

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE INFORMÁTICA  
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**Modelagem Temporal de Sistemas:  
uma Abordagem Fundamentada  
em Redes de Petri**

por

DANTE CARLOS ANTUNES

Dissertação submetida à avaliação,  
como requisito parcial para a obtenção  
do grau de Mestre em Ciência da Computação

Prof. Carlos Alberto Heuser

Orientador



Porto Alegre, março de 1997

**UFRGS**  
**INSTITUTO DE INFORMÁTICA**  
**BIBLIOTECA**

**CIP - CATALOGAÇÃO DA PUBLICAÇÃO**

Antunes, Dante Carlos

Modelagem temporal de sistemas: uma abordagem fundamentada em redes de Petri / por Dante Carlos Antunes. - Porto Alegre: CPGCC da UFRGS, 1997. 155 f. : il.

Dissertação (mestrado) - Universidade Federal do Rio Grande do Sul. Curso de Pós-Graduação em Ciência da Computação, Porto Alegre, BR-RS, 1997. Orientador: Heuser, Carlos Alberto

1. Modelagem conceitual. 2. Modelagem temporal. 3. Dimensão temporal. 4. Especificação de transações. 5. Redes de Petri. I. Heuser, Carlos Alberto. II. Título.

**UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL**

Reitora: Profa. Wrana Panizzi

Pró-Reitor de Pesquisa e Pós-Graduação: Prof. José Carlos Ferraz Hennemann

Diretor do Instituto de Informática: Prof. Roberto Tom Price

Coordenador do CPGCC: Prof. Flávio Rech Wagner

Bibliotecária-Chefe do Instituto de Informática: Zita Prates de Oliveira

MOI

Dedico esta dissertação

A Hilde, minha esposa,  
*que esteve ao meu lado em todos os momentos,  
com o seu carinho, incentivo e compreensão.*

A Anays, minha filha,  
*que veio ao mundo trazendo uma nova luz,  
tornando mais cristalina a minha caminhada.*

A Maria, minha mãe,  
*que despertou em mim a sede do saber  
e torce pelo meu êxito.*

## Agradecimentos

Aos ex-diretores da CELEPAR, Paulo Roberto de Mello Miranda, Anísio Ribas Bueno e Nelson de Marco Rodrigues, que, em 1994, permitiram o meu afastamento em tempo integral para frequentar o curso de pós-graduação em ciência da computação do Instituto de Informática da UFRGS.

Aos diretores atuais da CELEPAR, Francisco Luiz Albuquerque Krassuski, Lúcio Alberto Hansel e Danilo Scalet, que mantiveram o apoio, para que a minha participação no curso chegasse à sua conclusão.

À ex-gerente da GPT, Maria Alexandra V. C. da Cunha por permitir o meu afastamento das atividades da área e por ter me apoiado ao longo do curso.

À atual gerente da GPT, Sara Fichman Raskin, pelo apoio prestado, bem como, por ter aceito o meu pedido de prorrogação do prazo de afastamento, e ter intercedido, neste sentido, junto à Diretoria da CELEPAR.

Ao professor Heuser. A sua capacidade, aliada ao seu espírito sereno e aberto, propiciaram o clima ideal para o meu trabalho de pesquisa. Foi uma grande satisfação estudar sob a sua orientação.

Aos meus amigos, Maria Alexandra e Pedro Kantek, pelo “empurrão” e permanente incentivo. Não fosse vocês, dificilmente eu teria me aventurado a iniciar este curso de mestrado. Valeu a força!

Aos professores Flávio Bortolozzi e Dewey Wollman, da PUC-Pr, pelas suas recomendações, que muito contribuíram para que eu fosse aceito como aluno do CPGCC.

Ao amigo Ozir Zotto, por ter tido a paciência de revisar o texto.

Aos colegas de mestrado, Alexandre, André, José Osvaldo, Sérgio e Marcelo, pelo companheirismo e pelos interessantes debates que mantivemos.

Às bibliotecárias, funcionárias e funcionários das bibliotecas do Instituto de Informática e da CELEPAR, que sempre me atenderam com gentileza e eficiência.

A todos os colegas da GPT, pelo espírito de colaboração.

Ao pessoal da Gerência de Recursos Humanos da CELEPAR, pelo suporte prestado.

Ao pessoal da secretaria do CPGCC, pela presteza com que sempre fui atendido.

Aos projetistas e programadores anônimos do Word e de PowerPoint.

Às minhas cunhadas, Marli e Adriana, pela hospitalidade e generosidade.

Aos meus sobrinhos Leda e José Vicente. Brincar com vocês me ajudou a aliviar a carga.

Aos meus irmãos, Marciliano e Luciano, por segurarem a barra na retaguarda e pela torcida. Valeu manos!

## Sumário

<b>Lista das figuras .....</b>	<b>11</b>
<b>Lista das tabelas .....</b>	<b>14</b>
<b>Resumo .....</b>	<b>15</b>
<b>Abstract.....</b>	<b>17</b>
<b>1 Introdução.....</b>	<b>19</b>
1.1 Especificação de sistemas de informação ao nível conceitual.....	19
1.2 Vinculando os dados à dimensão temporal .....	19
1.3 Modelo dinâmico integrado ao modelo de dados temporal.....	22
1.4 Estrutura do texto .....	23
<b>2 Redes de Petri na modelagem de aspectos estáticos e dinâmicos de sistemas - retrospecto.....</b>	<b>25</b>
2.1 Redes de Petri elementares.....	26
2.2 A abordagem condição/evento.....	27
2.3 Redes de Petri compactas.....	28
2.3.1 Tipos de ramos.....	30
2.3.2 Linguagem de Anotação .....	30
2.3.3 Habilitação e efetivação de alterações .....	32
2.3.4 Referenciando todas as entidades de um lugar.....	32
2.4 Evidenciando propriedades estáticas através de redes de Petri.....	34
2.5 Tradução do modelo ER para rede de Petri .....	35

2.6	Modelo integrado ER/PN .....	37
2.7	Modelo ER-Tr.....	37

### **3 A dimensão temporal - conceitos e modelos..... 42**

3.1	Conceitos relativos à dimensão temporal .....	42
3.1.1	Eixo temporal.....	42
3.1.2	Intervalo de tempo.....	43
3.1.3	Elemento Temporal .....	44
3.1.4	Assinalamento Temporal.....	44
3.1.5	Domínio Temporal .....	45
3.1.6	Restrição Temporal .....	45
3.2	Modelos Entidade-Relacionamento temporais encontrados na literatura .....	45
3.2.1	O modelo ERT .....	45
3.2.2	O modelo TER.....	49
3.2.3	O modelo TEER.....	50
3.2.4	O modelo STEER.....	50
3.3	Demais modelos temporais.....	51

### **4 O modelo de dados temporal TempER ..... 52**

4.1	Justificativas.....	52
4.1.1	Porque um modelo ER temporal é necessário .....	52
4.1.2	Requisitos necessários a um modelo ER temporal.....	53
4.1.3	Cruzamento de modelos ER temporais com os requisitos.....	54
4.1.4	Uma proposta para atender os requisitos.....	55
4.2	Visão geral do modelo de dados TempER.....	56
4.2.1	O identificador interno de entidades - OID.....	60
4.3	As perspectivas temporal e intemporal das entidades.....	60
4.3.1	Perspectiva intemporal.....	60

4.3.2	Perspectiva temporal .....	60
4.4	Componentes do modelo de dados TempER.....	62
4.4.1	Entidades.....	63
4.4.1.1	Entidades Transitórias.....	63
4.4.1.2	Entidades Perenes.....	64
4.4.2	Relacionamentos.....	65
4.4.2.1	Relacionamentos Temporais.....	65
4.4.2.2	Relacionamentos Intemporais.....	66
4.4.2.3	Restrições de cardinalidade dos relacionamentos .....	67
4.4.3	Atributos .....	69
4.4.3.1	Atributos temporais .....	69
4.4.3.2	Atributos intemporais.....	70
4.5	Dicionário de dados .....	72
4.5.1	Sintaxe do dicionário de dados em BNF.....	72
4.5.2	Regras adicionais de formação do dicionário de dados.....	73
4.5.3	Convenções do dicionário de dados .....	73
4.5.4	Comentários .....	73
4.6	Generalização / especialização .....	74
4.6.1	Notação de Generalização/Especialização.....	74
4.6.2	Generalização / Especialização Temporal.....	75
4.6.3	Generalização / Especialização Intemporal.....	77
4.6.4	Atributo identificador de sub-entidade .....	78
<b>5</b>	<b>Modelagem de transações combinada com modelo de dados temporal.....</b>	<b>79</b>
5.1	Elementos dinâmicos do modelo TempER-Tr.....	80
5.1.1	Eventos externos .....	80
5.1.2	Fluxos externos .....	80
5.1.3	Transações .....	81
5.1.4	Fluxos internos.....	81
5.2	Tipos de fluxos internos .....	81
5.2.1	Fluxos internos de inclusão .....	82
5.2.2	Fluxos internos de exclusão .....	83
5.2.3	Fluxos internos de manipulação de existência de entidades transitórias.....	84

5.2.4	Fluxos internos de alteração.....	85
5.2.5	Fluxos de verificação de presença.....	85
5.2.6	Fluxos de verificação de ausência.....	86
5.3	Linguagem de anotação do modelo TempER-Tr.....	87
5.3.1	Anotações dos fluxos internos de uma transação.....	87
5.3.1.1	Regras adicionais de anotação conforme o tipo de fluxo interno.....	90
5.3.2	Anotações dos fluxos externos de uma transação.....	92
5.3.3	Anotações das fórmulas de uma transação.....	92
5.3.3.1	Escalares.....	92
5.3.3.2	Colunas derivadas.....	92
5.3.3.3	Tabelas derivadas.....	92
5.3.3.4	Elemento temporal.....	93
5.3.3.5	Assinalamento temporal.....	93
5.3.4	Sintaxe da linguagem de anotação do modelo TempER-Tr.....	94
5.3.4.1	Anotações específicas dos fluxos internos.....	94
5.3.4.2	Anotações específicas dos fluxos externos.....	94
5.3.4.3	Anotações específicas das fórmulas das transações.....	94
5.3.4.4	Elemento temporal.....	95
5.3.4.5	Variável e constante.....	96
5.4	Comportamento do modelo TempER-Tr.....	96
5.4.1	Objetos consumíveis de uma transação.....	96
5.4.2	Objetos produzíveis de uma transação.....	97
5.4.3	Habilitação de uma transação.....	98
5.4.4	A ocorrência de uma transação.....	98
5.4.5	Exemplificando o comportamento de uma transação.....	99
5.4.6	Conflito e concorrência.....	99
5.5	Conclusão.....	100

## **6 Mapeamento do modelo TempER-Tr para rede de Petri..... 101**

6.1	Mapeamento do modelo de dados temporal para rede de Petri.....	101
6.1.1	Mapeamento de conjuntos-entidade.....	101
6.1.1.1	Mapeamento de conjunto de entidades transitórias.....	102
6.1.1.2	Mapeamento de conjunto de entidades perenes.....	102
6.1.2	Mapeamento de relacionamentos.....	103
6.1.2.1	Mapeamento de relacionamentos temporais.....	103
6.1.2.2	Mapeamento de relacionamentos intemporais.....	105
6.1.3	Mapeamento de atributos.....	111

6.1.3.1	Mapeamento dos atributos temporais .....	111
6.1.3.2	Mapeamento dos atributos intemporais .....	112
6.1.3.3	Mapeamento de atributos identificadores .....	112
6.1.4	Mapeamento da estrutura generalização/especialização .....	112
6.1.4.1	Mapeamento de generalização/especialização temporal .....	113
6.1.4.2	Mapeamento de generalização/especialização intemporal .....	114
6.2	Mapeamento do modelo dinâmico - transações - para rede de Petri .....	120
6.2.1	Mapeamento dos fluxos externos .....	120
6.2.2	Mapeamento dos fluxos internos .....	122
6.2.2.1	Mapeamento dos fluxos de inclusão .....	122
6.2.2.2	Mapeamento dos fluxos de exclusão .....	125
6.2.2.3	Mapeamento dos fluxos de manipulação de existências .....	125
6.2.2.4	Mapeamento dos fluxos de alteração de atributos .....	126
6.2.2.5	Mapeamento dos fluxos de verificação de presença .....	127
6.2.2.6	Mapeamento dos fluxos de verificação de ausência .....	127
6.2.2.7	Mapeamento de fluxos que manipulam conjuntos de instâncias .....	130
6.2.3	Mapeamento das anotações .....	131
6.2.4	Simplificação da rede de Petri resultante .....	132

## **7 Exemplos de modelagem de transações - situações típicas ..... 133**

7.1	Aparecimento de entidade transitória .....	133
7.2	Aparecimento de entidade perene .....	133
7.3	Eliminação de entidade transitória .....	133
7.4	Encerramento de existência de entidade transitória .....	134
7.5	Reencarnação de entidade transitória .....	134
7.6	Inclusão e exclusão de relacionamento temporal .....	135
7.7	Alteração de atributo temporal de entidade .....	135
7.8	Alteração de atributo de relacionamento temporal .....	135
7.9	Alteração de atributo intemporal de entidade .....	136
7.10	Manipulação de conjuntos .....	136

<b>8 Conclusões finais .....</b>	<b>147</b>
8.1 Preenchendo requisitos da modelagem conceitual.....	147
8.2 Extensão temporal do modelo ER.....	147
8.3 Aspectos a serem aperfeiçoados.....	148
8.4 Estudos futuros .....	149
<b>Bibliografia .....</b>	<b>151</b>

## Lista das figuras

FIGURA 2.1 - Rede de Petri elementar.....	26
FIGURA 2.2 - Correspondência entre conceitos das redes elementares e da abordagem condição/evento.....	27
FIGURA 2.3 - Rede de Petri compacta.....	29
FIGURA 2.4 - Classificação dos ramos de uma rede de Petri.....	30
FIGURA 2.5 - Referenciando todas as entidades presentes em um lugar.....	33
FIGURA 2.6 - Referenciando todas as entidades ausentes de um lugar.....	33
FIGURA 2.7 - Evidenciando propriedades estáticas em uma rede de Petri.....	34
FIGURA 2.8 - Tradução do modelo ER para rede de Petri através de conexões mortas.....	36
FIGURA 2.9 - Tradução de atributos para rede de Petri.....	36
FIGURA 2.10 - Diagrama ER/PN integrando propriedades estáticas e dinâmicas.....	38
FIGURA 2.11 - Diagrama ER-Tr modelando aparecimento de entidade e relacionamento.....	39
FIGURA 2.12 - Mapeamento de diagrama ER-Tr para o modelo ER/PN.....	40
FIGURA 2.13 - Diagrama ER-Tr modelando alteração e o registro de histórico de relacionamento.....	40
FIGURA 3.1 - Simbologia do modelo ERT.....	46
FIGURA 3.2 - Modelo de dados temporal segundo a abordagem ERT.....	47
FIGURA 3.3 - Metamodelo do tempo na abordagem ERT.....	48
FIGURA 3.4 - Relações entre intervalos (períodos) de tempo.....	48
FIGURA 3.5 - Modelo de dados TER.....	49
FIGURA 4.1 - Comparação entre ER convencional e TempER.....	58
FIGURA 4.2 - Exemplo de povoamento de entidades e relacionamento em TempER.....	58
FIGURA 4.3 - Mesclando objetos de diferentes classificações temporais em um mesmo diagrama.....	59
FIGURA 4.4 - As duas perspectivas em relação ao tempo do conjunto-entidade <i>Empregado</i> .....	61
FIGURA 4.5 - Notação dos componentes básicos do modelo TempER.....	62
FIGURA 4.6 - Visualização da <i>existência</i> de uma entidade transitória.....	63
FIGURA 4.7 - Mapeamento de entidade transitória para ER convencional.....	63
FIGURA 4.8 - Visualização da <i>existência</i> uma entidade perene.....	64
FIGURA 4.9 - Mapeamento de entidade perene para ER convencional.....	65
FIGURA 4.10 Mapeamento de relacionamentos temporais e intemporais para ER convencional.....	66
FIGURA 4.11 - Mapeamento de atributos de entidade transitória para ER convencional.....	71
FIGURA 4.12 - Mapeamento de atributos de entidade perene para ER convencional.....	71
FIGURA 4.13 - Aplicação da estrutura Generalização/Especialização.....	74
FIGURA 4.14 - Os quatro tipos possíveis de estrutura generalização/especialização.....	75
FIGURA 4.15 - Mapeamento da estrutura generalização/especialização <i>temporal</i> para um diagrama ER convencional.....	76
FIGURA 4.16 - Mapeamento da estrutura generalização/especialização <i>intemporal</i> para um diagrama ER convencional.....	78

FIGURA 4.17 - Dicionário de dados ref. diagrama da fig. 4.13(a) .....	78
FIGURA 5.1 - Modelagem de uma transação .....	79
FIGURA 5.2 - Notação dos fluxos externos .....	80
FIGURA 5.3 - Tipos de fluxos internos que provocam mudança de estado no banco de dados .....	83
FIGURA 5.4 - Tipos de fluxos internos de verificação de presença/ausência.....	83
FIGURA 5.5 - Tupla de fluxo externo e pacotes de dados de fluxo interno de uma possível transação <i>AdmitirEmpregado</i> da figura 5.1 .....	89
FIGURA 5.6 - Efeito da transação <i>AdmitirEmpregado</i> .....	89
FIGURA 5.7 - Conjunto-transação <i>AumentarSaláriosDeUmGrupo</i> .....	91
FIGURA 5.8 - Tupla de fluxo externo e pacote de dados de fluxo interno de alteração de uma possível transação <i>AumentarSaláriosDeUmGrupo</i> .....	91
FIGURA 5.9 - Efeito da transação <i>AumentarSaláriosDeUmGrupo</i> sobre a população original do conjunto-entidade <i>Empregado</i> , mostrada na FIGURA 4.2.....	91
FIGURA 5.10 - Assinalamento temporal.....	93
FIGURA 6.1 - Mapeamento de entidade transitória para rede de Petri.....	102
FIGURA 6.2 - Mapeamento de entidade perene para rede de Petri.....	103
FIGURA 6.3 - Mapeamento de relacionamento temporal para rede de Petri .....	104
FIGURA 6.4 - Mapeamento de relacionamento intemporal para rede de Petri .....	106
FIGURA 6.5 - Mapeamento das cardinalidades (0, N) e (0, 1) de relacionamentos temporais.....	107
FIGURA 6.6 - Mapeamento das cardinalidades (1, N) e (1, 1) de relacionamentos temporais.....	108
FIGURA 6.7 - Mapeamento das cardinalidades (0, N) e (0, 1) de relacionamentos intemporais.....	109
FIGURA 6.8 - Mapeamento das cardinalidades (1, N) e (1, 1) de relacionamentos intemporais.....	110
FIGURA 6.9 - Mapeamento de atributos para rede de Petri .....	111
FIGURA 6.10 - Mapeamento de generalização/especialização temporal para rede de Petri.....	113
FIGURA 6.11 - Mapeamento de generalização/especialização intemporal para rede de Petri.....	115
FIGURA 6.12 - Mapeamento de generalização/especialização temporal do tipo disjunta (total e parcial).....	116
FIGURA 6.13 - Mapeamento de generalização/especialização temporal do tipo sobreposta (total e parcial).....	117
FIGURA 6.14 - Mapeamento de generalização/especialização intemporal do tipo disjunta (total e parcial).....	118
FIGURA 6.15 - Mapeamento de generalização/especialização intemporal do tipo sobreposta (total e parcial).....	119
FIGURA 6.16 - Mapeamento de fluxos externos para rede de Petri.....	121
FIGURA 6.17 - Mapeamento de fluxo de inclusão de entidade transitória para rede de Petri.....	123
FIGURA 6.18 - Mapeamento de fluxo de inclusão de entidade perene para rede de Petri.....	123
FIGURA 6.19 - Mapeamento de fluxo de inclusão de relacionamento temporal para rede de Petri.....	124
FIGURA 6.20 - Mapeamento de fluxo de inclusão de relacionamento intemporal para rede de Petri...	124
FIGURA 6.21 - Mapeamento de fluxo de ampliação de existência de entidade transitória para rede de Petri .....	125
FIGURA 6.22 - Mapeamento de fluxo de alteração atributos para rede de Petri .....	126

FIGURA 6.23 - Mapeamento do fluxo de verificação de presença de entidade.....	128
FIGURA 6.24 - Mapeamento do fluxo de verificação de presença de entidade transitória em um determinado período de tempo.....	128
FIGURA 6.25 - Fluxo de verificação de presença de relacionamento temporal.....	129
FIGURA 6.26 - Fluxo de verificação de presença de relacionamento intemporal.....	129
FIGURA 6.27 - Mapeamento de fluxo que manipula um conjunto variável de instâncias.....	130
FIGURA 6.28 - Mapeamento completo de uma transação TempER-Tr para rede de Petri.....	131
FIGURA 6.29 - Simplificação das fórmulas da rede de Petri resultante de um mapeamento.....	132
FIGURA 7.1 - Modelando o aparecimento de uma entidade perene.....	137
FIGURA 7.2 - Modelando a eliminação de entidade transitória.....	138
FIGURA 7.3 - Modelando o encerramento de existência de entidade transitória.....	139
FIGURA 7.4 - Modelando a reencarnação de entidade transitória.....	140
FIGURA 7.5 - Modelando a inclusão e exclusão de relacionamento temporal.....	141
FIGURA 7.6 - Modelando a alteração de um atributo temporal de entidade.....	142
FIGURA 7.7 - Modelando a alteração de atributo de relacionamento temporal.....	143
FIGURA 7.8 - Modelando a alteração de atributo intemporal de entidade.....	144
FIGURA 7.9 - Modelando a manipulação de conjuntos nos fluxos.....	145
FIGURA 7.10 - Rede de Petri da figura 7.9 após a simplificação das fórmulas.....	146

## **Lista das tabelas**

TABELA 4.1 - Características consideradas importantes a um modelo E-R Temporal.....	54
---	----

## Resumo

Neste trabalho é proposta a abordagem TempER-Tr, uma técnica de modelagem conceitual, fundamentada em rede de Petri, que integra a especificação das propriedades dinâmicas de um sistema a um modelo de dados temporal do tipo entidade-relacionamento.

Um modelo ou esquema conceitual descreve as propriedades identificadas de um sistema a ser desenvolvido. Estas propriedades podem ser classificadas em propriedades estáticas e propriedades dinâmicas. As propriedades estáticas descrevem os estados que o sistema pode alcançar, enquanto que as propriedades dinâmicas descrevem as transições entre estes estados. A modelagem conceitual das propriedades estáticas é normalmente conhecida como modelagem de dados. A modelagem das propriedades dinâmicas é denominada de modelagem funcional ou comportamental.

Mais especificamente, o modelo TempER-Tr é uma extensão de um trabalho anterior, conhecido como ER-Tr. No modelo ER-Tr, para descrever as propriedades estáticas de um sistema utiliza-se o modelo entidade-relacionamento convencional. No modelo TempER-Tr passa-se a adotar um modelo entidade-relacionamento temporal. Aliado a isto, uma nova linguagem de anotação, baseada em SQL, com mais poder de expressão é proposta.

O modelo entidade-relacionamento convencional não possui dispositivos de modelagem capazes de especificar restrições que envolvam a associação dos objetos com o tempo, exigindo que isto se faça ao nível da modelagem das propriedades dinâmicas. Em um modelo entidade-relacionamento convencional, os conjuntos de entidades e relacionamentos apresentam apenas duas dimensões: a primeira refere-se às instâncias (linhas) e a segunda aos atributos (colunas). Em uma abordagem entidade-relacionamento temporal, uma nova dimensão é acrescentada: o eixo temporal, possibilitando que as restrições temporais decorrentes da associação entre os objetos possam ser especificadas ao nível do modelo estático.

Um requisito importante a ser preenchido por um modelo de dados temporal é permitir que em um mesmo diagrama seja possível associar objetos (entidades, relacionamentos ou atributos) temporalizados com objetos não temporalizados. Isto porque em sistemas de informação alguns dados precisam ser explicitamente referenciados ao tempo e outros não, ou porque não mudam com o tempo, ou porque é irrelevante ao usuário saber quando os fatos ocorreram.

O modelo de dados temporal proposto neste trabalho, denominado TempER, pressupõe que todas as entidades, sejam elas temporalizadas ou não temporalizadas, apresentam uma "existência", ou seja, uma validade temporal. No caso das entidades temporalizadas esta existência é um subconjunto de pontos do eixo temporal. Em virtude disto são chamadas de entidades transitórias. Em relação às entidades não temporalizadas, é assumido que "existem sempre", ou seja, a sua validade temporal é constante, implícita e igual a todo o eixo temporal. Por isto são denominadas entidades perenes

Tanto as entidades transitórias quanto as entidades perenes, são focalizadas pelo modelo TempER através de duas perspectivas: uma intemporal e outra temporal. Através da perspectiva intemporal as entidades apresentam duas dimensões, à semelhança do que ocorre em um modelo entidade-relacionamento convencional. Através da perspectiva temporal as entidades apresentam três dimensões, as duas convencionais e mais o tempo.

Enquanto que o modelo de dados temporal descreve as propriedades estáticas de um sistema, o modelo comportamental, a outra face da abordagem TempER-Tr, focaliza as transações executadas no interior do sistema, em resposta a eventos que ocorrem no ambiente externo. Estas transações, quando efetivadas, provocam mudanças de estados no sistema. Entretanto, para estarem habilitadas a ocorrer, é necessário que um determinado conjunto de restrições dinâmicas sejam atendidas, o que se configura em um comportamento análogo ao de uma rede de Petri.

O modelo TempER-Tr é completamente mapeável, inclusive o modelo de dados temporal, para a rede CEM, um tipo de rede de Petri de alto nível. Isto permite que a sua semântica seja formalmente especificada e possibilita o aproveitamento das características das redes de Petri.

**Palavras-chave:** modelagem conceitual, modelagem temporal, dimensão temporal, especificação de transações, redes de Petri.

**Title:** “Temporal modeling of information systems: a Petri net based approach”

## Abstract

This dissertation presents TempER-Tr approach. TempER-Tr is a conceptual modeling technique based on Petri nets that integrates the specification of the dynamic properties of system to a temporal entity-relationship data model.

A model or conceptual schema describes the identified properties of a system. These properties can be classified into static and dynamic properties. The static properties describe the states that the system can reach, while the dynamic properties describe the transitions between the states. The conceptual modeling of the static properties is usually known as data modeling, while behavioral or functional modeling deals with dynamic properties.

The TempER-Tr model is an extension of a model known as ER-Tr. In the ER-Tr model, the conventional entity-relationship model is used to describe the static properties of a system. In the TempER-Tr model, it is adopted a kind of temporal entity-relationship model. In addition, a new notation language is proposed, based on SQL, with more expression power.

The conventional entity-relationship model doesn't provide tools to specify constraints that involve the association of objects with the time dimension, requiring that this have to be done at the dynamic properties modeling level. At the conventional entity-relationship model the entities and relationships sets present just two dimensions: the first one is related to the instance (lines) and the second to the attributes (columns). At a temporal entity-relationship approach, a new dimension is added: the time line. This way, the temporal constraints can be specified at the level of the static diagrams.

An important requirement to be supplied by any temporal data model is the possibility to relate, into the same diagram, time-varying objects with time-invariant objects. This is due to the fact that in information systems some data need to be explicitly related to time and others don't, either because they don't change with time, or because users don't need to know when the facts occurred.

The temporal data model proposed in this work, nominated TempER, presupposes that all entities, being them time-varying or time-invariant, have an “existence”, or a temporal validity. At the time-varying entities, named transitory entities, this existence is a subset of points from the time line. In time-invariant entities, named perennial entities, it is assumed that they “always exist”, i.e., their temporal validity is constant, implicit, and equal to all points of the time line.

Transitory entities, as much as perennial entities, are focused by the TempER model through two perspectives: a temporal perspective and a non-temporal perspective. Through the non-temporal perspective the entities present two dimensions - lines and columns - similar to a conventional entity-relationship model. Through the temporal perspective the entities present three dimensions: the two conventional dimensions and, in addition, the time dimension.

While the temporal data model describes the static properties of a system, the behavioral model in the TempER-Tr approach focus the transactions that are executed by the system, in response to the events that occur at the external environment. A certain set of dynamic constraints must be attended so that transactions are enable to occur. This configures a behavior similar to a Petri net.

The TempER-Tr model is completely mappable, inclusive the temporal data model, to the CEM net, a kind of high level Petri net. This way, the semantic of TempER-Tr model is formally specified. In addition, the utilization of the characteristics of Petri nets is possible.

**Keywords:** conceptual modeling, temporal modeling, time dimension, transaction specification, Petri nets.

# 1 Introdução

## 1.1 Especificação de sistemas de informação ao nível conceitual

Um modelo ou esquema conceitual descreve as propriedades identificadas de um sistema a ser desenvolvido. Estas propriedades podem ser classificadas em propriedades estáticas e propriedades dinâmicas. As propriedades estáticas descrevem os estados que o sistema pode alcançar, enquanto que as propriedades dinâmicas descrevem as transições entre estes estados. A modelagem conceitual das propriedades estáticas é normalmente conhecida como modelagem de dados. A modelagem das propriedades dinâmicas é denominada de modelagem funcional ou comportamental.

Conforme estabelecido por um grupo de trabalho da ISO [GRI 82], o modelo conceitual de um sistema deve especificar todas as propriedades de um sistema, tanto as estáticas quanto as dinâmicas, e não deve conter mais detalhes de implementação que os encontrados nos requerimentos.

O modelo TempER-Tr proposto nesta dissertação é uma ferramenta, ao nível da modelagem conceitual, que integra a especificação das propriedades dinâmicas de um sistema a um modelo de dados temporal. Trata-se de uma extensão do modelo ER-Tr [HEU 90, PER 90, HEU 91].

## 1.2 Vinculando os dados à dimensão temporal

Quando se utiliza um modelo ER convencional [CHE 76], a associação das entidades e relacionamentos com o tempo materializa-se normalmente através da inclusão de atributos comuns que armazenam datas, horas ou qualquer outra referência temporal. Além disso, caso se necessite registrar os diversos valores que um atributo pode apresentar ao longo do tempo, por exemplo os salários de um empregado, existem duas alternativas possíveis: ou se especifica uma nova entidade composta deste atributo e mais os atributos de referência ao tempo, ou se define tal atributo como composto e multivalorado.

Quando atributos que armazenam pontos do tempo são incluídos, é muito comum a necessidade de especificar restrições em relação a eles, tais como as abaixo descritas:

- Ao se transferir em 23/10/96 um empregado para um departamento, primeiro o empregado deve apresentar a sua data de admissão menor ou igual a 23/10/96 e não pode estar demitido, isto é, a data de demissão deve conter *null*; segundo, o departamento deve ter a sua data de criação menor ou igual a 23/10/96 e não pode estar fechado, isto é, a sua data de fechamento deve conter *null*.
- Um valor de salário deve ter o seu período de validade contido entre a data de admissão e a data de demissão do empregado, donde se deduz que não deve ser permitido aumentar o salário de um empregado após a sua data de demissão.
- Um empregado não pode estar lotado em mais que um departamento no mesmo ponto do tempo.

- Em todos os momentos do tempo que o empregado estiver vinculado à empresa, ele necessariamente deve estar lotado em um departamento, qualquer que seja ele.

O modelo ER convencional por si só não é capaz de especificar as restrições acima, exigindo que isto se faça ao nível da modelagem dinâmica.

Com o objetivo de dotar o modelo ER desta capacidade, algumas extensões temporais tem sido propostas, entre as quais: a abordagem ERT (Entity Relationship Time Model) [LOU 91], a abordagem TER (Temporal Entity-Relationship Model) descrita em [TAU 91] e a abordagem TEER (Temporal Enhanced Entity-Relationship Model) e a sua variante STEER descritas em [ELM 93] e [ELM 92] respectivamente.

Em um modelo de dados convencional, os conjuntos de entidades e relacionamentos apresentam duas dimensões: a primeira refere-se às instâncias (linhas) e a segunda aos atributos (colunas). Em um modelo ER temporal, uma nova dimensão é acrescentada: a dimensão temporal. A forma que tem se mostrado mais adequada de tratar a dimensão temporal em sistemas de informação é assumi-la como uma seqüência discreta, linear e finita de pontos consecutivos no tempo. A esta seqüência de pontos do tempo dá-se o nome de eixo temporal.

Dos modelos acima citados, o modelo TER [TAU 91] é o menos expressivo. Não associa propriamente os objetos modelados à dimensão temporal, ou seja, não oferece uma notação especial para representar entidades, relacionamentos e atributos temporalizados. Concentra-se apenas em desdobrar a representação das cardinalidades dos relacionamentos em cardinalidade *snapshot*, que determina o grau de participação da entidade no relacionamento a cada instante de tempo, e cardinalidade *lifetime*, que determina o grau de participação da entidade no relacionamento, tendo como unidade de referência a existência da entidade como um todo.

Um outro requisito importante a ser preenchido por um modelo de dados que incorpore a dimensão temporal é permitir que em um mesmo diagrama seja possível associar objetos (entidades, relacionamentos ou atributos) temporalizados com objetos não temporalizados. Isto se faz necessário porque em sistemas de informação, normalmente, alguns dados precisam ser explicitamente referenciados em relação ao tempo (a evolução dos salários dos empregados, a alocação de técnico em projetos, etc), e outros não apresentam esta necessidade, ou porque não mudam com o tempo, ou porque é irrelevante ao usuário saber quando os fatos ocorreram (o código de um empregado, a autoria de um artigo, etc). Normalmente os objetos não temporalizados são assumidos como existindo sempre, ou seja, adquirem uma validade temporal, implícita e constante, igual a todo o conjunto de pontos do eixo temporal.

O modelo TEER [ELM 93] não preenche o requisito acima pois assume que todos os objetos são temporalizados, isto é, são válidos por um certo período de tempo, o que simplifica bastante o modelo, mas exige que o modelador tenha que referenciar temporalmente até os dados que não precisam ser temporalizados.

A variante STEER [ELM 92] do modelo TEER, embora distinga os objetos em objetos conceituais e objetos temporais, em função da relação destes com o tempo, também não torna disponível um tratamento adequado aos objetos não temporalizados. Os objetos conceituais, que se caracterizam por uma validade temporal que inicia em um determinado momento e não mais se encerra, ainda continuam sendo temporalizados, pois a sua "existência" é um subconjunto de pontos do eixo temporal.

Dentre os modelos ER temporais acima citados, apenas o modelo ERT [LOU 91] permite mesclar em um mesmo diagrama objetos temporalizados e objetos não temporalizados. Contudo, apresenta alguns aspectos que dificultam o processo de modelagem.

No modelo ERT os atributos são explicitados graficamente, resultando em diagramas bastante carregados (ver figura 3.2). Esta característica contraria um dos principais objetivos que um modelo gráfico deve perseguir, que é o da facilidade de uso e visualização. Além disto, o modelo ERT utiliza como primitiva temporal o intervalo de tempo que, segundo [GAD 93], se mostra inferior à primitiva conhecida como “elemento temporal”.

O elemento temporal, uma união finita de intervalos de tempo, por ser fechado para as operações de união, interseção e complementação da teoria dos conjuntos, permite uma “substancial simplificação na habilidade do usuário de expressar consultas (*queries*) temporais” [GAD 88, 93].

Uma outra desvantagem dos intervalos de tempos, quando estes são utilizados como rótulos temporais dos objetos, é a necessidade de fragmentar em diversas tuplas os objetos “reencarnados”, uma tupla para cada intervalo de existência. Isto resulta em linguagens de difícil utilização, conforme mostra a comparação encontrada em [GAD 93] entre a linguagem Tquel, que utiliza intervalos, e a linguagem TempSQL, que utiliza elementos temporais.

Com a finalidade de preencher os requisitos acima citados, considerados necessários a uma abordagem ER que pretenda representar os objetos temporalmente, e com a intenção de suprimir os pontos desfavoráveis identificados nas propostas de modelagem temporal pesquisadas na literatura, é proposto nesta dissertação o modelo de dados **TempER**.

O modelo TempER permite representar a associação entre elementos temporalizados e não temporalizados. Para tanto adota o pressuposto que todas as entidades, sejam elas temporalizadas ou não, apresentam uma dimensão temporal, ou seja, uma “existência” ou validade temporal. No caso das entidades temporalizadas esta existência é um subconjunto de pontos do eixo temporal. Em virtude disto são chamadas de *entidades transitórias*. Em relação às entidades não temporalizadas, é assumido que “existem” durante todo o eixo temporal, ou seja, a sua validade temporal é constante, implícita e igual a todo o eixo temporal. Por isto são denominadas *entidades perenes*.

Qualquer que seja a classificação de uma entidade em relação ao tempo, seja transitória ou perene, ela sempre vai apresentar duas perspectivas: uma intemporal e uma temporal, como se fossem duas faces de uma mesma moeda (ver figuras 4.4, 4.7 e 4.9). Quando se focaliza os conjuntos de entidades pela perspectiva intemporal estes apresentam apenas duas dimensões (tuplas x atributos intemporais). Por outro lado, quando se focaliza estes mesmos conjuntos pela perspectiva temporal eles passam a apresentar três dimensões (tupla x atributos temporais x eixo temporal).

No tocante aos relacionamentos, ou as entidades se associam entre si na perspectiva temporal (relacionamentos temporais) ou na perspectiva intemporal (relacionamentos intemporais).

Especificamente em relação à abordagem ERT [LOU 91], que entre as técnicas citadas acima é a que mais se aproxima do modelo TempER, existem as seguintes implementações, que visam suprimir os aspectos considerados desfavoráveis:

- A primitiva temporal utilizada no modelo TempER é o *elemento temporal* e não o intervalo de tempo.
- No modelo TempER os atributos não são explicitados graficamente e sim através de um dicionário de dados associado ao diagrama ER, o que resulta em um modelo mais administrável visualmente.

Em resumo, o modelo de dados TempER apresenta uma série de extensões, em relação ao modelo ER convencional, que o dotam da capacidade de representar os objetos temporalmente.

### 1.3 Modelo dinâmico integrado ao modelo de dados temporal

Como já foi citado anteriormente, uma técnica de modelagem conceitual deve preencher os seguintes requisitos, conforme [GRI 82]:

- um modelo ou esquema conceitual deve especificar todas as propriedades de um sistema, tanto as estáticas como as dinâmicas;
- um modelo conceitual não deve conter detalhes de implementação, a não ser aqueles explicitamente declarados nos requerimentos.

Nos últimos anos têm surgido algumas abordagens propondo combinar modelagem Entidade-Relacionamento com algum tipo de rede de Petri de alto nível para especificar de forma combinada as propriedades estáticas e as propriedades dinâmicas de sistemas de informação. As redes de Petri de alto nível [BRA 87] são extensões das clássicas redes lugar/transição [PET 81, REI 85].

Na proposta encontrada em [KUN 89] são definidos dois modelos em diferentes níveis de abstração: um modelo informal chamado de modelo de interface, baseado nos DFD's [GAN 79] e um modelo formal denominado modelo comportamental, onde o comportamento é modelado de maneira similar às redes de Petri. Em [KAP 89] a rede de Petri utilizada, embora de alto nível, utiliza *tokens* e não possui uma linguagem de anotação.

Nenhum destes trabalhos, contudo, integra propriedades estáticas e dinâmicas somente com base da teoria das Redes de Petri. Visando suprir esta lacuna foi proposta a abordagem denominada ER/PN (Entidade-Relacionamento/Petri Net) [HEU 90, HEU 93]. Esta abordagem sustenta-se na possibilidade de mesclar em uma mesma rede *conexões vivas* com *conexões mortas*; as conexões vivas modelando os aspectos dinâmicos e as conexões mortas modelando os aspectos estáticos de um sistema.

Entretanto, um diagrama produzido via a abordagem ER/PN não mostra qual é a parte do sistema a ser implementada e qual é a parte que se constitui no ambiente da implementação. Aliado a isto, os diagramas ER/PN exigem do modelador uma certa fluência no formalismo rede de Petri, característica dificilmente encontrada entre os profissionais do mercado de trabalho. Buscando solucionar estes problemas, propôs-se o modelo ER-Tr (Entidade-Relacionamento-Transação) [HEU 90, PER 90, HEU 91], que se utiliza para tanto de diagramas semelhantes ao popular DFD [GAN 79].

Embora de entendimento e manuseio relativamente mais fácil, o modelo ER-Tr não dispensa o formalismo, sendo completamente mapeável para o modelo ER/PN.

Um dos aspectos do modelo ER-Tr que demanda um aperfeiçoamento é a sua linguagem de anotação. Quando um fluxo de uma transação especificada em ER-Tr refere-se a apenas um indivíduo (entidade ou relacionamento) a linguagem é satisfatória. Entretanto, quando existe a necessidade de tratar um conjunto de indivíduos em um fluxo, a linguagem carece de mecanismos mais consistentes e formais.

O presente trabalho tem por objetivo estender o modelo ER-Tr em duas direções. Primeiro, o modelo de dados utilizado passa a ser um modelo Entidade-Relacionamento temporal (o modelo TempER citado anteriormente). Em segundo lugar, a linguagem de anotação utilizada é substituída por outra fundamentada na declaração Select da linguagem SQL. Esta nova linguagem de anotação, inspirada em [HEU 9?], amplia a capacidade de tratar os pacotes de dados “movimentados” pelos fluxos das transações, pois aproveita-se da facilidade de manipular conjuntos inerente à linguagem SQL.

## 1.4 Estrutura do texto

Este texto está dividido em duas partes. A primeira parte, composta pelos capítulos 2 e 3, descreve de forma condensada as técnicas e conceitos encontrados na literatura, que serviram como ponto de partida para a concepção do modelo TempER-Tr, ou que com este guardam alguma correlação. A segunda parte, composta pelos capítulos 4, 5, 6 e 7 descreve, formaliza e exemplifica o modelo TempER-Tr, propriamente dito.

O capítulo 2 traz uma descrição condensada das abordagens ER-Tr e ER/PN que são as bases sobre as quais se elaborou o modelo TempER-Tr. Neste capítulo, também são descritas as redes de Petri elementares e as redes de Petri compactas condição/evento, que fundamentam as propostas ER/PN e ER-Tr.

O capítulo 3 discorre inicialmente sobre os conceitos relativos à dimensão temporal adotados no modelo TempER-Tr, definindo o que seja eixo temporal, intervalo de tempo, elemento temporal, assinalamento temporal, domínio temporal e restrição temporal. Em seguida, apresenta de forma resumida alguns modelos de dados temporais existentes na literatura técnica, que adotam a abordagem Entidade-Relacionamento. Em especial o modelo *Entity-Relationship-Time*, a fonte inspiradora do modelo de dados proposto nesta dissertação.

O capítulo 4 descreve o modelo de dados temporal TempER, que é a proposta desta dissertação no sentido de tornar disponível um modelo que permita especificar as propriedades estáticas de um sistema em relação a dimensão do tempo. Primeiramente é apresentada uma visão geral do modelo proposto, comparando-o com outros modelos ER temporais, e demonstrando as suas potencialidades em relação a um modelo ER convencional. Em seguida, discorre-se sobre o que se entende por perspectiva temporal e intemporal das entidades.

O núcleo do capítulo 4 é a descrição dos componentes do modelo TempER. É apresentada a notação gráfica a ser utilizada na modelagem de dados, bem como a forma como os elementos básicos (entidades, relacionamentos e atributos) são visualizados em função da dimensão temporal. Um novo tratamento às restrições de cardinalidade é

proposto, tendo em vista o relacionamento temporal entre as entidades. Por fim, é apresentado o desdobramento da estrutura generalização/especialização provocado pela adição da dimensão temporal às entidades. Visando facilitar o entendimento, todos os novos dispositivos de modelagem temporal propostos são mapeados em termos do modelo ER convencional.

O capítulo 5 é o ponto culminante da presente dissertação. É onde o modelo TempER-Tr é descrito. A ênfase deste capítulo é a modelagem das propriedades dinâmicas de um sistema, integradas ao modelo de dados temporal descrito anteriormente. O foco da modelagem dinâmica são as transações executadas no interior do sistema, acionadas a partir de eventos que ocorrem no ambiente externo. Estas transações, quando efetivadas, provocam mudanças de estados no sistema. Para que as transações estejam habilitadas a ocorrer, é necessário que as restrições por elas definidas estejam atendidas. Ou seja, as transações comportam-se como as alterações definidas pelas conexões das redes de Petri.

Ainda, no capítulo 5, encontra-se uma descrição dos elementos necessários para modelar uma transação: os eventos externos, os fluxos externos, os fluxos internos (de inclusão, de exclusão, de alteração, de verificação de presença/ausência, etc), bem como a nova linguagem de anotação dos termos e fórmulas. Também é explicado o comportamento de uma transação, ou seja, são detalhadas as regras de habilitação e as regras que regem o resultado de uma transação.

Em virtude de os dados do sistema apresentarem uma dimensão temporal, tanto os fluxos como a linguagem de anotação, no modelo TempER-Tr, oferecem mecanismos especiais para referenciamento temporal, como pode ser visto ao longo do capítulo 5.

No capítulo 6 é demonstrado que o modelo TempER-Tr é totalmente mapeável para uma rede de Petri, tal qual o modelo ER-Tr. O objetivo do mapeamento é, principalmente, formalizar a semântica do modelo TempER-Tr. Primeiramente, o modelo de dados temporal TempER é traduzido para rede de Petri. Em seguida, é a vez do modelo dinâmico.

O capítulo 7 apresenta alguns casos típicos de transações modeladas em TempER-Tr. O objetivo é mostrar através de exemplos algumas das possibilidades de uso da ferramenta. Para todas as transações modeladas é mostrada a correspondente tradução para rede de Petri.

Na conclusão da dissertação são apresentadas as contribuições proporcionadas pelo modelo TempER-Tr, bem como são apontados os possíveis estudos futuros visando aperfeiçoar a ferramenta.

## 2 Redes de Petri na modelagem de aspectos estáticos e dinâmicos de sistemas - retrospecto

A teoria de redes de Petri tem sido largamente utilizada na ciência da computação. Na área de modelagem de sistemas de informação, onde se insere o presente trabalho, as redes de Petri de alto nível [BRA 87] têm se mostrado mais adequadas. Este tipo de rede é uma extensão das clássicas redes lugar/transição [PET 81, REI 85].

As propriedades estáticas e dinâmicas de um sistema de informação podem ser modeladas através da integração do modelo Entidade-Relacionamento com o formalismo Rede de Petri de Alto Nível. Nos últimos anos têm surgido algumas abordagens propondo esta combinação de ferramentas na modelagem conceitual de sistemas. Na proposta encontrada em [KUN 89] são definidos dois modelos em diferentes níveis de abstração: um modelo informal chamado de modelo de interface, baseado nos DFD's [GAN 79] e um modelo formal denominado modelo comportamental, onde o comportamento é modelado de maneira similar às redes de Petri. Em [KAP 89] a rede de Petri utilizada, embora de alto nível, utiliza *tokens* e não possui uma linguagem de anotação.

Nenhum destes trabalhos, contudo, integra diagramas ER com redes de Petri somente com base da Teoria das Redes de Petri. Visando suprir esta lacuna foi proposta a abordagem denominada ER/PN (Entidade-Relacionamento/Petri Net) [HEU 90, HEU 93]. Esta abordagem sustenta-se na possibilidade de mesclar em uma mesma rede *conexões vivas* com *conexões mortas*; as conexões vivas modelando os aspectos dinâmicos e as conexões mortas modelando os aspectos estáticos de um sistema.

Entretanto, um diagrama produzido via a abordagem ER/PN não mostra qual é a parte do sistema a ser implementada e qual é a parte que se constitui no ambiente da implementação. Aliado a isto, os diagramas ER/PN exigem do modelador uma certa fluência no formalismo rede de Petri, característica dificilmente encontrada entre os profissionais do mercado de trabalho. Buscando solucionar estes problemas, propôs-se o modelo ER-Tr (Entidade-Relacionamento-Transação) [HEU 90, PER 90, HEU 91] que se utiliza para tanto de diagramas muito parecidos como o popular DFD [GAN 79]. Embora de entendimento e manuseio relativamente mais fácil, o modelo ER-Tr não dispensa o formalismo, sendo completamente mapeável para o modelo ER/PN.

Este capítulo ocupa-se primeiro em descrever as redes de Petri elementares e as redes de Petri compactas, do tipo condição/evento, que são os formalismos que fundamentam a abordagem ER/PN. Em seguida é mostrado de que forma as propriedades estáticas de um sistema podem ser evidenciadas através de conexões mortas e como se dá a tradução de diagramas ER convencionais para rede de Petri. Por fim, é apresentado um resumo do modelo ER-Tr, antecedido por uma descrição sucinta do modelo ER/PN.

## 2.1 Redes de Petri elementares

Uma rede de Petri elementar é um grafo que possui dois tipos de nós. O primeiro, representado por um pequeno círculo é denominado *lugar* e serve como depósito dos *tokens* ou marcas. O segundo, representado por um retângulo é conhecido como *conexão*. Os arcos que ligam os nós do grafo são conhecidos como *ramos*. Os ramos ligam lugares a conexões, nunca lugares a lugares ou conexões a conexões. A figura 2.1 mostra uma rede de Petri elementar do tipo condição evento, extraída de [HEU 90], que modela um mercado de trabalho cujo universo do discurso contém duas vagas ( $v1$  e  $v2$ ) e uma pessoa ( $p$ ). A rede apresentada contém os lugares:  $v1$  no mercado,  $v2$  no mercado,  $v1$  livre,  $v2$  livre,  $p$  detém  $v1$ ,  $p$  detém  $v2$ ,  $p$  livre e  $p$  no mercado e as dez conexões  $c1$  a  $c10$ . A conexão  $c1$  ao ser efetivada provoca o surgimento de um *token* ou marca nos lugares  $v1$  no mercado e  $v1$  livre, isto é, provoca uma alteração. A ocorrência de uma alteração afeta apenas os lugares ligados à conexão, fazendo com que surjam e/ou desapareçam marcas destes lugares, conforme estejam indicando as setas existentes nos ramos. Para que uma alteração possa ocorrer é necessário que a conexão esteja habilitada, isto é, que existam marcas presentes em todos os seus lugares de entrada e que os seus lugares de saída estejam vazios, isto é, as marcas estejam ausentes.

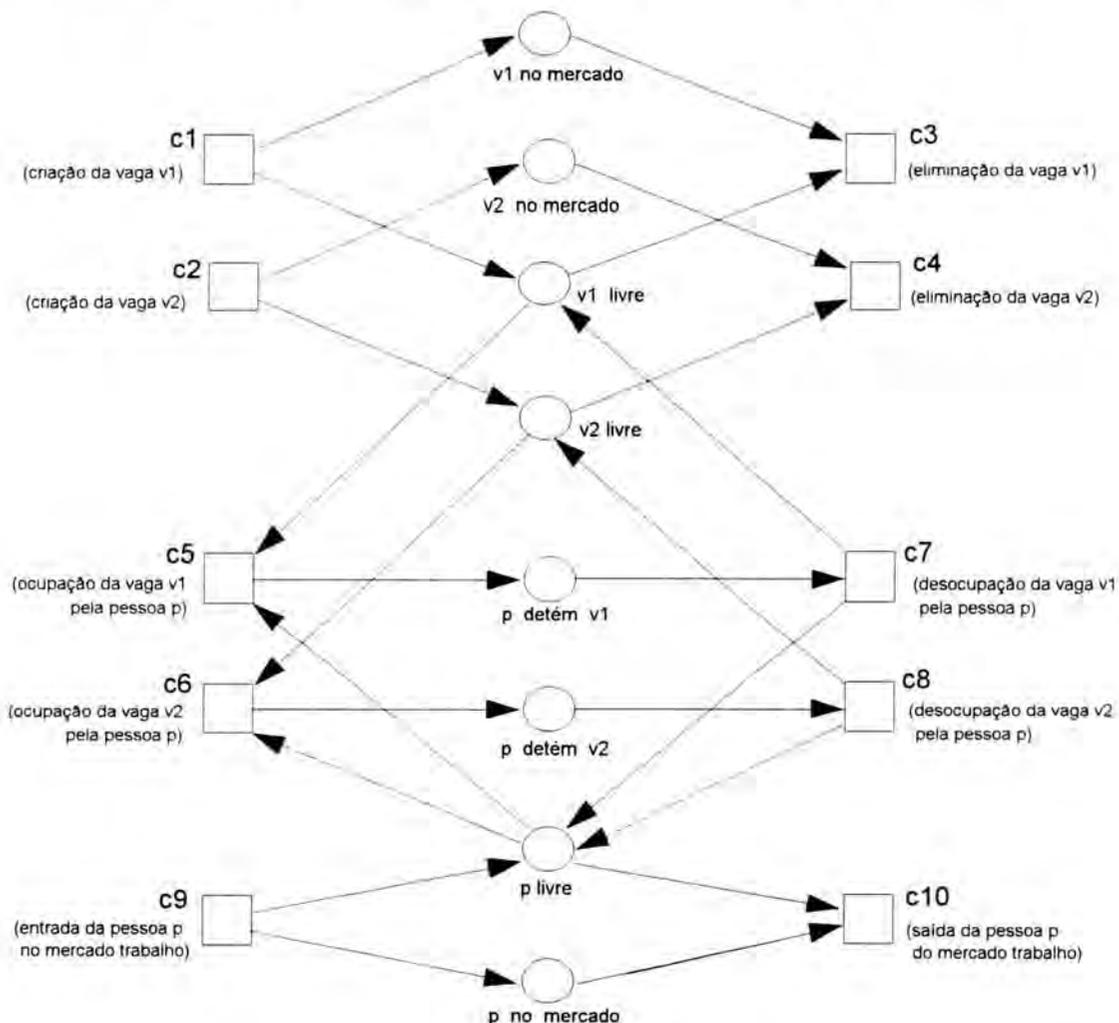


FIGURA 2.1 - Rede de Petri elementar

A fim de melhor esclarecer o conceito de habilitação, tome-se como exemplo a conexão *c5* da figura 2.1, que representa a ocupação da vaga *v1* pela pessoa *p*. A alteração definida por esta conexão, para ser efetivada, necessita que as seguintes condições de habilitação sejam verdadeiras:

- a) exista uma marca no lugar *p livre*, isto é, que a pessoa *p* esteja disponível.
- b) exista uma marca no lugar *v1 livre*, isto é, que a vaga *v1* não esteja ocupada.
- c) não exista marca no lugar *p detém v1*, isto é, este lugar deve estar vazio.

O conjunto de todos os lugares marcados (aqueles que possuem uma marca presente) é chamado de marcação da rede. Cada marca refere-se a um estado local do sistema modelado e a marcação representa o seu estado global. Cada vez que ocorre uma alteração na rede esta passa a apresentar uma nova marcação.

## 2.2 A abordagem condição/evento

Uma forma de traduzir os conceitos formais das redes de Petri elementares, visando adequá-las a tratar as peculiaridades de um sistema de informação, é através da abordagem condição/evento. Conforme está em [HEU 90] "dentro da interpretação condição/evento, cada uma das marcas representa uma condição no modelo; a presença de uma marca na rede representa o vigorar da correspondente condição, enquanto que a ausência representa o não vigorar da correspondente condição; cada alteração definida pela rede é interpretada como sendo um evento da realidade modelada ... tomando como exemplo o mesmo caso do mercado de trabalho verifica-se que as alterações representam os eventos do mercado de trabalho (Criar Vaga, Entrar no mercado, Fechar contrato, Rescindir contrato, etc) e as marcas existentes ou não nos lugares representam condições do mercado (vaga livre, contrato em vigor, etc)". A figura 2.2 mostra a correspondência entre a terminologia usada nas redes elementares e a terminologia dos modelos condição/evento.

Rede Elementar	Interpretação C/E
marca	condição
presença de marca	vigorar de condição
ausência de marca	não vigorar de condição
marcação da rede	estado do modelo
alteração	evento
ocorrência de alteração	ocorrência de evento
marca de entrada/saída	condição de entrada/saída

FIGURA 2.2 - Correspondência entre conceitos das redes elementares e da abordagem condição/evento

## 2.3 Redes de Petri compactas

As redes elementares mostram-se inadequadas para modelar sistemas de informação. Neste tipo de sistema existem normalmente grandes quantidades de entidades de uma mesma classe que são afetadas pelos mesmos tipos de eventos. As redes elementares resultantes seriam enormes e proibitivas, como pode ser deduzido a partir do exemplo do mercado de trabalho modelado na figura 2.1 caso viessem a ser modeladas centenas de vagas e centenas de pessoas em vez de se manipular apenas duas entidades ( $v1$  e  $v2$ ) da classe Vaga e uma entidade ( $p$ ) da classe Pessoa. Para resolver esta questão concebeu-se uma forma de condensar uma rede elementar em uma rede compacta. A rede resultante é conhecida como rede de Petri compacta do tipo condição/evento (rede CEM - *condition/event modeling*) [HEU 90].

Em vez dos *tokens* das redes de Petri clássicas, o que circula pelos lugares de uma rede compacta são as próprias entidades do universo de discurso do sistema. Além disto, enquanto que em uma rede elementar os lugares e as conexões identificam explicitamente as entidades a que se referem, como pode ser visto na figura 2.1, uma rede compacta generaliza lugares e conexões. Apoiada em uma linguagem de anotação auxiliar, permite referenciar em um mesmo lugar ou conexão todas as entidades que são afetadas por um determinado evento modelado. Uma descrição formal da rede de Petri compacta encontra-se [HEU 93].

Conforme [HEU 90] “as redes compactas, também conhecidas como redes de alto-nível, apareceram no final da década de 70, como resultado de pesquisas paralelas de dois grupos diferentes”. Uma das classes de redes compactas recebeu o nome de *redes predicado/transição* e apareceu pela primeira vez em [GEN 87]. A outra classe ficou conhecida como *redes coloridas*; estando descrita em [JEN 87]. As principais diferenças entre a rede compacta CEM e as redes de alto nível da literatura são as seguintes, transcrevendo [HEU 90]:

- As redes da literatura são redes com multiplicidade, ou seja, permitem que apareçam diversas cópias de um mesmo indivíduo em um lugar. Para obter o mesmo efeito nas redes CEM é necessário definir uma linguagem de anotação que trate multi-conjuntos, isto é, conjuntos que contêm diversas cópias de uma mesma entidade.
- As redes predicado/transição, na forma em que foram introduzidas originalmente, não permitem conjuntos de cardinalidade variável nos ramos, estando limitadas a um número fixo de termos designativos de indivíduos.
- A sintaxe da linguagem de anotação das redes que aparecem na literatura é um pouco diferente da utilizada neste texto, sendo que a das redes predicado/transição é a que mais se assemelha à das redes compactas CEM.

Na forma como foi concebida, uma rede compacta CEM sempre é traduzível para uma rede elementar, dita a sua subjacente.

As redes compactas só são possíveis porque aproveitam os padrões repetidos resultantes dos eventos da realidade modelada. Por exemplo, o evento *Rescindir contrato* de um sistema de mercado de trabalho é análogo para todas as pessoas empregadas. Se, para cada empregado este evento exigisse uma configuração diferente de rede, a compactação da rede teria resultado nulo pois produziria uma rede em relação a este evento de tamanho semelhante à rede elementar subjacente. Contudo, este tipo de

personalização de um evento é raro em sistemas de informação, fato que possibilita o emprego satisfatório da rede de Petri compacta.

A figura 2.3 extraída de [HEU 90] mostra uma rede compacta dentro da abordagem condição/evento que modela a aplicação do mercado do trabalho para quaisquer quantidades de vagas e pessoas que componham o universo do discurso. Como pode ser notado pelos nomes atribuídos aos lugares e conexões não se explicita mais quais vagas e pessoas são tratadas pela rede. Isto fica implicitamente indicado pelas variáveis anotadas junto aos ramos. Todas as variáveis de todos os ramos ligados a uma determinada conexão compõem o conjunto de variáveis desta conexão, as quais podem assumir qualquer valor dentro do especificado pelo domínio dos lugares que os respectivos ramos conectam. A conexão *Fechar contrato* da figura 2.3, por exemplo, envolve as variáveis  $v$  e  $p$ .

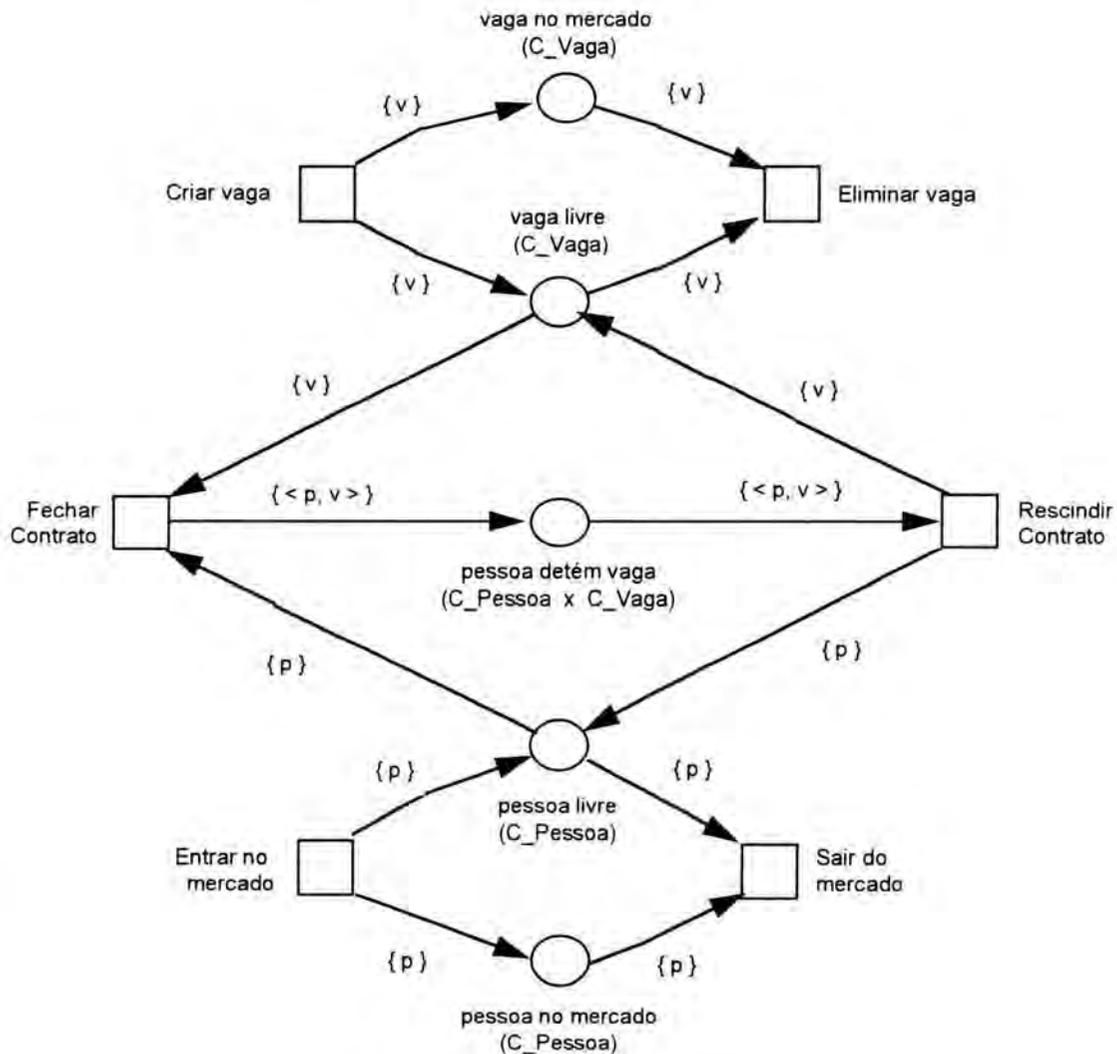


FIGURA 2.3 - Rede de Petri compacta

Supondo que o domínio  $C\_Vaga$  contivesse as entidades  $v1$  e  $v2$  e o domínio  $C\_Pessoa$  apenas a entidade  $p$ , a rede compacta da figura 2.3 seria equivalente à rede elementar modelada na figura 2.1, ou seja, a rede elementar da figura 2.1 seria a rede subjacente da mostrada na figura 2.3.

Na rede compacta em vez de marcas (*tokens*), o que vai ser "depositado" nos lugares são as próprias entidades. Por exemplo, caso a conexão *Criar vaga* seja efetivada para a vaga *vl*, esta vaga seria incluída no lugar *vaga no mercado* e no lugar *vaga livre*.

### 2.3.1 Tipos de ramos

Conforme o sentido da seta, os ramos são de entrada na conexão ou de saída da conexão. Ortogonalmente a isto, também classificam-se em ramos que especificam alterações nos lugares e ramos que especificam apenas consultas aos lugares, sem modificar o seu conteúdo. A figura 2.4 mostra as quatro possibilidades de ramos existentes.

Em uma rede compacta, um ramo alterador de entrada modela a exclusão de entidades do lugar conectado. Estas entidades são indicadas pelo termo anotado junto ao ramo. Um ramo alterador de saída, por sua vez, comporta-se de forma inversa, especificando a inclusão de entidades no lugar conectado.

Diferentemente dos ramos alteradores, os ramos restauradores não modificam o conteúdo dos lugares. O ramo restaurador de entrada serve para verificar se as entidades indicadas pela anotação estão presentes no lugar conectado; o ramo restaurador de saída, pelo contrário, serve para verificar se as entidades estão ausentes.

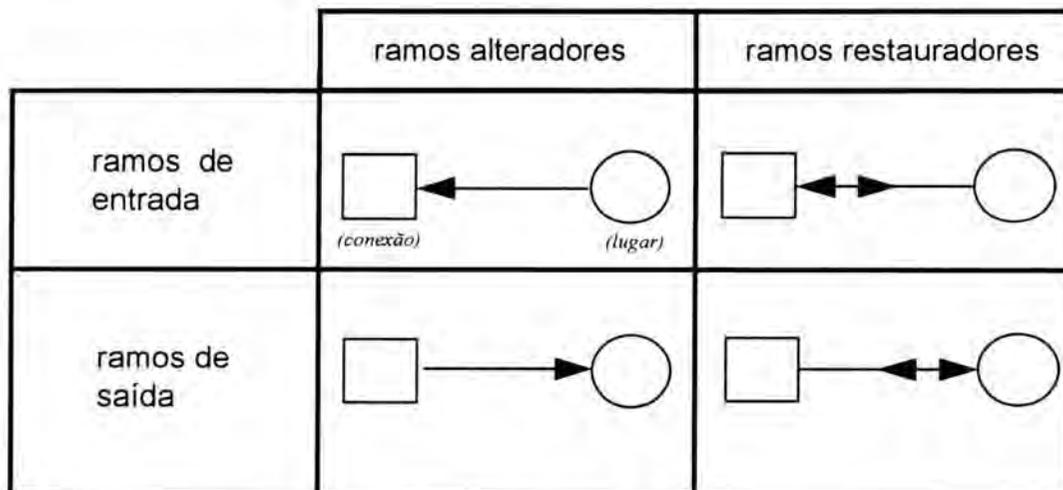


FIGURA 2.4 - Classificação dos ramos de uma rede de Petri

### 2.3.2 Linguagem de Anotação

Para condensar uma rede elementar em uma rede compacta é necessário o auxílio de uma linguagem de anotação que formalize as anotações dos lugares, conexões e ramos da rede.

Conforme [HEU 90], quando um modelo de sistema é criado, identifica-se na realidade modelada coisas, objetos que devem ser tratados por este modelo. A estes objetos dá-se a denominação de entidades. O conjunto de todas as entidades consideradas forma o Universo de Discurso (UD) deste modelo. Por exemplo, caso se estivesse modelando um sistema de contas correntes, o UD poderia conter: contas

correntes, saldos, lançamentos, pessoas, datas, agências bancárias e endereços, entre outras entidades. (É importante observar que o conceito de entidade aqui utilizado é mais amplo que o do modelo ER).

Para escrever expressões acerca das entidades que compõe o UD lança-se mão de uma Linguagem de Anotação (LA). Estas expressões são de dois tipos:

- *fórmulas*, que designam os valores lógicos Falso e Verdadeiro, e
- *termos*, que designam, cada um, uma entidade do UD.

Assim, “uma expressão que diga ‘*o saldo da conta 1025 é negativo*’, é uma fórmula, pois só pode designar um dos valores lógicos, Falso ou Verdadeiro. Já uma expressão que diga algo como ‘*o saldo da conta 1025*’ é um termo, pois designa uma das entidades do UD, no caso um saldo de conta corrente” [HEU 90].

A sintaxe e a semântica da Linguagem de Anotação da rede compacta encontra-se detalhada em [HEU 90]. As formas mais comuns de expressões da LA são:

- a) Individualização de uma entidade que pode ser feita colocando uma variável entre chaves. É o caso da variável *v* anotada junto ao ramo que liga a conexão *Fechar contrato* ao lugar *vaga livre* da figura 2.3. Uma variável é uma palavra composta exclusivamente de letras minúsculas.
- b) Especificação de conjuntos não fixos de entidades, representada através de variáveis que não estejam entre chaves. Por convenção devem ser representadas por palavras prefixadas pela partícula “*cj\_*”. Um exemplo deste tipo de variável é *cj\_artcongr* da figura 2.5, representando um determinado agrupamento de artigos submetidos a um congresso.
- c) Produto cartesiano entre dois termos cuja representação é a seguinte: <termo1> x <termo2>, onde termo1 e termo2 podem ser variáveis, resultados de funções, constantes, etc.
- d) Funções cuja representação é uma palavra contendo apenas letras maiúsculas em itálico. É o caso de *AUTORES* (*art*) que devolveria todos os autores que elaboraram o artigo indicado pela variável *art*.
- e) Constantes que simbolizam entidades específicas. Devem ser representadas por palavras entre apóstrofos, exceto quando a constante for numérica. Exemplo de constantes deste tipo: 'amarelo', 1995, 'Diretoria'.
- f) Constantes que representam domínios. São constantes especiais que devem ser simbolizadas por palavras antecidas pelo prefixo *C\_*. É o caso de *C\_Vaga* e *C\_Pessoa* da figura 2.3. Este termo representa o conjunto, que é de conteúdo constante, de todas as entidades de um determinado tipo. Geralmente são utilizadas para especificar o domínio de lugares e, quando for necessário, para especificar o conjunto inicial de entidades de um lugar. (Nesta dissertação dispensou-se o uso do prefixo “*C\_*”, como exemplifica os termos *OID* e *T* da figura 6.5 do capítulo 6).
- g) Especificação de entidades complexas compostas de outras entidades. Um exemplo é o elemento composto  $\{ \langle a, b, c, \dots, z, t \rangle \}$  presente nas redes da figura

6.5 (cap.6). A forma de representar elementos compostos de outros é colocá-los entre o sinal “<” e o sinal “>”.

### 2.3.3 Habilitação e efetivação de alterações

Em uma rede de Petri, para que uma alteração definida por uma conexão possa ocorrer, é necessário que um conjunto de pré-condições relacionadas a ela sejam verdadeiras, caso contrário tal alteração é considerada não habilitada.

Em relação às redes de Petri compactas, as regras que condicionam a habilitação das alterações definidas por uma conexão são as seguintes:

- a) As entidades definidas nos ramos de entrada da conexão, sejam eles alteradores ou restauradores, devem estar presentes nos respectivos lugares.
- b) As entidades definidas nos ramos de saída da conexão, sejam eles alteradores ou restauradores, devem estar ausentes dos respectivos lugares.
- c) A fórmula da conexão, caso presente, deve resultar, sob a valoração em questão, em verdadeiro.

Só após cumpridas as regras de habilitação, uma alteração pode ocorrer, fazendo com que o conjunto de entidades especificado em cada ramo alterador de entrada desapareça do lugar respectivo e o conjunto de entidades especificado em cada ramo alterador de saída seja incluído no lugar respectivo. Os ramos restauradores por sua vez não provocam qualquer modificação no conteúdo dos lugares, servem apenas para “consultá-los”.

### 2.3.4 Referenciando todas as entidades de um lugar.

Muitas vezes em um sistema surge a necessidade de tratar todas as entidades que estejam presentes ou ausentes em um determinado lugar. A forma de fazer isto é anotar junto ao ramo correspondente um termo especial utilizando a cláusula MAXSUB, que se encontra formalmente especificada em [HEU 92]. Um exemplo disto pode ser visto na figura 2.5 (a) onde a conexão *Obter todos os artigos de um congresso* referencia todos os artigos submetidos ao congresso especificado pela variável *congr*, isto é, todos os artigos referentes ao congresso *congr* que estão presentes no lugar *artigos submetidos*. Esta forma especial de anotação equivale à representada na figura 2.5 (b), do que se deduz que a notação MAXSUB substitui dois ramos da notação convencional.

Entre as possíveis alterações definidas pela conexão da 2.5 (a) apenas uma vai estar habilitada. Trata-se daquela, cuja variável *cj\_artcongr* refira-se a todos os artigos vinculados ao congresso, indicado pela variável *congr*, que estejam presentes no lugar *Artigos submetidos*.

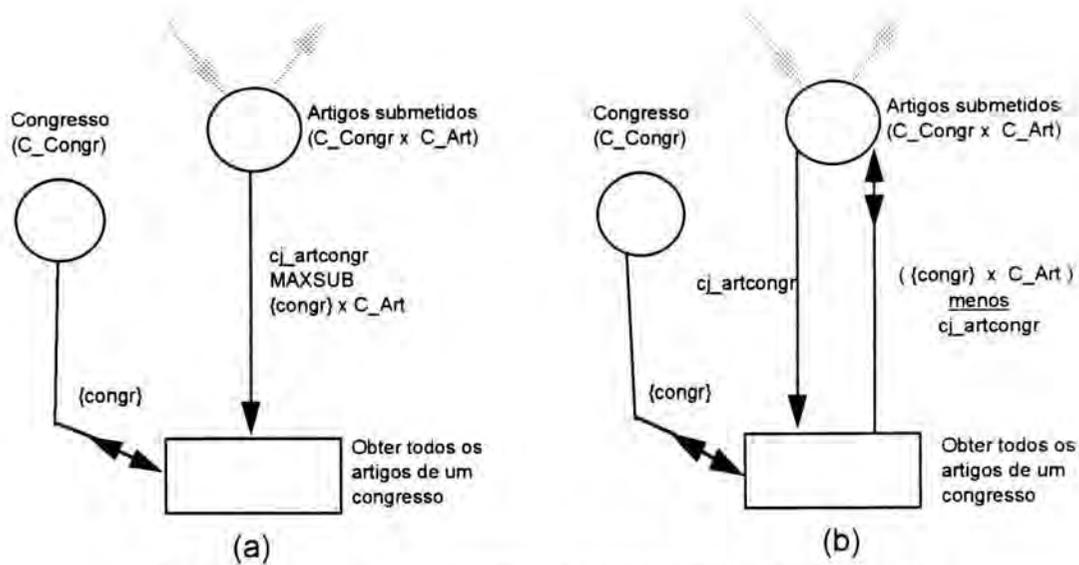


FIGURA 2.5 - Referenciando todas as entidades presentes em um lugar

Um exemplo mostrando como referenciar todas as entidades de um dado conjunto que estejam *ausentes* de um determinado lugar, encontra-se na figura 2.6 (a). Dentre as possíveis alterações definidas pela conexão *Receber artigos*, apenas uma estará habilitada. Será aquela cuja variável *cj\_ausentes* refira-se ao subconjunto de autores do artigo que ainda não estejam presentes no lugar *Autores*. Em outras palavras, esta conexão modela a inclusão daqueles autores do artigo que ainda não estejam cadastrados. Parte-se do pressuposto que a função *AUTORES(art)* devolve o conjunto de todos os autores de um artigo. O segmento de rede mostrado em 2.6 (a) equivale ao mostrado em 2.6 (b).

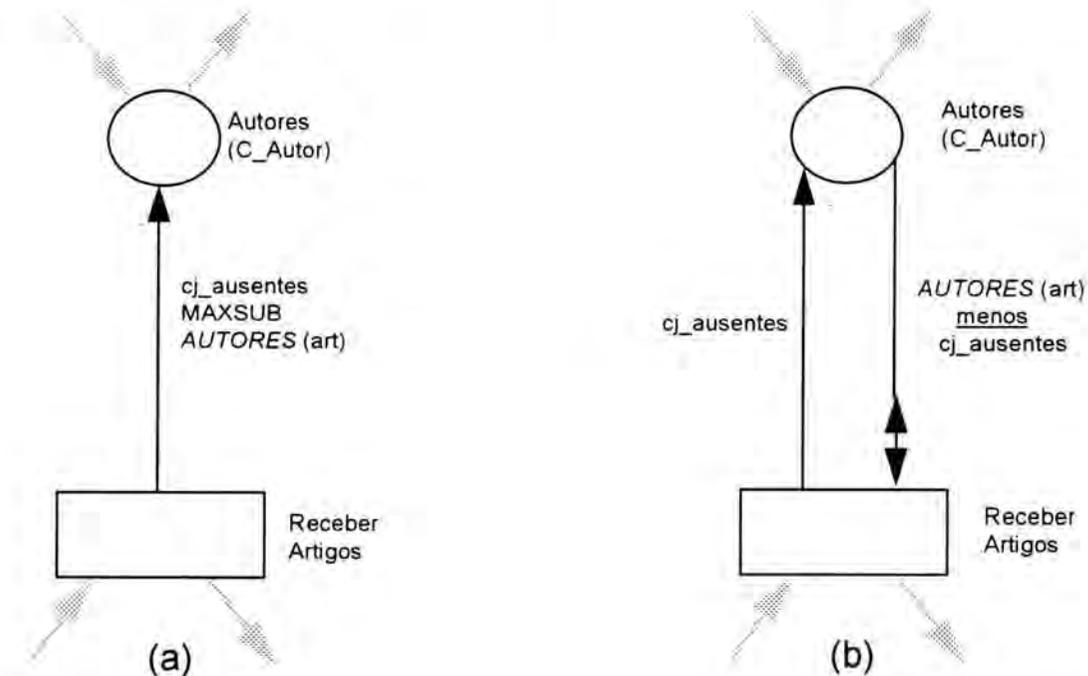


FIGURA 2.6 - Referenciando todas as entidades ausentes de um lugar

## 2.4 Evidenciando propriedades estáticas através de redes de Petri

Conforme [HEU 93] há algumas propriedades no sistema que não são explicitamente descritas através de uma rede de Petri mas podem ser encontradas através da simulação a partir da marcação inicial ou através de outra técnica de análise. Algumas destas propriedades implícitas são propriedades estáticas, cuja identificação é fundamental para o projeto das estruturas de dados, quando o sistema modelado pela rede vier a ser implementado. Um exemplo destas propriedades implícitas é aquela que define que nunca uma vaga pode estar livre e ocupada ao mesmo tempo.

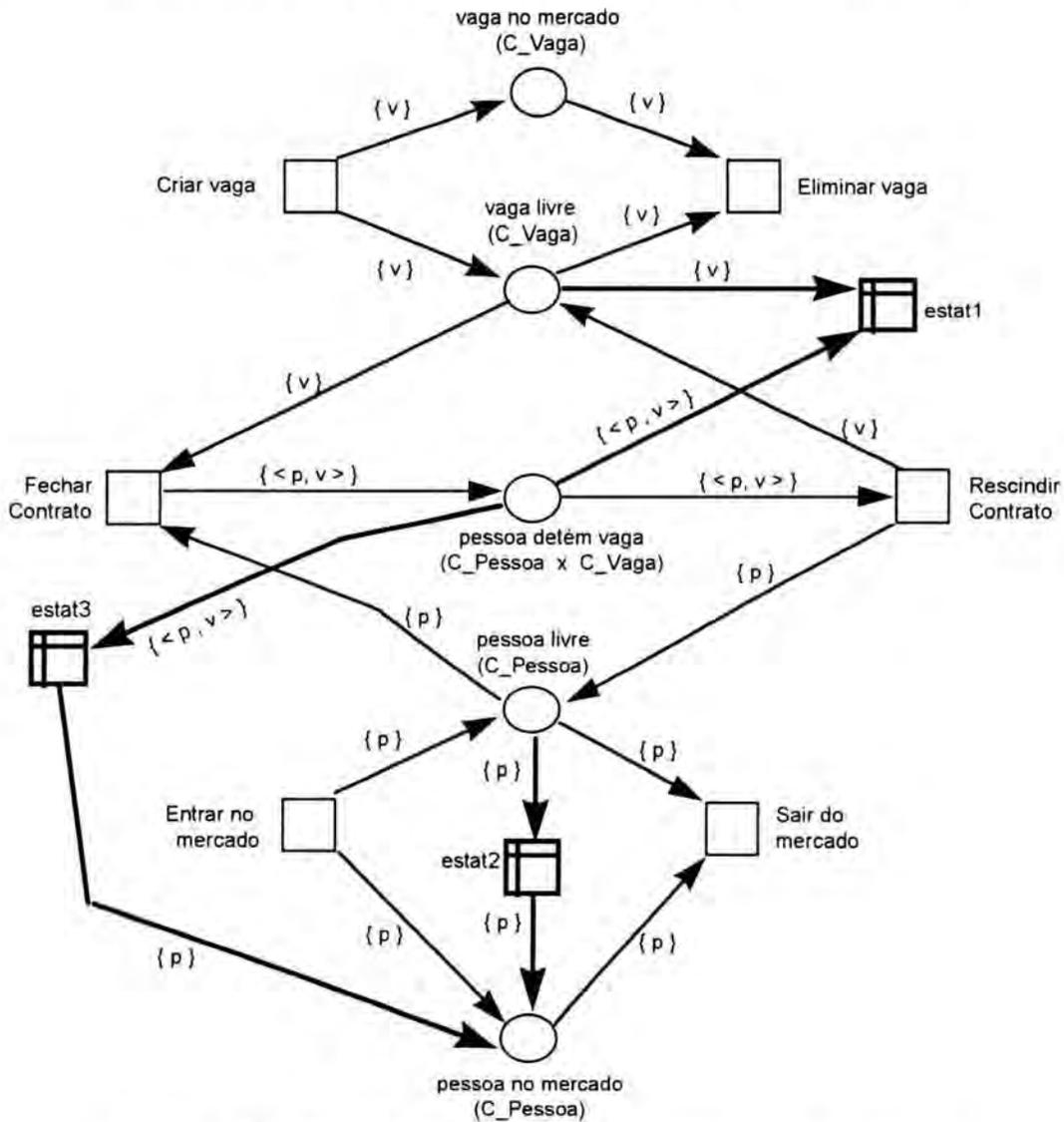


FIGURA 2.7 - Evidenciando propriedades estáticas em uma rede de Petri

Uma forma de evidenciar as propriedades estáticas de uma rede é através da adição de um tipo especial de conexão, conhecida como *conexão morta*, cujas alterações jamais estarão habilitadas, seja qual for a marcação que venha a ser alcançada. As conexões mortas são representadas por um retângulo contendo um F estilizado, pois

estas representam “fatos”. É o caso da conexão *estat1* da figura 2.7 que modela a mútua exclusividade entre vaga livre e vaga ocupada.

A conexão morta *estat2* evidencia o fato de que uma pessoa que esteja no lugar *pessoa livre* necessariamente também deve estar no lugar *pessoa no mercado*. A conexão morta *estat3* por sua vez, mostra que, se uma pessoa está detendo uma vaga, então esta pessoa deve estar presente no mercado de trabalho.

Além de simplesmente evidenciar propriedades implícitas de uma rede de Petri, as conexões mortas podem também ser utilizadas com o propósito de especificação, *permitindo restringir o conjunto de marcações alcançáveis* através da exclusão das indesejáveis. Utilizadas desta forma, as conexões mortas possibilitam que se especifique através de uma rede de Petri as restrições estáticas relativas ao modelo de dados do sistema. É explorando esta característica que um modelo do tipo Entidade-Relacionamento pode ser totalmente traduzido para uma rede de Petri, como será explicado a seguir.

## 2.5 Tradução do modelo ER para rede de Petri

Na tradução de um diagrama Entidade-Relacionamento para rede de Petri, como mostra a figura 2.8, cada conjunto de entidades e de relacionamentos vai corresponder a um lugar na rede. As restrições de cardinalidade dos relacionamentos, por sua vez, serão mapeadas para conexões mortas.

Pela ótica da abordagem Entidade-Relacionamento, os elementos e propriedades básicos de um sistema são: as entidades, os relacionamentos e os atributos. As restrições especificáveis via o modelo ER são, portanto, aquelas que definem a forma como tais elementos e propriedades podem ser combinados entre si. Destas restrições, a mais característica é a que define a cardinalidade de participação de entidades em relacionamentos. Como neste trabalho é adotado o princípio de participação das entidades nos relacionamentos, o mesmo empregado em [BAT 92], a leitura das anotações de cardinalidade é da seguinte forma, tomando como exemplo o diagrama ER da figura 2.8: uma instância que esteja presente no conjunto de entidades *Empregado* participa de no mínimo uma e no máximo uma instância do conjunto de relacionamentos *Lotação*. Por outro lado, uma instância que esteja presente no conjunto de entidades *Depto* não precisa participar de instâncias de *Lotação*, ou pode estar participando de uma ou mais.

A obrigatoriedade de participação de um empregado em pelo menos uma lotação é modelada pela cardinalidade mínima igual a 1 e corresponde à conexão morta *cm2* na rede de Petri encontrada na parte de baixo da figura 2.8. Como, por outro lado, esta participação se restringe a no máximo uma vez (cardinalidade máxima igual a 1), um empregado não pode estar lotado a mais que um departamento. Esta segunda restrição é traduzida para a conexão morta *cm3*.

O fato de uma entidade estar participando de um relacionamento implica que ela necessariamente deve estar presente no respectivo conjunto de entidades. Esta restrição, implícita ao nível do modelo ER, é mapeada para uma conexão morta semelhante às conexões *cm1* e *cm4*.

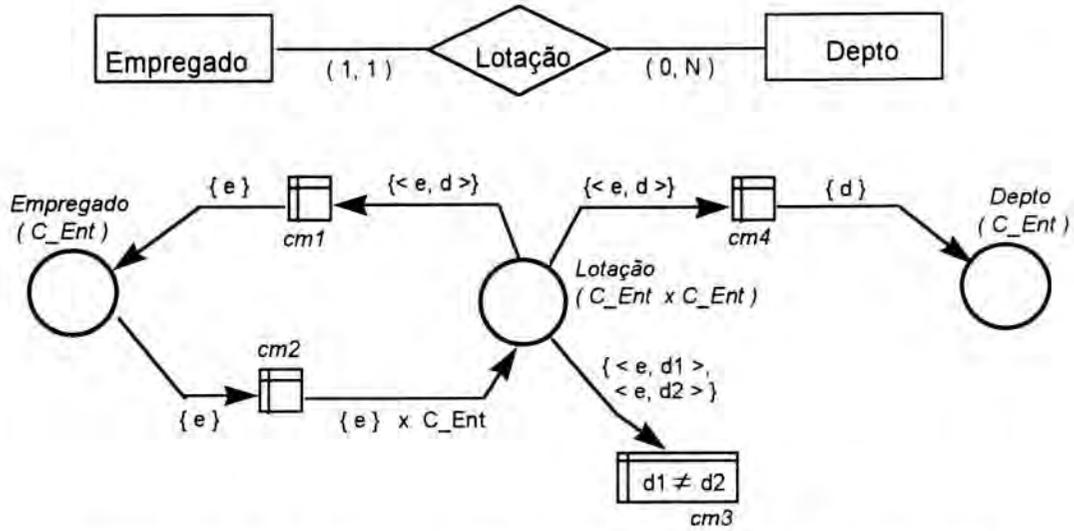


FIGURA 2.8 - Tradução do modelo ER para rede de Petri através de conexões mortas

O domínio do lugar na rede de Petri referente a um conjunto de entidades é o conjunto de todas as entidades do universo do discurso do sistema, sendo representado pela constante  $C\_Ent$ . O domínio do lugar que mapeia relacionamentos é o produto cartesiano de conjuntos  $C\_Ent$ .

Quanto à tradução dos atributos para rede de Petri CEM, um exemplo é mostrado na figura 2.9, abaixo.

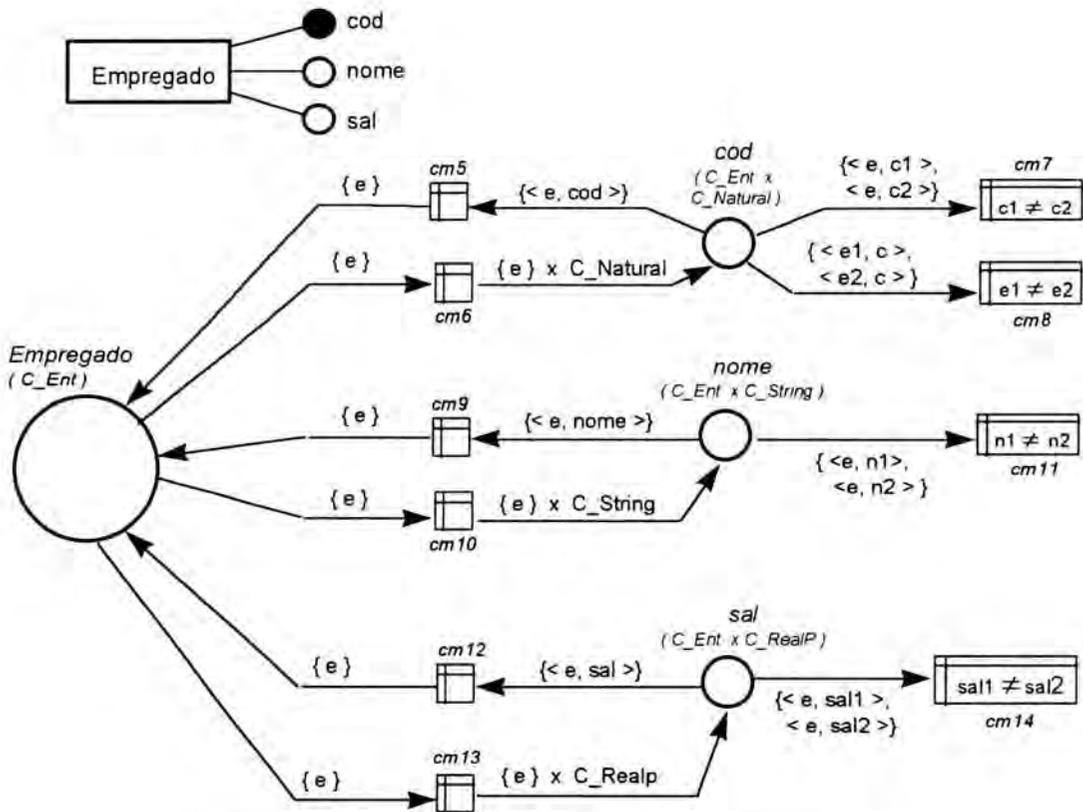


FIGURA 2.9 - Tradução de atributos para rede de Petri

Os atributos do diagrama acima são todos obrigatórios e monovalorados. A obrigatoriedade é representada na rede de Petri por uma conexão morta semelhante às conexões *cm6*, *cm10* e *cm13*. A monovaloração é representada por uma conexão morta semelhante às conexões *cm7*, *cm11* e *cm14*. Por sua vez, as conexões mortas *cm5*, *cm9* e *cm12* mapeam a necessidade da existência da entidade caso esta esteja associada a um valor.

Um atributo que seja especificado como identificador (chave primária) deve apresentar um valor único no contexto de todas as entidades pertencentes a um certo conjunto. Esta restrição é traduzida para uma conexão morta semelhante à conexão *cm8*, a qual estabelece que dois empregados diferentes, apresentando o mesmo código, nunca estarão presentes no lugar referente ao atributo *cod*.

## 2.6 Modelo integrado ER/PN

Tendo sido definido a semântica do modelo Entidade-Relacionamento em termos de uma rede de Petri, é possível obter uma ferramenta de modelagem que combine as duas técnicas e seja totalmente mapeável para uma rede de Petri. Esta ferramenta seria capaz então, de representar em um mesmo diagrama as propriedades estáticas e dinâmicas de um sistema, de maneira formal e consistente. Foi com base neste raciocínio que se concebeu a abordagem ER/PN [HEU 90, 93].

A figura 2.10 apresenta um diagrama ER/PN onde as conexões “vivas” *AdmitirEmpregado* e *TransferirEmpregado* estão combinadas com o diagrama ER especificado anteriormente nas figuras 2.8 e 2.9.

Para estabelecer a ordem em que as entidades participam de um relacionamento é necessária uma indicação adicional, semelhante aos dois pequenos círculos contendo um número cada, que aparecem próximo ao conjunto *Lotação* na figura 2.10.

## 2.7 Modelo ER-Tr

Embora capaz de modelar de forma integrada as propriedades estáticas e dinâmicas de um sistema, o modelo ER/PN, descrito acima, é de difícil manuseio por pessoas não familiarizadas com as redes de Petri. Além disto, um diagrama produzido via a abordagem ER/PN não mostra qual é a parte do sistema a ser implementada e qual é a parte que se constitui no ambiente da implementação.

Buscando solucionar os problemas acima citados, foram envidados esforços de pesquisa que culminaram no modelo ER-Tr [HEU 90, PER 90, HEU 91]. Uma das principais providências adotadas foi mapear o conceito de conexão para o conceito de transação, mais familiar entre os modeladores de sistema de informação. Para tanto lançou-se mão de diagramas muito semelhantes aos DFD's [GAN 79].

Uma conexão do modelo ER/PN é traduzida na verdade para um conjunto de transações no modelo ER-Tr. Portanto cada transação individualizada, ou seja, cada transformação que possa ocorrer, corresponde a uma determinada alteração, dentre as definidas pela conexão.



efetivação. No capítulo 5, onde a extensão temporal do modelo ER-Tr proposta nesta dissertação é descrita, encontram-se mais detalhes sobre estas regras.

Nos exemplos das figuras 2.11 e 2.13 o modelo de dados apresentado anteriormente sofre uma modificação: datas foram incluídas visando principalmente o armazenamento do histórico das lotações e a cardinalidade máxima de participação de um empregado no conjunto de relacionamentos *Lotação* foi alterada para “N”.

Uma transação do conjunto de transações *AdmitirEmpregado* da figura 2.11, estará habilitada a ocorrer quando os parâmetros de entrada estiverem presentes no fluxo externo, o empregado indicado pela variável *emp* estiver ausente do conjunto *Empregado*, o departamento especificado pela variável *dep* (cuja sigla seja igual ao parâmetro *esigdep* informado) esteja presente no conjunto *Depto* e o relacionamento entre o empregado representado pela variável *emp* e o departamento representado pela variável *dep* esteja ausente do conjunto *Lotação*. A ocorrência da transação, ou seja, a transformação resultante desta, provocará o aparecimento do empregado indicado pela variável *emp* no conjunto *Empregado* e o aparecimento do relacionamento indicado pelo par  $\langle emp, dep \rangle$  no conjunto *Lotação*.

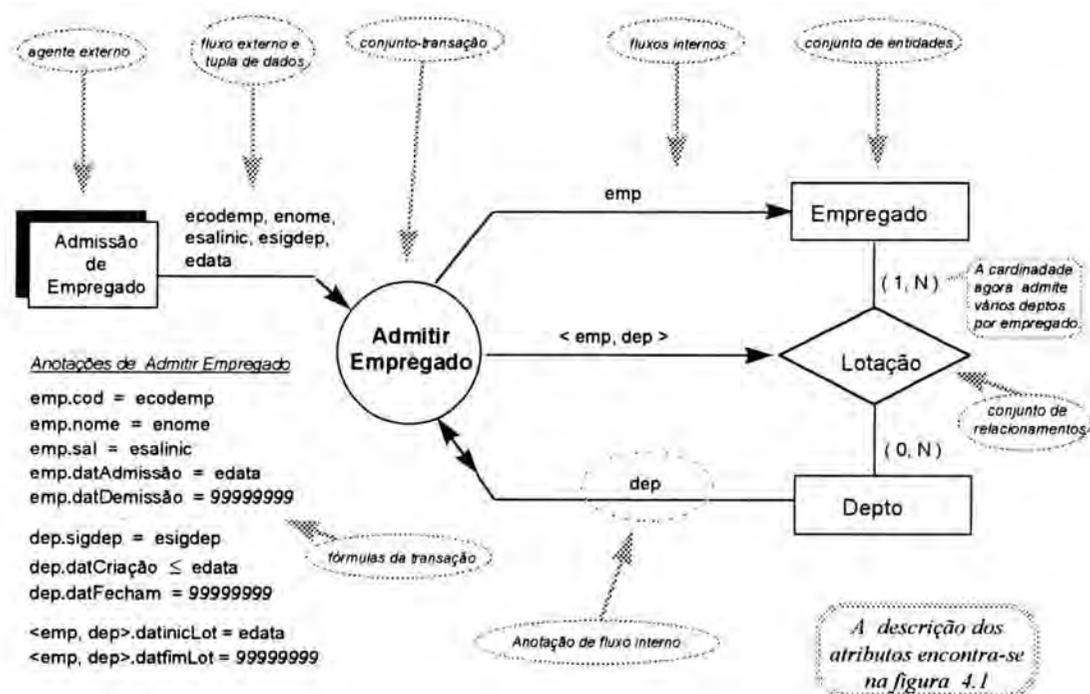


FIGURA 2.11 - Diagrama ER-Tr modelando aparecimento de entidade e relacionamento

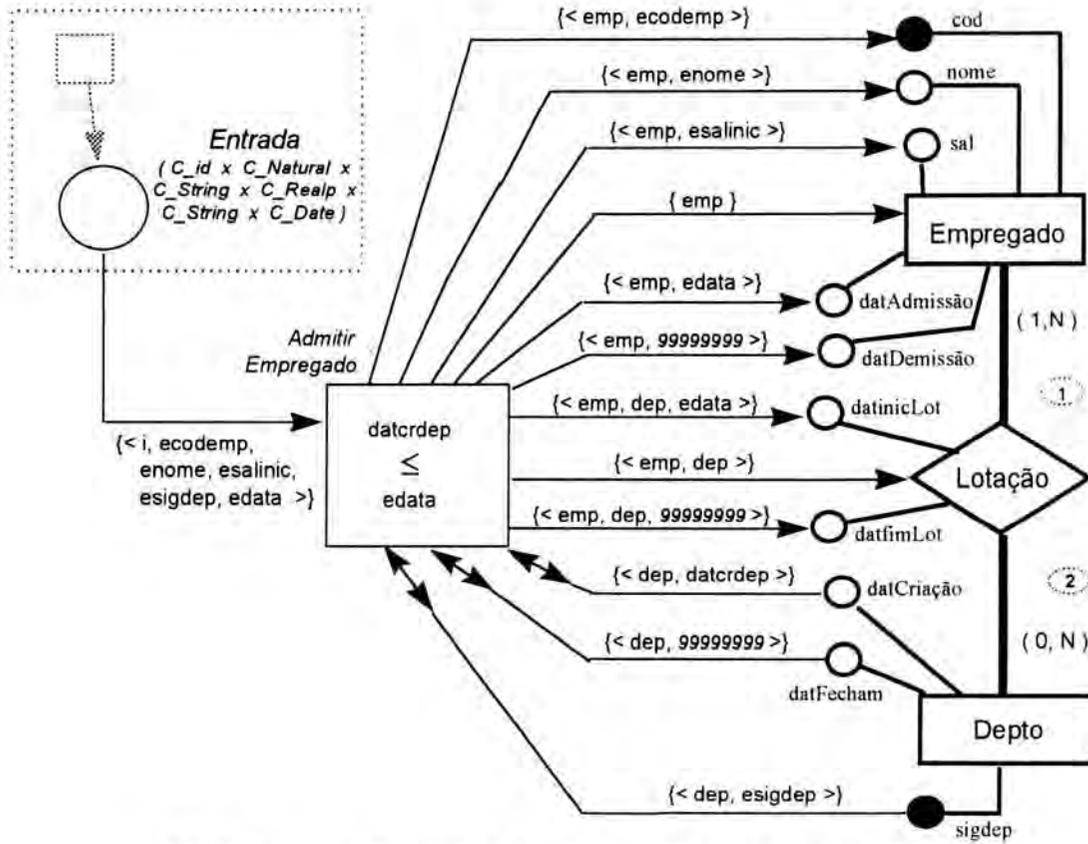


FIGURA 2.12 - Mapeamento de diagrama ER-Tr para o modelo ER/PN

A transferência de um certo empregado, de um departamento para outro, é modelado na figura 2.13.

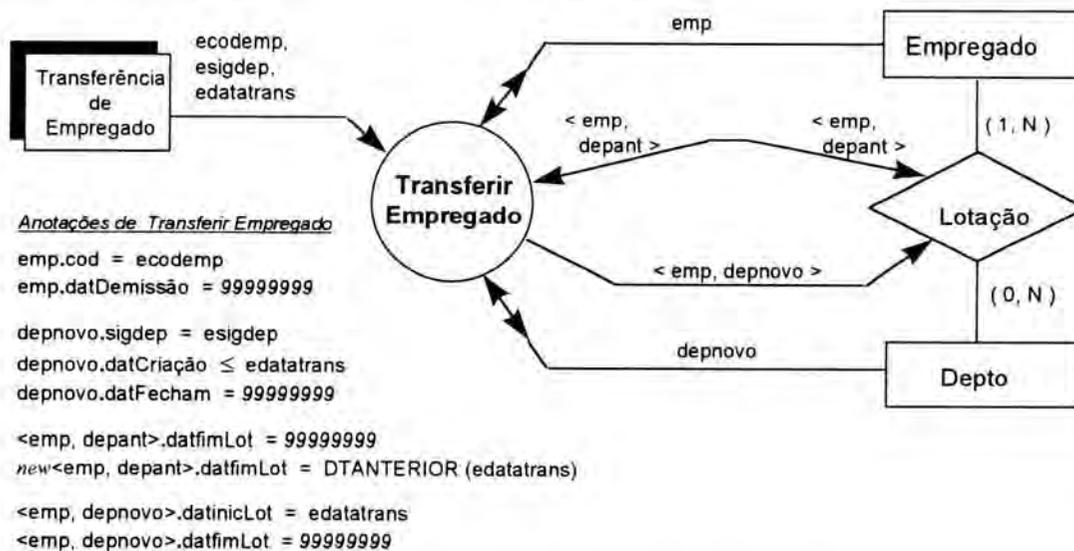


FIGURA 2.13 - Diagrama ER-Tr modelando alteração e o registro de histórico de relacionamento

Uma transação do conjunto *TransferirEmpregado* para estar habilitada exige que o empregado não esteja demitido (*datDemissão = 99999999*), que o novo departamento ao qual o empregado será alocado esteja vigente desde a data da transferência e que o empregado ainda não tenha sido lotado neste novo departamento. O efeito desta transação é o seguinte: a lotação anterior em que o empregado estava participando é fechada (*datfimLot* fica igual a *edatatrans* menos um dia) e uma nova lotação surge vinculando o empregado ao novo departamento.

O modelo de dados utilizado nas figuras 2.11 e 2.13 apresenta alguns problemas decorrentes da tentativa de dotá-lo de capacidade de registrar a evolução temporal das lotações, a saber:

1. A modificação da cardinalidade máxima para “N” referente à participação de empregados no conjunto de relacionamentos *Lotação* provocou um efeito colateral indesejável: agora é possível que ao mesmo tempo um empregado esteja lotado em dois ou mais departamentos. Evitar que isto ocorra fica por conta de restrições ao nível do modelo dinâmico, através do uso de fórmulas.
2. Mesmo podendo registrar histórico, o modelo ER em questão impede que um empregado volte a estar lotado em um departamento por onde já passou. Isto porque o conjunto *Lotação*, como é da natureza dos conjuntos, não pode conter duas vezes o mesmo par <empregado, depto>. Para resolver isto os modeladores geralmente transformam o conjunto de relacionamentos em conjunto de entidades, dotando este novo conjunto de uma chave primária composta por uma tripla, formada por um empregado, um departamento e um atributo adicional, possivelmente a data de início da lotação.

Estes problemas são solucionados quando se emprega um modelo de dados que incorpora a dimensão temporal, tal qual o proposto no capítulo 4.

### 3 A dimensão temporal - conceitos e modelos

Na primeira seção deste capítulo são definidos os conceitos relativos à dimensão temporal utilizados nos capítulos seguintes. Na segunda seção, tendo em vista que se optou pela abordagem ER para representar as propriedades estáticas de um sistema, são descritos alguns modelos Entidade-Relacionamento que apresentam extensão temporal.

#### 3.1 Conceitos relativos à dimensão temporal

Os conceitos abaixo definidos, oriundos de estudos da área de bancos de dados temporais, são os principais utilizados pelo modelo proposto nesta dissertação.

##### 3.1.1 Eixo temporal

A forma que se tem mostrado mais adequada de tratar a dimensão temporal em sistemas de informação é assumi-la como uma seqüência discreta, linear e finita de pontos consecutivos do tempo. Esta seqüência também recebe a denominação de *eixo temporal* e é representada no âmbito deste trabalho pela constante  $\mathbf{T}$ .

Na área de estudo dos bancos de dados temporais [TAN 93, CLI 95] normalmente consideram-se duas linhas de tempo ortogonais: o tempo de validade e o tempo de transação. O tempo de validade de um fato, conforme o glossário encontrado em [JCS 94], é o tempo quando o fato é verdadeiro na realidade modelada. Normalmente o tempo de validade é fornecido pelo usuário. O *tempo de transação*, conforme o mesmo glossário, é quando o fato é registrado no banco de dados, sendo o respectivo rótulo temporal suprido automaticamente pelo sistema gerenciador de banco de dados. Para efeito da ferramenta de modelagem proposta neste trabalho, existe apenas um eixo temporal e este representa o *tempo de validade*. Não é necessário referir-se ao tempo de transação porque os modelos produzidos estão no estágio conceitual, ou seja, não devem especificar aspectos inerentes à implementação física.

Para definir com mais precisão o eixo temporal, lança-se mão da teoria dos conjuntos [ALE 85] como segue.

Seja  $\mathbf{T}$  o conjunto não vazio de todos os pontos do tempo, então por definição  $\mathbf{T}$  é um *conjunto totalmente ordenado* pela relação BEFORE a qual satisfaz à seguinte condição, onde  $t_a$  e  $t_b$  são pontos do tempo:

$$\forall t_a, t_b : t_a, t_b \in \mathbf{T} \wedge t_a \neq t_b \rightarrow (t_a \text{ BEFORE } t_b \vee t_b \text{ BEFORE } t_a)$$

Para que a relação BEFORE seja uma relação de ordem estrita total é necessário que possua as seguintes propriedades:

*Irreflexibilidade:*

$$\forall t : t \in \mathbf{T} \rightarrow \neg(t \text{ BEFORE } t)$$

*Transitividade:*

$$\forall t_a, t_b, t_c : t_a, t_b, t_c \in \mathbf{T} \wedge t_a \text{ BEFORE } t_b \wedge t_b \text{ BEFORE } t_c \rightarrow t_a \text{ BEFORE } t_c$$

*Assimetria:*

$$\forall t_a, t_b : t_a, t_b \in \mathbf{T} \wedge t_a \text{ BEFORE } t_b \rightarrow \neg(t_b \text{ BEFORE } t_a)$$

A relação BEFORE é equivalente à relação “<” utilizada no âmbito dos números inteiros.

Quando se considera a dimensão temporal como sendo um eixo discreto assume-se que cada ponto do tempo pertencente a este eixo é atômico e dura exatamente um *chronon*. O *chronon* define a granularidade da dimensão temporal. Em um UD (Universo do Discurso) é possível coexistirem diferentes granularidades, por exemplo: em um determinado segmento modelado a granularidade pode ser diária (o *chronon* equivale a um dia) e em outro segmento a granularidade pode ser mensal. Para efeito do presente trabalho será assumido que o UD apresenta apenas uma única granularidade. Em [CLI 88] encontra-se uma forma de representar as diversas granularidades do calendário e as suas inter-relações.

O conceito de *chronon* e as propriedades acima especificadas estabelecendo que o eixo temporal  $\mathbf{T}$  é totalmente ordenado, permitem tratá-lo como sendo isomórfico ao conjunto dos número inteiros. Assim, entre dois pontos do tempo consecutivos não existe outro ponto do tempo. Caso o tempo fosse assumido como isomórfico ao conjunto dos números reais, entre dois pontos sempre existiria um outro ponto.

Os pontos limitrofes do eixo temporal são referenciados através de símbolos especiais. O limite inferior é representado pelo símbolo “«” e o limite superior é representado pelo símbolo “»”.

### 3.1.2 Intervalo de tempo

Um intervalo de tempo é um subconjunto de pontos do eixo temporal  $\mathbf{T}$ . Como por indução também é *totalmente ordenado* pela relação BEFORE, é possível através dos operadores first e last [CLI 88] extrair-lhe o primeiro e o último ponto de tempo. É o que se passa a demonstrar.

Seja  $\mathbf{I}$ , um intervalo de tempo e  $\mathbf{I} \subseteq \mathbf{T}$ , então:

$$\text{first}(\mathbf{I}) \text{ é o elemento } t \in \mathbf{I} \text{ tal que, } \forall t' \in \mathbf{I} : t \text{ BEFORE } t' \vee t = t'$$

$$\text{last}(\mathbf{I}) \text{ é o elemento } t \in \mathbf{I} \text{ tal que, } \forall t' \in \mathbf{I} : t' \text{ BEFORE } t \vee t' = t$$

Qualquer intervalo de tempo é identificável pelo seus pontos limitrofes que correspondem aos pontos definidos pelos operadores first e last acima definidos. A expressão que representa um intervalo de tempo tem o formato  $[t_{\text{FIRST}}, t_{\text{LAST}}]$ , onde  $t_{\text{FIRST}}$  é o primeiro ponto do intervalo e  $t_{\text{LAST}}$  é o último. O próprio eixo temporal  $\mathbf{T}$  pode ser considerado um intervalo de tempo, identificado pela expressão [«, »]. Um intervalo de tempo que dura exatamente um *chronon* apresenta  $t_{\text{FIRST}}$  igual a  $t_{\text{LAST}}$ .

Para que um conjunto de pontos do tempo seja realmente considerado um intervalo é necessário que sejam consecutivos, isto é não pode haver qualquer lacuna entre eles. Esta condição é formalmente representada pela expressão abaixo:

Seja  $I \subseteq T$  um intervalo,

então,  $\forall t_a \in I: t_a \neq \text{last}(I) \rightarrow$

$$\exists t_b \in I: (t_a \text{ BEFORE } t_b \wedge$$

$$\neg \exists t_c \in T: t_a \text{ BEFORE } t_c \wedge t_c \text{ BEFORE } t_b)$$

### 3.1.3 Elemento Temporal

Conforme o glossário produzido por um conjunto de especialistas da área de banco de dados temporal [JCS 94], elemento temporal é uma *união finita de intervalos de tempo*. São fechados para as operações de união, interseção e complementação da teoria dos conjuntos, isto é, qualquer destas operações sobre elementos temporais produz um novo elemento temporal. Como estas operações encontram contrapartida nos operadores booleanos *or*, *and* e *not*, isto produz uma substancial simplificação na habilidade do usuário de expressar consultas temporais [GAD 88, 93]. Dado que todos os intervalos são subconjuntos do eixo temporal  $T$ , então, por derivação, um elemento temporal também o é. Um exemplo de elemento temporal é  $[25, 40] \cup [51, 70]$ . Um intervalo é obviamente um elemento temporal. Um instante  $t$  pode ser identificado pelo intervalo unitário  $[t, t]$  e assim também é considerado um elemento temporal.

Na ferramenta de modelagem proposta nesta dissertação foi adotado o elemento temporal para servir de rótulo do tempo para as entidades, relacionamentos e atributos do modelo de dados. O elemento temporal se mostra superior ao uso da primitiva intervalo de tempo, primeiro porque os intervalos não são fechados para as operações da teoria de conjuntos mencionadas acima, e segundo porque, conforme [GAD 88, 93], quando os intervalos são usados como rótulos temporais os objetos são fragmentados em várias tuplas, uma para cada intervalo. Isto resulta em linguagens de difícil utilização, conforme mostra uma comparação, também encontrada em [GAD 93], entre a linguagem TQuel, que utiliza intervalos, e a linguagem TempSQL, que utiliza elementos temporais. Também a linguagem TSQL2 [SNO 94], que é uma extensão temporal do padrão SQL-92, utiliza o elemento temporal como a primitiva para rotular as tuplas das tabelas.

### 3.1.4 Assinalamento Temporal

Para capturar a evolução de valores de um atributo, [GAD 93] introduz a noção de assinalamento temporal. Um assinalamento temporal (ou simplesmente assinalamento) de um atributo  $A$  é uma função do elemento temporal para  $dom(A)$ , o domínio de valores de  $A$ . Por exemplo, a expressão  $\langle \text{vermelho } [25,32], \text{ azul } [33, \infty] \rangle$  é um assinalamento temporal do atributo  $COR$  e o seu significado é: a  $COR$  foi vermelha durante o intervalo de tempo  $[25,32]$  e azul durante os instantes  $[33, \infty]$  (33 em diante).

O assinalamento temporal é bastante utilizado nas fórmulas que referenciam atributos temporais encontradas nas transações do modelo TempER-Tr, como mostram as figuras 5.1 do capítulo 5 (terceira fórmula das anotações), 7.1 e 7.4 do capítulo 7. No capítulo 5, na subseção 5.3.3.5 encontram-se mais considerações a respeito do significado de um assinalamento temporal.

### 3.1.5 Domínio Temporal

Conforme [GAD 93], o domínio de um assinalamento  $\lambda$ , denotado  $\llbracket \lambda \rrbracket$ , é chamado de domínio temporal deste assinalamento. Por exemplo, se  $\lambda$  representar o assinalamento acima do atributo *COR*, então  $\llbracket \lambda \rrbracket = [25, \infty]$ . No último parágrafo da subseção 5.3.3.5, do capítulo 5, existem mais explicações sobre o conceito de domínio temporal.

### 3.1.6 Restrição Temporal

A restrição temporal pode ser entendida como uma operação que se faz sobre um assinalamento temporal visando obter um segmento deste. Ou seja, dado o assinalamento temporal  $\lambda$  acima, a expressão  $\lambda \upharpoonright \mu$  denota a restrição de  $\lambda$  ao elemento temporal  $\mu$ . Por exemplo,  $\lambda \upharpoonright [29,34]$  é igual a  $\langle \text{vermelho } [29,32], \text{ azul } [33,34] \rangle$ .

Na modelagem de transações descrita no capítulo 5 desta dissertação, é utilizado um mecanismo de restrição temporal equivalente ao acima citado, para delimitar a amplitude temporal dos pacotes de dados manipulados pelos fluxos. Este mecanismo se caracteriza pelo uso do operador  $\otimes$ . Na figura 5.1 do capítulo 5, por exemplo, o pacote de dados associado ao fluxo de inclusão, conectado ao conjunto de entidades *Empregado*, está restrito ao intervalo de tempo  $[inicio, \infty]$ .

## 3.2 Modelos Entidade-Relacionamento temporais encontrados na literatura

Nesta seção será apresentada uma visão geral de alguns modelos Entidade-Relacionamento estendidos, capazes de especificar o relacionamento dos dados com a dimensão temporal.

### 3.2.1 O modelo ERT

O modelo ERT (Entity-Relationship-Time) [LOU 91, a, b], a principal fonte inspiradora da proposta TempER apresentada no capítulo 4, é um dos componentes do projeto TEMPORA [LOU 90]. Este modelo de dados oferece uma série de dispositivos que permitem modelar aplicações complexas de banco de dados. Especificamente, ele possibilita a modelagem explícita do tempo e uma taxionomia para tratar hierarquias e objetos complexos, como mostra a figura 3.1.

O conceito mais primitivo em ERT é o de classe que é definido como uma coleção de objetos individuais que possuem propriedades em comum, isto é, são do mesmo tipo. Em um esquema ERT apenas as classes de objetos são especificadas. Em adição, cada relacionamento é visualizado como um conjunto nomeado de dois (entidade ou valor, papel) pares onde cada papel expressa a maneira que uma entidade ou valor específico é envolvido na associação.

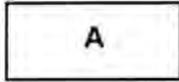
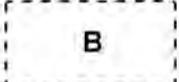
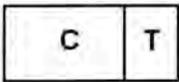
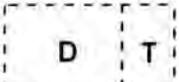
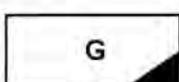
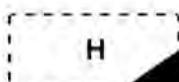
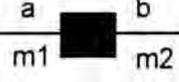
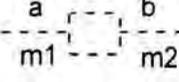
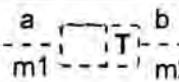
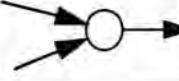
Símbolos utilizados pela abordagem ERT		
Notação Gráfica ERT		Interpretação
 		Classe de entidade A e classe de entidade derivada B.
 		Classe de entidade C e classe de entidade derivada D com marca de tempo. T é um período de tempo simbólico.
 		Classe de entidade complexa E e classe valor complexo F. Modelam agregações de objetos ou atributos.
 		Classe de valor simples G e classe de valor derivado H. Podem estar relacionados com nodos do tipo A, B, C, D, E e F. Modelam os atributos.
 		Relacionamentos binários que conectam nodos do tipo A, B, C, D, E e F - a e b são nomes de relacionamento (a inverso b), m1 e m2 indicam o mapeamento no formato (x:y) onde x e y são inteiros não-negativos ou N. As caixas brancas representam relacionamentos derivados.
 		Relacionamentos binários com marca de tempo. T é um período de tempo simbólico.
 		Relacionamentos ISA - círculo preto: total; círculo branco: parcial. Várias setas apontando para o mesmo círculo indicam subconjuntos disjuntos.

FIGURA 3.1 - Simbologia do modelo ERT

A primitiva temporal adotada em ERT é o intervalo de tempo. O período de tempo em que uma entidade é modelada chama-se *período de existência*. O período em que um relacionamento é válido denomina-se *período de validade*. Como mostra a figura 3.2, as classes são representadas por retângulos, aos quais é anexada uma marca de tempo (a caixa com a letra T), quando se trata de entidades que variam temporalmente, por exemplo a classe *Funcionário*. As classes de relacionamentos quando são temporais também apresentam esta marca de tempo. Os atributos (sempre mostrados de forma explícita nos diagramas) estão ligados às classes de entidades através de relacionamentos (temporais ou não). É o caso dos atributos *salário* e *nome* da classe *Funcionário* na figura 3.2.

No tocante aos relacionamentos, os seguintes axiomas se aplicam:

- I. Uma entidade para participar em um relacionamento deve estar presente na população da classe de entidades especificada no relacionamento. Em adição, o período de validade do relacionamento deve ser um sub-período da interseção dos períodos de existência das entidades associadas.

2. Cada entidade da população de uma subclasse tem uma instância correspondente na população de sua superclasse. Além disso, o período de existência da entidade especializada deve ser um sub-período do período de existência da entidade generalizada.

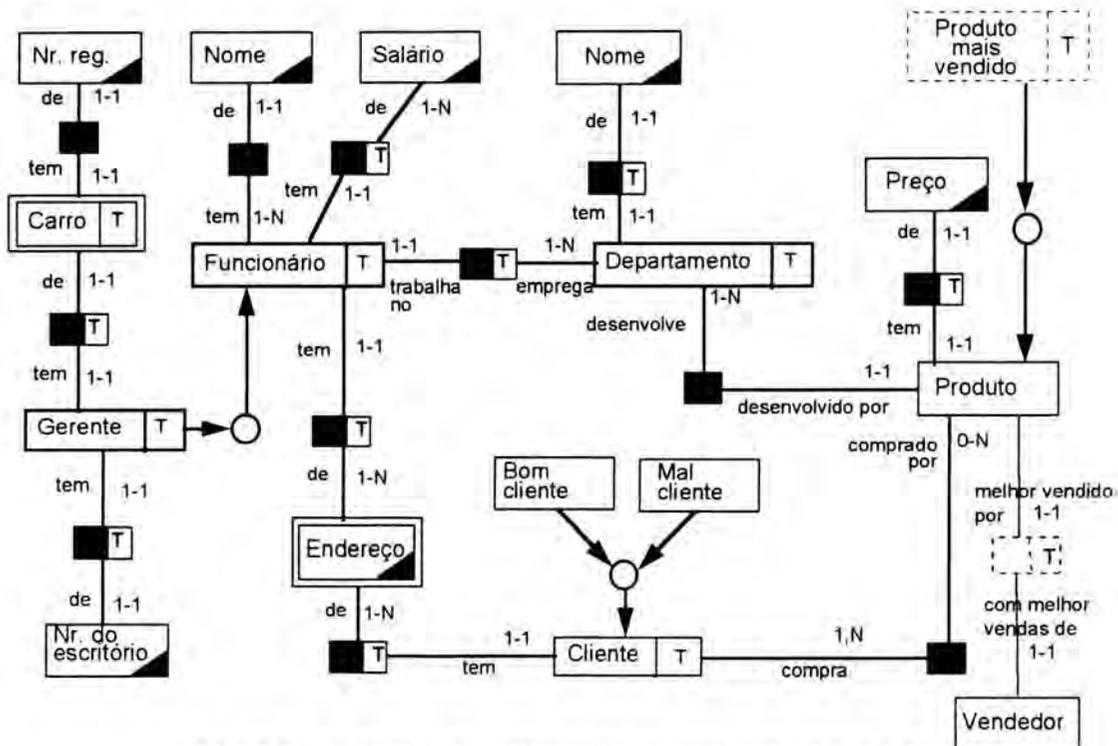


FIGURA 3.2 - Modelo de dados temporal segundo a abordagem ERT

A modelagem de informações usando períodos de tempo (intervalos) ocorre da seguinte forma: Cada objeto que varia no tempo é associado a uma instância da *classe de períodos simbólicos*. Instâncias desta classe são identificadores únicos gerados pelo sistema (*surrogates*), por exemplo: *SP001*, *SP002*, etc, os quais são relacionados entre si através de uma das treze relações entre intervalos de tempo especificadas em [ALL 83] (ver figura 3.4).

A figura 3.3 mostra através de um metamodelo como a dimensão temporal é tratada em ERT. As duas subclasses subordinadas à classe *PeriodoDeTempo* são disjuntas. Isto porque os períodos simbólicos são usados para modelar informação de tempo relativo, enquanto que os períodos de calendário modelam informação de tempo absoluto. O símbolo  $\beta$  representa o relacionamento temporal entre os períodos de tempo e o símbolo  $\beta\tau$  representa o seu inverso. Além disso, os períodos de tempo iniciam em um *tick* e terminam em um *tick* e possuem uma duração expressa em *ticks*.

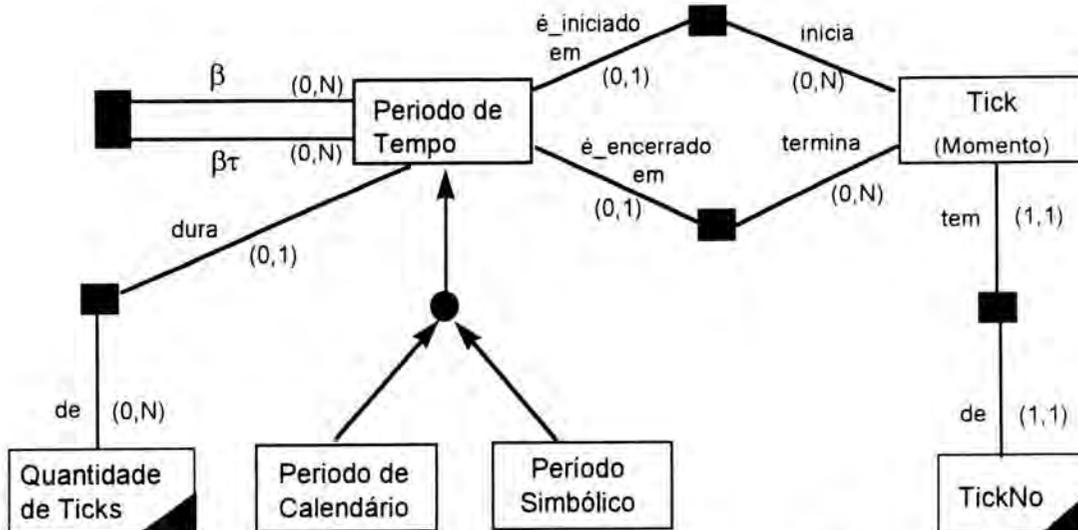


FIGURA 3.3 - Metamodelo do tempo na abordagem ERT

Relação entre intervalos	Símbolo ( $\beta$ )	Símbolo inverso ( $\beta\tau$ )	Ilustração
<i>X before Y</i>	<	>	XXX YYY
<i>X equal Y</i>	=	=	XXX YYY
<i>X meets Y</i>	m	mi	XXXYYY
<i>X overlaps Y</i>	o	oi	XXXX YYYY
<i>X during Y</i>	d	di	XXX YYYYYYY
<i>X starts Y</i>	s	si	XXX YYYYYY
<i>X finishes Y</i>	f	fi	XXX YYYYYY

FIGURA 3.4 - Relações entre intervalos (períodos) de tempo

É com base principalmente no modelo ERT que se concebeu o modelo TempER apresentado no próximo capítulo. As principais diferenças entre as abordagens situam-se na simbologia e na primitiva temporal adotada. No modelo TempER procurou-se manter a simbologia original proposta em [CHE 76] e, em relação à primitiva utilizada para especificar o tempo, adotou-se o elemento temporal em vez do intervalo de tempo.

### 3.2.2 O modelo TER

A principal característica do modelo TER (Temporal Entity-Relationship) [TAU 91] é o desdobramento da noção de cardinalidade em *snapshot* e *lifetime*. Diferentemente da abordagem descrita na seção anterior e do modelo proposto nesta dissertação, o modelo TER não diferencia, ao nível da notação gráfica, objetos temporais de objetos não temporais.

A cardinalidade *snapshot* define o número de instâncias de um conjunto de entidades alvo que se relacionam com uma determinada instância de um conjunto de entidades fonte em um ponto arbitrário do tempo. A cardinalidade *lifetime*, por outro lado, define o número de instâncias do conjunto de entidades alvo que se relacionam com uma instância do conjunto de entidades fonte, durante todo o tempo de existência da entidade fonte. Cardinalidade *snapshot* e *lifetime* são representadas por valores mínimo e máximo. Por exemplo, a cardinalidade *snapshot* máxima de *EMPREGADO* para *ESPOSA* (em uma sociedade monogâmica) é 1, a cardinalidade *lifetime* máxima é  $n$  (se as leis permitirem o divórcio), enquanto que ambas as cardinalidades mínimas são iguais a 0. Ou seja, um empregado pode ter no máximo uma esposa em um dado momento do tempo, entretanto, ao longo da sua existência, pode ter mais que uma.

A figura 3.5 mostra um exemplo de diagrama TER e a respectiva tradução para modelo ER convencional.

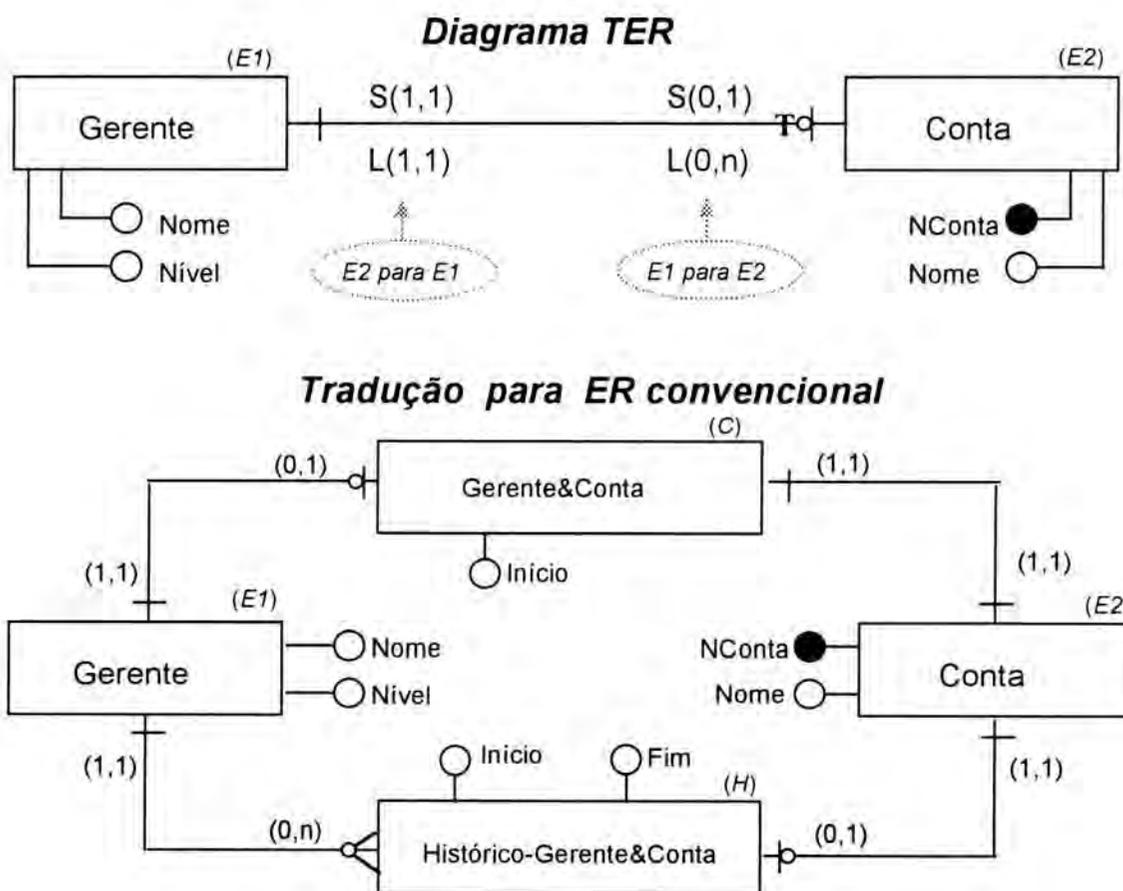


FIGURA 3.5 - Modelo de dados TER

O algoritmo para produzir a conversão do modelo TER para ER convencional é o seguinte:

1. Uma associação (C) representando a posição corrente das duas entidades relacionadas (E1, E2) é criada. O nome de C é construído pela justaposição dos seus constituintes (E1&E2).
2. O atributo *Inicio* (um rótulo temporal) é anexado a C.
3. Uma associação (H) representando a história das duas entidades relacionadas (E1, E2) é criada. O nome de H é igual ao nome de C mais o prefixo "Histórico-".
4. Os atributos *Inicio* e *Fim* (rótulos temporais) são anexados a H.
5.  $\min(E1 \text{ para } C)$  fica igual ao  $\min S(E1 \text{ para } E2)$ .
6.  $\max(E1 \text{ para } C)$  fica igual ao  $\max S(E1 \text{ para } E2)$ .
7.  $\min(E1 \text{ para } H)$  é ajustado para 0.
8.  $\max(E1 \text{ para } H)$  fica igual a  $\max L(E1 \text{ para } E2)$ .
9. Os passos 5 a 8 devem ser repetidos para a direção E2 para E1.
10. O relacionamento (direto) entre E1 e E2 é eliminado.

### 3.2.3 O modelo TEER

O modelo TEER [ELM 93] estende o modelo Entidade-Relacionamento incluindo informações temporais nas entidades, relacionamentos, super-classes/sub-classes e atributos. A dimensão temporal dos objetos é representada por um rótulo contendo um elemento temporal, cuja definição encontra-se na seção 3.1.3 deste capítulo. Este rótulo define a existência (*lifespan*) do objeto no mini-mundo modelado.

Em TEER, cada entidade tem um *surrogate* definido pelo sistema que a identifica em relação as demais. O valor deste atributo não é visível ao usuário e não se modifica ao longo da existência da entidade.

Neste modelo todos os objetos *variam no tempo*, não existindo diferenciação entre objetos temporais e não temporais. Assim, não há necessidade de uma notação gráfica especial para denotar a dimensão temporal dos objetos, que fica implícita, bastando a simbologia ER convencional. Embora esta abordagem torne mais simples a modelagem, impede que o analista consiga representar entidades e relacionamentos que não precisam estar associados ao tempo (aquelas que não se quer registrar a evolução histórica).

### 3.2.4 O modelo STEER

O modelo STEER (*Semantic Temporal model based on the Extended Entity-Relationship model*) [ELM 92] é uma variante do modelo TEER apresentado acima. A sua principal característica está em distinguir objetos conceituais de objetos temporais e relacionamentos conceituais de relacionamentos temporais.

Os *objetos conceituais* são as entidades propriamente ditas e a sua representação gráfica é um retângulo. O tempo de existência de uma entidade inicia quando esta é introduzida no mini-mundo modelado, e não mais se encerra, visto que um conceito (a entidade) uma vez realizado jamais cessa de existir.

Os *objetos temporais* são os papéis que os objetos conceituais desempenham na dimensão temporal. São representados por retângulos cuja área interna é preenchida com a cor cinza. O tempo de validade de um papel, ou *lifespan*, está contido no tempo de existência da entidade à qual o papel se refere. Um papel apresenta apenas atributos temporais.

Os relacionamentos conceituais são aqueles em que apenas objetos conceituais participam. Já nos relacionamentos temporais, ou todos os objetos participantes são conceituais (entidades), ou todos são temporais (papéis).

### 3.3 Demais modelos temporais

Além dos modelos temporais que adotam a abordagem Entidade-Relacionamento, existem aqueles que adotam o paradigma da orientação a objetos, entre eles: o modelo TF-ORM [EDE 94], que distingue entre classes de objetos e classes de papéis e registra implicitamente a evolução das propriedades cujo conteúdo varia no tempo, o modelo OODAPLEX [WUU 93] e o modelo TDM [SEG 88, 93], que trata as propriedades dos objetos através da estrutura denominada Sequência Temporal (Time Sequence). Embora estes modelos apresentem mecanismos bastantes expressivos para tratar a dimensão temporal, não são aproveitados neste trabalho. Isto porque se fez a opção pela abordagem Entidade-Relacionamento.

Em [OZS 95] é apresentado um tutorial que mostra uma visão geral dos principais modelos de dados temporais existentes na literatura técnica, com ênfase para os que adotam a abordagem relacional e a abordagem orientada a objetos.

Um outro esforço importante na área, a ser considerado, é a linguagem TSQL2 [SNO 94], a qual se pretende propor como padrão para os bancos de dados relacionais temporais.

## 4 O modelo de dados temporal TempER

O modelo de dados Entidade-Relacionamento (ER), introduzido por Peter Chen [CHE 76], tem sido ao longo das duas últimas décadas, a técnica mais empregada para modelar o aspecto estático de sistemas de informação. Desde o seu surgimento, muitas extensões têm sido propostas, visando ampliar a sua capacidade de representação.

Neste capítulo é apresentado e descrito o modelo de dados TempER, uma extensão do modelo Entidade-Relacionamento convencional, que permite a modelagem dos dados em relação ao tempo.

### 4.1 Justificativas

Nesta seção, primeiro é justificado porque é necessário estender o modelo ER convencional com dispositivos para tratar a dimensão temporal. Depois são listados os principais requisitos que um modelo ER temporal deve atender. Em seguida, é feita uma comparação de alguns modelos ER temporais disponíveis na literatura com a referida lista de requisitos. Por fim, é mostrado de que forma o modelo TempER, que é o modelo de dados temporal proposto nesta dissertação, preenche estes mesmos requisitos.

#### 4.1.1 Porque um modelo ER temporal é necessário

Quando se utiliza um modelo ER convencional [CHE 76], a associação das entidades e relacionamentos com o tempo materializa-se normalmente através da inclusão de atributos comuns que armazenam datas, horas ou qualquer outra referência temporal. Além disso, caso se necessite registrar os diversos valores que um atributo pode apresentar ao longo do tempo, por exemplo os salários de um empregado, existem duas alternativas possíveis: ou se especifica uma nova entidade composta deste atributo e mais os atributos de referência ao tempo, ou se define tal atributo como composto e multivalorado.

Quando atributos que armazenam pontos do tempo são incluídos em um modelo ER convencional, é muito comum a necessidade de ter que se especificar restrições em relação a eles, tais como as abaixo descritas:

- Ao se transferir em 23/10/96 um empregado para um departamento, primeiro o empregado deve apresentar a sua data de admissão menor ou igual a 23/10/96 e não pode estar demitido; isto é, a data de demissão deve conter *null*; segundo, o departamento deve ter a sua data de criação menor ou igual a 23/10/96 e não pode estar fechado, isto é, a sua data de fechamento deve conter *null*.
- Um valor de salário deve ter o seu período de validade contido entre a data de admissão e a data de demissão do empregado, donde se deduz que não deve ser permitido aumentar o salário de um empregado após a sua data de demissão.
- Um empregado não pode estar lotado em mais que um departamento no mesmo ponto do tempo.

- Em todos os momentos do tempo que o empregado estiver vinculado à empresa, ele necessariamente deve estar lotado em um departamento, qualquer que seja ele.

O modelo ER convencional não dispõe de instrumentos para representar restrições temporais, semelhantes às acima descritas, exigindo que isto se faça ao nível da modelagem dinâmica do sistema. Visando suprir esta carência, surgiram então os modelos Entidade-Relacionamento temporais.

#### 4.1.2 Requisitos necessários a um modelo ER temporal

Quando se propõe um modelo de dados temporal, um desafio importante é permitir que o modelador possa em um mesmo diagrama representar elementos (entidades, relacionamentos ou atributos) temporalizados e elementos não temporalizados. Isto se faz necessário porque normalmente em sistemas de informação, alguns dados precisam ser explicitamente referenciados em relação ao tempo (a evolução dos salários dos empregados, a alocação de técnicos a projetos, etc), e outros não apresentam esta necessidade, pois é irrelevante ao usuário do sistema saber quando os fatos ocorreram (a autoria de um artigo). Portanto, as principais características que uma técnica de modelagem ER temporal deve apresentar são as seguintes:

1. A dimensão temporal deve estar “embutida” no modelo, ou seja, é assumido que o tempo é um conjunto de pontos implícito, aos quais os objetos se associam. Desta forma, enquanto que no modelo ER convencional os conjuntos de entidades apresentam apenas duas dimensões: a das tuplas e a dos atributos, no modelo ER temporal passam a apresentar três: a das tuplas, a dos atributos e a do tempo.
2. Deve oferecer uma notação especial para diferenciar entidades temporalizadas (que estão associadas ao tempo) de entidades não temporalizadas (que não estão associadas com o tempo).
3. Deve permitir que uma entidade temporalizada se associe com uma entidade não temporalizada.
4. Deve permitir que um relacionamento entre entidades possa ser definido como temporalizado ou como não temporalizado, não importando qual seja a classificação temporal destas entidades.
5. Deve permitir que em uma mesma entidade possam conviver atributos temporalizados e atributos não temporalizados.
6. A restrição de cardinalidade que define o grau de participação de uma entidade em um conjunto de relacionamentos temporalizados deve considerar os pontos do tempo. Por outro lado, em se tratando de conjunto de relacionamentos não temporalizados, a cardinalidade não deve levar em conta os pontos do tempo, mantendo a mesma semântica do modelo ER convencional. (Para maiores detalhes sobre o significado das restrições de cardinalidade em um modelo ER temporal, consultar a seção 4.4.2.3).

### 4.1.3 Cruzamento de modelos ER temporais com os requisitos

Dentre as técnicas de modelagem ER temporal disponíveis na literatura, o presente trabalho concentrou-se em estudar as seguintes:

- modelo ERT (Entity Relationship Time Model) [LOU 91],
- modelo TEER (Temporal Enhanced Entity-Relationship Model) descrito em [ELM 93] juntamente com a sua variante STEER (Semantic Temporal model based on the Extended Entity-Relationship model) descrita em [ELM 92].
- modelo TER (Temporal Entity-Relationship Model) [TAU 91]

A tabela 4.1 mostra de que forma estas técnicas preenchem os requisitos citados anteriormente relativos a um modelo ER temporal.

O modelo TER [TAU 91] não introduz novas formas gráficas de representar entidades, relacionamentos e atributos temporalizados, concentrando-se apenas em desdobrar a representação das cardinalidades dos relacionamentos em cardinalidade *snapshot*, que determina o grau de participação da entidade no relacionamento a cada instante de tempo, e cardinalidade *lifetime*, que determina o grau de participação da entidade no relacionamento, tendo como unidade de referência a existência da entidade como um todo.

O modelo TEER [ELM 93] assume que todos os objetos são temporalizados, isto é, são válidos por um certo período de tempo, o que simplifica bastante o modelo, mas exige que o modelador tenha que referenciar temporalmente até os dados que não precisam ser temporalizados. A variante STEER [ELM 92] amplia a capacidade do modelo TEER ao distinguir objetos conceituais de objetos temporais, e relacionamentos conceituais de relacionamentos temporais. Entretanto, um objeto conceitual, cuja existência uma vez realizada jamais cessa, isto é, possui um momento inicial mas não um final, também apresenta uma referência ao tempo, não podendo ser considerado como não temporalizado.

TABELA 4.1 - Características consideradas importantes a um modelo E-R Temporal

Características buscadas em um modelo ER temporal	TER [TAU 91]	TEER [ELM 93]	STEER [ELM 91]	ERT [LOU 91a]
1. Dimensão temporal embutida no modelo (conjunto implícito de pontos do tempo).	Não contempla	SIM	SIM	SIM
2. Notação diferenciando objetos temporalizados de objetos não temporalizados.	Não	Não	Em termos	SIM
3. Permite relacionar entidade temporalizada com entidade não temporalizada.	Não se aplica	Não se aplica	Em termos	SIM
4. Permite modelar relacionamentos temporalizados e não temporalizados entre entidades.	Em termos	Não se aplica	Em termos	SIM
5. Permite mesclar em uma mesma entidade atributos temporalizados e não temporalizados.	Não	Não	Em termos	SIM
6. Cardinalidade da participação em relacionamento temporalizado referenciando um ponto do tempo.	SIM	SIM	SIM	SIM

O modelo ERT [LOU 91] é o único dos modelos pesquisados que permite modelar elementos temporalizados e não temporalizados, assumindo que os objetos não temporalizados são de existência permanente.

Maiores detalhes sobre os modelos acima descritos, principalmente sobre o modelo ERT, encontram-se no capítulo anterior.

Embora o modelo ERT [LOU 91] atenda a todos os requisitos considerados necessários a um modelo ER temporal, apresenta os aspectos abaixo considerados inadequados, principalmente quando se deseja integrar ao modelo de dados um modelo de transações, como é o caso da abordagem proposta nesta dissertação:

- O modelo ERT utiliza como rótulo temporal o intervalo de tempo, o que implica na necessidade de fragmentar em diversas tuplas os objetos “reencarnados”, uma tupla para cada intervalo de existência. Esta fragmentação não seria necessária caso o *elemento temporal* tivesse sido escolhido com rótulo temporal. O emprego de intervalos de tempo resulta em linguagens de difícil utilização, conforme uma comparação encontrada em [GAD 93] entre a linguagem Tquel, que utiliza intervalos, e a linguagem TempSQL, que utiliza elementos temporais.
- No modelo ERT os atributos são representados graficamente, através de retângulos e são ligados às entidades através de relacionamentos. Em aplicações de grande porte isto pode resultar em diagramas bastante poluídos.

#### 4.1.4 Uma proposta para atender os requisitos

Visando tornar disponível um modelo ER temporal que preenchesse todos os requisitos listados na tabela 4.1, bem como evitasse os aspectos considerados desfavoráveis no modelo ERT, acima descritos, concebeu-se o modelo TempER. O modelo TempER pode ser entendido como uma adequação do modelo ERT, no sentido de proporcionar uma integração mais facilitada com a modelagem de transações.

No estágio atual do modelo TempER não se adotou os conceitos de classes derivadas e de objetos complexos, disponíveis no modelo ERT, ficando esses para uma futura avaliação.

Em uma visão geral, as principais características do modelo de dados TempER são as seguintes:

- Oferece uma simbologia que diferencia elementos temporalizados de elementos não temporalizados.
- Permite que se associe em um mesmo diagrama entidades temporalizadas com não temporalizadas. Para tanto, é assumido que as entidades não temporalizadas também apresentam uma dimensão temporal implícita, igual a todo o conjunto de pontos do eixo temporal. Desta forma o adjetivo “não temporalizado” deixa de ter aplicação, passando-se a utilizar o adjetivo “perene” para tal tipo de entidade. Quanto às entidades temporalizadas, por se tratar de objetos cujo período de validade é um subconjunto de pontos do eixo temporal, adota-se o adjetivo “transitórias” para qualificá-las.
- Qualquer que seja a classificação das entidades em relação ao tempo, sejam elas perenes ou transitórias, ortogonalmente sempre apresentam duas perspectivas: uma intemporal e uma temporal (ver figuras 4.4, 4.7 e 4.9). Quando se focaliza os conjuntos de entidades pela perspectiva intemporal estes apresentam apenas

duas dimensões (tuplas x atributos intemporais). Por outro lado, quando se focaliza estes mesmos conjuntos pela perspectiva temporal eles apresentam três dimensões (tupla x atributos temporais x eixo temporal).

- No tocante aos relacionamentos, ou as entidades se associam entre si na perspectiva temporal (relacionamentos temporais) ou na perspectiva intemporal (relacionamentos intemporais), conforme mostra a figura 4.10.
- Possibilita que as restrições de cardinalidade levem em consideração os momentos do tempo de validade de um relacionamento temporal.
- Adota como primitiva temporal o elemento temporal e não o intervalo de tempo. O elemento temporal (uma união finita de intervalos de tempo) por ser fechado para as operações de união, interseção e complementação, permite uma “substancial simplificação na habilidade do usuário de expressar consultas temporais” [GAD 88, 93]. (Mais explicações sobre o elemento temporal encontram-se no capítulo anterior.)
- Faz uso de um dicionário de dados para descrever os atributos, evitando que estes sejam explicitados graficamente. Isto contribui para tornar os diagramas mais administráveis visualmente.

As características acima listadas, bem como outros aspectos do modelo de dados TempER, serão aprofundados nas próximas seções deste capítulo.

Para dar consistência à semântica do modelo TempER, os seus vários aspectos encontram-se precisamente especificadas no capítulo 6, através de mapeamento para rede de Petri.

## 4.2 Visão geral do modelo de dados TempER

Antes de iniciar a descrição das características do modelo TempER é importante ter em mente o que significa *existência* - ou validade temporal - no âmbito do presente trabalho. Existência de uma entidade nada mais é que o conjunto de pontos do tempo em que esta entidade é considerada como válida para efeito de inferências temporais no banco de dados. Não se deve confundir *existência* com a presença física da entidade no banco de dados, ou seja, uma entidade pode estar presente no banco de dados e em relação a um determinado ponto do tempo *não existir*.

O modelo TempER é um modelo de dados do tipo Entidade-Relacionamento que incorpora dispositivos que permitem referenciar os objetos (entidades, relacionamentos ou valores de atributos) à dimensão temporal.

No modelo TempER, a dimensão temporal é assumida como sendo um eixo de pontos discretos, isomórfico ao conjunto dos números inteiros, como já foi definido no capítulo anterior.

Assim como o modelo ER convencional, o modelo TempER apresenta também os seguintes elementos básicos: entidade, relacionamento e atributos.

Um grupo de entidades de mesma natureza e com a mesma estrutura de atributos, é denominado de *conjunto-entidade* e é este conjunto que se simboliza graficamente no diagrama. Um exemplo é o conjunto-entidade *Empregado* da figura 4.1, que representa

todos os empregados de uma certa empresa. Da mesma forma, um grupo de relacionamentos com as mesmas características (mesmos conjuntos-entidade associados e mesma finalidade) é denominado de *conjunto-relacionamento*. Por exemplo, todas as lotações existentes de empregados em departamentos são simbolizadas pelo conjunto-relacionamento *Lotação* da figura 4.1. Em resumo, no contexto desta dissertação, entidade é uma instância de um conjunto-entidade e relacionamento é uma instância de um conjunto-relacionamento.

Os atributos são propriedades das entidades e relacionamentos. A associação de um atributo com um valor, do domínio de valores deste atributo, é chamada, no contexto deste trabalho, de valoração de um atributo ou atribuição de valor a uma propriedade. Os domínios são conjuntos predeterminados e constantes de valores primitivos.

Quando um atributo de uma entidade é especificado como sendo temporal, está se referindo apenas à valoração deste atributo, ou seja ao seu conteúdo. A temporalização da propriedade em si, isto é a especificação de quando esta propriedade passa a fazer parte ou deixar de fazer parte da estrutura do objeto foge do escopo do presente trabalho, é um assunto a ser tratado no âmbito da evolução de esquemas [EDE 95].

Aos objetos de uma aplicação que forem especificados como temporalizados, sejam eles entidades, relacionamentos ou valoração de atributos, são implicitamente anexados rótulos temporais, que conterão o conjunto de pontos do tempo nos quais estes objetos são considerados como existentes no contexto da realidade modelada. O rótulo temporal utilizado no modelo TempER tem o formato do elemento temporal conforme explicado anteriormente.

Para ilustrar os conceitos acima descritos, tome-se como exemplo um modelo de dados contendo dois conjuntos-entidade, *Empregado* e *Depto*, e mais um conjunto-relacionamento, denominado *Lotação*, associando estes dois conjuntos-entidade. Suponha-se que esta associação deva obedecer à seguinte restrição: um empregado obrigatoriamente deve estar lotado a um departamento em cada momento da sua existência como empregado, não podendo estar lotado em mais de um departamento ao mesmo tempo. Um outro requisito seria a necessidade de representar as possíveis lotações que um empregado pode apresentar ao longo do tempo em função das suas transferências de um departamento para outro. Trata-se do mesmo exemplo já explorado na seção 2.7, do capítulo 2.

A figura 4.1 apresenta o caso acima descrito, modelado em ER convencional e em TempER. No ER convencional não é possível especificar a restrição que determina que um empregado não pode estar lotado em mais de um departamento em cada momento do tempo. A cardinalidade "(1, N)", que aparece na ligação entre *Empregado* e *Lotação*, especifica que um empregado deve estar associado a no mínimo um departamento, podendo estar associado a mais de um (adota-se a forma de indicar a cardinalidade encontrada em [BAT 92], que determina o grau de participação mínima e máxima de uma entidade em um conjunto de relacionamentos).

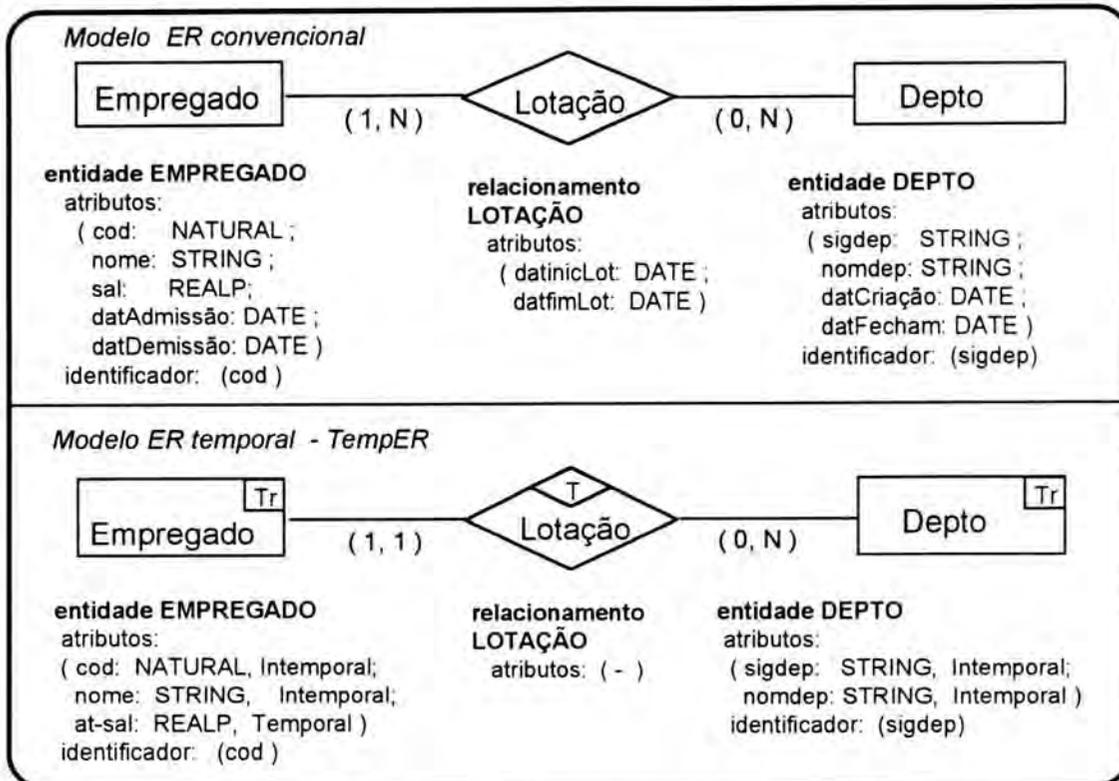


FIGURA 4.1 - Comparação entre ER convencional e TempER

Entidade <b>Empregado</b>					Relacionamento <b>Lotação</b>		
Existência	O I D	cod	nome	at-sal	Validade Temporal	O I D EMPREGADO	O I D DEPTO
[ 3,10 ] U [ 20, » ]	1001	e1	Gadia	180 [ 3, 6 ] 220 [ 7, 10 ] U [ 20, 25 ] 250 [ 26, » ]	[ 3, 10 ] U [ 20, 30 ]	1001	9011
[ 7, 35 ]	1002	e2	Segev	110 [ 7, 20 ] 180 [ 21, 35 ]	[ 31, » ]	1001	9013
[ 2, 20 ] U [ 30, » ]	1003	e3	Clifford	200 [ 2, 20 ] U [ 30, 35 ] 250 [ 36, » ]	[ 7, 20 ]	1002	9011
[ 25, » ]	1004	e6	Snodgrass	100 [ 25, 30 ] 130 [ 31, » ]	[ 21, 35 ]	1002	9014
[ 5, 25 ]	1005	e8	Jajodia	100 [ 5, 25 ]	[ 2, 10 ] U [ 15, 18 ] U [ 30, 35 ]	1003	9011
[ 10, » ]	1006	e4	Tansel	170 [ 10, 20 ] 190 [ 21, » ]	[ 11, 14 ] U [ 19, 20 ]	1003	9012
					[ 36, » ]	1003	9014
					[ 25, » ]	1004	9014
					[ 5, 15 ]	1005	9012
					[ 16, 25 ]	1005	9013
					[ 10, 20 ]	1006	9012
					[ 21, » ]	1006	9014

Entidade <b>Depto</b>		Existência	O I D	sigdep	nomdep
		[ 1, » ]	9011	defin	financeiro
		[ 3, 20 ]	9012	desis	sistemas
		[ 10, » ]	9013	depro	produção
		[ 21, » ]	9014	deinf	informática
		[ 7, 30 ]	9015	demat	materiais

FIGURA 4.2 - Exemplo de povoamento de entidades e relacionamento em TempER

Em virtude de o modelo ER convencional não dispor de primitivas de modelagem para representar a associação dos objetos com a dimensão temporal, o modelador refere-se ao tempo através de atributos comuns, tais como *datAdmissão*, *datDemissão*, *datinicLot*, *datfimLot*, *datCriação*, *datFecham* encontrados na figura 4.1. Isto implica em transferir para a modelagem dinâmica a responsabilidade de tratar as restrições temporais, ou seja, no caso do exemplo em questão, são as transações do sistema que devem se preocupar em impedir que um empregado possa estar lotado em dois ou mais departamentos ao mesmo tempo.

Este problema deixa de existir quando se utiliza o modelo TempER. Como pode ser visto no diagrama situado na parte inferior da figura 4.1, a cardinalidade que aparece na ligação entre *Empregado* e *Lotação* agora é "(1, 1)", a qual tem o seguinte significado: um empregado participa do conjunto-relacionamento *Lotação* no mínimo uma vez e no máximo uma vez *a cada momento do tempo*. Além disto, os atributos referindo-se a pontos do tempo que estavam no diagrama ER convencional deixam de ser necessários, pois são substituídos por rótulos temporais implícitos.

Para um melhor entendimento do diagrama TempER da figura 4.1, pode ser consultado o exemplo de povoamento do modelo mostrado na figura 4.2.

Uma outra vantagem em relação ao modelo ER convencional diz respeito ao conceito de reencarnação de objetos. Um empregado que foi demitido pode ser readmitido, um empregado que tenha sido alocado a um determinado departamento pode vir a ser realocado a este mesmo departamento em um outro período de tempo. O modelo TempER representa estas situações de forma natural, pois admite que a existência de um objeto possa ser segmentada em intervalos de tempo, como é o caso do empregado *1001* e do relacionamento  $\langle 1003, 9011 \rangle$  encontrados na figura 4.2.

O modelo TempER também permite que se mescle objetos de diferentes classificações temporais em um mesmo diagrama, como mostra a figura 4.3, onde um conjunto-entidade do tipo perene (*Função*) relaciona-se com um conjunto-entidade do tipo transitório (*Empregado*) de duas formas: uma temporalmente (*Alocação*) e outra intemporalmente (*Preferência*).

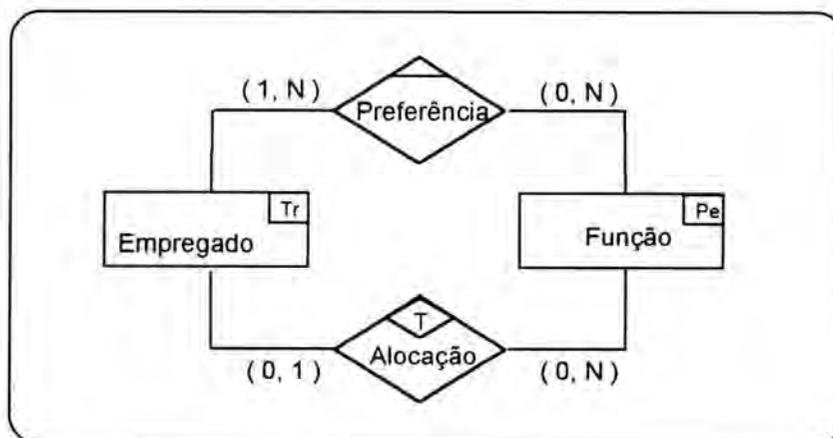


FIGURA 4.3 - Mesclando objetos de diferentes classificações temporais em um mesmo diagrama

### 4.2.1 O identificador interno de entidades - OID

É assumido que todas as instâncias dos conjuntos-entidade, e apenas dos conjuntos-entidade, possuem um identificador interno, gerado pelo sistema, que será denominado de OID (object identifier). Cada OID, por princípio, é único no âmbito do universo do discurso da aplicação, é invisível ao usuário e define a identidade de uma entidade. Quando duas entidades se associam, este fato pode ser representado pelo relacionamento dos OID's respectivos. Isto faz com que um relacionamento seja identificável pela composição dos OID's das entidades associadas.

A presença do OID não descarta a necessidade de que haja um atributo (ou composição de atributos) que desempenhe o papel de chave primária de uma entidade, de forma que um usuário do sistema possa identificar e acessar esta entidade, já que o OID é invisível externamente. Embora possa parecer redundante coexistirem OID's e chaves primárias, o que se busca é aproximar o modelo TempER de um dos mais importantes princípios da orientação a objetos, o da identidade dos objetos [KHO 86]. Além disso, existe a vantagem de ser possível alterar a chave primária de uma entidade sem que isto afete os relacionamentos em que ela participe.

A título de exemplo pode-se examinar a figura 4.2, onde tanto a tabela *Empregado* como a tabela *Depto*, possuem uma coluna referente ao OID, e a tabela referente ao conjunto-relacionamento *Lotação* possui duas colunas de OID's referentes aos conjuntos-entidade relacionados.

## 4.3 As perspectivas temporal e intemporal das entidades

As entidades são os elementos básicos de um modelo de dados que empregue a abordagem Entidade-Relacionamento. No modelo TempER, em virtude da dimensão temporal, as entidades apresentam sempre duas perspectivas, uma perspectiva temporal e uma perspectiva intemporal, como se fossem as duas faces de uma mesma moeda.

### 4.3.1 Perspectiva intemporal

Não leva em consideração a dimensão temporal, isto é, o OID da entidade não é visualizado como associado a um conjunto de pontos do tempo. Nesta perspectiva as entidades apresentam apenas duas dimensões: a dos atributos e a dos valores. É o que mostra a parte superior da figura 4.4.

### 4.3.2 Perspectiva temporal

Leva em conta a dimensão temporal ao referenciar um dado. Nesta perspectiva, o OID de uma entidade é visualizado com estando associado a um conjunto de pontos do tempo, conjunto este que define a validade temporal (ou existência) da entidade no contexto do banco de dados do sistema. As entidades, vistas pela perspectiva temporal, apresentam três dimensões: a dos atributos, a dos valores e a do tempo, como mostra a parte inferior da figura 4.4.

Na figura 4.4, o mesmo conjunto de entidades (*Empregado*) apresenta simultaneamente as duas perspectivas. Na perspectiva intemporal, as entidades do conjunto apresentam os atributos *cod* e *nome* que, por consequência, são do tipo

intemporal. Por outro lado, na perspectiva temporal, as entidades apresentam o atributo *at-sal* (salário), que é um atributo do tipo temporal.

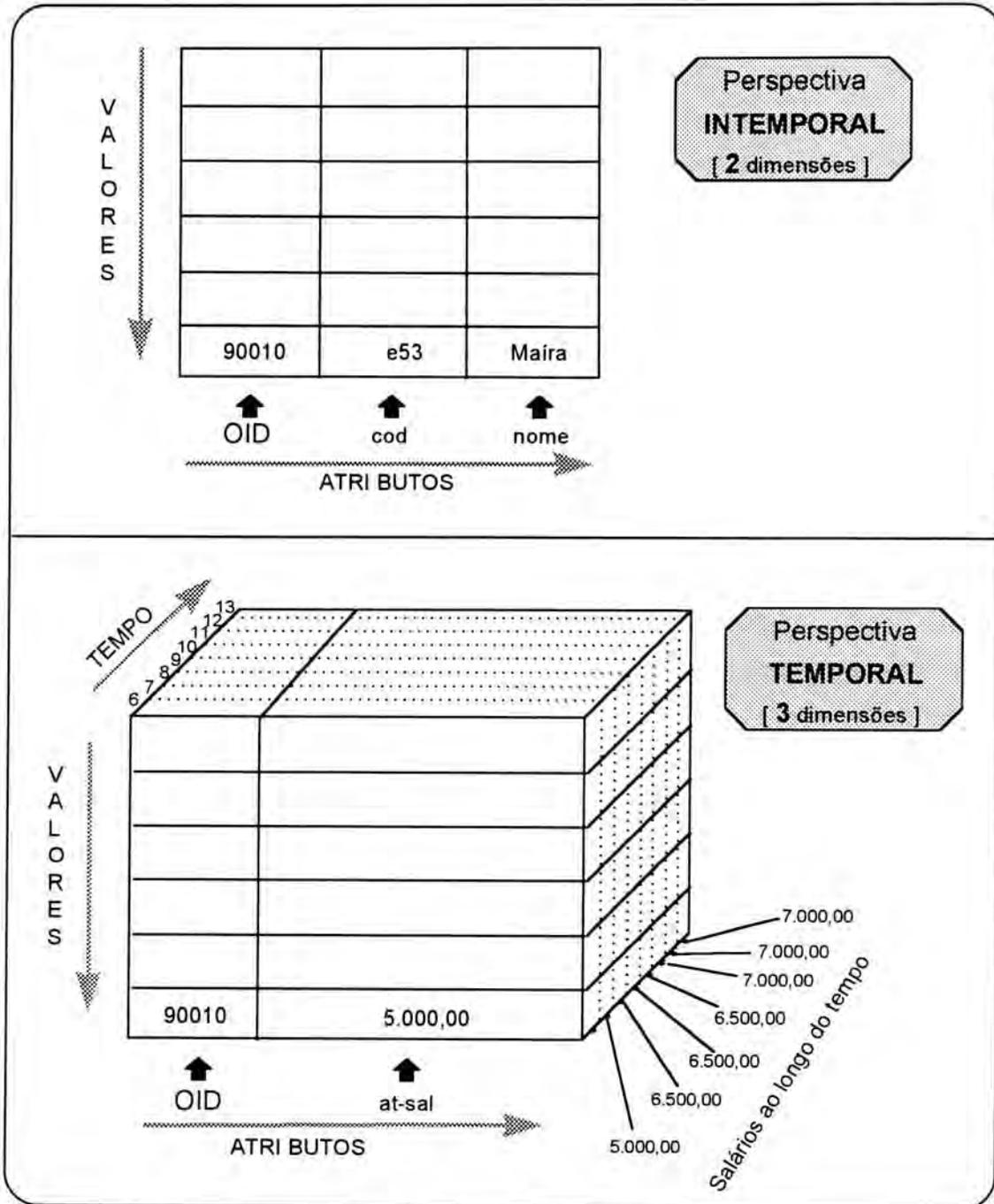


FIGURA 4.4 - As duas perspectivas em relação ao tempo do conjunto-entidade *Empregado*

## 4.4 Componentes do modelo de dados TempER

Na proposta de notação dos elementos gráficos do modelo TempER buscou-se manter os símbolos originais propostos em [CHE 76], adicionando-lhes apenas os sinais necessários para determinar qual é o tipo de relação destes elementos com a dimensão temporal. Portanto, a representação de um conjunto-entidade continua sendo um retângulo e a representação de um conjunto-relacionamento continua sendo um losango. Existem também símbolos destinados a modelar a estrutura hierárquica do tipo generalização/especialização que serão tratados mais adiante neste capítulo. A figura 4.5 mostra a simbologia gráfica básica utilizada no modelo TempER.

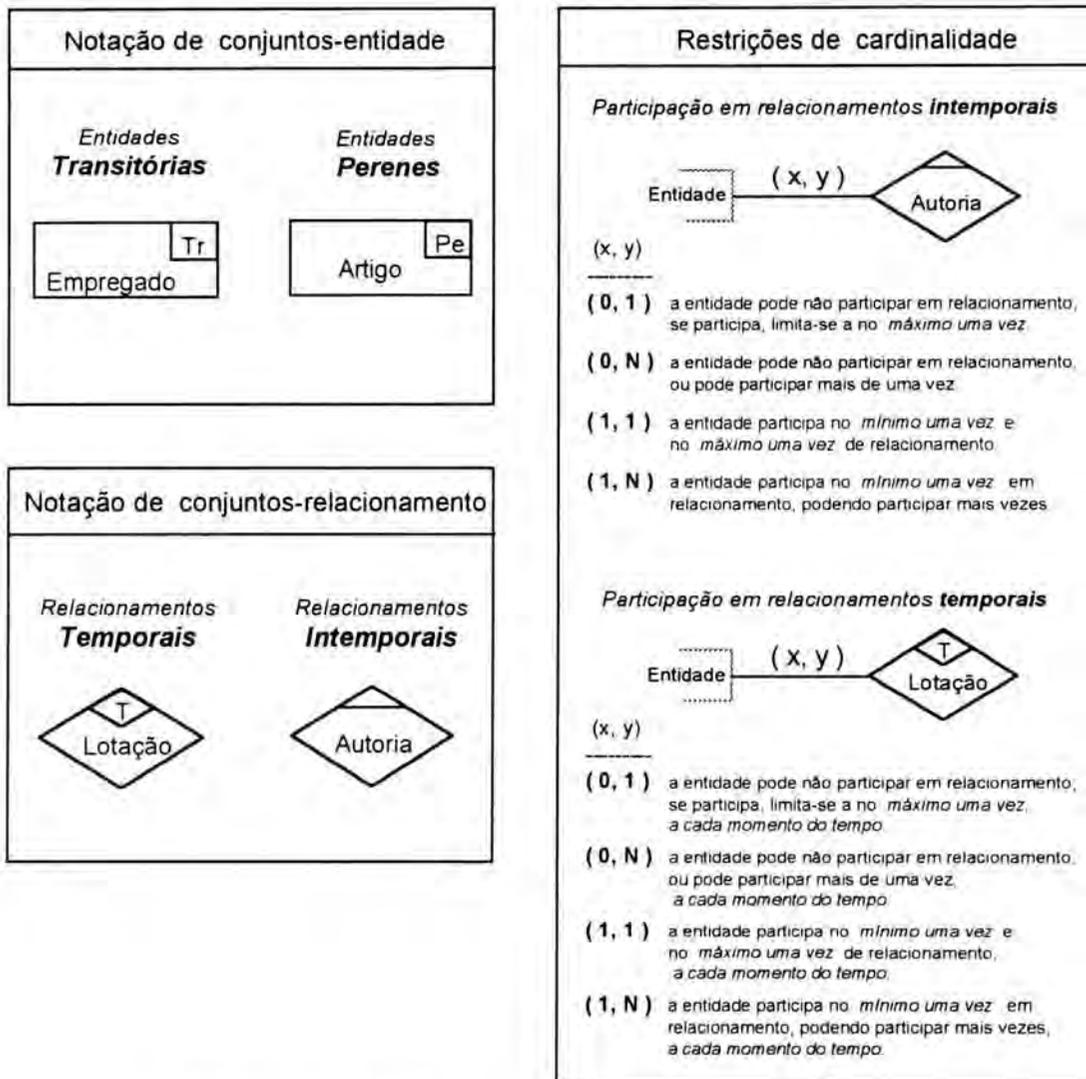


FIGURA 4.5 - Notação dos componentes básicos do modelo TempER

O modelo ER temporal de uma aplicação para estar completo precisa apresentar, além do diagrama entidade-relacionamento, a descrição dos atributos dos objetos modelados. Isto é feito através do dicionário de dados, cuja sintaxe é apresentada na seção 4.5 deste capítulo.

#### 4.4.1 Entidades

As entidades no modelo TempER são classificadas em transitórias e perenes, conforme a sua relação com a dimensão temporal.

##### 4.4.1.1 Entidades Transitórias

Entidades transitórias são aquelas cuja validade temporal é um subconjunto de pontos do tempo do eixo temporal. Normalmente lança-se mão deste tipo de entidade quando se quer modelar entidades que valem por um certo período de tempo. Por exemplo: os empregados de uma companhia que *existem* como tal desde a sua admissão até a sua demissão. Como é possível haver readmissão de empregados, a *existência* de uma destas entidades pode ser composta por mais de um intervalo de tempo, portanto o subconjunto de pontos do tempo que define a validade temporal de uma entidade não necessariamente é formado por momentos consecutivos. Isto é ilustrado pela figura 4.6, onde um certo empregado *empl* apresenta três vínculos com a empresa ao longo do tempo.



FIGURA 4.6 - Visualização da existência de uma entidade transitória

Uma outra forma de visualizar o que seja uma entidade transitória é através do seu mapeamento para um diagrama ER convencional, conforme mostra a figura 4.7.

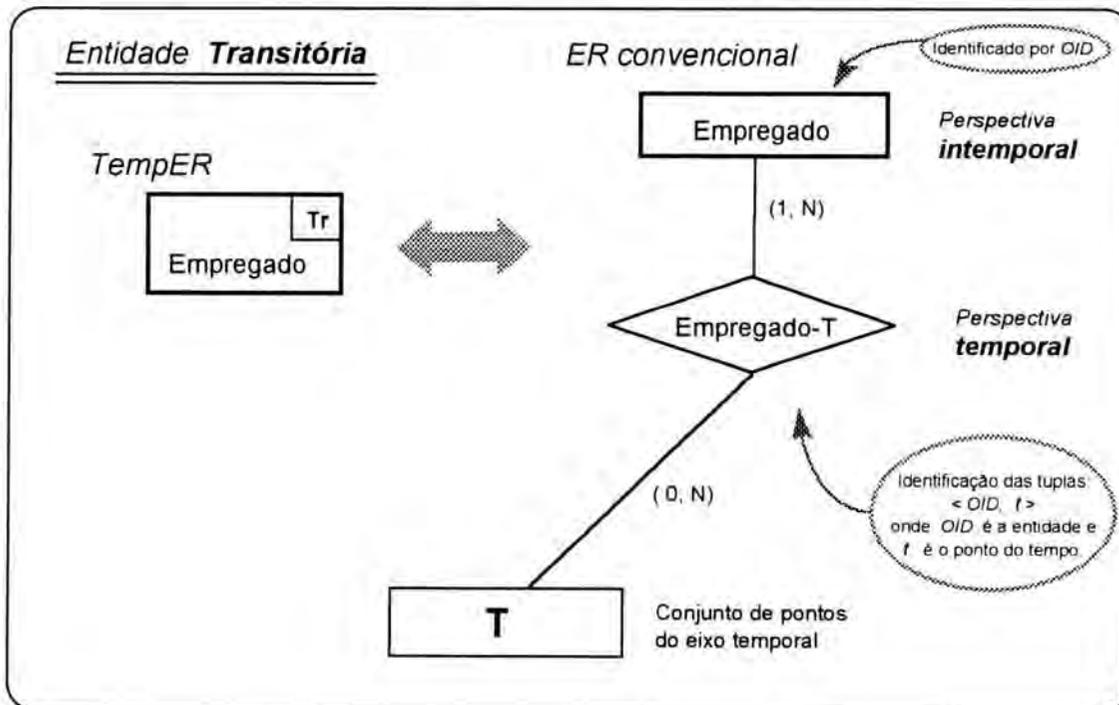


FIGURA 4.7 - Mapeamento de entidade transitória para ER convencional

O diagrama ER convencional da figura 4.7, cuja finalidade é mostrar de que forma uma entidade transitória relaciona-se com a dimensão temporal, apresenta um conjunto-entidade especial que contém todos os pontos do eixo temporal: o conjunto **T**. Cada empregado está associado a no mínimo um destes pontos do tempo, podendo estar associado a mais de um. O conjunto-relacionamento *Empregado-T* contém uma instância para cada ponto do tempo da validade temporal de cada empregado que esteja presente no conjunto-entidade *Empregado*.

Como demonstra o próximo capítulo, que trata dos aspectos dinâmicos de um sistema, é possível, através das transações, ampliar ou reduzir a validade temporal das entidades transitórias. Em outras palavras, como resultado do funcionamento do sistema, subconjuntos de pontos do tempo podem ser adicionados ou retirados da *existência* deste tipo de entidade. É dessa possibilidade de modificar a validade temporal que advém o qualificativo “transitórias” aplicado a estas entidades.

O símbolo para denotar entidades transitórias é um retângulo contendo a partícula “Tr” no canto superior direito.

#### 4.4.1.2 Entidades Perenes

São aquelas cuja validade temporal é exatamente igual a todo o eixo temporal. Toda a vez que uma entidade perene é incluída no banco de dados do sistema, assume-se que seu rótulo temporal é igual a “[«, »]”, isto é, a sua validade temporal inicia no primeiro ponto do eixo temporal e se estende até o último, conforme ilustra a figura 4.8. Normalmente as entidades que o modelador não necessita ou não deseja associar ao tempo são consideradas como perenes.



FIGURA 4.8 - Visualização da *existência* uma entidade perene

O fato de ser perene não significa que uma entidade não possa ser eliminada do banco de dados. Entretanto, enquanto uma entidade perene estiver presente no banco de dados, a sua validade temporal será constante, igual ao conjunto de todos os pontos do eixo temporal. Portanto, diferentemente das entidades transitórias, a validade temporal das perenes de forma alguma pode sofrer acréscimo ou redução.

Por ser constante, a validade temporal das entidades perenes não precisa ser registrada no banco de dados. O modelador deve visualizar o conjunto de pontos de tempo que define a existência das entidades perenes como sendo *implicitamente* especificado, ou seja sempre igual ao conjunto total de pontos do eixo temporal.

Também é possível representar uma entidade perene por intermédio de um mapeamento para diagrama ER convencional. É o que mostra a figura 4.9, onde para cada instância do conjunto entidade *Função*, o conjunto-relacionamento *Função-T* apresenta tantas tuplas  $\langle \text{OID}, t \rangle$  quantos forem os pontos do tempo do eixo temporal.

O símbolo para denotar entidades perenes é um retângulo contendo a partícula “Pe” no canto superior direito.

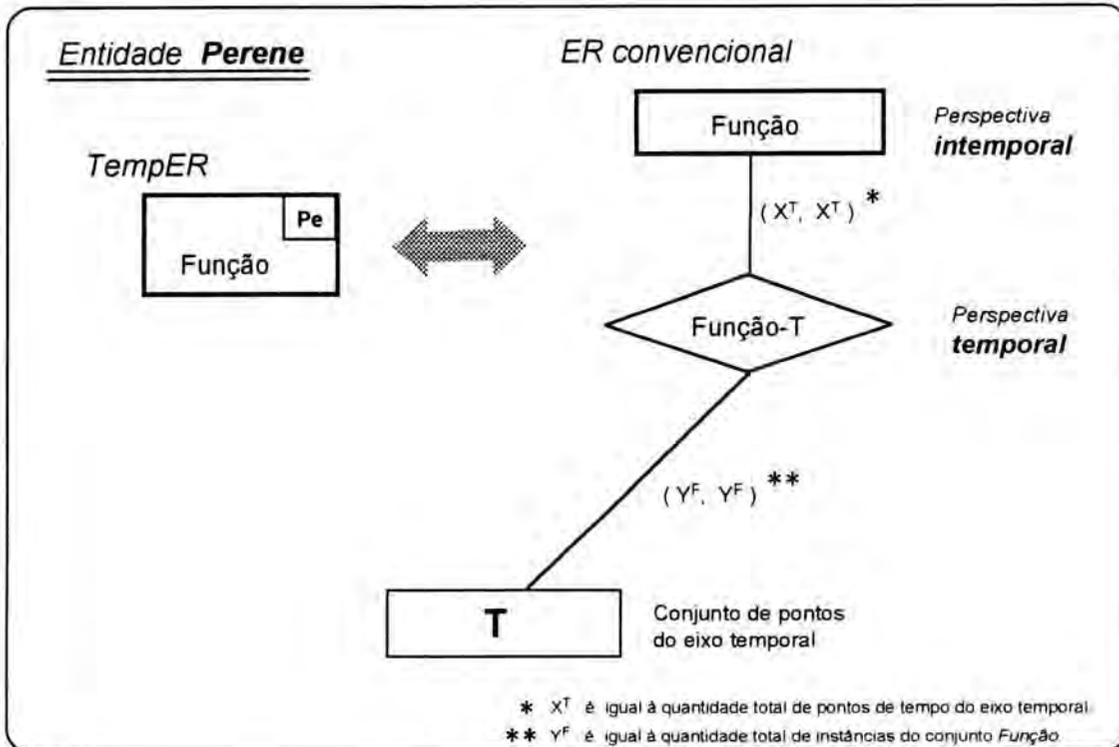


FIGURA 4.9 - Mapeamento de entidade perene para ER convencional

## 4.4.2 Relacionamentos

Relacionamentos são objetos resultantes da associação de duas ou mais entidades. Como já foi explicado anteriormente, o que se representa em um diagrama Entidade-Relacionamento são conjuntos de relacionamentos, agrupados conforme a sua natureza e estrutura de atributos. No modelo TempER as entidades podem estar associadas entre si, ou na perspectiva temporal, ou na perspectiva intemporal. Quando estão relacionadas na perspectiva temporal, este fato é simbolizado por um losango contendo no seu canto superior um pequeno losango com a letra "T". Quando estão relacionadas na perspectiva intemporal, a simbolização se dá através de um losango contendo um pequeno risco horizontal no canto superior. Estas duas notações são mostradas na figura 4.5.

### 4.4.2.1 Relacionamentos Temporais

São os relacionamentos que associam duas entidades no âmbito da dimensão temporal, isto é, além dos OID's das entidades, estes relacionamentos também se associam a pontos do tempo do eixo temporal. Este tipo de relacionamento serve para modelar as associações das quais se necessita conhecer a validade temporal. É o caso de *Alocação* da figura 4.3, de cujas instâncias se quer saber quando iniciaram e quando encerraram.

Uma forma de visualizar os relacionamentos temporais é através do seu mapeamento para ER convencional, como mostra a figura 4.10. O conjunto-relacionamento *Alocação*, que é do tipo temporal, é representado no diagrama ER convencional pelo conjunto-relacionamento que associa *Empregado-T* e *Função-T*, que são dois conjuntos-entidade derivados a partir de relacionamentos de empregados e funções com pontos do eixo temporal.

Uma instância de *Empregado-T* se associa apenas com instâncias de *Função-T* de mesmo ponto do tempo, e vice-versa, isto é, sendo  $\langle OID_E, OID_F, t \rangle$  uma tupla do conjunto *Alocação*, esta é resultante da associação de uma instância  $\langle OID_E, t_E \rangle$  de *Empregado-T* com uma instância  $\langle OID_F, t_F \rangle$  de *Função-T*, onde  $t_E = t_F = t$ .

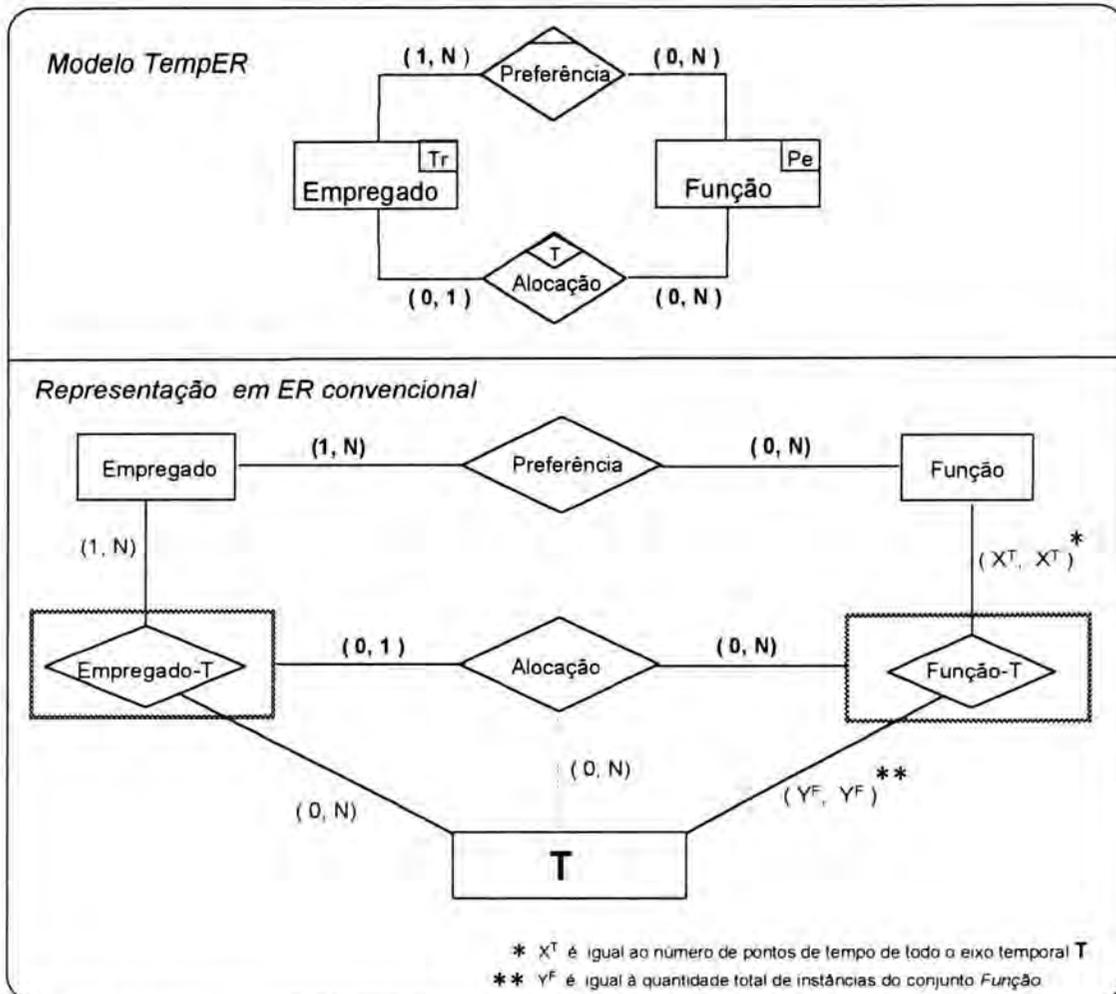


FIGURA 4.10 - Mapeamento de relacionamentos temporais e intemporais para ER convencional

Os relacionamentos temporais só são válidos nos momentos especificados pelo seus rótulos temporais. Por exemplo, a lotação do empregado de *OID 1003* no departamento de *OID 9012* (ver tabela referente ao conjunto-relacionamento *Lotação* da figura 4.2) só é válido nos momentos 11, 12, 13, 14, 19 e 20 do eixo temporal.

A validade no tempo de um relacionamento temporal sempre está contido dentro da interseção das existências das entidades associadas. Ou seja, o conjunto de pontos do tempo que define a validade de um relacionamento do tipo temporal é um subconjunto da interseção dos conjuntos de pontos que definem as existências das entidades associadas (por este relacionamento).

#### 4.4.2.2 Relacionamentos Intemporais

São relacionamentos que não levam em consideração a dimensão temporal, ou seja, neste tipo de relacionamento apenas participam os *OID's* das entidades, sem

qualquer menção a pontos do tempo. Os relacionamentos intemporais se materializam ao nível das perspectivas intemporais das entidades. É o que mostra o mapeamento para ER convencional do conjunto-relacionamento *Preferência*, presente na figura 4.10. No exemplo, a intenção do modelador é especificar que um empregado, ao estar presente no banco de dados, deve ter registrada a sua preferência por pelo menos uma função, e esta preferência não tem momento inicial ou final de validade, é intemporal.

Para efeito de recuperação de informação, os relacionamentos intemporais são sempre válidos, desde que estejam presentes no banco de dados. Argumentos temporais, que porventura estejam presentes em consultas (*queries*), não restringem, ou melhor, não produzem qualquer efeito sobre o acesso a este tipo de relacionamento.

#### 4.4.2.3 Restrições de cardinalidade dos relacionamentos

A restrição de cardinalidade anotada junto às linhas que conectam conjuntos-entidade (retângulos) a conjuntos-relacionamento (losangos) refere-se à participação das entidades nos conjuntos-relacionamento. Esta forma de anotar as cardinalidades é a mesma adotada na metodologia Merise, segundo [FER 91], e em outros trabalhos publicados tais como [BAT 92]. Segundo este princípio de notação da cardinalidade, uma entidade *participa* de relacionamentos. Por exemplo, no diagrama TempER da figura 4.1, a notação "(1, 1)" que está junto à linha que liga o conjunto-entidade Empregado ao conjunto-relacionamento Lotação significa que uma instância do conjunto-entidade Empregado, ou seja um empregado qualquer, participa de no mínimo um e no máximo um relacionamento do conjunto Lotação em cada momento do tempo da sua existência como empregado. Por sua vez, a notação "(0,N)" junto à linha que conecta o conjunto-entidade Departamento ao conjunto-relacionamento Lotação especifica que um departamento pode não participar de qualquer relacionamento do conjunto Lotação, ou seja, é um departamento sem empregados, ou pode participar do conjunto-relacionamento um número indeterminado de vezes, a cada momento do tempo. Em outros termos, em cada momento do tempo da sua validade temporal, um empregado está alocado obrigatoriamente a um e a somente um departamento, e um departamento pode ter nenhum, ou ter qualquer número de empregados lotados.

O significado das restrições de cardinalidade *depende* da classificação temporal do conjunto-relacionamento. Caso o conjunto-relacionamento seja temporal, como é o caso do exemplo acima descrito, a leitura da cardinalidade necessariamente se dá em relação a cada ponto do tempo. Por outro lado, caso o conjunto-relacionamento seja intemporal, deixa de existir esta referência a pontos do tempo e a leitura das restrições de cardinalidade fica semelhante à do modelo ER convencional.

Para efeito deste trabalho, como pode ser visto na figura 4.5, limitou-se em quatro as possibilidades de restringir a cardinalidade de participação de uma entidade em um conjunto-relacionamento.

Nas situações em que o conjunto-relacionamento é *temporal*, as restrições de cardinalidade têm o seguinte significado:

- cardinalidade *no mínimo zero, no máximo um*, notação gráfica: "(0, 1)", especifica que uma entidade não precisa estar obrigatoriamente participando de um relacionamento, mas se estiver participando, deve ser apenas uma vez a cada momento do tempo.

- cardinalidade *no mínimo zero, no máximo N*, notação gráfica: "(0, N)", especifica que uma entidade não precisa estar obrigatoriamente participando de um relacionamento, ou pode estar participando de um ou mais relacionamentos em alguns ou todos os momentos do tempo da sua existência na realidade modelada. Esta cardinalidade, ao permitir todas as possibilidades de participação, não se configura, portanto, em uma restrição.
- cardinalidade *no mínimo um, no máximo um*, notação gráfica: "(1, 1)", especifica que uma entidade obrigatoriamente deve participar em cada momento do tempo da sua existência de exatamente um relacionamento, nem mais, nem menos.
- cardinalidade *no mínimo um, no máximo N*, notação gráfica: "(1, N)", especifica que uma entidade obrigatoriamente deve participar em cada momento do tempo da sua existência de no mínimo um relacionamento, podendo participar de mais de um.

Nas situações em que o conjunto-relacionamento é *intemporal*, as restrições de cardinalidade têm o significado abaixo, não havendo referências a momentos do tempo:

- cardinalidade *no mínimo zero, no máximo um*, notação gráfica: "(0, 1)", especifica que uma entidade não precisa estar obrigatoriamente participando de um relacionamento, ou se estiver participando, deve ser apenas uma vez.
- cardinalidade *no mínimo zero, no máximo N*, notação gráfica: "(0, N)", especifica que uma entidade não precisa estar obrigatoriamente participando de um relacionamento, ou pode estar participando de um ou mais relacionamentos. Também, aqui esta cardinalidade não se configura em uma restrição.
- cardinalidade *no mínimo um, no máximo um*, notação gráfica: "(1, 1)", especifica que uma entidade obrigatoriamente deve participar de exatamente um relacionamento, nem mais, nem menos.
- cardinalidade *no mínimo um, no máximo N*, notação gráfica: "(1, N)", especifica que uma entidade obrigatoriamente deve participar de no mínimo um relacionamento, podendo participar de mais de um.

O mapeamento do modelo TempER para diagrama ER convencional mostrado na figura 4.10 explica a diferença de ponto de vista entre as restrições de cardinalidade referentes a relacionamentos temporais e as restrições de cardinalidade referentes a relacionamentos intemporais. Tome-se como exemplo a participação dos empregados no conjunto-relacionamento temporal *Alocação*, a qual é restringida pela cardinalidade "(0, 1)". Tal restrição de cardinalidade especifica que um empregado pode não estar alocado a qualquer função, ou se estiver, a no máximo uma função a cada momento do tempo. A forma de representar este fato, no ER convencional correspondente, é relacionar a perspectiva temporal dos empregados (*Empregado-T*), que nada mais é que a associação destes com os pontos do tempo referentes à sua validade temporal, com a perspectiva temporal das funções (*Função-T*).

As restrições de cardinalidade, na forma como descritas acima, estão formalmente especificadas no capítulo 6 desta dissertação, através do mapeamento do modelo TempER para Redes de Petri de alto nível CEM.

### 4.4.3 Atributos

Tanto as entidades como os relacionamentos apresentam propriedades que os caracterizam, as quais são denominadas de atributos.

Os atributos dos conjuntos-entidade e conjuntos-relacionamento são especificados através do dicionário de dados, que complementa um diagrama TempER. A descrição de um atributo apresenta os seguintes elementos: o nome do atributo, o domínio dos valores primitivos que podem ser associados ao atributo e a classificação do atributo em relação ao tempo (indicando se o atributo é temporal ou intemporal). Um pequeno exemplo de como é formado um dicionário de dados encontra-se no diagrama inferior da figura 4.1.

Com o objetivo de simplificar os estudos, o modelo TempER por definição não apresenta atributos opcionais e tampouco atributos multivalorados.

Caso haja a necessidade de associar um conjunto de valores a um atributo deve-se lançar mão do conceito de construtores de domínios derivados. É o caso do atributo *fone* do conjunto-entidade *VendedorAutônomo* presente no dicionário de dados apresentado na figura 4.17. Para efeito de simplificação, apenas o construtor SET está disponível até o presente momento; outros construtores podem vir a ser incluídos futuramente. Ao utilizar o construtor SET, um atributo deixa de ter como conteúdo um valor primitivo e passa a armazenar um conjunto de valores primitivos como se fosse um valor atômico, pois cada átomo do domínio é agora um dos possíveis conjuntos que podem ser formado com os valores primitivos. Em outros termos, quando se associa o construtor SET com um domínio de valores primitivos, está se especificando que o domínio do atributo passa a ser o conjunto potência do conjunto de valores primitivos, ou seja, um domínio derivado.

Um conjunto-entidade normalmente possui um identificador de uso externo, uma chave primária. Em geral trata-se de um único atributo, no entanto pode resultar da composição de dois ou mais atributos. O valor de um identificador é único dentro do contexto de um conjunto-entidade, isto é, não existem duas instâncias com o mesmo valor para o identificador. O identificador deve sempre ser formado por atributo (s) intemporal (is). Exemplos de definição de identificadores são: *cod* do conjunto-entidade *Empregado* e *sigdep* do conjunto-entidade *Depto*, mostrados na figura 4.1.

As entidades, sejam elas transitórias ou perenes, por apresentarem duas perspectivas em relação ao tempo (ver figura 4.4), podem combinar atributos temporais com atributos intemporais. Os relacionamentos não: se forem temporais podem possuir apenas atributos temporais, se forem intemporais podem apresentar apenas atributos intemporais.

#### 4.4.3.1 Atributos temporais

Os atributos temporais são aqueles cuja valoração (conteúdo) deve ser referenciada a pontos do tempo. O atributo *at-sal* das entidades do conjunto *Empregado* é um exemplo de atributo temporal. Como pode ser visto na parte inferior da figura 4.4, este atributo apresenta um valor a cada momento do tempo. É importante salientar que, primeiramente não há nada impedindo que um mesmo valor apareça em pontos diferentes do tempo, e, em segundo lugar que, a série de valores que um atributo temporal apresenta ao longo do tempo, não deve ser confundida com multivaloração de atributo, pois a cada momento do tempo apenas um e somente um valor é permitido.

O objetivo dos atributos temporais é permitir que se registre a evolução dos valores de uma propriedade ao longo do tempo.

Em virtude de no modelo TempER não existir atributo opcional, um atributo temporal deve obrigatoriamente apresentar um valor para cada momento da existência das entidades ou relacionamentos temporais a que pertença. Quando se tratar de entidade perene, isto implica em que o atributo temporal apresenta valor em todos os pontos do eixo temporal.

Na figura 4.11, a associação de um atributo temporal a uma entidade transitória é ilustrada através do mapeamento do modelo TempER para diagrama ER convencional. Trata-se do atributo *at-sal*, que para cada empregado contém toda a sua história salarial. No diagrama ER convencional este atributo aparece conectado ao conjunto-entidade especial que modela a perspectiva temporal dos empregados (*Empregado-T*), ou seja, para cada par  $\langle \text{OID do empregado, ponto do tempo} \rangle$  existe um e apenas um valor de salário.

Na figura 4.12, a associação de um atributo temporal a uma entidade do tipo perene é ilustrada também via o mapeamento para um diagrama ER convencional. Trata-se do atributo *at-salref* que, para cada função, contém todos os valores de referência que houveram ao longo do tempo. Por pertencer a uma entidade perene, este atributo deve apresentar valor em todos os momentos do eixo temporal. O exemplo de valoração referente à função de OID 70101, que aparece na figura, mostra como é possível cobrir todo o eixo temporal. Basta que o rótulo temporal do primeiro valor da série apresente como ponto inicial o símbolo “«”, e rótulo temporal do último valor da série apresente como ponto final o símbolo “»”.

A valoração dos atributos temporais é representada através de um *assinalamento temporal*, que se trata de um conceito emprestado de [GAD 88,93] e cuja descrição se encontra no capítulo anterior.

#### 4.4.3.2 Atributos intemporais

Os atributos intemporais são aqueles cujos valores não apresentam qualquer associação com a dimensão temporal. Através dos atributos intemporais é possível representar os atributos convencionais do modelo Entidade-Relacionamento. Exemplos deste tipo de atributo são: *cod* e *nome* do conjunto-entidade *Empregado*, da figura 4.11, e também, *codfun* e *descrfun* do conjunto-entidade *Função*, da figura 4.12.

Os atributos intemporais existem apenas na perspectiva intemporal das entidades ou associados a relacionamentos intemporais. As figuras 4.11 e 4.12 ilustram, através do mapeamento para ER convencional, de que forma os atributos intemporais são abordados pelo modelo TempER.

Em virtude de não apresentarem qualquer subordinação ao eixo temporal, os atributos intemporais estão vinculados apenas aos OID's das entidades. Portanto, diferentemente dos atributos temporais que estão restritos aos momentos da validade temporal das entidades, os seus valores sempre são *vigentes*, qualquer que seja o argumento temporal que porventura restrinja uma possível recuperação de informação; basta que as respectivas entidades estejam presentes no banco de dados, não importando quais sejam as suas *existências*.

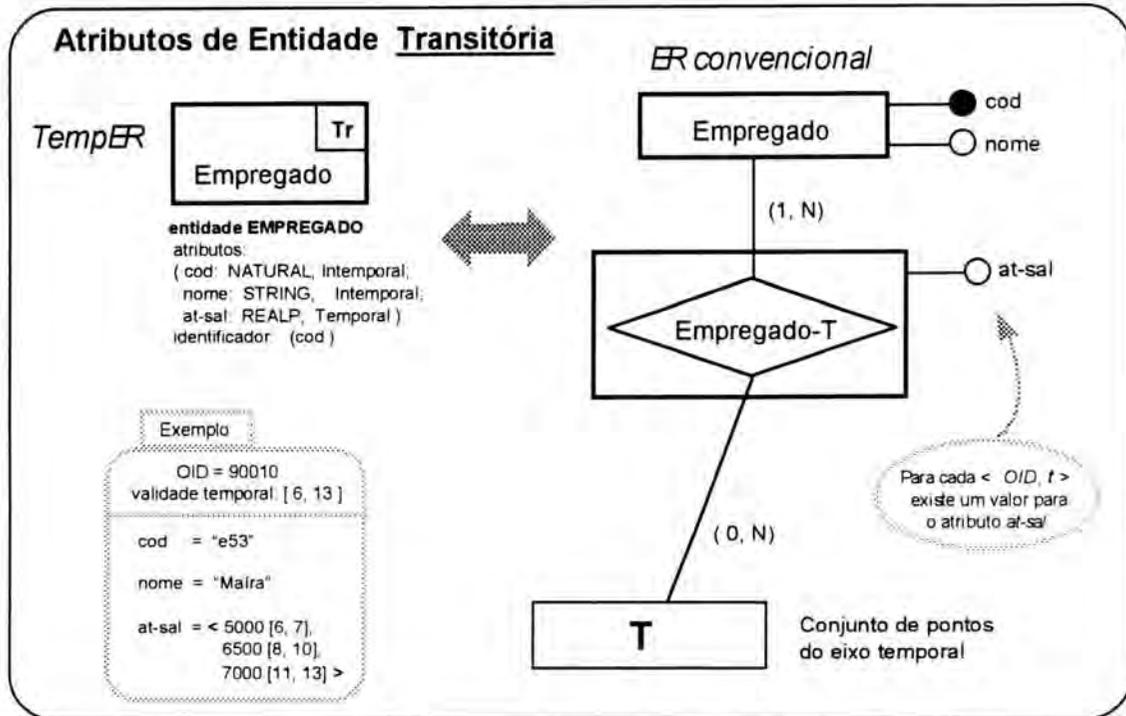


FIGURA 4.11 - Mapeamento de atributos de entidade transitória para ER convencional

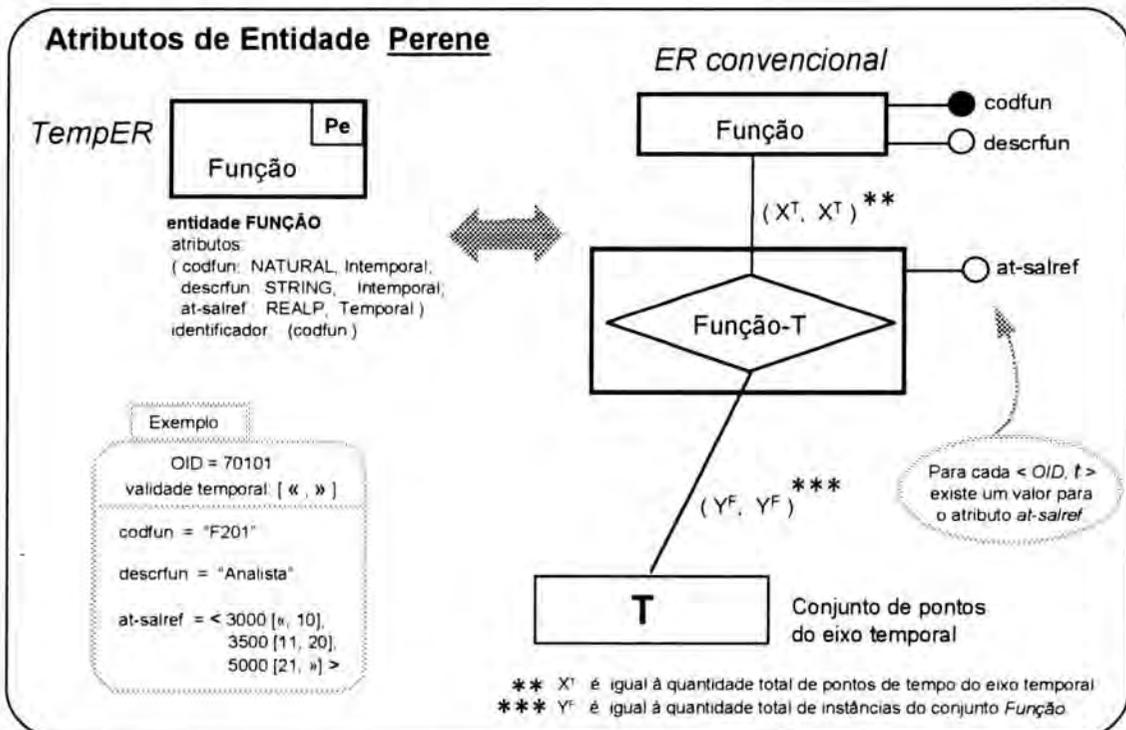


FIGURA 4.12 - Mapeamento de atributos de entidade perene para ER convencional

No capítulo 6 desta dissertação os atributos temporais e intemporais são definidos formalmente através do mapeamento do modelo de dados TempER para Redes de Petri de alto nível.

## 4.5 Dicionário de dados

Complementando os diagramas TempER, existe o dicionário de dados, cuja finalidade é permitir a especificação dos atributos das entidades e relacionamentos. A elaboração deste dicionário de dados deve obedecer às regras de formação abaixo apresentadas, especificadas em BNF, bem como deve respeitar as regras adicionais e observar as convenções e comentários descritos ao final desta seção.

### 4.5.1 Sintaxe do dicionário de dados em BNF

```

< dicionário de dados > ::= < descritor de objeto > | { < descritor de objeto > } ... |

< descritor de objeto > ::= < conjunto-entidade > | < conjunto-relacionamento >

< conjunto-entidade > ::= entidade < nome de entidade >
                        atributos: < descritores de atributos >
                        identificador: < ( > { < lista de atributos > | < - > } < ) >

< conjunto-relacionamento > ::= relacionamento < nome de relacionamento >
                                atributos: { < descritores de atributos > | < ( - ) > }

< descritores de atributos > ::= < ( > < especificação de atributo >
                                [ { < ; > < especificação de atributo > } ... ] < ) >

< especificação de atributo > ::= < nome de atributo > < : >
                                < domínio de valores > < , >
                                < classificação temporal >

< lista de atributos > ::= < nome de atributo > { { < , > < nome de atributo > } ... }

< domínio de valores > ::= < tipo de dado > |
                            < SET > < ( > < tipo de dado > < ) >

< tipo de dado > ::= NATURAL | INTEIRO | REAL | REALP | STRING

< classificação temporal > ::= Temporal | Intemporal

< nome de entidade > ::= // Sequência de letras e números, o primeiro caracter
                        deve ser letra maiúscula, pode conter "-" e "_". //

< nome de relacionamento > ::= // Sequência de letras e números, o primeiro
                                caracter deve ser letra maiúscula, pode conter "-" e "_". //

< nome de atributo > ::= // Sequência de letras e números, o primeiro caracter deve ser
                        letra minúscula, pode conter "-" e "_". //

```

## 4.5.2 Regras adicionais de formação do dicionário de dados

Algumas regras adicionais devem ser obedecidas na formação do dicionário de dados, a saber:

- O atributos que participam da composição do identificador de um conjunto-entidade devem ser todos do tipo intemporal.
- Nomes de entidades e de relacionamentos devem ser únicos no dicionário de dados.
- Dentro do contexto de um conjunto-entidade, ou conjunto-relacionamento, os nomes de atributos devem ser únicos.

## 4.5.3 Convenções do dicionário de dados

Visando padronizar os termos utilizados para denominar os elementos do dicionário de dados, algumas convenções são sugeridas:

- Recomenda-se utilizar substantivos no singular para nomes de conjuntos-entidade e conjuntos-relacionamento, a não ser que o uso de substantivo no plural seja mais eloqüente no contexto da modelagem. Devem ser evitados verbos para denominar conjuntos-relacionamento, pois tendem a informalmente estabelecer uma ordem de participação das entidades no relacionamento.
- Recomenda-se que os nomes de conjuntos-entidade e conjuntos-relacionamento compostos de mais de uma palavra sejam escritos justapondo-se as palavras, iniciando com letra maiúscula cada palavra além da primeira. Ex.: conjuntos-entidade: PedidoCompra e BeneficioComplementar, conjunto-relacionamento: DireitoDeAcesso, atributos: salárioBase e preçoMáximo.

## 4.5.4 Comentários

Para efeito do presente trabalho os tipos de dados permitidos são suficientes. Estes têm o seguinte significado: NATURAL refere-se ao conjunto dos números naturais, INTEIRO refere-se ao conjunto dos números inteiros, REAL refere-se ao conjunto dos números reais, REALP refere-se ao conjunto dos números reais positivos inclusive o zero, STRING refere-se a qualquer combinação de letras, números e caracteres.

## 4.6 Generalização / especialização

Em certas situações é necessário que o modelador especifique relações do tipo "é um" ("ISA") entre conjuntos de entidades. É o caso dos exemplos mostrados na figura 4.13. O diagrama (a) ilustra uma especialização que ocorre na *perspectiva temporal* das entidades, ou seja, um vendedor pode ser em um momento do tempo vendedor-funcionário e em outro momento do tempo vendedor-autônomo, entretanto não pode ter os dois papéis ao mesmo tempo, pois se trata de especialização do tipo disjunta. O diagrama (b), por outro lado, mostra uma especialização que se dá na *perspectiva intemporal* das entidades, isto é, um empregado que esteja presente no banco de dados, ou é homem ou é mulher, não se levando em conta os momentos da validade temporal deste empregado.

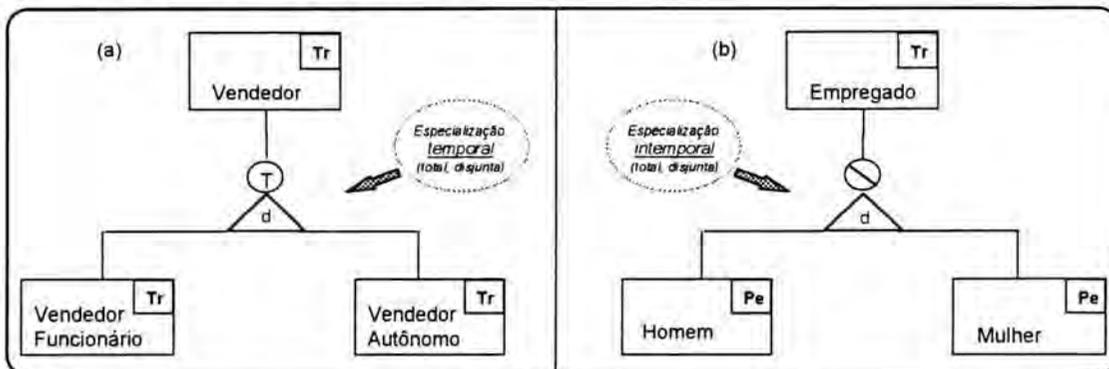


FIGURA 4.13 - Aplicação da estrutura Generalização/Especialização

Generalização e especialização são termos que correspondem ao mesmo conceito, a diferença está no sentido da visualização. É especialização quando se enfoca a partir do conjunto-entidade hierarquicamente superior em direção aos conjuntos-entidade que estão no nível hierárquico inferior. No sentido inverso é generalização. Um conjunto-entidade, que esteja num nível hierarquicamente superior em relação a outros, é chamado no âmbito deste trabalho, de conjunto-super-entidade. Um conjunto-entidade que esteja num nível hierárquico inferior em relação a outro é chamado de conjunto-sub-entidade. Uma estrutura generalização/especialização pode se repetir recursivamente, isto é, pode ter vários níveis hierárquicos à semelhança da estrutura de uma árvore. Sendo assim, um mesmo conjunto-entidade pode ser conjunto-sub-entidade de um conjunto superior e também conjunto-super-entidade de conjuntos inferiores.

### 4.6.1 Notação de Generalização/Especialização

O símbolo característico da estrutura generalização/especialização é um pequeno triângulo que associa um conjunto-super-entidade aos respectivos conjuntos-sub-entidade. O conjunto-super-entidade da estrutura é o conjunto-entidade conectado ao vértice superior do triângulo. Os conjuntos-sub-entidade da estrutura são os conjuntos-entidade conectados à base do triângulo. Esta definição deve ser "relativizada" quando se desenha o diagrama no sentido horizontal. Além do triângulo, há um pequeno círculo anexo ao canto superior do mesmo que define em qual perspectiva relativa ao tempo se insere o relacionamento de generalização/especialização. Se tal círculo contiver a letra "T", a generalização/especialização se insere na *perspectiva temporal* dos conjuntos-

entidade envolvidos. Por outro lado, se contiver um segmento de reta oblíquo, a generalização/especialização se insere na *perspectiva intemporal*.

Na figura 4.14 são apresentadas os quatro tipos possíveis de estrutura generalização/especialização do tipo temporal. A generalização/especialização intemporal também apresenta os mesmos tipos, a única diferença de notação diz respeito ao conteúdo do pequeno círculo, como explicado no parágrafo anterior.

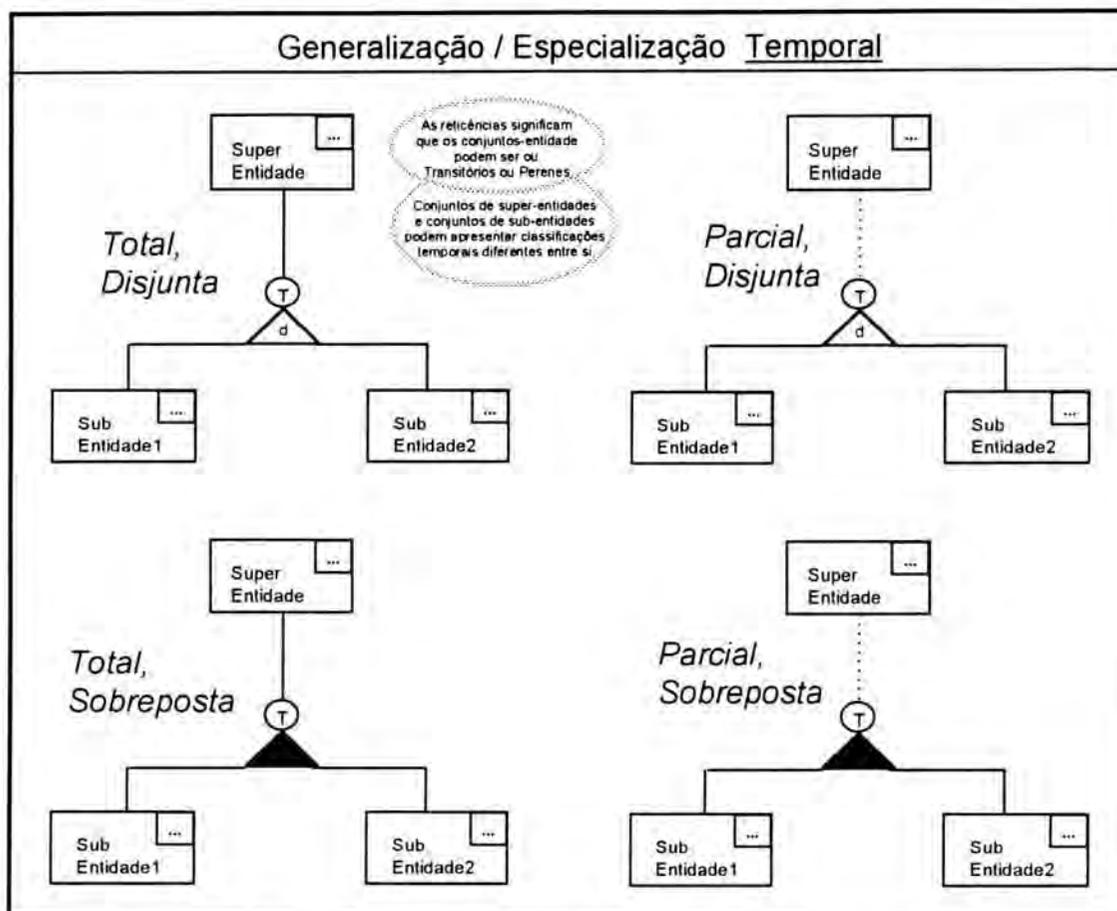


FIGURA 4.14 - Os quatro tipos possíveis de estrutura generalização/especialização

#### 4.6.2 Generalização / Especialização Temporal

Uma especialização do tipo temporal é análogo a um relacionamento temporal, pois ambos ocorrem no âmbito da perspectiva temporal das entidades associadas, isto é, o que se especializa são os pares  $\langle OID, t \rangle$ , onde  $OID$  identifica a entidade e  $t$  identifica um ponto do tempo qualquer da existência desta entidade.

Através da generalização/especialização temporal, que pode ser total ou parcial, e, disjunta ou sobreposta, é possível modelar a evolução dos papéis que as entidades podem apresentar ao longo de sua existência.

Uma especialização temporal se diz *total* se, sempre que uma entidade estiver presente no conjunto-super-entidade, ela obrigatoriamente estará presente em pelo menos um dos conjuntos-sub-entidade, em cada momento da sua validade temporal. Se diz *parcial* se não houver esta obrigatoriedade, isto é, pode estar no conjunto-super-entidade e não ter sido especializada em qualquer um dos conjuntos-sub-entidade. O

conceito *total* é representado por uma linha contínua ligando o conjunto-super-entidade ao triângulo e o conceito *parcial* é representado por uma linha pontilhada.

Uma especialização se diz *disjunta* quando no mesmo nível hierárquico uma entidade só pode estar presente em um único conjunto-sub-entidade, em cada momento do tempo. Se diz *sobreposta* se uma entidade pode estar presente em mais de um conjunto-sub-entidade do mesmo nível hierárquico, em um momento do tempo. Uma especialização *disjunta* é representada por um triângulo contendo a letra "d". Uma especialização *sobreposta* é representada por um triângulo cheio.

Existe uma restrição intrínseca à estrutura generalização/especialização do tipo temporal que determina que, se uma entidade está presente em um conjunto-sub-entidade, em um certo momento do tempo, necessariamente deverá estar presente no conjunto-super-entidade correspondente no mesmo momento do tempo, ou seja o conjunto de pontos que define a validade temporal de uma sub-entidade é um subconjunto do conjunto de pontos que define a validade temporal da respectiva super-entidade.

Uma forma de ilustrar o significado da estrutura generalização/especialização do tipo temporal é através do seu mapeamento para um diagrama ER convencional, como é mostrado na figura 4.15.

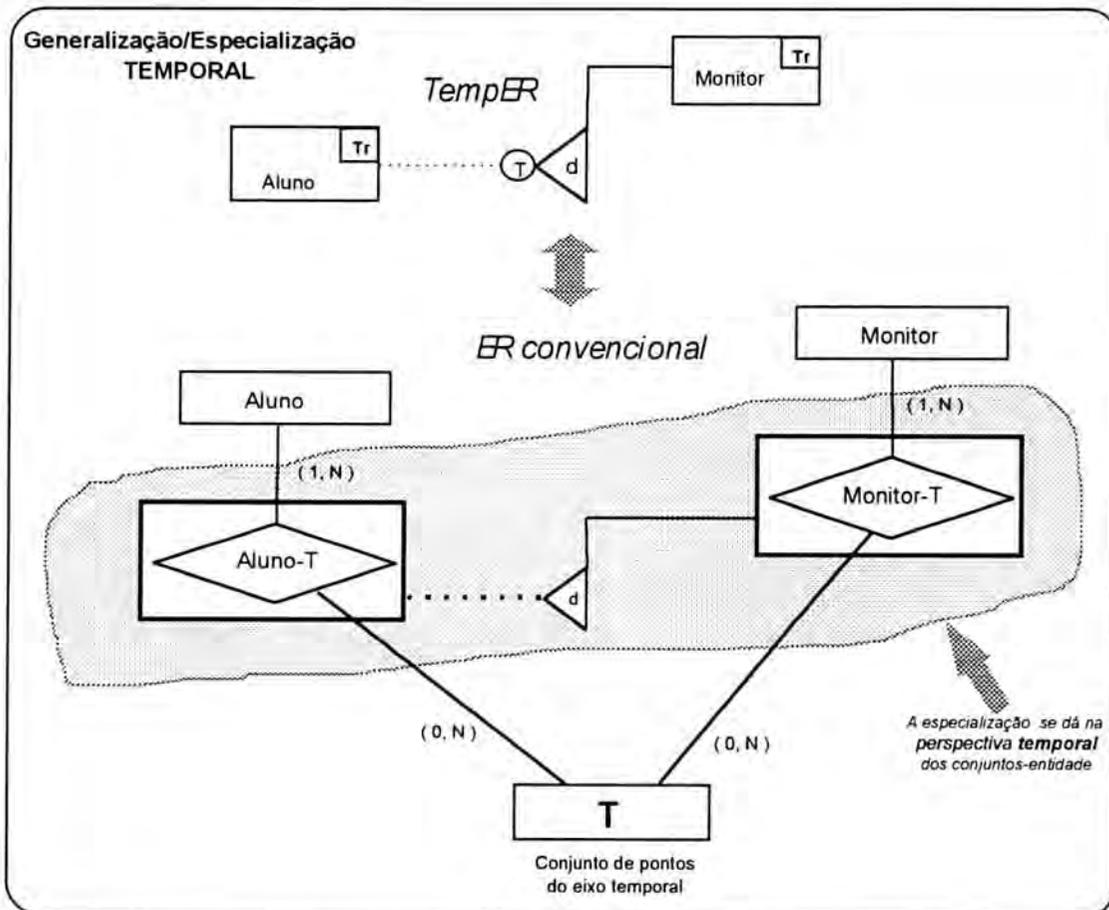


FIGURA 4.15 - Mapeamento da estrutura generalização/especialização *temporal* para um diagrama ER convencional

Pelo exemplo da figura, um aluno pode em certos momentos da sua validade temporal desempenhar o papel de monitor. Para representar este fato em ER convencional é necessário especializar o conjunto-super-entidade *Aluno-T* no conjunto-sub-entidade *Monitor-T*. O primeiro contém os relacionamentos dos alunos com os momentos do eixo temporal, e o segundo contém os relacionamentos dos monitores com os pontos do tempo. Como se trata de uma especialização parcial, um aluno em certos momentos do tempo pode não ser monitor e em outros ser.

### 4.6.3 Generalização / Especialização Intemporal

Uma especialização do tipo intemporal é análoga a um relacionamento intemporal, ambos ocorrem no âmbito da perspectiva intemporal das entidades associadas. Nesse tipo de especialização, apenas os OID's estão envolvidos, não há referência a pontos do tempo.

Assim como ocorre com a especialização temporal, a intemporal também pode ser total ou parcial, e, disjunta ou sobreposta. Uma especialização intemporal se diz *total* se, sempre que uma entidade estiver presente no conjunto-super-entidade, ela obrigatoriamente estará presente em pelo menos um dos conjuntos-sub-entidade. Se diz *parcial* se não houver esta obrigatoriedade, isto é, pode estar no conjunto-super-entidade e não ter sido especializada em qualquer um dos conjuntos-sub-entidade. O conceito *total* é representado por uma linha contínua, ligando o conjunto-super-entidade ao triângulo; o conceito *parcial*, por sua vez, é representado por uma linha pontilhada.

Uma especialização se diz *disjunta* quando, no mesmo nível hierárquico, uma entidade só pode estar presente em um único conjunto-sub-entidade. Se diz *sobreposta* se uma entidade pode estar presente em mais de um conjunto-sub-entidade do mesmo nível hierárquico. Uma especialização *disjunta* é representada por um triângulo contendo a letra "d". Uma especialização *sobreposta* é representada por um triângulo cheio.

Existe uma restrição intrínseca à estrutura generalização/especialização do tipo intemporal que determina que, se uma entidade está presente em um conjunto-sub-entidade, necessariamente deverá estar presente no conjunto-super-entidade correspondente, isto é, se o OID desta estiver no subconjunto, também deve estar no superconjunto.

A generalização/especialização do tipo intemporal também pode ser explicada através do seu mapeamento para um diagrama ER convencional, como ilustra a figura 4.16, onde um aluno pode ser um estrangeiro. Por se tratar de uma especialização parcial, não necessariamente todos os alunos precisam ser estrangeiros, alguns não estarão presentes no conjunto-sub-entidade *Estrangeiro*. Ao se dar na perspectiva intemporal, a especialização em questão não apresenta qualquer referência a momentos do tempo, restringindo-se pura e simplesmente aos OID's das entidades.

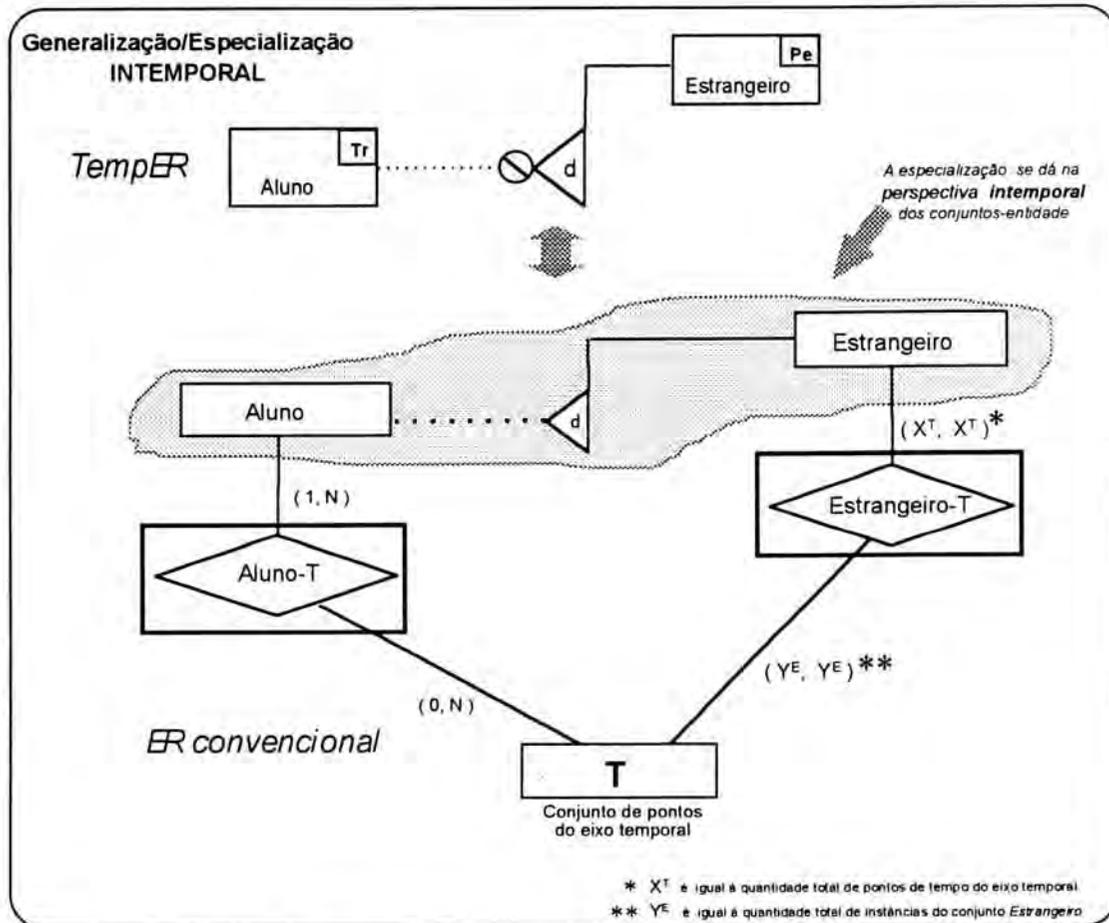


FIGURA 4.16 - Mapeamento da estrutura generalização/especialização *intemporal* para um diagrama ER convencional

#### 4.6.4 Atributo identificador de sub-entidade

Uma instância que esteja num conjunto-entidade definido como conjunto-sub-entidade pode não apresentar identificador, pois este está presente no conjunto-super-entidade correspondente. Entretanto, não é incorreto, caso se deseje, definir um identificador ao nível do conjunto-sub-entidade. É o caso do identificador *idfunc* do conjunto-entidade *VendedorFuncionário* na figura ao lado. Na verdade, cada instância de *VendedorFuncionário* tem dois identificadores, um indireto (*codv*) e outro direto (*idfunc*). Caso o acesso seja feito através do identificador indireto, primeiro será necessário navegar pela instância que está no conjunto-super-entidade, para depois se chegar na instância que está no conjunto-sub-entidade. Isto só é possível porque ambas as instâncias possuem o mesmo OID, ou seja referem-se ao mesmo objeto.

<b>DICIONÁRIO DE DADOS</b>	
<b>entidade Vendedor</b>	
atributos:	( codv: NATURAL, Intemporal; nome: STRING, Intemporal )
identificador:	( codv )
<b>entidade VendedorAutônomo</b>	
atributos:	( fone: SET (NATURAL), Intemporal; endtrab: STRING, Intemporal )
identificador:	( - )
<b>entidade VendedorFuncionário</b>	
atributos:	( idfunc: NATURAL, Intemporal; salário: REALP, Tempora )
identificador:	( idfunc )

FIGURA 4.17 - Dicionário de dados ref. diagrama da fig. 4.13(a)

## 5 Modelagem de transações combinada com modelo de dados temporal

O modelo TempER-Tr proposto nesta dissertação combina em um único diagrama a representação das propriedades estáticas e dinâmicas de um sistema de informação. As propriedades estáticas são modeladas através do modelo entidade-relacionamento temporal apresentado no capítulo anterior. As propriedades dinâmicas são especificadas pelo modelo de transações que será descrito neste capítulo.

O foco da modelagem dinâmica são as transações executadas no interior do sistema, em resposta a eventos que ocorrem no ambiente externo. As transações de mesmo comportamento, isto é, que estão associadas ao mesmo tipo de evento externo e que provocam alterações de estado nos mesmos conjuntos de entidades e relacionamentos do banco de dados, são agrupadas em *conjuntos-transação*. Sendo assim, o que se modela na verdade são estes conjuntos-transação. Por exemplo, o conjunto-transação *AdmitirEmpregado* diagramado na figura 5.1 modela todas as admissões de empregados que podem ocorrer ao longo do tempo. Cada vez que um empregado é admitido, instancia-se uma transação. O fato que determina a execução de uma transação é o aparecimento de uma nova tupla de valores no fluxo de entrada conectado ao evento externo *AdmissãoDeEmpregado*.

Utilizou-se o termo transação, emprestado da área de banco de dados, com o propósito de aproveitar o conceito de atomicidade. Por definição, todos os efeitos de uma transação ocorrem coincidentemente, configurando-se em uma *transição atômica* de um estado do banco de dados para outro, não havendo nenhum estado intermediário.

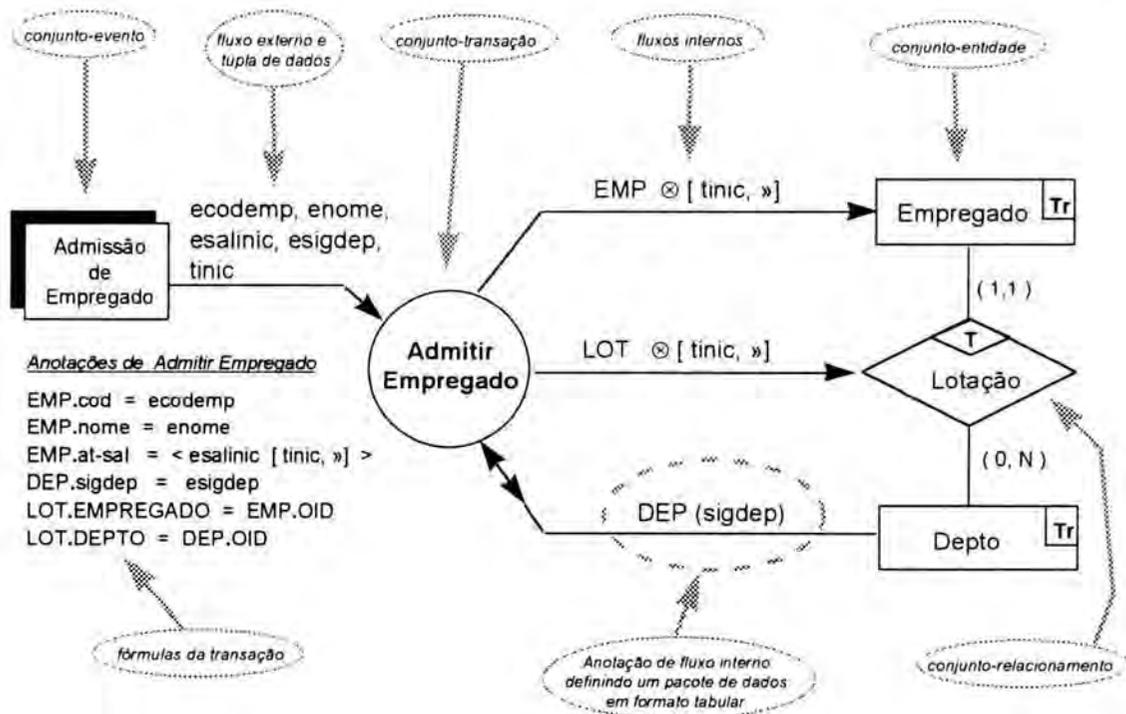


FIGURA 5.1 - Modelagem de uma transação

A modelagem de transações proposta guarda certa semelhança com a abordagem conhecida como análise essencial [McM 90], onde o modelador deve, a partir de cada tipo de evento do ambiente, especificar um diagrama contendo a atividade que o sistema deve executar para responder a tal evento. A diferença reside no formalismo que fundamenta o modelo TempER-Tr.

As transações do modelo TempER-Tr comportam-se de forma semelhante a uma Rede de Petri de alto nível, tipo condição-evento, ou seja, existem regras que definem quando uma transação está habilitada e regras que definem a efetivação de uma transação. Assim sendo, ao se modelar um determinado conjunto-transação, são especificadas as condições que restringem a ocorrência de uma transação, como também, são especificados as transformações que o banco de dados irá sofrer.

## 5.1 Elementos dinâmicos do modelo TempER-Tr

Um diagrama TempER-Tr é composto de dois grupos de elementos: um grupo representando as propriedades estáticas do sistema (o modelo ER temporal) e um grupo representando as propriedades dinâmicas do sistema (o modelo de transações). Os elementos estáticos são aqueles descritos no capítulo anterior, os elementos dinâmicos são os seguintes:

### 5.1.1 Eventos externos

Os tipos de eventos externos que afetam o sistema são representados por retângulos sombreados. Cada retângulo destes representa um conjunto de eventos de mesma natureza. Por exemplo: todas as admissões de empregados (cada uma delas é um evento) são agrupadas no tipo de evento *AdmissãoDeEmpregado*, conforme pode ser visto na figura 5.1. Para efeito desta dissertação, um determinado conjunto de eventos externos é denominado de *conjunto-evento*.

### 5.1.2 Fluxos externos

Os fluxos externos são setas que conectam um conjunto-evento ao círculo que define um conjunto de transações. Quando a seta está no sentido do conjunto de transações, os fluxos externos representam entrada de dados no sistema, caso contrário representam saída de dados, como é mostrado na figura 5.2.



FIGURA 5.2 - Notação dos fluxos externos

Cada vez que um evento ocorre no ambiente externo, uma tupla de dados é introduzida no fluxo externo de entrada. Este fato constitui-se no estímulo que resulta na execução de uma determinada transação, a qual é uma instância do conjunto de transações conectado. Um exemplo de fluxo externo é a linha que conecta o conjunto-evento *AdmissãoDeEmpregado* ao conjunto de transações *AdmitirEmpregado* na figura 5.1. Os dados referentes a um fluxo externo são especificados pela tupla de variáveis anotada junto a ele.

As respostas que uma transação pode gerar ao ambiente externo são modeladas através dos fluxos externos de saída.

### 5.1.3 Transações

As transações são as atividades que ocorrem no interior do sistema, decorrentes de eventos externos. Ao conjunto de transações que apresentam as mesmas características dá-se o nome de *conjunto-transação*. Cada valoração diferente dos fluxos externos de entrada conectados ao círculo que representa um conjunto-transação, instancia uma determinada transação deste conjunto. Um conjunto-transação é representado por um círculo, como é o caso de *AdmitirEmpregado* da figura 5.1. Por convenção, a denominação do conjunto-transação deve iniciar com um verbo no infinitivo.

### 5.1.4 Fluxos internos

São linhas que conectam um conjunto-transação a conjuntos-entidade ou a conjuntos-relacionamento do modelo ER temporal. Podem ser de vários tipos: fluxos de inclusão, fluxos de exclusão, fluxos de alteração, fluxos de verificação de presença e fluxos de verificação de ausência, conforme mostram as figuras 5.3 e 5.4. Os dados referentes aos fluxos internos são definidos pelas anotações que estão junto a eles. Como princípio geral, cada uma destas anotações especifica um *pacote de dados em formato de tabela*. Será uma tabela de entidades, quando o elemento conectado for um conjunto-entidade, ou uma tabela de relacionamentos quando o elemento conectado for um conjunto-relacionamento. Existe uma distinção gráfica entre as anotações que se referem a pacotes de dados contendo um único elemento e a pacotes que se referem a um número variável de elementos: estas últimas são prefixadas por um asterisco. Os fluxos de inclusão, de exclusão e de alteração expressam que as respectivas transações modificam o conteúdo dos conjuntos-entidade / conjuntos-relacionamento a que estão conectados. Os fluxos de verificação de presença e de ausência apenas testam o conteúdo destes lugares, sem alterá-los. Os fluxos de alteração apenas modificam valores de atributos, não incluem, nem excluem entidades ou relacionamentos, e também não ampliam e nem reduzem a dimensão temporal de qualquer objeto. Na figura 5.1 as linhas conectando o conjunto-transação *AdmitirEmpregado* ao conjunto-entidade *Empregado*, ao conjunto-relacionamento *Lotação* e ao conjunto-entidade *Depto* são exemplos de fluxos internos.

## 5.2 Tipos de fluxos internos

Os fluxos internos classificam-se em dois grupos gerais. O primeiro grupo, cuja simbologia encontra-se na figura 5.3, serve para especificar a modificação de estado do banco de dados do sistema. Pertencem a este grupo os seguintes tipos de fluxos:

- fluxos internos de inclusão,
- fluxos internos de exclusão,
- fluxos internos de manipulação de existências de entidades temporais,
- fluxos internos de alteração de atributos.

O segundo grupo de fluxos internos, cuja simbologia é apresentada na figura 5.4, é utilizado para modelar a verificação de estado do banco de dados do sistema, isto é, estes fluxos não provocam qualquer modificação. Fazem parte deste grupo os seguintes tipos de fluxos:

- fluxos internos de verificação de presença,
- fluxos internos de verificação de ausência

A seguir são descritas todas as possibilidades de utilização dos tipos de fluxos acima citados.

### 5.2.1 Fluxos internos de inclusão

Este tipo de fluxo é representado por uma seta que parte do círculo que representa o conjunto-transação e aponta para o conjunto-entidade ou conjunto-relacionamento onde um novo ou novos elementos deverão surgir. Conforme o tipo de conjunto de objetos que a seta conecta, os fluxos internos de inclusão podem ser:

#### *Fluxo de inclusão de entidade transitória*

Este tipo de fluxo especifica a inclusão de uma ou mais entidades transitórias no conjunto-entidade conectado. Se a anotação junto ao fluxo definir um pacote de dados unitário (o nome do pacote não é precedido por asterisco), uma e apenas uma entidade está sendo incluída. Se a anotação definir uma pacote de dados multi-elementos, uma ou mais entidades estão sendo incluídas. A inclusão de uma entidade transitória tem o seguinte significado: um novo OID surge no banco de dados do sistema e, além disto, este OID ocupa um determinado conjunto de pontos no eixo temporal conforme definido pela anotação referente a tempo associada ao fluxo. A este conjunto de pontos do tempo dá-se o nome de existência ou validade temporal da entidade. O resultado de um fluxo de inclusão de entidade transitória é mostrado na figura 5.6 que registra o efeito de uma possível transação *AdmitirEmpregado*, cujos dados são mostrados na figura 5.5. Neste tipo de fluxo é obrigatório que se referencie a dimensão temporal nas anotações.

#### *Fluxo de inclusão de entidade perene*

Especifica a inclusão de uma ou mais entidades perenes. Como foi descrito no capítulo anterior, estas entidades apresentam como dimensão temporal implícita todos os pontos do eixo temporal, sendo, portanto, impossível ampliar ou reduzir as suas existências, restando apenas a possibilidade de manipular os OID's. Desta forma, toda vez que uma entidade perene é incluída, um novo OID é gerado e é assumido que a sua existência ocupa todo o eixo temporal. Como regra, a anotação identificando as entidades perenes a serem incluídas não pode fazer qualquer menção a pontos do tempo, tendo em vista o caráter implícito da dimensão temporal deste tipo de entidade.

#### *Fluxo de inclusão de relacionamento temporal*

Especifica a inclusão de um ou mais relacionamentos temporais no conjunto-relacionamento conectado. Tendo em vista que um relacionamento temporal necessariamente está associado a um conjunto determinado de pontos do eixo temporal, quando da sua inclusão no banco de dados, deve-se explicitar quais são estes pontos. Se um dos relacionamentos a ser incluído já está presente no banco de dados em outro período de tempo, este fluxo tem o efeito de "reencarnar" tal relacionamento, isto é, um novo conjunto de pontos de tempo é adicionado à sua existência.

### Fluxo de inclusão de relacionamento intemporal

Especifica a inclusão de um ou mais relacionamentos intemporais no conjunto-relacionamento conectado. Um relacionamento intemporal não apresenta qualquer referência ao eixo temporal, apenas os OID's das entidades associadas participam dele.

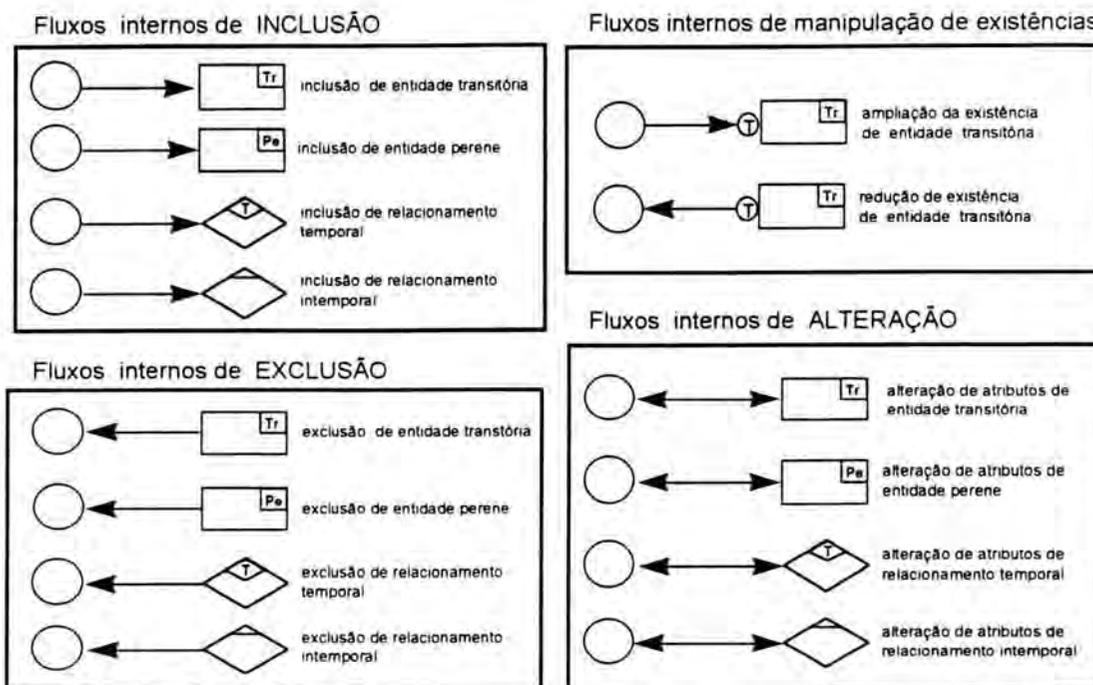


FIGURA 5.3 - Tipos de fluxos internos que provocam mudança de estado no banco de dados

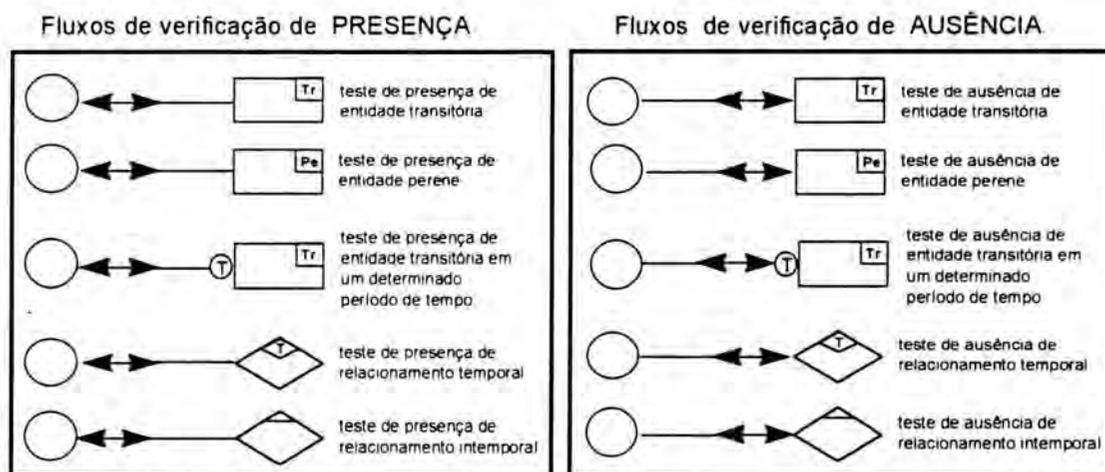


FIGURA 5.4 - Tipos de fluxos internos de verificação de presença/ausência

### 5.2.2 Fluxos internos de exclusão

Este tipo de fluxo é representado por uma seta que, partindo do conjunto-entidade ou conjunto-relacionamento de onde um ou mais elementos deverão desaparecer, aponta para o círculo que representa o conjunto-transação. Conforme o tipo de conjunto de objetos que a seta conecta, os fluxos internos de exclusão podem ser:

### Fluxo de exclusão de entidade transitória

Este tipo de fluxo especifica a exclusão de uma ou mais entidades transitórias do conjunto-entidade conectado. Se a anotação junto ao fluxo definir um pacote de dados unitário (o nome do pacote não é precedido por asterisco), uma e apenas uma entidade está sendo excluída. Se a anotação definir um pacote multi-elementos, uma ou mais entidades estão sendo excluídas. A exclusão de uma entidade transitória tem o seguinte significado: um OID desaparece do sistema e, além disto, toda a sua “existência” (o conjunto de pontos do tempo associado a este OID) também é excluída.

### Fluxo de exclusão de entidade perene

Especifica a exclusão de uma ou mais entidades perenes. Como a dimensão temporal de uma entidade perene é implícita e é sempre igual a todos os pontos do eixo temporal, assume-se que quando da sua exclusão, toda a sua existência também está sendo excluída. Como regra, a anotação junto ao fluxo que identifica as entidades perenes a serem excluídas, não pode fazer qualquer menção a pontos do tempo, tendo em vista o caráter implícito da dimensão temporal deste tipo de entidade.

### Fluxo de exclusão de relacionamento temporal

Especifica a exclusão de existências de um ou mais relacionamentos temporais do conjunto-relacionamento conectado. Caso o período de tempo informado na anotação junto ao fluxo seja igual a toda a existência do relacionamento temporal, este automaticamente desaparecerá do banco de dados. Por outro lado, se o período de tempo informado for um subconjunto de pontos de tempo da existência do relacionamento temporal, que é o caso mais comum, este relacionamento ainda permanecerá no banco de dados, contudo, a sua validade temporal sofrerá uma redução.

### Fluxo de exclusão de relacionamento intemporal

Especifica a exclusão de um ou mais relacionamentos intemporais do conjunto-relacionamento conectado. Por não apresentar qualquer referência ao eixo temporal, a anotação do fluxo que identifica o relacionamento ou relacionamentos a serem excluídos não pode apresentar qualquer menção a pontos do tempo.

## **5.2.3 Fluxos internos de manipulação de existência de entidades transitórias**

As entidades do tipo transitórias, diferentemente das entidades perenes, podem ter as suas existências ampliadas ou reduzidas. Isto se dá através de um tipo especial de fluxo, cuja característica é estar conectado a um pequeno círculo contendo a letra “T”, anexo ao conjunto-entidade, como pode ser visto na figura 5.3.

### Fluxo de ampliação da existência de entidade transitória

Especifica a inclusão de novos pontos do tempo na existência de uma ou mais entidades transitórias, ou seja, não cria novos OID's, apenas estende as suas existências. Este tipo de fluxo permite modelar a “reencarnação” de entidades, à semelhança do que está descrito em [CLI 88a]. Um exemplo deste conceito de reencarnação é o caso do empregado que, após ter sido demitido da empresa, é readmitido em um tempo futuro, reutilizando o mesmo OID.

#### Fluxo de redução de existência de entidade transitória

Especifica a exclusão de pontos do tempo da existência de uma ou mais entidades transitórias do conjunto-entidade conectado. Ou seja, a entidade ou entidades permanecem no banco de dados do sistema, mas as suas existências sofrem redução.

### **5.2.4 Fluxos internos de alteração**

Os fluxos de alteração apenas afetam o conteúdo dos atributos de entidades e relacionamentos. As anotações associadas a este tipo de fluxo devem especificar quais atributos serão alterados e também devem restringir a abrangência temporal da alteração, isto quando, entre os atributos a serem modificados, existirem atributos temporais. A notação que representa um fluxo de alteração constitui-se de uma linha bidirecional, isto é, com setas em ambas as extremidades.

#### Fluxo de alteração de atributos de entidade transitória

Especifica a alteração de valores de atributos de entidades transitórias. Podem ser alterados tanto atributos temporais como também atributos intemporais. No tocante aos atributos temporais, a alteração restringe-se aos pontos do tempo indicados na anotação associada ao fluxo. Como foi explicado acima, o fluxo de alteração modela apenas modificação de valores de atributos, sendo assim não cria e nem elimina entidades, como também não amplia e nem reduz a dimensão temporal destas.

#### Fluxo de alteração de atributos de entidade perenes

Especifica a alteração de valores de atributos de entidades perenes. Podem ser alterados tanto atributos temporais como também atributos intemporais. No tocante aos atributos temporais, a alteração restringe-se aos pontos do tempo indicados na anotação associada ao fluxo. O fluxo de alteração de entidades perenes apresenta a mesma funcionalidade do fluxo de alteração de entidades transitórias, ou seja, modela apenas a modificação de valores de atributos, não criando e nem eliminando entidades.

#### Fluxo de alteração de atributos de relacionamento temporal

Especifica a alteração de valores de atributos de relacionamentos do tipo temporal, não criando e nem eliminando relacionamentos. Como os atributos deste tipo de relacionamento são, por definição, necessariamente temporais, a anotação associada ao fluxo deve sempre indicar o conjunto de pontos de tempo que define a abrangência temporal da alteração.

#### Fluxo de alteração de atributos de relacionamento intemporal

Especifica a alteração de valores de atributos de relacionamentos do tipo intemporal, não criando e nem eliminando relacionamentos. Como os atributos deste tipo de relacionamento são, por definição, necessariamente intemporais, a anotação junto ao fluxo não pode indicar qualquer referência à dimensão temporal.

### **5.2.5 Fluxos de verificação de presença**

A finalidade deste tipo de fluxo é verificar se os elementos especificados na anotação junto a ele estão presentes no banco de dados. É representado por uma linha

contendo na extremidade conectada ao conjunto-transação duas setas em sentidos opostos. É importante não confundir esta notação com a dos fluxos de alteração.

Os fluxos de verificação de presença, bem como os de verificação de ausência, que serão descritos mais abaixo, são baseados no ramo restaurador da rede de Petri CEM descrito em [HEU92].

*Fluxo de teste de presença de entidade transitória / entidade perene*

Testa se a entidade ou entidades especificadas na anotação associada ao fluxo estão presentes no conjunto-entidade conectado. Não leva em conta a dimensão temporal, isto é, verifica apenas a presença dos OID's no banco de dados. Opcionalmente, a anotação deste tipo de fluxo pode mencionar um certo período de tempo. Neste caso, esta informação serve apenas para restringir a verificação do conteúdo dos atributos temporais, que porventura estejam citados na anotação junto ao fluxo.

*Fluxo de teste de presença de entidade transitória em um determinado período de tempo*

Testa se a entidade ou entidades transitórias especificadas na anotação associada ao fluxo estão presentes no conjunto-entidade conectado, no período de tempo indicado. Esta restrição temporal é simbolizada pelo pequeno círculo contendo a letra "T", anexo ao conjunto-entidade, do qual parte o fluxo, como mostra a figura 5.4.

*Fluxo de teste de presença de relacionamento temporal*

Verifica se o relacionamento ou relacionamentos temporais especificados na anotação associada ao fluxo estão presentes no conjunto-relacionamento conectado, no período de tempo informado. Por definição, sempre é necessário indicar o período de abrangência, pois tal tipo de relacionamento sempre está associado ao tempo, como é explicado no capítulo anterior.

*Fluxo de teste de presença de relacionamento intemporal*

Verifica se o relacionamento ou relacionamentos intemporais especificados na anotação associada ao fluxo estão presentes no conjunto-relacionamento conectado. Como este tipo de relacionamento, por definição, não está associado ao tempo, a anotação junto ao fluxo não pode conter qualquer referência à dimensão temporal.

## **5.2.6 Fluxos de verificação de ausência**

A finalidade deste tipo de fluxo é verificar se os elementos especificados na anotação junto a ele estão ausentes no banco de dados. É representado por uma linha contendo na extremidade conectada ao conjunto-entidade ou conjunto-relacionamento duas setas em sentidos opostos. Novamente salientando, é importante não confundir esta notação com a dos fluxos de alteração.

*Fluxo de teste de ausência de entidade transitória / entidade perene*

Testa se a entidade ou entidades especificadas na anotação associada ao fluxo estão ausentes do conjunto-entidade conectado. Não leva em conta a dimensão temporal, isto é, verifica apenas a ausência dos OID's no banco de dados.



onde *nome do pacote de dados* deve ser escrito em letras maiúsculas e identifica o conjunto de instâncias “movimentadas” pelo fluxo, a *lista de atributos* identifica as colunas do pacote de dados, o *argumento de pesquisa* define quais atributos servirão de argumentos restritivos no caso de fluxos de alteração e a *restrição temporal* (prefixada pelo símbolo  $\otimes$ ) delimita a abrangência temporal das instâncias do pacote de dados. Mais detalhes sobre o formato das anotações encontram-se na seção 5.3.4, onde a sintaxe da Linguagem de Anotação do modelo TempER-Tr é descrita.

Os pacotes de dados dos fluxos internos podem ser *pacotes unitários*, que são aqueles que contêm exatamente um elemento, ou podem ser *pacotes multi-elementos*, que contêm um número variável de elementos. Estes últimos apresentam a respectiva anotação prefixada por um asterisco (“\*”). A expressão “EMP  $\otimes$  [tinic, »]” que está associada ao fluxo interno de inclusão conectado ao conjunto-entidade *Empregado*, na figura 5.1, é um exemplo de anotação que especifica um pacote unitário ou seja pretende-se incluir um único empregado a cada ocorrência da transação *AdmitirEmpregado*. A restrição temporal existente na anotação especifica que a dimensão temporal de tal empregado inicia em *tinic* e segue em diante.

Para efeito deste trabalho, os pacotes de dados “movimentados” pelos fluxos internos recebem a denominação de *pacotes de dados de fluxo interno*.

Um pacote de dados de fluxo interno, por apresentar um formato tabular, é composto de colunas que são de dois tipos: colunas identificadoras, contendo OID's de entidades, que sempre devem estar presentes, e colunas comuns, contendo valores de atributos. Além disto, caso se trate de pacote de dados com dimensão temporal, também contém, associado a cada uma das suas instâncias, o elemento temporal que determina as respectivas abrangências temporais.

Quanto às colunas referentes a atributos, existem três alternativas de composição:

- 1) Se na anotação do fluxo for omitida a lista de atributos, será assumido que o pacote de dados conterà todos os atributos que se encontram definidos no esquema do dicionário de dados relativo ao conjunto-entidade ou ao conjunto-relacionamento conectado. É o caso do pacote de dados *EMP* da figura 5.5, referente ao fluxo interno de inclusão conectado ao conjunto-entidade *Empregado* da figura 5.1.
- 2) Se na anotação estiver explicitado uma lista de atributos, apenas estes comporão a tabela que forma o pacote de dados.
- 3) Por último, se em vez de uma lista de atributos estiver presente a literal “none”, o pacote de dados não conterà qualquer coluna referente a atributos.

Quanto às colunas referentes a OID's, duas situações podem existir:

- 1) Se o pacote de dados é de fluxo conectado a conjunto-entidade, então conterà apenas uma coluna de OID's e esta coluna levará o nome especial de "OID", como pode ser visto no pacote de dados *EMP* da figura 5.5.
- 2) Por outro lado, se o pacote de dados é de fluxo conectado a conjunto-relacionamento, então conterà uma coluna de OID para cada conjunto-entidade associado. Cada uma destas colunas será identificada pelo nome do conjunto-entidade associado, escrito em letras maiúsculas. É o caso do pacote de dados *LOT* da figura 5.5, que apresenta as colunas *EMPREGADO* e *DEPTO*. Caso se

trate de relacionamento recursivo, há a necessidade de associar o papel junto ao nome destas colunas, para diferenciá-los.

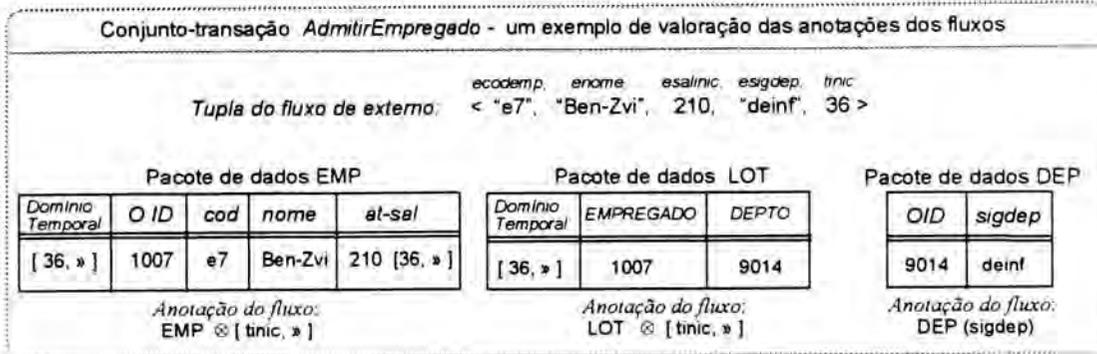


FIGURA 5.5 - Tupla de fluxo externo e pacotes de dados de fluxo interno de uma possível transação *AdmitirEmpregado* da figura 5.1

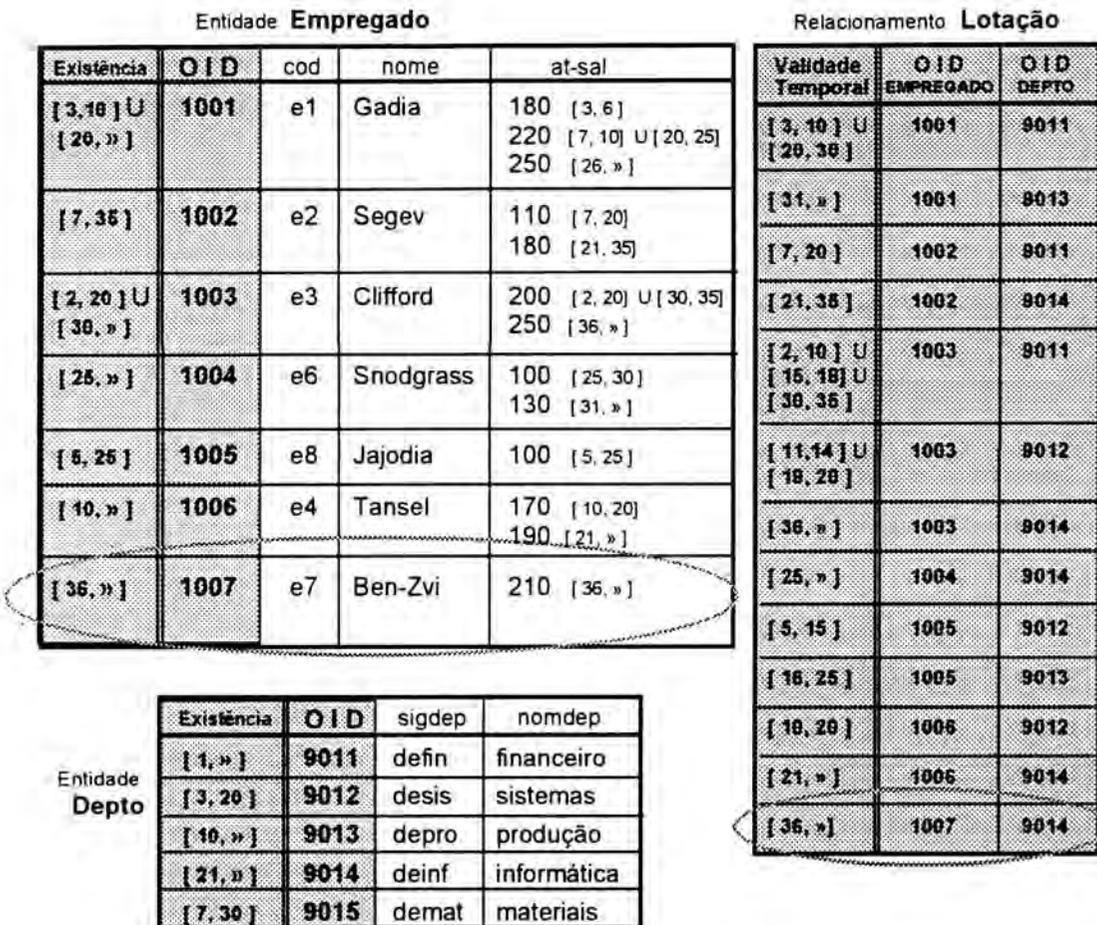


FIGURA 5.6 - Efeito da transação *AdmitirEmpregado*

### 5.3.1.1 Regras adicionais de anotação conforme o tipo de fluxo interno

Para cada tipo de fluxo interno existem algumas regras que devem ser obedecidas na respectiva anotação.

#### Anotação de fluxo de inclusão ou exclusão

A anotação de fluxos de inclusão e de exclusão deve sempre se referir a todos os atributos dos conjuntos-entidade ou conjuntos-relacionamento conectados, isto porque os atributos no modelo TempER-Tr são sempre obrigatórios, não existindo atributos opcionais.

Para denotar que todos os atributos estão sendo referidos, basta omitir a respectiva lista da anotação. É o caso da anotação “EMP ⊗ [tinic, »]”, que está associada a um dos fluxos de inclusão da figura 5.1. Neste exemplo, ao se incluir um empregado no banco de dados, todos os seus atributos também estão sendo incluídos, ou seja os atributos *cod*, *nome* e *at-sal* devem receber um valor, como está exemplificado no pacote de dados *EMP* mostrado na figura 5.5.

#### Anotação de fluxo de manipulação de existência

Quando se tratar dos fluxos que ampliam ou reduzem a existência de entidades, apenas os atributos temporais devem fazer parte dos pacotes de dados “movimentados”. Para denotar isto, também omite-se a lista de atributos da anotação. As figuras 7.3 e 7.4 mostram um exemplo deste tipo de fluxo (trata-se do fluxo que em ambas as figuras está conectado ao pequeno círculo contendo a letra “T”, anexo ao conjunto-entidade *Empregado*).

#### Anotação de fluxo de alteração

A anotação junto a um fluxo de alteração sempre exige que se explicitie os atributos a serem alterados. Internamente cada atributo a ser alterado corresponde a duas colunas no pacote de dados, uma contendo os valores a serem retirados das entidades (ou relacionamentos) e outra contendo os valores a serem incluídos. Para diferenciar estas duas colunas, o nome da que define os valores a serem substituídos recebe o prefixo “old#” e o nome da que define os valores a serem introduzidos recebe o prefixo “new#” (ver exemplo nas figuras 5.7 e 5.8). Além dos atributos a serem modificados, é possível indicar no fluxo de alteração, via a cláusula *arg*, um ou mais atributos que servem de argumentos em fórmulas, no sentido de restringir o conjunto de entidades (ou relacionamentos) que participam da transação.

#### Anotação de fluxo de verificação de presença ou ausência

A anotação junto a um fluxo de verificação de presença ou de ausência deve conter explicitamente os atributos que serão testados nas fórmulas. Ou conter a literal “none”, quando nenhum atributo é necessário ser “consultado”, como é o caso das transações que precisam apenas verificar se o OID está presente ou ausente do banco de dados, ou que precisam testar se determinada entidade ou relacionamento é vigente ou não em determinado período de tempo. As figuras 7.4 e 7.5 modelam transações que utilizam este tipo de fluxo. Trata-se daquele conectado ao conjunto-entidade *Depto*.

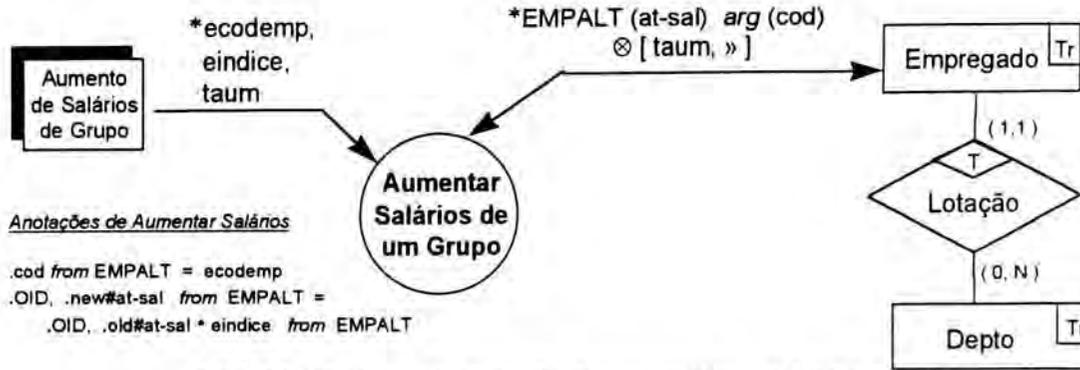


FIGURA 5.7 - Conjunto-transação *AumentarSaláriosDeUmGrupo*

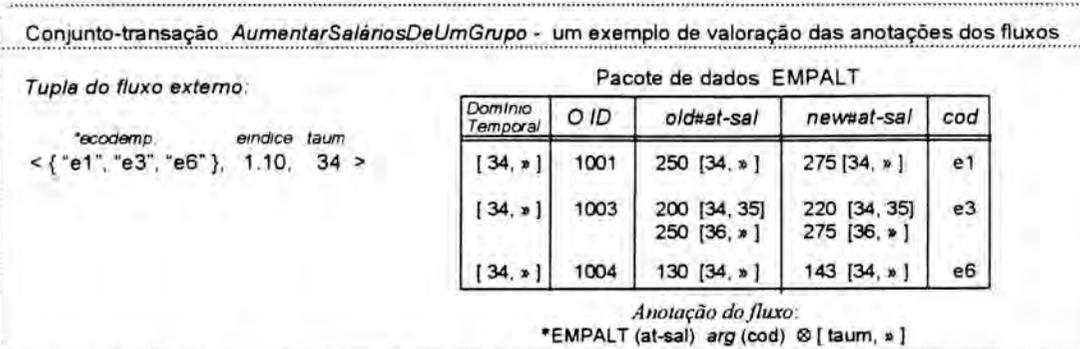


FIGURA 5.8 - Tupla de fluxo externo e pacote de dados de fluxo interno de alteração de uma possível transação *AumentarSaláriosDeUmGrupo*

Entidade **Empregado** (após alteração)

Existência	O ID	cod	nome	at-sal
[ 3, 10 ] U [ 20, * ]	1001	e1	Gadia	180 [ 3, 6 ] 220 [ 7, 10 ] U [ 20, 25 ] 250 [ 26, 33 ] 275 [ 34, * ]
[ 7, 35 ]	1002	e2	Segev	110 [ 7, 20 ] 180 [ 21, 35 ]
[ 2, 20 ] U [ 30, * ]	1003	e3	Clifford	200 [ 2, 20 ] U [ 30, 33 ] 220 [ 34, 35 ] 275 [ 36, * ]
[ 25, * ]	1004	e6	Snodgrass	100 [ 25, 30 ] 130 [ 31, 33 ] 143 [ 34, * ]
[ 5, 25 ]	1005	e8	Jajodia	100 [ 5, 25 ]
[ 10, * ]	1006	e4	Tansel	170 [ 10, 20 ] 190 [ 21, * ]

FIGURA 5.9 - Efeito da transação *AumentarSaláriosDeUmGrupo* sobre a população original do conjunto-entidade *Empregado*, mostrada na figura 4.2

### 5.3.2 Anotações dos fluxos externos de uma transação

As anotações junto aos fluxos externos definem tuplas de dados que entram ou que saem do sistema, ou seja, definem os parâmetros de entrada e saída de uma transação. Cada variável presente nestas anotações, ou refere-se a um dado atômico, ou a um conjunto de dados. Os dados atômicos recebem a denominação de *parâmetros atômicos*; os conjuntos de dados, por sua vez, de *parâmetros multivalorados*. Os parâmetros multivalorados podem ser simples, à semelhança de um vetor (ver *\*ecodemp* da figura 5.7), ou compostos, à semelhança de uma tabela. Neste último caso devem apresentar um esquema descrevendo as colunas que o compõe.

### 5.3.3 Anotações das fórmulas de uma transação

Cada conjunto-transação pode possuir um conjunto de fórmulas cuja finalidade é restringir o conjunto de entidades e/ou relacionamentos que participam de uma determinada transação. Uma fórmula é uma relação entre dois ou mais termos que pode ser verdadeira ou falsa. A figura 5.1 apresenta no quadrante inferior esquerdo o conjunto de fórmulas que devem ser verdadeiras para que uma determinada transação instanciada esteja habilitada a ocorrer. Os termos envolvidos nas fórmulas podem ser de várias categorias, como está descrito abaixo.

#### 5.3.3.1 Escalares

Os escalares são elementos que contém valores atômicos, por exemplo: o código de um empregado, o valor de um salário, o nome de um empregado, um ponto do tempo, um índice, etc. Os atributos intemporais de pacotes unitários de fluxo interno também são escalares, pois são atômicos. Os termos "EMP.nome" e "enome" da segunda fórmula da figura 5.1 são exemplos de escalares.

#### 5.3.3.2 Colunas derivadas

Uma coluna derivada é uma coluna resultante da operação de projeção sobre um pacote de dados de um fluxo interno. Também pode se tratar de um vetor de valores presente em um fluxo externo de entrada. Alguns exemplos de colunas derivadas são: ".cod from EMPALT" e "ecodemp", que se encontram na transação *AumentarSaláriosDeUmGrupo* especificada na figura 5.7.

#### 5.3.3.3 Tabelas derivadas

Uma tabela derivada é o resultado de operações de projeção, de seleção e de união sobre pacotes de dados existentes nos fluxos internos e externos da transação. Os próprios pacotes de dados e os parâmetros multivalorados compostos são, por indução, tabelas derivadas. O modelo TempER-Tr adota a semântica do comando Select da linguagem SQL [DAT 93] como um dos principais meios para produzir colunas e tabelas derivadas. Entretanto, no modelo TempER-Tr omite-se a palavra Select, utilizando-se apenas as cláusulas *from* e *where*. Na figura 5.7 são encontrados dois exemplos de tabelas derivadas (ambas apresentam duas colunas):

".OID, .new#at-sal from EMPALT" e ".OID, old#at-sal\*eindice from EMPALT". Se a tabela derivada possuir apenas uma coluna, pode ser considerada como um termo do tipo coluna derivada, explicado no item anterior.

### 5.3.3.4 Elemento temporal

Um elemento temporal é um conjunto de pontos do tempo. Pode ser representado por uma variável de fluxo externo do tipo multivalorado ou pode ser representado por uma construção que explicita uma união de intervalos de tempo, por exemplo: " $[ta, tb] \cup [tc, td] \cup [te, \gg]$ " ou " $[5, 10] \cup [20, 50]$ ", ou também pode ser representado pela função especial "ELEMTEMP (*assinalamento temporal*)", que obtém (extrai) do assinalamento temporal indicado, o conjunto de todos os pontos do tempo da sua existência.

### 5.3.3.5 Assinalamento temporal

Enquanto que a valoração de um atributo intemporal é um escalar ou um dado atômico, a valoração de um atributo temporal é denominada de assinalamento temporal, expressão que denota a série de valores que este tipo de atributo pode assumir ao longo do tempo. Não se deve confundir esta característica com multivaloração de atributo (não permitida no modelo TempER-Tr), pois em cada momento do tempo um atributo apresenta apenas um único valor. A coluna *at-sal* do conjunto-entidade *Empregado* da figura 5.9 apresenta alguns exemplos de assinalamentos temporais.

Um exemplo graficamente explícito de assinalamento temporal é o termo " $\langle \text{esalinic} [tinic, \gg] \rangle$ " que aparece na terceira fórmula da figura 5.1, o qual especifica que o valor contido na variável *esalinic* vale do momento *tinic* em diante.

Um outro exemplo de assinalamento temporal explícito poderia ser a expressão " $\langle \text{valx} [ta, tb], \text{valy} [tc, td], [\text{valz} [te, \gg]] \rangle$ " que especifica que o valor *valx* vale do momento *ta* até o momento *tb*, que o valor *valy* vale do momento *tc* até o momento *td* e que o valor *valz* vale do momento *te* em diante.

Além da forma explícita, é possível especificar um assinalamento temporal através da citação de atributos temporais ou através de funções que retornam assinalamentos temporais. Voltando à figura 5.1, o termo "EMP.at-sal", que aparece no lado esquerdo da terceira fórmula, é um exemplo de assinalamento temporal especificado através da referência a um atributo temporal de uma entidade. Um exemplo de função que retorna assinalamento temporal encontra-se na figura 5.7, onde a função binária "\*", que multiplica cada posição salarial do assinalamento temporal *a* pela constante 1.10, produz o assinalamento temporal *b*, como está ilustrado na figura 5.10.

O conjunto de todos os pontos do tempo de um assinalamento temporal é chamado de domínio temporal deste assinalamento. Por exemplo o domínio temporal do assinalamento *a* da figura 5.10 é igual a  $\{ 28, 29, 30, 31, 32, 33, 34, 35, 36, \dots \}$  e pode ser representado pelo elemento temporal  $[28, \gg]$  (o símbolo " $\gg$ " significa *em diante*).

Os conceitos de elemento temporal, assinalamento temporal e domínio temporal são também explicados no capítulo 3 desta dissertação.

*Aumento de 10 % do salário*

assin.temp. *a* \* 1,10 = assin.temp. *b*  
 assin.temp. *a* : <100 [28,30], 130 [31, \gg]  
 assin.temp. *b* : <110 [28,30], 143 [31, \gg]

(a)		(b)	
valor	ponto tempo	valor	ponto tempo
100	28	110	28
100	29	110	29
100	30	110	30
130	31	143	31
130	32	143	32
...	...	...	...
...	...	...	...
130	\gg	143	\gg

FIGURA 5.10 - Assinalamento temporal

### 5.3.4 Sintaxe da linguagem de anotação do modelo TempER-Tr

Esta seção apresenta a sintaxe da Linguagem de Anotação do modelo TempER-Tr especificada em BNF.

#### 5.3.4.1 Anotações específicas dos fluxos internos

```

< anotação de fluxo interno > ::= < pacote unitário > | < pacote multi-elementos >
< pacote unitário > ::= < pacote de dados >
< pacote multi-elementos > ::= < * > < pacote de dados >
< pacote de dados > ::= < nome pacote fluxo interno > [ < lista de atributos > ]
                               [ arg.pesquisa ] [ restrição temporal ]
< lista de atributos > ::= < ( > < atributo > [ { < , > < atributo > } ... ] < ) >
                               | < ( none ) >
< atributo > ::= // Nome de atributo da entidade ou relacionamento conforme esquema
                               do dicionário de dados. //
< atributo > ::= < atributo temporal > | < atributo intemporal >
< arg. pesquisa > ::= < lista de atributos >
< restrição temporal > ::= < ⊗ > < elemento temporal > | [ at SomeTimePoint ]
< nome pacote fluxo interno > ::= // Seqüência de letras MAIÚSCULAS e números,
                               o primeiro caracter deve ser letra, pode conter "-" e "_". //

```

#### 5.3.4.2 Anotações específicas dos fluxos externos

```

< anotação de fluxo externo > ::= < parâmetro > [ { < , > < parâmetro > } ... ]
< parâmetro > ::= < parâmetro atômico > | < * > < parâmetro multivalorado >
< parâmetro atômico > ::= < variável >
< parâmetro multivalorado > ::= < parâmetro multivalorado simples >
                               | < parâmetro multivalorado com esquema >
< parâmetro multivalorado simples > ::= < nome param. multiv. >
< parâmetro multivalorado com esquema > ::= < nome param. multiv. > < esquema >
< nome param. multiv. > ::= < variável >
< esquema > ::= < ( > < variável > [ { < , > < variável > } ... ] < ) >

```

#### 5.3.4.3 Anotações específicas das fórmulas das transações

```

< conjunto de fórmulas > ::= [ < fórmula > [ { < fórmula > } ... ] ]

```

/\* entre as fórmulas está implícito o conectivo *and*. \*/

```

< fórmula > ::= TRUE | FALSE
               | < termo > < simb. de relação > < termo >
               | < fórmula > < conectivo and > < fórmula >
               | < fórmula > < conectivo or > < fórmula >
               | not < fórmula >

```

/\* O conectivo *and* prevalece sobre o conectivo *or*, portanto deve-se usar parênteses quando se desejar que o *or* prevaleça. \*/

```

< termo > ::=
    < escalar >
    | < coluna derivada >
    | < tabela derivada >
    | < elemento temporal >
    | < assinalamento temporal >

< escalar > ::= < parâmetro atômico >
    | < nome pacote fluxo interno > < . > < atributo intemporal >
    | < constante >
    | < simb.função escalar > [ < ( > < termo > [ { < , > < termo > } ... ] < ) > ]
    | < termo > < simb.função escalar binária > < termo >

< coluna derivada > ::= < parâmetro multivalorado simples >
    | < coluna resultante de select >

< coluna resultante de select > ::= < coluna fluxo > from < tab.fluxo >
    [ where < condição > ] /* igual ao Select do SQL */

< tab.fluxo > ::= { < nome pacote fluxo interno > | < nome param.multiv. > }

< coluna fluxo > ::= [ nome pacote fluxo interno ] < . > < atributo >
    | [ nome param.multiv. ] < . > < variável >

< tabela derivada > ::= < nome pacote fluxo interno >
    | < nome param. multiv. >
    | < tabela resultante de select >

< tabela resultante de select > ::= < coluna fluxo > [ { < , > < coluna fluxo > } ... ]
    from < tab.fluxo > [ { < , > < tab.fluxo > } ... ]
    [ where < condição > ] /* igual ao Select do SQL */

< assinalamento temporal > ::= < assin. temporal explícito >
    | < nome pacote fluxo interno > < . > < atributo temporal >
    | < simb.função assin.temp. > [ < ( > < termo > [ { < , > < termo > } ... ] < ) > ]
    | < termo > < simb.função assin.temp. binária > < termo >

< assin. temporal explícito ::= < < > < escalar > < intervalo de tempo >
    [ { < , > < escalar > < intervalo de tempo > } ... ] < > >

```

#### 5.3.4.4 Elemento temporal

```

< elemento temporal > ::= < intervalo de tempo >
    | < união de intervalos >
    | < conjunto de pontos do tempo >
    | < ponto do tempo >
    | < ELEMTEMP ( > < assinalamento temporal > < ) >

< intervalo de tempo > ::= < intervalo aberto inicio >
    | < intervalo fechado >
    | < intervalo aberto fim >
    | < intervalo total eixo tempo >

< intervalo aberto inicio > ::= < [ > < « > < , > < ponto do tempo > < ] >

< intervalo fechado > ::= < [ > < ponto do tempo > < , > < ponto do tempo > < ] >

< intervalo aberto fim > ::= < [ > < ponto do tempo > < , > < » > < ] >

< intervalo total eixo tempo > ::= < [ « , » ] >

< ponto do tempo > ::= < variável > | < número natural >

```

```

< união de intervalos > ::= { < intervalo fechado > | < intervalo aberto inicio > }
                             [ { < U > < intervalo fechado > } ... ] < U >
                             { < intervalo fechado > | < intervalo aberto fim > }

< conjunto de pontos do tempo > ::= < nome de variável >

```

#### 5.3.4.5 Variável e constante

```

< variável > ::= // Seqüência de letras minúsculas e números, o primeiro caracter deve
                  ser letra, pode conter os caracteres especiais "-" e "_". //

< constante > ::= // Valor pertencente a um dos domínios de constantes do
                   Universo do Discurso. //

< número natural > ::= // Constante pertencente ao domínio dos Números Naturais. //

```

## 5.4 Comportamento do modelo TempER-Tr

O comportamento do modelo TempER-Tr é semelhante ao de uma rede de Petri de alto nível do tipo condição/evento, ou seja, existem regras que condicionam a habilitação de uma transação e regras que definem o efeito de uma transação.

Os eventos que acionam as transações, representados pelos conjuntos-evento, estão sempre habilitados por definição. Cada vez que ocorrem, uma nova tupla de dados é colocada no fluxo externo que conecta os respectivos conjuntos-evento aos conjuntos-transação.

Para entender o comportamento de uma transação, primeiro é preciso estabelecer o que sejam objetos consumíveis e objetos produzíveis desta transação.

### 5.4.1 Objetos consumíveis de uma transação

Os *objetos consumíveis* de uma transação são aqueles especificados nos fluxos internos que partem dos conjuntos-entidade e conjuntos-relacionamento e apontam para o círculo que representa o conjunto-transação, a saber:

- As entidades contidas nos pacotes de fluxo interno de exclusão de entidades (sejam elas transitórias ou perenes), juntamente com todos valores de seus atributos e com todos os pontos do tempo especificados na restrição temporal do fluxo.
- As “posições” que as entidades ocupam no eixo temporal, especificadas na restrição temporal junto aos fluxos de redução de existência de entidades transitórias (ver figura 5.3), ou seja, o conjunto de pontos do tempo que se quer retirar da existência de tais entidades. Neste tipo de fluxo, as entidades propriamente ditas não são objetos consumíveis da transação, pois os seus OID's permanecerão armazenados no banco de dados.
- Os assinalamentos temporais (ver subseção 5.3.3.5) referentes a todos os atributos temporais das entidades especificadas em fluxos de redução de existência de entidades transitórias. Estes assinalamentos temporais estão restritos ao conjunto de pontos do tempo indicados na anotação junto aos fluxos.

- Os relacionamentos contidos nos pacotes de dados de fluxos internos de exclusão conectados a conjuntos-relacionamento, bem como os valores de todos os seus atributos. Caso se trate de relacionamentos temporais, também são objetos consumíveis as “posições” que estes ocupam no eixo temporal, especificadas na restrição temporal do fluxo. Neste caso, a valoração a excluir dos atributos, que pelas regras do modelo TempER-Tr também são temporais, restringem-se ao período de tempo definido no referido fluxo.
- Os valores de atributos que serão substituídos (aqueles cujo nome leva o prefixo *old#*) especificados nos fluxos internos de alteração. Se tais atributos forem temporais, a valoração que está sendo excluída é um assinalamento temporal, restrito ao período de tempo especificado no fluxo.
- As entidades e relacionamentos indicadas em fluxos de verificação de presença. Quando se tratar de entidades transitórias ou relacionamentos temporais, também são objetos consumíveis da transação os pontos do tempo, especificados na restrição temporal do fluxo.
- Os valores de atributos especificados nos fluxos internos de verificação de presença. Se tais atributos forem temporais, a valoração em questão é um assinalamento temporal, restrito ao período de tempo especificado no fluxo.

#### 5.4.2 Objetos produzíveis de uma transação

Os *objetos produzíveis* por uma transação são aqueles especificados nos fluxos que se originam - que partem - do círculo que representa tal transação e apontam para os conjuntos-entidade e conjuntos-relacionamento, a saber:

- As entidades contidas nos pacotes de fluxo interno de inclusão de entidades (sejam elas transitórias ou perenes), juntamente com todos os valores de seus atributos e com todos os pontos do tempo especificados na restrição temporal do fluxo.
- As “posições” que as entidades ocupam no eixo temporal, especificadas na restrição temporal junto aos fluxos de ampliação de existência de entidades transitórias (ver figura 5.3), ou seja, o conjunto de pontos do tempo que se quer adicionar à existência de tais entidades. Neste tipo de fluxo, as entidades propriamente ditas não são objetos produzíveis da transação pois os seus OID's já estão armazenados no banco de dados.
- Os assinalamentos temporais referentes a todos os atributos temporais das entidades especificadas em fluxos de ampliação de existência de entidades transitórias. Estes assinalamentos temporais estão restritos ao conjunto de pontos do tempo indicados na anotação junto aos fluxos.
- Os relacionamentos contidos nos pacotes de dados de fluxos internos de inclusão conectados a conjuntos-relacionamento, bem como os valores de todos os seus atributos. Caso se trate de relacionamentos temporais, também são objetos produzíveis as “posições” que estes ocupam no eixo temporal, especificadas na restrição temporal do fluxo. Neste caso, a valoração a incluir dos atributos, que pelas regras do modelo TempER-Tr também são temporais, restringem-se ao período de tempo definido no referido fluxo.

- Os novos valores de atributos (aqueles cujo nome leva o prefixo *new=*) especificados nos fluxos internos de alteração. Se tais atributos forem temporais, a valoração que está sendo incluída é um assinalamento temporal, restrito ao período de tempo especificado no fluxo.
- As entidades e relacionamentos indicadas em fluxos de verificação de ausência. Quando se tratar de entidades transitórias ou relacionamentos temporais, também são objetos produzíveis da transação os pontos do tempo, especificados na restrição temporal do fluxo.
- Os valores de atributos especificados nos fluxos internos de verificação de ausência. Se tais atributos forem temporais, a valoração em questão é um assinalamento temporal, restrito ao período de tempo especificado no fluxo.

### 5.4.3 Habilitação de uma transação

Uma transação (uma determinada instância de um conjunto-transação) está habilitada a ocorrer se as condições que a restringem estiverem satisfeitas, ou seja:

- as tuplas de valores de entrada devem estar presentes em todos os fluxos externos de entrada conectados ao conjunto-transação;
- todos os *objetos consumíveis* estão *presentes* nos respectivos conjuntos de entidades e relacionamentos;
- todos os *objetos produzíveis* estão *ausentes* dos respectivos conjuntos de entidades e relacionamentos;
- o estado que o banco de dados vier a alcançar como resultado da ocorrência da transação não viola as restrições estáticas de integridade e outras que tiverem sido definidas ao nível do modelo de dados temporal.

### 5.4.4 A ocorrência de uma transação

A ocorrência ou efetivação de uma transação, que no modelo TempER-Tr é chamada de *transformação*, provoca os seguintes efeitos no banco de dados:

- as tuplas com os valores de entrada *desaparecem* dos fluxos externos de entrada;
- as tuplas com os valores de saída *aparecem* nos fluxos externos de saída;
- os objetos consumíveis referentes aos fluxos internos de exclusão *desaparecem* dos respectivos conjuntos de entidades ou relacionamentos;
- os objetos consumíveis referentes aos fluxos de redução de existência de entidade transitória *desaparecem* dos respectivos conjuntos-entidade;
- os objetos produzíveis referentes aos fluxos de inclusão *aparecem* nos respectivos conjuntos de entidades e relacionamentos;
- os objetos produzíveis referentes aos fluxos de ampliação de existência de entidade transitória *aparecem* dos respectivos conjuntos-entidade;

- os atributos especificados nos fluxos de alteração são modificados, ou seja os seus valores anteriores (#old) *desaparecem* e os novos valores (#new) *aparecem*.

Todas estas alterações ocorrem coincidentemente, isto é, uma transação é uma transição atômica de um estado do sistema para outro: nenhum estado intermediário é observado.

É importante lembrar que os fluxos de verificação de presença e de verificação de ausência não provocam qualquer modificação no banco de dados, isto é, os objetos consumíveis e produzíveis, por eles referidos, apenas são “lidos” do banco de dados.

#### 5.4.5 Exemplificando o comportamento de uma transação

Para ilustrar o comportamento de uma transação, tome-se como exemplo o conjunto-transação *AumentarSaláriosDeUmGrupo* modelado na figura 5.7 e que tem o seguinte enunciado:

Alterar o salário a partir do ponto do tempo *taum* em diante, de acordo com o índice *eindice*, de todos os empregados cujo código está informado no parâmetro multivalorado *ecodemp*. Apenas os empregados que estão vigentes de *taum* em diante participam da transação.

Uma possível transação do conjunto-transação *AumentarSalários* seria representada pelos dados mostrados na figura 5.8. Esta transação estaria habilitada, em relação à tupla existente no fluxo externo de entrada, se o pacote de dados *EMPALT* apresentasse a valoração que está indicada na mesma figura.

A ocorrência de tal transação provocaria a modificação do assinalamento temporal do atributo *at-sal* das entidades de OID igual a *1001*, *1003* e *1004* como é mostrado na figura 5.9.

#### 5.4.6 Conflito e concorrência

Duas transações que apresentem algum objeto consumível, ou algum objeto produzível, em comum, são ditas *em conflito*. Supondo que houvesse uma transação que transferisse um empregado *x* do departamento *a* para o departamento *b* e outra transação que transferisse o mesmo empregado *x* do departamento *a* para o departamento *c*, as duas estariam em conflito, pois disputariam uma mesma entidade. O modelo TempER-Tr apenas aponta o conflito, não estabelece qual transferência será efetivada. Isto se configura em uma situação de *não-determinismo*, que é desejável ao nível da modelagem conceitual de um sistema.

Duas transações que não possuam objetos consumíveis e objetos produzíveis em comum são ditas *concorrentes* e são independentes entre si, podendo ocorrer em paralelo. Seria o caso de duas transações *AdmitirEmpregado* referindo-se a pessoas diferentes. A admissão de um empregado não afetaria e não seria afetada pela admissão de outro. O modelo TempER-Tr não determina qualquer seqüência de efetivação de transações concorrentes, o que também é uma propriedade desejável ao nível da modelagem conceitual, pois a seqüencialização é uma questão a ser tratada na posterior implementação do sistema.

## 5.5 Conclusão

O modelo TempER-Tr, uma evolução do modelo ER-Tr [PER 90, HEU 93], à semelhança deste, também cumpre com os requisitos para modelagem conceitual definidos em [GRI82], pois além de permitir representar tanto as propriedades estáticas como as propriedades dinâmicas de um sistema, evita que se contamine o modelo com detalhes adicionais de implementação, além daqueles contidos exclusivamente nos requerimentos dos usuários.

Em relação ao modelo ER-Tr, a presente proposta traz duas contribuições:

- Utiliza para modelar as propriedades estáticas uma extensão temporal do modelo Entidade-Relacionamento.
- Propõe, com inspiração em [HEU 9?], uma evolução da linguagem de anotação para as transações, onde os pacotes de dados dos fluxos das transações passam a ser tratados como sendo tabelas, possibilitando assim uma melhor manipulação de conjuntos de entidades, com base no comando Select da linguagem SQL.

## 6 Mapeamento do modelo TempER-Tr para rede de Petri

O modelo TempER-Tr, que em um mesmo diagrama combina a especificação das propriedades estáticas e dinâmicas de uma transação, tem a sua semântica explicada formalmente através da rede de Petri de alto nível do tipo CEM [HEU 90]. No modelo TempER-Tr adota-se, portanto, o mesmo princípio seguido na formulação do modelo ER-Tr [PER 90, HEU 91].

Como especificado em [PER 90], “a representação de propriedades estáticas dentro da abordagem de redes é feita através da negação da habilitação das alterações, não se expressando diretamente o fato desejado, mas um outro, de cuja negação se conclui aquele que se queria representar”. É através das *conexões mortas*, um pequeno quadrado contendo uma letra F estilizada, que estas propriedades estáticas são especificadas (ver *cm1* na figura 6.1).

O objetivo principal do mapeamento do modelo TempER-Tr para rede de Petri é o de explicar formalmente as suas restrições estáticas e as suas propriedades dinâmicas, que nos dois capítulos anteriores foram apresentados de maneira informal. O objetivo secundário, ainda a ser melhor explorado no futuro, é o de aproveitar a possibilidade de simulação inerente às redes de Petri.

Primeiramente, na seção 6.1 deste capítulo, será demonstrado como o modelo de dados TempER pode ser traduzido para uma rede de Petri de alto nível do tipo CEM, conforme as regras estabelecidas pela abordagem ER/PN [HEU 93]. Em seguida, na seção 6.2, será apresentada a forma como o modelo TempER-Tr (o modelo dinâmico combinado com o modelo entidade-relacionamento temporal) pode ser completamente mapeado para uma rede de Petri CEM, através da combinação de *conexões mortas* com *conexões vivas*. Para um melhor entendimento dos mapeamentos que serão mostrados a seguir, deve ser consultado o capítulo 2 desta dissertação, que resume os conceitos e o funcionamento da rede de Petri CEM e da abordagem ER/PN.

### 6.1 Mapeamento do modelo de dados temporal para rede de Petri

Nesta seção, é mostrado como as construções do modelo de dados TempER apresentadas no capítulo 4 são traduzidas para rede de Petri.

#### 6.1.1 Mapeamento de conjuntos-entidade

Cada conjunto-entidade é mapeado para dois lugares na rede de Petri, um representando a sua perspectiva temporal e o outro representando a sua perspectiva intemporal. Os lugares na rede de Petri são simbolizados por círculos.

O domínio *OID* representa o conjunto de todas as entidades do Universo do Discurso. Quando uma entidade destas é introduzida no sistema, um destes *OID's* é ocupado. Um conjunto-entidade especial, que por ser de conteúdo constante não é representado por um lugar na rede de Petri, mas faz parte da Linguagem de Anotação, é o conjunto de todos os pontos do eixo temporal, simbolizado pela letra T.

### 6.1.1.1 Mapeamento de conjunto de entidades transitórias

Um conjunto de entidades transitórias é traduzido para dois lugares, um especificando os OID's pura e simplesmente, e o outro especificando as posições que os OID's ocupam na dimensão temporal. A figura 6.1 mostra como se dá este mapeamento.

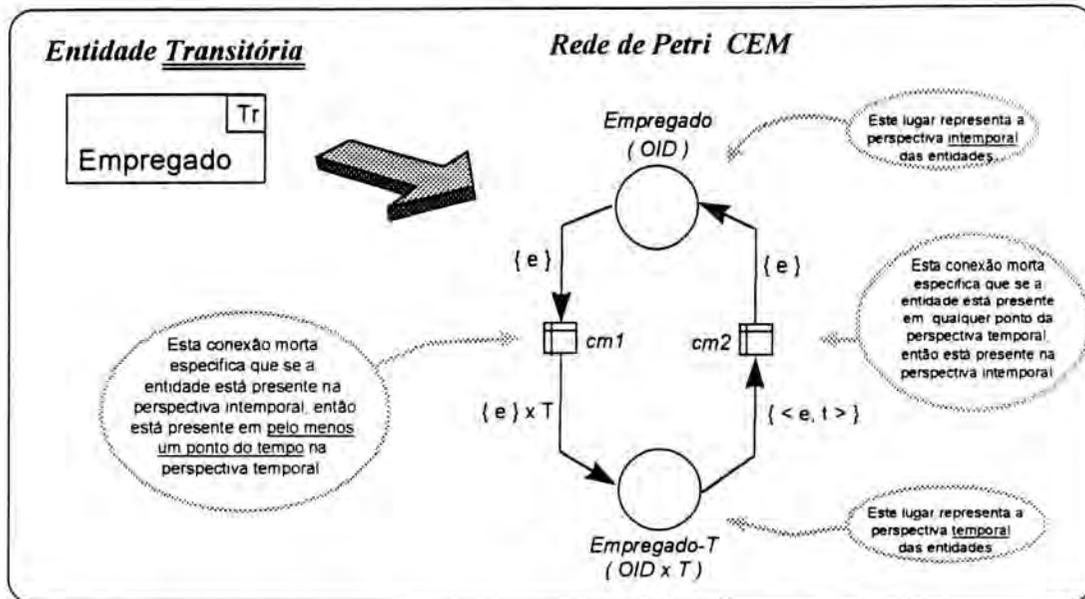


FIGURA 6.1 - Mapeamento de entidade transitória para rede de Petri

Como por definição as alterações definidas pelas conexões mortas nunca estão habilitadas, isto permite que se lance mão delas para modelar restrições estáticas. Por exemplo, as alterações definidas pela conexão morta *cm1* da figura acima para nunca estarem habilitadas, exigem que, para cada entidade presente no lugar *Empregado*, haja no lugar *Empregado-T* pelo menos um par  $\langle e, t \rangle$ , onde  $e$  refere-se ao OID da entidade e  $t$  refere-se a um ponto do tempo qualquer. Em relação à conexão morta *cm2*, para que as suas alterações jamais estejam habilitadas, é necessário que, para cada par  $\langle e, t \rangle$  presente no lugar *Empregado-T*, o OID representado pela variável  $e$  esteja presente no lugar *Empregado*. Esta segunda conexão morta estabelece que se uma entidade está presente na perspectiva temporal, então também está presente na perspectiva intemporal.

### 6.1.1.2 Mapeamento de conjunto de entidades perenes

Um conjunto de entidades perenes é traduzido para dois lugares, um especificando os OID's pura e simplesmente, e o outro especificando as posições que os OID's ocupam na dimensão temporal. Entretanto, diferentemente do que ocorre nas entidades transitórias, uma entidade perene existe em todos os pontos do eixo temporal. Esta propriedade é especificada pela conexão morta *cm3* da figura 6.2, cujas alterações para nunca estarem habilitadas exigem que, para cada entidade  $f$  que esteja presente no lugar *Função*, haja no lugar *Função-T* tantos pares  $\langle f, t \rangle$ , onde  $t$  é um ponto do tempo, quantos forem os momentos do eixo temporal  $T$ .

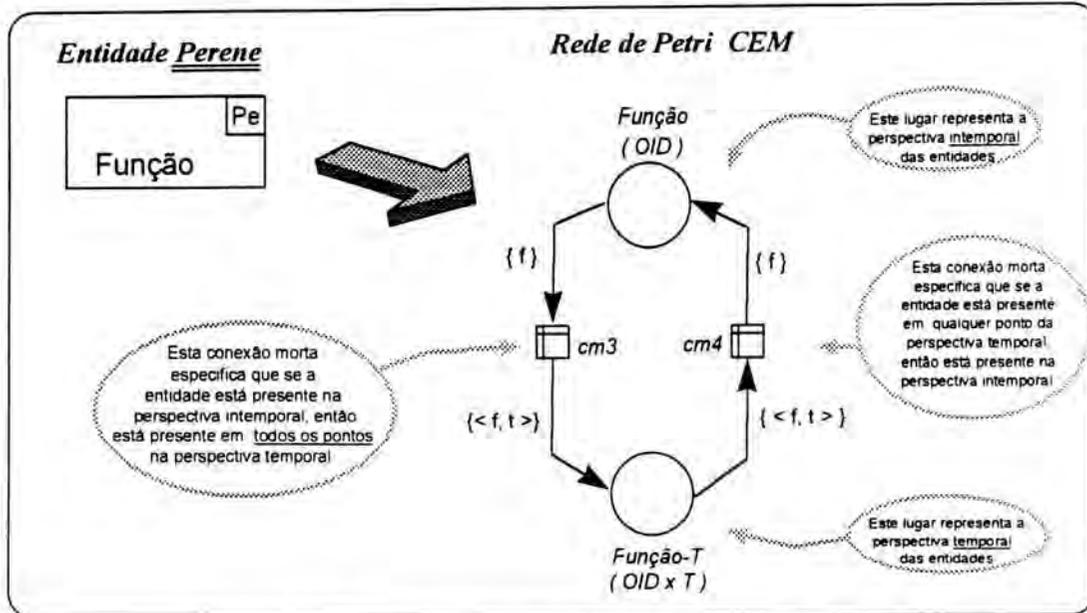


FIGURA 6.2 - Mapeamento de entidade perene para rede de Petri

Supondo, a título de exemplo, que o domínio  $T$  fosse igual a  $\{1, 2, 3, 4\}$ , isto é, contivesse hipoteticamente apenas quatro pontos do tempo, e supondo que o domínio  $OID$  fosse igual a  $\{1001, 1002\}$  e que estas duas entidades estivessem presentes no lugar *Função*. Tendo em vista estes domínios, são possíveis oito alterações em relação à conexão *cm3*; ou seja as variáveis  $f$  e  $t$  podem apresentar as seguintes combinações de valores:

$$\{ \langle 1001, 1 \rangle, \langle 1001, 2 \rangle, \langle 1001, 3 \rangle, \langle 1001, 4 \rangle, \langle 1002, 1 \rangle, \langle 1002, 2 \rangle, \langle 1002, 3 \rangle, \langle 1002, 4 \rangle \}.$$

Para que todas estas alterações *não* estejam habilitadas, o lugar *Função-T* deve necessariamente apresentar todos os pares acima. De forma alguma por exemplo o par  $\langle 1001, 3 \rangle$  deixaria de estar presente neste lugar, pois se assim fosse, uma das alterações de *cm3*, aquela cujas variáveis  $f$  e  $t$  são respectivamente iguais a 1001 e 3 se tornaria habilitada, fato que contrariaria a natureza da conexão morta.

## 6.1.2 Mapeamento de relacionamentos

Cada conjunto-relacionamento do modelo TempER, seja ele temporal ou intemporal, é mapeado para exatamente um lugar na rede de Petri, pois cada relacionamento se dá, ou na perspectiva temporal das entidades, ou na perspectiva intemporal, nunca nas duas ao mesmo tempo, como foi explicado no capítulo que descreve o modelo de dados temporal. Nesse mapeamento, as anotações de cardinalidade, que restringem a participação de entidades nos conjuntos-relacionamento, são especificadas através de conexões mortas.

### 6.1.2.1 Mapeamento de relacionamentos temporais

Cada relacionamento (instância) temporal é representado por tantas tuplas quantos forem os pontos do tempo da sua validade temporal. Cada uma destas tuplas é, portanto, composta pelas entidades associadas e pelo ponto do tempo. Por exemplo, supondo que no conjunto-relacionamento *Lotação* do modelo de dados da figura 6.3 exista a instância  $\langle 1002, 9011 \rangle$ , onde o primeiro elemento é o OID de um empregado e o

segundo elemento é o OID de um departamento, e supondo que a validade temporal deste relacionamento seja  $[7, 11]$ , isto corresponderá à presença das tuplas  $\langle 1002, 9011, 7 \rangle$ ,  $\langle 1002, 9011, 8 \rangle$ ,  $\langle 1002, 9011, 9 \rangle$ ,  $\langle 1002, 9011, 10 \rangle$  e  $\langle 1002, 9011, 11 \rangle$  no lugar que representa tal conjunto-relacionamento na rede de Petri.

Um conjunto-relacionamento temporal é representado na rede de Petri por um lugar cujo domínio é o produto cartesiano de tantos domínios OID quantos forem os conjuntos-entidade associados, juntamente com o conjunto de pontos do eixo temporal (o conjunto  $T$ ). É o caso do domínio do lugar *Lotação* da rede da figura 6.3 que é igual a  $\langle \text{OID} \times \text{OID} \times T \rangle$ . Por se referir a conjunto-relacionamento temporal, as conexões mortas que especificam as restrições de cardinalidade devem ligar este lugar àqueles que representam a perspectiva temporal das entidades (*Empregado-T* e *Depto-T*, no exemplo em questão).

Uma entidade que esteja participando de um relacionamento, obviamente, deve estar presente no conjunto-entidade respectivo, em todos os momentos do tempo da validade temporal deste relacionamento. Esta restrição é especificada na rede de Petri por uma conexão morta semelhante às conexões *cm9* e *cm12* da figura 6.3. A referência a pontos do tempo nessas conexões é representada pela variável  $t$  presente nas anotações dos seus ramos.

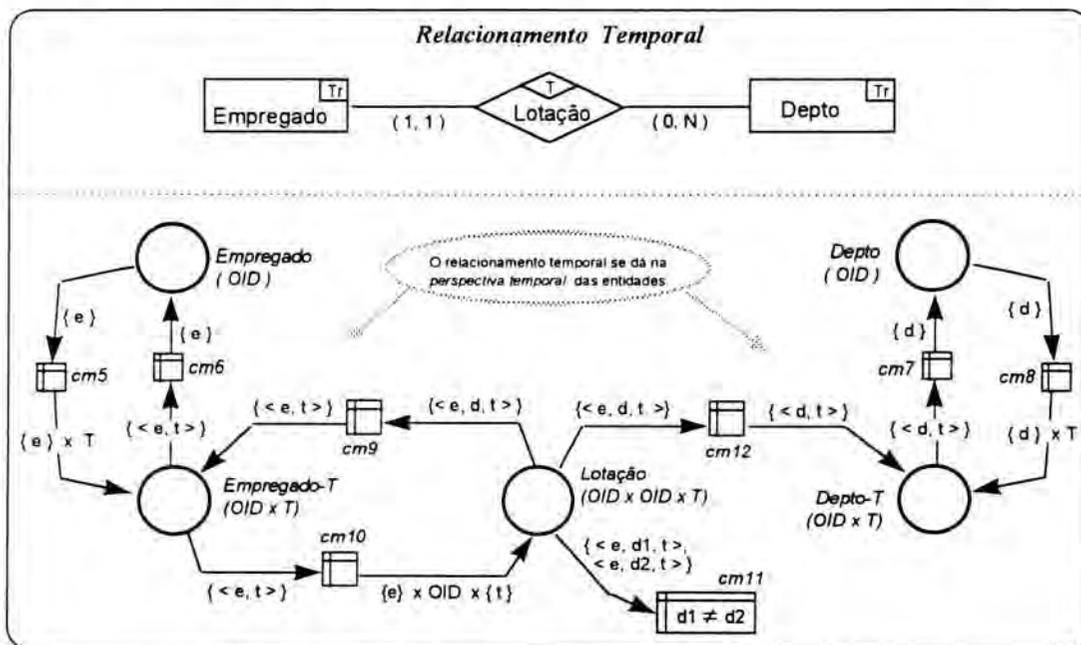


FIGURA 6.3 - Mapeamento de relacionamento temporal para rede de Petri

As restrições de cardinalidade do modelo TempER apresentam o formato  $(X, Y)$ , onde  $X$  estabelece a *participação mínima* das entidades no conjunto-relacionamento e  $Y$  determina a *participação máxima*. Se  $X$  for igual a 1, a participação das entidades no conjunto-relacionamento é obrigatória em todos os momentos do tempo das suas existências, mas se for igual a 0 podem haver pontos da validade temporal das entidades em que estas não participem de relacionamentos. Se  $Y$  é igual a 1, a participação limita-se a uma e a somente uma vez a cada momento do tempo, se igual a  $N$  não existe este limite máximo. Sendo assim, a cardinalidade do tipo  $(0, N)$ , na verdade não estabelece qualquer restrição a respeito da participação das entidades em relacionamentos, não exigindo portanto qualquer conexão-morta adicional além da que foi citada no parágrafo

anterior, é o caso da participação de departamentos no conjunto-relacionamento *Lotação*, conforme mostra o diagrama TempER da figura 6.3. Quanto aos demais tipos de cardinalidade: (0, 1), (1, N) e (1, 1), outras conexões mortas são necessárias.

A obrigatoriedade de participação no conjunto-relacionamento (quando  $X$  é igual a 1), deve ser representada por uma conexão morta semelhante à *cm10* da figura 6.3. Esta conexão morta determina que, em cada momento do tempo da validade temporal de um empregado, este está presente no mínimo uma vez no conjunto-relacionamento, ou seja, um empregado está lotado em no mínimo um departamento a cada momento do tempo da sua existência.

Quanto ao limite máximo de participação, quando este for igual a 1 ( $Y = 1$ ), na rede de Petri que está mapeando o conjunto-relacionamento deve haver uma conexão morta semelhante à *cm11* da figura 6.3. Esta conexão determina que um empregado, em um mesmo momento do tempo, não pode estar lotado em mais de um departamento.

As figuras 6.5 e 6.6 apresentam o mapeamento para rede de Petri de todas as restrições de cardinalidade previstas no modelo TempER relativas a conjuntos-relacionamento temporais. As redes que aparecem nestas figuras são parciais, pois mostram apenas o mapeamento relativo ao conjunto-entidade *EntidadeA*. A mesma sistemática de mapeamento é aplicável no que tange aos demais conjuntos-entidade associados ( $B, C, \dots, Z$ ), obedecidas as respectivas restrições de cardinalidade.

### 6.1.2.2 Mapeamento de relacionamentos intemporais

Cada relacionamento intemporal (instância) é representado por uma tupla composta das entidades associadas. Um conjunto-relacionamento intemporal é representado na rede de Petri por um lugar cujo domínio é o produto cartesiano de tantos domínios OID quantos forem os conjuntos-entidade associados (aqui o domínio  $T_1$  referente ao eixo temporal, não participa do produto cartesiano). É o caso do domínio  $\langle \text{OID} \times \text{OID} \rangle$  referente ao lugar *Preferência* presente na rede de Petri da figura 6.4. Por se referir a conjunto-relacionamento intemporal, as conexões mortas que especificam as restrições de cardinalidade devem ligar este lugar àqueles que representam a perspectiva intemporal das entidades (*Empregado e Função*).

Uma entidade que esteja participando de um relacionamento intemporal obviamente deve estar presente no conjunto-entidade respectivo, na sua perspectiva intemporal. Esta restrição é especificada na rede de Petri por uma conexão morta semelhante às conexões *cm17* e *cm19*. Como pode ser notado, estas conexões não fazem qualquer referência à dimensão temporal, apenas os OID's das entidades estão sendo representados pelas variáveis junto aos seus ramos.

Quanto às restrições de cardinalidade, quando se tratar de relacionamentos intemporais, as regras são análogas às dos relacionamentos temporais. A diferença é que agora não existem referências a momentos do tempo.

Como já foi visto anteriormente, as restrições apresentam o formato  $(X, Y)$ , onde  $X$  estabelece a participação mínima e  $Y$  define a participação máxima. No caso dos relacionamentos intemporais, se  $X$  é igual a 1, a participação das entidades é obrigatória no conjunto-relacionamento, entretanto se for igual a 0, é permitido existirem entidades que não estejam participando de relacionamentos. Em relação a  $Y$ , se este for igual a 1, a participação das entidades no conjunto-relacionamento limita-se a no máximo uma vez, se for igual a N não existe limite máximo. Desta forma, a cardinalidade do tipo (0, N),

não se caracteriza exatamente como uma restrição, pois permite qualquer grau de participação das entidades no conjunto-relacionamento, não exigindo assim qualquer conexão morta adicional. É o caso da participação das instâncias do conjunto-entidade *Função* no conjunto-relacionamento *Preferência* mostrado no modelo TempER da figura 6.4. Quanto aos demais tipos de cardinalidade: (0, 1), (1, N) e (1, 1), outras conexões mortas são necessárias.

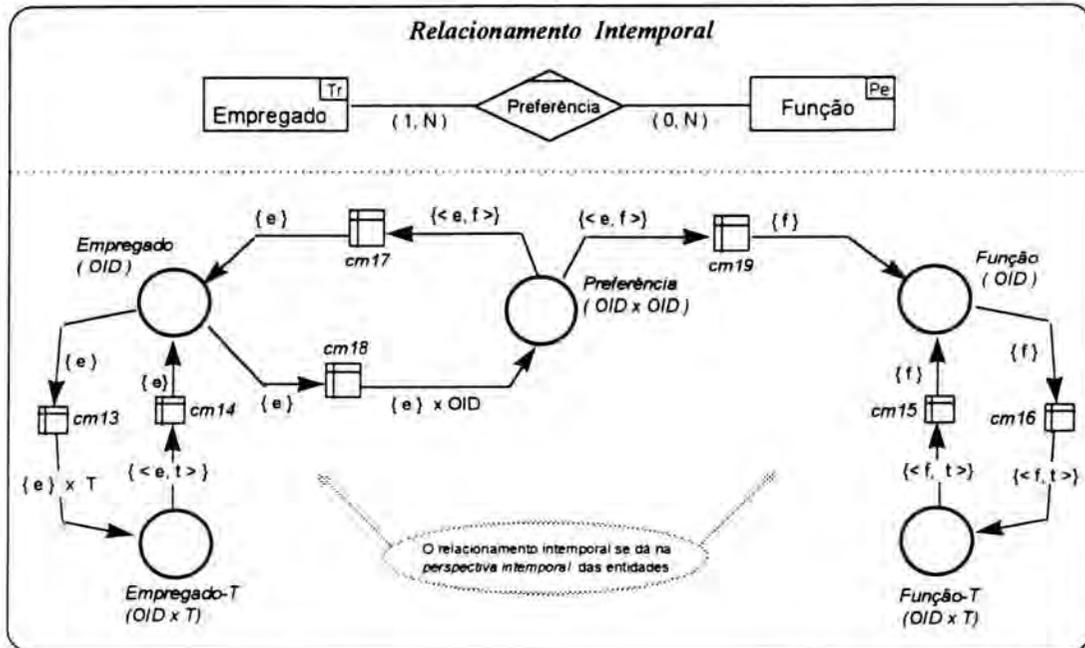


FIGURA 6.4 - Mapeamento de relacionamento intemporal para rede de Petri

A obrigatoriedade de participação do conjunto-relacionamento (quando  $X$  é igual a 1), deve ser representada por uma conexão morta semelhante à conexão *cm18* da figura acima. Esta conexão morta estabelece que, se o empregado estiver presente no lugar *Empregado*, necessariamente deve estar presente pelo menos uma vez no lugar *Preferência*. Em outras palavras, se um empregado estiver presente no banco de dados, este empregado deve apresentar preferência por pelo menos uma função, e esta preferência não está vinculada ao tempo.

Em relação ao limite máximo de participação, quando este for igual a 1 ( $Y = 1$ ), deve ser representado por uma conexão morta semelhante à conexão *cm3f* que aparece na figura 6.7. Esta conexão determina que uma instância do conjunto *EntidadeA* não pode aparecer no conjunto-relacionamento *Relac\_ABC-Z* mais que uma vez.

As figuras 6.7 e 6.8 apresentam o mapeamento para rede de Petri de todas as restrições de cardinalidade previstas no modelo TempER relativas a conjuntos-relacionamento intemporais. As redes que aparecem nestas figuras são parciais, pois mostram apenas o mapeamento relativo ao conjunto-entidade *EntidadeA*. A mesma sistemática de mapeamento é aplicável no que tange aos demais conjuntos-entidade associados ( $B, C, \dots, Z$ ), obedecidas as respectivas restrições de cardinalidade.

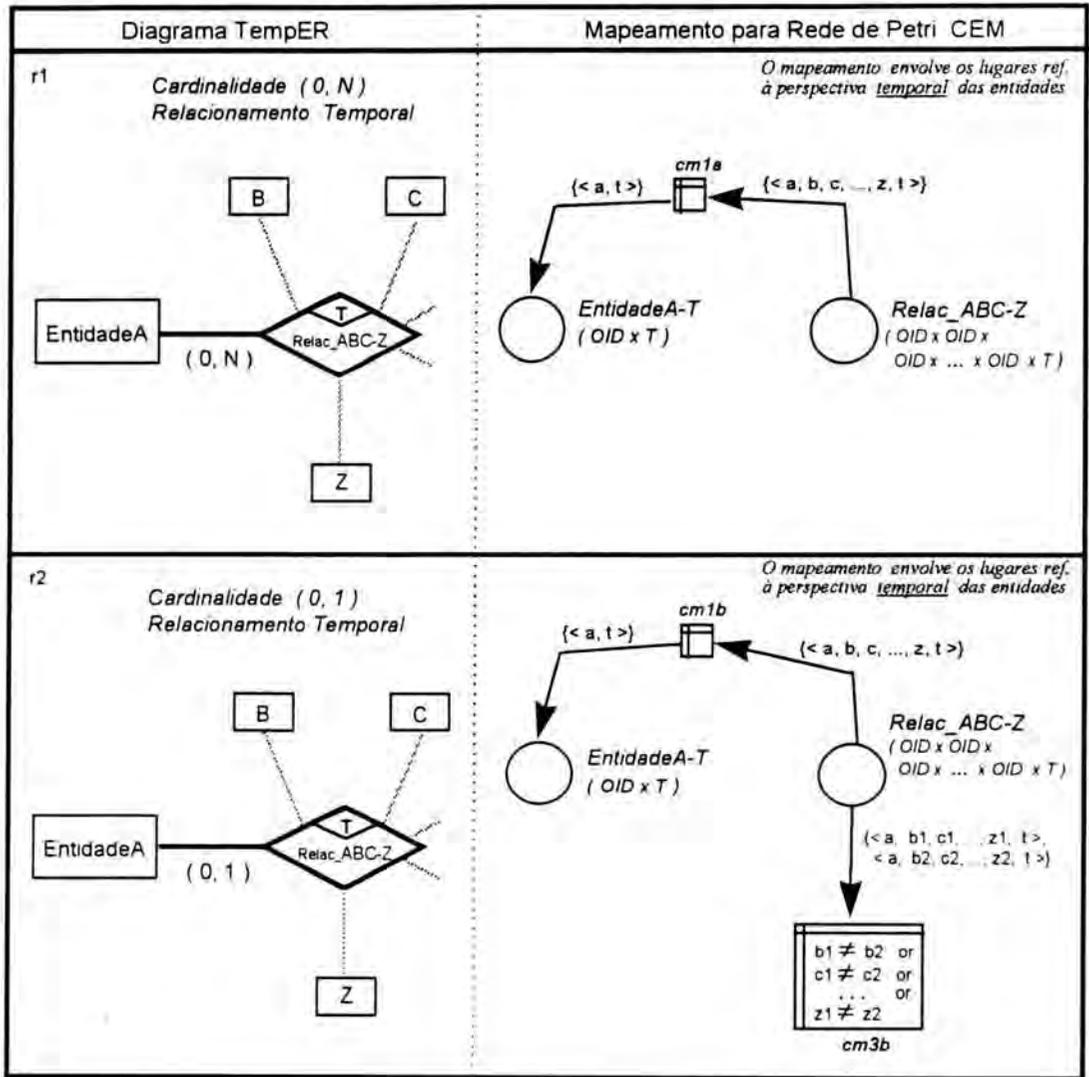


FIGURA 6.5 - Mapeamento das cardinalidades (0, N) e (0, 1) de relacionamentos temporais

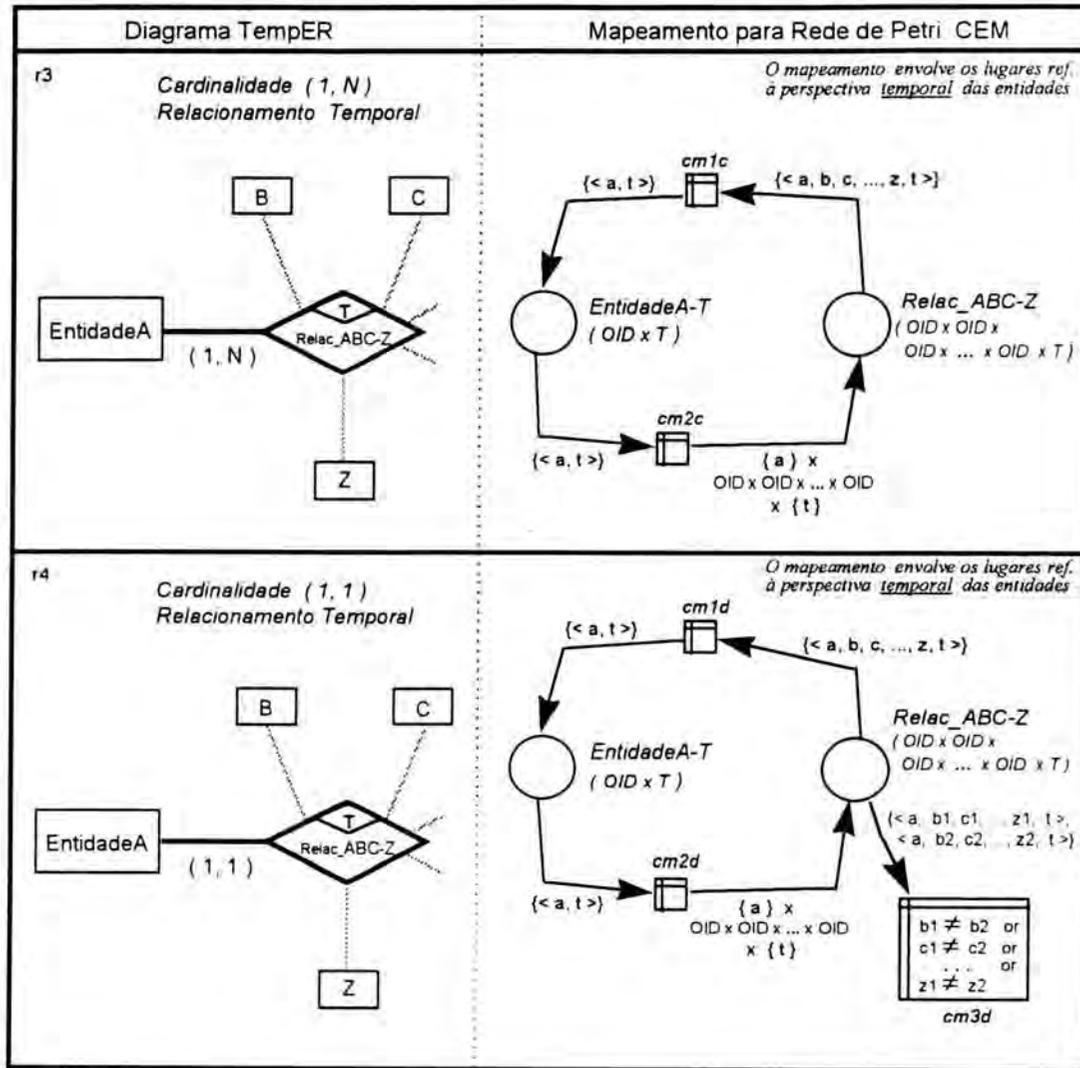


FIGURA 6.6 - Mapeamento das cardinalidades (1, N) e (1, 1) de relacionamentos temporais

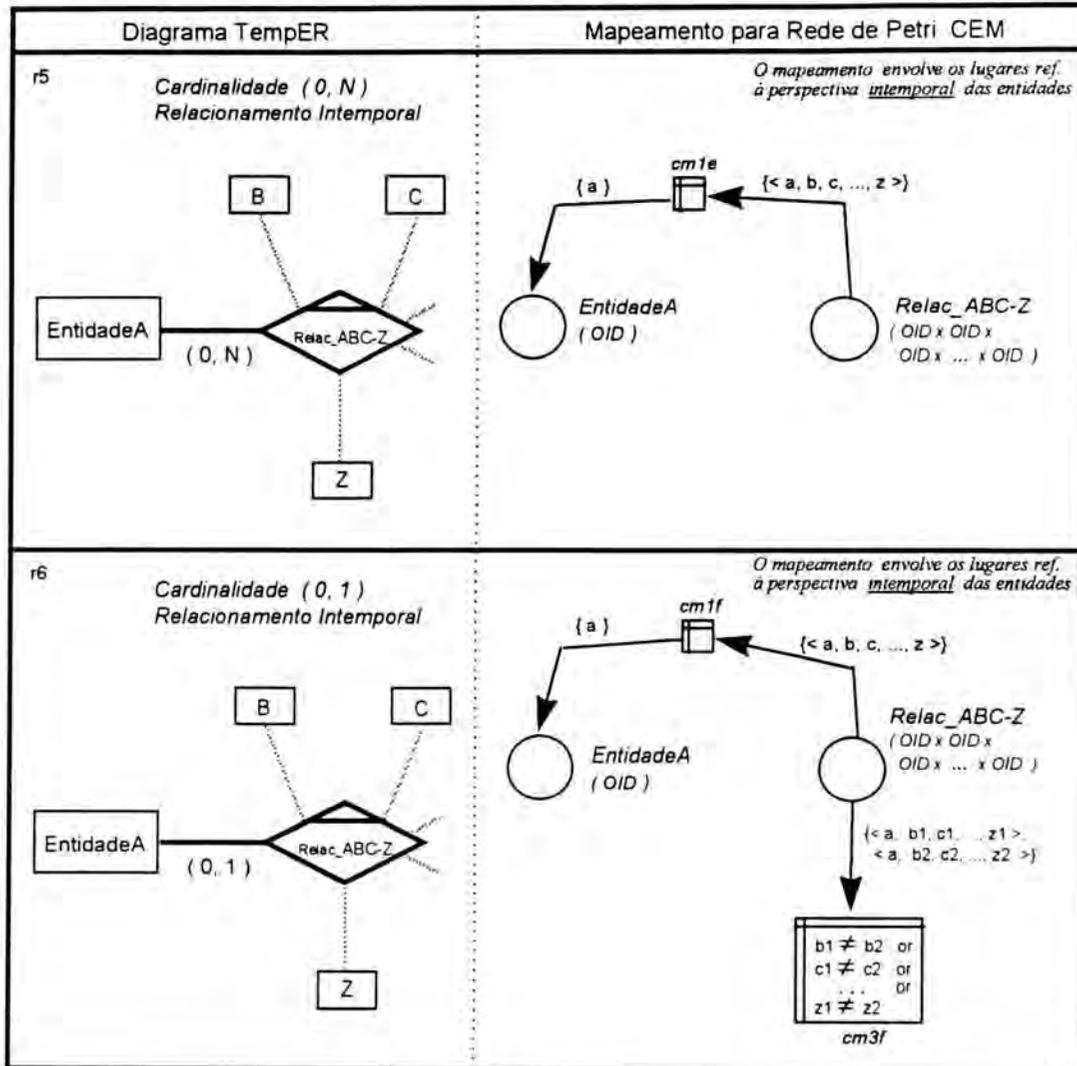


FIGURA 6.7 - Mapeamento das cardinalidades (0, N) e (0, 1) de relacionamentos intemporais

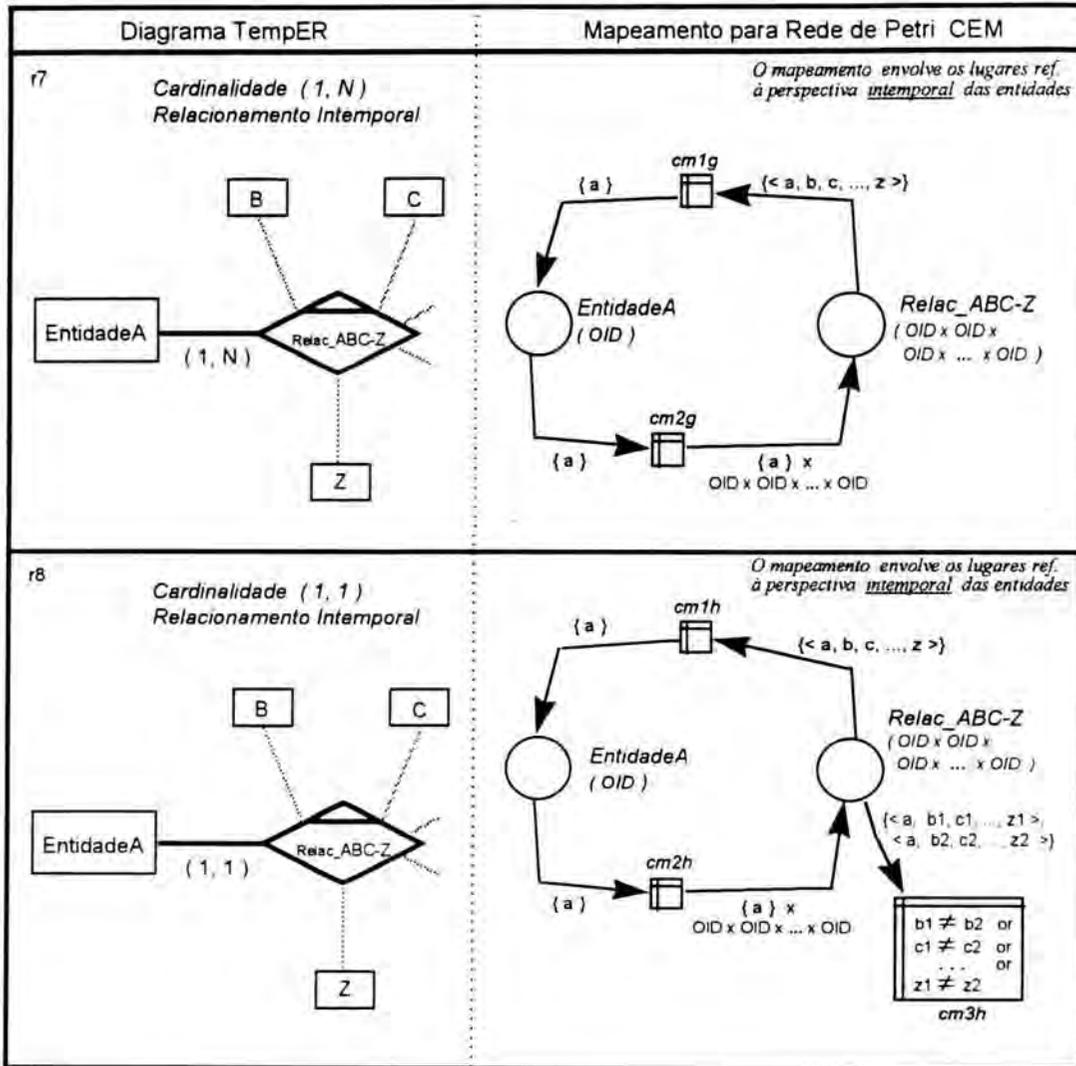


FIGURA 6.8 - Mapeamento das cardinalidades (1, N) e (1, 1) de relacionamentos intemporais

### 6.1.3 Mapeamento de atributos

Cada atributo no modelo TempER é representado na rede de Petri por exatamente um lugar, o qual apresenta como domínio o produto cartesiano entre o domínio do conjunto (entidade ou relacionamento) que contém tal atributo e o domínio dos seus valores possíveis. Um exemplo é o atributo intemporal *cod* que na rede de Petri da figura 6.9 apresenta como domínio (OID x NATURAL). Outro exemplo é o atributo temporal *at-sal* que apresenta como domínio (OID x REALP x T), onde além do domínio de OID's, também participa do produto cartesiano o domínio T que representa o conjunto de pontos do eixo temporal.

#### 6.1.3.1 Mapeamento dos atributos temporais

Em virtude de no modelo TempER apenas estarem disponíveis atributos do tipo obrigatório, os atributos temporais devem apresentar valor em todos os momentos da existência das entidades a que pertençam. Esta obrigatoriedade é representada na rede de Petri por uma conexão morta semelhante à conexão *cm30* da figura 6.9. Como, por definição, as alterações especificadas por uma conexão morta nunca estão habilitadas, então, para todo par <empregado, ponto do tempo>, representado na figura pelo termo {<e, t>}, que esteja presente no lugar *Empregado-T*, necessariamente deve haver no lugar *at-sal* pelo menos uma tupla do tipo <empregado, um numero real qualquer, ponto do tempo>.

Uma outra restrição estabelece que em um mesmo ponto do tempo, um atributo temporal não pode apresentar mais que um valor. Esta condição de monovaloração é representada na rede de Petri por uma conexão morta semelhante à conexão *cm31*.

Por fim, existe uma restrição intrínseca que estabelece que, se existe um valor de atributo em um certo ponto do tempo, necessariamente a dimensão temporal da entidade a que este atributo pertença contém este ponto do tempo. Tal restrição é representada na rede de Petri por uma conexão morta semelhante à conexão *cm29*.

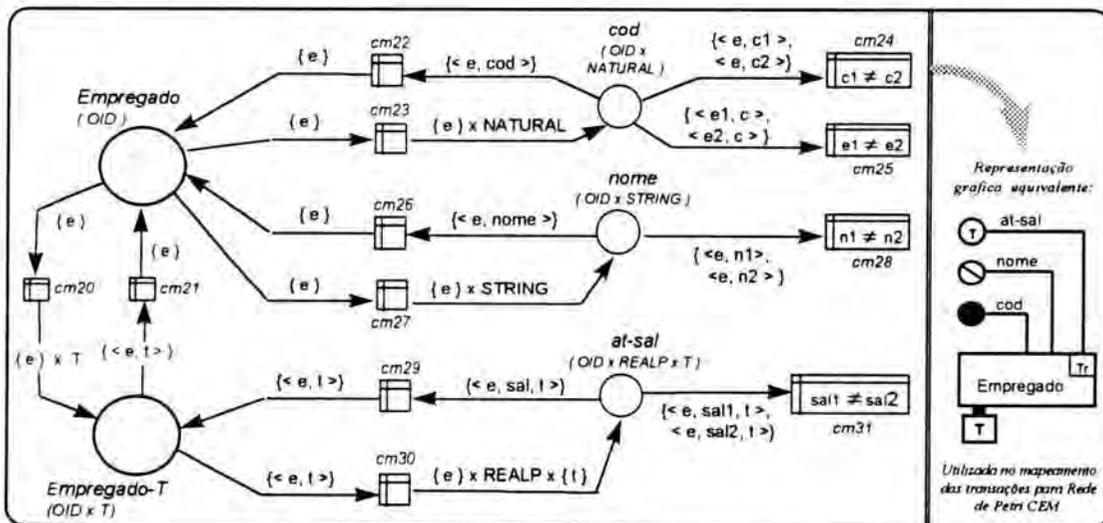


FIGURA 6.9 - Mapeamento de atributos para rede de Petri

Em relação ao mapeamento para rede de Petri de atributos pertencentes a relacionamentos, deve ser seguido um procedimento análogo ao dispensado às entidades, observando que os relacionamentos, ou existem na perspectiva temporal, ou existem na

perspectiva intemporal, nunca nas duas, como já foi explicado no capítulo 4. Se são relacionamentos do tipo temporal, possuem apenas atributos temporais; se são relacionamentos do tipo intemporal, apenas atributos intemporais.

### 6.1.3.2 Mapeamento dos atributos intemporais

Se os atributos intemporais forem de entidades, então estão vinculados única e exclusivamente ao OID destas, não é levado em conta a dimensão temporal. Por outro lado, se forem de relacionamentos, estão associados à tupla de OID's que configura tais relacionamentos. Sendo assim, e para que seja obedecida a regra de obrigatoriedade, sempre que uma entidade, ou relacionamento intemporal, esteja presente no banco de dados, os seus atributos intemporais necessariamente devem apresentar uma valoração. Esta restrição é representada na rede de Petri por uma conexão morta semelhante às conexões *cm23* e *cm27* da figura 6.9. Como por definição as alterações especificadas por uma conexão morta nunca estão habilitadas, então, tomando como exemplo a conexão *cm27*, para todo empregado, representado na figura pelo termo { *e* }, que esteja presente no lugar *Empregado*, necessariamente deve haver no lugar *nome* pelo menos um par <empregado, um string qualquer>, ou seja, esta conexão morta estabelece que um empregado deve ter no mínimo um nome.

Quanto à restrição de monovaloração que estabelece que um atributo intemporal deve apresentar no máximo um valor, esta é representada na rede de Petri por uma conexão morta semelhante às conexões *cm24* e *cm28* da figura 6.9.

A restrição intrínseca estabelecendo que, se um atributo intemporal apresenta um valor, necessariamente uma entidade (ou relacionamento) contendo este atributo deve estar presente no banco de dados, é representada na rede de Petri por uma conexão morta semelhante às conexões *cm22* e *cm26* da figura 6.9. Em outras palavras, nunca um valor que esteja presente no banco de dados é independente, sempre está vinculado a uma entidade (ou relacionamento).

### 6.1.3.3 Mapeamento de atributos identificadores

No modelo TempER, por motivo de simplificação, apenas atributos intemporais são admitidos no papel de identificadores externos. Para que um atributo possa atuar como identificador é necessário que o seu valor seja único no âmbito de todas as instâncias de um determinado conjunto-entidade. Esta restrição de unicidade é representada na rede de Petri por uma conexão morta semelhante à conexão *cm25* da figura 6.9. Para que as alterações definidas por esta conexão jamais venham a estar habilitadas, nunca dois empregados diferentes, representados na figura pelas variáveis *e1* e *e2*, podem possuir o mesmo valor para o atributo identificador *cod*.

## 6.1.4 Mapeamento da estrutura generalização/especialização

A estrutura generalização/especialização estabelece uma hierarquia entre conjuntos de entidades. Nesta hierarquia um conjunto desempenha o papel de conjunto-super-entidade e um ou mais conjuntos desempenham o papel de conjuntos-sub-entidade (em relação ao primeiro). Por se tratar de conjuntos de entidades, não importa o nível hierárquico que ocupem, todos serão representados na rede de Petri correspondente por dois lugares, um referente à perspectiva temporal das entidades e outro referente à perspectiva intemporal.

Em relação à dimensão temporal, este tipo de associação entre entidades classifica-se em *generalização/especialização temporal* e *generalização/especialização intemporal*. O mapeamento de ambas para rede de Petri segue regras semelhantes, sendo que a única diferença reside na referência a pontos do tempo, característica exclusiva da *generalização/especialização temporal*. Ambos os tipos apresentam duas sub-classificações, ortogonais entre si: *total* ou *parcial* e *disjunta* ou *sobreposta*, as quais serão representadas na rede de Petri pela presença ou ausência de conexões mortas, como será descrito nas subseções que seguem.

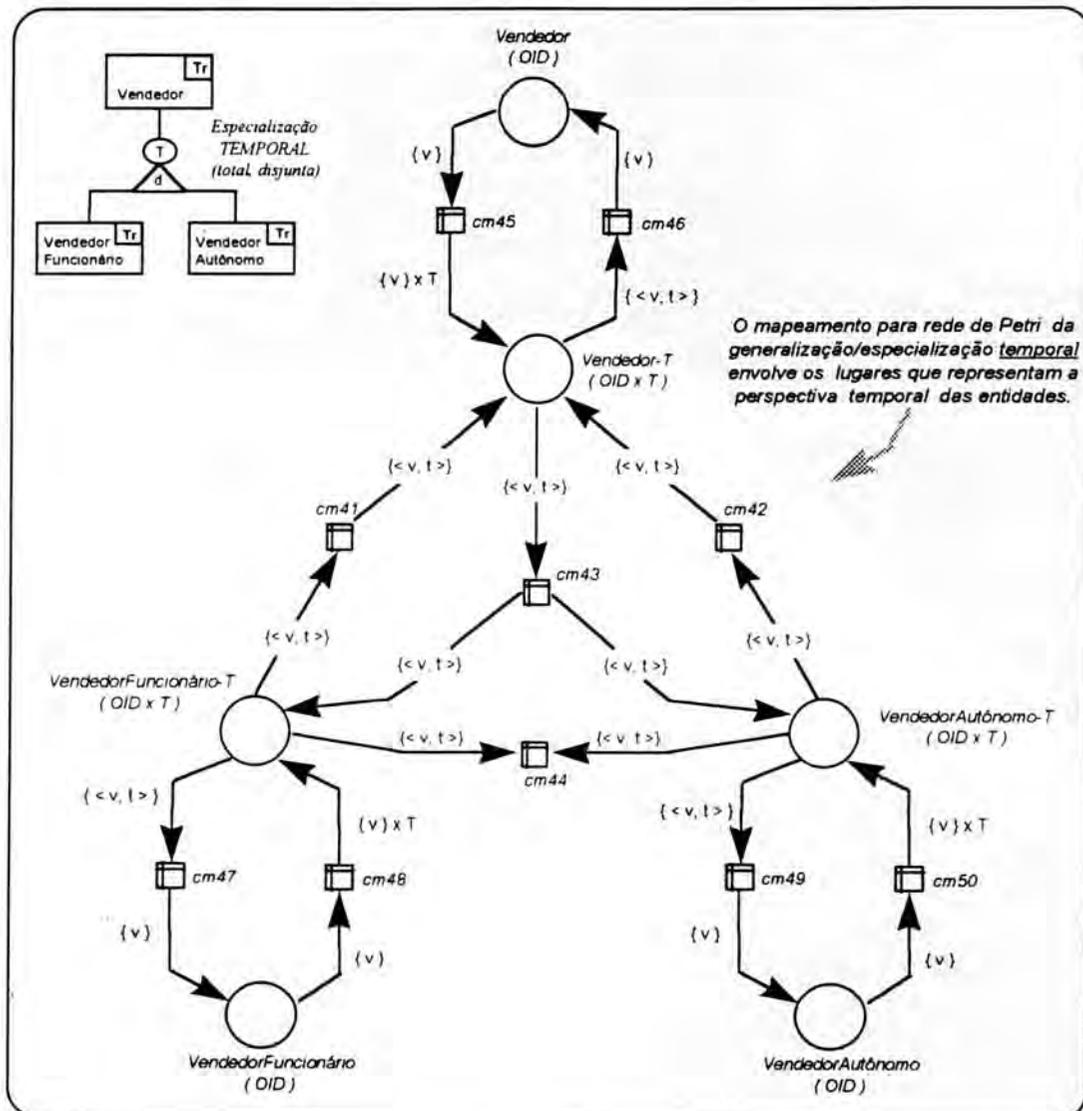


FIGURA 6.10 - Mapeamento de generalização/especialização temporal para rede de Petri

#### 6.1.4.1 Mapeamento de generalização/especialização temporal

A generalização/especialização temporal envolve apenas os lugares referentes à perspectiva temporal do conjunto-super-entidade e dos conjuntos-sub-entidade. É o que mostra a área interna branca da figura 6.10, onde os lugares que estão ligados via conexões mortas são *Vendedor-T*, *VendedorFuncionário-T* e *VendedorAutônomo-T*.

O exemplo mostrado na figura contempla a situação mais restritiva possível: a generalização/especialização total e disjunta.

Uma especialização temporal se diz *total* quando, para cada ponto do tempo da validade temporal de uma super-entidade, uma instância de mesmo OID está presente em pelo menos um conjunto-sub-entidade, neste mesmo ponto do tempo. Esta restrição é representada na rede de Petri por uma conexão morta semelhante às conexões *cm43* da figura 6.10, *cm3j* da figura 6.12(ge1) e *cm3l* da figura 6.13(ge3).

Em oposição, uma especialização temporal se diz *parcial* quando pode haver um ou mais pontos do tempo da validade temporal da super-entidade em que esta não apresenta qualquer instância correspondente no nível hierárquico inferior. Neste caso, a conexão morta acima citada deve ser omitida, como mostram as redes das figuras 6.12(ge2) e 6.13(ge4)

Uma especialização temporal se diz *disjunta* quando, em um determinado nível hierárquico e em um determinado ponto do tempo, uma entidade não pode estar presente em mais de um conjunto-sub-entidade. As conexões mortas *cm44* da figura 6.10, *cm4j* da figura 6.12(ge1) e *cm4k* da figura 6.12(ge2), mostram como representar esta restrição na rede de Petri. Destas, a conexão *cm44*, por exemplo, especifica que um vendedor (*v*) que esteja em um certo ponto do tempo (*t*) no papel de vendedor-funcionário, neste mesmo ponto do tempo não pode atuar como vendedor-autônomo, e vice-versa.

Em oposição, uma especialização temporal se diz *sobreposta* quando é possível que em um mesmo ponto do tempo uma entidade exista em dois ou mais conjuntos-sub-entidade de mesmo nível hierárquico. Este fato é representado pela omissão da conexão morta descrita no parágrafo anterior, como mostram as duas redes da figura 6.13.

Uma restrição intrínseca ao modelo TempER estabelece que, se uma entidade está presente em um conjunto-sub-entidade em um certo ponto do tempo, necessariamente uma instância de mesmo OID deve estar presente, neste mesmo ponto do tempo, no respectivo conjunto-super-entidade. Trata-se de uma relação de dependência, isto é, uma sub-entidade só existe se existir a super-entidade correspondente. Esta restrição é representada na rede de Petri por uma conexão morta semelhante às conexões *cm41* e *cm42* da figura 6.10.

#### 6.1.4.2 Mapeamento de generalização/especialização intemporal

A generalização/especialização intemporal é análoga a que foi descrita acima. A diferença é que agora são envolvidos apenas os lugares referentes à perspectiva intemporal do conjunto-super-entidade e dos conjuntos-sub-entidade, isto é, não existe referência a pontos do tempo. É o que mostra a área branca da figura 6.11, onde os lugares *Empregado*, *Homem* e *Mulher* estão ligados via as conexões mortas que estabelecem as restrições referentes à generalização/especialização.

O exemplo mostrado na figura contempla a situação mais restritiva possível: a generalização/especialização total e disjunta.

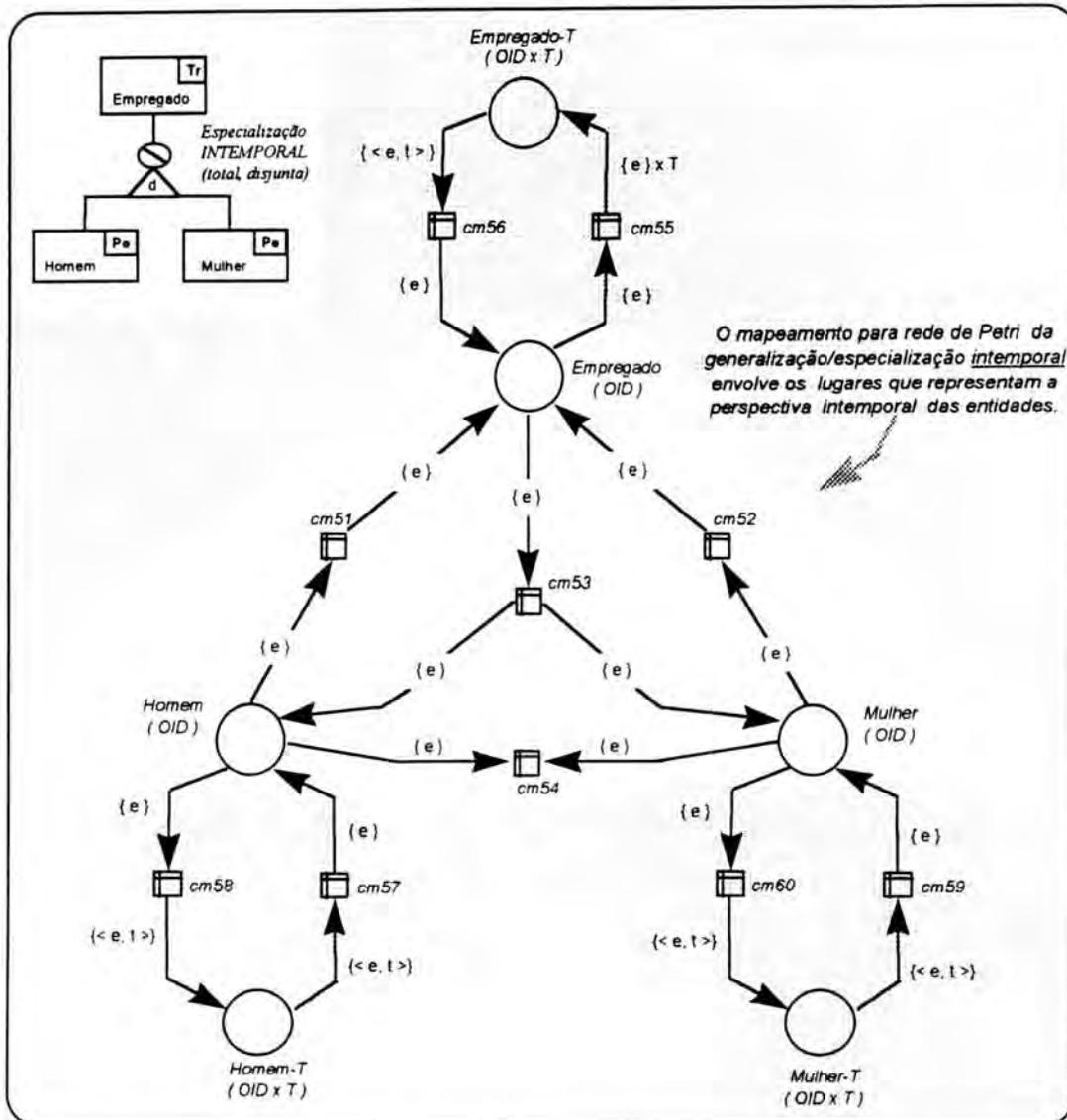


FIGURA 6.11 - Mapeamento de generalização/especialização intemporal para rede de Petri

Uma especialização intemporal se diz *total* quando a entidade, ou mais especificamente o seu OID, ao estar presente no conjunto-super-entidade, também deve estar presente no conjunto-sub-entidade. Esta restrição é representada na rede de Petri por uma conexão morta semelhante às conexões  $cm53$  da figura 6.11,  $cm3n$  da figura 6.14(ge5) e  $cm3p$  da figura 6.15(ge7).

Em oposição, uma especialização intemporal se diz *parcial* quando uma entidade que esteja presente no conjunto-super-entidade pode estar ausente no nível hierárquico inferior. Neste caso, a conexão morta acima citada deve ser omitida, como mostram as redes das figuras 6.14(ge6) e 6.15(ge8)

Uma especialização intemporal se diz *disjunta* quando, em um determinado nível hierárquico, uma entidade não pode estar presente em mais que um conjunto-sub-entidade. As conexões mortas  $cm54$  da figura 6.11,  $cm4n$  da figura 6.14(ge5) e  $cm4o$  da figura 6.14(ge6), mostram como representar esta restrição em rede de Petri. Destas, a conexão  $cm54$ , por exemplo, especifica que um empregado ( $e$ ) ou é homem ou é mulher, nunca as duas coisas.

Em oposição, uma especialização intemporal se diz *sobreposta* quando é possível que uma entidade esteja presente em dois ou mais conjuntos-sub-entidade de mesmo nível hierárquico. Este fato é representado pela omissão da conexão morta descrita no parágrafo anterior, como mostram as duas redes da figura 6.15.

Em relação à generalização/especialização intemporal, também existe uma restrição intrínseca ao modelo TempER estabelecendo que, se uma entidade está presente em um conjunto-sub-entidade, necessariamente uma instância de mesmo OID deve estar presente no respectivo conjunto-super-entidade. Trata-se de uma relação de dependência, isto é, uma sub-entidade só existe se existir a super-entidade correspondente. Esta restrição é representada na rede de Petri por uma conexão morta semelhante às conexões *cm51* e *cm52* da figura 6.11.

As figuras 6.12 a 6.15, que seguem, mostram como mapear para rede de Petri todas as combinações de generalização/especialização do modelo de dados TempER.

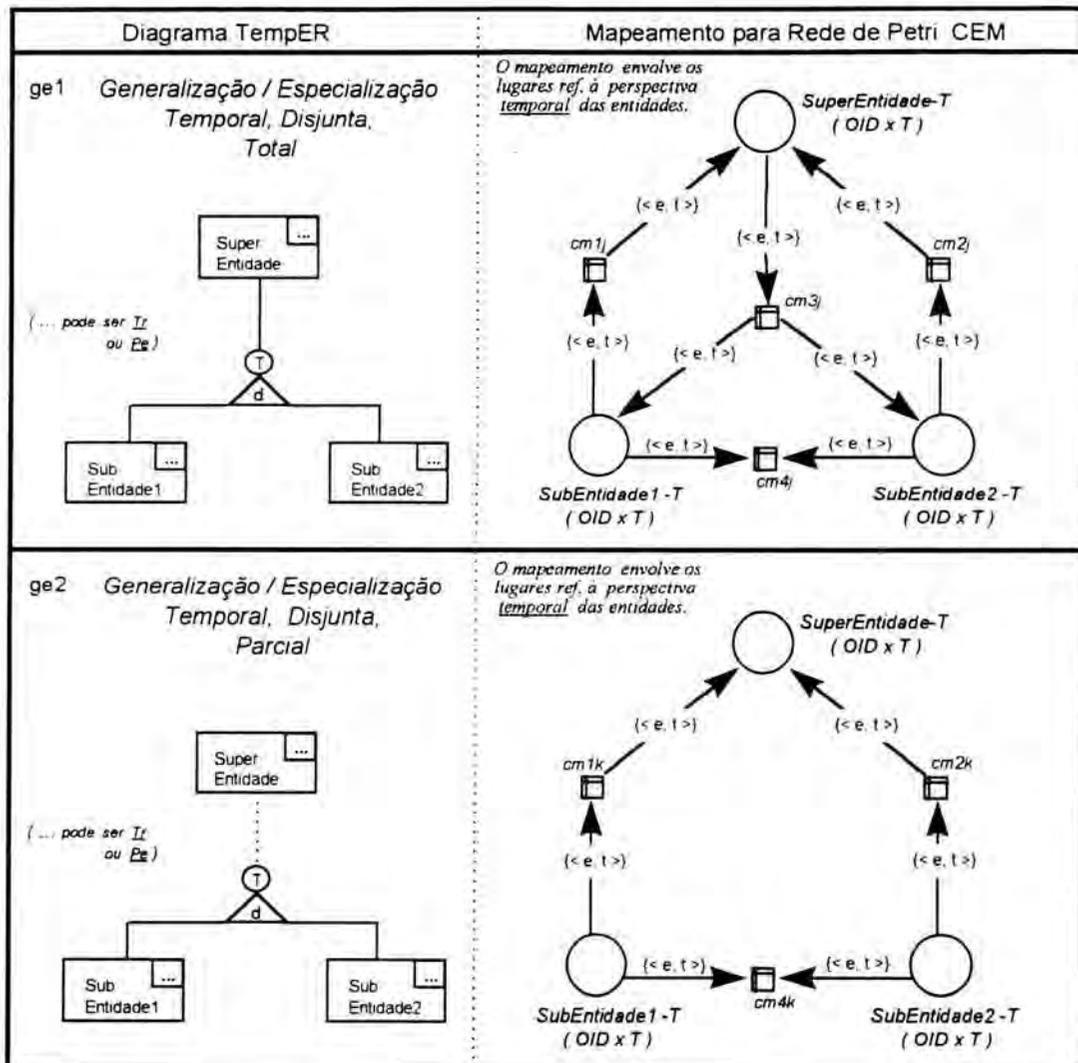


FIGURA 6.12 - Mapeamento de generalização/especialização temporal do tipo disjunta (total e parcial)

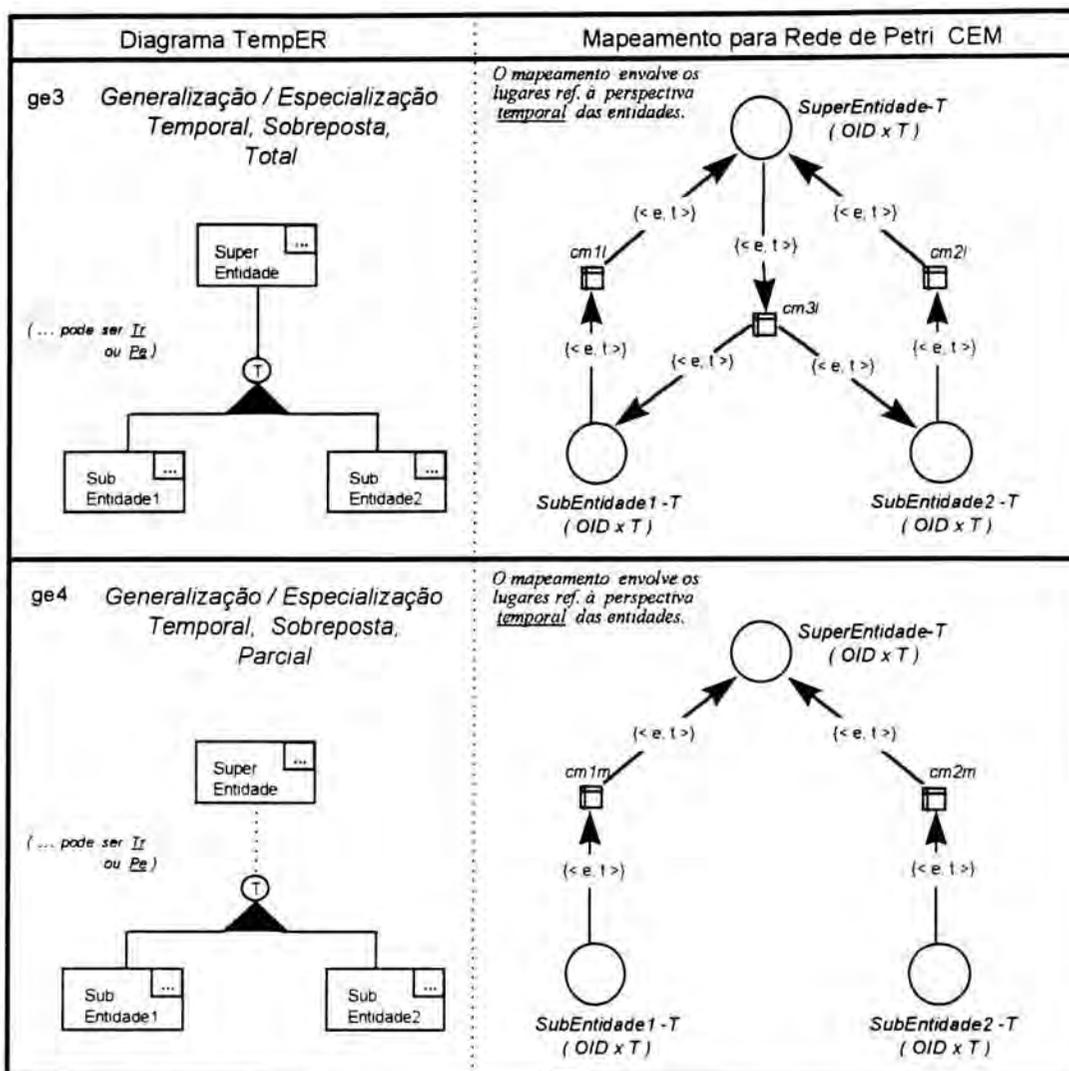


FIGURA 6.13 - Mapeamento de generalização/especialização temporal do tipo sobreposta (total e parcial)

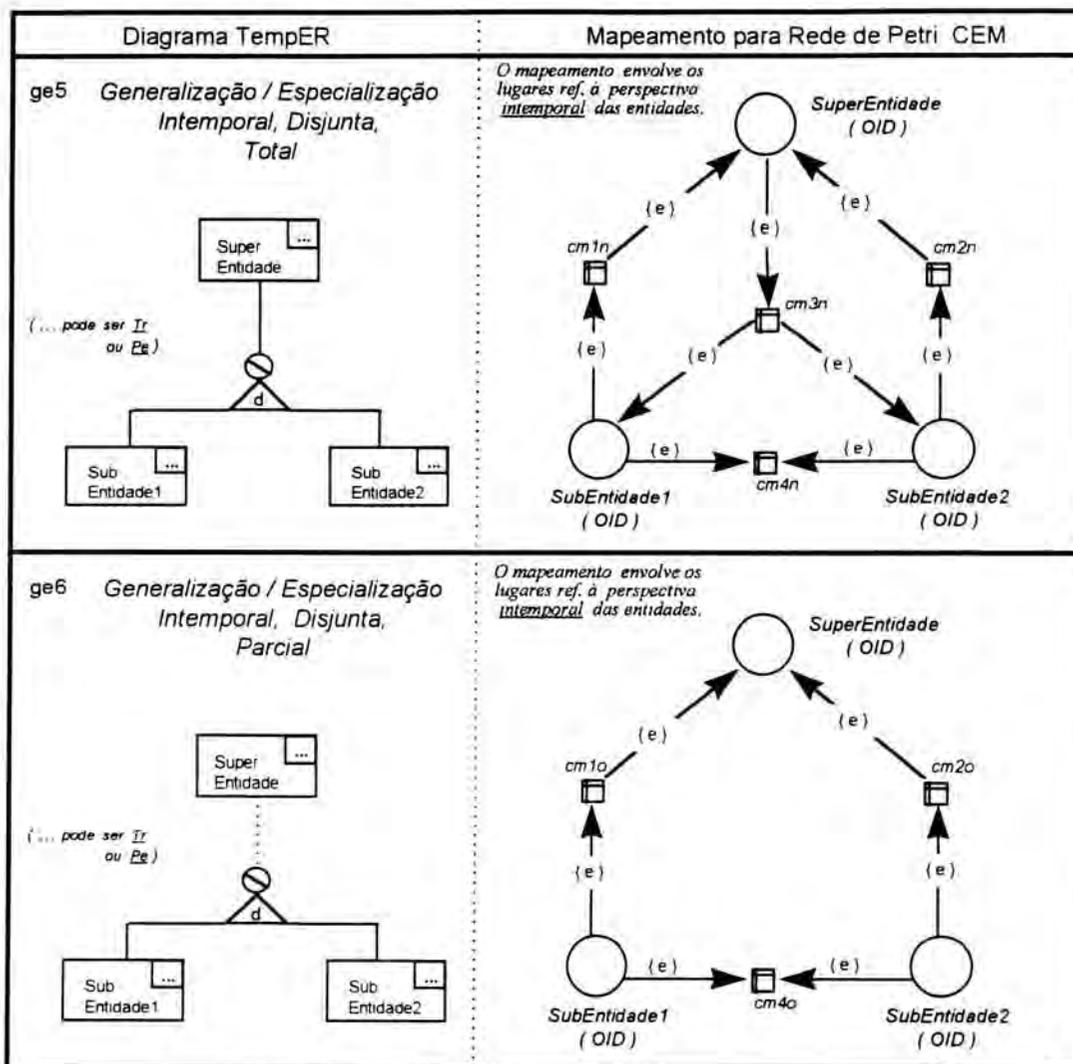


FIGURA 6.14 - Mapeamento de generalização/especialização intemporal do tipo disjunta (total e parcial)

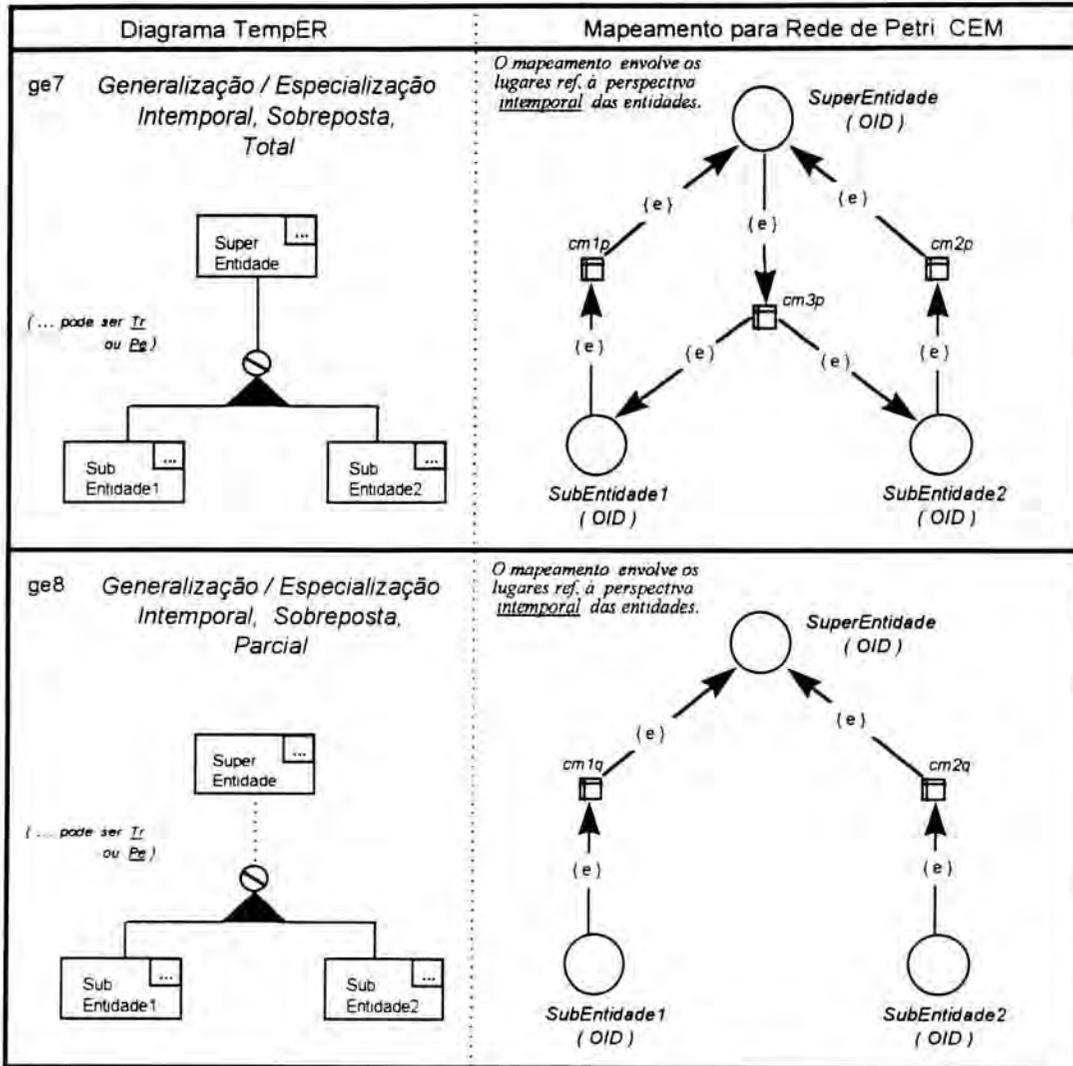


FIGURA 6.15 - Mapeamento de generalização/especialização intemporal do tipo sobreposta (total e parcial)

## 6.2 Mapeamento do modelo dinâmico - transações - para rede de Petri

Nesta seção será explicado de que forma o modelo TempER-Tr, que integra a modelagem de transações a um modelo de dados temporal, pode ser traduzido para uma rede de Petri. Como foi visto na seção anterior deste capítulo, o modelo de dados temporal é completamente mapeável para uma rede de Petri através do uso de conexões mortas. Falta traduzir o modelo dinâmico, ou seja, as transações do sistema, e isto se dá através das *conexões vivas*, que, para efeito de simplificação, doravante serão chamadas apenas de conexões.

Cada transação do modelo TempER-Tr é representada na rede de Petri por uma conexão cujo símbolo é um quadrado. Além disto, os fluxos externos e respectivos eventos, assim como os fluxos internos das transações, encontram uma simbologia correspondente na rede de Petri que será descrita a seguir.

A rede de Petri resultante da tradução de uma transação é a combinação de dois segmentos, um representando as propriedades dinâmicas e o outro representando as propriedades estáticas. Visando única e exclusivamente facilitar a visualização, adota-se a providência de substituir o segmento estático da rede pelo próprio modelo de dados temporal, já que são equivalentes entre si. A única adaptação necessária para tanto é a explicitação visual dos atributos, como mostra o diagrama que se encontra no lado direito da figura 6.9.

### 6.2.1 Mapeamento dos fluxos externos

Os fluxos externos de uma transação são traduzidos para um segmento de rede de Petri semelhante ao mostrado na figura 6.16. Cada fluxo externo corresponde a um lugar ligado à conexão que representa a transação por meio de um ramo alterador de mesmo sentido do fluxo. O evento externo é representado por uma conexão ligada ao lugar que representa o fluxo externo através de um ramo alterador oposto ao tipo de fluxo.

Exemplificando, toda vez que um evento externo ocorre, um dado de entrada é depositado no lugar que representa o fluxo. Por sua vez, a conexão referente à transação (*xyz* na figura 6.16) modela o consumo de um dado deste lugar. De forma semelhante, cada vez que um dado de saída é produzido pela conexão referente à transação, ele é colocado no lugar a partir do qual um processo externo ao sistema o consumirá.

Os eventos externos podem ocorrer concorrentemente e cada um deles dispara uma transação do sistema. Para empacotar os parâmetros de cada evento, é anexado um identificador de parâmetro (PID). O lugar *IdentificadoresOcupados* que aparece na rede de Petri da figura 6.16 tem a finalidade de controlar o uso destes identificadores, evitando que o mesmo seja utilizado por eventos diferentes.

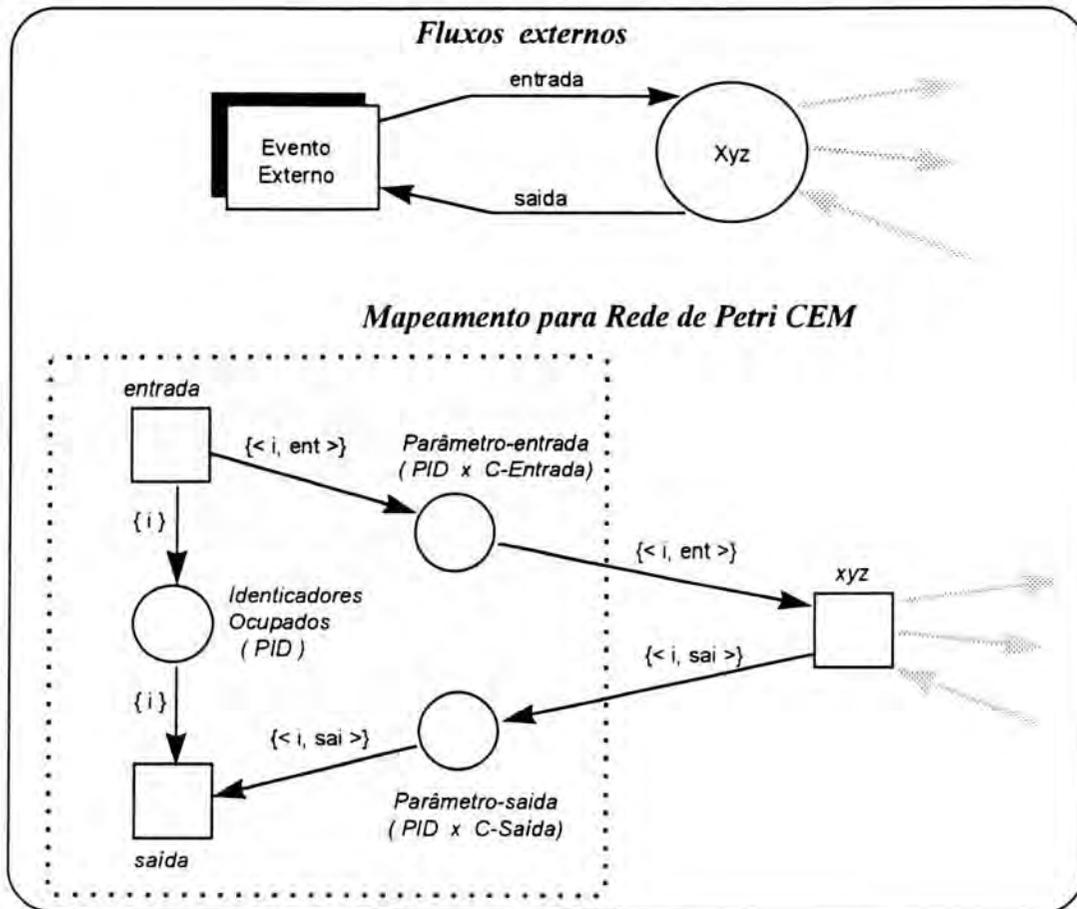


FIGURA 6.16 - Mapeamento de fluxos externos para rede de Petri

## 6.2.2 Mapeamento dos fluxos internos

Os fluxos internos são traduzidos para *ramos* na rede de Petri. As anotações associadas a eles, são traduzidas para *termos* dos ramos, com a ressalva que as restrições temporais que se encontram nestas anotações são mapeadas para produtos cartesianos entre indivíduos e o conjunto de pontos por elas especificados, conforme mostram as diversas figuras desta seção. Com relação aos termos de fluxos ligados a atributos temporais, estes também são organizados na forma de produtos cartesianos, entre os indivíduos e o assinalamento temporal definido pela variável que se refere ao atributo (ver referências ao atributo *at-y* e à variável de mesmo nome nas figuras desta seção).

### 6.2.2.1 Mapeamento dos fluxos de inclusão

Um fluxo de inclusão é traduzido para um ramo alterador de saída apontando para o conjunto-entidade ou conjunto-relacionamento, mais um ramo alterador de saída apontando para o lugar que representa a perspectiva temporal do conjunto-entidade, caso se trate de inclusão de entidade, e mais um ramo alterador de saída apontando para cada atributo que o conjunto-entidade/conjunto-relacionamento possuir, pois uma inclusão implica na inclusão de todos os atributos.

As figuras 6.17 a 6.20 mostram o mapeamento dos fluxos de inclusão para cada tipo de conjunto envolvido: entidades transitórias, entidades perenes, relacionamentos temporais e relacionamentos intemporais.

Na figura 6.17, a variável *at-y* contém o assinalamento temporal (ver definição na seção 3.14 do capítulo 3) a ser incluído no atributo de mesmo nome. O domínio temporal deste assinalamento deve ser, necessariamente, igual ao conjunto de todos os pontos da existência da entidade que está sendo incluída. Caso contrário, estaria sendo violada a regra de obrigatoriedade dos atributos. Conseqüentemente, a transação não se tornaria habilitada.

Como pode ser notado nas figuras 6.17 a 6.20, uma fórmula adicional deve ser incluída na rede de Petri, entre as demais fórmulas da conexão. Trata-se da fórmula que estabelece uma relação entre a variável *cj-pontos-tempo* e a restrição temporal que foi definida junto ao fluxo de inclusão. Se, por exemplo, a restrição temporal for do tipo  $[ti, tf]$ , que tem o formato de um intervalo de tempo, essa fórmula será igual à expressão abaixo, onde  $\mathbf{T}$  é o conjunto de todos os pontos do tempo.

$$cj\text{-pontos-tempo} = \{ t \mid t \in \mathbf{T} \text{ and } t \geq ti \text{ and } t \leq tf \}$$

Em relação à inclusão de entidades perenes, é importante notar que, no fluxo da transação não aparece qualquer restrição temporal, isto porque é assumido que este tipo de entidade *existe* em todos os pontos do eixo temporal. Entretanto, na rede de Petri correspondente, deve estar presente um ramo alterador de saída, apontando para o lugar que representa a dimensão temporal do conjunto-entidade. Este ramo especifica a inclusão implícita de todos os pontos do eixo temporal  $\mathbf{T}$ .

Caso uma entidade perene, que esteja sendo incluída, possua atributos temporais, estes receberão valor em todos os pontos do eixo temporal (regra da obrigatoriedade). Por exemplo, uma alteração, definida pela conexão da figura 6.18, só estará habilitada a ocorrer se o domínio temporal da variável *at-y* for igual a *cj-pontos-tempo* (que por sua vez deve ser igual a todo o conjunto de pontos do eixo temporal  $\mathbf{T}$ ).

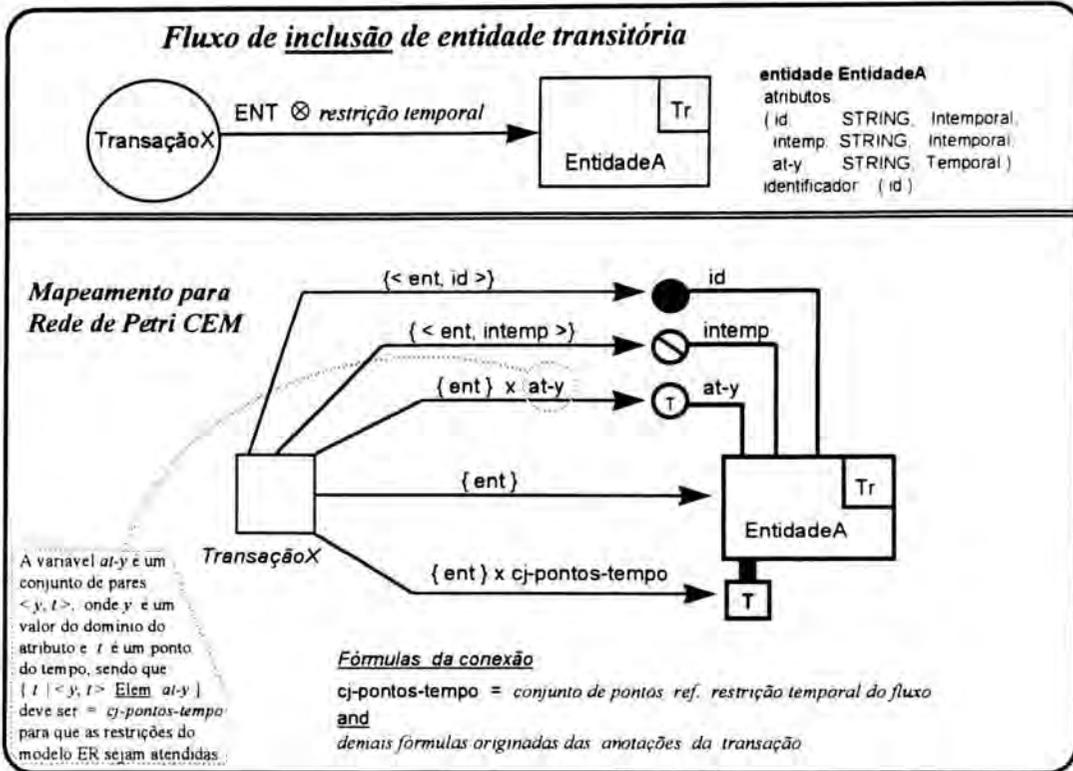


FIGURA 6.17 - Mapeamento de fluxo de inclusão de entidade transitória para rede de Petri

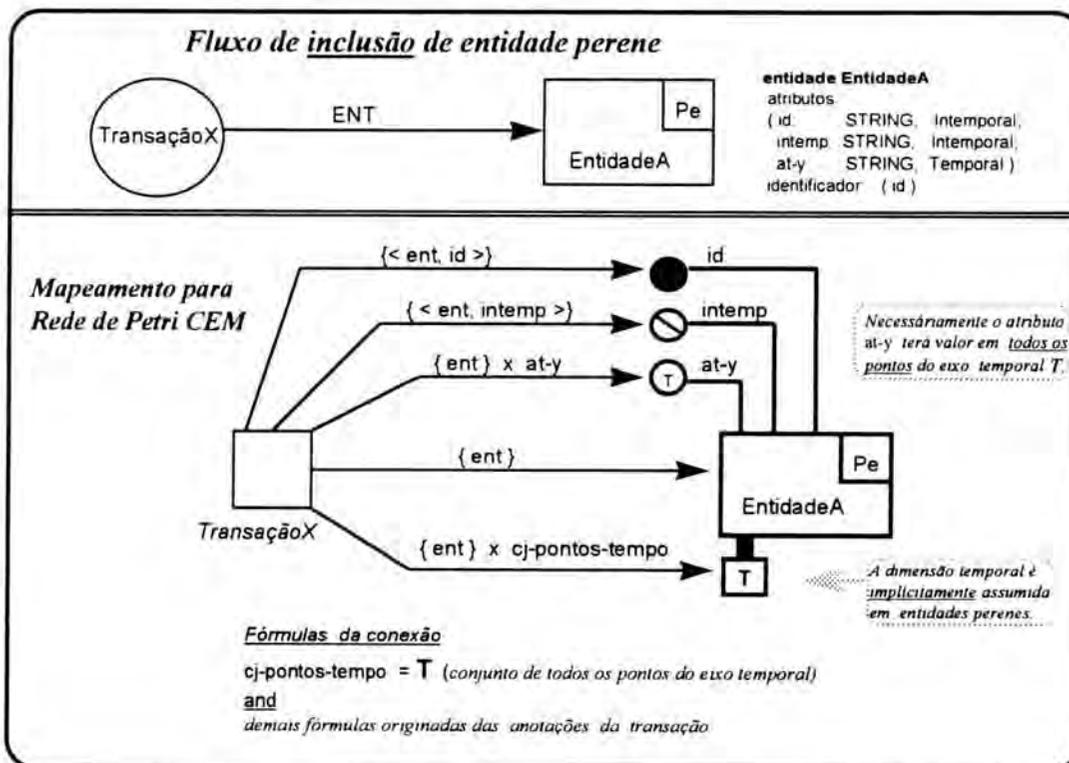


FIGURA 6.18 - Mapeamento de fluxo de inclusão de entidade perene para rede de Petri

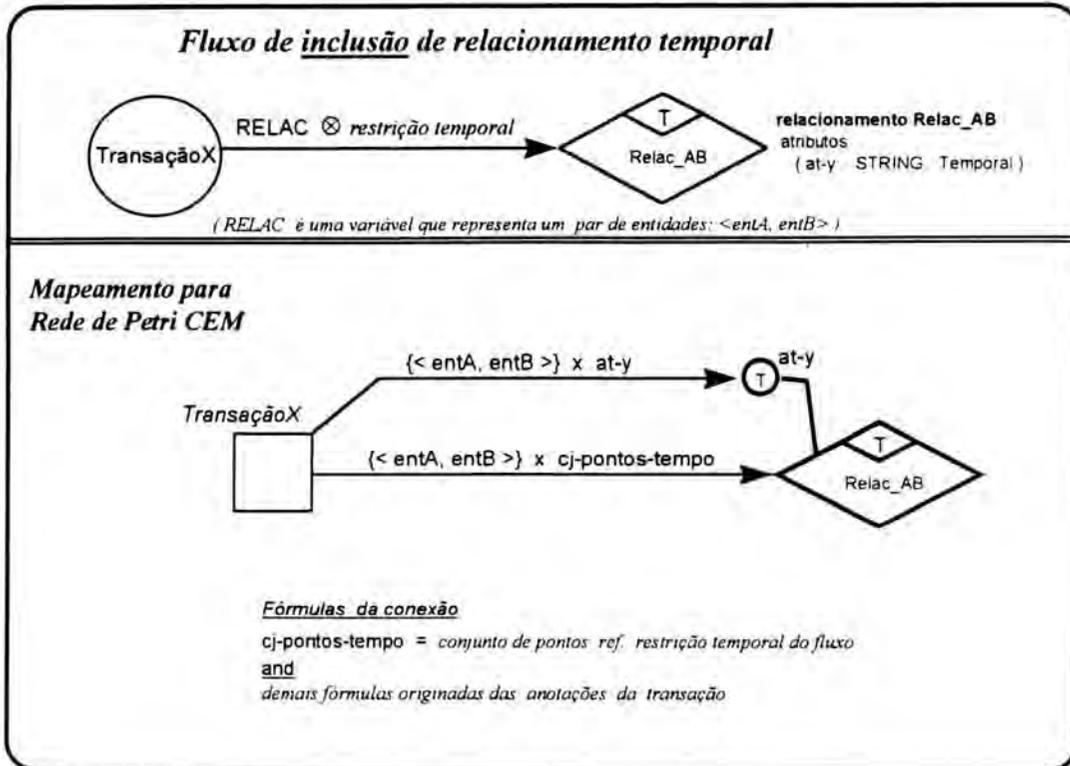


FIGURA 6.19 - Mapeamento de fluxo de inclusão de relacionamento temporal para rede de Petri

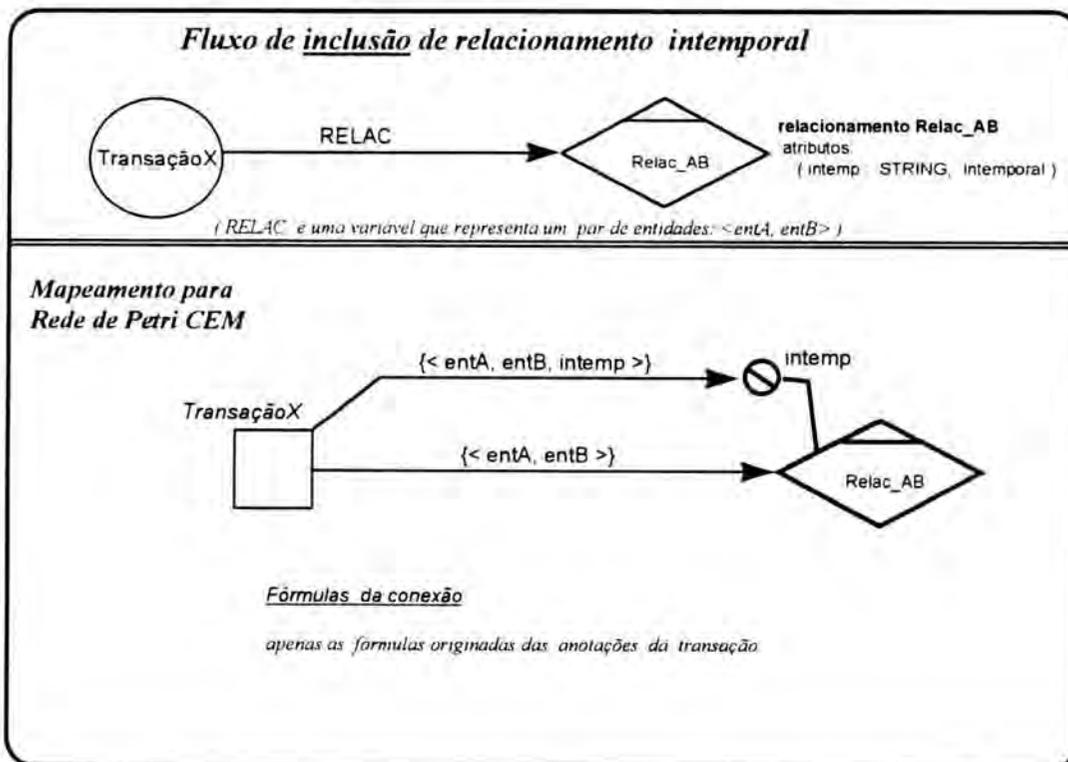


FIGURA 6.20 - Mapeamento de fluxo de inclusão de relacionamento intemporal para rede de Petri

### 6.2.2.2 Mapeamento dos fluxos de exclusão

Um fluxo de exclusão é traduzido para um ramo alterador de entrada originário do conjunto-entidade ou conjunto-relacionamento, mais um ramo alterador de entrada originário do lugar que representa a perspectiva temporal do conjunto-entidade, caso se trate de exclusão de entidade, e mais um ramo alterador de entrada para cada atributo que o conjunto-entidade/conjunto-relacionamento possuir, pois uma exclusão de entidade ou relacionamento, implica na exclusão de todos os respectivos atributos.

O mapeamento dos fluxos de exclusão é análogo aos de inclusão, bastando que se inverta o sentido dos ramos alteradores.

### 6.2.2.3 Mapeamento dos fluxos de manipulação de existências

Os fluxos de manipulação de existência atuam apenas na perspectiva temporal das entidades transitórias, permitindo que se amplie ou reduza a validade temporal destas.

A tradução para rede de Petri de um fluxo de ampliação de existência de entidade transitória origina um ramo alterador de saída apontando para o lugar que representa a perspectiva temporal do conjunto-entidade (o pequeno quadrado contendo a letra "T") e um ramo alterador de saída para cada atributo temporal que o conjunto-entidade possua. A figura 6.21 ilustra como se dá este mapeamento.

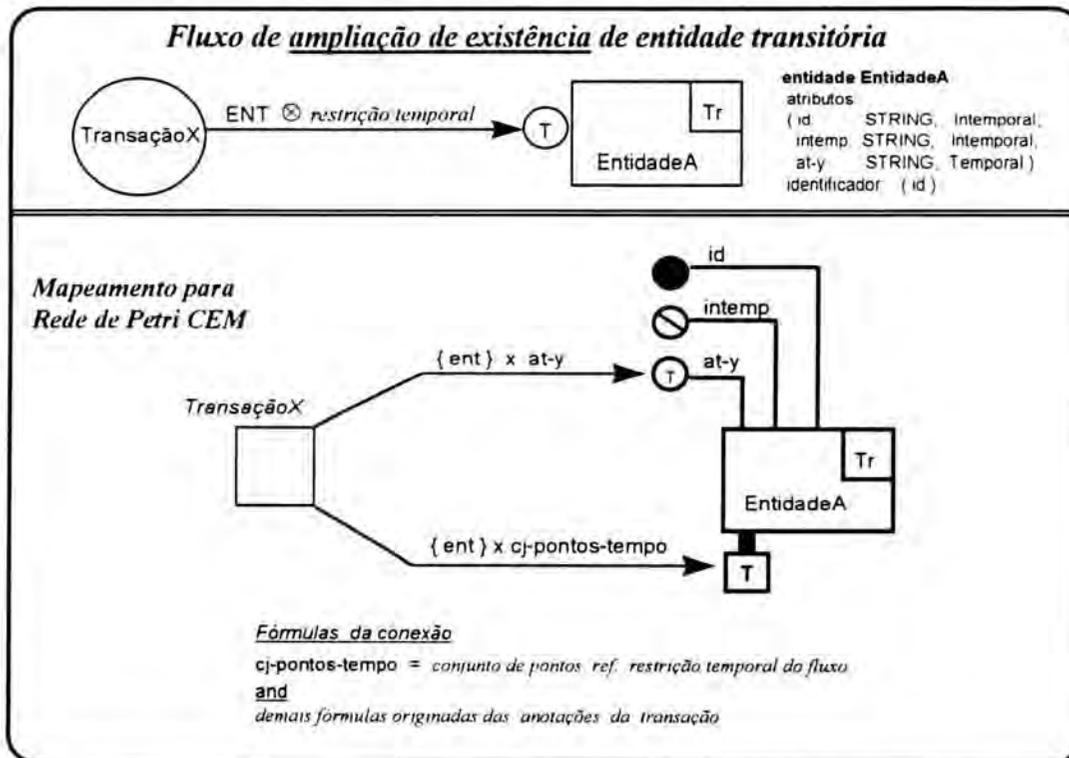


FIGURA 6.21 - Mapeamento de fluxo de ampliação de existência de entidade transitória para rede de Petri

Caso seja fluxo de redução de existência de entidade transitória, este deve ser traduzido para um ramo alterador de entrada originário do lugar que representa a perspectiva temporal do conjunto-entidade e um ramo alterador de entrada para cada

atributo temporal que o conjunto-entidade possua. O mapeamento deste tipo de fluxo é, portanto, análogo ao de ampliação de existência, bastando que se inverta o sentido dos ramos alteradores.

**6.2.2.4 Mapeamento dos fluxos de alteração de atributos**

Os fluxos de alteração são utilizados única e exclusivamente para modelar modificações de valores dos atributos de entidades e relacionamentos, não criando, nem excluindo instâncias, tampouco modificando a validade temporal de entidades transitórias e relacionamentos temporais.

Para cada atributo citado nas anotações de um fluxo de alteração são gerados dois ramos alteradores na rede de Petri, um de entrada e um de saída, ligados ao lugar que representa o atributo. O ramo alterador de entrada representa o valor que vai desaparecer, e o ramo alterador de saída representa o novo valor que vai aparecer.

A variável referente ao ramo alterador de entrada, que modela o valor a ser substituído, por convenção leva o sufixo “-old”. Por sua vez, a variável referente ao ramo alterador de saída, que modela o novo valor do atributo, leva o sufixo “-new”.

Caso, na anotação associada ao fluxo de alteração exista a cláusula *arg*, além dos ramos acima citados, o mapeamento resultará em mais um ramo restaurador de entrada para cada atributo referido nesta cláusula, cuja finalidade é verificar se as entidades ou relacionamentos possuem determinados valores.

A figura 6.22 ilustra o mapeamento descrito acima, em relação a alteração de entidades. Quanto aos relacionamentos, o tratamento é idêntico.

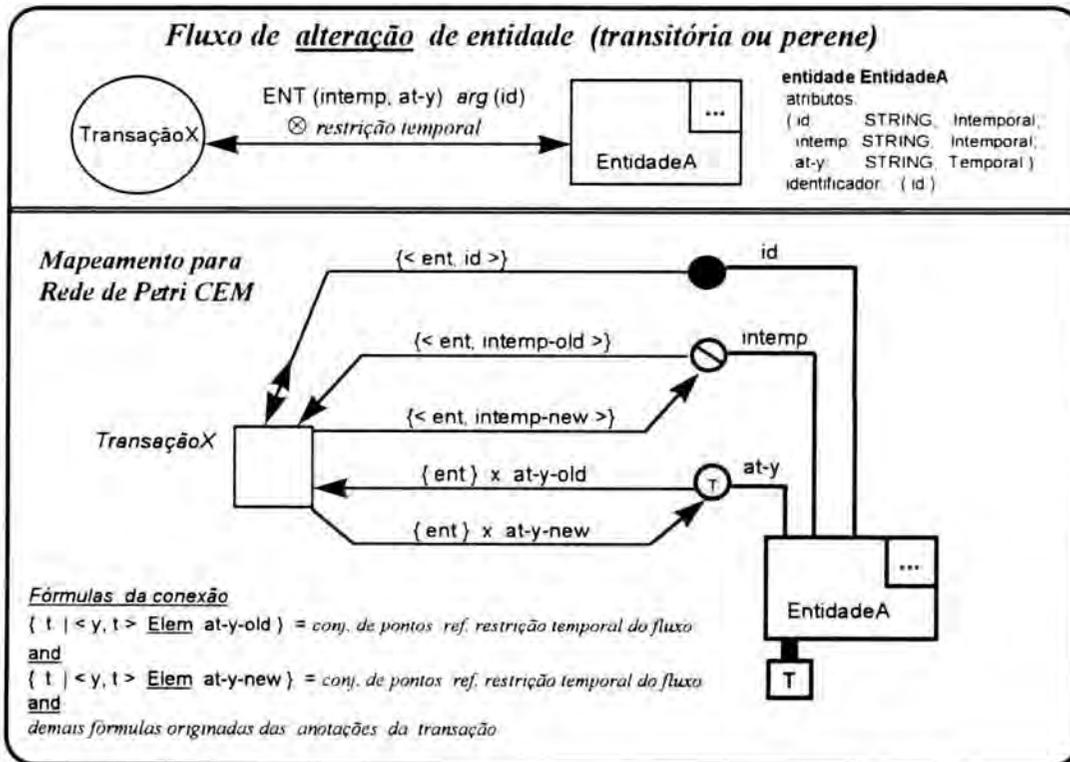


FIGURA 6.22 - Mapeamento de fluxo de alteração atributos para rede de Petri

Quando atributos temporais estão sendo modificados é importante notar que não se diminui e nem se amplia o domínio temporal destes, apenas altera-se os valores, no âmbito da faixa temporal determinada pela restrição associada ao fluxo.

#### 6.2.2.5 Mapeamento dos fluxos de verificação de presença

Os fluxos de verificação de presença não modificam o banco de dados, apenas testam a presença de entidades, de relacionamentos e de valores de atributos. São traduzidos para *ramos restauradores de entrada* quando do seu mapeamento para rede de Petri. A anotação junto ao fluxo pode conter referência a atributos, ou conter a partícula “*none*”. Se contiver referência a atributos, na rede de Petri correspondente necessariamente deve haver um ramo restaurador de entrada para cada atributo citado, sendo que neste caso o ramo restaurador de entrada ligado ao próprio conjunto-entidade pode ser dispensado, simplificando a rede. Tal simplificação só é possível porque, se um atributo está presente, necessariamente a respectiva entidade também está presente. A figura 6.23 ilustra o mapeamento deste tipo de fluxo.

Se o fluxo de verificação de presença for do tipo específico que testa se uma entidade transitória é vigente em um determinado período de tempo, então na rede de Petri, em vez do ramo restaurador de entrada estar ligado ao conjunto-entidade, estará ligado ao lugar que representa a sua perspectiva temporal (o pequeno quadrado contendo a letra “T”). Contudo este último ramo também pode ser dispensado, caso nas anotações associadas ao fluxo haja referência a pelo menos um atributo temporal. Esta simplificação só é possível porque se um atributo temporal apresenta uma valoração em um certo período de tempo, necessariamente a validade temporal da respectiva entidade conterà este período de tempo. A figura 6.24 ilustra de que forma se dá o mapeamento deste tipo de fluxo para rede de Petri.

Quando se tratar de fluxos que verificam a presença de relacionamentos, o mapeamento para rede de Petri é análogo ao das entidades, como mostram as figuras 6.25 e 6.26.

#### 6.2.2.6 Mapeamento dos fluxos de verificação de ausência

Este tipo de fluxo também não provoca qualquer modificação no estado do banco de dados, apenas verifica se entidades, relacionamentos, ou atributos destas instâncias estão ausentes dos respectivos conjuntos.

O mapeamento dos fluxos de verificação de ausência é análogo ao dos fluxos de verificação de presença. A única diferença é que agora os ramos correspondentes na rede de Petri são do tipo *restauradores de saída*.

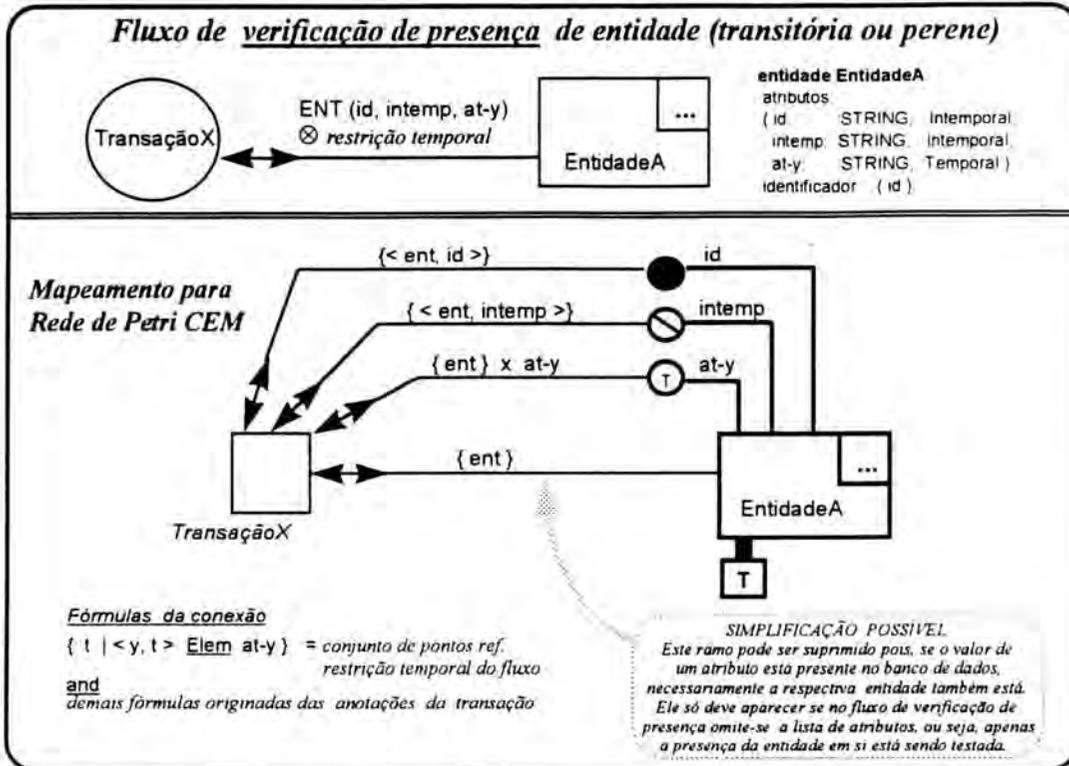


FIGURA 6.23 - Mapeamento do fluxo de verificação de presença de entidade

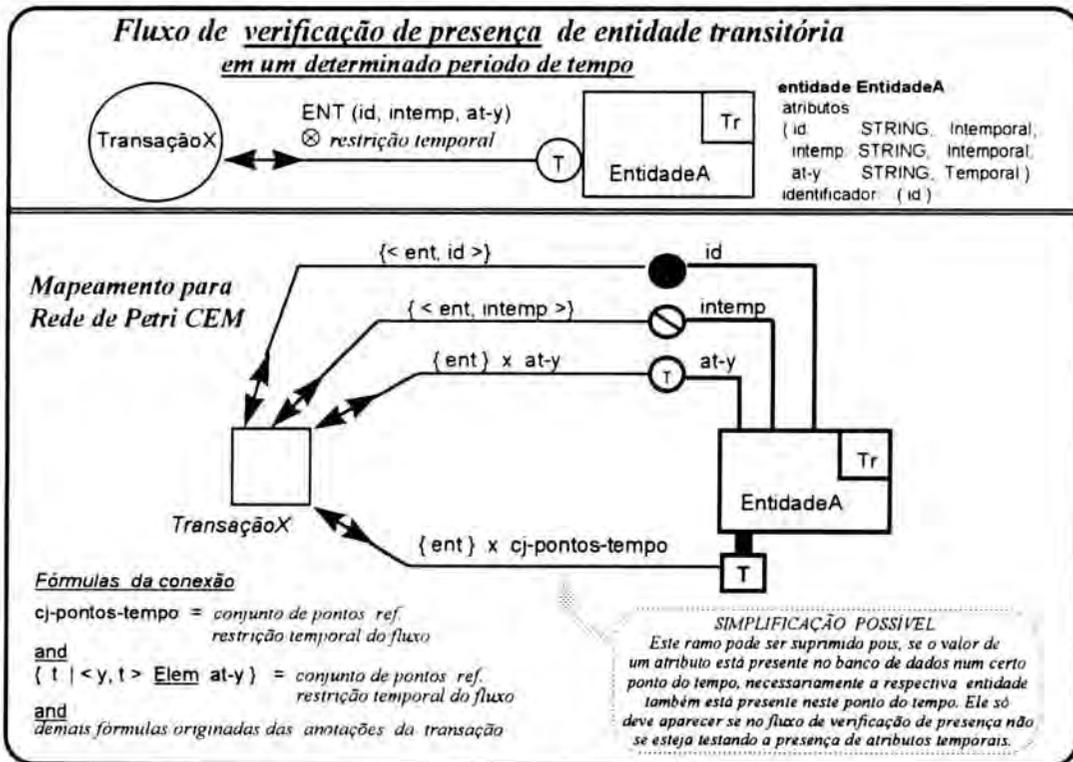


FIGURA 6.24 - Mapeamento do fluxo de verificação de presença de entidade transitória em um determinado periodo de tempo

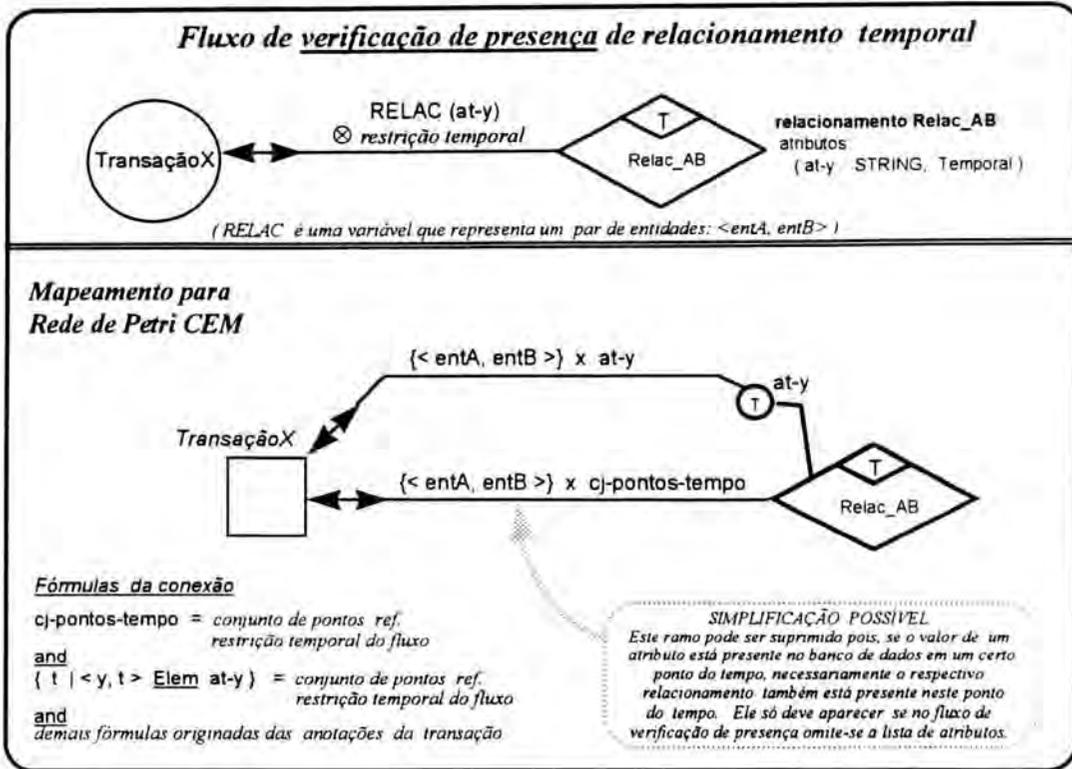


FIGURA 6.25 - Fluxo de verificação de presença de relacionamento temporal

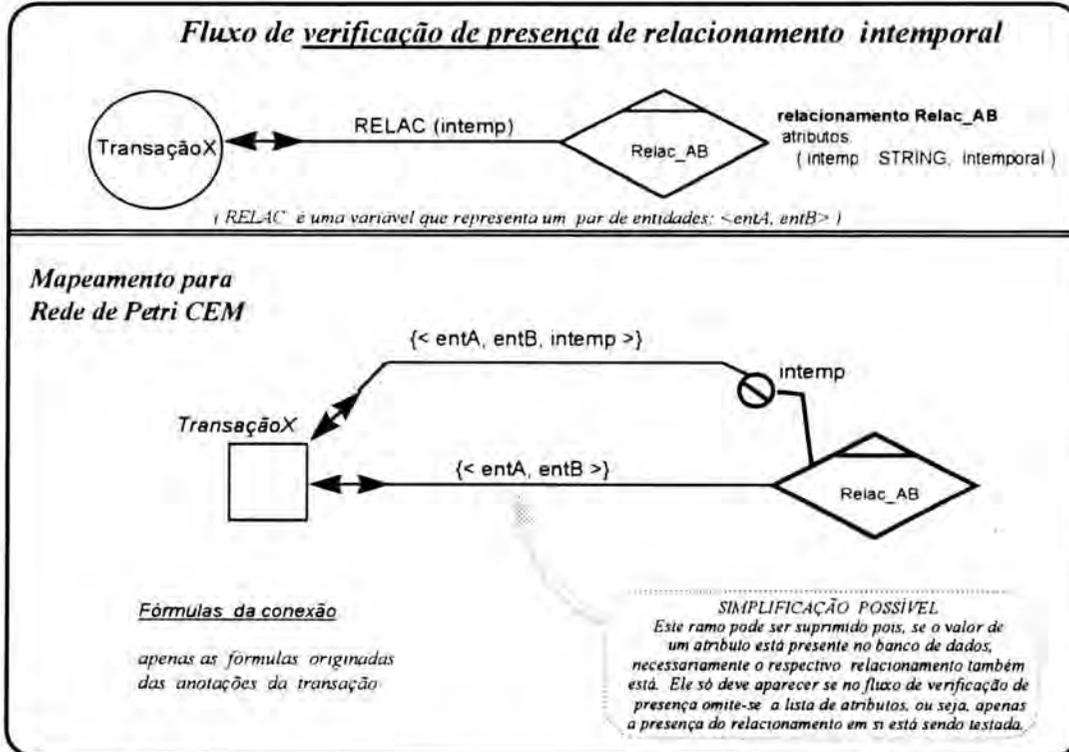


FIGURA 6.26 - Fluxo de verificação de presença de relacionamento intemporal

### 6.2.2.7 Mapeamento de fluxos que manipulam conjuntos de instâncias

Os mapeamentos mostrados acima referem-se a fluxos que manipulam instâncias individualizadas, contudo o modelo TempER-Tr permite a manipulação de conjuntos variáveis de instâncias, operação que é caracterizada pelo *asterisco* que prefixa o nome do pacote de dados associado ao fluxo. Quanto à tradução para rede de Petri, a estrutura de ramos é a mesma, a única diferença está na forma como expressar a anotação dos termos dos ramos. Fluxos que se referem a indivíduos, originam ramos que manipulam indivíduos, onde os termos são variáveis, ou tuplas de variáveis entre chaves. Por sua vez, fluxos que se referem a conjuntos, originam ramos que manipulam conjuntos de indivíduos, onde os termos se caracterizam por variáveis sem as chaves, as quais por convenção devem apresentar o prefixo “cj-“. A figura 6.27 ilustra este tipo de mapeamento, onde a variável \*ENT (do diagrama TempER-Tr) representa um conjunto de instâncias a serem incluídos.

Em relação à rede de Petri da figura 6.27, para que uma alteração (definida pela conexão) esteja habilitada a ocorrer, a regra de obrigatoriedade dos atributos, estabelecida ao nível do modelo ER temporal, deve ser obedecida. Ou seja, as variáveis junto aos ramos devem atender a algumas fórmulas implícitas, que aparecem no canto inferior direito da figura. Seria redundante incluir estas fórmulas no conjunto de fórmulas da conexão, pois as restrições que elas especificam já estão embutidas no modelo de dados.

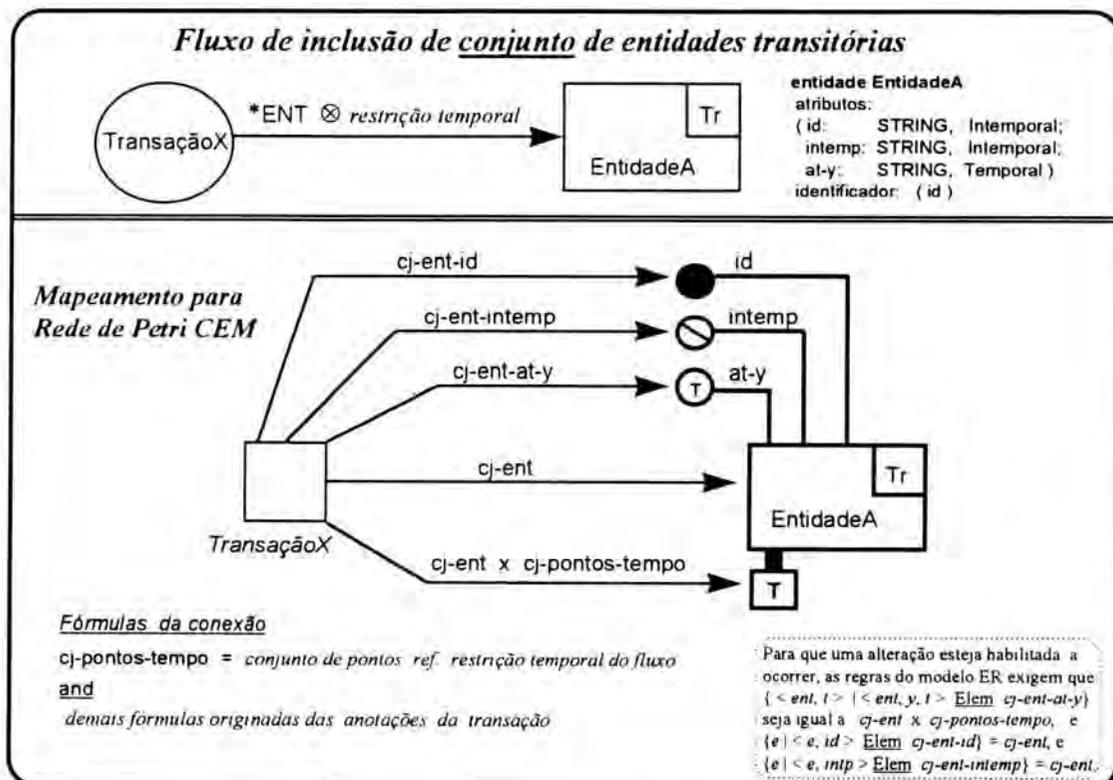


FIGURA 6.27 - Mapeamento de fluxo que manipula um conjunto variável de instâncias

### 6.2.3 Mapeamento das anotações

Cada fórmula que compõe as anotações de uma transação é traduzida para uma fórmula na rede de Petri. A finalidade destas fórmulas é restringir o conjunto de alterações possíveis definidas pela conexão que mapeia a transação. As fórmulas da rede de Petri, assim como os termos dos ramos, seguem a sintaxe e a semântica da Linguagem de Anotação, definida em [HEU 90], a qual se encontra descrita de forma resumida na seção 2.3.2 do capítulo 2.

A figura 6.28, apresenta um exemplo completo de mapeamento de uma transação TempER-Tr para rede de Petri, inclusive mostrando a tradução das anotações.

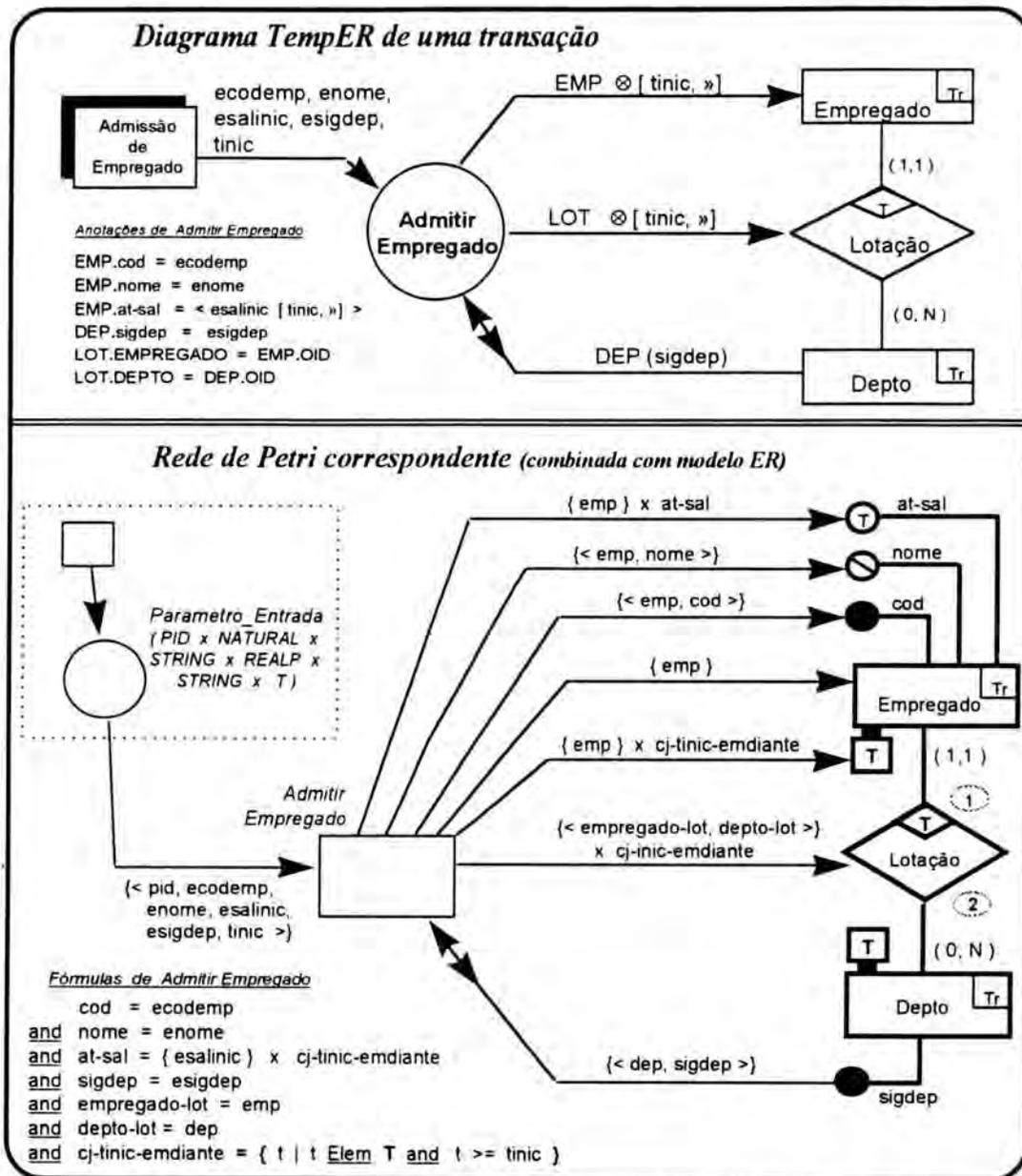


FIGURA 6.28 - Mapeamento completo de uma transação TempER-Tr para rede de Petri.

Os dois pequenos círculos contendo um número cada, que aparecem próximo ao conjunto *Lotação* na figura 6.28 servem para indicar a ordem de participação das entidades nos relacionamentos. No exemplo citado, o primeiro elemento de qualquer instância do conjunto-relacionamento *Lotação* é uma entidade do conjunto *Empregado*, o segundo elemento é uma entidade do conjunto *Depto*.

#### 6.2.4 Simplificação da rede de Petri resultante

É possível simplificar a rede de Petri resultante do mapeamento de uma transação TempER-Tr, principalmente no que diz respeito aos termos dos ramos, através de certas substituições. É o que mostra a figura 6.29 que é exatamente a mesma rede que aparece na figura 6.28. Como pode ser notado, o conjunto de fórmulas reduziu-se significativamente. Isto só foi possível por que se tratava de relações de igualdade entre termos. Por exemplo, a fórmula “cod = ecodemp” da rede da figura 6.28 pôde ser dispensada porque, no ramo ligado ao atributo *cod*, a variável também denominada *cod* foi substituída pela variável *ecodemp*, que é a mesma que aparece no ramo ligado ao lugar referente ao fluxo externo.

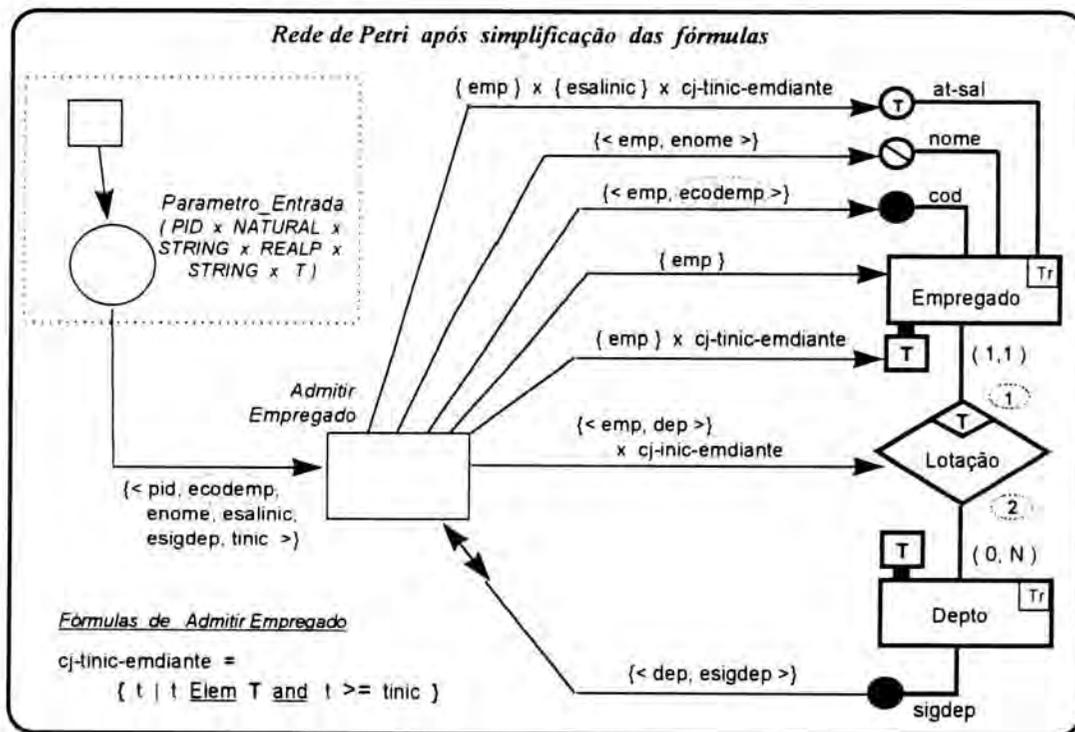


FIGURA 6.29 - Simplificação das fórmulas da rede de Petri resultante de um mapeamento

## 7 Exemplos de modelagem de transações - situações típicas

Neste capítulo são apresentados casos típicos de transações modeladas em TempER-Tr. O objetivo é mostrar, através de exemplos, algumas das possibilidades de uso da ferramenta. As figuras, que modelam as transações, apresentam na parte superior o diagrama TempER-Tr e na parte inferior o correspondente mapeamento para rede de Petri, nos moldes descritos no capítulo anterior.

### 7.1 Aparecimento de entidade transitória

As entidades quando surgem no banco de dados do sistema, adquirem um OID que as identifica, associado a um rótulo temporal que delimita as respectivas validades temporais. Além disto, ao se incluir uma entidade todos os seus atributos também estão sendo incluídos, ou seja, devem receber uma valoração.

A figura 5.1 do capítulo 5 modela, através do fluxo de inclusão conectado ao conjunto-entidade *Empregado*, a inclusão de uma entidade transitória, cuja validade temporal inicia no ponto do tempo indicado pela variável *time* e segue em diante. O mapeamento desta transação para rede de Petri se encontra na figura 6.28 do capítulo 6.

### 7.2 Aparecimento de entidade perene

A inclusão de uma entidade perene é exemplificada pela transação da figura 7.1, onde o cadastramento de uma nova função é modelada. Por se tratar de entidade perene, a sua validade temporal equivale ao conjunto de todos os pontos do eixo temporal. É o que especifica o ramo de inclusão, que está conectado ao lugar que representa a perspectiva temporal do conjunto-entidade *Função*, na rede de Petri correspondente. O termo “ $\{fun\} \times T$ ” associado ao ramo denota o produto cartesiano entre o OID da função sendo incluída e o conjunto  $T$  que representa o eixo temporal.

O lugar que especifica a perspectiva temporal de uma entidade perene, por ser de conteúdo constante, existe apenas ao nível da modelagem conceitual. Na posterior implementação física este lugar fica implícito, não sendo necessário anexar qualquer rótulo temporal à entidade.

A figura 7.1 também mostra como tratar um atributo temporal pertencente a entidade do tipo perene. É o caso do atributo *at-salref* que ao longo do tempo pode vir a apresentar diversos salários de referência. No exemplo apenas um valor está sendo incluído (representado pela variável *esalref*), o qual, até que haja uma alteração, vigora em todo o eixo temporal, conforme especifica o rótulo temporal “[«, »]”.

### 7.3 Eliminação de entidade transitória

Quando uma entidade transitória é eliminada, o seu OID e respectivo rótulo temporal desaparecem. Além disto, todos os valores de seus atributos são excluídos. Ou seja, não fica qualquer vestígio da entidade no banco de dados.

A transação da figura 7.2 modela a eliminação dos dados referentes a um empregado. Como não foi especificada nenhuma restrição temporal nos ramos, esta fica indeterminada, ou seja qualquer período de tempo pode ser assumido, definindo uma série de transformações possíveis. Entretanto apenas uma dessas transformações não fere as regras de integridade do banco de dados. Trata-se daquela cuja valoração engloba todos os pontos do tempo que estão presentes no lugar referente à perspectiva temporal da entidade que está sendo eliminada, bem como engloba todos os relacionamentos temporais em que o empregado participa. É a única transformação que não viola a restrição de integridade estática definida pela conexão morta *cm2* da figura 6.1 (capítulo 6) e não viola a restrição estática que define que se existe um relacionamento em um certo ponto do tempo, as entidades que participam deste relacionamento também existem neste ponto do tempo (vide conexão morta *cm1d* da figura 6.6 [r4] no capítulo 6).

#### 7.4 Encerramento de existência de entidade transitória

A validade temporal de uma entidade transitória nada mais é que a associação desta com um subconjunto de pontos do tempo. Encerrar a existência de uma entidade significa retirar da sua validade temporal o intervalo de tempo que começa a partir de um dado momento e segue em diante. A transação *DemitirEmpregado* da figura 7.3 modela, através do fluxo contendo na sua origem um pequeno círculo com a letra T, a redução da validade temporal de um empregado, representado pela variável *EMP*. Este tipo de fluxo, diferentemente do fluxo de exclusão, afeta apenas a perspectiva temporal de uma entidade, a qual permanece no banco de dados, como pode ser observado na rede de Petri presente na parte inferior da figura. No exemplo mostrado, o empregado cuja chave primária *cod* seja igual ao parâmetro *ecodemp* terá a sua validade temporal encerrada no momento *tdem*. O intervalo posterior:  $[tsuc, \infty)$ , onde *tsuc* é o momento consecutivo a *tdem*, será excluído. A função *SUCCESSOR\_TEMP*, que se encontra na primeira fórmula das anotações da transação, devolve o momento consecutivo em relação a um dado ponto do tempo.

Caso a entidade transitória possua atributos temporais, o encerramento da sua existência provoca o encerramento da validade temporal destes atributos. É o caso do atributo *at-sal* da figura 7.3. Para que a transação esteja habilitada, o assinalamento definido pela variável de mesmo nome, que aparece no ramo conectado a este atributo na rede de Petri, deve ter o seu domínio temporal igual ao conjunto de pontos que está sendo excluído da existência da entidade, ou seja, deve ser igual a  $[tsuc, \infty)$ .

#### 7.5 Reencarnação de entidade transitória

Uma entidade transitória que esteja presente no banco de dados, mas cuja validade temporal se encontra encerrada, pode ser *reencarnada* através de um fluxo de ampliação de existência (aquele que parte do conjunto-transação e aponta para um pequeno círculo contendo a letra T anexo ao conjunto-entidade). É o que mostra a transação *ReadmitirEmpregado* da figura 7.4, onde um novo intervalo de tempo ( $[timic, \infty)$ ) é adicionado à validade temporal do empregado, representado pela variável *EMP*, cuja chave primária *cod* é igual ao parâmetro *ecodemp*.

Quando a validade temporal de uma entidade transitória é estendida, necessariamente os atributos temporais também devem ter o seu domínio temporal

estendido. É o caso do atributo *at-sal*, que recebe uma valoração adicional conforme mostra a terceira fórmula que se encontra nas anotações da transação da figura 7.4.

## 7.6 Inclusão e exclusão de relacionamento temporal

As instâncias de um conjunto-relacionamento temporal necessariamente estão associados a pontos do tempo. A figura 7.5 mostra como estabelecer ou eliminar relacionamentos temporais através do uso de fluxos de inclusão e exclusão. Trata-se de uma transação que transfere um empregado de um departamento para outro. Como a participação de um empregado no conjunto-relacionamento *Lotação* é obrigatória e restringe-se a uma vez no máximo, conforme indica a cardinalidade (1, 1), toda vez que um empregado deixa de pertencer a um departamento, em um determinado momento do tempo, no momento consecutivo deve estar vinculado a um outro departamento. Em outras palavras, não pode haver um momento do tempo sequer, da validade temporal de um empregado, que ele não esteja lotado em um departamento, qualquer que seja este.

A validade temporal do novo departamento, em que o empregado vai ser lotado, deve conter o conjunto de pontos do tempo da validade temporal da nova lotação, ou seja, um empregado só pode ser alocado em um departamento que esteja aberto, do momento da transferência em diante. Esta restrição é modelada pelo fluxo de verificação de presença conectado ao pequeno círculo contendo a letra T, anexo ao conjunto-entidade *Depto*. No mapeamento para rede de Petri, este fluxo corresponde ao ramo restaurador de entrada que está ligado ao lugar que representa a perspectiva temporal do conjunto-entidade *Depto*, como mostra a figura 7.5. Na verdade este fluxo de verificação de presença precisaria testar apenas a presença do OID do departamento, sem se preocupar com a sua validade temporal, à semelhança da transação *ReadmitirEmpregado*, modelada na figura 7.4. Isto porque já existe uma restrição estática ao nível do modelo de dados que impede que se inclua um relacionamento temporal fora da validade temporal das entidades associadas.

## 7.7 Alteração de atributo temporal de entidade

A alteração de atributos temporais de uma entidade, seja ela transitória ou perene, é modelada por um fluxo de alteração, conforme mostra a figura 7.6. Na transação modelada, um empregado cuja chave primária *cod* seja igual ao parâmetro *ecodemp* terá uma parte do assinalamento temporal do atributo *at-sal* alterado, apenas aquela que se situa na faixa de tempo indicada pelo intervalo [*taum*, »]. A alteração é uma multiplicação de cada valor anterior do salário pelo parâmetro *eindice*. Esta operação encontra-se explicada na subseção 5.3.3.5 do capítulo 5.

## 7.8 Alteração de atributo de relacionamento temporal

Como já foi explicado no capítulo 4, seção 4.4.3, um relacionamento temporal apresenta apenas atributos temporais. Portanto, a alteração de atributos de relacionamentos temporais é análoga à alteração de atributos temporais de entidades. Supondo que o conjunto-relacionamento *Lotação* possuísse o atributo *at-tarefa*, uma alteração do conteúdo deste atributo se daria na forma mostrada na figura 7.7. A finalidade deste atributo seria o de armazenar a tarefa principal que um empregado estaria desempenhando em um certo período do tempo, quando estivesse lotado em um

departamento. Por ser do tipo temporal, o atributo *at-tarefa* tem a capacidade de armazenar a evolução das tarefas que uma lotação possa apresentar ao longo tempo.

## 7.9 Alteração de atributo intemporal de entidade

Um atributo intemporal existe apenas na perspectiva intemporal das entidades, sejam elas transitórias ou perenes. A figura 7.8 mostra como modificar o conteúdo deste tipo de atributo. A transação mostrada no exemplo modela a alteração de um atributo identificador (chave-primária), que por definição sempre deve ser do tipo intemporal. Quando se altera o conteúdo de um atributo intemporal o valor anterior desaparece do banco de dados, sendo substituído pelo novo. Não fica nenhum histórico.

## 7.10 Manipulação de conjuntos

Os exemplos mostrados até agora neste capítulo lidam apenas com entidades individuais. Entretanto é possível modelar a manipulação de conjuntos variáveis de entidades (ou relacionamentos). Uma transação deste tipo já foi mostrada no capítulo 5, na figura 5.7. No presente capítulo, uma variante daquela transação se encontra modelada na figura 7.9, trata-se de um aumento salarial concedido a partir de um determinado ponto do tempo (*taum*) a todos os empregados que estiveram lotados durante um certo período ( $[ta, tb]$ ) no departamento cuja sigla foi informada como parâmetro de entrada (*esigdep*).

No exemplo modelado aparece a cláusula *maxsub*, associada ao fluxo de verificação de presença conectado ao conjunto-relacionamento *Lotação*. Esta cláusula, que também aparece no mapeamento da transação para rede de Petri que está na parte de baixo da mesma figura, encontra-se fundamentada em [HEU 92] e é descrita na seção 2.3.4 do capítulo 2. Um fluxo de verificação de presença, ou um fluxo de inclusão, contendo a cláusula *maxsub* aponta para todas as instâncias que estejam *presentes* no banco dados em relação a um determinado conjunto de referência. Sendo assim, para que a transação da figura 7.9 esteja habilitada a ocorrer, entre outras condições, é necessário que o conjunto \*LOTSELEC seja igual a todos os relacionamentos, especificados pelo termo \*LOT  $\otimes$   $\square$   $[ta, tb]$  que estejam presentes no conjunto-relacionamento *Lotação*. Em outras palavras, o conjunto \*LOTSELEC deve se referir a todos os empregados que estão lotados durante o período  $[ta, tb]$  no departamento cuja chave primária *sigdep* é igual ao parâmetro *esigdep*.

A constante *ALL-OID* que se encontra na terceira fórmula das anotações da transação representa todo o conjunto de OID's do Universo do Discurso. Disto depreende-se que o conjunto de referência especificado pelo termo \*LOT  $\otimes$   $\square$   $[ta, tb]$  é uma combinação entre o departamento selecionado e todos os empregados possíveis. Não necessariamente todo este conjunto de referência está presente no banco de dados. Por isto há a necessidade de uma forma de se especificar apenas o conjunto presente. E isto se dá através da cláusula *maxsub*.

Na figura 7.10 é mostrada a mesma rede de Petri da figura 7.9 após o processo de simplificação das fórmulas. É o único exemplo neste capítulo onde se mostra os dois estágios da rede de Petri. Nas demais figuras, optou-se por apresentar apenas um dos estágios, em geral o que se refere à rede já simplificada.

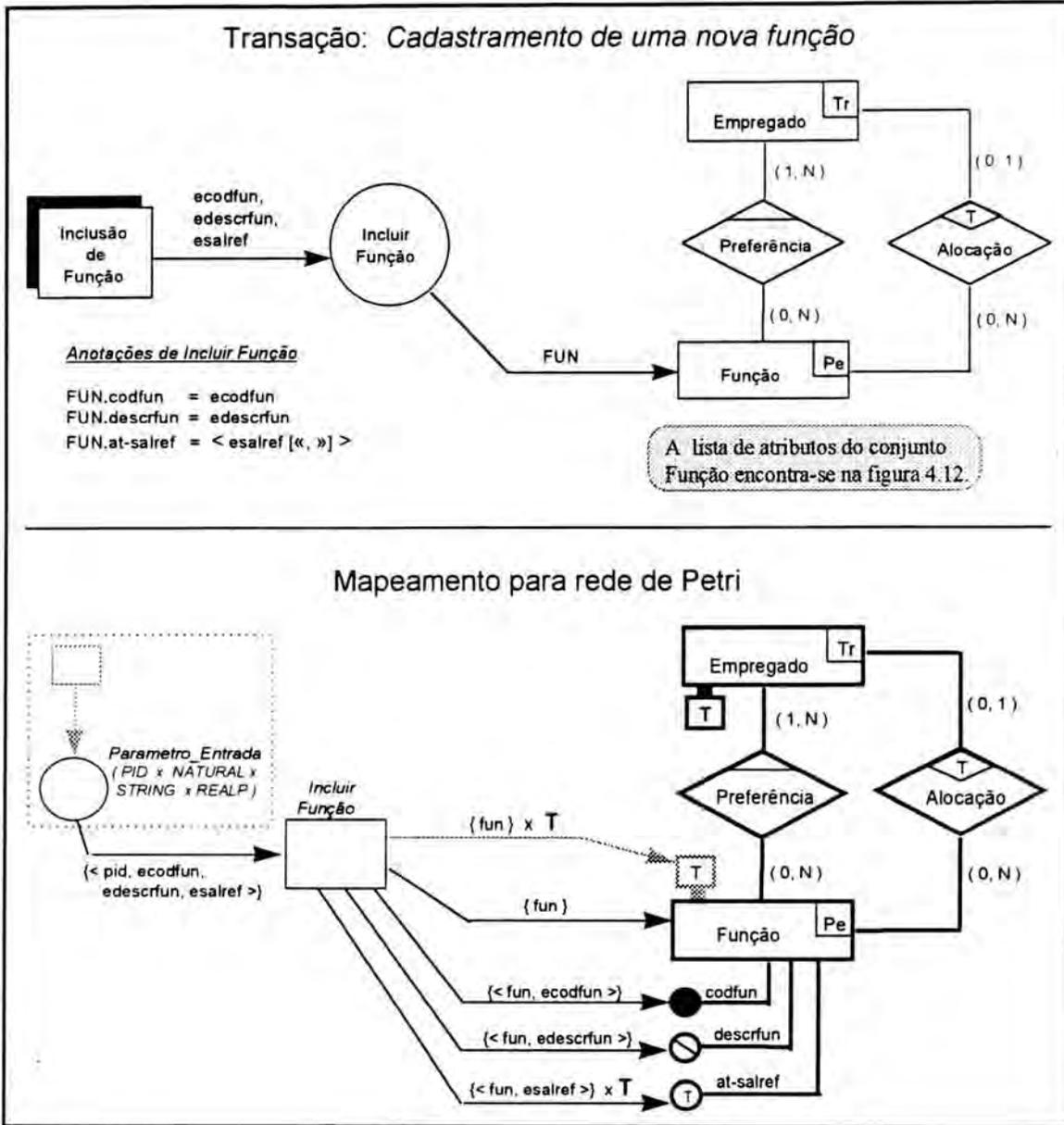


FIGURA 7.1 - Modelando o aparecimento de uma entidade perene

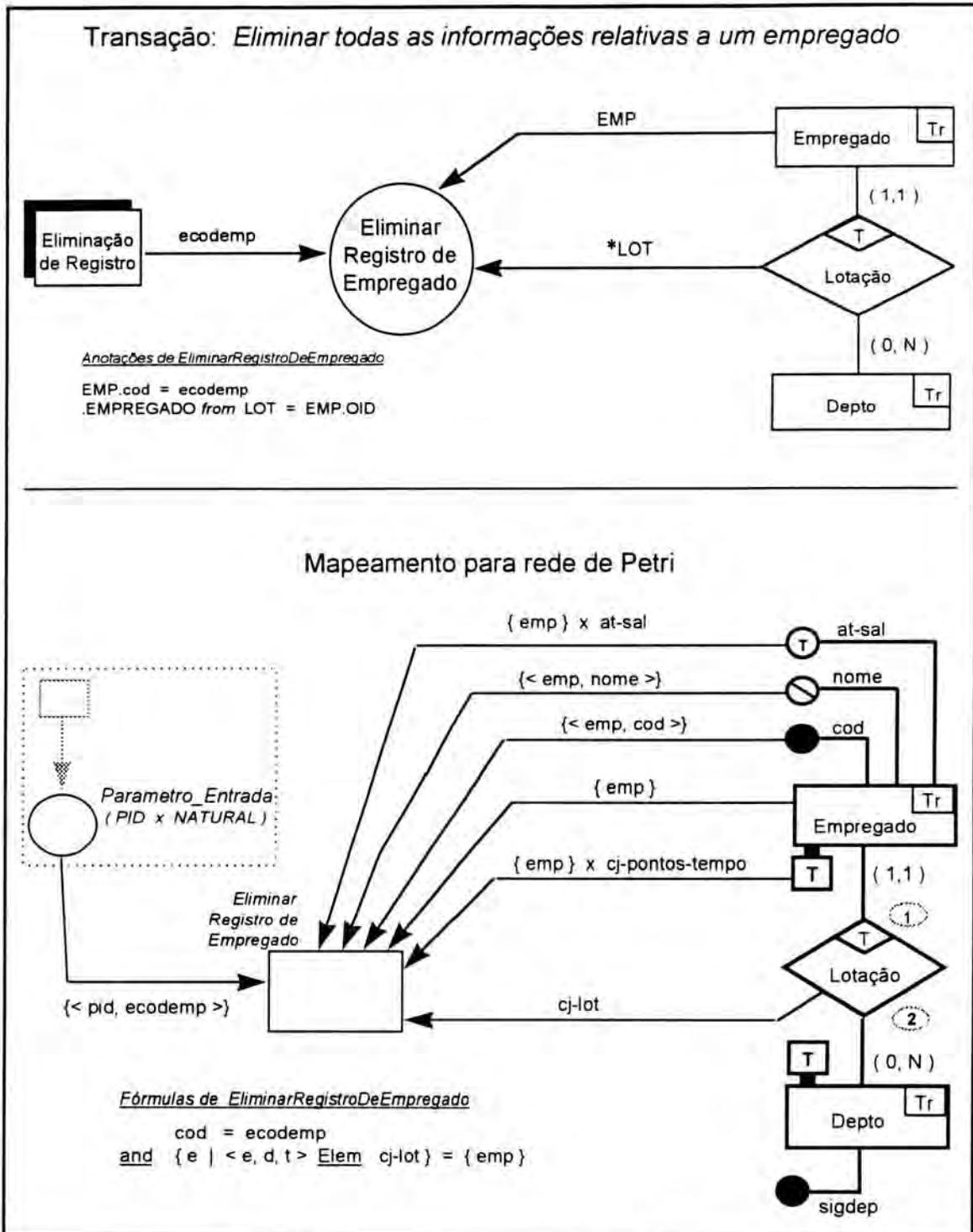


FIGURA 7.2 - Modelando a eliminação de entidade transitória

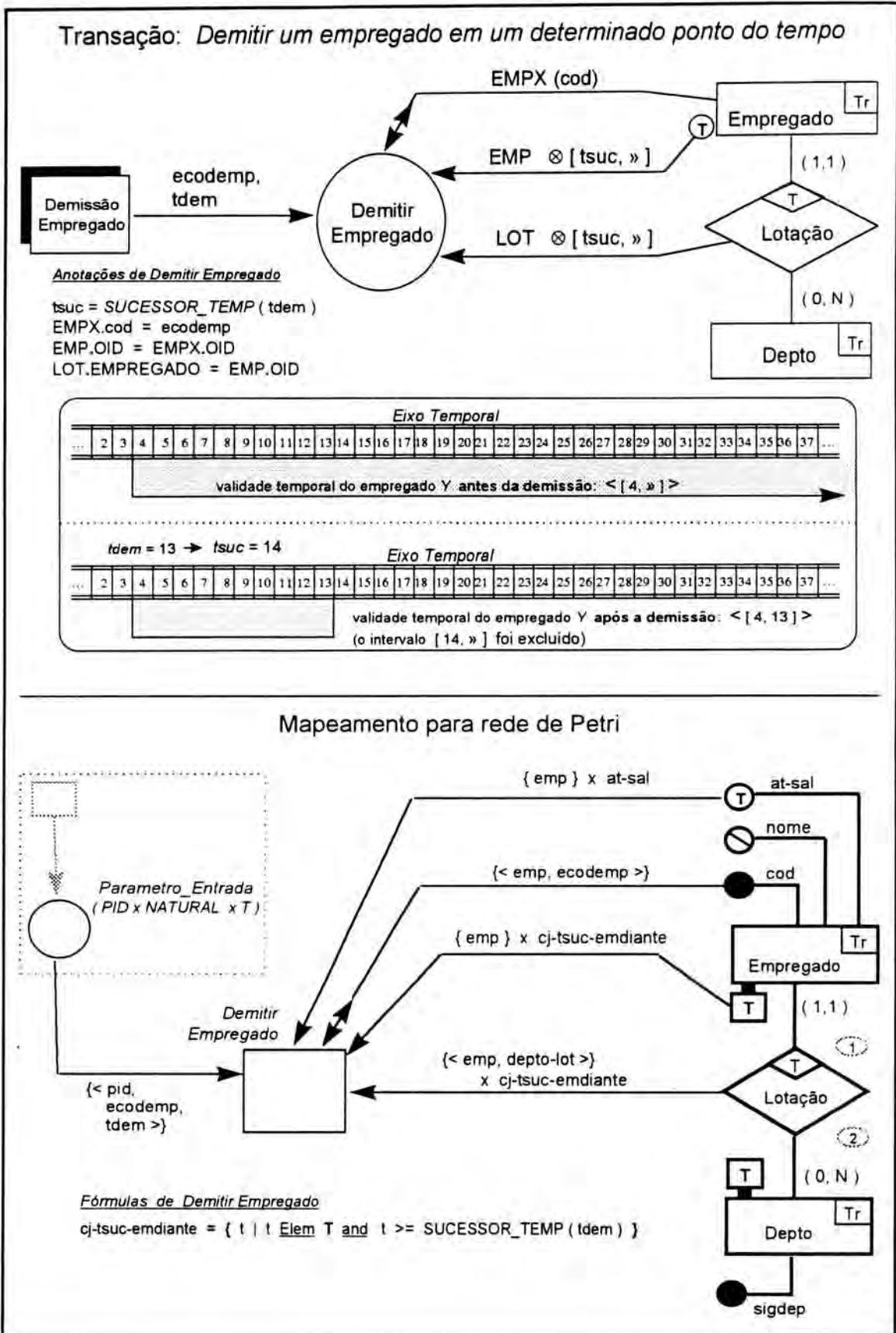


FIGURA 7.3 - Modelando o encerramento de existência de entidade transitória

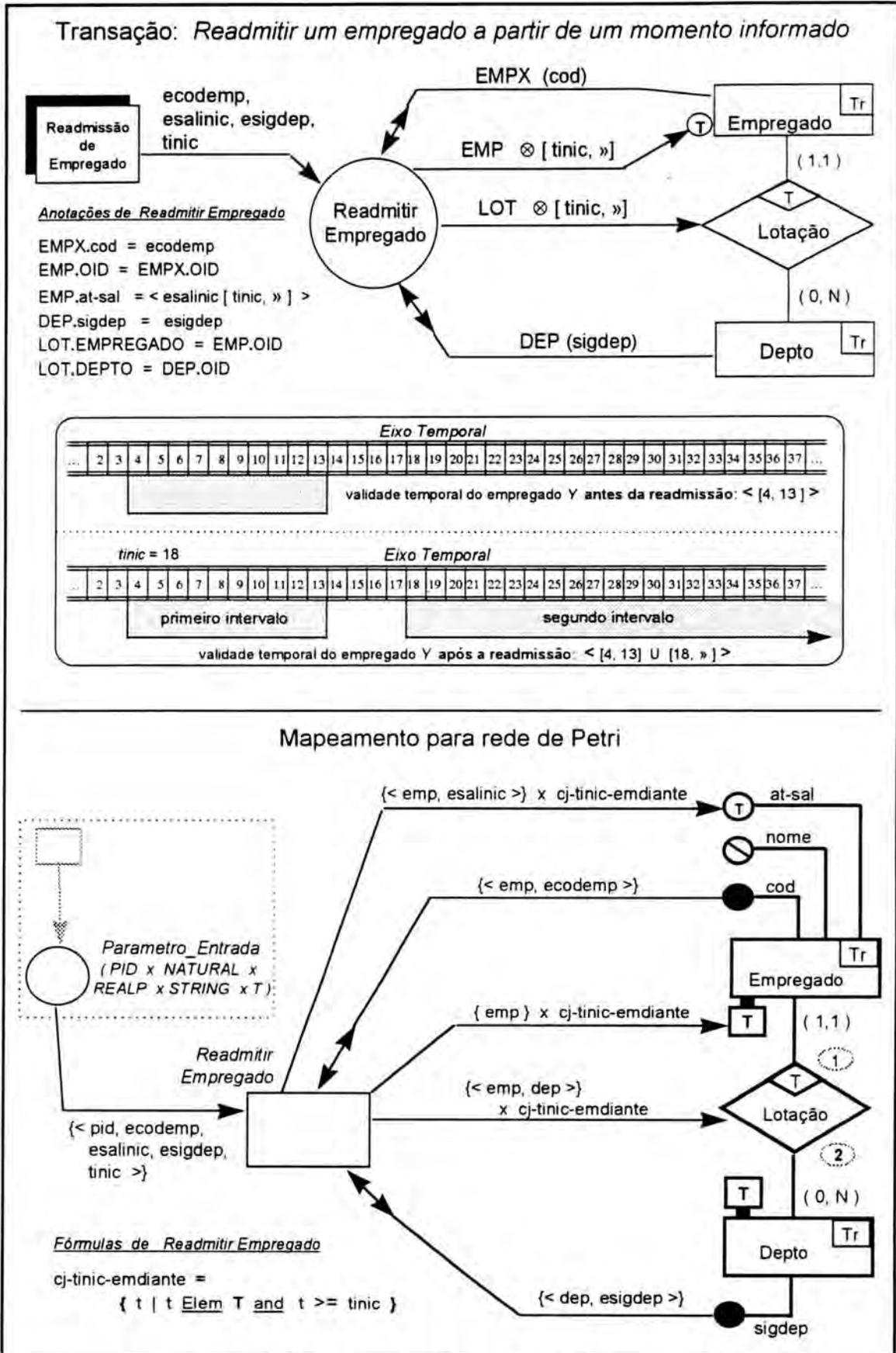


FIGURA 7.4 - Modelando a reencarnação de entidade transitória

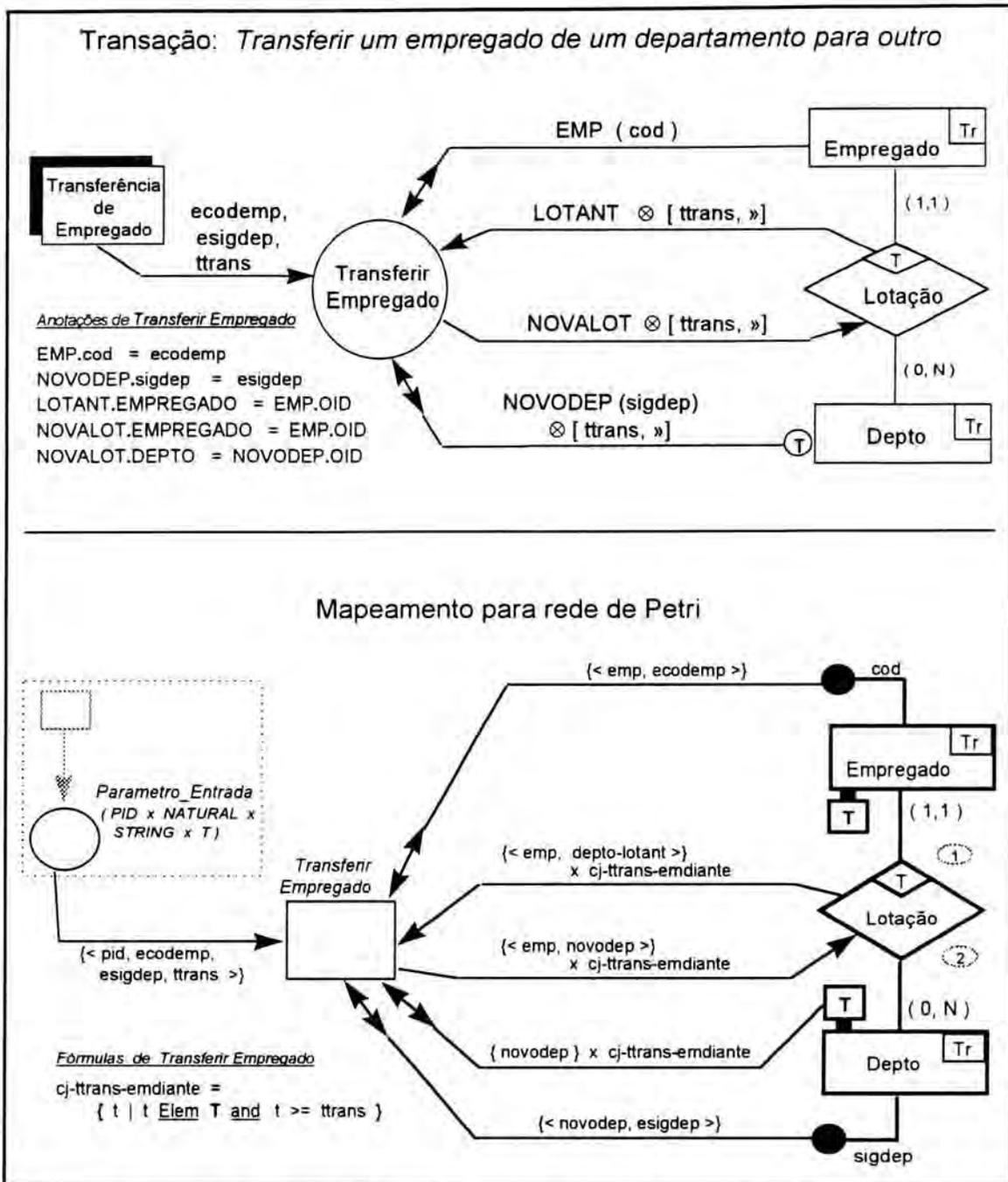


FIGURA 7.5 - Modelando a inclusão e exclusão de relacionamento temporal

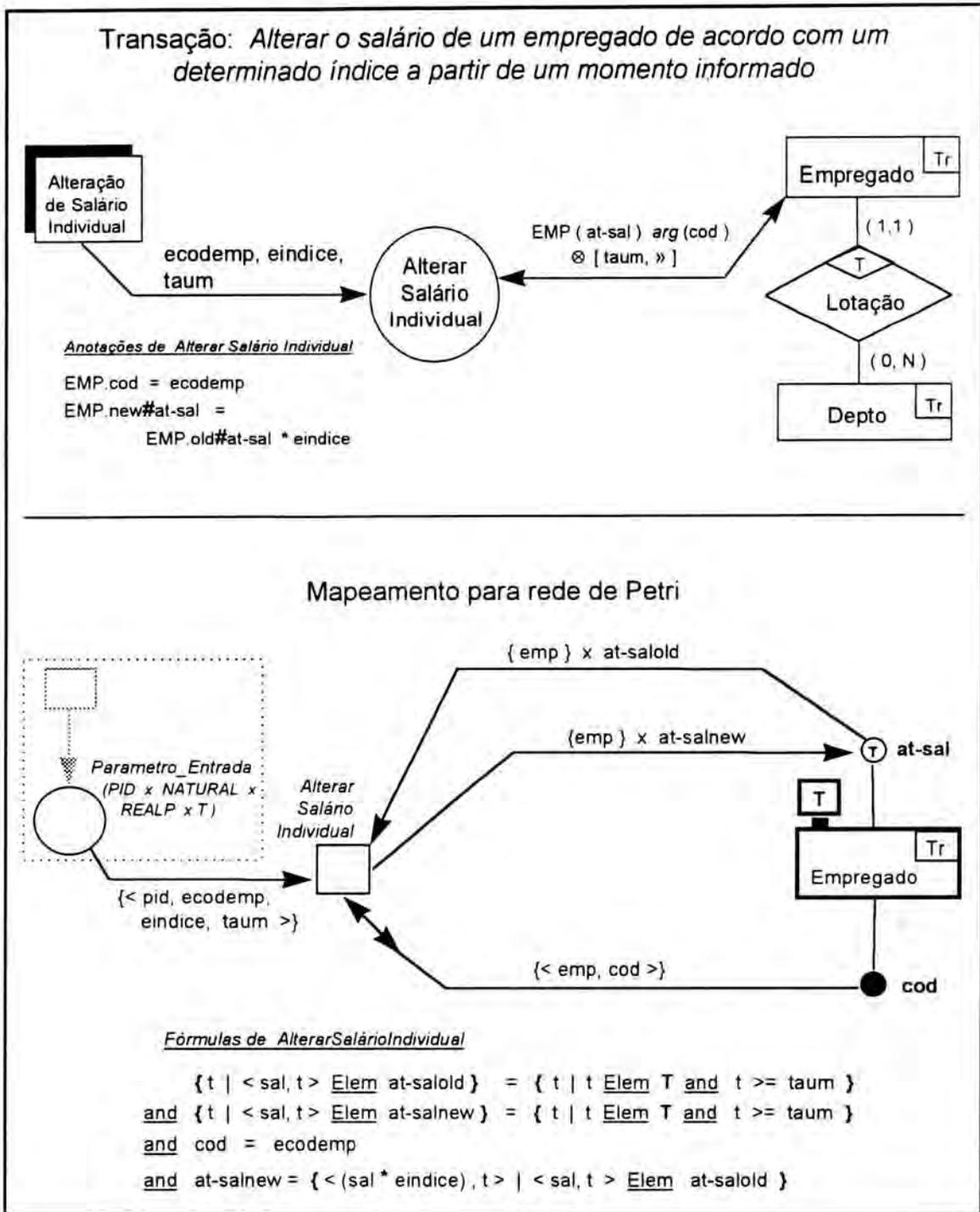


FIGURA 7.6 - Modelando a alteração de um atributo temporal de entidade

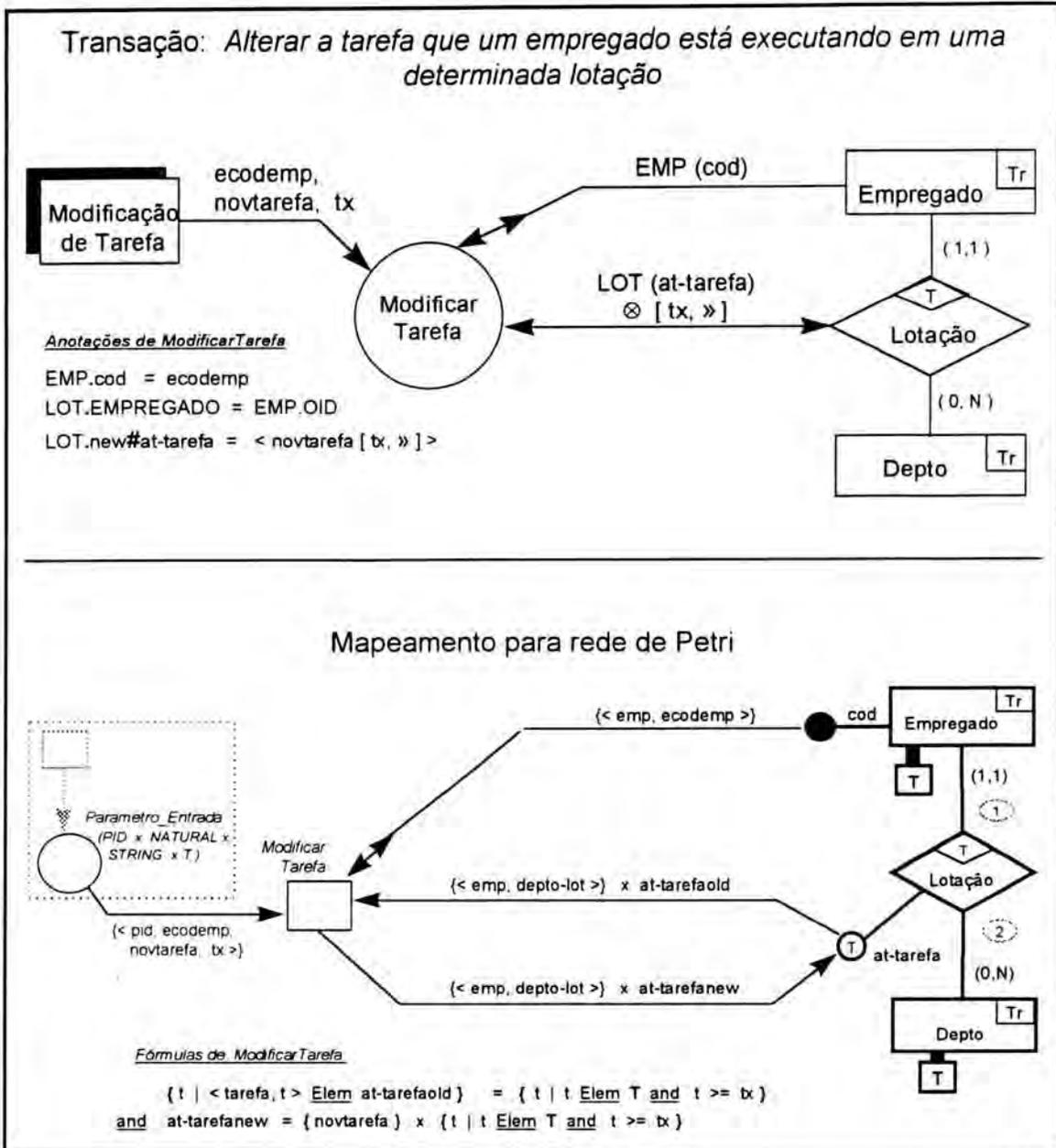


FIGURA 7.7 - Modelando a alteração de atributo de relacionamento temporal

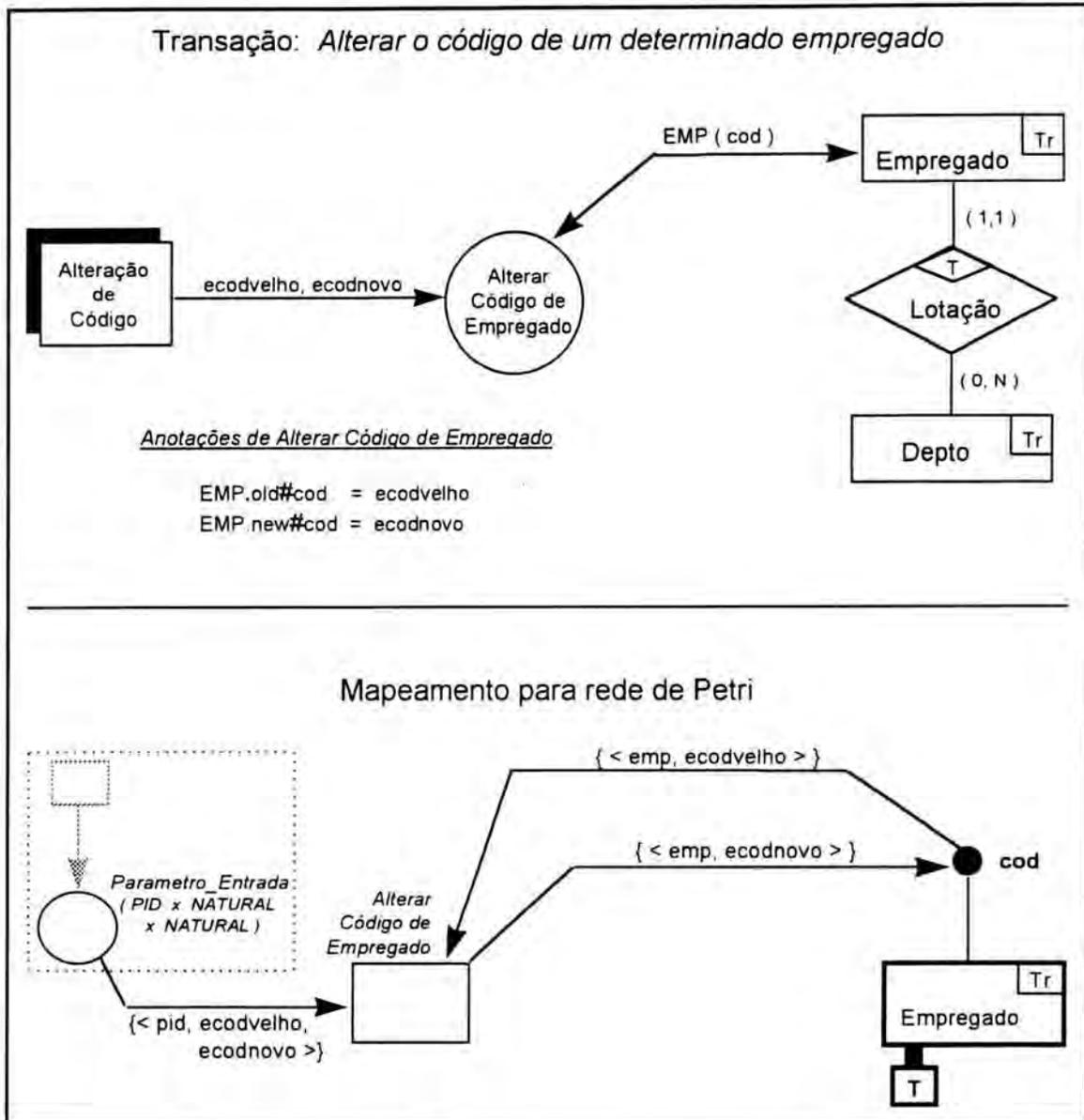


FIGURA 7.8 - Modelando a alteração de atributo intemporal de entidade

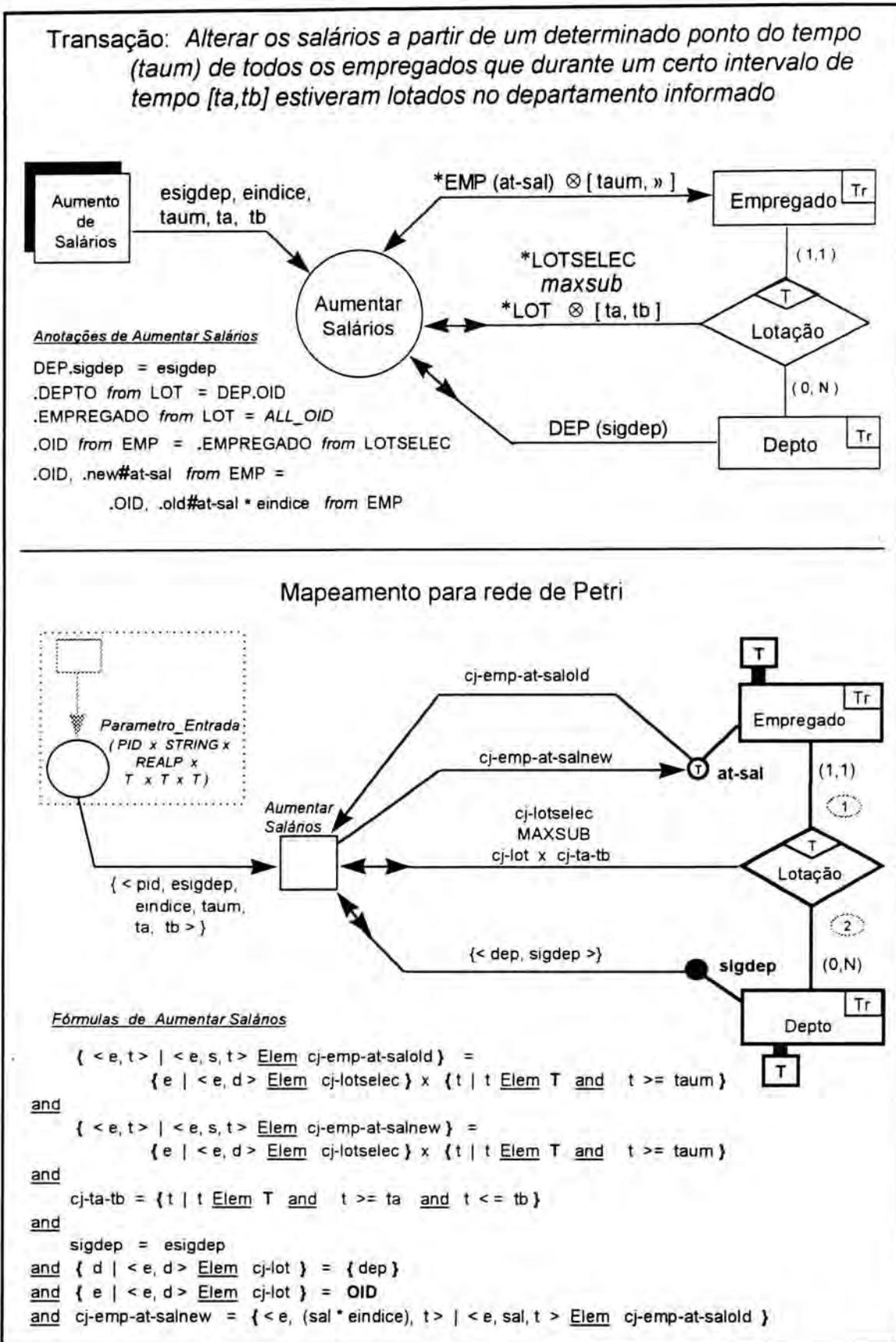


FIGURA 7.9 - Modelando a manipulação de conjuntos nos fluxos

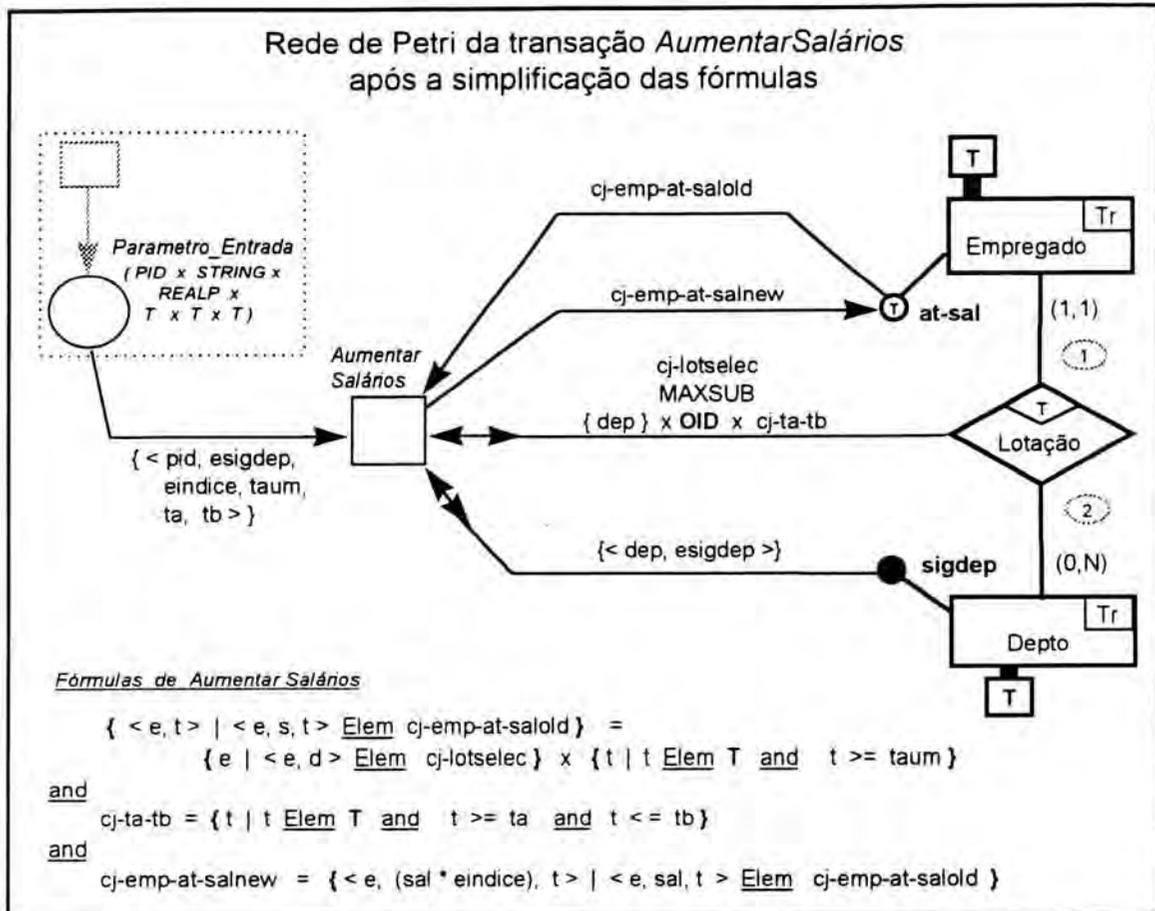


FIGURA 7.10 - Rede de Petri da figura 7.9 após a simplificação das fórmulas

## 8 Conclusões finais

O modelo TempER-Tr proposto neste trabalho é uma extensão do modelo ER-Tr [HEU 90, PER 90, HEU 91], em dois sentidos. Primeiro, para modelar as propriedades estáticas concebeu-se um modelo Entidade-Relacionamento temporal (TempER). Segundo, a linguagem de anotação do modelo ER-Tr foi substituída por outra capaz de especificar conjuntos de indivíduos de maneira mais consistente e formal, baseada na declaração Select na linguagem SQL.

### 8.1 Preenchendo requisitos da modelagem conceitual

À semelhança do modelo ER-Tr, também o modelo TempER-Tr é uma ferramenta que integra em um único diagrama a especificação das propriedades estáticas e dinâmicas de um sistema, cumprindo com isto um dos requisitos para modelagem conceitual estabelecidos em [GRI 82].

O modelo TempER-Tr é completamente traduzível para uma rede de Petri de alto nível do tipo condição/evento, mais precisamente para o modelo ER/PN [HEU 90, HEU 93], o qual nada mais é que uma rede de Petri cujos lugares e conexões mortas representando as propriedades estáticas, foram substituídos por um modelo ER. Desta forma, a semântica do modelo TempER-Tr é especificada formalmente através da rede de Petri CEM, conforme foi mostrado no capítulo 6.

O modelo TempER-Tr, por ser na sua essência uma rede de Petri CEM, herda desta o indeterminismo em relação às transações conflitantes e a característica de não estabelecer qualquer seqüência entre transações concorrentes. Em virtude disto, também atende àquele outro requisito que define que um modelo conceitual não deve conter detalhes de implementação do sistema, a não ser aqueles inerentes aos requerimentos formulados pelos usuários [GRI 82].

### 8.2 Extensão temporal do modelo ER

A concepção do modelo de dados temporal TempER, utilizado na proposta TempER-Tr, inspirou-se principalmente no modelo ERT [LOU 91] e em uma proposta da área de banco de dados temporais [GAD 88, 93]. A vantagem de se utilizar um modelo de dados temporal, no lugar de um convencional, está na sua capacidade de expressar a associação dos elementos com o tempo e de especificar as restrições decorrentes disto. Nos modelos de dados convencionais o tempo é representado por atributos comuns (datas, horas, etc) e não estão disponíveis mecanismos para representar restrições temporais, exigindo, por consequência, que o modelador as especifique ao nível do modelo dinâmico. O exemplo da figura 4.1 mostra as diferenças de notação entre um diagrama TempER e um diagrama ER convencional.

A adoção do modelo de dados temporal TempER impacta de forma positiva na modelagem dinâmica, quando se compara com a abordagem ER-Tr. Por exemplo, a transação *AdmitirEmpregado* modelada em ER-Tr na figura 2.11 mostra a necessidade de se incluir fórmulas adicionais para tratar atributos do tipo data, pois o modelo de dados adotado é do tipo convencional. Por outro lado, a mesma transação modelada em

TempER-Tr dispensa tais fórmulas, pois agora estão disponíveis facilidades que permitem a abordagem da dimensão temporal, como mostra a figura 6.28.

A dimensão temporal no modelo de dados TempER é assumida como sendo um conjunto implícito e constante de pontos, isomórfico ao conjunto dos números inteiros, denominado eixo temporal. As entidades associam-se a este eixo temporal de duas formas possíveis: de forma transitória ou de forma perene. Tanto as entidades perenes quanto as entidades transitórias, apresentam duas perspectivas em relação ao tempo: uma intemporal e uma temporal, como se fossem duas faces de uma mesma moeda (ver figuras 4.4, 4.7 e 4.9). Quando se focaliza os conjuntos de entidades pela perspectiva intemporal estes apresentam apenas duas dimensões (tuplas x atributos intemporais). Por outro lado, quando se focaliza estes mesmos conjuntos pela perspectiva temporal eles passam a apresentar três dimensões (tuplas x atributos temporais x eixo temporal). Quanto aos relacionamentos, ou as entidades se associam entre si na perspectiva temporal (relacionamentos temporais) ou na perspectiva intemporal (relacionamentos intemporais).

Especificamente em relação à abordagem ERT [LOU 91] (a que mais se aproxima do modelo TempER) existem as seguintes diferenças, que visam suprimir os aspectos considerados desfavoráveis:

- A primitiva temporal utilizada no modelo TempER é o *elemento temporal* e não o intervalo de tempo. O elemento temporal, por ser fechado para as operações de união, interseção e complementação da teoria dos conjuntos, simplifica substancialmente a possibilidade de se expressar consultas temporais.
- No modelo TempER os atributos não são explicitados graficamente e sim através de um dicionário de dados associado ao diagrama ER, o que resulta em um modelo mais administrável visualmente.

### 8.3 Aspectos a serem aperfeiçoados

O modelo TempER-Tr, por ser uma proposta em caráter experimental, demanda uma série de aperfeiçoamentos, entre os quais, alguns que foram sendo identificados ao longo dos estudos que culminaram no presente texto.

Os diagramas produzidos em TempER-Tr são *flats*, voltados para uma abordagem *bottom-up*. Para sistemas com poucas transações isto não se configura em um problema. Entretanto, antes aplicar a abordagem TempER-Tr na modelagem de sistemas de grande porte, há a necessidade de dotá-lo da capacidade de *clusterização*, ou seja, de visualizar as funcionalidades em refinamentos sucessivos à semelhança das abordagens estruturadas. O desafio reside em propor tal capacidade sem perder de vista o formalismo da rede de Petri que sustenta o modelo TempER-Tr.

Para efeito de simplificação das pesquisas foi assumido que todos os objetos de um sistema apresentam uma única granularidade temporal. Entretanto, em alguns sistemas pode haver a necessidade de tratar diversas granularidades. Por exemplo, em um sistema podem conviver objetos cuja existência deve ser medida em horas, outros em dias e outros em meses. Para que o modelo TempER-Tr possa manipular diferentes granularidades de tempo vai ser necessário propor uma cláusula adicional, ao nível do

dicionário de dados, associada aos objetos. Além disto, será preciso definir algumas funções e regras para possibilitar a conversão de uma granularidade para outra, bem como o mapeamento do eixo temporal para o calendário.

Ao nível do modelo de dados faz-se necessário um tratamento mais adequado dos fluxos que manipulam relacionamentos recursivos. Como se encontra definido no capítulo 5, as colunas referentes aos OID's das entidades contidas nos pacotes de dados "movimentados" por fluxos ligados a conjuntos-relacionamento levam o nome da própria entidade. Em se tratando de relacionamento recursivo seria necessário qualificar tais colunas, adicionando a denominação do papel desempenhado pelas entidades no relacionamento ao nome da coluna.

A notação da estrutura generalização/especialização (descrita na seção 4.6 do capítulo 4) ficou muito "carregada". Seria necessário repensar a simbologia utilizada, mantendo, contudo, a possibilidade de diferenciar especialização temporal de especialização intemporal.

A cláusula MAXSUB da linguagem de anotação do modelo TempER-Tr (ver figura 7.9), utilizada para referenciar todos os objetos presentes (ou ausentes) em determinado conjunto de entidades ou de relacionamentos, necessita de uma lapidação que facilite a sua compreensão. Esta cláusula tem origem da rede de Petri CEM (ver seção 2.3.4 do capítulo 2) e foi adotada em um formato muito próximo ao que lá está definido.

## 8.4 Estudos futuros

Explorar o formalismo que fundamenta o modelo TempER-Tr constitui-se em um caminho promissor de pesquisa, no sentido de estabelecer um conjunto de regras e procedimentos de mapeamento que permitam produzir, a partir dos diagramas conceituais de um sistema, o primeiro nível do projeto de implementação.

Uma outra vertente de pesquisa reside em estudar e propor uma forma de mapear o modelo de dados temporal utilizado em TempER-Tr para um banco de dados relacional. Primeiramente, para um banco de dados relacional convencional, em seguida para um banco de dados relacional temporal. Como resultado de tal estudo esperar-se-ia um conjunto de regras e de procedimentos que permitiriam transformar um esquema conceitual de dados em um projeto lógico de banco de dados.

O modelo de dados temporal TempER, como foi demonstrado neste trabalho, é traduzível completamente para uma rede de Petri de alto nível, sendo que as restrições, principalmente as de cardinalidade, são representadas por conexões mortas. Explorar a capacidade das conexões mortas de representar as propriedades estáticas, visando especificar restrições adicionais vinculadas à dimensão temporal, é também uma vertente de estudo que pode render alguns frutos. Poderiam haver notações especiais para representar restrições do tipo "um empregado só pode chefiar um departamento durante o período em que nele estiver lotado", "uma pessoa não pode ao mesmo tempo fiscalizar e dirigir uma empresa", "o salário de um empregado não pode ser reduzido", etc.

No modelo TempER-Tr, o tempo foi introduzido apenas como uma propriedade do modelo estático. Os eventos temporais que podem provocar o disparo de transações no interior do sistema não foram abordados. Normalmente estes eventos são gerados

pelo relógio do sistema. A modelagem deste relógio em TempER-Tr pode ser a mesma encontrada na abordagem ER-Tr [HEU 90, PER 90, HEU 91], a qual fundamenta-se, por sua vez, no estudo publicado em [RIC 85]. Contudo, são necessários estudos mais aprofundados a respeito do significado destes eventos temporais e dos impactos que podem provocar sobre o modelo dinâmico de um sistema.

Um outro aspecto importante a ser explorado diz respeito aos eventos retroativos. Supondo que, em 10/Nov/96 é assinada uma portaria designando um empregado como gerente financeiro, e que esta designação tem efeito a partir do mês anterior, mais precisamente a partir de 01/Out/96. Caso o sistema tivesse que responder quem era o gerente em 20/Out/96, para, por exemplo, apurar a responsabilidade de algum ato irregular, existiriam duas possibilidades, uma levando em conta o efeito retroativo do evento e outra não. É como se o empregado tivesse dois passados a serem considerados. Este tipo de evento, bem como os eventos com efeito futuro, são estudados em [CHA 93].

Por fim, para identificar outros aspectos a serem aperfeiçoados ou modificados seria necessário experimentar o modelo TempER-Tr em casos reais.

## Bibliografia

- [ALE 85] ALENCAR FILHO, Edgard de. **Teoria elementar dos conjuntos**. 20.ed. São Paulo: Nobel, 1985.
- [ALL 83] ALLEN, J. F. Maintaining knowledge about temporal intervals. **Communications of the ACM**, New York, v.26, n.11, p.832-843, Nov.1983.
- [BAT 92] BATINI, Carlo; CERI, Stefano; NAVATHE, Shamkant B. **Conceptual Database Design - an Entity-Relationship Approach**. Redwood City: The Benjamin/Cummings Publishing Company, 1992.
- [BRA 87] BRAUER, W. (Ed.). **Petri Nets - central models and their properties**. Berlin-Heidelberg: Springer, 1987. Part I. (Lecture Notes in Computer Science, v. 254).
- [CHA 93] CHAKRAVARTHY, Sharma; KIM, Seung-Kyum. Semantics of the time-varying information and resolution of time concepts in temporal databases. In: INTERNATIONAL WORKSHOP ON AN INFRASTRUCTURE FOR TEMPORAL DATABASES, 1993, Arlington. **Proceedings...** Tucson: University of Arizona, 1993.
- [CHE 76] CHEN, Peter S. The entity-relationship model - toward a unified view of data. **ACM Transactions on Database Systems**, New York, v.1, n.1, p.9-28, Mar.1976.
- [CLI 88] CLIFFORD, J.; RAO, A. A simple, general structure for temporal domains. In: ROLLAND, C.; BODART, F.; LEONARD, M. (Eds.). **Temporal Aspects in Information Systems**. Amsterdam: North-Holland, 1988. p.17-28.
- [CLI 88a] CLIFFORD, J.; CROKER, A. Objects in time. **Data Engineering**, Washington, v.11, n.4, p.11-18, Dec.1988.
- [CLI 95] CLIFFORD, J.; TUZHILIN, A. (Eds.). **Recent Advances in Temporal Databases**. Berlin: Springer-Verlag, 1995.
- [DAT 93] DATE, C. J.; DARWEN, H. **A guide to the SQL standard**. 3.ed. Reading, Massachusetts: Addison-Wesley, 1993.
- [EDE 94] EDELWEISS, Nina; OLIVEIRA, José Palazzo M. **Modelagem de aspectos temporais de sistemas de informação**. Recife: Universidade Federal de Pernambuco, 1994. Livro texto da Escola de Computação, 9., 1994, Recife.

- [EDE 95] EDELWEISS, Nina; OLIVEIRA, José Palazzo M.; CASTILHO, José Mauro V. Evolução de esquemas em banco de dados temporais. In: CONFERÊNCIA LATINO-AMERICANA DE INFORMÁTICA. 21., 1995, Canela, RS. **Anais...** Porto Alegre: Instituto de Informática da UFRGS, 1995.
- [ELM 92] ELMASRI, Ramez; KOURAMAJIAN, Vram. A temporal query language based on conceptual entities and roles. In: INTERNACIONAL CONFERENCE ON THE ENTITY RELATIONSHIP APPROACH, 11., 1992, Karlsruhe, Germany. **Proceedings...** Berlin: Springer Verlag, 1992. p.375-388. (Lecture Notes in Computer Science, v.645).
- [ELM 93] ELMASRI, Ramez; WUU, Gene T. J.; KOURAMAJIAN, Vram. A temporal model and query language for EER Databases. In: TANSEL, A. et al. (Eds.). **Temporal databases: theory, design and implementation.** Redwood City: The Benjamin/Cummings Publishing, 1993. p. 212-229.
- [FER 91] FERG, Stephen. Cardinality Concepts in Entity-Relationship Modeling. In: INTERNACIONAL CONFERENCE ON THE ENTITY RELATIONSHIP APPROACH, 10., 1991, San Mateo, California. **Proceedings...** [S.l.: s.n.], 1991.
- [GAD 88] GADIA, Shashi. A homogeneous relational model and query language for temporal databases. **ACM Transactions on Database Systems**, New York, v.13, n.4, p.418-448, Dec.1988.
- [GAD 93] GADIA, Shashi; NAIR, Sunil. Temporal databases: a prelude to parametric data. In: TANSEL, A. et al. (Eds.). **Temporal databases: theory, design and implementation.** Redwood City: The Benjamin/Cummings Publishing, 1993. p. 28-66.
- [GAN 79] GANE, C.; SARSON, T. **Structured Systems Analysis: Tools and Techniques.** Englewood Cliffs: Prentice-Hall, 1979.
- [GEN 87] GENRICH, H. Predicate/Transition nets. In: BRAUER, W. (Ed.). **Petri Nets - central models and their properties.** Berlin-Heidelberg: Springer, 1987. Part I, p.207-241. (Lecture Notes in Computer Science, v. 254).
- [GRI 82] GRIETHUYSEN, J. (Ed.). **Concepts and terminology for the conceptual schema and the information base.** ANSI, New York, 1982. Publication Number ISO/TC97/SC5-N695.
- [HEU 90] HEUSER, Carlos A. **Modelagem conceitual de sistemas: redes de Petri.** Buenos Aires: Ed. Kapelusz, 1990.

- [HEU 91] HEUSER, Carlos A.; PERES, Eduardo M. ER-T diagrams: an approach to specifying database transactions. In: INTERNACIONAL CONFERENCE ON THE ENTITY RELATIONSHIP APPROACH, 10., 1991, San Mateo, California. **Proceedings...** [S.l.: s.n.], 1991.
- [HEU 92] HEUSER, Carlos A.; RICHTER, Gernot. Constructs for modeling information systems with Petri nets. In: JENSEN, K. (Ed.). **Application and Theory of Petri Nets**. Berlin: Springer-Verlag, 1992. p.224-243. (Lecture Notes in Computer Science, v. 616).
- [HEU 93] HEUSER, Carlos A.; PERES, Eduardo M.; RICHTER, Gernot. Towards a complete conceptual model: Petri nets and entity-relationship diagrams. **Information Systems**, Oxford, v.18, n.5, p. 275-298, 1993.
- [HEU 9?] HEUSER, Carlos A.; PERES, Eduardo M. **ER-Tr diagrams**: a visual formalism for the specification of database transactions. Porto Alegre: Instituto de Informática da UFRGS, [199?].
- [JCS 94] JENSEN, Christian S. (Ed.). A consensus glossary of temporal database concepts. **ACM SIGMOD Record**, New York, v.23, n.1, p. 52-64, Mar.94.
- [JEN 87] JENSEN, K. Coloured Petri Nets. In: BRAUER, W. (Ed.). **Petri Nets - central models and their properties**. Berlin-Heidelberg: Springer, 1987. Part I, p.248-252. (Lecture Notes in Computer Science, v. 254).
- [KAP 89] KAPPEL, G.; SCHREFL, M. A behavior integrated entity-relationship approach for the design of object-oriented databases. In: **Entity-relationship approach**. Amsterdam: North-Holland, 1989. p.311-318.
- [KHO 86] KHOSHAFIAN, Setrag N.; COPELAND, George P. Object Identity. **ACM SIGPLAN Notices**, New York, v. 21, n. 11, p. 214-223, Sept.1986. Trabalho apresentado na Conference on Object-Oriented Programming Systems, Languages and Applications, OOPSLA, 1986, Portland.
- [KUN 89] KUNG, C. H. Conceptual modeling in the context of software development. **IEEE Transactions on Software Engineering**, New York, v.15, n.10, p.1176-1187, Oct.1989.
- [LOU 90] LOUCOPOULOS, P et al. TEMPORA - Integrating database technology rule based systems and temporal reasoning for effective software. In: ESPRIT CONFERENCE, 1990, Brussels, Bélgica. **Proceedings...** [S.l.: s.n.], 1990.

- [LOU 91] LOUCOPOULOS, P.; THEODOULIDIS, C.; WANGLER, B. The entity relationship time model and conceptual rule language. In: INTERNACIONAL CONFERENCE ON THE ENTITY RELATIONSHIP APPROACH, 10., 1991, San Mateo, California. **Proceedings...** [S.l.: s.n.], 1991.
- [LOU 91a] LOUCOPOULOS, P.; THEODOULIDIS, C.; WANGLER, B. A conceptual modelling formalism for temporal database applications. **Information Systems**, Oxford, v.16, n.4, p.401-416, 1991.
- [LOU 91b] LOUCOPOULOS, P.; THEODOULIDIS, C.; PANTAZIS, D. Business rules modelling: conceptual modelling and object-oriented specifications. In: **Object Oriented Approach in Information Systems**. Amsterdam: North-Holland, 1991. p.323-342.
- [McM 90] McMENAMIN, Stephen M.; PALMER, John F. **Análise essencial de sistemas**. São Paulo: McGraw-Hill, 1990.
- [OZS 95] ÖZSOYOGLU, G.; SNODGRASS, R. T. Temporal and real-time databases: a survey. **IEEE Transactions on Knowledge and Data Engineering**, New York, v.7, n.4, p.513-532, Aug.1995.
- [PER 90] PERES, Eduardo M. **Abordagem ER/T: uma proposta para a integração da modelagem conceitual de propriedades estáticas e dinâmicas de sistemas de informação**. Porto Alegre: CPGCC da UFRGS, 1990. Dissertação de Mestrado.
- [PET 81] PETERSON, J. **Petri net theory and the modeling of systems**. Englewood Cliffs: Prentice-Hall, 1981.
- [REI 85] REISIG, W. **Petri nets: an introduction**. Berlin-Heidelberg: Springer, 1985.
- [RIC 85] RICHTER, G. Clocks and their use for time modeling. In: IFIP WG 8.1 WORKING CONFERENCE ON THEORETICAL AND FORMAL ASPECTS OF INFORMATION SYSTEMS, 1985, Barcelona. **Proceedings...** Amsterdam: North-Holland, 1985. p.49-66.
- [SEG 88] SEGEV, A.; SHOSCHANI, A. Modeling temporal semantics. In: ROLLAND, C.; BODART, F.; LEONARD, M. (Eds.). **Temporal Aspects in Information Systems**. Amsterdam: North-Holland, 1988. p.47-57.
- [SEG 93] SEGEV, A.; SHOSCHANI, A. A temporal data model based on time sequences. In: TANSEL, A. et al. (Eds.). **Temporal databases: theory, design and implementation**. Redwood City: The Benjamin/Cummings Publishing Company, Inc., 1993. p. 248-270.
- [SNO 94] SNODGRASS, Richard T. et al. TSQL2 language specification. **ACM SIGMOD Record**, New York, v.23, n.1, p.65-86, Mar.1994.

- [TAN 93] TANSEL, A. et al. (Eds.). **Temporal databases: theory, design and implementation.** Redwood City: The Benjamin/Cummings Publishing, 1993.
- [TAU 91] TAUZOVIĆ, Branka. Towards temporal extensions to the entity-relationship model. In: INTERNACIONAL CONFERENCE ON THE ENTITY RELATIONSHIP APPROACH, 10., 1991, San Mateo, California. **Proceedings...** [S.l.: s.n.], 1991.
- [WUU 93] WUU, Gene T. J.; DAYAL, Umeshwar. A uniform model for temporal and versioned object-oriented databases. In: TANSEL, A. et al. (Eds.). **Temporal databases: theory, design and implementation.** Redwood City: The Benjamin/Cummings Publishing, 1993. p.230-247.

**Informática**



**UFRGS**

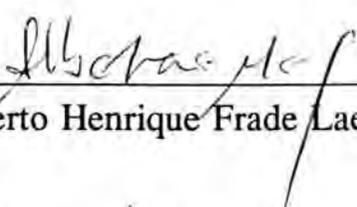
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

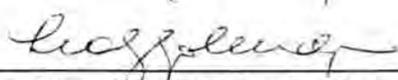
*Modelagem Temporal de Sistemas: uma Abordagem Fundamentada em Redes de Petri*

por

Dante Carlos Antunes

Dissertação apresentada aos Senhores:

  
\_\_\_\_\_  
Prof. Dr. Alberto Henrique Frade Laender (UFMG)

  
\_\_\_\_\_  
Profa. Dra. Lia Goldstein Golendziner

  
\_\_\_\_\_  
Profa. Dra. Nina Edelweiss

Vista e permitida a impressão.

Porto Alegre, 03 / 04 / 97.

  
\_\_\_\_\_  
Prof. Dr. Carlos Alberto Heuser,  
Orientador.

  
\_\_\_\_\_

Prof. Flávio Reich Wagner  
Coordenador do Curso de Pós-Graduação  
em Ciência da Computação (PG-11)  
Instituto de Informática - UFRGS