-lo para melhorar o entendimento do problema, é importante pois permite que novas equipes, que não participaram do desenvolvimento do *software*, tenham uma rápida imersão nos conceitos da organização quando necessitarem efetuar uma alteração ou manutenção do *software*.

Por fim, registrar os relacionamentos entre os requisitos de *software* e elementos do contexto organizacional facilita o entendimento do *software* e permite que a leitura do contexto organizacional seja simplificada, pois o rastreamento funciona como um filtro para os elementos relevantes. Por exemplo, se uma empresa é contratada para dar manutenção em uma parte do *software*, ela pode estudar o contexto organizacional relacionado à parte que deve ser alterada, e desprezar as partes do contexto que são irrelevantes, poupando tempo nessa atividade.

- Como uma mudança no contexto organizacional pode influenciar o requisito de *software*?

Duas situações podem envolver o modelo de rastreamento de requisitos. Em um primeiro momento, é preciso criar o modelo. Nessa primeira etapa, o rastreamento de requisitos não traz tantas vantagens. De certa forma, a utilização do modelo motiva seu usuário a registrar o contexto organizacional e a base racional de algumas decisões, o que é um ponto positivo para a elaboração do *software*. Entretanto, o rastreamento nesse momento permite apenas verificar se todas as necessidades foram atendidas e se não há especificação de *software* desnecessária.

A grande vantagem do rastreamento de requisitos se dá quando o modelo já está pronto e o contexto organizacional sofre uma nova alteração. Por meio do modelo de rastreamento, é possível analisar, com mais facilidade, se essa alteração do contexto organizacional afeta algum requisito de *software* já existente. Esse tipo de informação, portanto, permite executar uma análise mais precisa do impacto causado em um *software* por mudanças no contexto organizacional.



O exemplo apresentado ilustrou a especificação de um *software* decorrente da modificação ocorrida no contexto organizacional. Em resumo, ilustrou a criação do modelo, e não o seu uso. Entretanto, apresentou exemplos de rastreamento no modelo, que permitem analisar como alterações futuras no contexto organizacional poderiam afetar os requisitos de *software*.

- Quais são as justificativas para a elaboração do requisito de *software*?

O modelo proposto inclui uma base racional que serve de justificativa para características ou requisitos de *software*. O registro da base racional melhora o entendimento de um requisito, pois, ao registrar as discussões que levaram a ele, disponibilizam informações que facilitam o seu entendimento. Além disso, o registro das alternativas e argumentos utilizados evita retrabalho em futuras discussões e garante que alternativas não vencedoras também foram analisadas.

## 9.2 Trabalhos Futuros

Por questões de praticidade, apenas três dos aspectos organizacionais foram utilizados na construção de um modelo do contexto organizacional. Portanto, o presente trabalho pode ser estendido com a adição dos outros três aspectos faltantes: aspecto dos dados, do tempo e da localização. O aspecto dos dados, em particular, tende a ser valioso no contexto do modelo proposto. Em primeiro lugar, por ser o aspecto que representa o modelo de fatos de uma organização. O modelo de fatos, segundo Tony Morgan (2002), é a principal fonte de referências para as regras de negócio, e, provavelmente, para os requisitos e base racional. Em segundo lugar, por ser importante para o próprio aspecto dos processos. Essa importância pode ser percebida pelo IDEF3, que possui duas visões sobre os processos, uma sob a ótica dos próprios processos e outra sob a ótica dos produtos.

Outra possibilidade em relação ao presente trabalho é defini-lo de maneira mais formal com base na UML. Na criação do meta-modelo menciona-se duas possibilidades de extensão da UML. A opção adotada é a extensão das meta-classes da UML, mas a outra opção é a criação de perfis UML. Um perfil (*profile*) provê um mecanismo de adaptação de um meta-modelo existente (no caso, a própria UML) com constructos que são específicos para um domínio, plataforma ou método em particular (no caso, o rastreamento de requisitos) (OMG, 2007a, 2007b). Esta formalização do modelo com a UML permitiria a criação de uma ferramenta de rastreamento baseada no modelo. Para a utilização prática do rastreamento de requisitos, devido à sua natureza em demandar muitos relacionamentos entre conceitos distintos, uma ferramenta é fundamental.

Por fim, sugere-se a elaboração de estudos de caso para o modelo proposto, que atendam a duas situações: criação de uma instância do modelo e sua utilização quando o contexto organizacional sofrer alterações. Estudos de caso realizados para organizações com características semelhantes pode verificar se o modelo proposto é adequado e traz vantagens para os usuários dessas organizações. O resultado do estudo de caso pode verificar se os elementos selecionados para o contexto organizacional são suficientes e se os relacionamentos de rastreamento escolhidos formam a gama de relacionamentos ideal para a atividade de rastreamento de requisitos pré-ER.

## Referências

- BLEISTEIN, S; COX, K; VERNER, J. **Strategic alignment in requirements analysis for organizational IT: an integrated approach**. 20th ACM Symposium on Applied Computing (SAC'05), Santa Fe, NM, EUA, p. 1300-1307, mar. 2005.
- CHAPIN, D; ESTEP, C; HALL, J; HEALY, J; HEALY, K; KOLBER, A; ROSS, R; SCHACHER, M. **The Business Motivation Model: Business Governance in a Volatile World, release 1.2**. The Business Rules Group, set. 2005.
- CONKLIN, J; BEGEMAN, M. **gIBIS: A hypertext tool for exploratory policy discussion**. ACM Transactions on Office Information Systems, v. 6, n. 4, p.303-331, out. 1988
- CUMMINS, F; KOETHE, M; BAISLEY, D; FRANK, K; CHOBANTONOV, P; CASANAVE, C; RIVETT, P; **Organization Structure Model (OSM)**. Second Initial Submission. Disponível em < <http://www.omg.org/docs/bmi/06-11-02.pdf>>, nov. 2006.
- DEPARTMENT OF TREASURY. **Treasury Enterprise Architecture Framework**. v.1, Disponível em <<http://www.eaframeworks.com/TEAF/teaf.doc>>, Acesso em 21 mar. 2007, jul. 2000
- FAULK, S. **Software Requirements: A Tutorial**. Technical Report NRL/MR/5546-95-7775, Naval Research Laboratory, Washington DC, 1995.
- GOTEL, O; FINKELSTEIN, A. **An Analysis of the Requirements Traceability Problem**. 1<sup>st</sup> International Conference on Requirements Engineering (ICRE'94), Colorado Springs, p. 94-101, abr. 1994.
- JARKE, M. **Requirements tracing**. Communications of the ACM, New York, NY, EUA, v. 41, n. 12, p. 32-36, dez. 1998.
- LEE, J. **Extending the Potts and Bruns Model for Recording Design Rationale**. Proceedings of the 13<sup>th</sup> International Conference on Software Engineering (ICSE'91), Los Amigos, CA, EUA, p. 114-125, mai. 1991.
- LEFFINGWELL, D; WIDRIG, D. Introduction to Requirements Management. In: \_\_\_\_\_. **Managing Software Requirements: a Use Case Approach**. 2 ed. Addison-Wesley, 2003.
- LEIST, S; ZELLNER, G. **Evaluation of Current Architecture Frameworks**. Proceedings of the 2006 ACM symposium on Applied computing, p. 1546-1553, abr. 2006.
- LETELIER, P. **A framework for requirements traceability in UML-based projects**. 1<sup>st</sup> International Workshop on Traceability in Emerging Forms of Software Engineering. 17<sup>th</sup> IEEE International Conference on Automated Software Engineering, Reino Unido, 2002.
- LIST, B; KORHERR, B. **An Evaluation of Conceptual Business Process Modelling Languages**. Proceedings of the 2006 ACM symposium on Applied computing, p. 1532-1539, abr. 2006.
- LOURIDAS, P; LOUCOPOULOS, P. **A Generic Model for Reflective Design**. ACM Transactions on Software Engineering and Methodology, v. 9, n. 2, p. 199-237, abr. 2000

MAYER, R; MENZEL, C; PAINTER, M; DE WITTE, P; BLINN, T; PERAKATH, B. **IDEF3 Process Description Capture Method Report**. Knowledge Based Systems Inc, 1995.

MORGAN, T. Frameworks, Architectures, and Models. In: \_\_\_\_\_. **Business Rules and Information Systems**. Addison-Wesley, 2002.

OBJECT MANAGEMENT GROUP (OMG). **OMG Unified Modeling Language (OMG UML), Superstructure**. v. 2.1.2, disponível em <<http://www.omg.org/docs/formal/07-11-02.pdf>>, nov. 2007.

OBJECT MANAGEMENT GROUP (OMG). **OMG Unified Modeling Language (OMG UML), Infrastructure**. v. 2.1.2, disponível em <<http://www.omg.org/docs/formal/07-11-04.pdf>>, nov. 2007.

OBJECT MANAGEMENT GROUP (OMG). **Meta Object Facility (MOF) 2.0 Core Specification**. Disponível em <<ftp://ftp.omg.org/pub/docs/ptc/04-10-15.pdf>>, out. 2003.

PINHEIRO, F. Requirements Traceability. In.: Doorn, J. **Perspectives on Software Requirements**. Kluwer Academic Publishers, 2003.

RAMESH, B; JARKE, M. **Toward Reference Models for Requirements Traceability**. IEEE Transactions on Software Engineering, v. 27, n. 1, p. 58-93, jan. 2001.

RIVET, P; ASTIER, S; STRINGER, D; CUMMINS, F; CASANAVE, C; CHOBANTONOV, P; LONJON, A; KHUSIDMAN, V; BOCK, C. **Business Process Definition MetaModel (BPDM)**. Final Submission. Disponível em <<http://www.omg.org/docs/bmi/07-03-01.pdf>>, mar. 2007.

SOWA, J; ZACHMAN, J. **Extending and formalizing the framework for information system architecture**. IBM Systems Journal, v. 31, n. 3, p. 590-616, abr. 1992.

SPENCE, I; PROBASCO, L. **Traceability Strategies for Managing Requirements with Use Cases**. Rational Software ABCte Paper, 2000.

THE OPEN GROUP. **Welcome to TOGAF**. Disponível em <<http://www.opengroup.org/architecture/togaf8-doc/arch/>>, Acesso em 20 mar. 2007.

URBACZEWSKI, L; MRDALJ, S. **A Comparison of Enterprise Architecture Frameworks**. Issues in Information Systems, v. 7, n. 2, 2006

ZACHMAN, J. **A framework for information systems architecture**. IBM Systems Journal, v. 26, n. 3, p. 276-292, 1987.