

GLADYS DEIFAN BASTIDAS GUSTIN

**MODELAGEM E ANÁLISE DE SISTEMAS DISTRIBUÍDOS E
ABERTOS PARA AUTOMAÇÃO PREDIAL**

Tese apresentada à Escola
Politécnica da Universidade de São
Paulo para a obtenção do título de
Doutor em Engenharia.

São Paulo
2005

GLADYS DEIFAN BASTIDAS GUSTIN

**MODELAGEM E ANÁLISE DE SISTEMAS DISTRIBUÍDOS E
ABERTOS PARA AUTOMAÇÃO PREDIAL**

Tese apresentada à Escola
Politécnica da Universidade de São
Paulo para a obtenção do título de
Doutor em Engenharia.

Área de Concentração:
Engenharia de Controle e Automação
Mecânica

Orientador:
Prof. Dr. Paulo Eigi Miyagi

São Paulo
2005

Ao meu esposo Pablo, pelo amor, carinho, compreensão, ajuda e apoio em todo momento.

Aos meus pais Gilberto e Nery e aos meus irmãos Mauricio, Ronal, Jairo e Sergio por estar sempre comigo dando-me seu amor e apoio.

AGRADECIMENTOS

Expresso minha gratidão sem limites:

A Deus, o Menino Jesús e a Nossa Senhora.

Ao meu orientador Prof. Dr. Paulo Eigi Miyagi, pela orientação, diretrizes, incentivo, confiança, paciência e oportunidade que me concedeu para o desenvolvimento desta pesquisa e por todo o apoio e a colaboração prestada durante estes anos.

Ao Prof. Dr. Diolino Dos Santos Filho, pela colaboração, apoio, sugestões e diretrizes levantadas ao longo do período de pesquisa.

Ao Dr. Cláudio Luiz Marte, pelas valiosas sugestões e diretrizes levantadas no exame de qualificação.

Ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), pelo apoio financeiro deste trabalho.

Ao pessoal do HC-FMUSP, pela colaboração para a realização desta pesquisa.

Aos todos os meus amigos que estão e que tem passado pela sala MC – 1, pois com sua ajuda, sugestões, discussões, carinho e estímulo tornaram possível este trabalho.

A todos meus amigos e familiares, pela sua ajuda, apoio e carinho.

Agradeço em especial ao Pablo e ao Jairo, que estiveram do meu lado em todos os momentos com carinho, amor, compreensão e incentivo.

A todos os que direta ou indiretamente contribuíram para a realização deste trabalho.

RESUMO

O crescimento na demanda de sistemas prediais eficientes tem contribuído para a consolidação do conceito de edifício inteligente, no qual se procura a integração entre seus diferentes sistemas através da incorporação de novas tecnologias de modo a maximizar a produtividade dos seus ocupantes e minimizar os custos envolvidos. Entretanto, do ponto de vista prático existem ainda dificuldades para atingir plenamente estes objetivos. Neste contexto, propõe-se uma abordagem para a modelagem e análise de sistemas distribuídos e abertos para automação predial que permita a integração efetiva dos sistemas prediais em edifícios inteligentes. Em função da base técnica já desenvolvida e, do potencial de modelagem e análise da rede de Petri e da orientação a objetos, o trabalho está baseado na utilização de técnicas derivadas destas abordagens dentro do Padrão de Processamento Distribuído e Aberto (Open Distributed Processing–ODP). A abordagem proposta é ilustrada através de exemplos de aplicação em um prédio hospitalar.

Palavras chaves: Edifícios inteligentes, Sistemas Distribuídos e Abertos, Modelagem de Sistemas, Análise de Sistemas, Redes de Petri, ODP, UML.

ABSTRACT

The increasing demand of more efficient buildings systems has caused a growth in the complexity of the design as well as in the management of building automation. In this context and considering that open distributed systems could provide solution for many users problems related to aspects like flexibility, investment preservation, connectivity and interoperability with other systems, this paper introduce an open distributed systems approach for building automation. This approach is bases on the synergetic association among ODP (Open Distributed Processing), UML (Unified Modelling Language) and Petri nets

Key Words: Intelligent Buildings, Open Distributed Systems, Modeling of system, Analyze of Systems, Petri Nets, ODP and UML.

SUMARIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

LISTAS DE SÍMBOLOS

1. INTRODUÇÃO	1
1.1. JUSTIFICATIVA E MOTIVAÇÕES	1
1.2. OBJETIVO	5
1.3. ORGANIZAÇÃO DO TEXTO	6
2. EDIFÍCIOS INTELIGENTES E SISTEMAS DISTRIBUÍDOS E ABERTOS	8
2.1. EDIFÍCIOS INTELIGENTES	8
2.1.1. DEFINIÇÃO	9
2.1.2. CARACTERÍSTICAS	11
2.1.3. SISTEMAS EM UM EDIFÍCIO INTELIGENTE	12
2.1.4. INTEGRAÇÃO DE SISTEMAS EM EDIFÍCIOS INTELIGENTES	13
2.2. SISTEMAS DISTRIBUÍDOS	16
2.3. SISTEMAS ABERTOS	17
2.3.1. SISTEMA DE CONTROLE DE ARQUITETURA ABERTA	19
2.4. SISTEMAS DISTRIBUÍDOS E ABERTOS	20
2.5. OPEN DISTRIBUTED PROCESSING (ODP)	21
2.5.1. ESTRUTURA DA NORMALIZAÇÃO DO ODP	23
2.5.2. PONTOS DE VISTA	24
2.5.3. TRANSPARÊNCIAS E FUNÇÕES	34

2.5.3. TRANSPARÊNCIAS E FUNÇÕES	34
2.5.4. NOTAÇÃO DO ODP	35
2.6. ORIENTAÇÃO A OBJETOS E UNIFIED MODELLING LANGUAGE (UML)	35
2.6.1. ORIENTAÇÃO A OBJETOS	36
2.6.2. UNIFIED MODELLING LANGUAGE (UML)	41
2.6.3. RELAÇÃO ENTRE UML E ODP	44
2.7. CONSIDERAÇÕES DO CAPÍTULO	44
<u>3. MODELAGEM E ANÁLISE DE SISTEMAS DISTRIBUÍDOS E ABERTOS</u>	<u>46</u>
3.1. ESCOLHA DO FORMALISMO	46
3.2. REDE DE PETRI MODULAR COLORIDA BASEADA EM OBJETOS (RPMCO)	50
3.2.1. ODP E RPMCO	55
3.2.2. CARACTERÍSTICAS DA RPMCO	56
3.2.3. ESTRUTURA DE UM MÓDULO EM RPMCO	57
3.2.4. DEFINIÇÃO DE RPMCO	60
3.2.5. ANÁLISE DA RPMCO	66
3.3. CONSIDERAÇÕES DO CAPÍTULO	74
<u>4. PROCEDIMENTO PARA A MODELAGEM E ANÁLISE DE SISTEMAS DISTRIBUÍDOS E ABERTOS PARA AUTOMAÇÃO PREDIAL</u>	<u>76</u>
4.1. PROCEDIMENTO DE MODELAGEM E ANÁLISE	78
4.1.1. ETAPA 1: MODELAGEM DO DOMÍNIO DO SISTEMA PREDIAL	80
4.1.2. ETAPA 2: MODELAGEM DOS RELACIONAMENTOS E SERVIÇOS DO SISTEMA PREDIAL	85
4.1.3. ETAPA 3: MODELAGEM DINÂMICA E ANÁLISE DOS MODELOS	87
4.2. CONSIDERAÇÕES DO CAPÍTULO	93

5. EXEMPLO DE APLICAÇÃO	94
5.1. COMPLEXO HOSPITALAR DAS CLÍNICAS	95
5.2. PRÉDIO DOS AMBULATÓRIOS – PAMB	97
5.2.1. CARACTERÍSTICAS DOS USUÁRIOS	97
5.2.2. CARACTERÍSTICAS FÍSICAS	98
5.2.3. CARACTERÍSTICAS DOS SISTEMAS PREDIAIS INSTALADOS	100
5.2.4. CONSIDERAÇÕES ESPECÍFICAS SOBRE O PAMB	105
5.3. PROCEDIMENTO APLICADO NO SISTEMA DE AR CONDICIONADO – SAC	110
5.3.1. ETAPA 1: MODELAGEM DO DOMÍNIO DO SISTEMA	110
5.3.2. ETAPA 2: MODELAGEM DOS RELACIONAMENTOS E SERVIÇOS DO SISTEMA	114
5.3.3. ETAPA 3: MODELAGEM DINÂMICA E ANÁLISE FORMAL DOS MODELOS	118
5.4. PROCEDIMENTO APLICADO NO SISTEMA DE CONTROLE DE ACESSO – SCA	123
5.4.1. ETAPA 1: MODELAGEM DO DOMÍNIO DO SISTEMA	123
5.4.2. ETAPA 2: MODELAGEM DOS RELACIONAMENTOS E SERVIÇOS DO SISTEMA	125
5.4.3. ETAPA 3: MODELAGEM DINÂMICA E ANÁLISE DOS MODELOS	128
5.5. PROCEDIMENTO APLICADO NO SISTEMA DE GERENCIAMENTO DE ENERGIA SGE	133
5.5.1. ETAPA 1: MODELAGEM DO DOMÍNIO DO SISTEMA	133
5.5.2. ETAPA 2: MODELAGEM DOS RELACIONAMENTOS E SERVIÇOS DO SISTEMA	134
5.5.3. ETAPA 3: MODELAGEM DINÂMICA E ANÁLISE FORMAL DOS MODELOS	137
5.6. PROCEDIMENTO APLICADO NO SISTEMA DE ILUMINAÇÃO – SI	140
5.6.1. ETAPA 1: MODELAGEM DO DOMÍNIO DO SISTEMA	140
5.6.2. ETAPA 2: MODELAGEM DOS RELACIONAMENTOS E SERVIÇOS DO SISTEMA	142
5.6.3. ETAPA 3: MODELAGEM DINÂMICA E ANÁLISE FORMAL DOS MODELOS	144
5.7. PROCEDIMENTO APLICADO NO SISTEMA DE DETECÇÃO E COMBATE A INCÊNDIO – SDCI	147
5.7.1. ETAPA 1: MODELAGEM DO DOMÍNIO DO SISTEMA	147

5.7.2. ETAPA 2: MODELAGEM DOS RELACIONAMENTOS E SERVIÇOS DO SISTEMA _____	149
5.7.3. ETAPA 3: MODELAGEM DINÂMICA E ANÁLISE FORMAL DOS MODELOS _____	153
5.8. CONSIDERAÇÕES DO CAPÍTULO _____	163
6. CONCLUSÃO	164
ANEXO A	167
ANEXO B	174
REFERÊNCIAS BIBLIOGRÁFICAS	186

LISTA DE FIGURAS

Fig. 4.1 Procedimento para a modelagem e análise de sistemas distribuídos e abertos para automação predial	79
Fig. 4.2 Processo de abstração/ refinamento na abordagem proposta	80
Fig. 4.3 Detalhamento da etapa “Modelagem do domínio do sistema predial”	81
Fig. 4.4 Detalhamento da etapa “Modelagem dos relacionamentos e serviços do sistema predial”	85
Fig. 4.5 Decomposição da descrição de um sistema em parte operativa e parte de controle	86
Fig. 4.6 Detalhamento da etapa “Modelagem dinâmica do sistema e análise formal dos modelos”	88
Fig. 5.1 Localização do PAMB no Complexo Hospitalar das Clínicas.....	96
Fig. 5.2 Planta ilustrativa do prédio dos ambulatorios (PAMB).....	99
Fig. 5.3 Diagrama da visão empresa –Domínios identificados no PAMB (Interação entre os objetos do ponto de vista empresa).....	108
Fig. 5.4 Diagrama de caso de uso –comunidade de usuários do PAMB	109
Fig. 5.5 Metamodelo das classes que compõem um sistema predial.....	110
Fig. 5.6 Diagrama de casos de uso do sistema de ar condicionado.....	114
Fig. 5.7 Modelo do fluxo de atividades do caso de uso ‘Estratégia área não utilizada’	115
Fig. 5.8 Modelo do fluxo de atividades do caso de uso ‘Estratégia em caso de incêndio’ ...	116
Fig. 5.9 Modelo do fluxo de atividades do caso de uso ‘estratégia para área utilizada’	117
Fig. 5.10 Descrição das classes do sistema de ar condicionado.....	119
Fig. 5.11 Diagrama de classes do sistema de ar condicionado	120
Fig. 5.12 Modelo em RPMCO da classe C1-ventilador de retorno.....	120
Fig. 5.13 Modelo em RPMCO da classe C ₂ – caixa de mistura.....	121
Fig. 5.14 Modelo em RPMCO da C ₃ - classe serpentina.....	121
Fig. 5.15 Modelo em RPMCO da C ₄ - classe torre de resfriamento	122
Fig. 5.16 Modelo em RPMCO da C ₅ - classe Chiller.....	123
Fig. 5.17 Modelo em RPMCO da C ₆ - classe Aquecedor.....	124
Fig. 5.18 Modelo em RPMCO da C ₇ - classe Bomba	125
Fig. 5.19 Fig. 5.19 Modelo em RPMCO da C ₈ - classe Controlador	125
Fig. 5.20 Diagrama de casos de uso do sistema de controle de acesso	126

Fig.5. 21 Modelo do fluxo de atividades do caso de uso 'identifica usuário e avalia acesso'	126
Fig. 5.22 Modelo do fluxo de atividades do caso de uso 'Estratégia em caso de incêndio'	126
Fig. 5.23 Modelo do fluxo de atividades do caso de uso 'Comunica a central'	127
Fig. 5.24 Modelo do fluxo de atividades do caso de uso 'Detecta presença'	128
Fig. 5.25 Descrição das classes do sistema de controle de acesso	130
Fig. 5.26 Diagrama de classes do sistema de controle de acesso	130
Fig. 5.27 Modelo em RPMCO da C ₁ - classe alarma	131
Fig. 5.28 Modelo em RPMCO da C ₂ – indicador luminoso	131
Fig. 5.29 Modelo em RPMCO da C ₃ – fechadura eletrônica	132
Fig. 5.30 Modelo em RPMCO da classe C ₄ – detector de presença	134
Fig. 5.31 Modelo em RPMCO da classe controlador	135
Fig. 5.32 Diagrama de casos de uso do sistema de gerenciamento de energia	135
Fig. 5.33 Modelo do fluxo de atividades do caso de uso 'Estratégia em caso de incêndio'	135
Fig. 5.34 Modelo do fluxo de atividades do caso de uso 'seleciona retira e libera carga	136
Fig. 5.35 Modelo do fluxo de atividades do caso de uso 'Estratégia falta de Energia'	136
Fig. 5.36 Descrição das classes do sistema de gerenciamento de energia	138
Fig. 5.37 Diagrama de classes do sistema de sistema de gerenciamento de energia	139
Fig. 5.38 Modelo em RPMCO da C ₁ – classe gerador	139
Fig. 5.39 Modelo em RPMCO da C ₂ - classe controlador	141
Fig. 5.40 Modelo em RPMCO da C ₃ - classe interface de usuário	142
Fig. 5.41 Diagrama de casos de uso do sistema de iluminação	142
Fig. 5.42 Modelo do fluxo de atividades do caso de uso 'ligar iluminação	143
Fig. 5.43 Modelo do fluxo de atividades do caso de uso 'desligar iluminação	143
Fig. 5.44 Descrição das classes do sistema de iluminação	145
Fig. 5.45 Diagrama de classes do SI	145
Fig.5.46 Modelo em RPMCO da C ₁ -lâmpada	146
Fig.5.47 Modelo em RPMCO da C ₂ - classe sensor de luminosidade	146
Fig. 5.48 Modelo em RPMCO da C ₃ - classe interruptor	148
Fig. 5.49 Modelo em RPMCO da classe controlador	148
Fig. 5.50 Diagrama de casos de uso do sistema de detecção e combate a incêndio	150

Fig. 5.51 Modelo do fluxo de atividades do caso de uso 'detecta e controla incêndio'.....	150
Fig. 5.52 Modelo do fluxo de atividades do caso de uso 'comunica detecção de incêndio'...	151
Fig. 5.53 Modelo do fluxo de atividades do caso de uso 'verifica equipamento de incêndio'	152
Fig. 5.54 Descrição das classes do sistema de detecção e combate a incêndio	152
Fig. 5.55 Diagrama de classes do sistema de detecção e combate a incêndio	154
Fig. 5.56 Modelo em RPMCO da classe sensor de fumaça.....	155
Fig. 5.57 Modelo em RPMCO da classe sensor de temperatura.....	155
Fig. 5.58 Modelo em RPMCO da classe sprinkler.....	156
Fig. 5.59 Modelo em RPMCO da classe ventilador.....	157
Fig. 5.60 Modelo em RPMCO da classe bomba de incêndio	159
Fig. 5.61 Modelo em RPMCO da classe bomba jockey	177
Fig. 5.62 Modelo em RPMCO da classe alarme.....	178
Fig. 5.63 Modelo em RPMCO da classe controlador.....	179
Fig. 5.64 Disposição das áreas consideradas dentro do quarto pavimento	180
Fig. B.1 Exemplo de um modelo em redes de Petri com a identificação de seus elementos	177
Fig. B.2 a) Arcos múltiplos. b) Representação compacta.....	178
Fig. B.3 a) TRANSIÇÃO habilitada. b) MARCAÇÃO após o disparo.....	179
Fig. B.4 Modelo em RP de um sistema computacional.....	180

LISTA DE TABELAS

Tabela 2.1 Classificação das funções de um edifício inteligente	11
Tabela 3.1 Propriedades dos multi-sets	61
Tabela 5.1 Pavimentos funcionais do PAMB	98
Tabela 5.2 Algumas informações do sistema de elevadores do PAMB	101
Tabela 5.3 Andares servidos e paradas dos elevadores do PAMB	102
Tabela 5.4 Tabela 5.4 dispositivos pertencentes aos diversos sistemas prediais	160

LISTA DE ABREVIATURAS E SIGLAS

ABCI	Associação Brasileira da Construção Industrializada
ABNT	Associação Brasileira de Normas Técnicas
ANSI	<i>American National Standards Institute</i>
ASHRAE	<i>American Society of Heating Refrigerating and Air Conditioning Engineers</i>
ASCII	<i>American Standard Code for Information Interchange</i>
ATM	<i>Asynchronous Transfer Mode</i>
BACNET	<i>Building Automation and Control Network</i>
BAS	<i>Building Automation System</i>
BCU	<i>Bus Coupling Unit</i>
BISDN	<i>Broadbands Integrated Services Digital Network</i>
BMS	<i>Building Management Systems</i>
CA	<i>Collision Avoidance</i>
CAB	<i>Canadian Automated Building Protocol</i>
CAL	<i>Common Application Language</i>
CD	<i>Collision Detection</i>
CEBus	<i>Consumer Electronic Bus</i>
CIB	<i>Computer Integrated Building</i>
CMS	<i>Communication Management System</i>
CO	<i>Cooperative Objects</i>
COM	<i>Component Object Model</i>
CORBA	<i>Common Object Request Broker Architecture</i>
CPD	Central de Processamento de Dados
CR	<i>Customization Rules</i>
CSMA	<i>Carrier Sense Multiple Access</i>
DCE	<i>Distributed Computing Environment</i>
DCOM	<i>Distributed Component Object Model</i>
DDC	<i>Direct Digital Control</i>
DP	<i>Distributed Peripherals</i>
EI	Edifício Inteligente

EIA	<i>Electronics Industry Association</i>
EIB	<i>European Installation Bus</i>
EIBA	<i>European Installation Bus Association</i>
EIBG	<i>European Intelligent Building Group</i>
EMS	<i>Energy Management Control System</i>
FMS	<i>FieldBus Message Specification</i>
FND	<i>Firm-Neutral Data Transmission</i>
HVAC	<i>Heating, Ventilating and Air Conditioning</i>
HOOD	<i>Hierarchical Object Oriented Design</i>
IBI	<i>Intelligent Building Institute</i>
IBSC	<i>Intelligent Building Study Committee</i>
ICHC	Instituto Central do Hospital das Clínicas
ICS	<i>Integrated Communication Systems</i>
IDL	<i>Interface Definition Language</i>
IEEE	<i>Institute of Electrical and Electronic Engineers</i>
InCOR	Instituto do Coração
ICr	Instituto da Criança
InRad	Instituto de Radiologia
IOT	Instituto de Ortopedia
IP	<i>Internet Protocol</i>
ISDN	<i>Integrated Services Digital Network</i>
ISO	<i>International Standard Organization</i>
ITU	<i>International Telecommunications Union</i>
LON	<i>Local Operating Network</i>
MAN	<i>Metropolitan Area Network</i>
MS	<i>Master-Slave</i>
NEST	<i>Novell Embedded Systems Technology</i>
OBM	<i>Object Behaviour Model</i>
OCN	<i>Object Coordination Net</i>
ODP	<i>Open Distributed Processing</i>
OMG	<i>Object Management Group</i>
OMT	<i>Object Modeling Technique</i>

OO	<i>Object Oriented</i>
OOD	<i>Object Oriented Design</i>
OOPN	<i>Object-Oriented Petri Net</i>
OOSE	<i>Object-Oriented Software Engineering</i>
OPN	<i>Object Petri Nets</i>
ORB	<i>Object Request Broker</i>
OSF	<i>Open Software Foundation</i>
OSGI	<i>Open Services Gateway Initiative</i>
OSI	<i>Open Systems Interconnection</i>
PA	<i>Process Automation</i>
PAMB	Prédio do Ambulatórios
PLC	<i>Programmable Logic Control</i>
PN-TOX	<i>Petri Net Tool for Object Concurrency specification</i>
PROFIBus	<i>PROcess FIeld Bus</i>
RMI	Remote Method Invocation
RP	Redes de Petri (PN - <i>Petri Nets</i>)
RPC	Redes de Petri Coloridas
RPC	<i>Remote Procedure Call</i>
RPMCO	Rede de Petri Modular Colorida baseada em Objetos
SAA	Sistema Aberto de Automação
SAC	Sistema de Ar Condicionado
SCA	Sistema de Controle de Acesso
SCFTV	Sistema de Circuito Fechado de Televisão
SDA	Sistemas Distribuídos e Abertos
SDCI	Sistema de Detecção e Combate a Incêndio
SDL	<i>Specification and Description Language</i>
SE	Sistema de Elevadores
SED	Sistemas a Eventos Discretos
SFA	Sistema de Fornecimento de Água
SGE	Sistema de Gerenciamento do Energia
SGP	Sistema de Gerenciamento do Prédio
SI	Sistema de Iluminação

SIMCON	<i>Simple Integrated Model for Complex Object Networks</i>
SS	Sistema Supervisorio
TCP	<i>Transmission Control Protocol</i>
THORNS	<i>Timed Hierarchical Object-Related Nets</i>
TP	<i>Token-Passing</i>
UART	<i>Universal Asynchronous Reception Transmition</i>
UML	<i>Unified Modeling Language</i>
UpnP	<i>Universal Plug & Play</i>
USP	Universidade de São Paulo
VAV	Vazão Variável de Ar
XML	<i>eXtensible Markup Language</i>

LISTA DE SÍMBOLOS

Redes de Petri

T	Transição $t_1 \dots t_n \rightarrow$ conjunto de transições.
P	Lugar $p_1 \dots p_m \rightarrow$ conjunto de lugares.
I	Função que define os arcos de entrada em relação às transições (p_1 para t_1)
k	$k \in \mathbb{N}$, $n \in \mathbb{N}$, $m \in \mathbb{N}$
O	Função que define os arcos de saída em relação às transições
M	Marcação.
M_0	Marcação inicial.
$M(p_i)$	Marcação do lugar p_i o número do lugar p_i
R	Conjunto de alcançabilidade.
N	Rede marcada.
C	Matriz de incidência ($n \times m$)
f	Vetor de lugares
s	Vetor característico.
m_x	Vetor coluna que representa a marcação imediatamente alcançável a partir de m_{k-1}
U_k	Vetor que representa o disparo da transição t_1

Redes de Petri Modular Colorida baseada em Objetos

Λ	Conjunto finito e não vazio de cores (tipos)
T	Transição $t_1 \dots t_n \rightarrow$ conjunto de transições.
P	Lugar $p_1 \dots p_m \rightarrow$ conjunto de lugares.
A	Numero finito de arcos.
L	Função nó
W	Função de cores.
m, m_i	Multi-set.
S	Conjunto não vazio.
s	Elemento de S
$m(s)$	Numero de vezes que aparece o elemento s no multi-set m.
S_{MS}	Conjunto de todos os multi-sets sobre S

W(p)	Conjunto de cores.
G	Função guarda.
E	Função expressa no arco.
$C(p(a))_{MS}$	Expressão que produz um multi-set no conjunto de cores anexado ao lugar.
J	Função de inicialização.
MT	Conjunto finito e não vazio de métodos MT_o : métodos apreciados. MT_r : métodos requisitados.
II	Função de inovação interna.
me	Método.
RI	Função resposta interna.
IE	Função de inovação externa.
RE	Função de resposta externa.
Var(t)	Conjunto de variáveis de t.
E (x_1, x_2)	Chamado de expressão (x_1, x_2)
b	Função vínculo.
B(t)	Conjunto de todos os vínculos.
TE	Conjunto de todas as marcas.
BE	Conjunto de todos os elementos vinculados.
Y	Etapa

1. INTRODUÇÃO

1.1. JUSTIFICATIVA E MOTIVAÇÕES

O desenvolvimento de sistemas prediais* tem avançado na medida em que os recursos tecnológicos tem correspondido às exigências do homem por maior confiabilidade, conforto, produtividade e segurança nas suas edificações.

Neste contexto, a diversificação e evolução dos serviços oferecidos e demandados em um edifício, a difusão e popularização das tecnologias mecatrônicas (incluindo dispositivos de tele-operação e monitoração remota) e de informação e, a necessidade de maior flexibilidade e versatilidade da infraestrutura predial, envolvem novos paradigmas para a concepção de edifícios (Wang, Xie, 2002). Estes novos paradigmas passam a incorporar até mesmo fatores comportamentais que promovem a produtividade nas tarefas que se realizam em seu interior (Takiuchi et al., 2004); (Abramsom, 1995).

Considerando estes fatores para o caso em que o objetivo é o projeto de edifícios inteligentes, observa-se que novas metodologias estão sendo desenvolvidas no sentido de considerar peculiaridades relacionadas às funcionalidades que se desejam associar a cada um dos sistemas que compõem o edifício. De acordo com estas abordagens, o comportamento e a interação dos usuários de um determinado edifício devem ser considerados e devidamente estudados para atender às suas necessidades e incrementar sua produtividade neste ambiente (Wong; Wang, 2005); (Finley et al., 1991).

* Neste trabalho o termo 'sistemas prediais' é utilizado para referir somente os sistemas de automação predial

Estes edifícios devem assim, ser projetados de forma que possam considerar adequadamente as necessidades dos usuários e absorver novas tecnologias, isto é, sua concepção deve considerar novas técnicas de planejamento, construção, manutenção, gestão e atualização. Estas considerações visam incrementar sua capacidade de adaptar-se aos requerimentos que a futura sociedade exigirá, procurando mantê-los produtivos ao longo de todo seu ciclo de vida (evitando a obsolescência prematura).

De acordo com as considerações anteriores, um edifício inteligente para alcançar suas metas deve incorporar sistemas mecatrônicos (sistemas mecânicos e eletro-eletrônicos integrados por computador) e de informação, que o capacitem para a prestação de serviços avançados com elevado grau de autonomia. Estes edifícios devem permitir além do controle automatizado, a monitoração e manutenção dos diferentes sistemas do edifício, considerando critérios de otimização e integração, com operação local e/ou remota, e com flexibilidade para possibilitar de maneira simples e econômica a incorporação de novos sistemas (Wong; Wang, 2005); (Roth, 1997).

O interesse por este tipo de edifício é justificado, quando se considera o custo inicial das edificações, projetadas em média para durar pelo menos 50 anos, e os custos de operação (5 a 10% em consumo de energia) e manutenção que constituem cerca de 50 a 80% dos custos totais do edifício (BRW, 2003); (Han, 1997). Em poucos anos é possível justificar uma despesa em automação predial que poderia girar em torno de 3,5% a 5% do total do empreendimento com retorno garantido resultante da redução dos custos de operação e manutenção (Saramago, 2002); (Pertusier, 1993). A instalação de sistemas de automação predial, além de trazer potenciais vantagens operacionais e de segurança ao edifício, corresponde a um item importante do ponto de vista econômico, contribuindo de forma significativa na diminuição de despesas e garantido o retorno do investimento (Saramago, 2002). Estudos indicam também que a produtividade dos profissionais alocados em edifícios com maiores graus de automação pode melhorar de 9% a 10% (ABCI, 1993). Em geral pode-se afirmar que a automação predial representa uma evolução tecnológica que pode gerar otimização operacional e benefício financeiro.

No projeto de edifícios inteligentes, os acabamentos, revestimentos, distribuição de energia, comunicações e os diversos sistemas prediais (como por exemplo, sistema de iluminação, sistema de elevadores, sistema de ar condicionado, etc.) são, do ponto de vista de sua operação e gestão, os elementos mais importantes, no entanto, somente uma efetiva integração[†] entre eles define a eficiência destes edifícios.

Assim, é fundamental que o projeto especifique uma construção funcional, que procure a integração entre os diferentes sistemas e a incorporação efetiva de novas tecnologias de modo a maximizar a produtividade dos ocupantes, otimizar a utilização de recursos, promover a conservação de energia, facilitar o gerenciamento e operação das instalações e minimizar o custo (Wong; Wang, 2005); (Fu, Shih, 2000); (Arkin; Paciuk, 1995); (Becker, 1995).

Neste sentido, os avanços em sistemas prediais específicos têm sido significativos nas últimas três décadas devido principalmente à introdução de novos sistemas de controle. Entretanto só alguns avanços têm se verificado na integração dos sistemas prediais. Esta integração, segundo diversos autores (Wong; Wang, 2005); (Fu, Shih, 2000); (Arkin; Paciuk, 1995); (Clark et al, 1995); (Flax, 1991) e (Fujie; Mikami, 1991), é uma necessidade para a operação efetiva de um edifício inteligente, pois é da interação entre as partes, que funcionalidades associadas a segurança, conforto dos ocupantes, confiabilidade e robustez do controle dos sistemas podem ser de fato implementadas reduzindo globalmente os custos de operação. Assim, é evidente que o projeto de edifícios inteligentes e especificamente o de seus sistemas prediais necessita de abordagens de modelagem e análise, que visem sua efetiva integração.

Dentro desta perspectiva e, considerando que a integração entre os diversos sistemas de um edifício é fundamental para o suporte das atividades dos usuários, deve-se estudar também meios e procedimentos para a sua implementação (Clark et al., 1995).

[†] Integração entendida como o processo de conectar física e logicamente sistemas e dispositivos em uma arquitetura comum de maneira que possam compartilhar e intercambiar informação (Wong; Wang, 2005).

Os métodos atualmente disponíveis carecem de uma visão global do edifício e não cobrem satisfatoriamente a integração dos sistemas prediais. No entanto, alguns métodos têm sido aplicados com relativo sucesso, quando enfocados para requerimentos de integração em situações específicas, como em caso de emergência ou monitoração remota. De qualquer modo, não existe um procedimento claro para se especificar e estruturar um sistema integrado de automação predial.

Evidencia-se, assim a necessidade de desenvolver estudos mais aprofundados sobre a integração de sistemas nos edifícios inteligentes e sua especificação, em particular no que se refere à identificação da interoperabilidade entre os diversos sistemas do edifício. Especificamente é necessária uma abordagem, na qual se considere o sistema como um todo e onde as atividades concernentes a cada sistema sejam devidamente planejadas e facilmente integradas, assegurando flexibilidade para adaptações nas políticas de segurança, de conforto e de conservação de energia do edifício enquanto, que ao mesmo tempo se minimize o esforço para estruturar e interfacear os sistemas envolvidos.

Por outro lado, os problemas que envolvem a integração de sistemas têm sido evidenciados em outros setores tecnológicos. Para tratar estas dificuldades têm sido concebidos os sistemas abertos, cujo objetivo é permitir a integração e interoperabilidade de sistemas e equipamentos de diferentes características e fornecedores.

Segundo Tanumaru (1994), um sistema de automação com características de sistema abertos pode atender as mais diversas exigências dos usuários, como flexibilidade, preservação do capital investido, interoperabilidade e conectividade com outros sistemas. Algumas características destes sistemas são: apresentar uma arquitetura flexível tanto de hardware como de software (sistemas padronizados facilitam sua adição e atualização), possibilitar a implantação de técnicas avançadas de controle e de monitoramento, permitir o desenvolvimento de hardware e software por terceiros e permitir a integração com um sistema de planejamento (Schofield, 1996).

Além disso, segundo Wang, (2002) a utilização de arquiteturas distribuídas de automação permitem um controle local efetivo, confiável e rápido.

Portanto, com base nos pontos levantados sobre a integração de sistemas prediais em edifícios inteligentes, o trabalho procura explorar o conceito dos chamados sistemas distribuídos e abertos que incluem as características mais importantes deste dois tipos de sistemas. Para isto propõe-se a utilização de uma abordagem que permita modelá-los e analisá-los de acordo com os requisitos de integração que se requer neste tipo de sistemas visando contribuir na sua especificação.

Neste sentido, e considerando que o *Open Distributed Processing* (ODP) além de ser um modelo de referência é a base de um *framework* para descrever e especificar sistemas distribuídos e abertos, o presente trabalho considera sua aplicação em edifícios inteligentes. A utilização do ODP é efetivada através de uma linguagem orientada a objetos, o que permite o tratamento de sistemas complexos e heterogêneos e a estruturação do processo de modelagem do sistema. A linguagem considerada adequada para esta finalidade é a *Unified Modeling Language* (UML) por representar diferentes perspectivas de um mesmo sistema e por ser um padrão de fato. No entanto a UML não conta com um método formal totalmente aceito para analisar os modelos gerados. Neste contexto, e levando em conta que a rede de Petri é uma técnica formal e gráfica para a modelagem e análise de sistemas a eventos discretos –entre os quais podem consideram-se os sistema prediais e ainda de comprovada efetividade quando usada para a modelagem destes (Bastidas, 2000); (Villani, 2000); (Gomes, 1997)- ela é considerada como o formalismo base neste trabalho.

1.2. OBJETIVO

O objetivo deste trabalho é propor uma abordagem para a modelagem e análise de sistemas distribuídos e abertos para automação predial.

A abordagem considerada para este propósito baseia-se no modelo de referência ODP, no uso do conceito de orientação a objetos e na teoria dos sistemas a eventos discretos, através da aplicação de técnicas derivadas da rede de Petri .

Dentro do objetivo apresentado, o presente trabalho envolve as seguintes metas:

- Comprovar que uma abordagem baseada no ODP e auxiliada, na modelagem na orientação a objetos e na análise na rede de Petri, é efetiva no tratamento dos sistemas distribuídos e abertos no domínio da automação predial, especificamente nos edifícios inteligentes, e contribui para à integração de seus sistemas.
- Propor um formalismo para suporte à modelagem e análise de sistemas distribuídos e abertos, no qual se considere os conceitos e pontos de vista do ODP, a orientação a objetos e as redes de Petri.
- Introduzir um procedimento de modelagem e análise de sistemas distribuídos e abertos para automação predial e que é baseada no formalismo proposto. Este procedimento considera as características e requerimentos dos sistemas prediais e do prédio do qual fazem parte e gera um conjunto de modelos baseados nos diagramas da UML, de maneira a estruturar o processo e definir o modelo dinâmico e para análise através da rede de Petri
- Demonstrar a utilidade e vantagem da abordagem proposta através de exemplos de aplicação em um prédio hospitalar

1.3. ORGANIZAÇÃO DO TEXTO

A seguir, apresenta-se uma descrição do conteúdo dos capítulos que compõem o trabalho:

Capítulo 1 - *Introdução*: Neste capítulo apresenta-se a justificativa, as motivações e o objetivo para o desenvolvimento deste trabalho, bem como a organização do texto.

Capítulo 2 - *Edifícios Inteligentes e Sistemas Distribuídos e Abertos*: Este capítulo apresenta as definições propostas para edifício inteligente, assim como as

suas funções e características, ressaltando o papel da integração entre seus diversos sistemas prediais.

Em seguida são apresentados os conceitos relevantes em relação aos sistemas distribuídos e abertos assim como as principais definições consideradas dentro do ODP.

Capítulo 3 - *Modelagem e Análise de Sistemas Distribuídos e Abertos*: Neste capítulo é apresentado o formalismo proposto para suporte à modelagem e análise de sistemas distribuídos e abertos, no qual se consideram os conceitos do ODP, a orientação a objetos e o uso da Rede de Petri para a análise dos modelos gerados

Capítulo 4 - *Procedimento para a modelagem e análise de sistemas distribuídos e abertos para automação predial*: Este capítulo apresenta uma sistematização do procedimento de modelagem e análise de sistemas distribuídos e abertos para automação predial e que visa a integração dos sistemas presentes neste tipo de aplicação. Esta abordagem baseia-se no formalismo proposto no capítulo 3.

Capítulo 5 - *Exemplo de aplicação*: Neste capítulo é apresentado o exemplo de aplicação em um prédio hospitalar para ilustrar a utilidade e vantagens da abordagem proposta.

Capítulo 6 - *Conclusão*: O capítulo apresenta a conclusão do trabalho, e as sugestões de novos tópicos a serem pesquisados em futuros trabalhos.

Anexos A e B: apresentam outros tópicos que são relevantes ao contexto deste trabalho como as comunicações em edifícios inteligentes e os conceitos básicos de rede de Petri.

2. EDIFÍCIOS INTELIGENTES E SISTEMAS DISTRIBUÍDOS E ABERTOS

Neste capítulo apresentam-se os conceitos básicos que estão envolvidos no presente trabalho. Assim, inicialmente trata-se os edifícios inteligentes, suas definições, suas características e os aspectos mais relevantes a serem considerados quando da integração entre seus diferentes sistemas prediais. Em seguida, são abordados os sistemas abertos e distribuídos, sua definição e as características que permitem que estes possam ser adotados na integração de sistemas prediais. Finalmente tratam-se os conceitos sobre o modelo de referência ODP, proposto pela ISO e a ITU para o desenvolvimento de sistemas distribuídos e abertos e as características da linguagem orientada a objetos usada para expressá-lo.

2.1. EDIFÍCIOS INTELIGENTES

Cada vez mais, novas tecnologias são disponibilizadas e desenvolvidas para os locais de trabalho, lazer e residência. Assim, mudanças conceituais na arquitetura de construções civis, projeto das instalações e na própria utilização das edificações estão transformando estes ambientes, constituindo-se em um tema de estudo amplo e multi-disciplinar.

Neste contexto, os edifícios inteligentes representam um produto que é o resultado da fusão de vários campos envolvidos no projeto, construção e operação de edifícios, alguns dos quais haviam sido considerados no passado como essencialmente distintos e sem interseção entre eles: a arquitetura interior e exterior, as tecnologias da computação e de telecomunicações, a ergonomia, os fatores

humanos, os processos construtivos e as tecnologias de suporte e operação de edifícios em geral (isto é, aquecimento, ventilação e ar condicionado –HVAC–, segurança predial, transporte e outras tecnologias envolvidas como: construção civil, mecânica, elétrica e mecatrônica) (Han, 1997).

2.1.1. DEFINIÇÃO

Atualmente não existe uma definição precisa do que pode-se denominar como edifício inteligente. Este termo foi inicialmente usado nos anos 70 por motivos meramente comerciais, anunciando alta qualidade e rápido retorno do investimento. Neste sentido, a definição dos serviços que estes deveriam prestar e o significado da chamada “inteligência do edifício” tem sido imprecisa e, sujeita ao critério pessoal de proprietários e usuários (Wong, Wang, 2005).

A “inteligência do edifício” tem sido objeto de diferentes interpretações em termos mundiais, isto é, nos Estados Unidos a característica mais importante dos edifícios inteligentes é a inter-conexão dos sistemas de serviços para o benefício dos ocupantes. Na Europa, um edifício inteligente deve tratar efetivamente a interação entre os sistemas e os elementos estruturais. No Japão, foi interpretada como o uso de novas e avançadas tecnologias para melhorar a capacidade dos edifícios em aspectos organizacionais (Arkin; Paciuk, 1995).

Posteriormente, foram desenvolvidas várias abordagens procurando obter uma definição única para este conceito, as quais acabaram sempre limitadas ao campo de atuação de quem as definiu, ou ainda do interesse específico de algum caso prático.

Entre as várias definições que têm sido apresentadas destacam-se os seguintes (Wigginton; Harris, 2002):

- Nos Estados Unidos, a definição aceita pelo IBI (*Intelligent Building Institute*) é: “Um edifício inteligente é aquele que permite a criação de ambientes produtivos e com custos efetivos, através da otimização de seus quatro elementos básicos: (1) estrutura, (2) sistemas, (3) serviços e (4) gerenciamento, assim como o inter-relacionamento entre eles” (Wigginton; Harris, 2002); (Maeda, 1993).

- A ABCI (Associação Brasileira da Construção Industrializada) define os edifícios com alta tecnologia como uma extensão à definição anterior incrementando: “aqueles que possuem um bom e atualizado projeto e uma construção racional e econômica; ou aqueles que são bem projetados e construídos, levando-se em conta as exigências de uso e evolução tecnológica”.
- Na Europa, o EIBG (*European Intelligent Building Group*) define um edifício inteligente como sendo aquele que “cria um ambiente que permite às organizações atingir os seus objetivos e maximiza a eficiência dos seus ocupantes enquanto, ao mesmo tempo, permite uma gestão eficiente dos recursos com um mínimo de custos em termos de ocupação humana” (Becker, 1995).
- No Japão, o IBSC (*Intelligent Building Study Committee*) define um edifício inteligente sob três pontos de vista: (a) um bom ambiente para as pessoas e os equipamentos, (b) bom suporte para alta produtividade dos trabalhadores no edifício e (c) boa segurança contra incêndio, patrimonial e individual e operação altamente econômica (So et al., 2001); (Maeda, 1993).

Em geral, todas as definições apresentadas têm como objetivo comum a criação de um ambiente eficiente e produtivo, através de custos mínimos.

Assim, os edifícios inteligentes têm evoluído como resposta às demandas da sociedade atual (eficiência, produtividade e segurança), à globalização, à diversificação e evolução dos serviços oferecidos e disponibilizados em um edifício, à difusão e popularização das tecnologias mecatrônicas e da informação e a necessidade de maior flexibilidade e versatilidade dos recursos. Neste contexto, estes aspectos têm-se convertido em um novo paradigma para a concepção, o projeto e a operação de edifícios (Wong, Wang, 2005); (Abramsom, 1995).

No projeto de edifícios inteligentes existem assim muitos aspectos que se tornam elementos definidores da sua eficiência. Desta forma, é fundamental que seu projeto especifique uma solução funcional, eficiente, flexível, segura e econômica que integre os seus diversos recursos e sistemas.

2.1.2. CARACTERÍSTICAS

Fujie; Mikami, (1991) afirmam que os edifícios inteligentes têm quatro objetivos principais:

- assegurar a satisfação das pessoas que trabalham dentro dele (segurança, eficácia e conforto);
- racionalizar a administração (controle de energia, controle dos serviços de manutenção, etc.);
- responder rapidamente, de modo flexível e econômico às diversas alterações que ocorrem em seu interior (adaptabilidade às funções requeridas);
- racionalizar a recepção e transmissão de informação, atuando como um recurso base para o gerenciamento das atividades internas e externas.

Atualmente, nos projetos de novos edifícios estes objetivos procuram ser atendidos através da introdução de vários níveis de automação em áreas distintas e com equipamentos diversos (Kroner, 1997).

Do ponto de vista da classificação das funções de um edifício inteligente, várias propostas têm sido apresentadas. Arkin; Paciuk, (1995) propõem uma divisão por exemplo em:

- Sistemas
- Serviços
- Gestão

As descrições destas funções são apresentadas na Tabela 2.1.

<i>2.1 Classificação das funções de um edifício inteligente</i>	
Funções	Descrição
Sistemas	Elevadores, HVAC (aquecimento, ventilação e ar condicionado), iluminação, energia elétrica, drenagem e abastecimento de água, controle de acesso, segurança (fogo e vida), intrusão, telecomunicações e processamento de dados.
Serviços	Voz/ vídeo/ dados, automação de escritório, serviços (compartilhados) a locatários, gestão de segurança, operações fora de hora, etc.
Gestão	Incluindo limpeza, manutenção, treinamento, propriedade, arrendamento, etc.

2.1.3. SISTEMAS EM UM EDIFÍCIO INTELIGENTE

De acordo com Wong, Wang (2005) e Flax (1991) um edifício inteligente é composto pela integração de uma grande quantidade e variedade de sistemas prediais dos quais os mais relevantes são a seguir listados:

- Sistema de iluminação;
- Sistema de elevadores;
- Sistema de ar condicionado e ventilação (*Heating, Ventilating and Air Conditioning – HVAC*);
- Sistema de controle de acesso;
- Sistemas de detecção e combate a incêndio;
- Sistemas de Gerenciamento de Energia (*EMS – Energy Management Control System*);

De acordo com o tipo de gerenciamento que executam, os sistemas prediais podem ser agrupados em (Saramago, 2002):

- **SISTEMAS PARA GESTÃO DA SEGURANÇA FÍSICA E PATRIMONIAL:** inclui os sistemas que preservam a segurança dos usuários do edifício, assim como o seu patrimônio. Aqui estão o sistema de controle de acesso, sistema de circuito fechado de televisão e o sistema de detecção e combate a incêndios.
- **SISTEMAS PARA GESTÃO DE CONTROLE AMBIENTAL:** dentro deste grupo consideram-se os sistemas que mantêm um meio ambiente com conforto térmico, lumínico, acústico, visual, olfativo e espacial. Entre eles estão o sistema de ar condicionado e ventilação, o sistema de fornecimento de água e o sistema de iluminação.
- **SISTEMAS PARA GESTÃO DE ENERGIA:** Abrange a distribuição e utilização da energia elétrica (controle de demanda e conservação de energia).

- **SISTEMAS PARA GESTÃO DE COMUNICAÇÕES:** estão incluídos dentro deste grupo os sistemas que permitem o acesso às facilidades internas e externas de telecomunicação.

Em geral, a classificação dos sistemas não é rigorosa pois estes podem servir a várias funções.

2.1.4. INTEGRAÇÃO DE SISTEMAS EM EDIFÍCIOS INTELIGENTES

Embora um edifício possa ser projetado ergonomicamente e com a incorporação de computadores e sistemas de telecomunicação, este ainda não é realmente um edifício inteligente, a menos que o mesmo ofereça a integração das facilidades e serviços necessária para atingir as suas metas (Wang; Xie, 2002).

A integração dos sistemas em um edifício inteligente torna-se uma característica indispensável para poder satisfazer os seus objetivos e oferecer melhor segurança, conforto, confiabilidade, robustez do controle dos sistemas e diminuição dos custos nas operações (Clark et al., 1995). Esta integração permite que todos os sistemas atuem cooperativamente e como um único sistema.

A integração de sistemas em edifícios inteligentes requer uma administração de complexidades que envolve a necessidade do conhecimento de cada processo e do efeito da automação no desempenho global, assim como a melhor escolha entre as inúmeras técnicas de controle e otimização disponíveis.

Segundo Azegami; Fujiyoshi (1993), a melhor integração entre os vários sistemas de um edifício inteligente é aquela realizada através da convergência do processamento da informação e das tecnologias mecatrônicas e de telecomunicações.

Em Fu et al. (2000), destaca-se que a meta atual de um edifício inteligente é integrar toda a informação necessária para gerenciar os sistemas eficientemente e disponibilizar recursos efetivos aos usuários.

A implantação de sistemas que visam a integração segue um caminho no qual deve-se atender às necessidades atuais e futuras dos usuários, sem obsolescência

antes de se obter o retorno do investimento (Wong, Wang, 2005). O grupo de serviços de valor agregado adicional, requer estudos cuidadosos de análise e implementação. Os sistemas devem ser flexíveis em suas instalações e devem prever sua expansão quando as necessidades dos usuários aumentam ou se alteram.

Segundo Marte (1994), a automação em um edifício inteligente deve prever o gerenciamento dos processos, onde o produto final devido à integração (sinergia), seja maior do que a soma das possibilidades de controle isolado dos componentes da edificação.

Em geral, as instalações e os projetos arquitetônicos e estruturais devem procurar alcançar um melhor grau de integração entre os sistemas, pois quanto mais efetivo o planejamento e o projeto da integração (considerando-se aspectos de interoperabilidade, engenharia dos sistemas e infra-estrutura “inteligente”), mais tangíveis serão os benefícios obtidos com a automação nas edificações. Considera-se assim que a chave da operação efetiva em um edifício inteligente é a integração entre os serviços, os sistemas e a estrutura (Fu et al., 2000).

Existem neste contexto, várias propostas para integrar os diversos sistemas prediais em um edifício inteligente. Inicialmente, o projeto de integração nestes edifícios consistia em coletar os dados de controladores locais dedicados à proteção contra incêndios, segurança, ar condicionado e ao gerenciamento de energia em uma central de monitoração destes serviços. No caso dos dados apresentarem desvios dos valores de referência era prevista a intervenção de um operador humano (Wang, Xie, 2002); (Mehta et al., 1995).

Uma outra proposta para a integração é a que considera a adaptação dos sistemas para tratar situações de emergência em contraposição a situações de operação normal. Tipicamente são considerados nesta integração: o sistema de detecção de incêndios, o sistema de elevadores e o sistema de ar condicionado. Sistemas mais avançados incluem a integração do sistema de controle de acesso e detecção de presença, o sistema de incêndio e o gerenciamento de evacuação. Com relação ao sistema de gerenciamento de energia, são considerados ainda controladores que avaliam a influência entre sistemas como o de iluminação e o de ar condicionado para gerenciar políticas globais mais econômicas (Wang, Xie, 2002).

Abordagens mais recentes consideram técnicas de inteligência distribuída, onde são usados microprocessadores nos controladores locais conectados em rede, permitindo que um maior número de parâmetros possam ser monitoradas e mais funções possam ser realizadas a menor custo. Estes controladores são conhecidos como estações locais e são conectados a um computador central *Building Management System* (BMS). Nesta arquitetura, o BMS implementa as funções de monitoração e controle avançadas (Wang; Xie, 2002).

Algumas outras abordagens consideram uma estrutura hierárquica para o sistema. Uma delas é aquela que contempla dois níveis na estrutura: um nível superior no qual se encontra o BMS. Sua função é o compartilhamento e processamento de informações em tempo real visando o gerenciamento do edifício como um todo, para a otimização global dos controles que envolvem os requerimentos de conforto, eficiência energética e funcionamento do edifício. E, um segundo nível hierárquico (subordinado ao BMS), onde se encontram os diversos sistemas prediais do edifício; os quais inter-atuam através do BMS (Wang; Xie, 2002); (Thumm, 1995).

Para implementar a integração, existe ainda o problema da comunicação entre os diferentes sistemas. Alguns esforços no sentido desta integração envolvem as redes digitais de serviços integrados (*Integrated Services Digital Network* - ISDN) e o conceito de sistemas distribuídos para redes digitais de serviços integrados (*Broadbands Integrated Services Digital Network* – BISDN), as quais combinam funções de telecomunicações e tele-computação em um sistema único que manipula sinais de voz, dados e imagens na forma digital (Kroner, 1997). Um compêndio sobre as tecnologias de comunicação atualmente disponíveis para edifícios inteligentes é encontrado no Anexo A

A integração de sistemas em edifícios inteligentes implica o compartilhamento de informações entre eles de maneira a simplificar as estratégias de controle e propiciar uma redução significativa dos custos (Han, 1997).

Em geral, o nível de integração nos edifícios varia, principalmente, de acordo, com a tecnologia disponível e o grau de prioridade e autonomia outorgado para cada sistema.

A seguir são apresentados os principais conceitos em relação aos sistemas distribuídos, sistemas abertos e sistemas distribuídos e abertos, com a finalidade de ressaltar as características que estes possuem e que estão sendo consideradas para a integração de sistemas prediais.

2.2. SISTEMAS DISTRIBUÍDOS

Um sistema distribuído é aquele onde os diferentes elementos que o constituem possuem as capacidades de processamento de informações, armazenamento de dados e comunicação, distribuídas computacional e/ou geograficamente (Chen; Taylor, 2002) (Chang et al., 1998).

Os sistemas distribuídos são compostos por um número de grupos de processamento homogêneos ou heterogêneos que estão ligados em rede (Buchs, Guelfi, 2000); (Wen-Tsung et al., 1998). Os sistemas implementados nos nós cooperam para realizar uma determinada tarefa e sistemas em diferentes nós interagem com outros através da rede. O tempo gasto em invocação entre os nós é o maior “gargalo” que afeta o desempenho total deste tipo de sistema (Chang et al., 1998).

Os aplicativos de processamentos distribuídos residem em diferentes computadores, cada um com capacidade de processamento e serviço de dados autônomos. Cada componente de um sistema distribuído possui um ambiente que realiza uma parte da tarefa total do sistema.

Em geral, o projeto, desenvolvimento e gerenciamento de sistemas distribuídos apresentam características particulares e desafiadoras que aumentam sua complexidade (Katchabaw et al., 1999). Algumas das suas principais características são: heterogeneidade, gerenciamento de recursos, coordenação, distribuição, concorrência (processos executados em diferentes nós do sistema implicam na possibilidade de comportamento concorrente, o que torna necessário a coordenação dos mesmos), modularidade e integração dos componentes do projeto (Giese; Wirtz, 2001).

Nos sistemas distribuídos, a transparência no acesso, a localização dos elementos através de sistemas intermediários – *middleware* (que consistem de um

conjunto de serviços disponíveis que possibilita que múltiplos processos, executando em uma ou mais máquinas, interajam através da rede de comunicação) e a confiabilidade das mensagens garantem a efetiva distribuição e a sua heterogeneidade (Kam et al., 2004).

2.3. SISTEMAS ABERTOS

Segundo o Comitê Técnico de Sistemas Abertos do *Institute of Electrical and Electronics Engineers* (IEEE), um sistema aberto é um conjunto amplo e consistente de perfis de padrões de tecnologia de informação internacionais e padrões funcionais, que especificam interfaces, serviços e formatos de suporte para alcançar a interoperabilidade e a portabilidade de aplicativos e dados.

Segundo Brock (2000), os sistemas abertos são implementações de hardware e software que se enquadram no corpo de padrões que permitem acesso livre e relativamente fácil a soluções de diferentes fornecedores. Os padrões utilizados podem ser definidos formalmente ou ser especificações seguidas pelos principais fornecedores e usuários na esfera tecnológica.

Martucci (1992) define o sistema aberto como aquele que possui uma arquitetura, cujos componentes e suas interligações estão apoiados em padrões que propiciam a interoperabilidade e a interconectividade dos componentes além da portabilidade de dados e aplicações.

Segundo Pizzica (2001), a implementação de sistemas abertos é mais fácil em processadores distribuídos.

Considerando as aplicações computacionais, os sistemas abertos apresentam as seguintes características (Kokai et al., 1998); (Wheeler, 1994):

- interoperabilidade: possibilidade de troca de informações entre computadores de diferentes fornecedores;
- portabilidade: operação consistente do *software* (sistemas operacionais e aplicativos) em computadores de diferentes fornecedores;

- interconectividade: acesso a diferentes sistemas através da rede de comunicação.

Discussões sobre sistemas abertos têm envolvido algumas discrepâncias entre fornecedores de bens e serviços e os órgãos governamentais e normalizadores como a ISO, ANSI, IEEE, entre outros. Estes últimos têm se mostrado preocupados em normas para orientar a utilização das novas tecnologias (Oliveira, 1998). Por outro lado os padrões são fundamentais para que os sistemas tenham condições de integração às novas tecnologias com baixo custo e atendendo as especificações de sistemas abertos.

Entretanto, o tempo para a elaboração e o estabelecimento de padrões formais é considerável. Como conseqüência, produtos ditos abertos colocados no mercado se baseiam apenas na interoperabilidade em algum grau com outros produtos (Oliveira, 1998). Os consórcios industriais (X/Open, OSF - *Open Software Foundation*, OMG - *Object Management Group*, etc.), nos quais os próprios fornecedores, clientes e interessados participam ativamente da elaboração de padrões em várias sub-áreas, têm de certo modo suprido as carências formais (Oliveira, 1998). No entanto, os organismos internacionais de normalização não deixaram de intensificar seus esforços nos últimos anos na criação de grupos para a formalização de padrões em vários segmentos: hardware, comunicação de dados e software. Assim, com o aprimoramento dos padrões formais e com o surgimento dos consórcios industriais, a diferença semântica de sistemas abertos tende a diminuir cada vez mais.

Um fator relevante para este direcionamento são os próprios usuários pois, interessa a estes a possibilidade de selecionar livremente seus fornecedores, com base em tecnologias mais avançadas e convenientes, e de realizar uma livre análise de custos, qualidade e manutenção. Nesta análise podem ainda ser considerados o grau de flexibilidade, portabilidade e interoperabilidade fornecido pelos equipamentos e sistemas disponíveis. O único e talvez o maior problema para os usuários, é saber quais padrões tecnológicos adotar e como lidar com a integração dos módulos que compõem os seus sistemas.

Do ponto de vista do usuário os sistemas abertos relacionam-se com (Kokai et al., 1998):

- a transferência do software no caso de mudanças nas características do sistema;
- atualização dos sistemas através da incorporação de novas tecnologias;
- conexão com outros sistemas através das redes.

Ainda, os benefícios dos sistemas abertos aos fornecedores estão no maior número de usuários e na redução de custos de desenvolvimento. Em geral, a aplicação de sistemas abertos permite que as empresas determinem a melhor solução com base em suas necessidades tecnológicas e no melhor equipamento disponível.

2.3.1. SISTEMA DE CONTROLE DE ARQUITETURA ABERTA

Um sistema de controle de arquitetura aberta tem a capacidade de integrar elementos de diferentes fornecedores e obter soluções com interfaces programáveis, mantendo o mesmo desempenho a baixo custo (Asato et al., 2002). Suas principais características são (Schofield, 1996); (Wright et al., 1996):

- permite maior flexibilidade tanto no hardware como no software para todos os níveis de controle;
- possibilita o uso de técnicas avançadas de controle e monitoramento;
- permite o desenvolvimento e a integração de hardware (controladores, sensores e atuadores) e software (algoritmos de controle) por terceiros;
- facilita a integração de controladores e destes com outros sistemas;
- tem capacidade de programação e execução de programas em várias plataformas;
- tem capacidade de aumentar ou diminuir a funcionalidade do sistema de acordo com a demanda.

Entre as principais vantagens dos sistemas de controle de arquitetura aberta estão: maior produtividade, maior capacidade de resposta e melhor relação custo-benefício.

2.4. SISTEMAS DISTRIBUÍDOS E ABERTOS

Considera-se um sistema distribuído e aberto, aquele que incorpora uma coleção de dispositivos autônomos (com sistemas operacionais e aplicativos heterogêneos) conectados em rede e que é projetado com uma arquitetura que atende um conjunto consistente de padrões internacionais (Katchabaw et al., 1999). Estes padrões especificam interfaces, serviços e formatos de forma a propiciar a interoperabilidade e interconectividade dos componentes, além da portabilidade dos dados e das aplicações do sistema (Pianegiani et al., 2002); (Kokai et al., 1998). Este tipo de sistema possui assim características dos dois sistemas anteriormente mencionados.

Nos sistemas distribuídos e abertos existem duas preocupações importantes: a definição de arquiteturas e a necessidade de ambientes para seu desenvolvimento (Scherer, 1997); (Rosa, 1995); (Barr, 1993).

Um trabalho que procura abordar a primeira preocupação é o trabalho de Martucci, (1992) no qual propõe-se o chamado: Sistema Aberto de Automação (SAA). Neste, procura-se conceber sistemas sem vinculações definitivas e que permitam através da configuração dos seus componentes obter sistemas genéricos, visando a definição de arquiteturas abertas, que poderão ser implementadas em diversos domínios de aplicação da automação sem necessidade de se refazer o projeto e a implementação do sistema. O SAA é um sistema aberto, inerentemente distribuído com uma estrutura abstrata e hierárquica. Entre os trabalhos baseados na aplicação do SAA estão o trabalho de Matsuyama (1997), onde foram determinadas arquiteturas de automação aplicados a processos contínuos, processos discretos e processos de serviços e, o trabalho de Becerra (1993), que foi aplicado na automação de sistemas de manufatura.

Por outro lado e devido à complexidade dos sistemas abertos e distribuídos, torna-se necessária a utilização de abordagens que consigam tratar de modo racional e sistemático todas as suas características. Neste sentido a ISO (*International Standard Organization*) e a ITU (*International Telecommunications Union*) propuseram o modelo de referência ODP (*Open Distributed Processing*) cuja estrutura visa suportar esta classe de sistemas.

2.5. OPEN DISTRIBUTED PROCESSING (ODP)

O modelo de referência ODP, é um padrão formal, criado pela ISO e a ITU. É considerada a iniciativa mais importante para o estabelecimento de padrões que direcionem o desenvolvimento de sistemas de processamento distribuído baseados no conceito de sistemas abertos (Blanc et al., 2000); (Gaspoz, 1996).

O ODP propõe um modelo cuja arquitetura permite a interoperabilidade e portabilidade de aplicações distribuídas e o desenvolvimento de padrões abertos. Este modelo objetiva a estruturação de um ambiente distribuído, aberto e heterogêneo cujos serviços para o processamento de informação são utilizados sem restrição por todos os elementos do sistema (Farooqui et al., 1995); (Sinnott; Turner, 1997).

Existem outros modelos, dentro da padronização dos sistemas distribuídos e abertos que compartilham os mesmos objetivos que o ODP, entre estes estão: o *Distributed Computing Environment* (DCE) do grupo *Open Software Foundation* (OSF) e o *Computer Object Request Broker Architecture* (CORBA) do *Object Management Group* (OMG) (Shankaran; Klefstad, 2004); (Becerra, 1998); (OMG, 1995); (Roseberry, 1992). Contudo, a proposta do ODP é mais abrangente, pois os objetivos das propostas DCE e CORBA se encaixam dentro do contexto da computação e da engenharia da especificação do modelo de referência ODP (Boiten et al., 2000); (Eckert, 1996); (Wolisz, 1993). O padrão ODP complementa assim estes modelos, um exemplo disto são os trabalhos conjuntos realizados com o grupo CORBA, como acontece na padronização da função ODP denominada de "Trader" (Becerra, 1998). O aparecimento do ODP tem permitindo a composição de federações e a cooperação de maneira transparente de aplicações heterogêneas (Diagne, 1996).

O ODP está organizado em dois modelos (ISO, 1995):

- Modelo descritivo: no qual se agrupam os conceitos necessários à conceição orientada a objetos das aplicações distribuídas.
- Modelo prescritivo: propõe uma arquitetura genérica para tais aplicações.

O padrão ODP tem como objetivo principal suportar a criação de sistemas distribuídos e abertos onde exista interoperabilidade entre diversos sistemas, sejam estes sistemas ODP, ou sistemas proprietários ou outros sistemas padronizados do mercado (Becerra, 1998).

Os sistemas distribuídos e abertos desenvolvidos a partir do ODP possuem as seguintes propriedades (Becerra, 1998); (ISO, 1995); (Tschammer, 1994):

- **abertura:** esta propriedade é viabilizada pela utilização de padrões através dos quais se obtêm altos graus de portabilidade e de interoperabilidade dentro do sistema;
- **flexibilidade:** permite a inclusão de novas tecnologias em função a novas especificações de funcionamento e novas configurações, graças à mobilidade física dos elementos dentro do sistema;
- **integração:** esta propriedade visa compor um sistema altamente produtivo e eficiente através da incorporação de vários sistemas e recursos de forma relativamente simples e com baixo custo;
- **modularidade:** permite a definição de módulos com interfaces bem definidas e serviços encapsulados de forma a garantir a autonomia destes;
- **segurança:** propriedade que garante a utilização dos recursos e da informação no sistema através de esquemas e mecanismos de segurança;
- **gerenciamento:** através desta propriedade se realiza o controle, supervisão e coordenação das ações/atividades e recursos do sistema visando atender as mudanças nas configurações, assegurar a qualidade dos serviços e as regras de negócio;
- **federação:** propriedade que permite a integração de diferentes domínios administrativos e tecnológicos dentro de um sistema distribuído;
- **transparência:** representa a capacidade de ocultar aos usuários e às aplicações, detalhes dos mecanismos utilizados para implementar diversos aspectos relacionados com a distribuição.

Estas propriedades no projeto de um sistema aberto e distribuído transformam-se em requisitos do sistema.

O ODP propõe a padronização do processo de especificação dos sistemas distribuídos e abertos, através de um modelo em alto nível que deixa aberta a forma de implementação, protegendo a criatividade e a inovação tecnológica.

2.5.1. ESTRUTURA DA NORMALIZAÇÃO DO ODP

A normalização ODP está organizada em quatro documentos:

- I. ISO/IEC 10746-1/ Recomendação ITU-T X.901: *Apresentação e guia do usuário*. Neste documento, apresenta-se de maneira geral o ODP, destacando seus objetivos, conceitos básicos e arquitetura. Este documento não constitui uma norma (ISOa, 1998); (Sinnott; Turner, 1997).
- II. ISO/IEC 10746-2/ Recomendação ITU-T X.902: *Fundamentos - Modelo descritivo* (provê um vocabulário comum). Define os conceitos gerais e as relações capazes de especificar sistemas distribuídos e abertos. Apresenta também um vocabulário para entender e aplicar a norma ISO/IEC 10746-3 (ISOa, 1996); (Sinnott; Turner, 1997);
- III. ISO/IEC 10746-3/ Recomendação ITU-T X.903: *Arquitetura - Modelo prescritivo* (restringe o que pode ser construído). Contém a especificação das características que são requeridas para qualificar um sistema distribuído como aberto. Contém também, a descrição de cada um dos pontos de vista do ODP, seus conceitos, estruturas, regras e funções (ISOa, 1996) (Sinnott; Turner, 1997)
- IV. ISO/IEC 10746-4/ Recomendação ITU-T X.904: *Semântica arquitetural*. Formaliza os conceitos referentes à arquitetura da norma ISO/IEC 10746-2 e as linguagens dos pontos de vista da norma ISO/IEC 10746-3, com a finalidade de permitir uma interpretação uniforme. Esta formalização é realizada pela interpretação de cada conceito utilizando diferentes

técnicas de descrição formal, como Lotos, Estelle, SDL e Z (ISO, 1998); (Sinnott; Turner, 1997).

Nos projetos de sistemas distribuídos e abertos, recomenda-se considerar os documentos anteriores na seguinte ordem de importância: o documento III é o documento mais importante a ser utilizado pois, contém as informações relacionadas com as especificações de cada ponto de vista. O documento II deve ser utilizado como subsídio teórico do documento III, e o documento I é uma referência de informações gerais. O documento IV deve ser utilizado quando o sistema é de tempo real ou em sistemas que contenham requisitos críticos (Becerra, 1998).

2.5.2. PONTOS DE VISTA

Os pontos de vista são meios para tratar a complexidade de um sistema. Assim, cada ponto de vista converte-se em uma abstração do sistema onde se destacam ou ocultam características deste. Este tipo de abordagem permite o desenvolvimento consistente de um sistema através da decomposição e consideração de seus diferentes aspectos (Blanc et al., 2000); (Farooqui ; Logrippo, 1995).

Cada ponto de vista, define requisitos e restrições ao projeto de um sistema. A especificação do sistema em um ponto de vista está baseada nos seus requisitos e deve refletir os requisitos identificados em outros. O projeto de um sistema, assim, deve considerar a integração dos modelos de cada ponto de vista. (Farooqui ; Logrippo, 1995).

Os pontos de vista constituem visões parciais da especificação completa do sistema. Desta forma, a descrição de um mesmo objeto (componente) pode ser realizada através de diferentes pontos de vista o que leva à necessidade de verificar a consistência entre eles (Kandé et al., 1998). A consistência entre pontos de vista é analisada através da identificação de conceitos e/ou propriedades em cada um deles. Assim, em um ponto de vista realizam-se abstrações ou refinamentos de conceitos e/ou propriedades de outro ponto de vista e vice-versa (Farooqui; Logrippo, 1995).

Neste contexto, o ODP define cinco pontos de vista e engloba conceitos e procedimentos voltados para orientação a objetos adaptando-se a diversos domínios

de aplicação dentro da automação (Marte, 2000). Os pontos de vista do ODP são abstrações consideradas como um mecanismo que possibilita manipular e selecionar as informações e características mais relevantes do sistema, necessárias para sua definição. Estas abstrações permitem observar o sistema através de diferentes visões consistentes entre si (Putman, 2001); (Marte, 2000); (Boiten et al., 2000); (Kandé et al., 1998). Os pontos de vista definidos pelo ODP são: de empresa, de informação, de computação, de engenharia e de tecnologia. Os modelos gerados em cada ponto de vista diferenciam-se entre si pela ênfase de sua abstração (Farooqui; Logrippo, 1995).

A especificação do ponto de vista é constituída de informações em linguagem natural e em conjuntos de diagramas orientados a objetos.

Em geral, o uso dos pontos de vista do ODP permitem especificar de forma completa e consistente uma arquitetura com características de interoperabilidade, distribuição, transparência e portabilidade (Putman, 2001).

A. PONTO DE VISTA DE ■

Define os requisitos básicos do sistema considerando o contexto da empresa e as necessidades dos seus usuários. Aqui o sistema é modelado em termos das políticas de negócio, dos seus objetivos, da funcionalidade requerida, dos domínios e dos atores (com seus papéis) envolvidos (Steen; Derrick, 1999); (Sinnott; Turner, 1997).

Este ponto de vista está focado no propósito, escopo e políticas do sistema. A linguagem usada nele inclui conceitos, regras e estruturas. Os conceitos aqui utilizados são independentes dos detalhes de distribuição e implementação (Tanaka et al., 2001) (Putman, 2001).

A estrutura do ponto de vista de empresa contém os seguintes conceitos (Lam-Son; Wegmann, 2005); (Putman, 2001); (Becerra, 1998):

- escopo da empresa: contém as características principais da empresa, enxergando esta como uma entidade de negócios;
- funções empresariais: funções realizadas pelos elementos ou objetos dentro da empresa;

- políticas: regras definidas na empresa ou por agentes externos a esta, as quais estão constituídas de obrigações, permissões e proibições;
- procedimentos: contém um conjunto de ações específicas realizadas pelos objetos empresa;
- comunidades: agrupamento de objetos com um propósito comum;
- federação: agrupamento de objetos pertencentes a diferentes domínios, sejam estes tecnológicos ou empresariais e com um propósito comum;
- contrato: acordo entre as partes.

A linguagem usada no ponto de vista da empresa permite identificar os principais componentes de um sistema, ajudando na organização dos requerimentos, da estrutura e das restrições do processo de negócio.

A especificação do ponto de vista de empresa contém (Putman, 2001):

- a comunidade de objetos que satisfazem os propósitos da empresa;
- declaração dos objetivos: porque o sistema é necessário e o que deve fazer;
- identificação dos objetos que fazem parte da comunidade de empresa, incluindo atores seus papéis e os recursos.
- papel desempenhado pelos objetos dentro da comunidade, limitado as restrições dadas pelas políticas da comunidade.
- regras para os objetos, as quais restringem ou habilitam ações: 1. obrigações: o que deve ser realizado. 2. permissão: o que pode ser realizado. 3. proibição: o que não deve ser realizado.
- atividades realizadas pelos objetos empresa;
- configuração dos objetos;
- contratos que definem restrições.

A estrutura básica do ponto de vista de empresa considera: o escopo da empresa, as funções empresariais, as políticas, os procedimentos, a comunidade e a federação da empresa e seus objetos (Becerra, 1998)

Especificar um sistema segundo o ponto de vista de empresa permite a criação de um modelo do sistema e do ambiente da empresa com o qual este interage (Marte, 2000).

B. PONTO DE VISTA DE ■

Descreve de maneira consistente, uma visão comum dos recursos que apóiam os requisitos de informação do ponto de vista de empresa. Este também define as relações entre elementos de informação e o processamento da informação.

Neste ponto de vista procura-se um modelo que permita identificar as origens e destinos das informações e define-se a semântica da informação e seu processamento no sistema (Marte, 2000); (Sinnott; Turner, 1997).

A especificação do ponto de vista de informação contém (Putman, 2001):

- identificação dos objetos informação;
- atributos estáticos dos objetos informação;
- associação entre objetos informação;
- especificação do comportamento entre objetos informação;
- relações entre o comportamento dos objetos informação;
- contratos para os objetos informação.

Os conceitos usados neste ponto de vista são independentes dos detalhes de distribuição e implementação (Putman, 2000).

Existem 3 esquemas que constituem a estrutura do ponto de vista de informação:

- **esquema invariante:** conjunto de condições ou predicados de um ou mais objetos 'informação' que, para todos os comportamentos válidos do sistema, sempre devem ser verdadeiros. No esquema invariante,

obrigações de uma política são entendidas como restrições (Marte,2000); (Becerra, 1998); (ISOa, 1996).

- **esquema estático:** é a especificação do estado de um objeto ou conjunto de objetos ‘informação’ em um ponto particular no tempo sujeito às restrições do esquema invariante. Esta é uma visão instantânea da informação (Marte,2000); (Becerra, 1998); (ISOa, 1996).
- **esquema dinâmico:** define todos os estados permitidos de um objeto ou conjunto de objetos ‘informação’ sujeito às restrições do esquema invariante. Aqui se realiza uma descrição da informação que reflete o comportamento e a evolução do sistema. Especifica como a informação pode evoluir durante a operação do sistema. No esquema dinâmico, são capturadas ações que causam uma mudança no estado de um objeto (Marte 2000); (Becerra, 1998); (ISOa, 1996).

C. PONTO DE VISTA DE ■

Este ponto de vista trata da decomposição funcional do sistema em objetos que interagem (trocam informação) através de interfaces específicas. Aqui também é abordada a distribuição das funções realizadas dentro do sistema sem se preocupar com a infra-estrutura de comunicação (Akehurst et al., 2003);(Becerra, 1998); (Nankman; Nieuwenhuis, 1996).

Neste ponto de vista são representados os componentes individuais que constituem as fontes e destinos dos fluxos de informação.

Os objetos no ponto de vista de computação podem ser objetos de aplicação, de suporte de serviços ou ainda de infra-estrutura (Putman, 2001).

A linguagem computacional faz grande uso do modelo de objetos. Um objeto computacional oferece uma interface através da qual acontecem as interações com outros objetos. Os objetos podem oferecer mais de uma interface, cada uma das quais pode ser de tipos diferentes ou iguais mas permitindo diferentes interações. Uma interação é um conjunto de serviços oferecidos através de uma interface e que estão unidos a outros objetos através de uma ligação. Objetos encapsulam

comportamentos. e incluem atividades internas a este as quais podem ser oferecidas a outros objetos através de uma interação (Putman, 2001).

A especificação computacional trata da estruturação de aplicações independentes de como os objetos computacionais estão distribuídos ou do tipo de computador ou rede utilizados. Assim, a especificação computacional é um modelo lógico e não um modelo físico da topologia da rede de comunicações e seus módulos. A especificação computacional foca o processamento de componentes de aplicação, as interações (em paralelo ou em seqüência) e ligações entre componentes e a maneira como as restrições entre as ligações são atendidas (Putman, 2001).

No ponto de vista de **computação**, os objetos são mapeados a partir dos objetos ‘empresa’ e ‘informação’ e são representados por classes. As interfaces computacionais (utilizadas pelos objetos ‘computação’ para realizar seus pedidos de serviço) podem ser determinadas pelo tipo de serviço assim, definem-se as seguintes interfaces (Romero; Vallecillo, 2004); (Becerra, 1998):

- **interface de fluxo:** suporta interações que representam fluxos contínuos de informação tal como o fluxo de sinal de áudio em uma aplicação multimídia;
- **interface de operação:** suporta interações semelhantes aos procedimentos ou rotinas utilizados em programação de sistemas;
- **interface de sinal:** suporta interações que representam invocações tais como eventos e interrupções.

A conexão entre as interfaces pode se dar de duas maneiras:

- **conexão explícita:** estabelecida entre interfaces de operação ou de fluxo;
- **conexão implícita:** estabelecida entre interfaces de operação.

A interação entre objetos ‘computação’ é essencialmente assíncrona (Marte, 2000).

A visão da **computação** do modelo de referência ODP descreve também como as aplicações e os componentes distribuídos do sistema ODP interagem de forma

cooperativa e transparente. A linguagem do ponto de vista de computação define as ações que um objeto pode executar, permitindo a criação de novos objetos e interfaces e o estabelecimento de ligações. Neste contexto, esta linguagem define as bases para permitir a interoperabilidade e a portabilidade dos componentes do sistema (Becerra, 1998).

D. PONTO DE VISTA DE ■

A especificação do ponto de vista de engenharia, complementa a visão de computação definindo os mecanismos e funções necessárias para suportar interações dos objetos distribuídos e as transparências que ocultam a complexidade destas interações. Este ponto de vista define as necessidades de comunicação e o desdobramento das funcionalidades do sistema (Wegmann; Naumenko, 2001)

As partes básicas do ponto de vista de engenharia são (Putman, 2001):

- modelo do canal : é um modelo genérico da infraestrutura de distribuição;
- transparências de distribuição: garantem abstrações da infraestrutura distribuída;
- funções ODP: definem os serviços básicos que habilitam a distribuição;
- modelo do nó: é a configuração dos objetos que formam a infraestrutura de um único nó (um computador).

A linguagem de engenharia estabelece os conceitos e regras para especificar os mecanismos que permitem a distribuição física e suportam o processamento lógico das aplicações. Assim, define-se como os objetos interagem e como a infraestrutura e os mecanismos de comunicação dão suporte às interações entre objetos (Putman, 2001).

Uma especificação de engenharia define a infraestrutura requerida para suportar a distribuição funcional do sistema. Esta especificação se pode realizar baseada em: (Putman, 2001):

- objetos básicos de engenharia que correspondem a objetos computacionais que não podem ser executados isoladamente;

- interfaces de engenharia que correspondem às interfaces computacionais;
- as funções ODP requeridas para gerenciar a distribuição física, a comunicação, o processamento e a armazenagem de informações;
- os papéis de diferentes objetos de engenharia dão suporte às funções ODP;
- as transparências requerem esconder os detalhes de distribuição do desenvolvedor, da aplicação e do usuário;
- uma configuração de objetos 'engenharia', estruturados como grupos (*clusters*), cápsulas e nós.
- as atividades que ocorrem dentro destes objetos 'engenharia'.
- os mecanismos de interconexão de ligações locais ou distribuídas para dar suporte às interações computacionais.

Existe uma relação direta entre o ponto de vista de computação e o ponto de vista de engenharia, pois os objetos da computação são vistos no ponto de vista de engenharia como objetos básicos da engenharia. E, as conexões entre objetos da computação são vistas como canais de comunicação ou conexões locais.

A estrutura de gerenciamento definida neste ponto de vista tem por objetivo organizar a infra-estrutura do sistema para otimizar a forma de gerenciamento dos objetos distribuídos. Os elementos que fazem parte desta estrutura são:

- **nó:** composto por um processador de informação onde são executadas as funções de processamento, armazenamento e comunicação, de maneira similar ao computador e seu sistema operacional;
- **núcleo:** encarregado da coordenação das funções de processamento, armazenamento e comunicação dentro do nó;
- **cápsula:** agrupamento de objetos tratados como uma única unidade de processamento.

- **grupo** (*Cluster*): conjunto de objetos básicos de engenharia que são tratados como uma unidade e cujo objetivo é executar funções específicas como ativação, desativação, reativação, recuperação e migração.

A comunicação entre objetos ‘engenharia’ pode ser realizada em duas formas:

- quando os objetos pertencem ao mesmo grupo ou nó: neste caso utiliza-se uma conexão que pode ser implantada utilizando mecanismos específicos do sistema que não se encontram padronizados.
- quando os objetos pertencem a diferentes cápsulas ou nós: aqui, utiliza-se uma conexão distribuída que está padronizada no ODP. Para implantar este tipo de conexão é necessário configurar uma estrutura denominada canal. Um canal é constituído dos seguintes objetos:
 - adaptador (*Stub*): serve como conversor dos dados que são gerados na interação dos objetos básicos de engenharia;
 - conector (*Binder*): mantêm a integridade ponto-a-ponto do canal, resolvendo problemas de distribuição;
 - protocolo: provê a comunicação entre os objetos;
 - interceptor: é utilizado para compatibilizar requerimentos técnicos existentes entre dois nós como a conversão de protocolos, o controle de acesso, a compressão de dados ou a utilização de mecanismos de criptografia.

O ponto de vista de engenharia permite que as características de sistemas abertos possam ser alcançadas através do estabelecimento das transparências e das funções ODP (Marte, 2000).

E. PONTO DE VISTA DE ■

A especificação do ponto de vista de tecnologia contém os requisitos referentes à tecnologia (componentes de *software* e *hardware*), que serão utilizados na implementação dos elementos do ponto de vista de engenharia. Isto é, a

tecnologia que suportará o processamento distribuído e sobre a qual o sistema será construído (Marte, 2000).

A especificação da tecnologia define (Putman, 2001):

- escolha da tecnologia;
- escolha do produto;
- escolha de padrões;
- configuração de objetos da tecnologia;
- especificação de todas as interfaces usadas entre os objetos ‘tecnologia’;
- como as especificações ODP são implementadas;
- tecnologias relevantes e mapeamento para padrões e especificações relevantes.

A especificação tecnológica define mecanismos que devem ser suportados pelo software, o hardware, as redes de comunicação, os sistemas operacionais, os dispositivos de armazenamento e os protocolos de comunicação entre outros. O ponto de vista de tecnologia não entra nos detalhes de como mapear a tecnologia e os produtos, mas indica que a correspondência da escolha de uma tecnologia envolve o mapeamento das especificações de outros pontos de vista. Assim, a linguagem da tecnologia propicia uma ligação entre as especificações dos outros pontos de vista e os detalhes da implementação (Putman, 2001).

As escolhas de tecnologia sempre mudam mas a especificação da arquitetura (como um grupo de especificações de empresa, informação, computacional e de engenharia) pode permanecer imutável às evoluções tecnológicas. Assim, as escolhas de tecnologia devem ser separadas da especificação de outros pontos de vista, de forma a permitir que o sistema possa ser planejado com novas tecnologias, sem ter que se re-arquitetar o sistema (Putman, 2001).

Para a implementação e comunicação entre objetos distribuídos, existem atualmente três padrões: um padrão “de jure” o CORBA da OMG e padrões de mercado como: COM/DCOM (*Component Object Model / Distributed Component*

Object Model) da Microsoft e o JavaBeans da Sun Microsystems. Estes três padrões implementam os canais de comunicação entre os diferentes nós da visão de engenharia, os quais, por convenção, são chamados ORBs (*Objetc Request Brokers*) na visão de tecnologia (Marte, 2000).

Embora todos utilizem, em geral, um protocolo comum de comunicação, o TCP/IP (*Transmission Control Protocol /Internet Protocol*), a comunicação entre objetos destes padrões é incompatível, pois cada um tem seu próprio mecanismo de transparência da distribuição, baseados em funções ODP próprias como o DCE RPC (*Distributed Computing Enviroment / Remote Procedure Call*) no padrão COM, o IIOP no CORBA e o RMI no JavaBeans. Para permitir a troca de mensagens entre objetos destes padrões é necessário acrescentar *bridges*, que realizam a conversão entre estes protocolos (Marte, 2000).

2.5.3. TRANSPARÊNCIAS E FUNÇÕES

O ODP identifica através dos pontos de vista os elementos e as estruturas do sistema. Para completar esta especificação em termos de integração e cooperação existem os conceitos de transparências e de funções ODP (Putman, 2001).

A. TRANSPARÊNCIAS

O conceito de transparência envolve os mecanismos abstratos que objetivam esconder dos sistemas e usuários as complexidades dos serviços oferecidos pela estrutura distribuída, otimizando a execução de tarefas cooperativas entre os elementos do sistema. As regras das transparências são especificações em linguagem natural que apresentam as restrições e requisitos que devem ser considerados nas funções ODP. O padrão ODP define as seguintes transparências: acesso, falha, localização, migração, persistência, replicação, realocização e transaccional (Marte, 2000); (Becerra, 1998).

B. FUNÇÕES

O conceito de funções envolve os objetos computacionais que permitem a estruturação e implementação das transparências. As funções do ODP estão

organizadas em quatro grupos genéricos: gerenciamento, coordenação, repositórios e segurança (Marte, 2000); (Becerra, 1998).

2.5.4. NOTAÇÃO DO ODP

Em cada ponto de vista têm sido definidos os correspondentes conceitos e regras para serem usados na especificação do sistema. No entanto, tem sido deixado em aberto qual notação usar em cada um deles. Alguns dos requisitos que se considera devem ser levados em conta quando da escolha da notação são (Duran et al., 2003); (Kandé et al., 1998):

- representação adequada dos conceitos de orientação a objetos, pois o ODP é baseado neste paradigma;
- expressar apropriadamente os conceitos de cada ponto de vista (usar a mesma linguagem em todos os pontos de vista);
- facilidade de entendimento e uso de forma a propiciar a comunicação entre as pessoas envolvidas no projeto;
- possibilidade de verificação da consistência entre os pontos de vista;

Neste contexto, a seguir apresentam-se as características da orientação a objetos que são fundamentais quando consideradas à luz do desenvolvimento dos sistemas distribuídos e abertos. Também é justificada a escolha da técnica de orientação a objetos para o presente trabalho.

2.6. ORIENTAÇÃO A OBJETOS E UNIFIED MODELLING LANGUAGE (UML)

A Orientação a objetos visa organizar o mundo real como uma coleção de objetos que incorporam uma estrutura de dados e um conjunto de operações que manipulam estes dados (Douglass, 1998). Para esta finalidade várias linguagens têm sido criadas sendo uma das mais importantes é a UML .

2.6.1. ORIENTAÇÃO A OBJETOS

O paradigma da orientação a objetos pode ser aplicado a sistemas distribuídos e abertos de modo a estabelecer técnicas que visem sua implementação (Eckert, 1996). Este paradigma permite a especificação e implementação de aplicações distribuídas e modulares independentes de domínios tecnológicos e organizacionais.

Entre os benefícios do uso do paradigma de objetos em sistemas distribuídos e abertos estão (Deboni, 2003); (Eckert, 1996); (Douglass, 1998):

- permite um mapeamento das entidades do mundo real em entidades de programação;
- independentemente da localização do objeto (domínio local ou remoto), se usará este sempre da mesma forma;
- independência de fornecedores de software: a especificação das interfaces de um objeto em uma linguagem de especificação padrão permite que possam ser usados por diferentes fornecedores;
- reusabilidade: componentes de software testados podem ser usados diretamente em novos projetos o que permite um desenvolvimento rápido;
- refinamento gradual por especialização: os sistemas distribuídos e abertos permitem modificações incrementais. Assim objetos novos e mais especializados podem ser substituídos por objetos mais gerais. Esta habilidade permite que sistemas de objetos distribuídos possam ser construídos, expandidos e reconfigurados quando necessário;
- melhoramento do problema da abstração do domínio: a abordagem baseada em objetos mantém uma forte relação entre os dados e os objetos que manipulam estes dados, ficando assim mais próxima do domínio do problema;
- maior estabilidade na presença de mudanças pois a orientação a objetos está baseada em elementos do mundo real;

- concorrência: na orientação a objetos os modelos são inerentemente concorrentes e representam este comportamento;
- encapsulamento, portabilidade e autonomia: a separação das interfaces de suas especificações comportamentais é um pré-requisito para o encapsulamento e a portabilidade. Os conceitos de abstração e encapsulamento permitem que a interação entre objetos seja restrita a interfaces bem definidas e atribuem independência de implementação aos componentes;
- escalabilidade: nos sistemas de objetos distribuídos, a granularidade de um objeto é transparente ao usuário. O objeto pode executar e definir seus serviços em um amplo grupo de plataformas de *hardware* e *software*;
- consistência entre visões do modelo: pode se usar o mesmo conjunto de visões para todas as fases de desenvolvimento de um sistema baseado na orientação a objetos, sendo que uma visão deve manter consistência com as outras;
- manutenibilidade /extensibilidade;
- permitem tratar a complexidade de um sistema e tratá-lo de forma modular;
- reduz significativamente, a partir de composições estruturadas de sistemas menores e confiáveis, o risco de construção de sistemas complexos;
- facilita a comunicação eficiente entre usuários e desenvolvedores de software;
- confiabilidade e robustez devido à utilização do conceito de encapsulamento e reutilização.

Os principais conceitos relacionados com a orientação a objetos são (Deboni, 2003); (Rumbaugh et al., 1991)

OBJETO: é uma entidade composta por: nome (identificação), atributos (características, dados) e métodos (operações que o objeto pode realizar). Segundo Paludetto (1991), o objeto é uma entidade independente, pois apresenta um comportamento idêntico mesmo dentro de contextos de utilização diferentes. O objeto deve possuir coesão interna, no sentido que os dados e operações que ele contém devem ser fortemente ligados e devem representar um subconjunto lógico significativo do sistema. Além disso os objetos devem ser fracamente ligados entre si, no sentido de que a interface entre os objetos deve ser a mais simples possível.

CLASSE: é um conjunto de objetos com atributos e métodos semelhantes. Cada objeto apresenta uma estrutura de dados e realiza as operações definidas para a classe a qual pertence. Neste contexto, um objeto é uma instância de uma classe.

COMUNICAÇÃO: Os objetos se comunicam através de mensagens, isto é, um sinal enviado de um objeto a outro requisitando serviço através da execução de uma operação. Trata-se de um ciclo completo onde uma mensagem é enviada a um objeto, operações são executadas dentro desse com base nos dados de seu alcance na hierarquia de classes, e uma mensagem contendo o resultado da operação é enviada ao objeto solicitante. Como um sistema é formado por um conjunto de objetos, o processamento do sistema é obtido mediante a troca de mensagens entre os objetos. Para que os objetos se comuniquem, é necessário que haja algum tipo de vínculo integrando esses objetos. Esses vínculos que podem ser relacionamentos existentes entre os objetos, asseguram o conhecimento que um objeto possui da existência do outro.

INTERFACE: é o conjunto de todas as assinaturas* definidas pelas operações de um objeto. Em uma assinatura, cada operação declarada por um objeto especifica o nome da operação, os objetos que toma como parâmetros e o valor de retorno da operação (Gamma et. al, 1995). A interface de um objeto portanto especifica todas as operações que este pode realizar.

* *Assinatura:* é o nome e parâmetros de uma característica comportamental. A assinatura pode incluir opcionalmente um parâmetro de retorno (OMG, 2003); (Montez, 1997)

O que permite a comunicação entre objetos é a definição de uma interface precisa.

As características básicas das mensagens entre objetos são:

PADRÃO DE CHEGADA: descrevem o comportamento das mensagens no tempo.

Podem ser:

- Episódico: quando as mensagens são imprevisíveis mas na maioria dos casos podem ser limitadas a uma “janela” temporal.
- Periódicos: são caracterizados por um período ou frequência de ocorrência e pela variação deste que é também chamada de *jitter*.

PADRÃO DE SINCRONIZAÇÃO: podem ser assíncronos e síncronos.

- Assíncronos: ocorre quando um objeto requisita um método a outro objeto, sem esperar por uma resposta de fim de operação. Neste caso, nada garante que o método anteriormente requisitado foi completamente realizado.
- Síncronos: ocorre quando um objeto requisita um método de um outro objeto e aguarda uma mensagem de resposta confirmando o fim da operação.

Outros conceitos fundamentais na orientação a objetos são:

ENCAPSULAMENTO: consiste na separação das características internas de um objeto dos aspectos externos. Assim, os detalhes internos de um objeto ficam inacessíveis para as demais partes do sistema o que permite reduzir o risco de corrupção involuntária do sistema.

ABSTRAÇÃO: consiste em ignorar detalhes que não são essenciais, concentrando-se no que um objeto é e o que ele realiza, antes de decidir como deve ser implementado. A abstração preserva a liberdade de se tomar decisões sobre a implementação dos objetos, evitando, tanto quanto possível, comprometimentos prematuros com detalhes.

POLIMORFISMO: existe quando a mesma mensagem pode resultar em eventos diferentes quando recebidas por objetos distintos.

CLASSIFICAÇÃO: consiste no agrupamento de objetos com dados e operações semelhantes, em classes.

HERANÇA: define o relacionamento, no qual uma classe compartilha a estrutura e comportamento definido em uma ou mais classes. A classe filho que “herda” os atributos e os métodos é chamada subclasse e a classe mãe, é denominada superclasse.

Em OO, existem duas formas de hierarquia para os objetos em sistemas estes são:

HIERARQUIA DE OBJETOS: é o resultado de uma decomposição estrutural onde se mostra como os diferentes objetos interagem. Aqui a relação entre as partes resultantes da decomposição e o nível hierarquicamente superior é do tipo “é parte de” (Booch, 1999).

HIERARQUIA DE CLASSES: é o resultado direto da aplicação de relações de herança e evidencia a estrutura e os comportamentos em comum de elementos dentro do sistema. É representada pela organização do sistema através de relações do tipo “é um de” entre um elemento de nível hierárquico inferior e seu correspondente no nível superior (Booch, 1999).

A orientação a objetos permite cobrir as diferentes fases do ciclo de vida do *software* desde a análise das necessidades até a realização final da solução, capturando e estruturando progressivamente a informação necessária à criação do mesmo. Todas estas vantagens da orientação a objetos tem servido para a escolha deste como base do modelo prescritivo do ODP (Diagne, 1996). Esta adequação da orientação a objetos com as atividades da criação de software são reforçadas nos sistemas distribuídos enquanto propõem o uso da noção de objeto como entidade elementar a qual está de acordo com a noção de módulos como entidade de estruturação. Assim, a orientação a objetos é uma base válida para construir tais sistemas

Considerando os benefícios da orientação a objetos e uma técnica que obedeça este paradigma e que também atenda os requisitos para a notação do ODP levantados no item 2.5.4, escolheu-se a linguagem UML (*Unified Modelling Language*) que é a seguir apresentada.

2.6.2. UNIFIED MODELLING LANGUAGE (UML)

Diversos trabalhos publicados (Deboni, 2003);(Booch et al., 1999); (Furlan, 1998) indicam que a UML está-se tornando uma linguagem de modelagem padrão de fato, na qual se procura definir a semântica do modelo do objeto, além de fornecer uma notação para capturar e comunicar a estrutura e o comportamento do objeto. A UML resultou dos esforços de vários especialistas na área de orientação a objetos entre os que se encontram: Grady Booch (OOD – *Object Oriented Design*), Ivar Jacobson (OOSE - *Object-Oriented Software Engineering*), Jim Rumbaugh (OMT - *Object Modeling Technique*) e David Harel (*Statecharts*) e foi reconhecida em 1997 pela OMG (Object Management Group) como notação para a modelagem de sistemas (Deboni, 2003)

A UML é uma linguagem que define uma notação diagramática e cuja semântica usa os conceitos da orientação a objetos (Kande et al., 1998). De uma forma geral, a UML pode ser considerada como uma linguagem que, a partir da representação gráfica de alguns elementos básicos e dos tipos de relação entre estes elementos, define diagramas padronizados que têm como finalidade a representação de diferentes perspectivas de um mesmo sistema (Booch et al, 1998). No entanto a UML não constitui uma linguagem visual de programação mas, os modelos obtidos através de seu uso podem ser convertidos para diversas linguagens de programação (Booch et al, 1998).

A UML possui basicamente dois tipos de diagramas: estáticos para a representação dos dados e dinâmicos para a representação do controle do sistema. Alguns dos principais elementos estruturais da UML são: objetos, classes, componentes entre outros. Para representar o comportamento dinâmico na UML são considerados estados e interações.

Os diagramas definidos pela UML são (OMG, 2003); (Booch et al., 1999):

- **Diagrama de casos de uso** (*Use case diagram*) explicita as interações e relacionamentos do sistema com agentes externos. Mostra uma visão estática do sistema (contexto do sistema).
- **Diagrama de objetos** (*Object diagram*): Apresenta os objetos que fazem parte do sistema e seus relacionamentos. Mostra uma visão estática das partes de um sistema através da implementação de um caso real ou protótipo.
- **Diagrama de classes** (*Class diagram*): apresenta as classes que compõem um sistema e identifica seus relacionamentos. Constitui uma visão estática das partes de um sistema.
- **Diagrama de interação** (*Interaction diagram*): apresenta um conjunto de objetos e suas interações (mensagens que podem ser trocadas entre eles). Este tipo de diagrama constitui uma visão dinâmica do sistema. Existem dois tipos de diagramas de interação:
 - Diagrama de Seqüência (*Sequence diagram*): apresenta uma ordenação no tempo das mensagens trocadas entre objetos.
 - Diagrama de Colaboração (*Collaboration diagram*) enfatiza na organização estrutural dos objetos que enviam e recebem mensagens.
- **Diagrama de estados** (*Statechart diagram*): indica a dinâmica de cada classe. Este diagrama é similar a uma máquina de estados.
- **Diagrama de atividade** (*Activity diagram*): enfatiza na seqüência de atividades que acontecem em um sistema. Este diagrama especifica comportamentos internos de paralelismo. Constitui uma visão dinâmica do sistema.
- **Diagrama de componentes** (*Component diagram*): Apresenta um conjunto de componentes e suas dependências. Este diagrama está relacionado como diagrama de classes, pois um componente mapea

uma ou mais classes e interfaces. Constitui uma visão estática da implementação de um sistema.

- **Diagrama de disposição** (*Deployment diagram*): indica a configuração e alocação dos nós do sistema. Apresenta uma visão estática da arquitetura do sistema.

Os tipos de relações definidas nestes diagramas incluem:

- **ASSOCIAÇÃO:** é uma relação que descreve um conjunto de vínculos entre elementos do modelo. Uma associação pode ser: (Deboni, 2003); (Furlan, 1998):
 - Agregação: é uma forma especial de associação utilizada para mostrar que um tipo de objeto é composto, pelo menos em parte, de outro em uma relação de todo/parte.
 - Composição: é uma forma de agregação com uma relação de propriedade da parte com o todo.
- **DEPENDÊNCIA:** nesta relação a mudança na definição de um elemento pode afetar a semântica dos elementos que dependam dele.
- **GENERALIZAÇÃO:** é um relacionamento entre um elemento mais geral e um elemento mais específico que é completamente consistente com o primeiro elemento somando a este informação especializada. É também chamado de especialização ou herança.

2.6.3. RELAÇÃO ENTRE UML E ODP

Existem trabalhos que usam a UML como linguagem para a especificação dos pontos de vista do ODP, pois as notações deste facilitam a migração de um ponto de vista para outro (Blanc, 1999); (Becerra, 1998); (Kandé et al., 1998).

No trabalho de Costa et al.(2001) foram aplicados vários dos diagramas da UML (de casos de uso, de classes, de seqüência, de estados, de atividades e de componentes) para a modelagem de um sistema de informação através dos pontos de vista empresa, informação e computação. Destaca-se neste trabalho que a UML além de prover uma notação integrada para representar o sistema de informação,

permite a migração natural entre as fases do projeto deste. Dentro desta proposta, os diagramas de casos de uso, de componentes e de seqüência são usados para a modelagem do ponto de vista de empresa, enquanto que os diagramas de classe e os diagramas de estados são usados para o pontos de vista de informação. O ponto de vista de computação é modelado através do uso dos diagramas de seqüência e os diagramas de atividades.

No trabalho de Kandé et al. (1998), é proposto que se utilize para cada ponto de vista da ODP um diagrama apropriado da UML, de modo a projetar um sistema de forma consistente e com alta rastreabilidade dentro dos pontos de vista.

No entanto, a pesar das vantagens da UML, alguns trabalhos têm sido desenvolvidos relacionando a UML e a Rede de Petri de maneira a suportar a análise formal dos modelos gerados nesta linguagem (Giese et al., 1999); (Douglass, 1998); (Baresi; Pezze, 1998). A principal característica da rede de Petri é possuir uma semântica formal efetiva para modelar e analisar uma grande variedade de sistemas. No Anexo B são apresentadas a principais características desta técnica.

Para o caso do presente trabalho considera-se especificamente a rede de Petri porque além da característica já citada esta constitui uma técnica de comprovada eficiência para tratar sistemas que possuem um comportamento dinâmico definido através das mudanças de estados discretos devido à ocorrência de eventos instantâneos (Sistemas a Eventos Discretos -SED-) (Miyagi, 1996); (Ramadge; Wonham, 1989). Dentro desta caracterização emolduram-se os sistemas prediais (entre cujos eventos podem considerar-se por exemplo: abertura e fechamento de portas, acionamento e desligamento de lâmpadas, chamada de elevadores, etc) para os quais vários trabalhos usando a rede de Petri tem sido realizados com sucesso (Bastidas, 2000); (Villani, 2000) (Gomes, 1997).

2.7. CONSIDERAÇÕES DO CAPÍTULO

Neste capítulo foram apresentadas algumas das definições existentes sobre edifícios inteligentes e foram consideradas suas principais características e objetivos. Neste âmbito ressaltou-se a importância da integração entre os sistemas que o

compõem, como um meio para atingir suas metas de produtividade, segurança e baixo custo.

O capítulo tratou também sobre os sistemas distribuídos e abertos os quais se fundamentam em uma arquitetura que suporta um conjunto consistente de padrões internacionais, que visam garantir a interoperabilidade e portabilidade das aplicações. Dentro deste contexto e considerando que um sistema de automação com as características dos sistemas abertos e distribuídos abrange uma efetiva integração dos diversos sistemas envolvidos, assegurando a flexibilidade e interoperabilidade entre seus elementos, este trabalho aborda os edifícios inteligentes e seus sistemas prediais através deste conceito.

Considerando que os sistemas distribuídos e abertos requerem para sua concepção e operação técnicas efetivas para modelagem e análise, este capítulo apresentou também o ODP como um modelo de referência que permite, através de diversas abstrações, criar um sistema com características de abertura, distribuição, integração, flexibilidade e modularidade. A aplicação do ODP é realizada através do uso dos conceitos da orientação a objetos (o que permite entre outras coisas a estruturação do processo de modelagem do sistema) e especificamente através de uma linguagem que considere estes conceitos. Assim, dentro das linguagens existentes, escolheu-se a UML por representar diferentes perspectivas de um mesmo sistema e por ser atualmente um padrão de fato. Para realizar a análise formal considera-se o uso das redes de Petri por ser esta técnica matemática e gráfica de grande eficácia para a modelagem e análise de sistemas a eventos discretos, dentro dos quais pode-se caracterizar os sistemas prediais.

3. MODELAGEM E ANÁLISE DE SISTEMAS DISTRIBUÍDOS E ABERTOS

Neste capítulo é apresentado o formalismo que está sendo proposto para o suporte à modelagem e análise de sistemas distribuídos e abertos para automação predial. Este formalismo é derivado da aplicação dos conceitos de orientação a objetos à rede de Petri.

3.1. ESCOLHA DO FORMALISMO

A modelagem é uma das mais importantes atividades a serem realizadas quando se projeta ou se modifica um sistema. A escolha da técnica que melhor pode representar em um modelo as características do sistema é de grande importância e define além da forma de realizar a análise do sistema, a maneira como o projeto e a proposta final serão apresentados.

Neste contexto e como foi apresentado no capítulo anterior (item 2.5) o ODP tem um papel especial pois permite modelar através de seus cinco pontos de vista um sistema distribuído e aberto. No entanto, não estipula nenhuma linguagem para realizar esta tarefa apesar de estar claro que a linguagem escolhida deve representar adequadamente os conceitos de orientação a objetos, pois o ODP é baseado neste paradigma. Por outro lado a orientação a objetos não conta com um método formal abrangente e totalmente aceito para analisar os modelos gerados. Assim, vários

trabalhos têm considerado este aspecto propondo técnicas de modelagem e análise formal dentro do ODP baseadas principalmente em LOTOS, SDL e Z (Diagne, 1996). Neste contexto e, considerando os trabalhos que relacionam as redes de Petri com a orientação a objetos (Giese et al., 1999); (Douglass, 1998); (Baresi; Pezze, 1998) de modo que, por um lado a procura-se uma maior estruturação e reutilização dos modelos gerados e de outro lado que estes mesmos modelos possam suportar a análise formal. Além disso, sua comprovada efetividade para modelar e analisar sistemas a eventos discretos e especificamente sistemas prediais (Villani, 2004); (Bastidas, 2000); (Villani, 2000) fazem desta abordagem a escolhida como formalismo base neste trabalho. (O anexo B apresenta os fundamentos básicos da rede de Petri, que permitem uma maior compreensão desta técnica).

De maneira geral, existem diferentes formas de realizar a associação entre rede de Petri e orientação a objetos. Segundo Zapf; Heinzl (1999) e Bastide (1995), as propostas neste sentido podem classificar-se em:

A. Integração de conceitos de orientação a objetos dentro das redes de Petri

Neste tipo de rede as MARCAS representam objetos que modelam as propriedades estáticas do sistema (estrutura de dados) enquanto que a rede controla a dinâmica do sistema (estrutura de controle do sistema). As MARCAS da Rede de Petri são consideradas instâncias de classes descritas em alguma linguagem orientada a objetos. Através do disparo das TRANSIÇÕES, os atributos dos objetos podem ser lidos e manipulados ou seus métodos podem ser executados. O disparo de uma transição também pode criar e/ou destruir objetos. Um exemplo destas redes são: SimCon (*Simple Integrated Model for Complex Object Networks*) que procura ligar as propriedades estáticas e dinâmicas do sistema (Zapf; Heinzl, 1999) e THORNS (*Timed Hierarchical Object-Related Nets*), nesta rede a linguagem C++ é utilizada para a definição das classes e os objetos (Schof; Sonnenschein; Wieting, 1995).

B. Integração da rede de Petri dentro de técnicas de orientação a objetos

Neste tipo de rede o sistema é estruturado com base em técnicas orientadas a objetos, assim, primeiro os objetos relevantes ao sistema e suas relações são identificadas. Depois, a descrição do comportamento do objeto e a comunicação entre objetos é especificada com a ajuda da rede de Petri. Aqui a MARCAÇÃO da rede indica o estado de um objeto e as TRANSIÇÕES representam a execução de um método deste objeto. Assim, a rede modela a disponibilidade de métodos e as seqüências potenciais destes. Para a construção do modelo global de um sistema unem-se as redes que correspondem a cada objeto do sistema (descrevendo formas de comunicação e interação entre os objetos). Alguns exemplos são: OBM (*Object Behaviour Model*) (Zapf; Heinzl, 1999), o qual utiliza conceitos da orientação a objetos como: classes, encapsulamento, herança e polimorfismo; HOOD (*Hierarchical Object Oriented Design*) (Paludetto, 1991); G-CPN (*G-Coloured Petri Net*) derivada da união da Rede de Petri coloridas e as G-net (Perkusich; Perkusich; Chang, 1996); OOPN (*Object-Oriented Petri Net*) (Wang, 1996) e, CO (*Cooperative Objects*) baseadas na Rede de Petri colorida (Bastide et al. 1995, 1999).

C. Integração entre técnicas de orientação a objetos e rede de Petri

Este tipo de rede mistura as duas abordagens anteriores e pode ser considerado como uma evolução dos modelos de rede embarcados dentro dos objetos. Neste modelo, inicialmente os objetos são usados para determinar a estrutura do sistema, a seguir o comportamento dos objetos é modelado com a ajuda de redes. No entanto, as redes não precisam carregar MARCAS indistintas, estas podem conter referências a outros objetos cujo comportamento também é modelado através de uma rede e assim sucessivamente. Este processo de aninhamento facilita modelos de múltiplas camadas com comportamentos concorrentes. Nesta abordagem, tanto a marca como a rede são objetos e podem ser utilizados em qualquer contexto (Zha, 2000), assim, o modelador é livre para escolher como as várias atividades serão compostas determinando se um objeto será passivo ou ativo (Lakos, 1995a). Alguns exemplos deste tipo de rede são as: PN-TOX (*Petri Net Tool for Object Concurrency specification*) (Zapf; Heinzl, 1999), que oferece a capacidade de utilizar diferentes tipos de rede e a OPN (*Object Petri Net*) (Valk, 1998, 2000); (Lakos, 1995a, 1995b,

1996), que está baseada na Rede de Petri colorida e pode ser traduzida para estas. A rede OPN procura integrar os conceitos de orientação a objetos à rede de Petri incluindo polimorfismo e herança.

Da consideração das propostas para relacionar a orientação a objetos e rede de Petri, para o presente trabalho destacam-se os seguintes aspectos:

- é desejável que seja possível construir um modelo global do sistema através da interação das redes que representam o comportamento dos objetos que o compoem;
- os objetos devem poder interagir mantendo seu encapsulamento;
- não é aconselhável a incorporação de regras complexas que comprometam a visualização gráfica do modelo em rede de Petri.

Da análise dos diferentes trabalhos citados anteriormente e dentre as características dos tipos de redes de Petri existentes, escolheu-se como técnica a ser usada a rede de Petri colorida. Esta escolha deveu-se a que esta rede apresenta características adequadas para a co-utilização com diferentes abordagens.

Entre as características principais desta rede estão as seguintes (Jensen, 1996):

- conta com uma semântica que define sem ambigüidade o comportamento da rede;
- permite modelar processos assíncronos complexos;
- suas MARCAS possuem atributos (MARCAS distinguíveis ou “coloridas”) que podem estar associadas a diferentes informações como: o processo, o controle, etc. Os atributos podem assumir qualquer tipo¹;
- a coloração da rede permite representar estruturas de dados;
- pode incorporar o conceito de tempo permitindo a validação de propriedades como os tempos de resposta, os atrasos, etc;

O tipo determina os valores que um atributo pode ter (representação interna destes valores) e as operações que podem ser realizadas com ele (Cantù, 1995).

- permite descrever uma grande variedade de sistemas de natureza distinta;
- descreve tanto o estado como o comportamento;
- possui métodos de análise formal que permitem a verificação das propriedades da rede.
- permite construir modelos adequados à expressão e avaliação de propriedades

Neste contexto e considerando também as características levantadas por Holvet e Verbaeten (2001) para os sistemas distribuídos e abertos a abordagem escolhida deve ser capaz de descrever:

- Concorrência: as atividades concorrentes que podem ser inter-objetos (ocorre quando os objetos comunicam-se) ou intra-objetos (ocorre da execução concorrente de métodos internos);
- Autonomia: quando não existe a priori uma relação mestre/escravo ou escravo/mestre entre as entidades concorrentes;
- Heterogeneidade: quando a arquitetura de computadores, as redes de comunicação, os sistemas operativos e até as linguagens de programação podem ser heterogêneos.

Introduz-se assim, uma nova rede que permita considerar estas características e os conceitos de orientação a objetos à rede de Petri colorida e ainda leve em conta os aspetos mais relevantes do ODP para realizar a modelagem e análise de sistemas distribuídos e abertos.

3.2. REDE DE PETRI MODULAR COLORIDA BASEADA EM OBJETOS (RPMCO)

Com o intuito de aumentar ainda mais as potencialidades da rede de Petri colorida e visando a sua utilização para a modelagem e a análise de sistemas distribuídos e abertos para integração de sistemas prediais, é introduzida uma

extensão desta baseada na aplicação dos conceitos de objetos. Esta rede é denominada Rede de Petri Modular Colorida baseada em Objetos (RPMCO).

Antes de apresentar a definição formal da rede proposta tratam-se os conceitos considerados nela:

a). Objetos e Classes

Em geral as técnicas orientadas a objetos permitem a estruturação de um sistema distribuído em entidades que gerenciam recursos e que interagem entre si. A estrutura de uma entidade é inteiramente determinada pelos princípios de encapsulação e de interface. Assim, um objeto encapsula seu estado e seu comportamento e a única forma para modificar o estado de um objeto é através dos métodos disponibilizados na sua interface.

b). Módulo

Em conformidade com o paradigma da orientação a objetos pode-se considerar um módulo como uma classe que oferece serviços para sua manipulação e a partir da qual se podem criar instâncias que têm similitudes de comportamento.

O serviço pode ser definido como o grupo de operações (no sentido de métodos de uma classe) subordinados às restrições de composição como a precedência e a escolha. Os serviços utilizados por um módulo devem ser especificados assim como os eventuais processamentos de exceção associados a sua invocação.

Um serviço oferecido é completamente especificado pela lista das operações que lhe compõem. Um serviço utilizado é completamente especificado pela lista das operações e os processamentos de exceção eventualmente associados a certos resultados.

As instâncias de um módulo permitem modelar diferentes políticas de prestação de serviços.

Cada módulo tem autonomia de comportamento e objetivos próprios ou deve atender aos objetivos que lhe são demandados.

A interface de um módulo tem uma abstração de seu ambiente² ao representar o que é atendido pelo ambiente e o que este requisita.

- *Autonomia de um módulo* A autonomia de um módulo é fundamental na especificação de sistemas distribuídos e neste caso particularmente dos sistemas prediais. Para isto o módulo deve ser capaz de reagir aos eventos locais de maneira independente do ambiente. Estes eventos podem desencadear processamentos autônomos isto é, não invocáveis pelo ambiente, mas que podem iniciar interações com o este. Os módulos devem também ser capazes de tratar processamentos de exceção para continuar a interações com seu ambiente. Os processamentos de exceção são necessários para garantir às interações, isto é, são meios para isolar o módulo em relação aos comportamentos de seu ambiente que se julga como erro. Isto é importante no contexto deste trabalho porque uma das características de um sistema predial é a determinação dos seus comportamentos em função das interações com os outros sistemas prediais. Um módulo deve ser capaz de avaliar seu estado atual e desencadear os processamentos locais e as interações com seu ambiente.
- *Processamentos de um módulo* O comportamento de um módulo está definido pelos seus processamentos e suas interações. Os serviços oferecidos e utilizados especificam as dependências entre os processamentos dos diferentes módulos

Os processamentos de um módulo são classificados em função da sua semântica em dois tipos:

- o Transações: podem ser invocadas a partir do ambiente. Estas podem ser interrogativas e portanto, produzir um resultado ou ser enunciativas.

² Entende-se ambiente como o conjunto de todos os outros módulos do sistema.

- o De exceção: não podem ser invocadas localmente. Estas constituem derivações do fluxo de controle que seguem às interações.

c). Interação

É uma relação dinâmica entre dois módulos que segue o princípio de ação e reação dentro dos limites permitidos pela autonomia de cada um dos participantes sendo que um assume o papel de servidor e o outro de cliente. Os critérios de correção de uma interação são estabelecidos por um compromisso entre as restrições impostas pelo servidor para seu acesso e o objetivo expressado pelo cliente. As interações que respeitam tais critérios são ditas válidas.

Um módulo pode ser visto como uma entidade encapsulada e parcialmente parametrizada. Uma interação toma uma parte destes parâmetros nos dados transmitidos em uma invocação e desencadeia um funcionamento (parcial) do módulo. Dois módulos que interagem constituem uma entidade temporária que é uma abstração de uma funcionalidade no sistema. Assim, a interação é um meio de composição entre módulos que permite garantir algumas das funcionalidades do sistema.

Para poder especificar o comportamento de um módulo a partir de seus processamentos é necessário conhecer suas interações.

Distinguem-se três tipos de interação nos sistemas distribuídos:

- demanda de serviço: corresponde ao fornecimento dos dados necessários à realização do serviço. Esta interação faz com que o servidor produza um resultado que deve ser esperado pelo cliente.
- notificação: corresponde a uma interação na qual se espera a ocorrência de um evento. Esta interação não produz resultado.
- intercâmbio contínuo de dados: corresponde ao fornecimento de dados sobre um meio de transmissão e funcionando como um fluxo de dados compartilhado para leitura/ escrita. Esta interação não produz resultado.

A notificação corresponde à noção de sinal (definido pelo ODP como “uma ação atômica resultante da comunicação em um sentido de um objeto inicializador a um objeto que aceita esta” (ISO/IEC 10746-1, 1998))

As interações válidas permitem representar o comportamento dos módulos e do sistema. Existe um paralelismo inter-módulos no sentido que um módulo servidor pode estar em interação com mais de um módulo cliente e ao mesmo tempo ter outras interações com outros módulos para as quais é cliente. Assim, quando se especifica cada módulo estabelecem-se as seqüências de interação que ele autoriza a seu ambiente e, estas seqüências determinam os serviços oferecidos. Um módulo pode utilizar os serviços oferecidos por outros módulos, estes serviços constituem então os serviços utilizados. Os serviços oferecidos e utilizados constituem as obrigações que determinam a validade das interações.

As interações com o ambiente são subordinadas aos critérios de avaliação que fazem com que seus resultados sejam aceitos ou não. Em caso de não aceitação, o módulo deve desencadear, através da realização de processamentos, as ações necessárias a ser seguidas a partir deste erro de interação.

- ***Serviços requeridos e a abstração do ambiente:*** Cada invocação de uma operação de um serviço utilizado, deve ter em conta se o servidor está em estado de processar a operação (restrições de seqüência de operações). O não respeito das restrições é considerado como um erro na modelagem. Se as restrições são respeitadas, o resultado (no caso de uma interrogação) produzido pelo servidor leva em conta o esperado pelo cliente e as eventuais exceções se as houver.
- ***Mecanismos de modularização*** Existem mecanismos que permitem aos módulos se unirem em sistemas, estes são:
 - ***Composição estrutural:*** Um laço estrutural é uma associação conceptual entre duas entidades (Rumbaugh, 1991). Este tipo de laço denota uma certa visão de um sistema que pode ou não influenciar sua dinâmica. Por exemplo um laço de agregação entre duas entidades estabelece uma semântica ligada à dinâmica porque a composição utiliza as funcionalidades do

módulo. Pelo contrario um laço de generalização/especialização não inclui necessariamente semântica ligada ao controle.

- o *Composição funcional*: Cada módulo define na sua interface as interações as quais ele se compromete a participar. Esta definição se faz em duas fases:
 - definir os serviços oferecidos pelo módulo que especificam as restrições que ele impõe ao ambiente;
 - importar do ambiente os serviços oferecidos pelos outros módulos e especificar o esperado ao módulo requisitado a respeito destes serviços.

O fato de impor restrições na especificação dos módulos permite validar todas as interações no sistema e evitar assim, a ocorrência de erros no sistema. Estas restrições servem de filtro determinando as interações válidas entre todas as que são possíveis.

3.2.1. ODP E RPMCO

A RPMCO considera alguns dos principais conceitos adotados no ODP para a modelagem e a análise dos sistemas distribuídos e abertos.

O ODP é um modelo genérico que não considera uma especificação particular para uma classe dada de sistemas distribuídos:

“The RM-ODP is generic, that is, independent of, and equally applicable to, arbitrary application domains making use, of requiring distributed system technology. Specific application domains may consider refining and specializing the RM-ODP to suit their particular needs, resulting in a model and standards for the realization of functions and components identified in the specific reference model” (ISO/IEC 10746-1, 1998).

Assim, com base no enunciado anterior este trabalho estuda a modelagem da interação distribuída dos sistemas prediais por utilização de serviços.

As interações sobre os objetos são apoiadas na noção de método. O ODP tem proposto três tipos de interação (item 2.5.2 B) que tem por finalidade um maior controle destas e das suas restrições.

A RPMCO tem em conta as expressões dos pontos de vista da informação e da computação do ODP. Nesta abordagem eles se tratam de maneira integrada pois eles não podem ser considerados independentemente um de outro.

Os processamentos constituem a visão funcional sobre os módulos de um sistema. Estes são a especialização do ponto de vista computacional do ODP na abordagem por módulos ao utilizar-se da repartição lógica das funcionalidades do sistema sobre seus módulos.

A RPMCO não considera o conceito de herança porque se busca uma descrição dos módulos e da sua interação (Matsuoka, 1993).

3.2.2. CARACTERÍSTICAS DA RPMCO

A seguir apresentam-se as principais características desta rede:

A RPMCO considera princípios de abstração, modularização e encapsulamento. Na modelagem realizada usando a RPMCO considera-se que um sistema complexo pode ser percebido como resultante da composição de módulos e a cooperação entre eles. Desta forma, descreve-se o sistema global através de módulos e suas conexões através das interfaces destes. Cada módulo tem uma estrutura independente. Esta é uma característica particularmente útil quando se considera que um sistema abordado através de módulos facilita e aumenta a rastreabilidade de sua especificação.

Cada módulo representa uma classe com seus estados e comportamentos (isto é, uma estrutura de controle interna de cada classe a qual pode ser implementada de maneira seqüencial ou concorrente) e define um grupo de métodos oferecidos e/ou utilizados de ou para outros módulos do sistema (cada módulo se encontra sobre estímulos de eventos externos e internos).

Em geral a organização em módulos permite que estes sejam convenientemente combinados de modo a suportarem de forma eficaz o desenvolvimento de sistemas complexos.

Um módulo especifica a parte observável de um sistema. Dentro de um módulo podem ser realizadas as seguintes transformações básicas:

- Refinamento: procura identificar mensagens de entrada e saída, definir concretamente os serviços oferecidos e utilizados, determinar as relações e/ou atributos relacionadas aos serviços;
- Abstração: produz módulos menos específicos que definem os mesmos serviços. Isto permite o re-uso da especificação ao deixar que o mesmo modelo possa ser usado em diferentes níveis de precisão;
- Composição: permite construir um módulo a partir de módulos menores. Aqui o módulo resultante deve satisfazer a especificação requerida dado que os módulos menores também satisfazem as suas respectivas especificações;
- Decomposição: permite dividir um módulo em módulos menores. Aqui deve assegurar que os requerimentos do módulo maior sejam atendidos pelos módulos menores.

As transformações podem ser aplicadas através das suas combinações.

Os módulos devem encapsular a estrutura interna a fim de restringir o acesso desde módulos externos. A encapsulação é garantida devido a acessibilidade de cada módulo que é restrita aos métodos oferecidos.

Os módulos de um sistema são modelados através de uma RPMCO o que permite analisar suas propriedades comportamentais.

3.2.3. ESTRUTURA DE UM MÓDULO EM RPMCO

Cada módulo da RPMCO é composto estruturalmente por duas subestruturas:

1. Interface³ do Módulo - IM: determina a visão do módulo para o resto do sistema. Nela estão declarados os atributos (dados sobre os quais o módulo atua), métodos encapsulados (serviços oferecidos pelo módulo), os métodos que o módulo utiliza (serviços oferecidos por outros módulos) e as restrições que permitem encapsular adequadamente os dados. Estas informações devem assegurar sua interoperabilidade.

Também se expressam as obrigações de utilização que permitem realizar as interações descrevendo os comportamentos admissíveis.

Aqui se expressam as propriedades em nível de módulos e as hipóteses sobre o ambiente que permitem avaliar o que pode ser efetuado pelo módulo e seu ambiente. As hipóteses de um módulo a seu ambiente devem ser consideradas nas propriedades dos módulos deste ambiente.

Somente através da interface do módulo os outros módulos do sistema podem ter acesso aos métodos e atributos encapsulados dentro do módulo (este proporciona um grupo de serviços e operações –métodos- que podem ser invocados por outros módulos do sistema) e por sua vez apenas através da interface do módulo este é capaz de acessar os serviços de outros módulos.

2. Unidade Comportamental – UC : é a estrutura interna que especifica a abstração dos dados da interface. Aqui é modelado o comportamento interno do módulo. Esta modelagem é realizada através de uma rede de Petri colorida. Nesta abordagem, considera-se a rede de Petri colorida baseada em objetos, onde a sua estrutura de dados interna é definida por sua cor. Assim, a MARCA é usada para reter os dados do módulo, a MARCAÇÃO da rede representa o estado deste e as TRANSIÇÕES descrevem a execução de seus métodos (para o qual opera com os dados do objeto – MARCA-). Desta forma, a rede de Petri modela a disponibilidade de métodos e os estados a potencial seqüência de execução destes. Em geral a rede ilustra o fluxo de controle e o fluxo de dados ao indicar o fluxo de MARCAS através desta. Cada método

³ Interface é uma coleção de operações que especifica os serviços disponibilizados por uma classe e as formas de interação de um objeto desta classe com outros objetos do sistema (Booch et al., 1999).

definido na interface corresponde a uma funcionalidade interna do módulo que é expressada na unidade comportamental.

Para a construção do modelo global de um sistema unem-se as redes que correspondem a cada unidade comportamental de cada módulo do sistema (descrevendo as formas de interação entre os módulos).

Um módulo especifica suas propriedades e adota hipóteses sobre seu ambiente. As propriedades do módulo devem ser verificáveis qualquer que seja a evolução dos estados do módulo. As hipóteses devem ser validadas pelas propriedades de outros módulos do ambiente ou a composição de tais propriedades. As hipóteses permitem completar a abstração do ambiente de um módulo

As propriedades e hipóteses que são consideradas procuram:

- encontrar estados de erro como os estados bloqueantes (Um estado bloqueante é aquele em que um ou mais módulos do sistema não asseguram algumas de suas funcionalidades);
- provar a validade da concorrência e do paralelismo entre os módulos como a exclusão mútua e os erros de resposta nas interações;
- corretude de um servidor por relacionar os diferentes clientes que podem interagir com ele de maneira independente os uns dos outros.

Assim, a RPMCO introduz uma estruturação de sistemas distribuídos de uma maneira mais restritiva que as resultantes geralmente das metodologias orientadas a objetos

Um exemplo simples da representação gráfica e dos conceitos básicos da rede RPMCO é apresentado na Figura 3.1. Nesta rede o comportamento de um módulo (classe) é modelado na unidade comportamental. Aqui define-se também a cor de seus lugares que no caso do exemplo é MSG o qual limita o acesso ao lugar somente a marcas desta cor. Na interface do módulo são indicadas as TRANSIÇÕES que

representam os métodos que este oferece (método 1) ou utiliza (método 2) e as suas restrições.

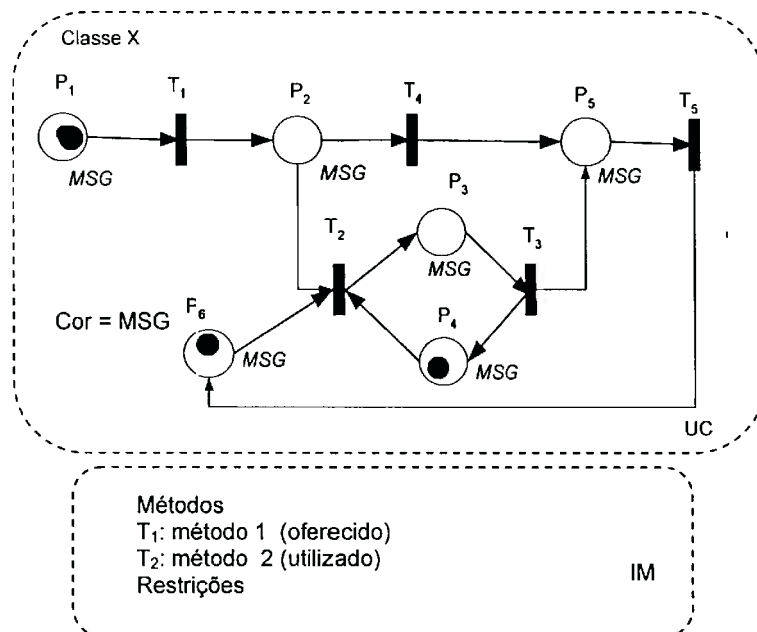


Fig. 3.1 Representação gráfica de uma classe em RPMCO

3.2.4. DEFINIÇÃO DE RPMCO

A seguir é apresentada a formalização da RPMCO. Assim, inicialmente serão apresentadas as definições matemáticas básicas usadas:

Um *multi-set* é intuitivamente um conjunto, mas neste caso podem existir múltiplas aparições de um mesmo elemento dentro deste.

Definição 3.1: Um multi-set m sobre um conjunto não vazio de S , é uma função $m \in [S \rightarrow \mathbb{N}]$. O inteiro não negativo $m(s) \in \mathbb{N}$ é o número de vezes que aparece o elemento s no *multi-set* m . Usualmente representado pela soma formal:

$$m = \sum m(s) \cdot s$$

$$s \in S.$$

Por notação é colocado o símbolo ` entre cada coeficiente e o elemento do *multi-set*

Por S_{MS} denota-se o conjunto de todos os *multi-sets* sobre S

Definição 3.2: as operações adição, multiplicação escalar, comparação e tamanho de *multi-sets* são definidos conforme apresentado na tabela 3.1 para $m, m_1, m_2 \in S_{MS}$ e todo $n \in \mathbb{N}$.

Tabela 3.1 Propriedades dos *multi-sets*

OPERAÇÕES		DEFINIÇÃO
Adição	$m_1 + m_2$	$\sum_{s \in S} (m_1(s) + m_2(s)) \cdot s$
Multiplicação escalar	$n * m$	$\sum_{s \in S} (n * m(s)) \cdot s$
Comparação; \geq e $=$ são definidos analogamente para \leq	$m_1 \neq m_2$ $m_1 \leq m_2$	$\exists s \in S: m_1(s) \neq m_2(s)$ $\forall s \in S: m_1(s) \leq m_2(s)$
Tamanho	$ m $	$\sum_{s \in S} m(s)$ Quando $ m = \infty$ se diz que m é infinito. Caso contrário m é finito
Subtração	$m_1 - m_2$	$\sum_{s \in S} (m_2(s) - m_1(s)) \cdot s$ Quando $m_1 \leq m_2$

A semântica e a sintaxe da linguagem usada nas expressões da RPMCO seguem as mesmas normas estabelecidas na rede de Petri colorida definida em Jensen (1996).

Definição 3.3: A rede de Petri é um grafo bipartido constituído de um conjunto de LUGARES P , um conjunto de TRANSIÇÕES T e arcos que conectam LUGARES a TRANSIÇÕES e vice-versa (Pre, Pos). Os conjuntos de LUGARES e TRANSIÇÕES são disjuntos. Sendo dado um LUGAR p ou uma TRANSIÇÃO t adota-se a seguinte notação:

$$(i) \bullet p = \{t \in T, \text{Pre}(p,t)\} \text{ e } p \bullet = \{t \in T, \text{Pos}(p,t)\},$$

(ii) $\bullet t = \{p \in P, \text{Pre}(p,t)\}$ e $t\bullet = \{p \in P, \text{Pos}(p,t)\}$.

A seguir apresenta-se a definição formal da RPMCO.

Definição 3.4: Uma rede de Petri Modular Colorida baseada em Objetos – RPMCO, N_{RPMCO} é uma tupla $\langle UC, \text{Ident}, IM \rangle$ onde:

UC: é a Unidade Comportamental e constitui uma rede de Petri colorida (Jensen, 1992) a qual é uma tupla $\langle \Lambda, P, T, A, L, W, G, E, J \rangle$ onde:

Λ é o conjunto finito e não vazio de cores

P é um conjunto finito de LUGARES.

T é um conjunto finito de TRANSIÇÕES.

A é um conjunto finito de ARCOS tal que:

$$P \cap T = P \cap A = T \cap A = \emptyset$$

L : é uma função nó. que relaciona cada ARCO a um par onde o primeiro elemento é a origem do nó e o segundo é destino do nó. Os dois nós têm de ser diferentes (um LUGAR e uma TRANSIÇÃO ou vice-versa). Assim, define-se o conjunto de ARCOS A como um conjunto separado de nós.

$$L: A \rightarrow P \times T \cup T \times P$$

W : é uma função denominada conjunto de cores que associa a cada LUGAR p de P uma cor pertencente a Λ , mapeando cada LUGAR, p , para cada conjunto de cores $W(p)$. Intuitivamente, isto significa que cada MARCA em p deve pertencer ao tipo $W(p)$.

$$W: P \rightarrow \Lambda.$$

G : é uma função, denominada função “guarda”. Esta função relaciona cada TRANSIÇÃO t a uma expressão de tipo booleana, isto é um predicado. Entretanto, toda variável em $G(t)$ deve ter tipos que pertencem a Λ . G é definida em T com expressões tal que:

$$\forall t \in T: [\text{Tipo}(G(t)) = \text{BOOLEANA} \wedge \text{Tipo}(\text{Var}(G(t))) \subseteq \Lambda]$$

E: é uma função expressa no ARCO. Esta relaciona cada ARCO com uma expressão que deve ser do tipo $W(p(a))_{MS}$. Isto significa que o valor de cada expressão deve produzir um *multi-set* no conjunto de cores que é anexado ao correspondente LUGAR. E é definida a partir de A em expressões tal que:

$\forall a \in A: [\text{Tipo}(E(a)) = W(p(a))_{MS} \wedge \text{Tipo}(\text{Var}(E(a))) \subseteq \Lambda]$ onde $p(a)$ é o LUGAR de $L(a)$.

$W(p)_{MS}$ é o conjunto de todos os *multi-sets* definidos sobre W.

J: é uma função de inicialização. Esta é definida a partir de P em expressões fechadas (sem variáveis) tais que:

$\forall p \in P: [\text{Tipo}(J(p)) = W(p)_{MS}]$

Ident: identificador único de maneira que para cada RPMCO diferentes N_{RPMCO_1} e N_{RPMCO_2} se tem: $\text{Ident}_1 \cap \text{Ident}_2 = \emptyset$

IM é a Interface do Módulo a qual é uma dupla $\langle MT, \text{Rest} \rangle$ onde:

MT é um conjunto finito e não vazio de métodos (me).

MT_U é o conjunto de métodos utilizado pelo módulo.

MT_O é o conjunto de métodos oferecidos pelo módulo

$$MT_U \cap MT_O = \emptyset$$

Comportamento da RPMCO

Para definir o comportamento de uma RPMCO introduz-se a seguinte notação:

$\forall t \in T: \text{Var}(t) = \{v \mid v \in \text{Var}(G(t)) \vee \exists a \in A(t): v \in \text{Var}(E(a))\}$

$\forall (x_1, x_2) \in (P \times T \cup T \times P): E(x_1, x_2) = \sum E(a)$ onde $a \in A(x_1, x_2)$

Onde, $\text{Var}(t)$ é chamado de conjunto de variáveis de t enquanto que $E(x_1, x_2)$ é chamado de expressão de (x_1, x_2) . A somatória indica a adição das expressões e é definida porque toda expressão participante tem um tipo de *multi-set* comum.

Definição 3.5: Um vínculo (neste contexto vínculo refere-se a outorgar um valor à variável que aparece expressada nos arcos) de uma TRANSIÇÃO t é uma função b definida em $\text{Var}(t)$, tal que:

$$\forall v \in \text{Var}(t): b(v) \in \text{Tipo}(v) \wedge G(t) \langle b \rangle.$$

$B(t)$ é chamado o conjunto de todos os vínculos para t

Definição 3.6: Uma MARCA é um par (p, c) onde $p \in P$ e $w \in W(p)$, enquanto um elemento vinculado é um par (t, b) onde $t \in T$ e $b \in B(t)$. O conjunto de todas as MARCAS é denotado por TE , enquanto que o conjunto de todos os elementos vinculados é denotado por BE .

Definição 3.7: Uma MARCAÇÃO de uma RPMCO é um *multi-set* em TE enquanto uma etapa é um *multi-set* não vazio e finito sobre BE . A MARCAÇÃO inicial M_0 é uma marcação obtida pela avaliação da expressão de inicialização:

$$\forall (p, w) \in TE: M_0(p, w) = (J(p))(w)$$

Os conjuntos de todas as MARCAÇÕES e etapas são denotadas por M e Y respectivamente.

Definição: 3.8: Uma etapa Y é habilitada em uma marcação M se e somente se as seguintes propriedades forem satisfeitas:

$$\forall p \in P: \sum E(p, t) \langle b \rangle \leq M(p), \text{ onde } (t, b) \in Y$$

Tem-se que para todo ARCO de entrada da TRANSIÇÃO t , o resultado da expressão de cada um destes ARCOS deve ser um *multi-set* pertencente (menor) ao *multi-set* existente em cada LUGAR p de origem de cada ARCO.

Caso a expressão do arco não seja coerente com a função de “guarda” a TRANSIÇÃO não é habilitada.

Definição 3.9: Quando uma etapa Y é habilitada em uma MARCAÇÃO M_1 , esta etapa pode ocorrer, mudando a marcação M_1 para outra MARCAÇÃO M_2 , definida por:

$$\forall p \in P: M_2(p) = (M_1(p) - \sum E(p, t) \langle b \rangle) + \sum E(t, p) \langle b \rangle \text{ onde } (t, b) \in Y$$

A primeira soma envolve as MARCAS removidas, enquanto a segunda envolve as de MARCAS adicionadas na RPMCO.

Neste caso, M_2 é dita diretamente alcançável por M_1 pela ocorrência da etapa Y , a qual também denota-se por: $M_2 [Y] M_1$

Definição 3.10: Uma seqüência finita de ocorrências é uma seqüência de MARCAÇÕES e etapas:

$M_1 [Y_1 > M_2 [Y_2 > M_3 \dots M_n [Y_n > M_{n+1}$ onde $n \in \mathbb{N}$ e $M_i [Y_i > M_{i+1}$ para todo $i \in 1..n$. A MARCAÇÃO M_1 é chamada de MARCAÇÃO inicial da seqüência ocorrida, enquanto a MARCAÇÃO M_{n+1} é chamada de MARCAÇÃO final. O inteiro não negativo n é chamado de número de etapas na seqüência ocorrida.

Definição 3.11: Uma MARCAÇÃO M'' é alcançável a partir de uma MARCAÇÃO M' se e somente se existe uma seqüência finita de ocorrências tendo M' como MARCAÇÃO inicial e M'' como MARCAÇÃO final, por exemplo: se para algum $n \in \mathbb{N}$ há uma seqüência de etapas $Y_1 Y_2 \dots Y_n$ tal que:

$$M'[Y_1 Y_2 \dots Y_n > M'']$$

Também diz-se que M'' é alcançável de M' em n etapas. O conjunto das MARCAÇÕES alcançáveis de M' são denotadas por $[M']>$.

Comunicação entre módulos

O paradigma cliente servidor é comumente usado em objetos quando estes inter-atuam (Tokmakkof, 1998). Na presente proposta este paradigma é usado considerando como cliente o objeto que realiza a chamada de métodos e como servidor o objeto que é chamado.

A comunicação considerada entre os módulos da RPMCO é baseada no uso de um espaço de dados (Holvoet; Verbaeten, 2001); (Tokmakkoff, 1998). Os módulos se comunicam de duas maneiras:

- Assíncrona: esta é modelada através de um lugar compartilhado (o meio de comunicação) e de uma MARCA (mensagem) que o cliente coloca para ser enviada ao servidor do qual ele está requerendo um serviço.
- Síncrona: a sua modelagem utiliza se de dois lugares compartilhados. No primeiro é colocado através de uma marca a requisição do serviço ao servidor e no segundo é colocada a resposta deste.

Estas duas formas de comunicação: assíncrona e síncrona em uma RPMCO, permitem modelar a forma de comunicação comum em sistemas distribuídos (assíncrona) e na maioria de linguagens de programação (síncrona) (Maier & Mold, 2001).

As mensagens usadas na comunicação seguem a seguinte estrutura:

Definição 3.12: A estrutura de uma mensagem (mes) é dada por $\langle \text{em}, \text{re} \rangle$, Dado, Oper> onde:

em: corresponde à identificação da classe que invoca (emissor)

re: é a identificação da classe que recebe a invocação (receptor)

Dado: refere-se à informação contida na mensagem

Oper: refere-se à operação que deve ser realizada

A cor de uma mensagem (MSG) é definida por:

$\text{Cor MSG} = \text{produto End} * \text{Dado} * \text{Oper};$

A cor End refere-se ao reconhecimento da identidade dos módulos envolvidos na comunicação e é definida por:

$\text{Cor End} = \text{produto Id} * \text{Id};$

Cor Id = com em, re;

3.2.5. ANALISE DA RPMCO

Em relação à análise de sistemas pode-se dizer que o aumento da complexidade dos sistemas de controle, das exigências sobre os custos, da confiabilidade e da segurança, têm enfatizado a necessidade de métodos de análise que garantam o funcionamento correto e seguro da solução adotada (com base na análise do comportamento do sistema através do seu modelo). Nestes métodos tanto a validação do modelo como sua verificação têm um papel fundamental.

Validação: consiste em assegurar que o modelo proposto para um sistema representa adequadamente suas funcionalidades. Isto assegura que a especificação é

completa, coerente e cumpre os objetivos fixados (Diagne, 1997). É a prova que determina se o modelo concorda com os requisitos do projeto e se o sistema se comporta de acordo com a sua especificação. A validação consiste em mostrar que o sistema não apresenta nenhuma propriedade inadequada que possa alterar o seu funcionamento.

Verificação: É a prova de que a semântica interna do modelo é correta independentemente da especificidade do sistema modelado. Por exemplo, as propriedades em geral investigadas nos modelos são: *deadlock*, estabilidade de seu comportamento dinâmico, existência de certos estados, etc.

A validação e a verificação permitem analisar se um sistema é consistente e correto.

Em geral, os métodos de análise diferem de acordo com as particularidades de cada modelo e das características a serem avaliadas (Holloway et. al, 1997).

Por outra parte e tratando especificamente o ODP se considera que a formalização dos conceitos de modelagem e de especificação no modelo ODP é um importante passo em direção a melhorar o controle da qualidade no projeto de sistemas distribuídos e abertos. O ODP estabelece as bases necessárias para a federação destes sistemas e é conveniente então proceder a sua análise para assegurar que esta federação atenda condições de segurança e confiabilidade especificadas.

Em geral, para analisar um sistema distribuído e aberto deve-se considerar que (Diagne, Estrailier, 1997):

- estes sistemas são compostos de módulos construídos independente, os quais também podem ser compostos por outros módulos. Assim, estes sistemas podem ser melhor abordados quando considerados de maneira modular;
- os sistemas distribuídos e abertos são construídos da união de várias partes independentes. Assim a verificação destes não pode ser realizada no sistema como um todo e se faz necessário dividir a análise nas diferentes partes e realizar esta de maneira composicional. As propriedades do sistema são inferidas das propriedades provadas nos seus módulos;

- a validação destes sistemas também precisam de um abordagem modular. Assim cada parte do sistema deve ser validada com base nos requerimentos que deve cumprir. No entanto deve-se ter cuidado quando se considera a integração das partes.

Neste contexto, não é adequado construir uma rede de Petri para o sistema inteiro. Assim, a abordagem proposta para a análise de sistemas distribuídos e abertos baseia-se em módulos, isto é, a rede de Petri de cada módulo é analisada separadamente. Desta maneira, as propriedades obtidas são usadas para obter informação de um módulo.

Propriedades a verificar

No processo de análise, a cada etapa de desenvolvimento pode se proceder de maneira formal à avaliação das propriedades que convém considerar a esse nível. Estas propriedades podem ser ditadas pelas características do sistema ou pela sinergia do sistema com seu ambiente. Algumas destas propriedades são:

- **Segurança:** determina sua capacidade de assegurar as funcionalidades que lhe são determinadas.
- **Confiabilidade:** determina sua capacidade de assegurar as funcionalidades entre seus módulos.
- **Desempenho:** determina sua capacidade de assegurar a seu ambiente humano e/ou técnico um grau de satisfação conveniente
- **Conformidade:** determina sua capacidade de resolver os problemas inicialmente estabelecidos e sua adequação com seu ambiente técnico e humano.
- **Integridade:** determina sua capacidade de gerenciar os recursos.

Como apresentado no capítulo 2, neste trabalho consideram-se para o caso dos sistemas prediais que as propriedades de segurança e confiabilidade são cruciais para o adequado funcionamento dos sistemas e a sua integração. A segurança entendida como a garantia que o sistema não chega a atingir estados que corrompam seu comportamento ou que lhe empeça de continuar suas interações com o seu ambiente.

A confiabilidade assegura que o sistema realiza as funcionalidades que seu ambiente espera dele.

Assim, são estas as propriedades que serão verificadas nos modelos desenvolvidos em RPMCO. Estas podem ser expressas nos módulos que compõem os sistemas sob a forma de propriedades locais e de hipóteses sobre seu ambiente.

Tratamento da análise

Deve-se destacar que tanto na modelagem como na análise de sistemas, abordagens como a abstração e refinamento equivalem à representação do sistema em vários níveis de detalhe. Estas abordagens são imprescindíveis quando se trata sistemas complexos tanto pelo seu porte como pelo grau de integração de suas partes. Assim para a análise dos modelos tem-se optado por um método de refinamento *bottom – up*: no qual se parte de estruturas conhecidas e funcionais, -módulos- previamente utilizadas ou provadas, e a partir destas se organiza um sistema maior e mais genérico com as funcionalidades desejadas.

Assim, as técnicas de análise aproveitam a modularidade ligada à introdução dos conceitos de orientação a objetos para evitar a explosão combinatória do espaço de estados devido ao porte dos modelos (em lugar de avaliar todas as possíveis instâncias combinadas das opções de controle, avalia se cada uma separadamente e combinam-se os resultados depois). Desta maneira o procedimento de análise é melhorado em contraposição à abordagem do sistema completo.

Os processos de verificação e validação são realizados de maneira incremental. Este aspecto incremental da validação/verificação permite proceder à avaliação de todos os modelos obtidos e evitar a influência de erros de uma fase nas outras. Pode-se considerar a análise de sistemas específicos com diferentes níveis de detalhamento nos módulos. Em cada caso é conveniente adotar uma abstração adequada do ambiente. Esta abstração garante a qualidade da validação e da verificação.

Verificação

Para realizar a verificação de modelos existem duas abordagens:

Model checking: é a prova de propriedades através da enumeração dos estados atingíveis por um modelo.

Theorem proving: nesta abordagem se infere ou contradiz uma propriedade através de regras lógicas da matemática (Clarke, Wing, 1996).

A abordagem de verificação que se utiliza no presente trabalho é o *model checking*. No presente abordagem escolheu-se realizar a verificação através da análise do espaço de estados através do grafo de alcançabilidade ou grafo de ocorrência. Esta análise beneficia-se da decomposição do sistema e da modularidade da rede e permite a construção do espaço de estados de redes menores com o qual evita-se o problema de explosão de estados. A idéia é gerar o espaço de estados para cada módulo (aqui as interações são vistas do ponto de vista interno e consideram as condições de invocação) e a informação necessária para capturar a interação entre os módulos.

A construção do grafo de alcançabilidade envolve essencialmente a enumeração de todas as marcações alcançáveis. Este grafo permite que através dele podem ser analisadas várias propriedades dinâmicas como a vivacidade, segurança, verificar *deadlock*. Este método é útil para a correção progressiva da especificação do sistema com diferentes níveis de precisão.

A exploração exaustiva dos estados alcançáveis se realiza a partir de um estado inicial determinado pelas configurações dos módulos RPMCO do sistema. A configuração de uma RPMCO é determinada pela marcação inicial. Assim se procura determinar as evoluções dos módulos que violem as restrições colocadas na especificação ou as que conduzam a estados de erro.

A construção progressiva de uma RPMCO considerando seus clientes e servidores produz a cada etapa uma rede de Petri que modela com mais ou menos precisão o funcionamento e as interações em um subsistema. A precisão dependerá do conhecimento exaustivo de todas as interações entre todos os módulos.

O grafo de alcançabilidade indica a existência de falhas de interação como o não respeito das restrições definidas ou as falhas de resposta. Permite verificar também se o servidor é robusto para suportar a concorrência das interações com seus clientes de semânticas eventualmente diferentes. Este gráfico também avalia os eventuais estados bloqueantes para o subsistema dado e seu estado inicial considerado.

Esta abordagem incremental é interessante quando se considera sistemas parcialmente especificados ou que integram módulos que não são conhecidos por seu comportamento.

Construção do Grafo de alcançabilidade

Segundo Murata (1994) se tem a seguinte definição:

Definição 3.13 Um grafo de alcançabilidade é uma 5-tupla (V, D, H, Q, v_0) onde:

(i) (V, D) é um grafo orientado finito tal que $\forall v \in V$, existe um caminho orientado de v_0 a v

(ii) Q é chamado o grupo dos rótulos e $H: D \rightarrow Q$ associa a cada arco do grafo orientado um rótulo.

O estado v_0 é chamado estado inicial do grafo. O grafo de alcançabilidade de uma rede de Petri é um grafo pelo qual os arcos são etiquetados pelas instâncias das TRANSIÇÕES da rede.

Definição 3.14

Sejam dois grafos de alcançabilidade

$G_{a1} = (V_1, D_1, H_1, Q_1, v_1)$ e $G_{a2} = (V_2, D_2, H_2, Q_2, v_2)$. Chama-se combinação de G_{a1} e G_{a2} seguinte $T_1 \cap T_2$ ao grafo $G_{a12} = (V_{12}, D_{12}, H_{12}, Q_{12}, v_{12})$ onde:

$$V_{12} \subseteq V_1 \times V_2$$

$$D_{12} = D_1 \cup D_2$$

$$v_{12} = (v_1, v_2)$$

V_{12}, D_{12} e H_{12} são construções recursivas como segue:

$$V_{12} = \{v_{12}\} = \{(v_1, v_2)\}, D_{12} = \emptyset$$

$$\forall v = (x, y) \in V_1 \times V_2, \text{ se } \exists v' = (x', y') \in V_{12} \exists q \in Q_{12}$$

$(H_1(x, x') = q \wedge y = y') \vee (x = x' \wedge H_2(y, y') = q) \vee (H_1(x, x') = q \wedge H_2(y, y') = q)$ então $V_{12} = V_{12} \cup \{(x, y)\}, D_{12} = D_{12} \cup \{(v, v')\}$ e $H_{12}(v, v') = q$

A combinação de dois grafos de alcançabilidade pode ser estendida a n grafos ($n \geq 2$).

A estrutura do módulo deve apoiar a expressão das propriedades do módulo e hipóteses estabelecidas para o ambiente. Estes são predicados que devem ser conservados qualquer que seja a evolução do sistema. A verificação consiste em provar estes predicados estabelecidos pelo modelador sobre os módulos e detectar os erros de interação (erros de resposta).

A expressão e a verificação de propriedades são facilitadas pelos meios utilizados para formalizar os aspetos apropriados do modelo do módulo. As propriedades e as hipóteses são expressadas ao nível modular afim de se beneficiar – para sua verificação da simplicidade relativa em relação ao sistema global

Os predicados permitem caracterizar o comportamento lícito para um módulo ou para seu ambiente. Eles são constituídos de propriedades limitadas de maneira implícita às interações entre módulos e às propriedades estabelecidas pelo modelador. As propriedades implícitas que caracterizam as interações corretas são:

- a ausência de erros de resposta nos pedidos, isto é, para cada pedido deve ser tratado e produzida uma resposta no caso de este ser interrogativo.
- todos os módulos devem verificar as obrigações estabelecidas pelos servidores sobre as funcionalidades que utilizam.

Estes predicados são proposições lógicas que devem ser verdade qualquer que seja a evolução do módulo sobre as quais sejam estabelecidas. Um predicado especifica uma correspondência entre os estados do módulo (ou de seu ambiente) e do que deve ser verdadeiro em estes estados. Esta correspondência pode ser verificada sobre o grafo de estados construídos para o módulo com uma abstração de seu ambiente.

Verificação das propriedades de segurança e confiabilidade

A rede de Petri permite realizar uma verificação formal das propriedades de segurança e confiabilidade. Esta verificação consiste na procura dos estados do sistema que possam alterar seu funcionamento normal (estados bloqueantes)

A análise dos estados bloqueantes é realizada sobre o grafo de ocorrência do módulo concernido.

A análise de um estado bloqueante pode ser realizado em dois níveis:

- localmente em um módulo: isto implica considerar em detalhe as ações que são efetuadas. Elas permitem ter em conta a causa ou efeito do bloqueio sobre seu módulo;
- sobre o conjunto dos módulos implicados: pode-se analisar a seqüência de interação anterior ao bloqueio durante a realização da abstração dos efeitos locais aos módulos. Desta maneira podem ser detectadas as evoluções do sistema susceptíveis de conduzir ao bloqueio e encontrar os mecanismos corretivos ou de prevenção

Validação

Uma primeira etapa da validação consiste na análise da completude e coerência global do modelo. Assim é analisada a coerência entre as diferentes partes do módulo (interface, estrutura, etc.), e a coerência entre os serviços oferecidos e requisitados assim como sua semântica de utilização.

A coerência global de um modelo de sistema é o objeto da fase de validação e se baseia na validação sintática do modelo obtido.

Sobre um modelo em RPMCO pode-se tratar um certo nível de coerência considerando se:

- todos os serviços requeridos pelo módulo cliente devem ser oferecidos por um servidor. O cliente deve assegurar o respeito às restrições associadas às operações do serviço;
- a interface do módulo deve estar conforme a sua implementação. As interações específicas devem ser suportadas pela estrutura do módulo.

A validação é realizada por uma simulação/animação com a finalidade de avaliar os modelos com relação aos requisitos expressados pelos usuários do sistema e detectar se o modelo construído alcança os estados não desejáveis do sistema. A simulação pode ser realizada em diferentes níveis de granularidade: módulo, sistema, ou sistema global. Esta simulação/animação (com entradas consistentes) é suportada na rede de Petri pelas ferramentas computacionais existentes que no caso do presente trabalho é o *CPN Tools*.

Simulações ajudam a conferir a lógica de controle dos sistemas prediais. Em automação predial modelos de simulação podem ser usados para prever o comportamento dinâmico atual do sistema por ter em conta os dados de entrada dos sensores e outras fontes. A simulação pode ajudar a gerar e avaliar múltiplas alternativas de controle baseado em um certo conjunto dado de critérios de preferências sem ser restringido pelas restrições que os edifícios e seus sistemas podem ter.

3.3. CONSIDERAÇÕES DO CAPÍTULO

Neste capítulo foi proposto um formalismo chamado Rede de Petri Modular Colorida baseada em Objetos – RPMCO para a modelagem e análise de sistemas distribuídos e abertos para automação predial.

A RPMCO considera as características e expressões dos pontos de vista da informação e da computação (cuja interdependência é manifesta) do ODP. Nesta abordagem ambos os pontos de vista são tratados de maneira integrada pois se considera que eles não podem ser avaliados independentemente um de outro.

A RPMCO permite a obtenção de um modelo modular. Cada módulo é descrito de maneira não ambígua. Esta descrição permite caracterizar um módulo dado em relação a seu ambiente e de maneira independente de sua implementação.

A proposta descreve os módulos (como um modelo genérico de estruturação de dados) e os mecanismos de composição para construir subsistemas e estabelece:

- uma composição estrutural que permite proceder de maneira incremental à modelagem e a análise de um sistema em uma abordagem *bottom - up*.
- uma composição funcional baseada sobre uma topologia aberta no sentido de que um módulo não conhece de seu ambiente as abstrações que decretam as regras de comportamento.

Considera-se que o formalismo proposto é consistente para a modelagem e a análise pois cada módulo tem uma parte encapsulada (unidade comportamental) e

uma parte fornecida para o ambiente (interface do módulo) e deve respeitar as hipóteses estabelecidas sobre o ambiente.

O modelo de um módulo estabelece uma semântica baseada nas redes de Petri coloridas, as quais se adaptam de maneira satisfatória às diferentes granularidades que se pode encontrar na especificação de sistemas distribuídos (módulos, sistema, sistema global). O formalismo é baseado em objetos sem oferecer o conceito de herança.

A RPMCO tem em conta os aspetos específicos dos módulos de sistemas prediais como a autonomia e a sua interação com os outros sistemas.

O formalismo possui um poder de abstração adequado para os sistemas prediais. Este permite descrever de maneira explícita tanto as interações entre os módulos como as operações que eles suportam o que é fundamental para a análise do sistema.

A análise da RPMCO tira proveito da modularidade desta rede. As propriedades que foram consideradas para verificação foram: a segurança do sistema através da garantia da ausência de estados bloqueantes e a confiabilidade provando que os estados que são de interesse são efetivamente alcançados.

A verificação é incremental no sentido que se podem considerar os módulos que podem ser tratados como um cliente ou um servidor. Esta visão permite validar o comportamento de um cliente frente a seus servidores e vice-versa. Assim, se pode saber se um cliente utiliza corretamente o servidor e se o servidor responde às exigências de seus clientes e suporta as interações.

4. PROCEDIMENTO PARA A MODELAGEM E ANÁLISE DE SISTEMAS DISTRIBUÍDOS E ABERTOS PARA AUTOMAÇÃO PREDIAL

Neste capítulo é apresentado o procedimento desenvolvido para a modelagem e análise de sistemas distribuídos e abertos para automação predial. O procedimento é baseado na aplicação do modelo de referência ODP através de seus pontos de vista, empresa, informação e computação e a utilização da UML e as redes de Petri.

Dentro da configuração de um edifício inteligente e do ponto de vista de sua operação e gestão, são os sistemas prediais que devem atender às necessidades dos usuários e as demandas de cada ambiente.

Atualmente, os produtos existentes para os sistemas prediais, têm como desvantagem não permitir a integração efetiva entre os diferentes sistemas, isto é, não é possível conseguir a interação entre estes de maneira a formar um sistema integrado de automação do edifício. Neste contexto, carece-se de um modelo do sistema capaz de prever o comportamento deste, não existindo nem mesmo um guia claro para estruturar um sistema predial.

Assim, para que o sistema de automação como um todo e especificamente para que cada um dos sistemas prediais sejam capazes de realizar funções integradas e complexas é necessária uma sistemática de projeto e implementação que assegure a sua

correta concepção e detalhamento, permitindo flexibilidade para adaptações nas políticas de segurança, conforto e conservação de energia do edifício.

Neste contexto, o procedimento a ser considerado deve auxiliar, orientar e fornecer o suporte necessário para o desenvolvimento da modelagem e análise de um sistema distribuído e aberto para automação predial (sejam de edifícios a serem construídos, onde já se tem definidas, as características dos sistemas prediais a instalar ou, para tornar inteligente um prédio convencional), considerando uma eficiente integração entre os sistemas prediais. Este procedimento deve portanto organizar e estruturar as etapas de trabalho de modo sistemático com a finalidade de outorgar o suporte adequado à flexibilidade que os edifícios inteligentes devem apresentar.

A utilização de diferentes técnicas na modelagem e análise de sistemas é uma prática que visa assegurar vantagem das características de cada uma das abordagens consideradas. Assim, define-se um compromisso para sua aplicação onde as possíveis formas consideradas são:

- **Integração:** consiste em associar modelos de cada um dos paradigmas e usar as transformações necessárias para converter um tipo de modelo em outro.
- **Co-utilização:** consiste em estender um modelo de um dos paradigmas com os aportes de outro. Os modelos assim gerados assumem um compromisso aceitável entre os dois paradigmas.

No presente trabalho, considera-se a técnica de co-utilização pois a abordagem de integração pode gerar problemas de coerência entre as diferentes abordagens consideradas para o desenvolvimento de modelos. Assim, apresenta-se a seguir o procedimento para a modelagem de sistemas distribuído e aberto para automação predial considerando a co-utilização.

4.1. PROCEDIMENTO DE MODELAGEM E ANÁLISE

O presente procedimento para a modelagem e análise de sistemas distribuídos e abertos para automação predial, baseia-se na aplicação do modelo de referência ODP através de seus pontos de vista: empresa, informação, computação e a utilização da UML e das redes de Petri, de forma a complementarem-se na modelagem e permitir a análise do sistema. O trabalho restringiu-se a estes pontos de vista por considerar que são os que melhor se adaptam à modelagem dinâmica e à análise.

O procedimento que está sendo proposto promove um desenvolvimento iterativo e no qual, em cada etapa se tem a possibilidade de considerar e revisar etapas precedentes de forma a assegurar sua rastreabilidade. No procedimento são definidas três etapas gerais que são refinadas em atividades. Cada etapa pode ter relação com um ou vários dos pontos de vista da ODP.

As etapas gerais do procedimento proposto são:

Etapa 1: Modelagem do domínio do sistema predial;

Etapa 2: Modelagem dos relacionamentos e serviços do sistema predial;

Etapa 3: Modelagem dinâmica e análise dos modelos.

A Figura 4.1 apresenta o procedimento proposto e o seu relacionamento com os pontos de vista do ODP.

O procedimento proposto está baseado na representação do sistema predial em vários níveis de detalhe (característica imprescindível quando se trata de integração) de acordo a uma abordagem modular, o que facilita sua compreensão e utilização. A especificação obtida a partir do procedimento procura ser rastreável para reduzir problemas de interpretação.

Considerando que a capacidade de estruturar um modelo é essencial na especificação de um sistema predial, na modelagem de um sistema de média ou elevada complexidade, se faz necessária a utilização de mecanismos de estruturação do modelo que permitam a consideração de diferentes níveis de abstração.

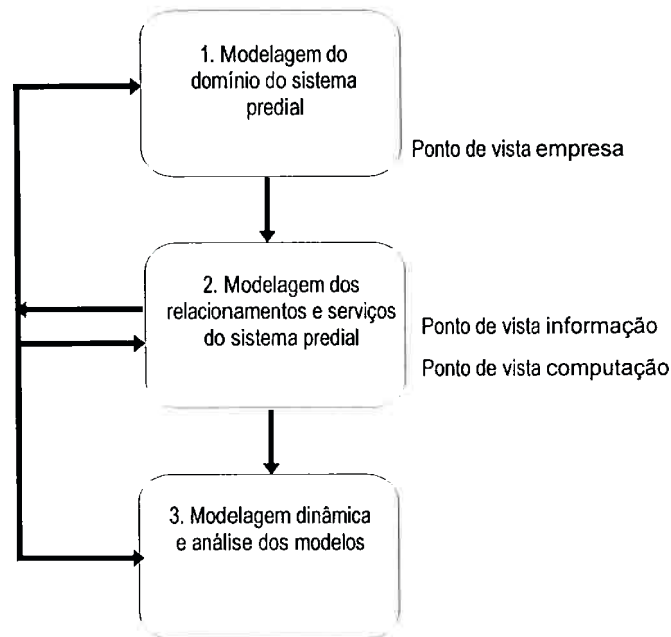


Fig. 4.1 Procedimento para a modelagem e análise de sistemas distribuídos e abertos para automação predial

O procedimento proposto está baseado no conceito de modularidade, de composição e agregação que são extensivamente utilizadas nas abordagens de modelagem e análise de sistemas em geral e comumente classificadas como abordagens descendente (*top-down*) ou de refinamento e ascendente (*bottom up*) ou de abstração. Na abordagem descendente cada um dos sistemas, obtidos em qualquer nível de representação, pode ser alvo de um refinamento posterior, aplicado iterativamente, até que os sub-modelos obtidos sejam suficientemente simples para proceder a sua análise ou implementação direta. Nas abordagens ascendentes, a utilização e combinação de blocos (isto é, modelos de partes ou de funções específicas) previamente disponíveis permite obter o comportamento desejado para o sistema.

O processo de abstração/ refinamento no procedimento proposto é definido a partir dos modelos gerados usando a UML e que permitem a estruturação do processo de modelagem associado com a utilização da rede de Petri com a finalidade de dispor de uma técnica para a modelagem dinâmica e a análise formal dos modelos gerados. A figura 4.2 ilustra este processo.

O procedimento procura minimizar o esforço para estruturar a descrição do sistema predial a partir de refinamentos e abstrações dos modelos realizados em diferentes níveis.

A utilização das técnicas de modelagem e análise no procedimento é realizada através da co-utilização que permite o uso de diferentes formalismos considerados em razão de sua relevância para as etapas e atividades desenvolvidas. Para isto avaliaram-se as etapas do ciclo de desenvolvimento e identificaram-se os pontos, onde a introdução de procedimentos e ferramentas são relevantes para alcançar os objetivos propostos.

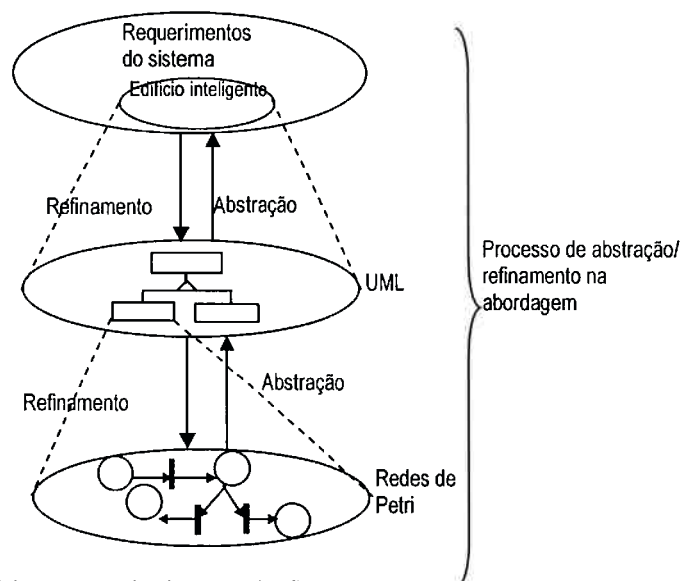


Fig. 4.2 Processo de abstração/ refinamento na abordagem proposta

4.1.1. ETAPA 1: MODELAGEM DO DOMÍNIO DO SISTEMA PREDIAL

Nesta etapa são obtidas, definidas e organizadas as informações e as características dos sistemas prediais que permitem desenvolver a sua especificação e integração. Aqui, fornecem-se recursos para a aquisição do conhecimento relativo ao domínio do problema.

Esta etapa está diretamente relacionada com o ponto de vista empresa e apresenta uma visão geral do sistema. Aqui é representada a funcionalidade deste a partir do nível mais alto de abstração e é onde definem-se os objetivos e propósitos do sistema, assim como as políticas (permissão e obrigações) aplicadas a este e suas interações com o ambiente externo.

O ambiente da empresa impõe requisitos e direcionamentos ao sistema a ser modelado, neste caso um sistema distribuído e aberto composto por vários sistemas prediais. Tais considerações são incluídas na sua especificação e na forma de objetos que representam o ambiente no qual o sistema opera. Assim, a abordagem proposta deve ser abrangente o suficiente de forma a incluir os diferentes serviços que podem ser implementados nos edifícios de acordo a sua função. Dependendo da finalidade do edifício, outras características podem ser adicionadas.

Nesta etapa procura-se ainda uma representação em alto nível do sistema de automação, apresentando somente as características relevantes à definição do sistema. Desta maneira facilita-se o mapeamento no modelo dos diversos aspectos que devem ser manipulados na especificação de um sistema distribuído e aberto como gerenciamento de recursos, comunicação de dados e processamento de informações (Garrahan, 1993).

Em geral, nesta etapa define-se o que o sistema predial deve realizar e sob quais restrições deve funcionar. O objetivo desta etapa é obter uma especificação dos requisitos, objetivos e propósitos que a integração dos diferentes sistemas prediais deve apresentar para o edifício considerado. Este conjunto de especificações é elaborado em um nível de abstração conceitual e de maneira consistente (sem contradições) e, sem ambigüidades. Isto se faz possível através de um processo de desenvolvimento iterativo e incremental das especificações o que permite postergar algumas situações da modelagem.

Esta etapa deve apresentar de maneira adequada a informação necessária para as etapas seguintes

Nesta etapa são consideradas as seguintes atividades (figura 4.3):

A. Identificação dos requisitos.

B. Definição dos domínios do sistema e modelagem dos casos de uso.

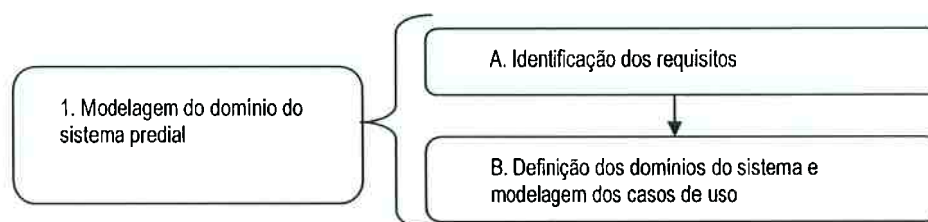


Fig. 0.3 Detalhamento da etapa "Modelagem do domínio do sistema predial"

A. IDENTIFICAÇÃO DOS REQUISITOS

Com a finalidade de assegurar que o sistema predial dispõe de fato dos serviços previstos e de soluções efetivas de controle é necessário caracterizar os prováveis ocupantes do edifício e seus processos de trabalho. Isto envolve identificar as especificações funcionais para as interfaces de controle dos diferentes sistemas prediais.

Nesta atividade realiza-se o levantamento dos requisitos e funções necessárias do sistema. Aqui são respondidas perguntas como: que tipo de serviços são necessários para a finalidade do edifício? Quais devem ser os sistemas prediais envolvidos e quais suas funcionalidades para realizar estes serviços? O que o sistema deve realizar e sob quais restrições? Estas informações servem como meio para realizar uma análise preliminar e identificar os dados relevantes e pertinentes para a modelagem.

Por diversos motivos, é muito difícil definir de maneira consistente e completa o conjunto de requisitos para um problema dado. Alguns destes motivos são:

- Para sistemas de automação de grande porte é difícil prever as repercussões de implantar um novo sistema sobre o existente;
- A comunidade de usuários de um edifício geralmente é muito diversa e com diferenças tanto formativas quanto de expectativas a respeito do sistema;
- Não se tem disponível ferramentas e métodos eficientes para identificar, especificar e validar requisitos;
- São necessárias notações próximas tanto ao pessoal técnico como ao pessoal menos especializado;
- É quase impossível capturar a totalidade das necessidades dos usuários, entre outras razões porque nem sempre os usuários sabem definir claramente o que desejam do sistema predial;
- É habitual que os requisitos evoluam mesmo antes de se concluir o desenvolvimento. Isto implica na necessidade de documentar para uso futuro as suposições e decisões que se tomaram na identificação de requisitos além das distintas alternativas que se exploraram.

Os requisitos em geral devem expressar o que o sistema predial deve realizar e não o como deve fazer, isto é, deve-se adquirir o conhecimento e entender as

necessidades dos usuários do edifício. É importante ressaltar que os requisitos abordados nesta abordagem são essencialmente funcionais.

Em geral, os requisitos de automação de processos industriais são diferentes dos de automação predial. O primeiro está associado com alta precisão, operação em torno de um ponto de equilíbrio e alta velocidade de resposta. O segundo, por sua parte, exige respostas para manter o edifício em determinados estados ao longo de condições operacionais (níveis de iluminação, condições de segurança, faixas de conforto térmico, entre outras) e para reagir a distúrbios. Assim, é fundamental identificar e estabelecer adequadamente os requisitos para os sistemas prediais.

Nesta etapa definem-se os requisitos de acordo com:

- Finalidade do edifício: hospital, shopping, escritórios, educativo, etc.;
- Tipos de usuários;
- Configuração do edifício;
- Sistemas prediais a serem considerados: neste item consideram-se requisitos como: funções associadas, capacidade de atuação ou processamento, controle, comunicação, confiabilidade e disponibilidade, interface homem-máquina, gerenciamento de dados entre outros.

Além disso, nesta etapa, as informações levantadas são analisadas e é avaliada a validade e viabilidade técnica do atendimento das necessidades do sistema predial. Aqui, são consideradas as normas oficialmente estabelecidas e/ou adotadas pela comunidade relacionadas com os sistemas prediais dependendo do tipo e função do prédio.

Uma avaliação cuidadosa e uma análise das informações levantadas nesta etapa pode identificar erros de interpretação e verificar a necessidade de novos levantamentos em função de:

- Informações não suficientemente claras;
- Informações contraditórias ou incorretas.

Estes estudos são fundamentais para avaliar as críticas e reconsiderações do sistema predial e dos objetivos ao se identificar:

- Objetivos não realizáveis com a tecnologia disponível;
- Objetivos não alcançáveis por razões de custo – benefício.

Assim, são definidas entre outras características do sistema predial: o tipo de controle, e as estratégias para seu funcionamento. As informações obtidas nesta etapa são a base das etapas subseqüentes, o que denota sua importância.

Uma vez que os requisitos dos sistemas prediais tenham sido definidos, a concepção de modelos como forma de descrição das funcionalidades do sistema permite a organização do conhecimento, facilita a comunicação entre diferentes pessoas, detecta erros e permite determinar a validade dos requerimentos levantados, tornando mais clara e objetiva a especificação do comportamento do sistema.

A análise nesta etapa tem como finalidade fazer a validação prévia dos requerimentos do sistema junto com as pessoas envolvidas no projeto e permite, se for o caso, alterar os parâmetros inicialmente adotados para o sistema.

B. DEFINIÇÃO DOS DOMÍNIOS DO SISTEMA E MODELAGEM DOS CASOS DE USO

Nesta atividade definem-se os domínios e as comunidades de objetos que se inter-relacionam com os sistemas prediais. Aqui, trata-se também cada sistema predial como um domínio específico e estudam-se as interações que são necessárias entre estes para alcançar seus objetivos

Ainda nesta atividade definem-se e modelam-se os casos de uso os quais, constituem uma descrição da funcionalidade do sistema do ponto de vista dos usuários. A modelagem dos casos de uso é realizada através dos diagramas de casos de uso onde devem ser representados os casos de uso propriamente ditos e os usuários e sistemas externos que interagem com estes e que são conhecidos como atores e entidades. Os diagramas de casos de uso descrevem o modo no qual os usuários podem usar o sistema. Assim, um sistema pode ser descrito pelos diferentes casos de uso que este oferece.

4.1.2. ETAPA 2: MODELAGEM DOS RELACIONAMENTOS E SERVIÇOS DO SISTEMA DE AUTOMAÇÃO PREDIAL

Esta etapa está relacionada com o ponto de vista computação o qual está focado na decomposição funcional do sistema.

Aqui são modelados os aspectos mais relevantes do sistema de acordo com os requisitos definidos na etapa anterior.

Esta etapa está dividida nas seguintes atividades (figura 4.4):

- A. Abstração dos relacionamentos e interações através de rede de Petri;
- B. Definição e construção do diagrama de classes



Fig. 0.4 Detalhamento da etapa “Modelagem dos relacionamentos e serviços do sistema predial”

A. ABSTRAÇÃO DOS RELACIONAMENTOS E INTERAÇÕES ATRAVÉS DE REDES DE PETRI

Para realizar a integração efetiva de vários sistemas, faz-se necessária a modelagem e a avaliação de cada um deles tendo em conta suas funcionalidades, os relacionamentos e interações à luz dos requisitos do sistema global de automação e o grau de integração.

Assim, nesta atividade constrói-se a rede de Petri Condição Evento que descreve a estrutura e a dinâmica dos processos de cada caso de uso. Esta rede modela as seqüências de atividades realizadas pelo sistema em cada caso de uso. Este modelo, neste tipo de rede de Petri, permite obter uma visão do fluxo de atividades do sistema.

B. CONSTRUÇÃO DO DIAGRAMA DE CLASSES

Realiza-se a decomposição funcional dos sistemas em objetos de forma a permitir que as aplicações e os componentes envolvidos sejam estruturados de maneira transparente. Embora sejam identificados os candidatos para a distribuição física, não são tratados os mecanismos de comunicação entre estes elementos.

Considerando que o objetivo que se procura através da integração é a manutenção do controle efetivo de todo o edifício inteligente, o tratamento de cada sistema deve considerar seus componentes (objetos) no contexto da automação de forma bipartida, dado que estes podem ser representados mediante a inter-conexão de duas partes fundamentais que se comunicam e operam cooperativamente. Estas representam a parte operativa relacionada ao sistema controlado¹ ou objeto de controle (por exemplo cabinas dos elevadores no caso do sistema de elevadores, etc.) e a parte de controle relacionada ao sistema de controle (vide figura 4.5)

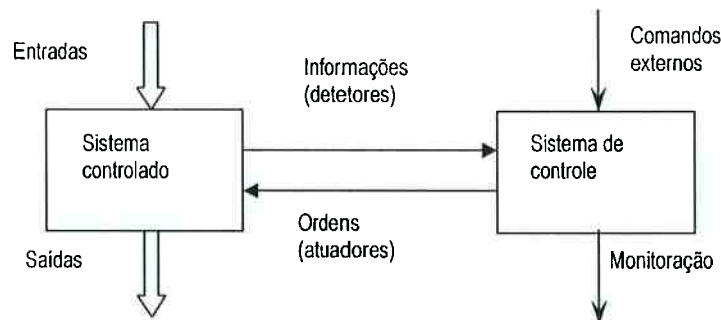


Fig. 0.5 Decomposição da descrição de um sistema em parte operativa e parte de controle

Assim, dentro do sistema controlado os principais candidatos a objetos são os recursos e equipamentos do sistema que enviam para o sistema de controle informações sobre seu estado possibilitando a realização do controle.

Dentro do sistema de controle consideram-se objetos as entidades que interagem com o sistema controlado e que enviam a este ordens de operação por intermédio dos atuadores.

¹ Utiliza-se o termo “sistema controlado” em lugar de “objeto de controle” para evitar confusões quando se trata de objetos relacionados com o paradigma de orientação a objetos

Assim, considerando o sistema controlado e o sistema de controle, o conjunto de objetos que participam dos casos de uso e que realizam cada ação neste são identificados a partir do modelo em rede de Petri condição/ evento e são então organizados em classes. O procedimento para obter as classes é o seguinte:

- Identificam-se os relacionamentos entre objetos e as classes;
- Adicionam-se os atributos de cada classe;
- Identificam-se as operações de cada classe;
- Identificam-se as relações de generalização/especificação (observando similaridades e diferenças) e de decomposição/composição;
- Construi-se o diagrama de classes.

4.1.3. ETAPA 3: MODELAGEM DINÂMICA E ANÁLISE DOS MODELOS

A complexidade e a utilização de um número elevado de informações de entradas e saídas dos sistemas automatizados traz consigo a dificuldade de realizar sua especificação funcional de maneira clara, concisa e não ambígua.

Em geral, esta etapa está relacionada com o ponto de vista informação o qual está focado na informação e nos processos associados ao sistema.

A modelagem nesta etapa permite a localização e distribuição adequada das complexidades envolvidas nos requisitos do sistema, tais como: funções, objetos, capacidade de processamento, controle, comunicação, confiabilidade e disponibilidade, interface homem-máquina, gerenciamento de dados, entre outras.

Nesta etapa, é onde se especifica e modela detalhadamente cada uma das funcionalidades dos sistemas considerados para o edifício, assim como a forma como eles inter-atuam.

Os modelos gerados nesta etapa são realizados através da RPMCO e estão, baseados nos modelos obtidos nas etapas anteriores (especificamente nos modelos em rede de Petri condição/evento e os digramas de classes). A segunda atividade depois dos

modelos serem obtidos é realizar sua análise com a finalidade de validar e verificar se representam adequada e corretamente os requisitos e funcionalidades do sistema.

As atividades dentro desta etapa são (figura 4.6):

A. Modelagem do sistema em RPMCO;

B. Análise dos modelos em RPMCO;

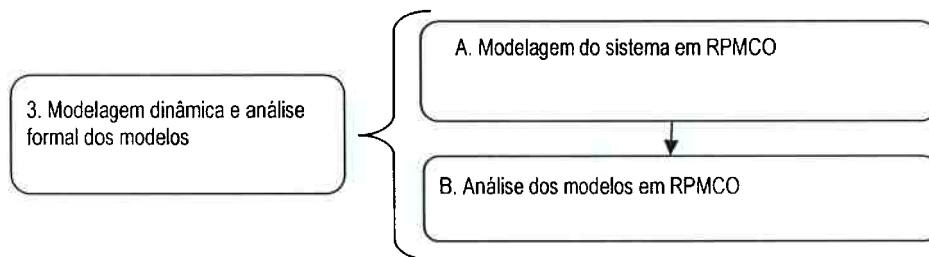


Fig. 0.6 Detalhamento da etapa “Modelagem dinâmica do sistema e análise formal dos modelos”

A. MODELAGEM DO SISTEMA EM RPMCO

Nesta atividade, são modeladas as informações que são trocadas entre os sistemas prediais do edifício além do comportamento dinâmico destes. Devido à natureza dinâmica das atividades e serviços em edifícios, estes suportam uma grande variedade de funcionalidades e informações. Cada fluxo de informação no sistema contém um ou mais dados que se modificam juntamente com as ações de controle do sistema, através de transformações (processos/funções). O processo de transformação das informações de entrada em informações de saída pode requerer outras informações, já armazenadas no sistema, ou pode envolver usuários ou outros sistemas.

A especificação da informação é gerada a partir dos componentes identificados na modelagem do ponto de vista empresa. As informações que o sistema manipula são modeladas considerando as variações esperadas no comportamento do sistema.

Para modelar a troca de informações entre os sistemas, o que equivale a modelar o comportamento dinâmico dos casos de uso, podem ser usados os diagramas de interação da UML (diagramas de colaboração e de seqüência). No entanto, estes diagramas apresentam a desvantagem de serem baseadas em uma técnica semi-formal e não possuem uma adequada representação para modelar a concorrência de processos.

Assim, na presente abordagem é proposto o uso das redes de Petri em substituição aos diagramas de interação.

Neste contexto e na abordagem proposta, a modelagem do comportamento dinâmico é realizada através da RPMCO. Aqui as classes e objetos identificados nas etapas anteriores (especificamente nos modelos dos digramas de classes e nas redes de Petri condição/evento), são descritos e detalhados através de módulos RPMCO. Assim, para cada classe identificada na modelagem estática é criado um módulo RPMCO, no qual são modeladas as operações dos objetos correspondentes. A obtenção do modelo dinâmico envolve os seguintes passos:

1. Identificação das funcionalidades de cada módulo: corresponde às funcionalidades das suas operações;
2. Definição da interface de cada módulo: indica os atributos, as operações e seus argumentos de entrada e saída, restrições;
3. Definição da unidade comportamental de cada módulo: modelagem das operações (métodos) do módulo.

A obtenção do modelo dinâmico em redes de Petri é realizado tendo em conta os seguintes aspetos:

- As associações no diagrama de classes geralmente representam mensagens; A representação explícita das mensagens ajuda a determinar os seus atributos;
- Nas classes que tenham dados relacionados, aplicar relações de agregação de maneira a obter classes nas quais a classe mãe tenha só operações e as classes filhas só atributos;
- Os métodos de cada classe são convertidos em TRANSIÇÕES;
- As associações de classes são convertidas em LUGARES com o correspondente cor associada com seus atributos;

Declarações

- As cores das MARCAS são definidas dos atributos das classes;
- Variáveis são definidas para cada cor;
- Definir as restrições.

Regras

- As regras são especificadas nos ARCOS de entrada e de saída das transições usando as variáveis definidas nas declarações.

Marcação Inicial

- A MARCAÇÃO inicial é derivada da redes de Petri condição/evento.
- Estes indicam os valores para os atores externos e os valores necessários para qualquer dado disponível nas classes que não correspondem a associações e que são modelados como TRANSIÇÕES.

B. ANÁLISE DOS MODELOS EM REDES DE PETRI

Uma vez obtidos os modelos em RPMCO, nesta etapa realiza-se a análise do comportamento dos mesmos. Como apresentado no capítulo 3 a análise de um sistema envolve verificá-lo e validá-lo.

Validação

A simulação mostra para certas condições e durante um intervalo de tempo limitado, uma das possíveis seqüências de estados de um sistema predial.

A simulação é útil para entender e depurar o modelo de um sistema em RPMCO. No entanto, através da simulação, se a condição inicial não é completamente conhecida, se o comportamento do sistema depende de interações com o ambiente exterior, ou ainda se a análise não se restringe a um intervalo de tempo limitado não é possível obter uma prova completa das propriedades dinâmicas de uma RPMCO e por tanto nem sempre é efetiva para garantir o comportamento desejado do sistema predial.

Dentro do procedimento proposto a validação dos modelos é realizada por simulação e através do uso de um *software* para redes de Petri coloridas. Os passos a serem seguidos nesta atividade são:

- avaliar a sintaxe dos modelos: aqui é conferida a sintaxe das expressões e a consistência dos modelos através do compilador da linguagem usada para as inscrições da rede de Petri.

- simular o comportamento dos modelos: aqui avalia-se se o modelo representa efetivamente os requisitos inicialmente estabelecidos para o sistema e demonstra a funcionalidade requerida. A simulação envolve compilar a rede de Petri considerando sua estrutura e suas inscrições. Aqui é analisada a coerência entre as diferentes partes do módulo (interface, estrutura), entre os serviços oferecidos e requisitados e a sua semântica de utilização.

Verificação

O objetivo nesta parte da análise é verificar as propriedades comportamentais do modelo construído em RPMCO. Estas propriedades não estão necessariamente restritas a um intervalo de tempo específico ou a uma determinada interação com o ambiente externo. Para realizar a verificação, as propriedades do modelo são comparadas com os requisitos referentes ao comportamento do sistema predial. Isto garante que se a propriedade for verificada é cumprida sob as mais diferentes circunstâncias nas quais o sistema predial possa vir a operar.

A técnica de análise usada -como apresentado no capítulo 3- é o grafo de alcançabilidade ou de ocorrência. Este representa o espaço de estados do modelo.

O grafo de ocorrência é constituído de nós conectados por arcos onde cada nó representa uma MARCAÇÃO (ou estado) da rede de Petri e cada arco representa a ocorrência de uma TRANSIÇÃO com um vínculo particular às variáveis que conduzem à próxima MARCAÇÃO da rede de Petri. Para cada uma das muitas combinações legais de vínculos para as variáveis nos arcos de entrada de uma TRANSIÇÃO, um nó que representa a MARCAÇÃO resultante do sistema é criado quando a transição é disparada com o vínculo específico. Este processo é repetido para todas as transições habilitadas e com cada possível vínculo de variáveis nos ARCOS de entrada das transições.

Para realizar a verificação das propriedades é proposto uma sistemática que explora a modularidade da RPMCO dada pela introdução dos conceitos de orientação a objetos de maneira a decompor a tarefa de análise que envolveria o modelo do sistema predial como um todo em análises mais simples que consideram os módulos que compõem o sistema e suas interações.

Assim, a verificação é realizada através da busca de cenários representando as possíveis evoluções do sistema predial, onde apenas os eventos e estados relacionados à propriedade são considerados.

Com o grafo de ocorrência é provada a alcançabilidade entre marcações o que corresponde à prova de veracidade (ou falsidade) de um cenário definido.

Procedimento para a verificação de RPMCO

Uma das principais características do procedimento é a decomposição do processo de verificação.

Este procedimento pode ser organizado na seguinte seqüência de passos:

- Definição das propriedades e das restrições: a verificação se inicia com a definição das propriedades que devem ser consideradas. Neste trabalho e como foi estabelecido no capítulo 3 as propriedades a serem verificadas são: a propriedade de segurança, que garante que estados considerados impróprios ou proibidos nunca sejam atingidos, e a confiabilidade dada na rede de Petri através da propriedade de alcançabilidade que assegura que em determinadas circunstâncias, um estado sempre é atingido garantindo assim as funcionalidades do sistema. Junto com as propriedades deve-se definir as restrições que são imposições sobre o estado ou a ocorrência de eventos nos objetos. Estas limitam os casos onde se espera que as propriedades sejam verdadeiras.
- Análise das possíveis interações entre os módulos: aqui se analisam as interações segundo o explicitado no modelo em RPMCO. Estas interações podem gerar obrigações de análise para outros módulos, isto é, a veracidade da propriedade está condicionada pelo comportamento de outros módulos com os quais este interage
- Análise do módulo: esta análise começa com a busca de cenários onde determina-se os cenários relativos a evolução daquele módulo nas condições definidas pela propriedade. Em seguida constroi-se o grafo de ocorrência de cada módulo. O processo de verificação continua até que todas as obrigações de análise tenham sido consideradas.

- Verificar se todas as propriedades consideradas são satisfeitas baseada na construção do grafo de ocorrência.

4.2. CONSIDERAÇÕES DO CAPÍTULO

Neste capítulo foi apresentado o procedimento proposto para a modelagem e análise de sistemas distribuídos e abertos para automação predial. O procedimento considera a modelagem dos pontos de vista empresa, informação, computação do ODP através do uso da UML e dos conceitos de rede de Petri de forma a complementarem-se por um lado na estruturação da modelagem e por outro na análise formal dos modelos gerados. Esta é uma abordagem iterativa que é iniciada com a modelagem do domínio do sistema predial. Em seguida, são modelados os relacionamentos e serviços do sistema, para depois desenvolver a modelagem do comportamento dinâmico e a análise formal dos modelos.

Através da UML explora-se a representação de diversos aspectos do sistema que permitam além de garantir que os requisitos do sistema predial sejam atendidos, uma estruturação dos modelos e sua modularização. Por sua parte, a rede de Petri tem o papel de realizar a ligação entre os modelos e de representar a dinâmica do sistema permitindo a análise formal dos mesmos.

O procedimento apresentado beneficia-se tanto da notação gráfica como do uso do padrão ODP, pois o uso de diferentes pontos de vista permite um melhor controle do processo de modelagem. Estas duas visões do sistema predial conduzem a uma estruturação do sistema global.

5. EXEMPLO DE APLICAÇÃO

Dentre os exemplos de aplicação que poderiam ser consideradas para ilustrar a aplicação da abordagem proposta, foi escolhido o prédio dos ambulatórios (PAMB) que faz parte do Complexo Hospitalar das Clínicas da Faculdade de Medicina da Universidade de São Paulo (HC-FMUSP). Este prédio não é de fato um edifício inteligente entretanto, em virtude de suas características é considerado aqui como um objeto de estudo relevante.

Os processos na área da saúde têm influencia direta da infra-estrutura, condições e recursos dos edifícios nos quais eles são desenvolvidos assim, a qualidade do espaço e os serviços disponibilizados nos hospitais podem afetar o resultado dos cuidados médicos (Visconti, 1999).

Além disso, atualmente, a tendência dos projetos de edifícios hospitalares consiste em assegurar uma aparência humanizada, procurando um caminho entre a modernidade, a funcionalidade e a hospitalidade (Visconti, 1999)

Por outro lado, os rápidos avanços das ciências médicas, dos métodos de tratamento e ainda as alterações na população atendida, a ampliação da demanda de leitos e de tratamentos, as novas tendências em educação, pesquisa, administração e gerenciamento hospitalar, fazem com que os edifícios hospitalares sofram contínuas adaptações e ampliações.

Assim, um aspecto importante a considerar quando se trata de ambientes hospitalares é a gestão da qualidade em saúde, a qual consiste de três perspectivas a saber (Pinto, 1996):

estruturação: engloba a planta física do hospital, equipamentos, recursos humanos, financeiros e tecnológicos;

processos: inclui todas as operações e processos de trabalho;

avaliação final: envolve o atendimento ao cliente. Todo hospital deve ter por objetivo atender as necessidades e superar as expectativas dos pacientes.

Neste contexto, este exemplo de aplicação procura tratar a perspectiva da estruturação especificamente no referente aos recursos tecnológicos –sistemas prediais- e seus serviços que devem ser integrados para atender as necessidades dos usuários (pacientes, médico, funcionários, visitantes, etc.) do hospital.

5.1. COMPLEXO HOSPITALAR DAS CLÍNICAS

O Complexo Hospitalar das Clínicas da Faculdade de Medicina da Universidade de São Paulo (HC - FMUSP) é um hospital de atendimento terciário¹. Este complexo é formado por 2 hospitais auxiliares, 1 divisão de medicina de reabilitação e pelos seguintes institutos e prédios:

- Instituto central (IHC) : cujo prédio conta com 44.760 m² de área construída. Além deste prédio, fazem parte deste instituto:
 - Prédio dos Ambulatórios (PAMB): com cerca de 111.000 m² de área construída.
 - Instituto de Radiologia (InRad): com um prédio de cerca de 4.000 m² de área construída.

•

¹ O nível de atendimento terciário caracteriza-se pela maior capacidade resolutiva dos casos mais complexos do sistema de saúde, nas modalidades de atenção ambulatorial, internação e de urgência. Os estabelecimentos neste nível são, basicamente, os ambulatórios de especialidades, hospitais especializados e os hospitais de especialidades (Pinto, 1996)

- Instituto do Coração (InCor): com cerca de 31.800 m² de área construída.
- Instituto de Ortopedia (IOT): com cerca de 22.668 m² de área construída.
- Instituto de Psiquiatria (IPq): com cerca de 13.067 m² de área construída.
- Instituto da Criança (ICr): com cerca de 11.000 m² de área construída.
- Prédio da Residência Médica: com cerca de 5.100 m² de área construída. Este prédio se dispõe de acomodações para os residentes médicos.
- Prédio da CPD: Central de processamento de dados
- Prédio da Administração: com cerca de 12.000 m² de área construída.
- Garagem da divisão de transporte: com cerca de 2.700 m² de área construída.
- Depósito de lixo.
- Casa de caldeiras.
- Oficina de conservação e reparo: com cerca de 2.882 m² de área construída.

O Complexo Hospitalar das Clínicas está localizado no triângulo formado pela Av. Rebouças e Av. Dr. Enéas Carvalho de Aguiar e a divisa com os fundos dos terrenos da Rua Oscar Freire e do outro lado da Av. Dr. Enéas de Carvalho de Aguiar, em uma faixa de terreno atrás da Faculdade de Medicina da USP e do Instituto de Infectologia Emilio Ribas, na cidade de São Paulo, SP. A disposição de alguns dos prédios do Complexo Hospitalar das Clínicas podem ser observados na Figura 5.1

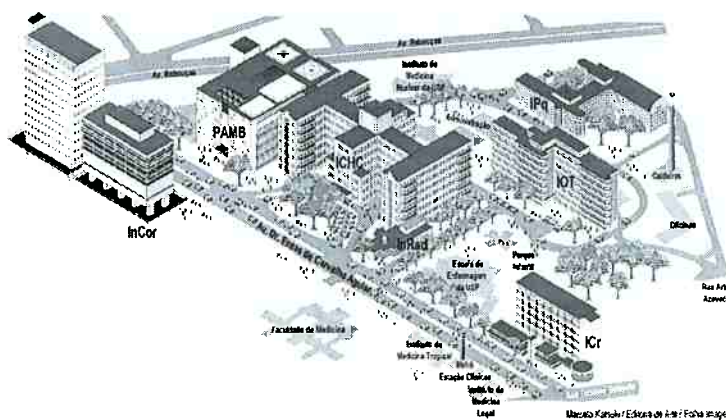


Fig. 5.1 Localização do PAMB no Complexo Hospitalar das Clínicas

A parte de processamento de dados dos pacientes é realizada pela PRODESP (Companhia de Processamento de Dados do Estado de São Paulo) através de um servidor que se comunica com um computador UNIVAC que se localiza no prédio da CPD. O Complexo Hospitalar das Clínicas possui um sistema com servidores em todos os prédios, ligados através de uma intranet (envolvendo cerca de 300 terminais). Cada edifício possui ainda uma intranet interna com processamento de dados compartilhados e seu banco de dados.

5.2. PRÉDIO DOS AMBULATÓRIOS - PAMB

O Prédio dos Ambulatórios foi projetado pela empresa Hidroservice em 1972 e inaugurado em 1979. Está localizado ao lado do ICHC e com frente para a Avenida Dr. Enéas Carvalho de Aguiar.

O PAMB é um ambulatório de especialidades destinado a prestar assistência médica especializada, em regime ambulatorial, a uma população determinada, tendo como característica o atendimento médico permanente de especialidades básicas (clínicas médicas, gineco-obstétrica, pediátrica e cirúrgica) e estratégicas. A atuação deste tipo de ambulatório impede o encaminhamento desnecessário de casos aos hospitais de maior complexidade como os hospitais especializados e os hospitais de Base.

Pelo PAMB circulam atualmente, aproximadamente 2300 pacientes ao dia e são realizadas diariamente cerca de 5500 consultas por dia.

Para este edifício foram ainda transferidos os seguintes serviços que funcionavam no ICHC: centro cirúrgico, banco de sangue, farmácia e laboratórios centralizados. Este prédio comporta toda a área de agendamento de consultas, administração, salas de aula, anfiteatros, centro endoscópico, 295 salas para atendimento médico, 139 consultórios, 86 consultórios especiais e 70 salas de exames especializados.

5.2.1. CARACTERÍSTICAS DOS USUÁRIOS

Tipos de usuários: correspondem basicamente aos seguintes tipos:

- funcionários: pessoal de saúde (médicos, enfermeiras, auxiliares) e pessoal administrativo e operativo assistencial;
- público geral: pacientes, visitantes, estudantes de medicina

Os pacientes que chegam ao PAMB são chamados de pacientes ambulatoriais e são pacientes não internados que procuram o hospital para consulta e exames; também pertencem a este grupo aqueles pacientes que, tendo recebido alta de sua internação, necessitam acompanhamento para prosseguir o seu tratamento.

5.2.2. CARACTERÍSTICAS FÍSICAS

O PAMB tem 111.000 m² de área construída e é composto por 11 pavimentos funcionais (discriminados de acordo com a Tabela 5.1) e mais 7 pavimentos técnicos (inter-andares ou pavimentos específicos para a instalação, manutenção e reparação de equipamentos (infra-estrutura elétrica e hidráulica) e outras facilidades previstas para a flexibilidade funcional do edifício. Através destes pavimentos técnicos e das prumadas (aberturas verticais) localizadas nas extremidades do edifício, passam todos os dutos: de esgoto, ar condicionado, água quente, água fria, vapor, gases medicinais, rede elétrica, rede de telefone, etc.).

Tabela 5.1 Pavimentos funcionais do PAMB

Nome do Pavimento	Função do Pavimento
Cobertura (andar 10)	Heliporto e área para primeiros socorros
Nono pavimento (andar 9)	Centro cirúrgico com 33 salas cirúrgicas
Oitavo pavimento (andar 8)	Farmácia industrial
Sexto pavimento (andar 7)	Andar de ambulatórios.
Quinto pavimento (andar 6)	Andar de ambulatórios.
Quarto pavimento (andar 5)	Serviços administrativos, arquivos, agendamento de consulta, farmácia, área convênio.
Segundo pavimento (andar 4)	Laboratório central (certificado ISO 9002)
Primeiro pavimento (andar 3)	Banco de sangue (Hemocentro)
Pavimento térreo (andar 2)	Acesso de funcionários e unidades de serviço de apoio aos funcionários e visitantes
Subsolo (andar 1)	Serviços gerais (cabina primária de alimentação de energia elétrica)
Casa de máquinas	Caldeiras, Geradores

Como pode ser apreciado segundo a nomeação dos andares, no PAMB não existem pavimentos identificados como terceiro e sétimo devido à correspondência com os identificadores existentes nos pavimentos do prédio do ICHC. Para efeito deste trabalho considerou-se adequado nomear os andares de 1 ao 10 a partir do subsolo (sem considerar os pavimentos técnicos).

Ressalta-se também que o prédio possui dois acessos externos para seus usuários, localizados em diferentes partes do prédio devido à irregularidade do terreno onde este se encontra, uma delas é destinada para os funcionários do hospital e está localizada no pavimento térreo (andar 2), a outra entrada é usada pelos pacientes e visitantes e se encontra no nível do quarto pavimento (andar 5).

A estrutura de distribuição física dos ambientes no PAMB é genérica, isto é, basicamente tem-se a mesma planta estrutural para todos os seus andares, sendo que cada ambulatório adaptou-a de acordo a suas necessidades.

O prédio dos ambulatórios está dividido em quatro blocos de circulação (A, B, C, D) (vide Figura 5.2)

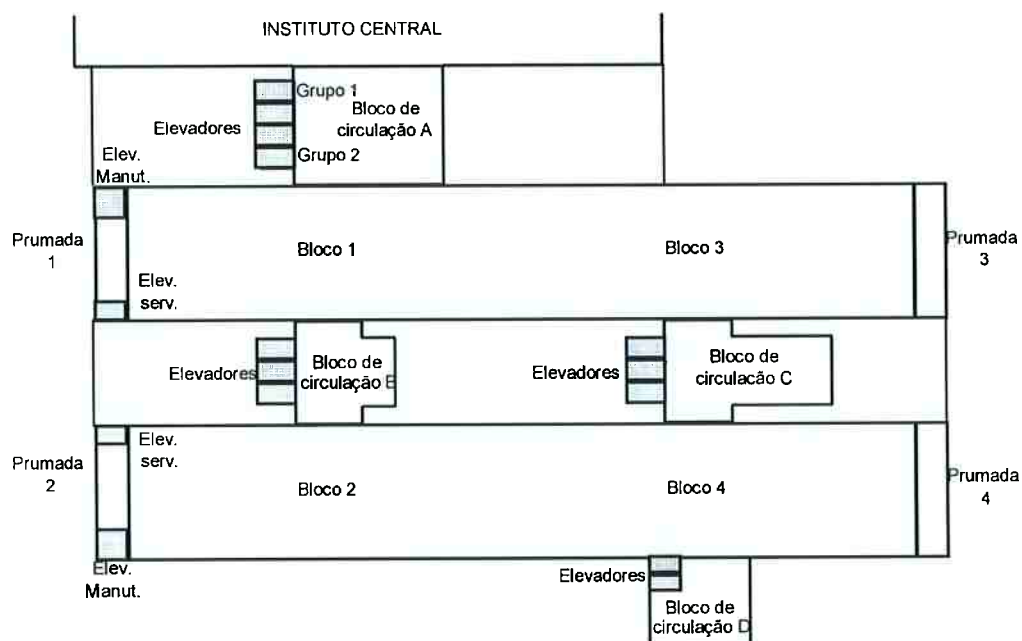


Fig. 5.2 Planta ilustrativa do prédio dos ambulatórios (PAMB).

O PAMB tem o “core” (área de serviços, áreas molhadas como sanitários, e copas, *hall* de acesso ao pavimento e escadas de incêndio) deslocado para as extremidades, de forma a privilegiar a iluminação e a ventilação natural.

Atualmente não existe nenhum tipo de Sistema de Gerenciamento do Edifício (SGE) instalado no PAMB

O PAMB é um prédio que embora tenha sido concebido prevendo um certo nível de automação predial adequada à função que realiza, atualmente isto não se verifica e, a operação de praticamente todos os seus sistemas prediais é manual. Os sistemas prediais atualmente instalados no PAMB atuam de forma independente, utilizando informações para a realização de suas tarefas sem compartilhar elementos comuns, gerando, com isto, sobreposições de recursos, necessidade de maiores espaços para as instalações, além de requerer maior suporte de manutenção. Como consequência desse quadro, verifica-se:

- multiplicidade de redes de comunicação e cabos elétricos;
- manutenção cara e complicada;
- dificuldade de integrar novos serviços e interligar redes de dados;
- ampla utilização dos produtos “adaptadores”;
- problemas de reposição de peças;
- obsolescência a curto prazo.

5.2.3. CARACTERÍSTICAS DOS SISTEMAS PREDIAIS INSTALADOS

A seguir apresentam-se as principais características dos sistemas prediais presentes no PAMB.

Sistema de Iluminação (SI)

O sistema de iluminação do PAMB conta com luminárias para lâmpadas fluorescentes de 40 W de potência e não existe nenhum tipo de controle para o seu acionamento. Apesar do projeto do prédio considerar o aproveitamento da luz

natural, existem áreas como o centro cirúrgico e o laboratório que permanecem 24 horas com as lâmpadas acesas. O esquema de distribuição das luminárias (todas de teto) é caracterizado por circuitos independentes que são ligados e desligados em horários programados. O sistema de iluminação artificial usado no PAMB é o de iluminação geral e uniforme em todos os pavimentos, caracterizado pela distribuição regular de luminárias no teto. Este sistema proporciona iluminância horizontal sobre a área de trabalho ou circulação, com um certo grau de uniformidade. Alguns casos em que a tarefa visual é de precisão, costuma-se complementar a iluminação geral com uma iluminação local, instalando-se luminárias individuais nos planos de trabalho. A iluminação não é controlada pelo usuário.

O atual sistema de iluminação artificial implantado no prédio não explora as técnicas de automação visando a otimização de energia. De fato, as únicas ações neste sentido estão no uso de lâmpadas e luminárias mais eficientes. As operações sobre esse sistema são manuais, em quadros de distribuição próprios, localizados em cada pavimento.

Sistema de Elevadores (SE)

O sistema de elevadores do PAMB conta com 16 elevadores distribuídos nas zonas de circulação e em vários grupos (vide Tabela 5.2 e Figura 5.2).

Atualmente existem elevadores diferentes para uso geral, serviço e manutenção. A capacidade de carga e dimensões de cada elevador no PAMB variam segundo a sua função (vide Tabela 5.2).

Tabela 5.2 Algumas informações do sistema de elevadores do PAMB

Bloco de circulação	No de elevadores	Lotação (pessoas – kg-f)	Usuários (atuais)
A	4 (2 grupos)	21 – 1470	funcionários e público geral
B	3 (1 grupo)	24 – 1680	funcionários e público geral
C	3 (1 grupo)	24 – 1680	público geral
D	2 (1 grupo)	18 - 1260	funcionários
Prumadas 1 e 2	2	16 –1120	funcionários
Prumadas 1 e 2	2 (manuten.)	3500 Kg	peçoal manutenção

Para o PAMB atualmente, tanto os grupos de elevadores como os elevadores isolados (de serviço), realizam paradas em diferentes pavimentos funcionais, como apresentado na Tabela 5.3.

Tabela 5.3 Andares servidos e paradas dos elevadores do PAMB

Bloco de circulação	Grupos de elevadores	Paradas (andares servidos) (situação atual)
A	Grupo 1	Andar 3 até andar 10 : 8 paradas
	Grupo 2	Andar 1 até andar 10: 10 paradas
B		Andar 1 até andar 7: 7 paradas
C		Andar 3 até andar 7: 5 paradas
D		Andar 1 até andar 9: 9 paradas

Atualmente os elevadores não trabalham em grupo, isto é, cada um deles opera de maneira independente comandados por um ascensorista.

O sistema de elevadores conta com um sistema eletromecânico usado em caso de falta de energia elétrica, que é encarregado de fazer descer (mediante a entrada em operação do gerador de energia de emergência) as cabinas dos elevadores (um de cada vez) até o pavimento térreo (andar 2) ou o quarto pavimento (andar 5) e ali abrir as portas e travar seus movimentos. As cabinas contam também com dispositivos para, por meios mecânicos (chaves), fazer sua movimentação independente, em resposta a um atendimento preferencial ou uma emergência. Atualmente só se dispõe da monitoração dos movimentos dos elevadores (descida e subida) entre os andares.

A lógica de controle instalada está baseada em relés eletromecânicos. Atualmente só é realizada a monitoração geral dos elevadores por meio de lâmpadas sinalizadoras, em um painel instalado em uma sala no subsolo (andar 1), que indicam a direção (descida e subida) de movimento das cabinas dos elevadores e os pavimentos funcionais onde estão sendo efetuadas as chamadas. Os elevadores são acionados por motores elétricos de corrente alternada em um sistema de tração com redutor

O PAMB conta com rampas entre os pavimentos funcionais para facilitar a mobilidade dos pessoal e dos veículos (macas, cadeiras de rodas, carros de roupas, carros de comida, carros de instrumentos, dispositivos portáteis, etc.) e conseguir maior circulação vertical, embora seja proibido o transporte de pacientes em maca por este meio (Min. Saúde, 1995). Estas rampas facilitam o manejo de materiais

cirúrgicos e de materiais da farmácia em conjunto com os montacargas. O PAMB possui 4 montacargas, dois no centro cirúrgico para o transporte de lixo e dois na farmácia para o transporte de medicamentos.

Sistema de Ar Condicionado (SAC)

O sistema de ar condicionado do PAMB é do tipo central e possibilita o resfriamento, e ventilação dos ambientes. A umidade é um fator importante para a garantia da qualidade do ar, mas no PAMB esta não é controlada. O sistema instalado no PAMB é de expansão direta e é constituído por três circuitos distintos: circuito básico de ar, circuito básico de água gelada e circuito frigorífico.

O resfriamento da água gelada usada no sistema de ar condicionado do PAMB é produzido por 4 *chillers*. A água na saída dos *chillers* tem uma temperatura de 7°C. Em cada *chiller*, válvulas de bloqueio interrompem o fluxo de água quando este é desligado. Este sistema conta também com 8 torres de resfriamento nas quais, para permitir a troca de calor, a água quente é circulada em forma de chuva contra uma corrente de ar atmosférico (em uma temperatura mais baixa) no sentido contrário. A quantidade de água que passa por um *chiller* é resfriada usando 2 torres de resfriamento. No caso de um *chiller* ou uma torre de resfriamento necessitar de manutenção, a passagem de água fria é obstruída parcial ou totalmente nas unidades de condicionamento de ar das áreas de menor prioridade com a finalidade de não sobrecarregar o sistema. Fazem parte também do sistema 8 bombas, sendo que uma é de reserva. A função destas bombas é transportar a água entre os *chillers* localizados no subsolo (andar 1) e as torres de resfriamento que estão na cobertura (andar 10), e entre os *chillers* e os locais de condicionamento do ar.

O aquecimento da água quente usada no PAMB é produzido por 2 aquecedores (caldeiras) a vapor. A água na saída dos aquecedores tem uma temperatura de 75°C. O transporte desta água é realizado por 2 bombas. O PAMB conta também com um sistema de aquecimento de água por coletores solares.

O condicionamento de ar é realizado nos *fan-coils*. Ao todo o PAMB conta com 370 *fan-coils* instalados segundo a utilização e prioridade das áreas do hospital. Em alguns pavimentos funcionais há 12 *fan-coils* e em outros até 36 *fan-coils*.

Na maioria das áreas do PAMB a renovação do ar é de cerca de 40% com exceção do centro cirúrgico, banheiros, refeitórios, laboratório, vestiário, farmácia (que utiliza fluxo laminar para a área de produtos injetáveis) onde a renovação é de 100%.

Os equipamentos de condicionamento de ar localizam-se na cobertura (andar 10) do PAMB.

Sistema de Gerenciamento de energia (EMS – *Energy Management Control*) (SGE)

Atualmente não existe um sistema de gerenciamento de energia no PAMB. A energia do PAMB é fornecida por 4 subestações da concessionária de energia elétrica – AES Eletropaulo- que se encontram no sub-solo do hospital. A distribuição da energia elétrica é realizada através de 4 circuitos de média tensão todos interligados. Cada circuito é alimentado por uma subestação. A transferência de carga de um circuito para outro é realizada manualmente em um painel de chaveamento. O comando dos circuitos e geradores de média e baixa tensão é alimentado por um circuito adicional em CA, e a supervisão e acionamento da proteção é realizado em CC, alimentados por um sistema de baterias o que possibilita uma certa independência da concessionária.

Este sistema foi concebido para selecionar as cargas de maneira automática, entretanto devido a problemas de manutenção atualmente opera unicamente de modo manual.

O PAMB conta com um sistema eletromecânico de 3 geradores de energia a Diesel que assumem 1/3 da carga instalada do prédio. A carga refere-se especificamente ao atendimento do:

- centro cirúrgico (iluminação, ar condicionado, etc.);
- sistema de iluminação de emergência;
- área de convênios (pavimento 4);
- sistema de elevadores;
- algumas tomadas para alimentação de equipamentos especiais.

A prioridade mais baixa para a alimentação de energia é a parte administrativa, mas esta parte conta com sistemas de proteção para os computadores. Este sistema demora cerca de 8 segundos para entrar em funcionamento.

Atualmente não existe controle de demanda de energia. O consumo de energia é medido para o prédio como um todo.

Sistema de Controle de Acesso (SCA)

O controle de acesso no PAMB é realizado por pessoal da vigilância que habilita ou restringe o acesso a algumas áreas.

Sistema de Detecção e Combate a Incêndio (SDCI)

O sistema de detecção e combate a incêndio do PAMB consta de sensores de fumaça, rede de *sprinklers*, bomba para pressurização da rede de água de combate ao fogo, portas corta fogo e extintores (de água, pó químico e CO₂). Existe também em cada pavimento uma instalação de hidrante e, no exterior do prédio, uma central de hidrantes.

5.2.4. CONSIDERAÇÕES ESPECÍFICAS SOBRE O PAMB

Antes de aplicar propriamente a abordagem proposta ao PAMB estabelecem-se algumas considerações gerais para o seu estudo.

A. Função

A primeira etapa do projeto de um sistema de automação predial integrado é descrevê-lo de maneira a deixar claros os objetivos a serem atingidos sem a preocupação dos detalhes tecnológicos. Neste contexto, a função principal, isto é, o objetivo maior do PAMB é prestar assistência médica especializada em regime ambulatorial. Este é responsável pelos casos médicos a ele trasladados de algum posto de saúde ou algum hospital regional que não têm condições de atendimento e impede o encaminhamento desnecessário de casos a hospitais de maior complexidade.

B. Requisitos

Considera-se que para se obter um ambiente adequado em um prédio deve-se priorizar qualidades baseadas nas sensações humanas fundamentais (Leite, 1997) que são:

- qualidade espacial;
- qualidade térmica;
- qualidade do ar;
- qualidade acústica;
- qualidade visual;
- integridade do edifício.

As cinco primeiras se referem aos requisitos ocupacionais e as necessidades básicas de saúde, segurança e bem estar. A última se refere a integridade do edifício segundo sua aparência e propriedades críticas de degradação pela umidade, mudanças de temperatura, movimento do ar, radiação, ataques químicos e biológicos, depreação humana e desastres naturais (fogo, inundação, terremotos).

Por outro lado, a ISO define para os edifícios os seguintes requisitos dos usuários:

- estabilidade;
- segurança ao fogo;
- segurança de utilização;
- estanqueidade;
- conforto higrotérmico;
- pureza do ar;
- conforto acústico;
- conforto visual;
- conforto tátil;
- conforto antropodinâmico;

- higiene;
- adaptação dos espaços para usos específicos;
- durabilidade;
- economia.

Neste contexto e considerando a função do PAMB e as normas estabelecidas para prédios hospitalares, os requisitos mais importantes que os sistemas prediais devem atender são:

- flexibilidade;
- interoperabilidade;
- confiabilidade;
- segurança;
- conforto;
- custo;
- compatibilidade;
- expansibilidade;
- manutenção.

Os sistemas prediais considerados no PAMB devem levar em conta a finalidade deste e todos os requisitos antes mencionados. Para o presente estudo os sistemas considerados são os seguintes:

- Sistema de Iluminação (SI);
- Sistema de Ar Condicionado (SAC);
- Sistema de Elevadores (SE);
- Sistema de Controle de Acesso (SCA);
- Sistema de Detecção e Combate a Incêndio (SDCI);
- Sistema de Gerenciamento de Energia (SGE);

Cada um dos sistemas considerados apresentam seus próprios requisitos e funcionalidades que serão considerados quando tratados especificamente.

Como o prédio já está construído e em uso é importante considerar quais destes sistemas se encontram instalados e quais são suas reais condições operacionais.

C. Domínios

A abstração dos domínios permite identificar as comunidades de objetos do ambiente real e os relacionamentos entre estes e o PAMB.

O PAMB incorpora quatro domínios (ou objetos empresa segundo a terminologia ODP) estes são: administração do edifício, os sistemas prediais, as empresas que fornecem os serviços (água, energia elétrica, telefonia e gás), os usuários (que podem ser pacientes, visitantes ou empregados que trabalham nele). A Figura 5.3 apresenta a abstração dos domínios e as comunidades de objetos do ambiente real para o PAMB. Estas entidades foram modeladas segundo a UML como classes dentro de um diagrama de pacotes.

Os objetos da visão empresa podem ser descritos através das interações mostradas entre os domínios e a comunidade. O conjunto de característica (atributos e operações) especificadas deve ser considerado como um conjunto mínimo necessário para o alcance dos objetivos do sistema.

No modelo empresa deve-se considerar subdivisões para o gerenciamento dos serviços, em que atuam os grupos de trabalho que nem sempre podem ser especificados como parte da atividades permanentes do edifício. Um exemplo são as empresas terceirizadas para a manutenção de sistemas específicos como: ar condicionado, elevadores, etc.

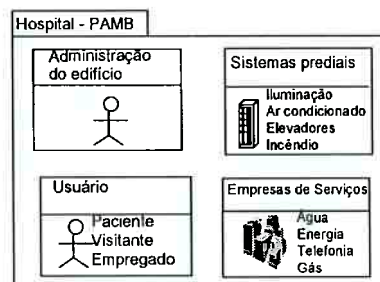


Fig. 5.3 Diagrama da visão empresa –Domínios identificados no PAMB (Interação entre os objetos do ponto de vista empresa)

Cada sistema predial considerado no PAMB corresponde a um domínio com funções de controle específicas e é através da sua interação e integração com outros sistemas que possibilita o cumprimento de seus objetivos de maneira eficiente e econômica.

A funcionalidade do sistema é descrita através dos casos de uso da UML, onde é mostrado como os atores/entidades interagem com o sistema. Este é o caminho para definir as responsabilidades de cada sistema. Este modelo no caso do PAMB é apresentado na Figura 5.4.

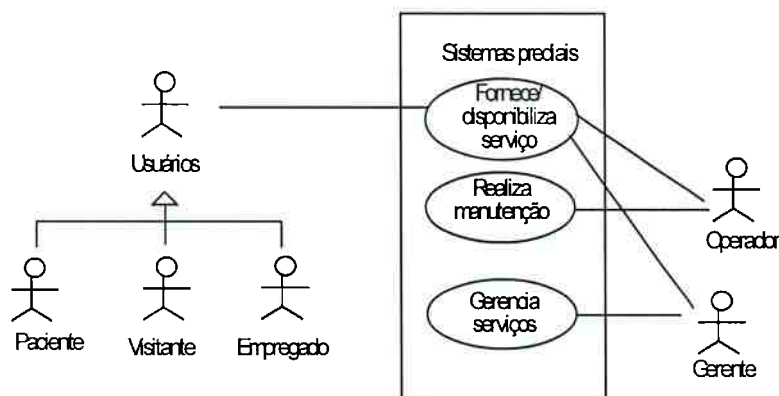


Fig. 5.4 Diagrama de caso de uso –comunidade de usuários do PAMB

Na aplicação da proposta em cada um dos sistemas prediais considerados constroem-se os diagramas de caso de uso para suas respectivas funcionalidades.

D. Definição do Diagrama de Classes para os Sistemas prediais

Desenvolver a uma representação genérica de um sistema é útil para aumentar a modularidade e a flexibilidade do mesmo. Representações genéricas permitem manter a independência do vendedor e a interoperabilidade de maneira a poder acomodar uma variedade de fabricantes e produtos sem uma alteração significativa da estrutura lógica do sistema o que facilita as possíveis extensões futuras.

Neste contexto, realiza-se a decomposição funcional de cada um dos sistemas prediais considerados no PAMB identificando seus objetos e o relacionamento entre eles. Da análise das diferentes funcionalidades dos sistemas prediais e seus elementos constitutivos é derivado um metamodelo de classes que procura abranger as suas características. Este modelo é apresentado na Figura 5.5

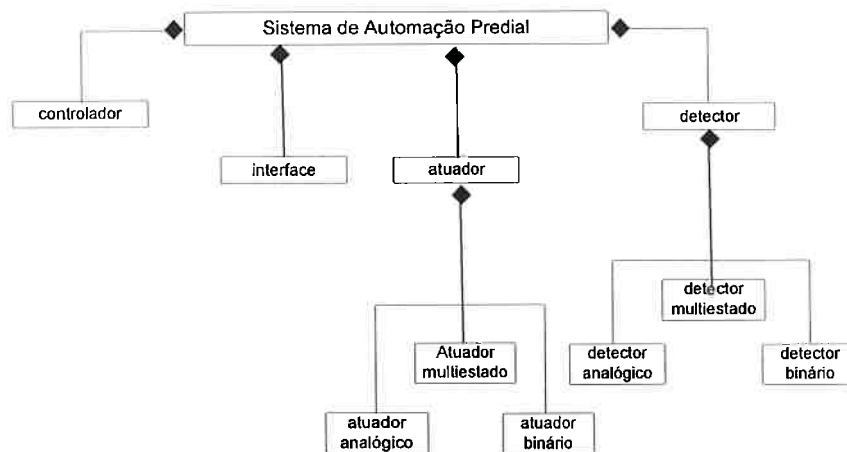


Fig. 5.5 Metamodelo das classes que compõem um sistema predial

Segundo este metamodelo um sistema predial é composto pelas classes detector, atuador, controlador e interface de usuário.

As classes detector e atuador fazem parte da interface do processo físico e representam as variáveis de entrada e de atuação do processo respectivamente.

A classe controlador destina-se a representar as funções de controle de processos. Nesta classe são consideradas as estratégias de controle do sistema.

Na classe interface representam-se as interfaces com os usuários. Estas podem ser de entrada como um teclado, um *mouse*, ou um microfone ou de saída como um *display*, um alto-falante ou vídeo.

A seguir apresenta-se a aplicação da abordagem proposta para os sistemas prediais considerados no PAMB. Salienta-se que a atividade 2 da etapa 3 é apresentada no final do capítulo considerando os vários sistemas prediais.

5.3. PROCEDIMENTO APLICADO NO SISTEMA DE AR CONDICIONADO - SAC

5.3.1. ETAPA 1: MODELAGEM DO DOMÍNIO DO SISTEMA

O sistema de ar condicionado tem como principal propósito manter o controle ambiental visando o conforto térmico das pessoas e as condições adequadas para o funcionamento dos equipamentos. O sistema de ar condicionado controla a

temperatura, umidade e o fluxo de ar os quais são considerados no índice de conforto térmico.

Os componentes do PAMB como paredes, janelas e cortinas interagem com as condições do tempo exterior e influenciam a condição térmica interna. Pessoas e equipamentos têm um impacto no ambiente térmico ao introduzir ao interior calor e umidade.

- Fatores que influenciam o sistema de ar condicionado
- Fatores naturais: insolação solar, vento, temperatura exterior, umidade.
- Fatores de ocupação: número de ocupantes, localização dos ocupantes, tipo de atividade, programação de ocupação, roupas, etc.
- Fatores da edificação: geometria do edifício, orientação, projeção, área de janelas, construção das paredes, materiais das paredes, acabamento interior, materiais do teto e do chão, volume dos quartos, área da superfície construída.
- Fatores do sistema: características das bombas, dos aquecedores, dos chilleres, do sistema de dutos, etc.

As exigências do sistema de ar condicionado do PAMB variam de acordo com a utilização que é realizada das diversas áreas. Assim, existem áreas que exigem apenas condicionamento para fins de conforto como as salas administrativas, salas de consulta e diagnóstico, áreas de circulação entre outros. Algumas áreas necessitam de condicionamento não apenas para o conforto mas também para assepsia tais como salas de cirurgia, salas de pós-operatório e farmácia onde devem ser tomados os devidos cuidados no que se refere aos sistemas de filtragens e renovação de ar, principalmente por envolver trabalhos e tratamentos destinados à análise e erradicação de doenças infecciosas (Normas, 1985).

A. Identificação dos Requisitos:

O sistema de ar condicionado tem um papel importante em um hospital, e deve atender normas como a NBR 6401: Instalações centrais de ar condicionado para conforto, NBR 7256: Tratamento de ar em unidades medição-assistenciais de modo a::

- fornecer calefação, refrigeração e ventilação conseguindo manter a temperatura e umidade adequadas ao conforto humano;
- condicionar os ambientes através de ar livre de contaminantes (microorganismos, bactérias e fungos);
- evacuar o ar contaminante, gases desinfetantes, anestésicos e outros gases residuais; usar mecanismos para evitar a dispersão de fumaça;
- eliminar o risco de fluxo de ar das zonas menos higiênicas para as zonas mais higiênicas. As áreas de maior nível higiênico são no caso do PAMB as salas cirúrgicas, salas de pós-operatório, as salas da farmácia e as adjacências a cada uma de elas. O restante das áreas como salas de exame, rádio diagnóstico, endoscopia, fisioterapia entre outras o nível de higiene é normal. Para manter o nível de higiene é recomendável não reaproveitar o ar de retorno entre áreas de diferentes níveis de higiene (no caso de ser possível o reaproveitamento do ar em áreas com o mesmo nível de higiene, pode-se adotar soluções que permitam poupar energia).
- mudar a ventilação de acordo com a informação do sistema de incêndio;
- insuflar ar (pressão positiva) na área das escadas de emergência, nos vestíbulos dos elevadores imediatamente adjacentes às zonas de fogo (em caso de detecção de fumaça) para minimizar a potencial migração da fumaça.

O sistema de ar condicionado pode funcionar de acordo com as seguintes estratégias (Villani, 2000):

Estratégia em caso de incêndio:

Esta estratégia é adotada para cada área² em caso de detecção de incêndio nela. As medidas que devem ser adotadas são:

- _____

² Área: partes nas que se pode dividir um espaço e que geralmente apresentam características próprias

- para a área onde foi detectada fumaça, renova-se 100% do ar;
- a pressão na área afetada é diminuída através do aumento da velocidade dos ventiladores de retorno do ar;
- insufla-se ar na área das escadas para dispersar possível fumaça

Esta estratégia é desativada quando o sistema de ar condicionado recebe a informação de que não existe mais incêndio.

Estratégia para áreas não utilizadas

No PAMB existem áreas que precisam ser acondicionadas constantemente como as salas do centro cirúrgico. Outras como as salas dos ambulatorios, são utilizadas apenas durante períodos de oito ou dez horas. Assim, dependendo da área pode-se desligar o condicionamento de ar durante os períodos que ela não é utilizada. A decisão de desligamento deriva de situações como:

- quando não existem pessoas na sala;
- quando os equipamentos de uma sala são desligados;
- combinações das opções anteriores.
- Nesta estratégia as medidas a serem tomadas são:
- a caixa de mistura é posicionada para renovação 0%;
- os ventiladores de insuflamento e de retorno são desligados;
- o condicionamento de ar é desligado.

Esta estratégia é desativada quando se detecta que a área vai ou está sendo utilizada.

Estratégia para áreas utilizadas

Esta estratégia é realizada quando é detectada a presença de pessoas na área.

Durante o período de utilização das áreas são tomadas as seguintes medidas:

- a caixa de mistura é posicionada para renovação parcial, ou total, de acordo com a área;

- os ventiladores de insuflamento e de retorno são ligados a velocidade média;
- o condicionamento de ar é acionado.

Em função da temperatura do ambiente é realizado o chaveamento entre o aquecimento e o resfriamento. Se o sistema está sendo aquecido e a temperatura é elevada acima de um limite superior estabelecido, desliga-se o aquecimento e inicia-se o resfriamento. Se o sistema está sendo resfriado e a temperatura no ambiente decresce abaixo de um determinado limite inferior predefinido, desliga-se o resfriamento e inicia-se o aquecimento.

Esta estratégia é desativada quando não restam pessoas na sala.

B. Definição dos Domínios do Sistema e Modelagem dos Casos de Uso:

Sendo o sistema de ar condicionado um domínio, e considerando o caso de uso ‘fornece/disponibiliza serviço’ (ver Figura 5.4) que envolve todos os sistemas prediais, identifica-se para este sistema o diagrama de caso de uso apresentado na Figura 5.6

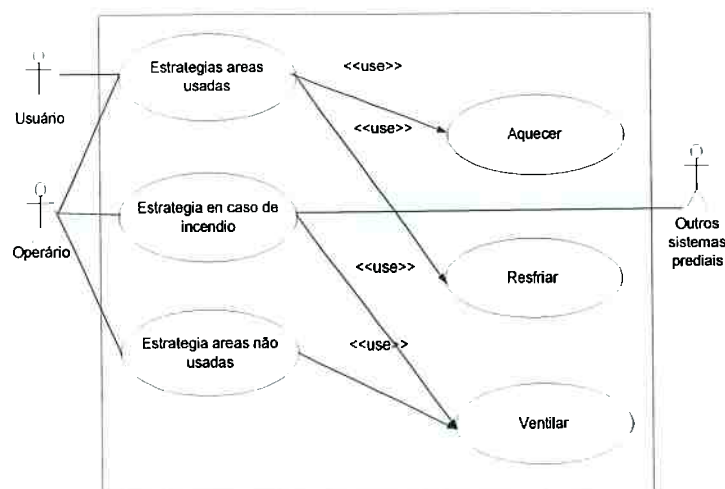


Fig 5.6 Diagrama de casos de uso do sistema de ar condicionado

5.3.2. ETAPA 2: MODELAGEM DOS RELACIONAMENTOS E SERVIÇOS DO SISTEMA

A. Abstração dos Relacionamentos e Interações através de Rede de Petri:

Nesta atividade modelam-se através de rede de Petri condição/evento o fluxo de atividades de cada um dos casos de uso identificados na etapa anterior.

Caso de uso ‘Estratégia para áreas não utilizadas’

Este caso de uso representa a realização da estratégia do sistema de ar condicionado quando não são detectadas pessoas em uma área. O modelo em rede de Petri condição/evento do caso de uso ‘Estratégia para áreas não utilizadas’ é apresentado na Figura 5.7

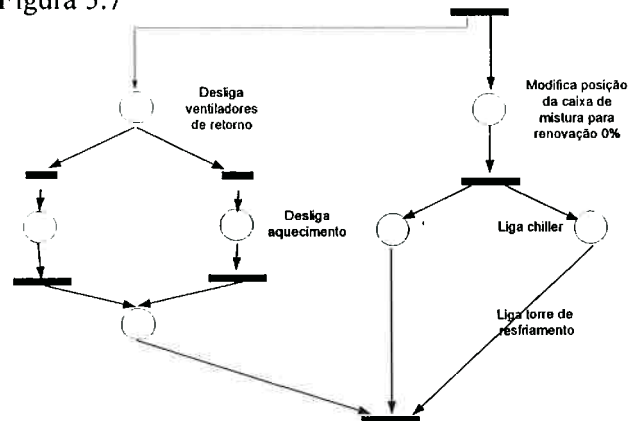
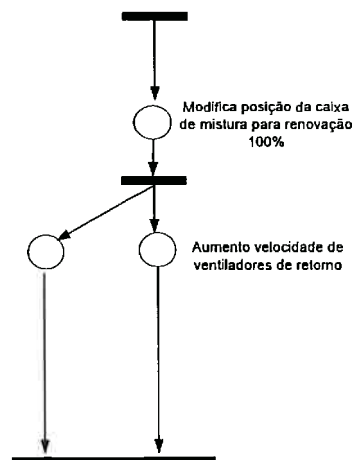


Fig. 5.7 Modelo do fluxo de atividades do caso de uso ‘Estratégia área não utilizada’

Caso de uso: ‘Estratégia em caso de incêndio’

Quando um incêndio é detectado em um área o sistema de ar condicionado recebe esta informação e se encarrega de tomar as medidas pertinentes, neste caso coloca a caixa de mistura para renovação 100% e aumenta a velocidade do ventilador de retorno na área do incêndio. Adicionalmente é insuflado ar na área das escadas para dispersar a fumaça. O caso de uso ‘Estratégia em caso de incêndio’ modelado através da rede de Petri condição/evento é apresentado na Figura 5.8



5. 8 Modelo do fluxo de atividades do caso de uso ‘Estratégia em caso de incêndio’

Caso de uso: 'Estratégia para áreas utilizadas'

Este caso de uso apresenta o fluxo de atividades do sistema de ar condicionado quando é detectada ao menos uma pessoa em uma área em cujo caso coloca-se a caixa de mistura em renovação parcial do 40%, ligam-se os ventiladores e realiza-se o condicionamento do ar na citada área. O caso de uso 'Estratégia para áreas utilizadas' modelado através de rede de Petri condição/evento é apresentado na Figura 5.9

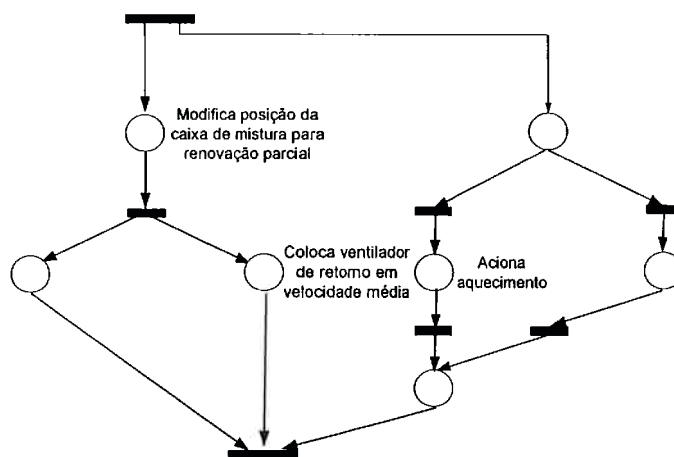


Fig. 5.9 Modelo do fluxo de atividades do caso de uso 'estratégia para área utilizada'

B. Definição e Construção do Diagrama de Classes

A partir dos modelos do fluxo de atividades de cada caso de uso e considerando as funcionalidades do sistema de ar condicionado que se desejam representar identificam-se os objetos que realizam cada atividade. Assim os objetos identificados no sistema de ar condicionado são:

- controlador do sistema de ar condicionado: sua função é controlar os dispositivos nas diferentes zonas e realizar as estratégias de controle;
- interface;
- ventilador de retorno;
- ventilador de insuflamento;
- caixa de mistura;
- serpentina;
- torre de resfriamento ;

- bomba;
- aquecedor;
- chiller.

Uma vez identificados os objetos do sistema de ar condicionado identificam-se as classes do sistema, e suas operações (métodos) (ver Figura 5.10).

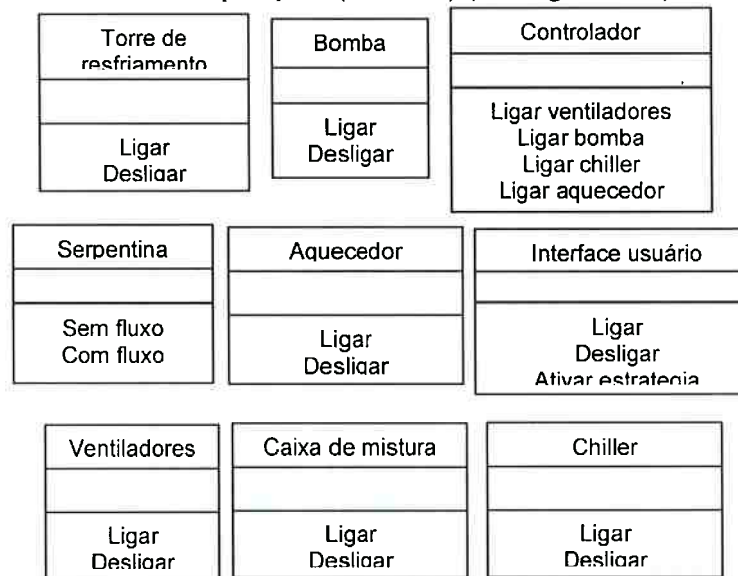


Fig. 5.10 Descrição das classes do sistema de ar condicionado

Os relacionamentos entre as classes são identificados tendo em conta o metamodelo da Figura 5.5 (correspondendo às instâncias deste) e as relações de generalização/especialização, composição/decomposição entre elas. O diagrama de classes do sistema de ar condicionado do PAMB é apresentado na Figura 5.11.

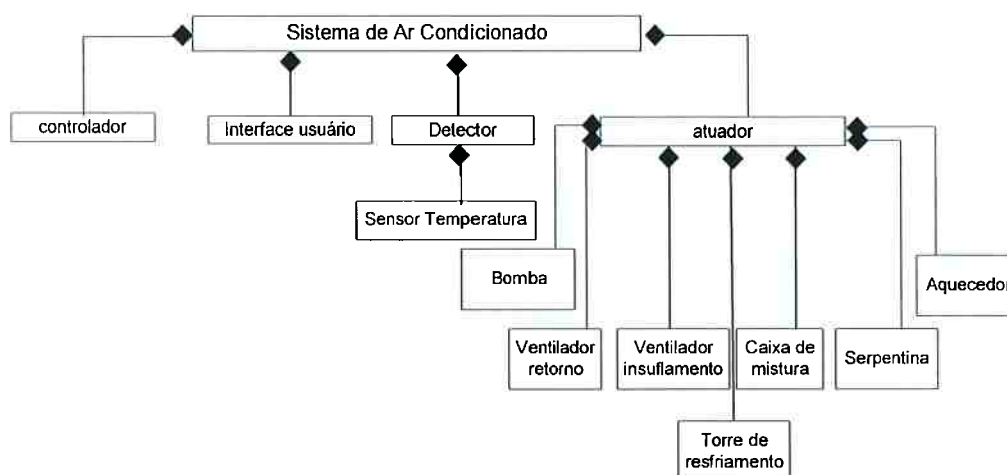


Fig 5.11 Diagrama de classes do sistema de ar condicionado

5.3.3. ETAPA 3: MODELAGEM DINÂMICA E ANÁLISE FORMAL DOS MODELOS

As informações trocadas entre os elementos do sistema de ar condicionado do PAMB envolvem:

- o status dos sensores e atuadores;
- saídas dos sensores;
- sinais de controle aos atuadores;
- o compartilhamento de informações com os seguintes sistemas e finalidades:
 - SCA: para acionar ao condicionamento de ar no caso de ser detectada uma pessoa na área.
 - SGE: em caso de interrupção do fornecimento de energia elétrica da concessionária prioriza-se o condicionamento e ar em algumas áreas como a farmácia e o centro cirúrgico.
 - SDCI: para insuflar ar (pressão positiva) na área das escadas de emergência.

A. Modelagem do Sistema em RPMCO:

Nesta atividade é construído o modelo em RPMCO de cada uma das classes identificadas na etapa 2.

Considerando o formato da mensagem apresentado no capítulo 4, especificam-se as seguintes declarações para a modelagem do sistema de ar condicionado.

Cor E = com e;

Cor MSG = produto End * Dado*Oper;

Cor End = produto Id * Id;

Cor Dado = com ar1 (área 1), ar2 (área 2), ar3 (área 3), ar4 (área 4), arn (área n, onde n total de áreas definidas), vb(velocidade baixa), vn (velocidade normal), va (velocidade alta), pf (posição 0%), pp (posição 40%), pt (posição 100%),

Cor Oper = com li (ligar), de (desligar), pcm (posicionar caixa de mistura)

Cor Id = com VR(Ventilador de retorno), VI (Ventilador de Insuflamento), CXM (Caixa de mistura), SER (Serpentina), TR (torre de resfriamento), Chiller (CH), AQ (Aquecedor), BA (Bomba), IUA (Interface usuário do SAC), CAC (controlador do SAC);

Cor SPR = com SGE, SI, SAC, SCA, SDCI, SE;

Var mes: MSG;

Var em: ID;

Var re: ID;

C₁ - Classe Ventilador de retorno:

Nesta classe consideram-se os estados: ventilador em velocidade baixa (P_{1_1}), media (P_{1_2}) e alta (P_{1_3}). O modelo desta classe em RPMCO é apresentado na Figura 5.12 O modelo para o ventilador de insuflamento é similar a este mas com as inscrições devidas.

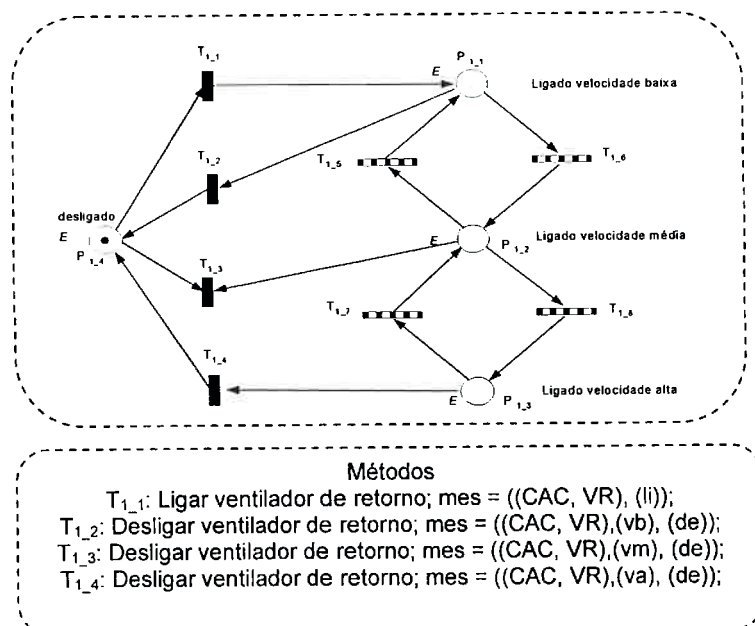


Fig.5. 12 Modelo em RPMCO da classe CI-ventilador de retorno

C₂ - Classe Caixa de mistura:

O modelo desta classe representa os estados da caixa de mistura em renovação 100% (P_{2_1}), renovação parcial 40% (P_{2_2}) e sem renovação 0% (P_{2_3}). A Figura 5.13 representa o modelo desta classe em RPMCO.

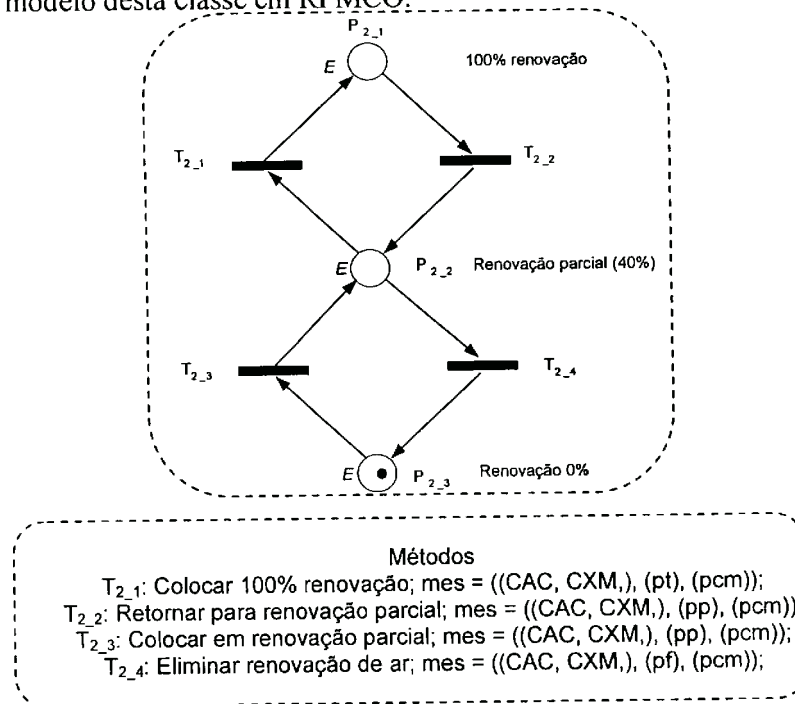


Fig. 5.13 Modelo em RPMCO da classe C₂ – caixa de mistura

C₃ - Classe Serpentina

Nesta classe consideram-se os estados: serpentina com fluxo (P_{3_1}) e serpentina sem fluxo (P_{3_2}). O modelo desta classe em RPMCO é apresentado na Figura 5.14

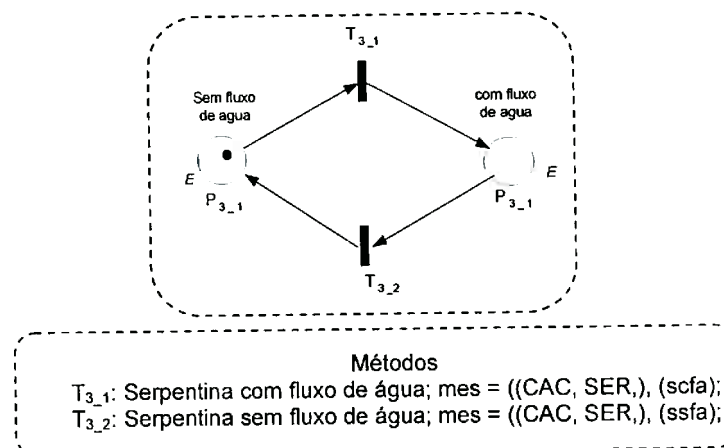


Fig. 5.14 Modelo em RPMCO da C₃ - classe serpentina

C₄ – Classe Torre de resfriamento

Consideram-se nesta classe os estados: torre de resfriamento ligada (P_{4_1}) e torre de resfriamento desligada (P_{4_2}). O modelo desta classe em RPMCO é apresentado na Figura 5.15

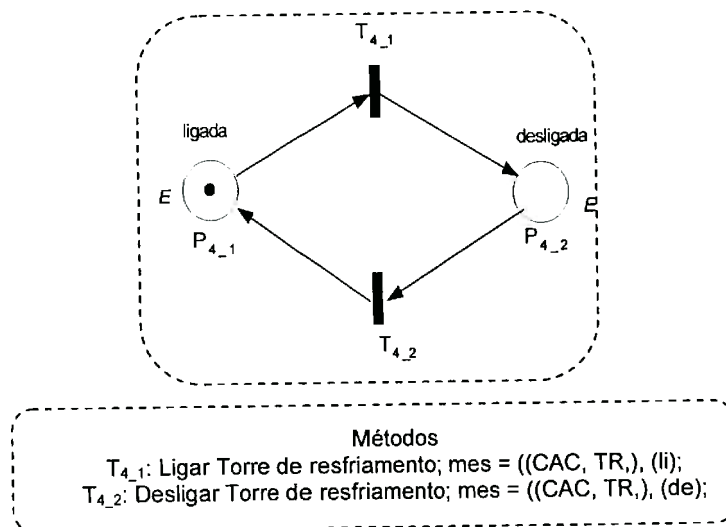


Fig.5.15 Modelo em RPMCO da C₄ - classe torre de resfriamento

C₅ – Classe Chiller

A classe Chiller considera os estados: ligado (P_{5_1}) e desligado (P_{5_2}). O modelo desta classe em RPMCO é apresentado na Figura 5.16

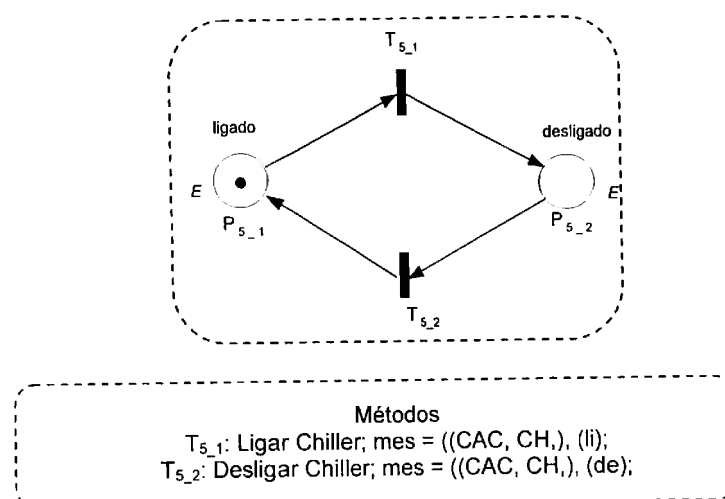


Fig.5.16 Modelo em RPMCO da C₅ - classe Chiller

C₆ – Classe Aquecedor

Esta classe considera os estados do aquecedor como: ligado (P_{6_1}) e desligado (P_{6_2}). O modelo desta classe em RPMCO é apresentado na Figura 5.17

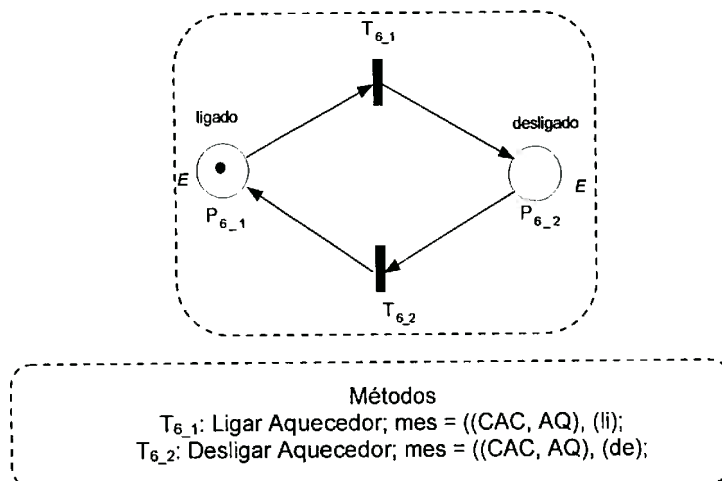


Fig.5.17 Modelo em RPMCO da C₆ - classe Aquecedor

C₇ – Classe Bomba

Nesta classe consideram-se os estados: bomba ligada (P_{7_1}) e bomba desligada (P_{7_2}). O modelo desta classe em RPMCO é apresentado na Figura 5.18

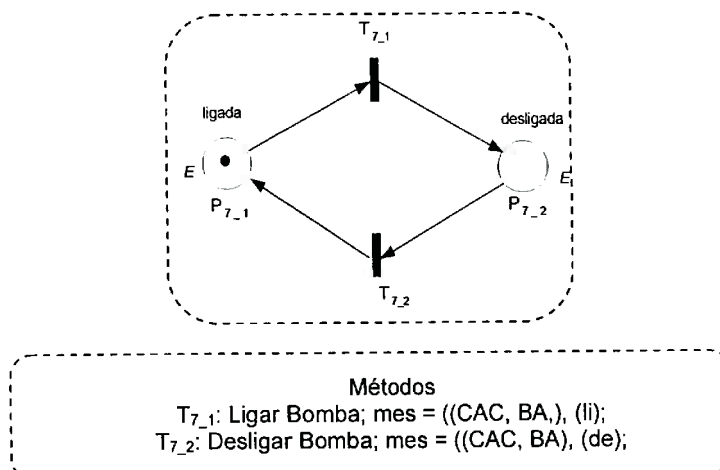


Fig.5.18 Modelo em RPMCO da C₇ - classe Bomba

C₈ - Classe controlador

Esta classe encarrega-se de realizar as ações que permitem garantir o condicionamento do ar: resfriamento ou aquecimento através do acionamento do

chiller ou do aquecedor. Também é quem comanda a realização das diferentes estratégias de controle. O modelo desta classe é apresentado na Figura 5.19

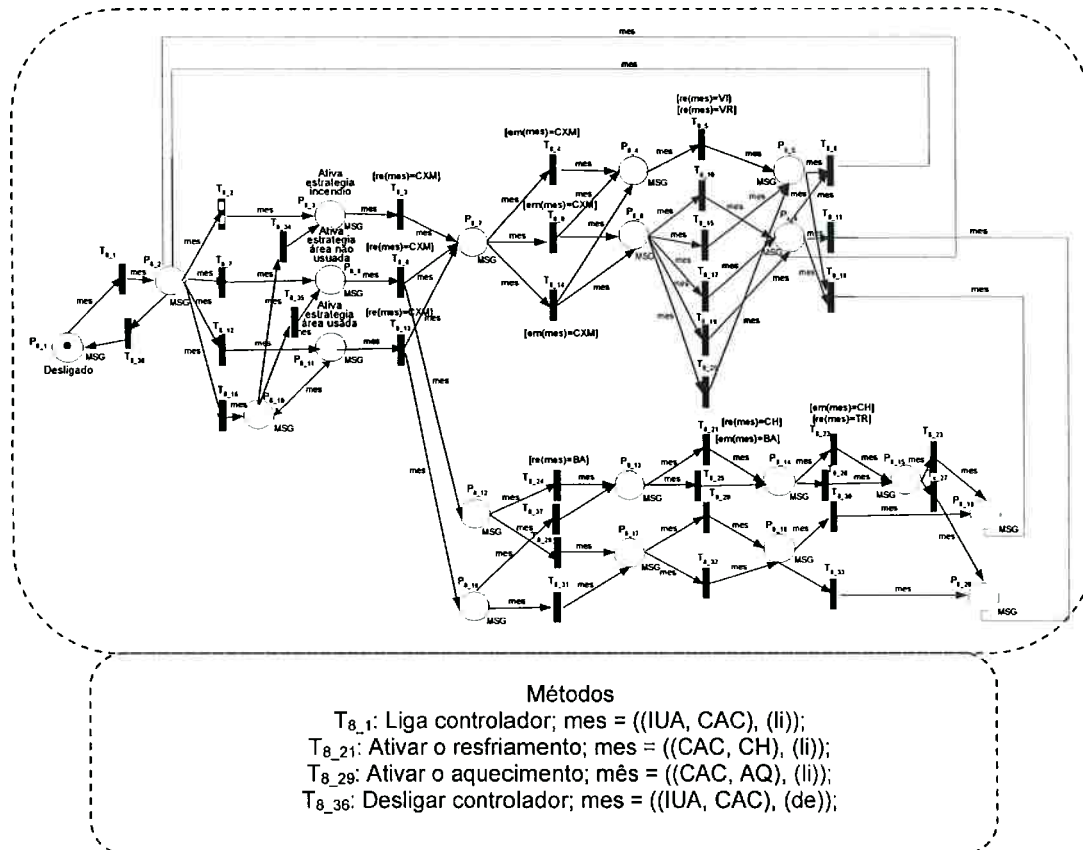


Fig.5.19 Modelo em RPMCO da C₈ - classe Controlador

5.4. PROCEDIMENTO APLICADO NO SISTEMA DE CONTROLE DE ACESSO - SCA

5.4.1. ETAPA 1: MODELAGEM DO DOMÍNIO DO SISTEMA

O sistema de controle de acesso objetiva proporcionar segurança às pessoas e a propriedade física e intelectual. Este sistema reporta um alarma de alerta quando uma condição potencialmente perigosa tem ocorrido.

A. Identificação dos Requisitos:

O SCA deve no caso do PAMB:

- Proteger área através de acesso restrito. Esta medida visa proteger o trabalho que se realiza no interior destas áreas (por exemplo: salas de cirurgia, laboratório, estoque de medicamentos, manejo de dados dos pacientes, farmácia).
- Comunica alarma a uma central externa;
- Detectar a presença de pessoas de maneira a verificar se uma área está sendo usada ou não para desta forma facilitar o subministro de outros serviços

O sistema de controle de acesso pode funcionar em caso de incêndio da seguinte forma:

Estratégia em caso de incêndio:

Quando tem sido detectado incêndio em uma determinada área onde o acesso é restrito, liberam-se as fechaduras eletrônicas segundo políticas de seguridade predeterminadas.

B. Definição dos Domínios do Sistema e Modelagem dos Casos de Uso:

Sendo o SCA é um domínio, e considerando o caso de uso 'fornece/disponibiliza serviço' (ver Figura 5.4) que envolve todos os sistemas prediais, identifica-se para este sistema o diagrama de caso de uso apresentado na Figura 5.20

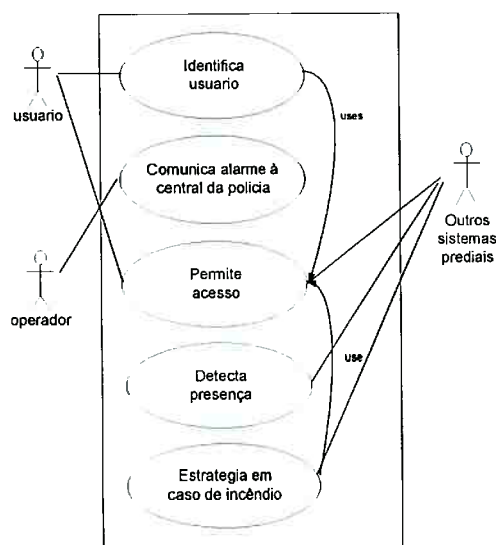


Fig 5.20 Diagrama de casos de uso do sistema de controle de acesso

5.4.2. ETAPA 2: MODELAGEM DOS RELACIONAMENTOS E SERVIÇOS DO SISTEMA

A. Abstração dos Relacionamentos e Interações através de RP:

Nesta atividade modelam-se através de rede de Petri condição evento o fluxo de atividades de cada um dos casos de uso identificados na etapa anterior.

Caso de uso ‘Identifica usuário e avalia acesso’

Este caso de uso representa as atividades realizadas no sistema quando da chegada de um usuário e a avaliação de seu acesso para liberar ou não a entrada. O modelo em rede de Petri condição evento do caso de uso ‘*Identifica usuário e avalia acesso*’ é apresentado na Figura 5.21

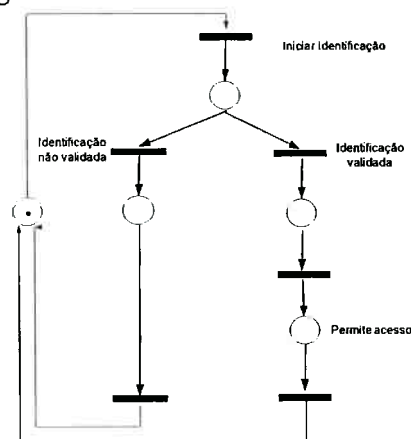
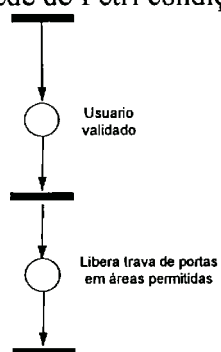


Fig. 5.21 Modelo do fluxo de atividades do caso de uso ‘identifica usuário e avalia acesso’

Caso de uso: ‘Estratégia em caso de incêndio’

Quando um incêndio é detectado liberam-se as fechaduras eletrônicas de áreas predeterminadas para facilitar a evacuação. O caso de uso ‘*Estratégia em caso de incêndio*’ modelado através de rede de Petri condição evento é apresentado na Figura 5.22



5. 22 Modelo do fluxo de atividades do caso de uso ‘Estratégia em caso de incêndio’

Caso de uso: 'Comunica alarma a central'

Este caso de uso apresenta as atividades que são realizadas no sistema de controle de acesso quando é disparada uma alarma.. O caso de uso '*Comunica a central*' modelado através de redes de Petri é apresentado na Figura 5.23

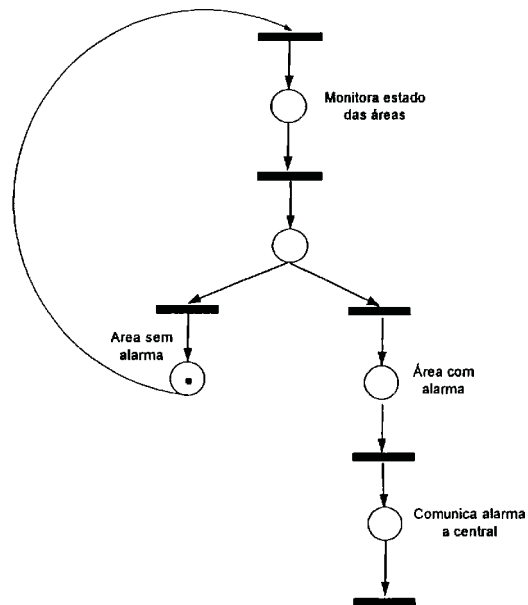


Fig. 5.23 Modelo do fluxo de atividades do caso de uso 'Comunica a central'

Caso de uso: 'Detecta presença'

Este caso de uso apresenta o fluxo de atividades do sistema de controle de acesso quando da detecção de pessoas nas diferentes áreas. O caso de uso '*Detecta presença*' modelado através de redes de Petri é apresentado na Figura 5.24

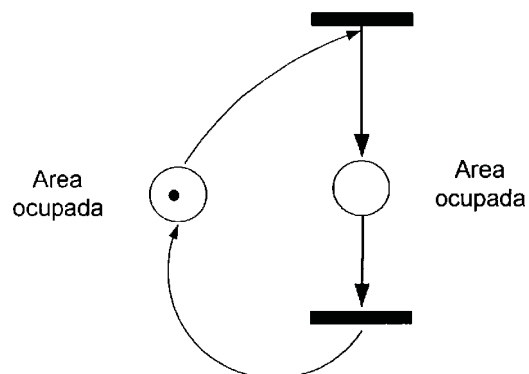


Fig. 5.24 Modelo do fluxo de atividades do caso de uso 'Detecta presença'

B. Definição e Construção do Diagrama de Classes

A partir dos modelos do fluxo de atividades de cada caso de uso e considerando as funcionalidades do SCA que se desejam representar identificam-se os objetos que realizam cada atividade. Assim os objetos identificados no SCA são:

Controlador do SCA: sua função é monitorar e controlar os dispositivos nas diferentes zonas;

- Alarma;
- Fechadura eletrônica;
- Detetor de presença
- Unidade leitora;
- Detetor magnético;

Uma vez identificados os objetos do SCA identificam-se as classes do sistema, e suas operações (métodos) (ver Figura 5.25).

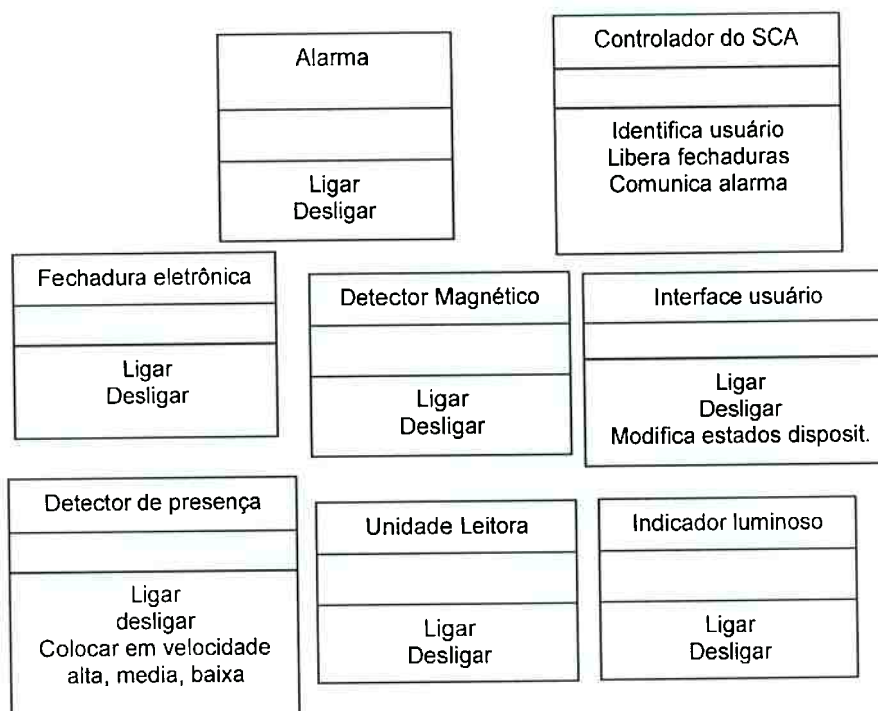


Fig. 5.25 Descrição das classes do sistema de controle de acesso

Os relacionamentos entre as classes são identificados tendo em conta o metamodelo da Figura 5.5 (correspondendo às instâncias deste) e as relações de generalização/especialização, composição/decomposição entre elas. O diagrama de classes do SCA do PAMB é apresentado na Figura 5.26.

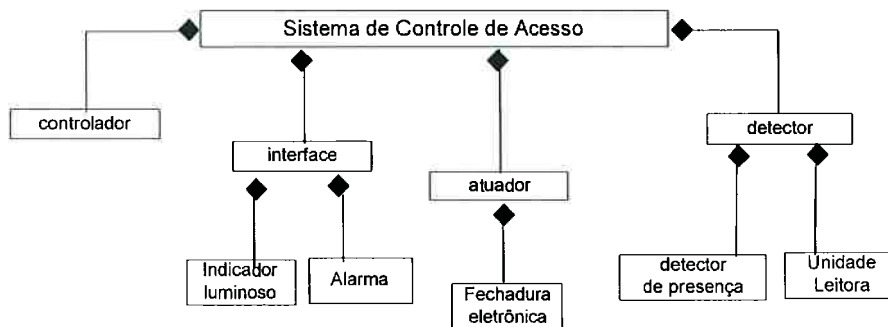


Fig 5.26 Diagrama de classes do sistema de controle de acesso

5.4.3. ETAPA 3: MODELAGEM DINÂMICA E ANÁLISE DOS MODELOS

As informações trocadas entre os elementos do SAC do PAMB envolvem:

- o status dos sensores e atuadores;
- saídas dos sensores;
- sinais de controle aos atuadores
- o compartilhamento de informações com os seguintes sistemas e finalidades:
 - SAC: para que seja tomada a medida correspondente ao ser detectada uma pessoa.
 - SI: ligar lâmpadas na área onde for detectada uma pessoa.

A. Modelagem do Sistema em RPMCO:

Nesta atividade é construído o modelo em RPMCO de cada uma das classes identificadas na etapa 2.

Considerando o formato da mensagem apresentado no capítulo 3, especificam-se as seguintes declarações para a modelagem do SDCI.

Cor E = com e;

Cor MSG = produto End * Dado*Oper;

Cor End = produto Id * Id;

Cor Dado = com ar1 (área 1), ar2 (área 2), ar3 (área 3), ar4 (área 4), arn (área n, onde n total de áreas definidas)

Cor Oper = com li (ligar), de (desligar), acc (aciona), lib (libera), comunica alarma detectada (cad), vau(valida usuário)

Cor Id = com CCA (Controlador do SCA), ALA (Alarma controle de acesso), IL (Indicador luminoso), IUCA (Interface de usuário do SCA), FE (Fechadura eletrônica); UL (unidade leitora); DP (detector de presença), DM (detector magnético).

Cor SPR = com SI, SGEI, SDCI, SAC; SE, SCA

Var mes: MSG;

Var em: ID;

Var re: ID;

C₁ - Classe Alarma:

Nesta classe consideram-se que a alarma pode estar nos estados: o ventilador de retorno pode estar nos seguintes: Ativada (P_{1_1}) ou desativada (P_{1_2}). O modelo desta classe em RPMCO é apresentado na Figura 5.27

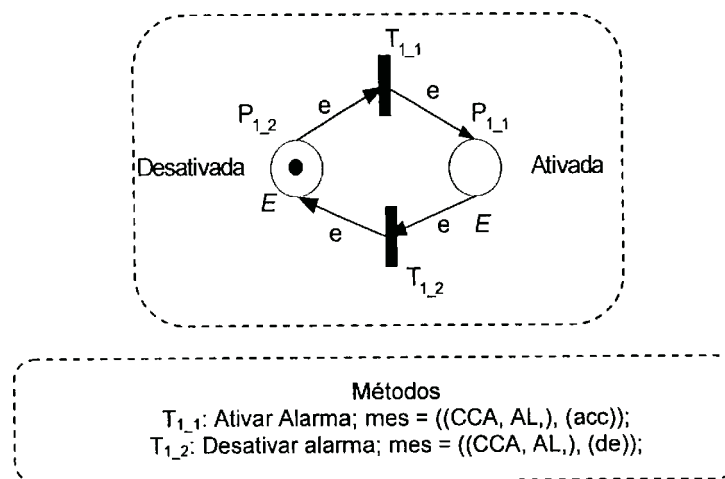


Fig.5.27 Modelo em RPMCO da C₁ - classe alarma

C₂ - Classe Indicador luminoso

Nesta classe consideram-se os estados: indicador luminoso ligado (P_{2_1}) e indicador luminoso desligado (P_{2_2}). O modelo desta classe em RPMCO é apresentado na Figura 5.28

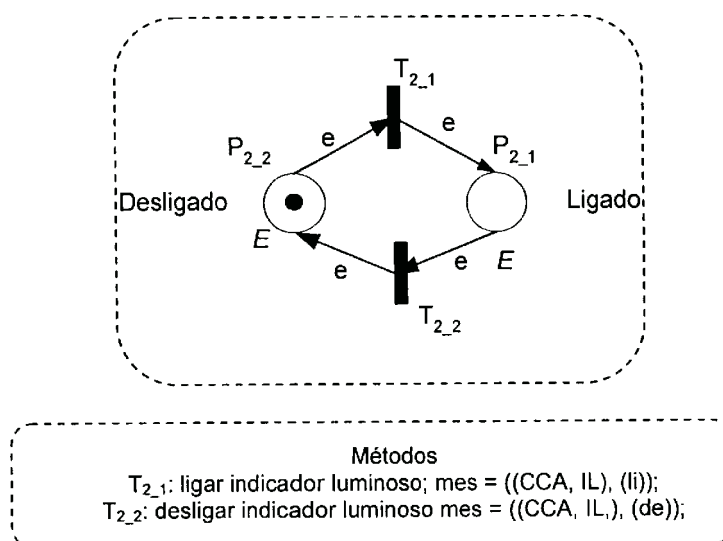


Fig.5.28 Modelo em RPMCO da C₂ – indicador luminoso

C₃ - Classe fechadura eletrônica

O modelo desta classe representa os estados da fechadura eletrônica: ligada e (P_{3_1}) e desligada (P_{3_2}). A Figura 5.29 representa o modelo desta classe em RPMCO.

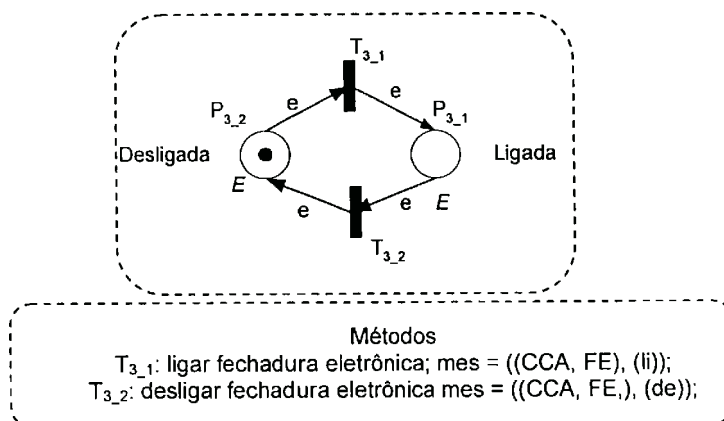


Fig.5.29 Modelo em RPMCO da C3 – fechadura eletrônica

C₄ - Classe Detector de presença

A classe detector de presença considera os estados ligado (P_{4_1}) e desligado (P_{4_2}). A Figura 5.30 representa o modelo desta classe em RPMCO.

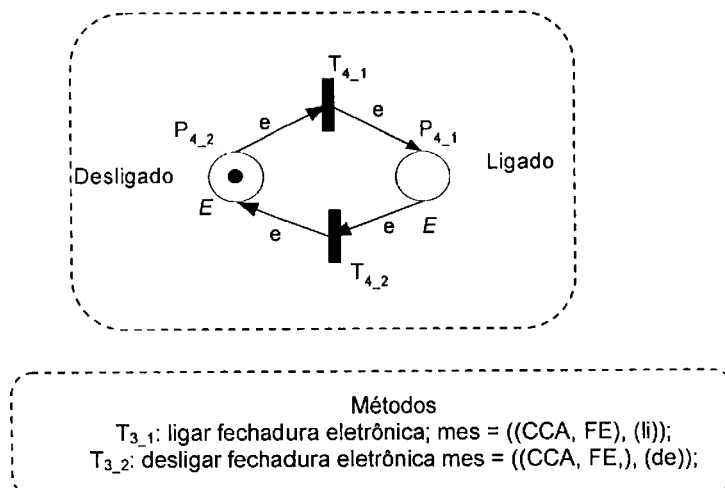


Fig.5. 30 Modelo em RPMCO da classeC4 – detector de presença

5.5. PROCEDIMENTO APLICADO NO SISTEMA DE GERENCIAMENTO DE ENERGIA - SGE

5.5.1. ETAPA 1: MODELAGEM DO DOMÍNIO DO SISTEMA

O principal propósito do sistema de gerenciamento de energia é realizar o controle da demanda e consumo de energia e gerenciar os grupos geradores.

Em um hospital é de fundamental importância que o fornecimento de energia seja mantido de maneira contínua e confiável. No PAMB o fornecimento de energia prioriza-se: para o centro cirúrgico, a farmácia, o laboratório central, a área de atendimento a convênios e o hemocentro. Os sistemas prediais que devem ser mantidos em funcionamento estão o sistema de detecção e combate a incêndio, o sistema de controle de acesso o sistema de ar condicionado para algumas áreas, o sistema de iluminação de emergência, o sistema de elevadores (alguns grupos).

A. Identificação dos Requisitos:

O sistema de gerenciamento de energia considerando o PAMB deve:

- Controlar o consumo de energia dentro de algumas metas estabelecidas atuando se necessário sobre alguns sistema e equipamentos (cargas);
- Gerenciar os grupos de geradores
- Controlar a demanda de energia mantendo-a dentro de metas predeterminadas, atuando, se necessário, sobre alguns sistemas e equipamentos (cargas) e segundo as regras de faturamento vigentes;
- Selecionar, retirar e liberar cargas

A estratégia considerada para este sistema por sua interação com outros sistemas prediais é a estratégia em caso de incêndio e a estratégia falta de energia.

Estratégia em caso de incêndio:

O sistema de gerenciamento de energia deve atuar da maneira seguinte dentro desta estratégia retira o fornecimento de energia na área de incêndio

Esta estratégia é desativada quando se recebe a informação de que não existe mais incêndio.

Estratégia falta de energia:

O sistema de gerenciamento de energia deve priorizar em algumas áreas o condicionamento de ar e o serviço de elevadores. Deve também ativar a iluminação de emergência.

Esta estratégia é desativada quando se recebe a informação de que não existe mais incêndio.

B. Definição dos Domínios do Sistema e Modelagem dos Casos de Uso:

Sendo o SGE um domínio, e considerando o caso de uso 'fornece/disponibiliza serviço' (ver Figura 5.4) que envolve todos os sistemas prediais, identifica-se para este sistema o diagrama de caso de uso apresentado na Figura 5.32

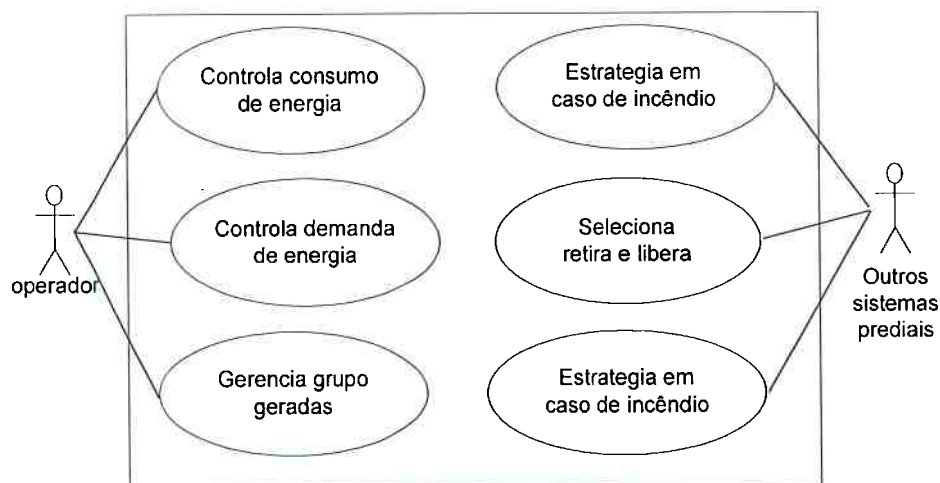


Fig 5.32 Diagrama de casos de uso do sistema de gerenciamento de energia

5.5.2. ETAPA 2: MODELAGEM DOS RELACIONAMENTOS E SERVIÇOS DO SISTEMA

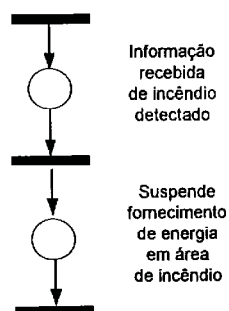
A Abstração dos Relacionamentos e Interações através de RP:

Nesta atividade modelam-se através de rede de Petri condição evento o fluxo de atividades de cada um dos casos de uso identificados na etapa anterior. Os casos

de uso modelados aqui correspondem a casos nos quais se faz evidente a interação com outros sistemas prediais.

Caso de uso: 'Estratégia em caso de incêndio'

Quando um incêndio é detectado em uma área o SGE recebe esta informação e se encarrega de suspender o fornecimento de energia nessa área. O caso de uso '*Estratégia em caso de incêndio*' modelado através da rede de Petri condição evento é apresentado na Figura 5.33



5.33 Modelo do fluxo de atividades do caso de uso 'Estratégia em caso de incêndio'

Caso de uso: 'seleciona, retira e libera carga'

O caso de uso '*seleciona, retira e libera carga*' de Petri é apresentado na Figura 5.34.

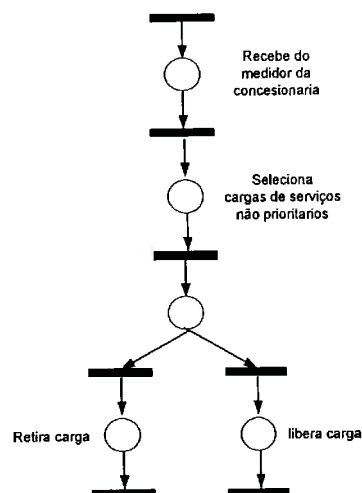


Fig. 5.34 Modelo do fluxo de atividades do caso de uso 'seleciona retira e libera carga'

Caso de uso: 'Estratégia falta de Energia'

Prioriza condicionamento de ar e serviço de elevadores e ativa iluminação de emergência. O caso de uso '*Estratégia falta de Energia*' modelado através da rede de Petri condição evento é apresentado na Figura 5.35

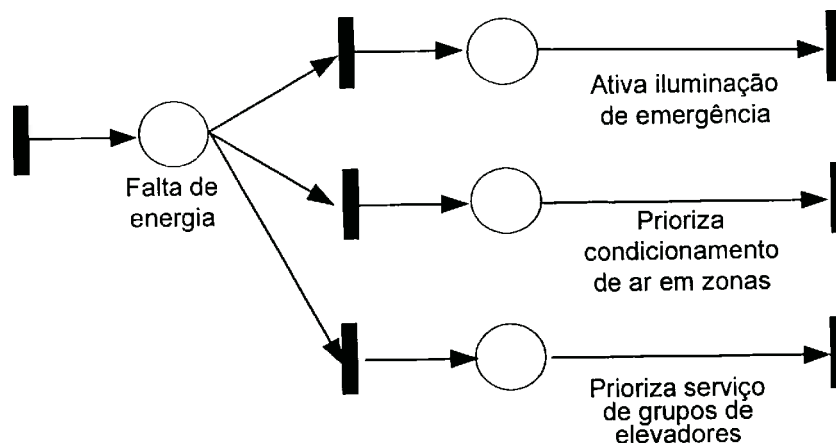


Fig. 5.35 Modelo do fluxo de atividades do caso de uso 'Estratégia falta de Energia'

B. Definição e Construção do Diagrama de Classes

A partir dos modelos do fluxo de atividades de cada caso de uso e considerando as funcionalidades do SGE que se desejam representar identificam-se os objetos que realizam cada atividade. Assim os objetos identificados no SGE são:

controlador do SGE: sua função é controlar os dispositivos nas diferentes zonas e realizar as estratégias de controle;

interface;

Geradores;

Uma vez identificados os objetos do SAC identificam-se as classes do sistema, e suas operações (métodos) (ver Figura 5.36).



Fig. 5.36 Descrição das classes do sistema de gerenciamento de energia

Os relacionamentos entre as classes são identificados tendo em conta o metamodelo da Figura 5.5 (correspondendo às instâncias deste) e as relações de generalização/especialização, composição/decomposição entre elas. O diagrama de classes do SGE do PAMB é apresentado na Figura 5.37.

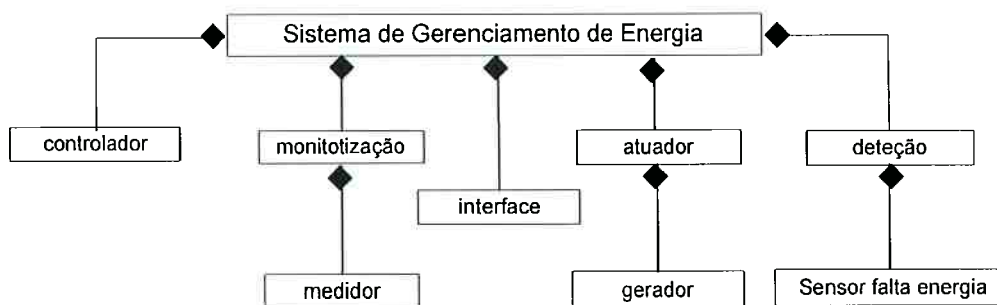


Fig 5.37 Diagrama de classes do sistema de sistema de gerenciamento de energia

5.5.3. ETAPA 3: MODELAGEM DINÂMICA E ANÁLISE FORMAL DOS MODELOS

As informações trocadas entre os elementos do SGE do PAMB envolvem:

- A informação dos medidores;
- sinais de controle aos atuadores
- o compartilhamento de informações com os seguintes sistemas e finalidades:
 - SDCI: para suspender o fornecimento de energia na área de incêndio.
 - SI: para retirar ou liberar cargas por iluminação.
 - SE: para retirar ou liberar cargas por elevadores
 - SAC para retirar ou liberar cargas por condicionamento de ar.

A. Modelagem do Sistema em RPMCO:

Nesta atividade é construído o modelo em RPMCO de cada uma das classes identificadas na etapa 2.

Considerando o formato da mensagem apresentado no capítulo 4, especificam-se as seguintes declarações para a modelagem do SGE.

Cor E = com e;

Cor MSG = produto End * Dado*Oper;

Cor End = produto Id * Id;

Cor Dado = com ar1 (área 1), ar2 (área 2), ar3 (área 3), ar4 (área 4), arn (área n, onde n total de áreas definidas).

Cor Oper = com susp (suspender), selec (selecionar), ret (retirar), lib(liberar)

Cor Id = com GER (gerador), ISGE (Interface de usuário do SGE), CSGE (Controlador do SGE), SFE (sensor falta de energia)

Cor SPR = com SCA, SGE, SDCI, SI, SE, SAC;

Var mes: MSG;

Var em: ID;

Var re: ID;

C₁ – Classe Gerador

Consideram-se nesta classe os estados: Gerador ligado (P_{1,1}) e gerador desligado (P_{2,2}). O modelo desta classe em RPMCO é apresentado na Figura 5.38

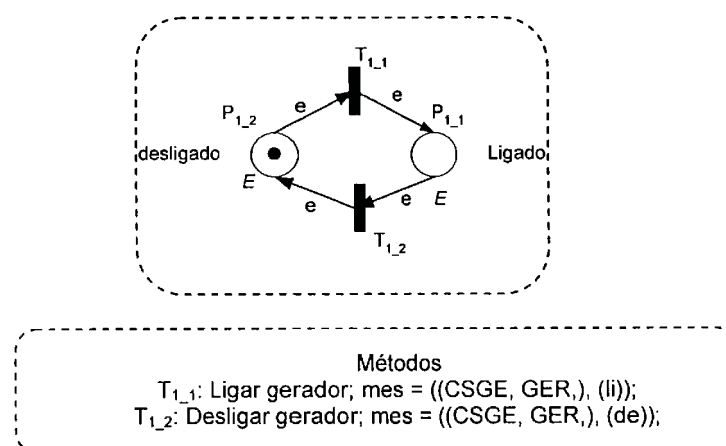


Fig.5.38 Modelo em RPMCO da C₁ – classe gerador

C₂ - Classe controlador

Esta classe encarrega-se de realizar as ações que permitem gerenciar os geradores, retirar e liberar cargas e ativar as estratégias que permitem atuar quando é detetada a falta de energia ou tem acontecido um incêndio.). O modelo desta classe em RPMCO é apresentado na Figura 5.39

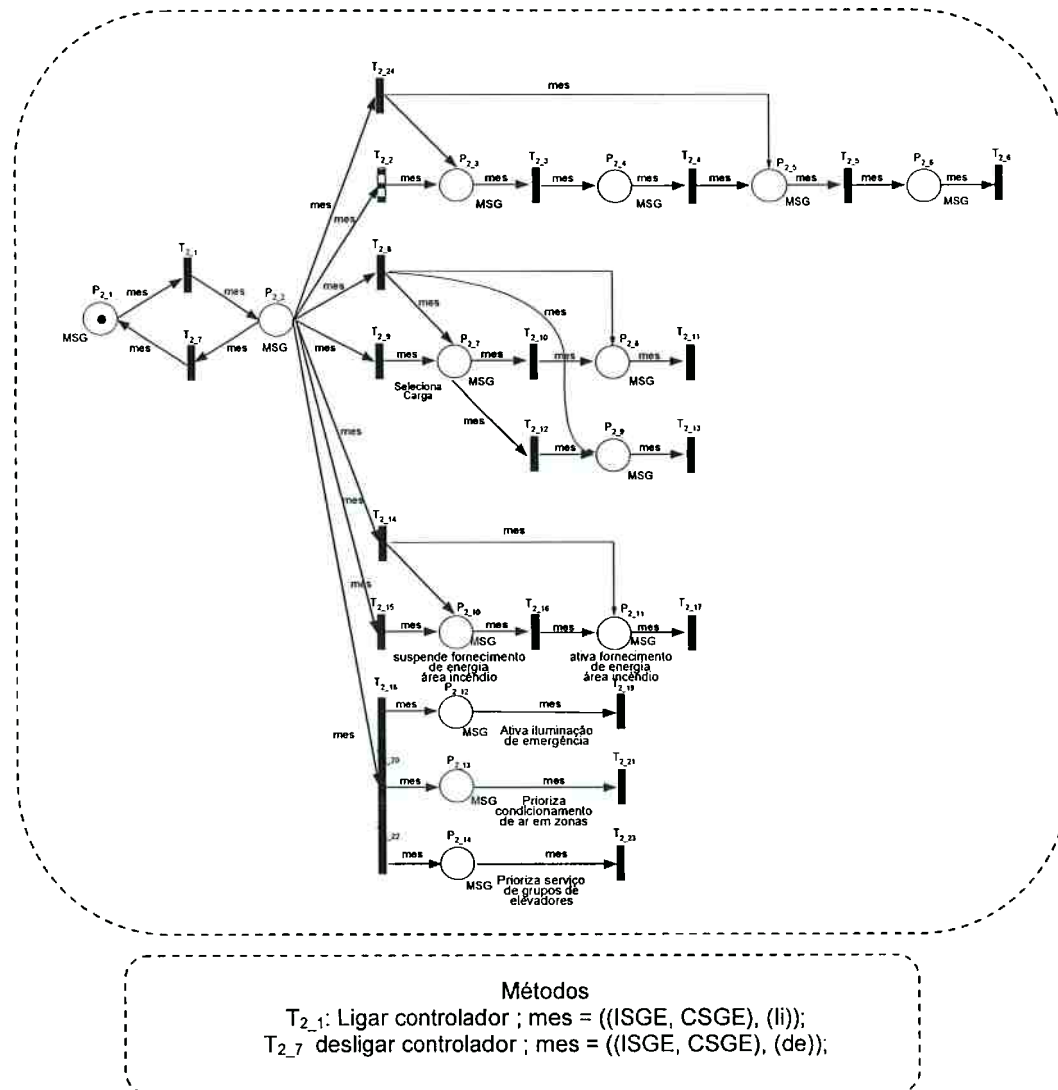


Fig..5.39 Modelo em RPMCO da C₂ - classe controlador

C₃ - Classe interface de usuário

Através desta classe o usuário pode iniciar procedimento em caso de falta de energia, retirar o liberar cargas e suspender o ativar o fornecimento de energia de uma determinada área. A Figura 5.40 apresenta o modelo desta classe.

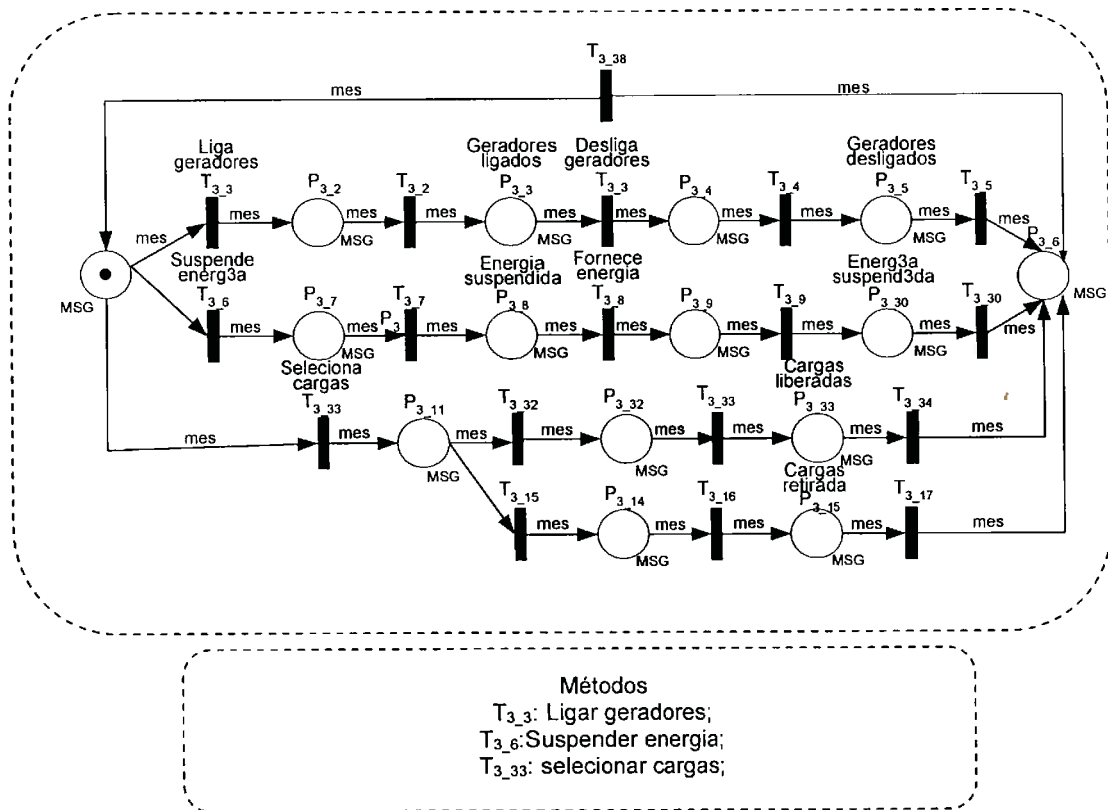


Fig 5.40 Modelo em RPMCO da C₃ - classe interface de usuário

5.6. PROCEDIMENTO APLICADO NO SISTEMA DE ILUMINAÇÃO - SI

5.6.1. ETAPA 1: MODELAGEM DO DOMÍNIO DO SISTEMA

O sistema de iluminação encarrega-se de proporcionar luz artificial nos lugares onde esta é necessária.

Fatores que influenciam o sistema de iluminação

- Fatores naturais: irradiação solar, condições do céu, superfície refletante do chão.
- Fatores de ocupação: numero de usuários, localização dos usuários, tipo de atividade –intensidade lumínica-, programação de ocupação.
- Fatores do edifício: longitude e latitude de sua posição, geometria do edifício, orientação, área de janelas, reflectancia das janelas, transmissão do

vidro, acabamentos interiores, área da superfície interior, superfície de reflectancia interior, .

- Fatores do sistema: lâmpadas, luminárias.

A. Identificação dos Requisitos:

O sistema de iluminação do PAMB deve:

- subministrar luz artificial, ainda em horários onde a luz natural é disponível, pois, pelas características próprias do prédio a iluminação natural não cobre as necessidades de luminosidade das diversas áreas;
- proporcionar diferentes níveis de luminosidade dependendo da atividade realizada na área em questão. As áreas no PAMB que precisam de altos níveis de luminosidade são: o centro cirúrgico, a farmácia industrial e o laboratório central.
- mudar a iluminação de uma área dependendo da ocupação desta.
- ativar a iluminação de emergência em caso de incêndio;

O SI pode funcionar nas seguintes estratégias:

Estratégia em caso de incêndio:

Nesta estratégia quando o SDCI detecta um incêndio em uma determinada área, o SI ativa a iluminação de emergência para esta área. Esta estratégia é desativada quando o sistema de ar condicionado recebe a informação de que não existe mais incêndio.

Estratégia para áreas não utilizadas

No PAMB existem áreas que precisam ser iluminadas constantemente como as salas do centro cirúrgico. Outras como as salas dos ambulatorios, são utilizadas apenas durante períodos de oito ou dez horas. Assim, dependendo da área pode-se desligar a iluminação durante estes períodos de não ocupação informada pelo SCA.

Esta estratégia é desativada quando se detecta que a está sendo utilizada.

Estratégia para área utilizada

Esta estratégia é realizada quando é detectada a presença de pessoas na área e ante este evento o SI liga a iluminação da área. Esta estratégia é desativada quando não restam pessoas na sala

B. Definição dos Domínios do Sistema e Modelagem dos Casos de Uso:

Sendo o SI um domínio, e considerando o caso de uso ‘fornece/disponibiliza serviço’ (ver Figura 5.4) que envolve todos os sistemas prediais, identifica-se para este sistema o diagrama de caso de uso apresentado na Figura 5.41

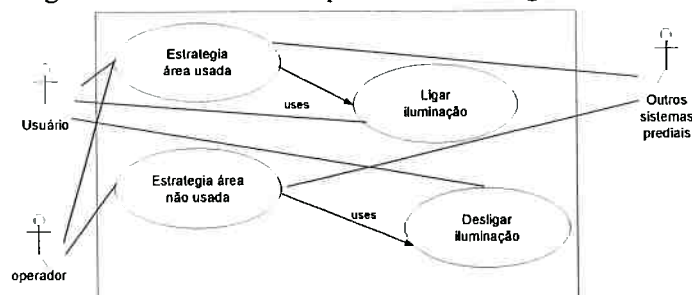


Fig 5.41 Diagrama de casos de uso do sistema de iluminação

5.6.2. ETAPA 2: MODELAGEM DOS RELACIONAMENTOS E SERVIÇOS DO SISTEMA

A. Abstração dos Relacionamentos e Interações através de RP:

Nesta atividade modelam-se através de uma rede de Petri condição evento o fluxo de atividades de cada um dos casos de uso identificados na etapa anterior.

Caso de uso: ‘Ligar iluminação’

Este caso de uso apresenta o fluxo de atividades do SI quando é detectada uma pessoa em uma área em cujo caso se liga a iluminação. O caso de uso ‘ligar iluminação’ modelado através de redes de Petri é apresentado na Figura 5.42. O modelo é o mesmo quando se considera ‘estratégia área usada’

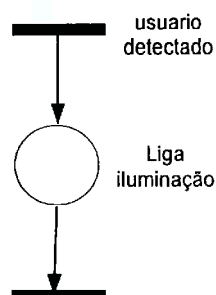


Fig. 5.42 Modelo do fluxo de atividades do caso de uso ‘ligar iluminação’

Caso de uso: 'Desligar iluminação'

Este caso de uso apresenta o fluxo de atividades do SI quando não detecta mais pessoas em uma área desliga a iluminação. O caso de uso *'Desativar iluminação'* modelado através de redes de Petri é apresentado na Figura 5.43. O modelo é o mesmo quando se considera *'estratégia área não usada'*

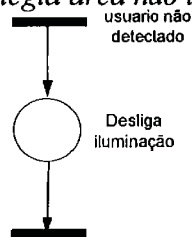


Fig. 5.43 Modelo do fluxo de atividades do caso de uso 'desligar iluminação'

B. Definição e Construção do Diagrama de Classes

A partir dos modelos do fluxo de atividades de cada caso de uso e considerando as funcionalidades do SI que se desejam representar identificam-se os objetos que realizam cada atividade. Assim os objetos identificados no SI são:

- controlador do SI: sua função é controlar os dispositivos nas diferentes áreas e realizar as estratégias de controle;
- interface que indica os estados dos sensores e atuadores e permite a ativação das estratégias por parte de um operário.
- Lâmpada;
- Sensor de luminosidade;
- Interruptor

Uma vez identificados os objetos do SI identificam-se as classes do sistema, e suas operações (métodos) (ver Figura 5.44).

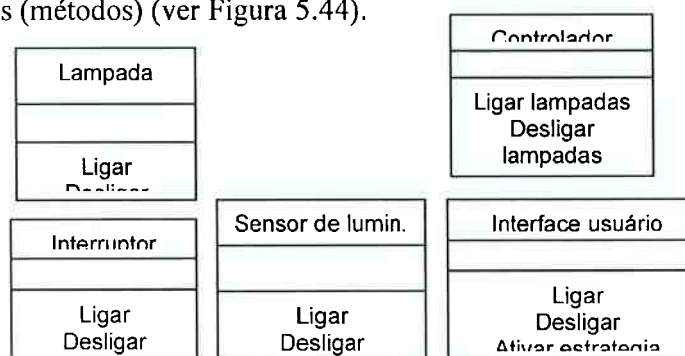


Fig. 5.44 Descrição das classes do sistema de iluminação

Os relacionamentos entre as classes são identificados tendo em conta o metamodelo da Figura 5.5 (correspondendo às instâncias deste) e as relações de generalização/especialização, composição/decomposição entre elas. O diagrama de classes do SI do PAMB é apresentado na Figura 5.45.

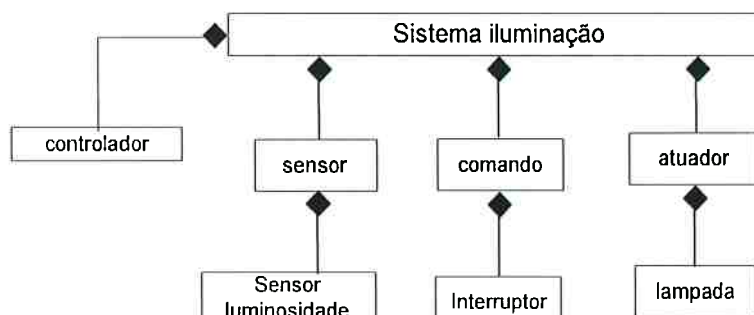


Fig 5.45 Diagrama de classes do SI

5.6.3. ETAPA 3: MODELAGEM DINÂMICA E ANÁLISE FORMAL DOS MODELOS

As informações trocadas entre os elementos do SI do PAMB envolvem:

- o status dos sensores e atuadores;
- saídas dos sensores;
- sinais de controle aos atuadores
- o compartilhamento de informações com os seguintes sistemas e finalidades:
 - SCA: para acionar a iluminação no caso de ser detectada uma pessoa na área.
 - SGE: ao acabar o subministro de energia, prioriza-se a iluminação em algumas áreas como a farmácia e o centro cirúrgico.

A. Modelagem do Sistema em RPMCO:

Nesta atividade é construído o modelo em RPMCO de cada uma das classes identificadas na etapa 2.

Considerando o formato da mensagem apresentado no capítulo 4, especificam-se as seguintes declarações para a modelagem do SDCI.

Cor E = com e;

Cor MSG = produto End * Dado*Oper;

Cor End = produto Id * Id;

Cor Dado = com ar1 (área 1), ar2 (área 2), ar3 (área 3), ar4 (área 4), arn (área n, onde n total de áreas definidas), vb(velocidade baixa), vn (velocidade normal), va (velocidade alta), pf (posição 0%), pp (posição 40%), pt (posição 100%),

Cor Oper = com li (ligar), de (desligar), pcm (posicionar caixa de mistura)

Cor Id = com LAM (Lâmpada), INT (interruptor), SL (Sensor de luminosidade), IUSI (Interface usuário do SI), CIL (controlador do SIL);

Cor SPR = com SCA, SGEI, SDCI;

Var mes: MSG;

Var em: ID;

Var re: ID;

C₁ - Classe Lâmpada

Nesta classe consideram-se os estados: lâmpada ligada (P_{1_3}) e lâmpada desligada (P_{1_2}). O modelo desta classe em RPMCO é apresentado na Figura 5.46

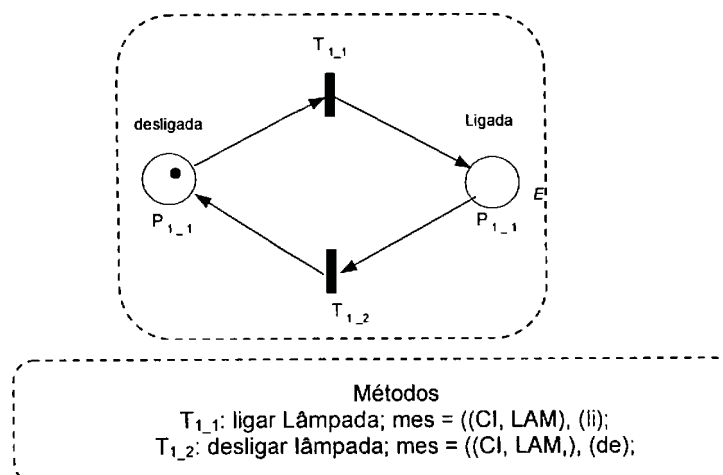


Fig.5.46 Modelo em RPMCO da C₁ -lâmpada

C₂ - Classe sensor de luminosidade

Nesta classe do sensor de luminosidade consideram-se os estados: ligado (P_{2_3}) e desligado (P_{2_2}). O modelo desta classe em RPMCO é apresentado na Figura 5.47

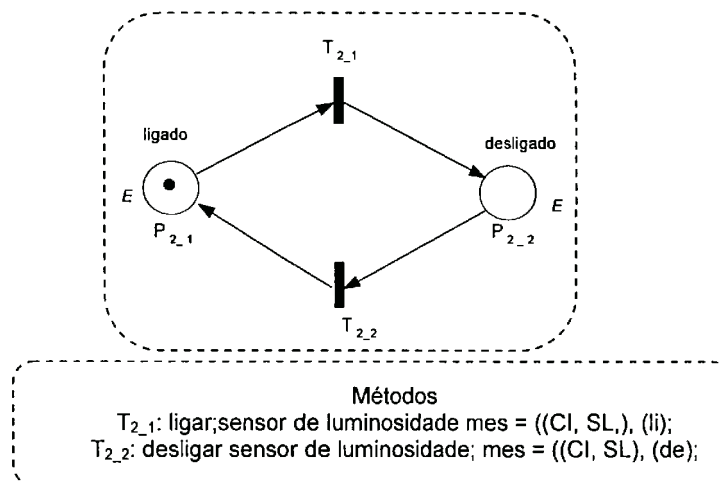


Fig.5.47 Modelo em RPMCO da C₂- classe sensor de luminosidade

C₃ - Classe Interruptor

Nesta classe o interruptor pode estar ligado (P_{3_1}), ou desligado (P_{3_2}). O modelo em RPMCO desta classe é apresentado na Figura 5.48.

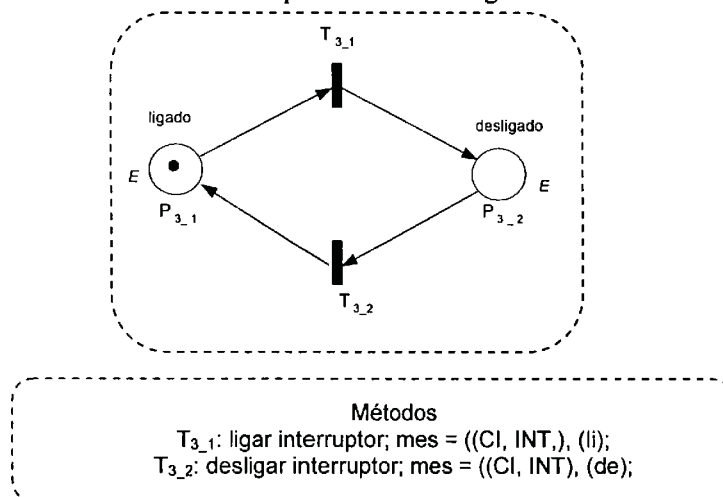


Fig.5.48 Modelo em RPMCO da C₃- classe interruptor

C₄ - Classe controlador

Esta classe é a encarregada de ligar e desligar a iluminação segundo a presença ou não de usuários. O modelo desta classe é apresentado na Figura 5.49

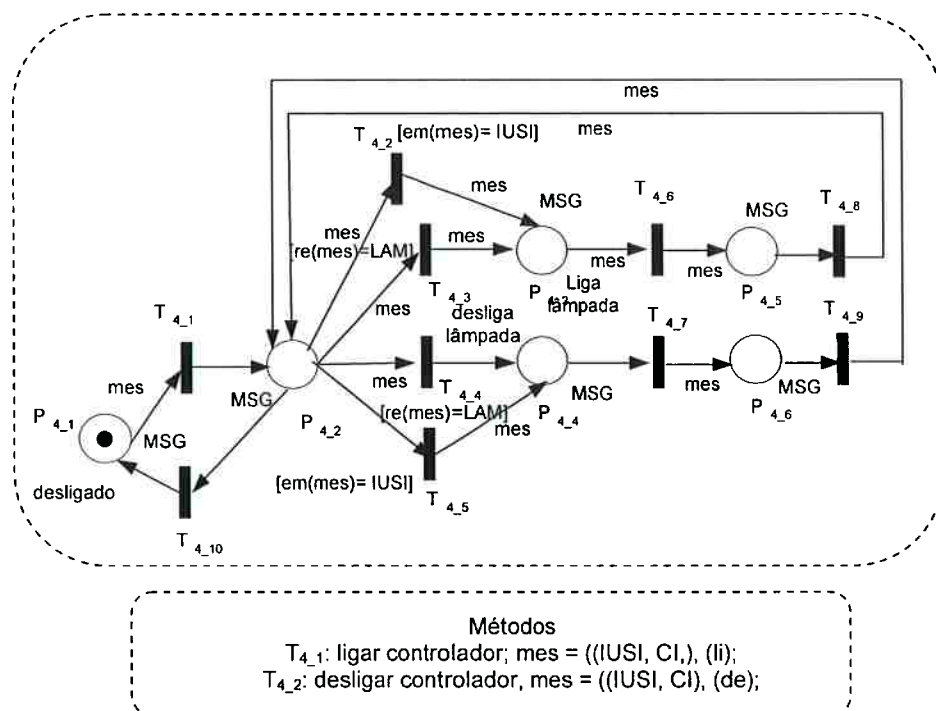


Fig..5.49 Modelo em RPMCO da classe controlador

5.7. PROCEDIMENTO APLICADO NO SISTEMA DE DETECÇÃO E COMBATE A INCÊNDIO – SDCI

5.7.1. ETAPA 1: MODELAGEM DO DOMÍNIO DO SISTEMA

O objetivo do sistema de prevenção e combate a incêndio (SDCI) é a proteção da vida e da propriedade do potencial dano que possa ocasionar o fogo procurando assegurar a continuidade das operações essenciais. No entanto, nenhum SDCI pode eliminar completamente a possibilidade de fogo, dano à propriedade ou perda de vidas. A proteção ao fogo é realizada através de ações de detecção e combate (Liu, et al., 2001).

Os locais de maior risco de incêndio no PAMB segundo as consultas como pessoal técnico são: laboratórios, farmácia, central de material esterilizado, área de armazenagem, salas do grupo gerador e subestação elétrica, área para central de gases, depósitos de resíduos sólidos (lixo) e arquivo.

A. Identificação dos Requisitos:

O SDCI por força da norma NBR9441: Execução de sistemas de detecção e alarme de incêndio – procedimento constitui-se em um sistema autônomo, no entanto com uma alta interação com outros sistemas prediais.

O sistema de prevenção e combate a incêndio deve:

- detectar condições de emergência (presença de fumaça e fogo através de sensores adequados);
- alertar aos bombeiros e ocupantes do prédio;
- combater e extinguir os focos de incêndio em seus estágios iniciais de maneira confiável;
- usar mecanismos para evitar a dispersão da fumaça;
- estabelecer através de sinalização adequada rotas de fuga para todos os ocupantes do prédio inclusive os incapacitados e as pessoas idosas;
- comunicar a ocorrência de incêndio a outros sistemas prediais para que estes possam realizar suas estratégias em caso de incêndio e assim contribuam para as ações do SDCI.
- prover rotinas e funções para o teste de equipamentos de incêndio.
- garantir níveis mínimos nos reservatórios de água e a pressurização dos hidrantes.
- ligar pressurização do ambiente das escadas

Segundo a norma NBR9441/1993 –Sistema Autônomo para Detecção e Alarme de Incêndio- devem ser instalados sistemas completamente independentes, com centrais microprocessadas que interligam detectores, atuadores manuais e automáticos com possibilidade de ajuste remoto do detetor.

B. Definição dos Domínios do Sistema e Modelagem dos Casos de Uso:

Considerando o SDCI como um domínio e modelando o caso de uso ‘fornece/disponibiliza serviço’ (ver Figura 5.5) que envolve todos os sistemas prediais tem-se para este sistema o diagrama de caso de uso apresentado na Figura 5.50

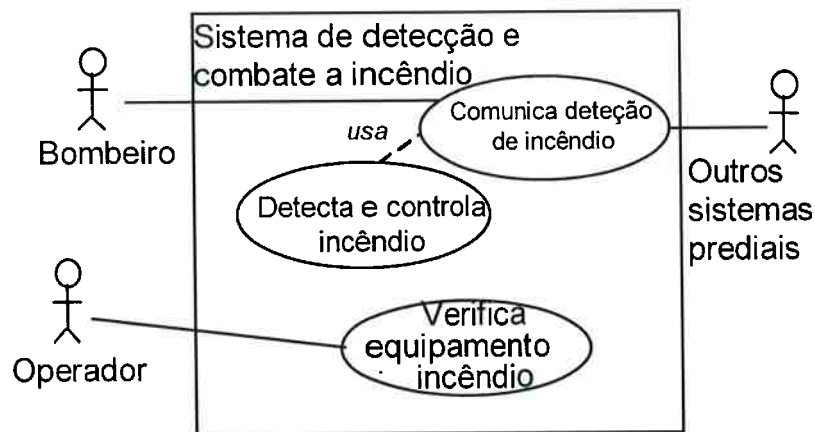


Fig.5.50 Diagrama de casos de uso do sistema de detecção e combate a incêndio

5.7.2. ETAPA 2: MODELAGEM DOS RELACIONAMENTOS E SERVIÇOS DO SISTEMA

A. Abstração dos Relacionamentos e Interações através de rede de Petri:

Nesta atividade modela-se através de rede de Petri condição evento o fluxo de atividades de cada um dos casos de uso identificados na etapa anterior.

Caso de uso ‘Detecta e controla incêndio’

Este caso de uso representa a detecção de um incêndio através do sensor de temperatura ou de fumaça. Uma vez que o incêndio é detetado o sistema executa tarefas para seu combate inicialmente ligando a alarme e comunicando sobre o incêndio aos outros sistemas prediais. A seguir são ligadas a bomba de incêndio e a bomba *jockey* e depois os *sprinklers*. O modelo em rede de Petri condição evento do caso de uso ‘deteta e controla incêndio’ é apresentado na Figura 5.51

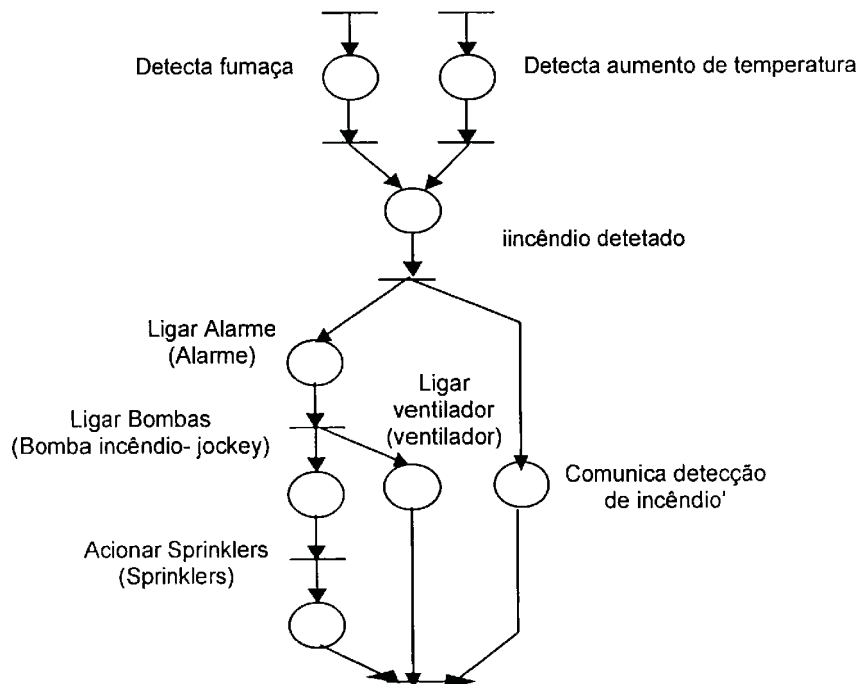


Fig.5.51 Modelo do fluxo de atividades do caso de uso 'detecta e controla incêndio'

Caso de uso: 'Comunica detecção de incêndio'

Quando o incêndio é detetado o SDCI se encarrega de enviar esta informação para outros sistemas prediais para que estes realizem suas respectivas estratégias em caso de incêndio. Assim o SGC providencia a comunicação com os bombeiros, o SI liga a sinalização e a iluminação de emergência, o SAC liga o insuflamento de ar, o SCA destrava portas, o SE conduz os elevadores até o térreo e desliga-os, o SGEI desenergiza as zonas de risco. O caso de uso 'comunica detecção de incêndio' modelado através de rede de Petri condição evento é apresentado na Figura 5.52

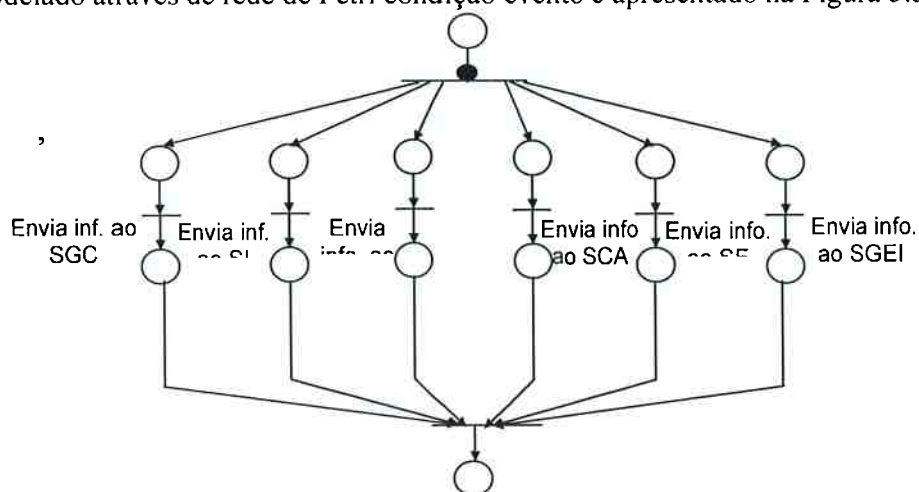


Fig. 5.52 Modelo do fluxo de atividades do caso de uso 'comunica detecção de incêndio'

Caso de uso: 'Verifica Equipamento de Incêndio'

Neste caso de uso verifica-se o funcionamento de cada um dos equipamentos pertencentes ao SDCI. O caso de uso 'verifica equipamento de incêndio' modelado através de redes de Petri é apresentado na Figura 5.53

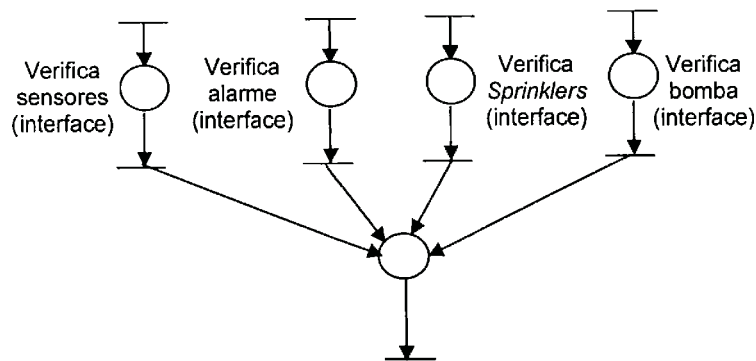


Fig.5.53 Modelo do fluxo de atividades do caso de uso 'verifica equipamento de incêndio'

B. Definição e Construção do Diagrama de Classes

A partir dos modelos do fluxo de atividades de cada caso de uso e considerando as funcionalidades do SDCI que se desejam representar identificam-se os objetos que realizam cada atividade. Assim os objetos identificados no SDCI são:

- controlador de incêndio com a função de controlar os dispositivos em uma zona, executar as ações para os casos contemplados e exercer controle sobre os equipamentos de bombeamento.;
- interface que indica os estados dos sensores e atuadores e permite a ativação de procedimentos em caso de verificação dos equipamentos o simulado de evacuação por incêndio.
- sensor de fumaça;
- sensor de temperatura;
- sprinklers;
- alarme;
- bomba *jockey* (bomba secundaria de incêndio);
- bomba de hidrantes (bomba principal de incêndio).
- ventilador.

Uma vez identificados os objetos do SDCI identificam-se as classes do sistema, e suas operações (métodos) (ver Figura 5.54).

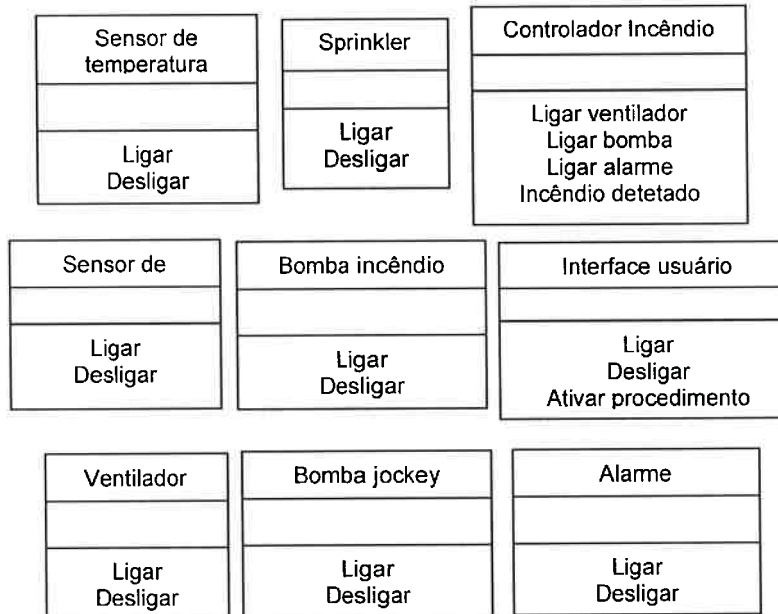


Fig.5.54 Descrição das classes do sistema de detecção e combate a incêndio

Os relacionamentos entre as classes são identificados tendo em conta o metamodelo da Figura 5.5 (correspondendo às instâncias deste) e as relações de generalização/especialização, composição/decomposição entre elas. O diagrama de classes do SDCI do PAMB é apresentado na Figura 5.55.

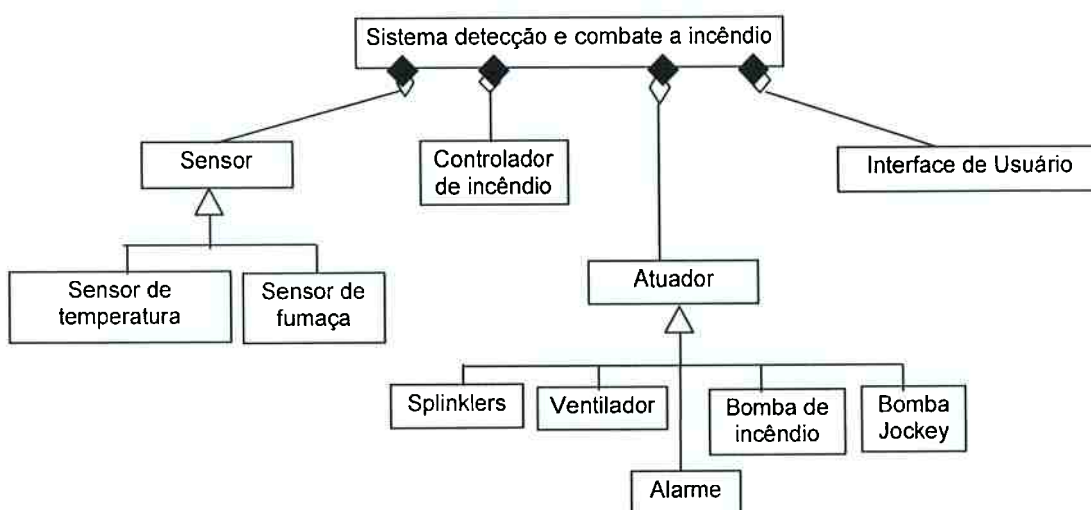


Fig. 5.55 Diagrama de classes do sistema de detecção e combate a incêndio

5.7.3. ETAPA 3: MODELAGEM DINÂMICA E ANÁLISE FORMAL DOS MODELOS

As informações trocadas entre os elementos do SDCI do PAMB envolvem:

- o status dos sensores e atuadores;
- saídas dos sensores;
- sinais de controle aos atuadores
- o compartilhamento de informações com os seguintes sistemas e finalidades:
 - SI: para acionar a iluminação de emergência e a sinalização (através de avisos luminosos e indicadores) das rotas de fuga.
 - SE: para posicionar os elevadores no térreo ou no andar mais recomendado para fuga de eventuais ocupantes e depois desligá-los.
 - SAC: para insuflar ar (pressão positiva) nas escadas de emergência, nos vestíbulos dos elevadores imediatamente adjacentes às zonas de fogo (em caso de detecção de fumaça) para minimizar a potencial migração da fumaça.
 - SGE: para proceder à desenergização da zona afetada, de baixa prioridade e zonas próximas que possam vir a aumentar a magnitude do fogo (evitando que a rede elétrica alimente o fogo). Fornecer energia através dos geradores para manter a iluminação de emergência, as alarmes, as bombas, comunicações e o insuflamento de ar.
 - SCA: para destravar o acesso de algumas zonas (fechar as portas de proteção contra incêndio)
 - SGC: para estabelecer contato com os bombeiros. E informar o estado do incêndio.

A. Modelagem do Sistema em RPMCO:

Nesta atividade é construído o modelo em RPMCO de cada uma das classes identificadas na etapa 2.

Considerando o formato da mensagem apresentado no capítulo 4, especificam-se as seguintes declarações para a modelagem do SDCI.

Cor E = com e;

Cor MSG = produto End * Dado * Oper;

Cor End = produto Id * Id;

Cor Dado = com ta (temperatura alerta), tn (temperatura normal), fa (fumaça alerta), fn (fumaça normal)

Cor Oper = com li (ligar), de (desligar), idt (comunica incêndio detetado), ni (comunica não incêndio), at (ativar procedimento), ta (temperatura alerta), tn (temperatura normal), fa (fumaça alerta), fn (fumaça normal), ab (abrir), fe (fechar), veq (verifica equipamento);

Cor Id = com SF (Sensor de fumaça), CI (Controlador de incêndio), AL (Alarme), ST (Sensor de temperatura), VE (ventilador), BI (Bomba de incêndio), BJ (Bomba jockey), IUI (Interface de usuário do SDCI);

Cor SPR = com SI, SAC, SE, SCA, SGEI, SGC |;

Var mes: MSG;

Var em: ID;

Var re: ID;

Classe sensor de fumaça

Nesta classe considera-se os estados: detecção de fumaça (P_{2_3}) e fumaça não detetada (P_{2_2}). O modelo desta classe em RPMCO é apresentado na Figura 5.56

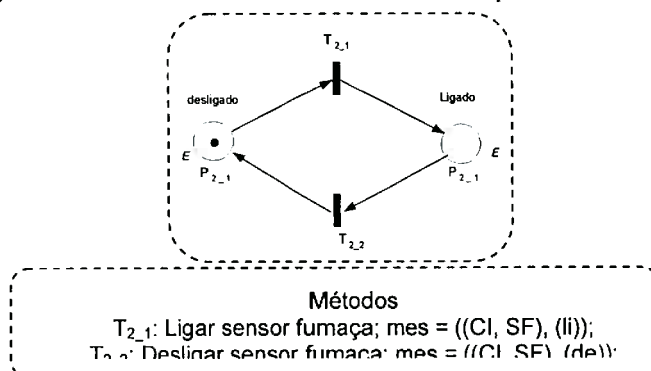


Fig. 5.56 Modelo em RPMCO da classe sensor de fumaça

Classe sensor de temperatura

O modelo desta classe representa os estados de um sensor de temperatura em funcionamento normal (P_{3_2}) ou quando a temperatura é considerada de risco (P_{3_3}). A Figura 5.57 representa o modelo desta classe em RPMCO.

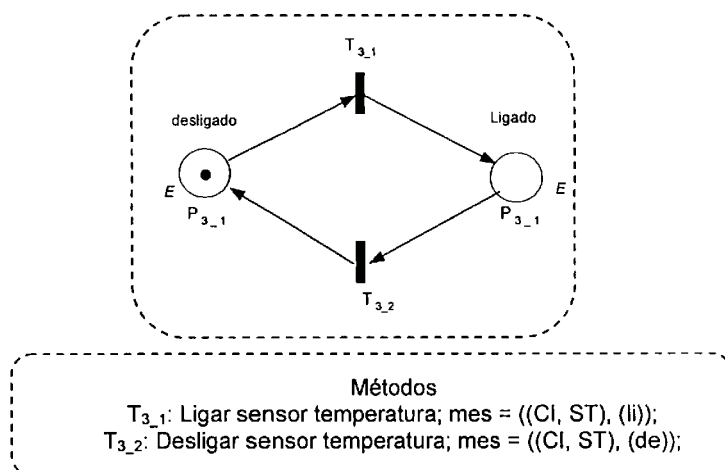


Fig. 5.57 Modelo em RPMCO da classe sensor de temperatura

Classe sprinkler

A classe *sprinkler* considera os estados aberto (P_{5_2}) e fechado (P_{5_1}). A Figura 5.58 representa o modelo desta classe em RPMCO.

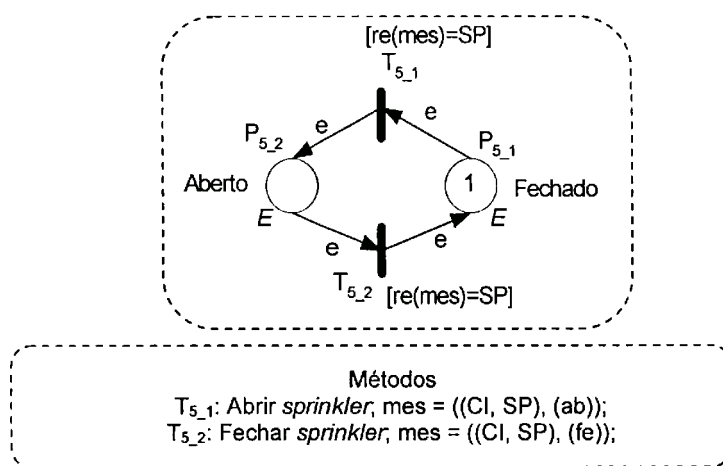


Fig. 5.58 Modelo em RPMCO da classe sprinkler

Classe ventilador

Um objeto da classe ventilador pode estar no estado desligado (P_{6_1}) ou ligado (P_{6_2}). O modelo desta classe é apresentado na Figura 5.59

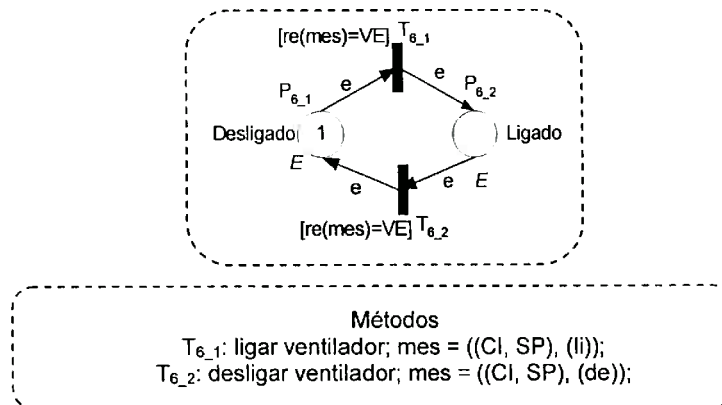


Fig5.59 Modelo em RPMCO da classe ventilador

Classe bomba de incêndio

Nesta classe, um objeto pode estar no estado desligado (P_{7_2}) ou ligado (P_{7_1}).

A Figura 5.60 apresenta o modelo desta classe.

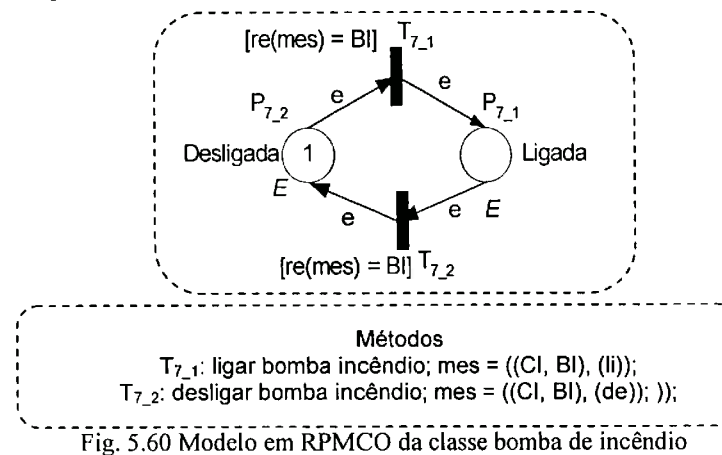


Fig. 5.60 Modelo em RPMCO da classe bomba de incêndio

Classe bomba jockey

Um objeto desta classe pode estar no estado desligado (P_{8_2}) ou ligado (P_{8_1}). A

Figura 5.61 apresenta o modelo em RPMCO desta classe.

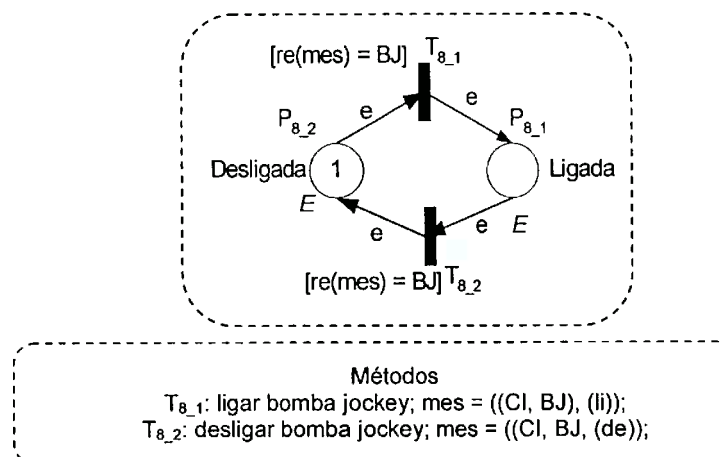


Fig. 5.61 Modelo em RPMCO da classe bomba jockey

Classe Alarme

Nesta classe o alarme pode estar ligada (p9_1) ou desativada (p9_2). o modelo desta classe é apresentado na figura 5.62.

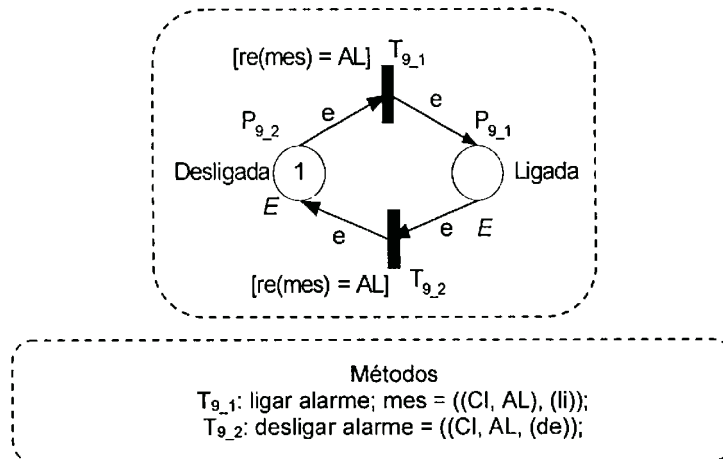


Fig. 5.62 Modelo em RPMCO da classe alarme

Classe controlador

Uma vez o fogo é detectado pelos sensores do sistema, esta classe é responsável por tomar as ações apropriadas para combatê-lo como ligar o alarme, as bombas e o ventilador de insuflamento. O modelo desta classe é apresentado na Figura 5.63

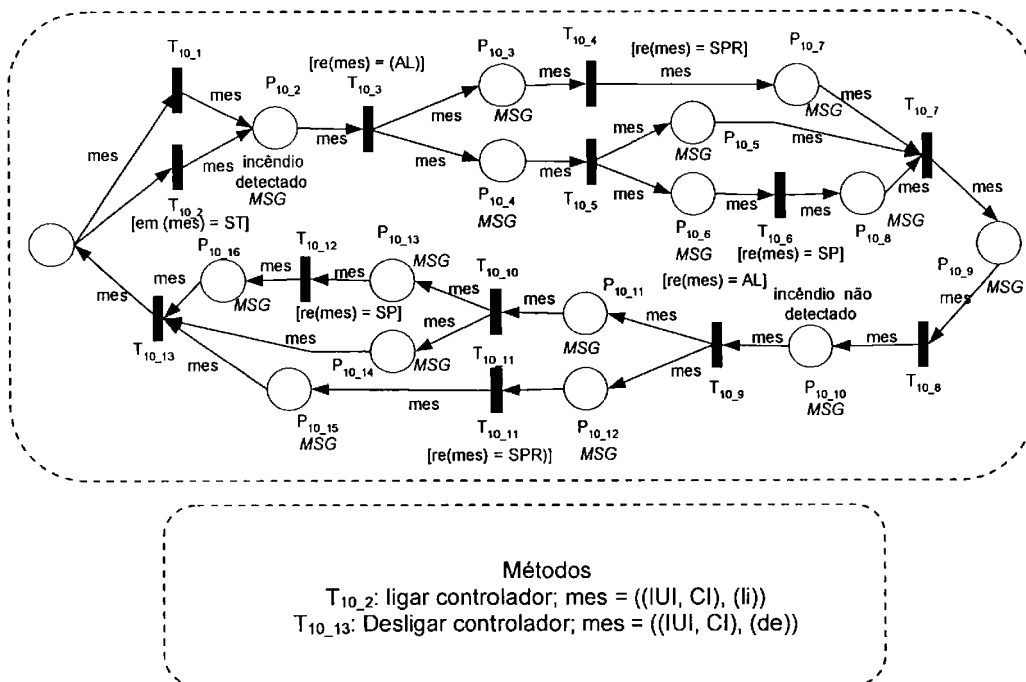


Fig. 5.63 Modelo em RPMCO da classe controlador

B. Análise dos Modelos

A continuação é apresentada a análise dos modelos realizados para os subsistemas prediais considerados no PAMB.

Validação

A simulação é um importante meio para depurar e validar os modelos realizados e ajudar no entendimento do comportamento dinâmico do sistema. Assim, para esta atividade foi usado um *software* para criar, editar, simular e analisar redes de Petri coloridas, o *CPN Tools* desenvolvido no *Department of Computer Science* da University of Aarhus da Dinamarca.

Neste software os modelos são criados usando para as inscrições da rede a linguagem de programação ML (Jensen, 1996). Esta criação é conferida incrementalmente, isto é, cada vez que um elemento é inserido à rede, o *software* testa se a sua sintaxe é correta e identifica os possíveis erros nas inscrições e na estrutura da rede. O presente trabalho utilizou-se desta característica do programa para avaliar cada um dos módulos criados em RPMCO. Em cada um deles criaram-se LUGARES e TRANSIÇÕES adicionais para simular a troca de mensagens de um módulo com outros módulos.

As simulações destes módulos revelaram seu papel na identificação de erros e permitiram rever e reconsiderar os modelos propostos. Assim, os erros foram localizados e corrigidos, assim como os módulos foram aprimorados.

Para simular os vários sistemas modelados no PAMB e a sua integração e levando em conta a complexidade que este prédio apresenta e o que representa sua consideração completa na simulação em termos de tempo de execução e processamento, considerou-se que para melhor ilustrar as vantagens e limites da abordagem proposta o mais adequado seria selecionar algumas áreas para sua aplicação. Assim e pelas características (isto é, os diferentes serviços que apresenta) do quarto pavimento do PAMB, as áreas em questão foram nele consideradas.

No quarto pavimento são prestados os seguintes serviços:

- marcação de consultas;
- consulta médica a pacientes conveniados;

- arquivo médico (arquivos em papel);
- farmácia (venda de medicamentos);
- salas administrativas.

Como exemplo forma consideradas 4 áreas dentro deste pavimento (ver Figura 5.64)

Área 1: composta por 3 salas de consulta médica a pacientes conveniados;

Área 2: composta por 1 sala de marcação de consulta;

Área 3: composta por 1 sala de farmácia;

Área 4: composta por 1 sala de arquivo médico

Os dispositivos pertencentes aos diversos sistemas prediais consideradas são apresentados na Tabela 5.4

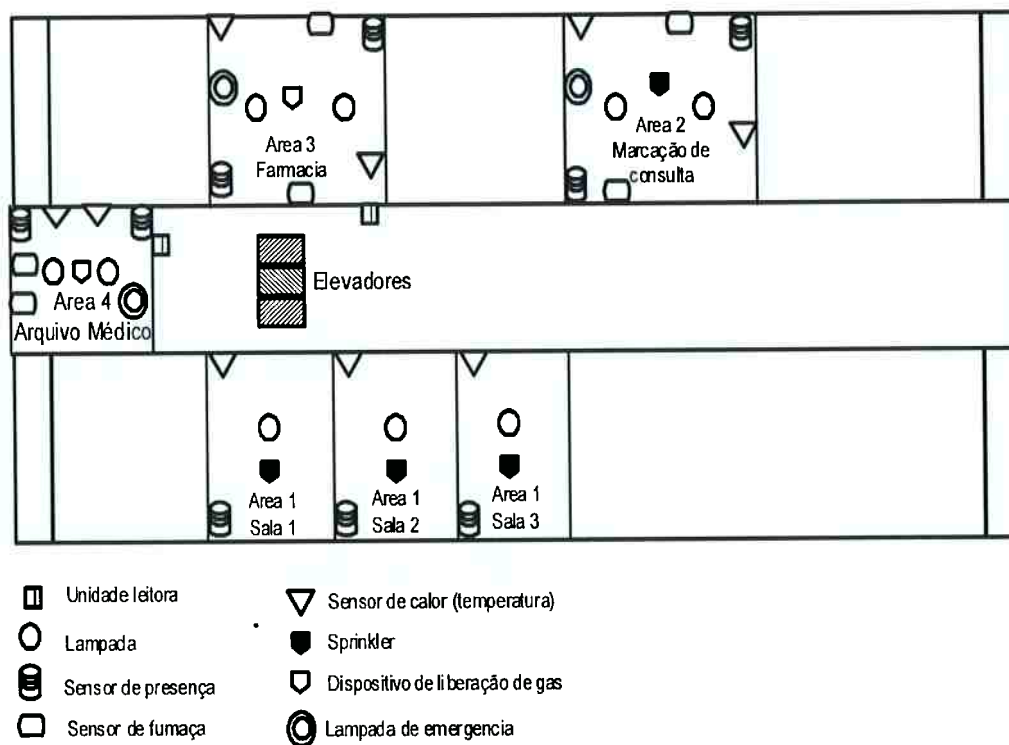


Fig. 5.64 Disposição das áreas consideradas dentro do quarto pavimento

Tabela 5.4 dispositivos pertencentes aos diversos sistemas prediais

ÁREA	DISPOSITIVOS
1 (3 salas de consulta médica de pacientes conveniados)	3 sensores de presença (1 por sala) 3 lâmpadas (1 por sala) 3 sensores de calor (1 por sala) 3 sensores de fumaça (1 por sala) 3 <i>sprinklers</i> (1 por sala) 1 ventilador de retorno 1 ventilador de insuflamento 1 serpentina 1 caixa de mistura de 3 posições: sem renovação (0%), renovação parcial (40%), renovação total (100%)
2 (1 sala de marcação de consulta)	2 sensores de presença 2 lâmpadas 1 lâmpada de iluminação de emergência 1 sensores de calor 1 sensores de fumaça 1 <i>sprinkler</i> 1 ventilador de retorno 1 ventilador de insuflamento 1 serpentina 1 caixa de mistura de 3 posições: sem renovação (0%), renovação parcial (40%), renovação total (100%)
3 (1 sala de farmácia)	2 sensores de presença 1 leitora de identificação 2 lâmpadas 1 lâmpada de iluminação de emergência 2 sensores de calor 2 sensores de fumaça –iônicos- 1 dispositivo de liberação de gas 1 ventilador de retorno 1 ventilador de insuflamento 1 serpentina 1 caixa de mistura de 3 posições: sem renovação (0%), renovação parcial (40%), renovação total (100%)
4 (1 sala de arquivo médico)	2 sensores de presença 1 leitora de identificação 2 lâmpadas 1 lâmpada de iluminação de emergência 2 sensores de calor 2 sensores de fumaça –fotoelétricos- (2 por sala) 1 dispositivo de liberação de gas 1 ventilador de retorno 1 ventilador de insuflamento 1 serpentina 1 caixa de mistura de 3 posições: sem renovação (0%), renovação parcial (40%), renovação total (100%)

Como pode ser apreciado na Tabela 5.4 todas as áreas contam com dispositivos dos subsistemas de controle de acesso, iluminação, detecção e combate a incêndio e ar condicionado. No entanto e pelas características do controle de acesso e do controle de incêndio que envolvem as áreas 3 e 4 adicionou-se em cada uma destas áreas uma leitora de identificação e substituiu-se os *sprinklers* por dispositivo de liberação de

gas para evitar danificar ainda mais os medicamentos e arquivos. O sistema de elevadores disponível para uso de todas as salas é um grupo de 3 elevadores que realizam o transporte entre os diferentes andares do PAMB.

Para realizar a simulação consideraram-se o número de instâncias de cada classe igual ao número presente de objetos dessa classe em cada área. A identificação dos objetos identificação foi realizada segundo a nomenclatura dada para cada classe nos modelos explicitando a área à qual pertence e uma numeração interna à área, por exemplo: SFarls2 (sensor de fumaça da área 1 e da sala 2).

Uma vez construído o modelo, a simulação mostra a evolução do sistema sob certas condições. O objetivo é garantir para estas condições um comportamento desejado. Como o objetivo deste trabalho é ver a integração dos diversos sistemas prediais e um dos sistemas que maior integração tem com os outros é o sistema de incêndio, considerou-se: A situação de incêndio na área 3

O primeiro passo para a simulação é definir o estado inicial do sistema, isto é, a marcação inicial dos modelos. Assim, define-se: que todos os sistemas encontram-se ligados em condições normais assim, são detectadas pessoas em todas as áreas, as lâmpadas encontram-se ligadas, os ventiladores de retorno e de insuflamento encontram-se em velocidades normais, a caixa de mistura se encontra na posição de renovação 40% e as áreas estão sendo resfriadas. Em seguida foi modificado o estado do sensor de calor para a área 4. Isto deveria ativar as estratégias de incêndio em todos os sistemas prediais da área 4. Para esta área, deveria assim ser ativado o dispositivo de liberação de gás e desligado o resfriamento, colocou-se o ventilador de retorno em velocidade alta e a caixa de mistura posiciona-se em renovação 100%, ligou-se a iluminação de emergência, liberou-se a trava da porta. Nas áreas 1,2,3, desligou-se o resfriamento, os ventiladores de insuflamento são colocados em velocidade alta e a caixa de mistura posicionou-se em renovação 100%.

Os resultados obtidos a partir da simulação contribue para a validação da proposta apresentada, uma vez que o sistema apresentou o comportamento esperado diante da ocorrência de eventos predeterminados onde foi observada a efetiva integração entre os diversos subsistemas prediais presentes.

Verificação

A verificação dos modelos foi realizada gerando o grafo de ocorrência. Para isto também foi usado o software *CPN Tools* embora a versão usada não conta com a representação gráfica do grafo.

A análise considera cada módulo e suas interações. No módulo considerado analisaram-se as propriedades de:

Segurança entendida como a garantia que o sistema não chega a atingir estados que corrompam seu comportamento ou que lhe impeça de continuar suas interações com o seu ambiente. Violações da propriedade de segurança conduzem a estados indesejados para o sistema.

Confiabilidade dada na rede de Petri a través da propriedade de alcançabilidade que assegura que em determinadas circunstâncias, um estado sempre é atingido garantindo assim as funcionalidades do sistema.

Estas propriedades verificadas podem ser expressas nos módulos que compõem os sistemas sob a forma de restrições locais e de hipóteses sobre seu ambiente.

No caso deste trabalho e do sistema apresentado, considerou-se a verificação do sistema de detecção e combate a incêndio através do módulo: controlador de incêndio quando tem sido detectado um incêndio através dos sensores.

O enunciado de uma propriedade de alcançabilidade considerada foi: A detecção de incêndio conduz à produção de uma alarma. A restrição considera que a alarma deve estar em condições de funcionamento normais. A hipótese considera que a chamada do método da alarma é imediatamente executada. A análise do grafo de ocorrência destes módulos mostra que sendo que a restrição se cumpre se verifica segundo os estados alcançados que quando foi detectado um incêndio a alarme foi efetivamente ativada.

O enunciado de uma propriedade de segurança considerado foi: o ventilador de insuflamento nunca pode ser ligado em caso de incêndio. A restrição considera que o ventilador deve estar em condições de funcionamento normais. Não existem hipóteses. A análise do grafo de ocorrência mostra que existe um caminho no grafo de ocorrência no qual só os ventiladores de retorno são ligados.

5.8. CONSIDERAÇÕES DO CAPÍTULO

Neste capítulo foi apresentado um exemplo de aplicação da abordagem proposta, para modelar e analisar um sistema distribuído e aberto para automação predial. Para este exemplo consideraram-se alguns dos principais sistemas prediais que possui o prédio dos ambulatórios (PAMB) do Hospital das Clínicas da Faculdade de Medicina da Universidade de São Paulo (HC-FMUSP).

A aplicação desta abordagem permitiu inicialmente definir as funcionalidades que são desejadas para os de sistemas prediais no PAMB e seus requisitos. Seguidamente foram identificados seus relacionamentos e se realizou a modelagem dinâmica de cada uma das classes que compõem cada sistema. Finalmente se realizou a análise dos modelos gerados através do software de simulação. Esta análise foi realizada de maneira incremental considerando as redes de cada módulo e suas interações o que permitiu validar e verificar os modelos e sua integração através da consideração de suas interações.

6. CONCLUSÃO

No presente trabalho estudou-se a utilização dos conceitos de sistemas distribuídos e abertos para automação predial visando a integração entre os sistemas prediais de um edifício inteligente. Assim, foi utilizado o padrão de processamento distribuído e aberto – ODP, o qual define um *framework* através dos seus pontos de vista para modelar e especificar esta classe de sistemas. Neste sentido, os pontos de vista abordados: empresa, informação e computação, orientaram o tratamento da complexidade que os edifícios inteligentes apresentam e permitiram de forma abrangente descrever as interações de seus subsistemas.

O ODP não define nenhuma linguagem a ser usada ao tratar os seus diferentes pontos de vista; no entanto, por estar baseado na orientação a objetos, esta linguagem deve ser baseada neste paradigma. Assim, nesta pesquisa, a linguagem usada com este padrão foi a UML. Contudo, a orientação a objetos não conta com um método formal totalmente aceito e abrangente para analisar os modelos desenvolvidos deste modo, procurou-se neste trabalho realizar esta tarefa através do uso das redes de Petri.

Através da UML realizou-se a modelagem estruturada de diversos aspectos do sistema que permitem garantir que os requisitos dos sistemas sejam atendidos. Por sua parte, as redes de Petri estabelecem uma ligação entre os modelos e são fundamentais para representar a dinâmica do sistema.

Neste contexto, este trabalho propõe uma abordagem para a modelagem e análise de sistemas distribuídos e abertos para automação predial. Assim, inicialmente foi proposto um formalismo de modelagem e análise que considera as características dos sistemas distribuídos e abertos e os sistemas prediais. Aqui foi usada a rede de Petri colorida como formalismo base. A proposta baseou esta rede

nos conceitos de orientação a objetos, com a finalidade de lhe outorgar uma estrutura modular que permita a modelagem de sistemas complexos, e nos conceitos do ODP. Este formalismo utiliza principalmente as informações ligadas aos modelos estrutural e dinâmico.

Seguidamente apresentou-se um procedimento para modelar e analisar sistemas distribuídos e abertos para automação predial baseado no formalismo apresentado. Aqui são consideradas as características dos sistemas prediais e utilizam-se os diagramas da UML para definir as funcionalidades dos sistemas e estruturar o processo. As redes de Petri são utilizadas visando representar os relacionamentos entre os sistemas e realizar a modelagem dinâmica das classes e objetos identificados e a sua análise.

Finalmente avaliou-se a abordagem proposta a través de sua aplicação em um prédio hospitalar.

Em geral pode-se ressaltar que:

A abordagem apresentada se beneficia tanto da notação gráfica como do uso do padrão ODP, pois o uso de diferentes pontos de vista permite um melhor controle do processo de modelagem.

Da aplicação da abordagem proposta verificou-se a efetividade de se utilizar o UML em conjunto com as redes de Petri como ferramentas para a modelagem e análise dos pontos de vista da ODP e especificamente conseguiu-se representar e avaliar a integração entre os sistemas prediais de um hospital considerado no estudo de caso.

Em geral, a abordagem proposta procura sistematizar o projeto de um sistema predial integrado de maneira a evitar custos adicionais que podem advir de um sistema que considere apenas os recursos tecnológicos sem um procedimento de modelagem e análise da proposta.

Trabalhos Futuros

Considerando que o desenvolvimento das pesquisas científicas é uma tarefa contínua e que evolui em diferentes direções, sugerem-se a partir da presente investigação os seguintes trabalhos:

- Continuar o estudo da abordagem apresentada para os pontos de vista engenharia e tecnologia visando obter uma metodologia para o projeto e implementação de sistemas distribuídos e abertos para automação predial.
- Desenvolvimento de um aplicativo que auxilie na modelagem e na análise de redes de Petri modulares coloridas baseadas em objetos – RPMCO.
- Um ponto que se considera especialmente importante a ser estudado e aprimorado é a aplicação de outros métodos de análise como, por exemplo, através da aplicação da lógica temporal.
- Aplicação dos procedimentos propostos a outros estudos de caso e a outras aplicações distribuídas;

ANEXO A

A seguir é apresentado um compêndio sobre as tecnologias de comunicação atualmente disponíveis para edifícios inteligentes.

A.1 COMUNICAÇÕES EM EDIFÍCIOS INTELIGENTES

Um edifício inteligente pode ser considerado como o resultante da junção do sistema de automação do edifício (BAS – *Building Automation System*) e do sistema integrado de comunicações (ICS - *Integrated Communication Systems*). O primeiro é o encarregado de estabelecer a interação dos sistemas de controle de acesso, de iluminação, de ar condicionado entre outros. O segundo abrange os serviços relacionados com a transmissão de dados, voz e imagens (Sánchez et al., 1998)

Estes dois sistemas precisam de um suporte adequado para realizar o intercâmbio de informação, seja do usuário ou do controle, que conecte estes sistemas e permita que interajam entre si, transportando os sinais relacionados com os diferentes sistemas de voz, dados, vídeo e controle presentes no edifício. Em geral, este sistema deve ser modular e o mais flexível possível para permitir a instalação de uma ampla gama de serviços e, no caso de ser necessária, a incorporação de novos serviços ou seu redimensionamento (ao ocorrer mudanças na distribuição física de ambientes, pessoas e equipamentos).

No sistema integrado de comunicações são necessárias tecnologias que sejam capazes de transmitir texto, dados, voz e imagens através de um meio físico comum. Neste contexto os mais importantes avanços têm sido derivados das redes de comunicação de dados entre as quais estão: as redes de dados locais (LAN – *Local Area Network*), as redes de área metropolitana (MAN – *Metropolitan Area Network*) e as redes digitais de serviços integrados (ISDN – *Integrated Services Digital*

Network). Destas redes uma das que consegue satisfazer todas as necessidades referentes à transmissão de dados é a ISDN que se apoia no modo de transferência assíncrono (ATM – *Asynchronous Transfer Mode*).

Por outro lado, o sistema de automação do edifício incorpora sua própria estrutura de transmissão de sinais (cabeamento) e centrais de supervisão e controle envolvendo equipamentos como o PLC (*Programmable Logic Control*) e estruturas como o DDC (*Direct Digital Control*), que são os encarregados de processar as informações provenientes dos sensores e atuar em correspondência a estas. O sistema de automação do edifício se estende através de todas as suas instalações e é composto por diferentes sistemas que possuem um cabeamento e controle dedicados a tarefas específicas o que dificulta a sua integração. Assim, para um edifício funcionar como um sistema integrado, é necessária uma rede que de fato integre a transmissão de sinais (Callaghan et al., 1999). Neste contexto, cada vez mais tem-se evidenciado a necessidade de adotar um protocolo padrão, que permita que dispositivos de diferentes fabricantes possam comunicar-se uns com outros em uma linguagem comum. A seguir é apresentada uma sinopse das tendências dos diferentes padrões existentes para automação predial.

X-10: é o mais antigo e o mais comum entre os padrões disponíveis para a transmissão de informações em edifícios e sistemas de automação residencial. Foi desenvolvido em 1970 com seus primeiros produtos disponíveis em 1979. Permite um limitado controle de dispositivos domésticos através da fiação elétrica existente em uma edificação. O protocolo X-10 transmite dados modulados na frequência da tensão (AC) da rede elétrica. Este protocolo permite endereçar 256 pontos diferentes. Os atuadores e sensores recebem/emitem sinais unidirecionais não possuindo nenhum processamento local destes. Uma desvantagem deste protocolo é que seu funcionamento é afetado pela rede elétrica que pode apresentar alguns comportamentos erráticos devido a ocorrências como duplicidade de fase, falta de energia ou descargas eletromagnéticas, o que limita sua confiabilidade. Por isto, o protocolo X-10 não é recomendável para aplicações críticas, ligadas a segurança. Outras desvantagens envolvem sua limitação a pequenas e médias distâncias e sua baixa integração com outros sistemas prediais que utilizam cabeamentos dedicados,

tais como vídeo e áudio (Saramago, 2002); (Callaghan et al., 1999); (Sanchez et al., 1998).

Bati-Bus: tem sido desenvolvido para interconectar dispositivos de baixo nível de complexidade no interior de um edifício. Além da transmissão de dados pode ser usado para a distribuição de uma pequena quantidade de energia elétrica. Usa o modo de transmissão assíncrono podendo ter 240 pontos direcionáveis e chegar até 1000 pontos por rede. Opera em forma distribuída (não precisa de um nó *master*) e sua velocidade de transmissão é de 4800 bps (Sanchez et al., 1998).

CAB (*Canadian Automated Building Protocol*): tem sido desenvolvido pela *Public Works* do Canada como um conjunto de serviços projetados com o objetivo de supervisionar a operação e manutenção dos sistemas de um edifício. Algumas das intervenções em tempo real que realiza são: análise da distribuição de energia, diagnóstico estimado dos equipamentos do edifício e a comunicação de estados ao sistema de administração do edifício (Sanchez et al., 1998).

Neste protocolo os dados são divididos em duas categorias: dados gerais e dados de funções específicas. Os dados gerais incluem pontos tradicionais como entradas/saídas digitais e entradas/saídas analógicas. Os dados de funções específicas estão relacionados com as operações comuns de controle do edifício tais como: controle de acesso, eventos e alarmes, etc.

Data Highway Plus: é uma rede aberta de controladores da empresa Rockwell/Allen-Bradley que permite compartilhar dados no nível de planta e de equipamento com o programa de manutenção. É disponível em vários meios e velocidades de comunicação. Suas interfaces podem ser implementadas usando adaptadores de comunicações assíncronas (Callaghan et al., 1999).

FND (*Firm-Neutral Data Transmission*): foi criado na Alemanha. O FND permite a conexão dos DDC (*Direct Digital Control*) com uma estação central de controle de modo que mesmo os sistemas com diferentes fornecedores (marcas) possam comunicar-se e interoperar em uma rede com conexão ponto a ponto (Saramago, 2002).

Modbus: originalmente foi projetado para a comunicação entre controladores programáveis, no entanto pode ser usado para integrar computadores, terminais e outros dispositivos de monitoração e controle. Este é um protocolo tipo *master/slave*

que suporta modos de transmissão ASCII (*American Standard Code for Information Interchange*) e binário. O número máximo de escravos que pode comportar é 247 (Saramago, 2002).

N2: é usado para conectar unidades terminais de controle a unidades supervisoras. Está baseado em um protocolo ASCII amplamente usado na indústria. Emprega como meio de transmissão três cabos RS-485 conectando um máximo de 100 dispositivos a 9600 bps, e cobrindo uma distância de 100 m. Este é um protocolo tipo *master/slave* que proporciona a leitura e escrita nos seus dispositivos de entrada/saídas analógicos e digitais (Saramago, 2002).

LONWorks: é um padrão desenvolvido pela empresa Echelon para automação predial e industrial. É um sistema aberto e de controle distribuído, projetado para implementar a interoperabilidade de redes de controle. Este padrão é considerado como um dos mais completos em termos de recursos e utilização. A tecnologia LONWork inclui a infra-estrutura em *hardware* e *software* para a operação da rede local LON (*Local Operating Network*). O protocolo utilizado é denominado LONTalk (padrão ANSI -*American National Standards Institute*- 709.1) que implementa as sete camadas do modelo de referência da ISO (*International Organization for Standardization*) e foi projetado para aplicações que envolvem funções de sensoriamento, controle, monitoração e identificação. O esquema de acesso ao meio físico se faz por CSMA p-persistente (*Carrier Sense Multiple Access*). Permite seis meios físicos de transmissão de sinais: cabos elétricos, cabo trançado de baixa voltagem, cabo coaxial, infravermelho, rádio frequência e fibra óptica. A parte principal de uma rede LONWorks é um chip, de nome Neuron chip, o qual atua como um nó da rede e inclui o hardware de comunicações e o protocolo. O controle da rede é distribuído nos dispositivos (nós). Cada nó tem seu próprio programa aplicativo, de acordo com a sua função e se comunica com os outros nós através do protocolo LONTalk. A associação LONMark estabelece padrões e certifica dispositivos de controle interoperáveis (Saramago, 2002); (Callaghan et al., 1999); (Sanchez et al., 1998).

BACnet (*Building Automation and Control Network*): é o padrão desenvolvido pela ASHRAE (*American Society of Heating, Refrigerating and Air Conditioning Engineers*) e adotado pela ANSI (ANSI/ASHRAE 135-1995). É um

protocolo aberto de comunicação de dados. Está baseado nas camadas: de aplicação, de rede, de enlace de dados e na camada física do modelo de referência OSI. BACnet especifica quatro tipos de redes LAN: Ethernet, ARCNET, MS/TP (Master-Slave/Token-Passing), LONTalk (Saramago, 2002); (Callaghan et al., 1999); (Sanchez et al., 1998).

EIB (*European Installation Bus*): originalmente foi desenvolvido por um grupo de empresas lideradas pela Siemens, mas hoje é controlado pela EIBA (*European Installation Bus Association*). Este sistema é usualmente implementado em arquitetura descentralizada com os dispositivos de campo comunicando-se diretamente entre si e sem nenhum nível supervisor. No entanto também é possível implementar instalações centralizadas. O número máximo de dispositivos que pode comportar é 61455 distribuídos em linhas dentro de áreas. Alguns dos meios físicos que podem ser usados para o barramento são: par trançado, fiação elétrica existente (*powerline*), rádio frequência e redes locais (LAN) existentes, tais como Internet e Intranet. Os dispositivos componentes do sistema EIB constam de uma unidade de acoplamento ao barramento (BCU- *Bus Coupling Unit*) que serve de interface e que é composta por um *transceiver* e um controlador de comunicação (Saramago, 2002).

PROFIBus (*PROcess FieId Bus*): originalmente foi desenvolvido na Alemanha mas em 1997 foi ratificado como parte da norma europeia. É atualmente o tipo de *FieldBus* mais popular que existe. Existem três variantes do mesmo (Saramago, 2002).

- PROFIBus – FMS (*FieldBus Message Specification*): é uma solução universal para as tarefas de comunicação nos níveis superiores e para a automação. É empregado para a comunicação entre dispositivos de área e controladores e para o intercâmbio de informação entre os controladores.
- PROFIBus – DP (*Distributed Peripherals*): Está dedicado à comunicação entre sistemas de automação e periféricos distribuídos.
- PROFIBus – PA (*Process Automation*): conecta os sistemas de automação e os dispositivos de forma descentralizada.

Pode ser implementado em microprocessadores que contenham uma UART (*Universal Asynchronous Reception Transmition*) interna ou externa.

HomePnP: padroniza a estrutura das mensagens e os códigos de controle usados nas mensagens fornecendo assim, todos os detalhes necessários para construir uma mensagem para efetuar uma operação específica em um dispositivo ou subsistema. Este protocolo é baseado na CAL (*Common Application Language*) e no nível de aplicação do protocolo CEBus (*Consumer Electronics Bus*) (CIC, 1998).

Jini: foi desenvolvido pela empresa Sun Microsystems e posto no mercado em 1999. Sua arquitetura é totalmente distribuída, de maneira que todos os dispositivos conectados em rede podem oferecer serviços e comunicar-se entre si. Jini estende os conceitos da linguagem de programação Java de uma máquina simples para uma rede de máquinas, podendo funcionar em estações de trabalho, microcomputador, e até em eletrodomésticos que estejam conectados em uma rede. É possível também a conexão de dispositivos que não executem a máquina virtual Java através do uso da arquitetura *surrogate* (Sun, 2001).

UPnP (*Universal Plug & Play*): é uma arquitetura aberta que permite a troca de informações e dados de forma transparente entre os dispositivos conectados na rede, sem necessidade de configurar estes. Nela são utilizados protocolos padrões tais como: TCP, UDP, IP entre outros. O seu projeto buscou a independência de fabricantes, da linguagem de programação de cada dispositivo e do meio físico para implementar a rede. A descrição dos serviços e dispositivos é mantida pelo UPnP Forum - *Universal Plug and Play Forum*. Os elementos básicos do UPnP são: os dispositivos (*devices*), os serviços (*services*) e os controladores (*control points*), cujas informações são estruturadas utilizando XML (*eXtensible Markup Language*). A *UPnP Device Architecture* (Microsoft, 2000) define o modelo genérico para a criação de dispositivos e para descrição de serviços para qualquer dispositivo e tipo de serviço.

OSGI (*Open Services Gateway Initiative*): foi criada em 1999 com a finalidade de projetar especificações abertas para viabilizar múltiplos serviços sobre redes geograficamente distribuídas (WANs), para redes locais e dispositivos. Os serviços *gateway* são os componentes principais da especificação OSGI. Estes são servidores inseridos em uma rede para conectar a rede externa com clientes internos, atuando como uma plataforma para instalação e execução do software que trabalha em cooperação com dispositivos em uma rede doméstica. Este software permite que

provedores de serviços externos interajam com os dispositivos dessa rede, disponibilizando os serviços para seus proprietários (Saramago, 2002).

ANEXO B

Este anexo apresenta os conceitos básicos de redes de Petri

B.1 REDES DE PETRI (RP)

O conceito de RP foi introduzido por Carl Adam Petri em 1962 na sua tese de doutorado, como ferramenta para descrever relações entre condições e eventos no estudo de protocolos de comunicação entre componentes assíncronos. Embora ocorresse uma considerável divulgação acadêmica ao longo das três décadas seguintes, o seu potencial só foi reconhecido na década de oitenta, onde esta teoria foi usada para implementações práticas nas áreas de informática e manufatura devido à disponibilidade de novos recursos de *hardware* e *software*.

A RP é uma ferramenta matemática e gráfica que oferece um ambiente uniforme para modelagem, análise e projeto de SED (Zurawki; Zhou, 1994). Sua aplicação tem-se estendido a uma grande quantidade e variedade de sistemas. Os principais sistemas onde é aplicada esta técnica são: sistemas de comunicações, sistemas de software, sistemas de processamento de informação, além das aplicações em modelagem, simulação e seqüenciamento de sistemas de manufatura. Mais recentemente têm sido aplicada também nos sistemas prediais (Villani, 2000); (Bastidas, 1999); (Gomes, 1997). Em geral, utilizam-se as RP e suas extensões para a modelagem e análise da estrutura do sistema e de seu comportamento dinâmico, assim como uma especificação de estratégias de controle de sistemas.

B.1.1 CARACTERÍSTICAS E VANTAGENS DAS REDES DE PETRI

Em Hasegawa (1996) destacam-se as seguintes características e vantagens desta técnica:

- representa a dinâmica e a estrutura do sistema segundo o nível de detalhamento desejado.
- identifica estados e ações de modo claro e explícito, facilitando com isto a monitoração do sistema em tempo real.
- tem a capacidade para representar de forma natural as características dos SED, isto é, a sincronização, assincronismo, concorrência, causalidade e conflito entre processos, compartilhamento de recursos, etc.
- associa elementos de diferentes significados em uma mesma representação, ou segundo o propósito do modelo que pode ser para avaliação de desempenho, implementação do controle, etc.
- oferece um formalismo que permite a documentação e monitoração do sistema, facilitando assim o diálogo entre o projetista e as pessoas que participam no processo de projeto ou de análise do comportamento do sistema, isto é, entre o projetista, operador, gerente, etc.
- se constitui como uma teoria muito bem fundamentada para a verificação de propriedades qualitativas.
- possui uma semântica formal e precisa que permite que o mesmo modelo possa ser utilizado tanto para a análise de propriedades comportamentais (análise quantitativa e/ou qualitativa) e avaliação do desempenho, assim como para a construção de simuladores a eventos discretos e controladores. Além de servir para verificar comportamentos indesejáveis do sistema como bloqueio de processos, limitação de recursos, etc.
- incorpora conceitos de modelagem do tipo refinamento (*top down*) e do tipo composição modular (*bottom up*) através de técnicas como: modularização, reutilização e refinamento.

Como uma ferramenta matemática, um modelo em RP pode ser descrito por um sistema de equações lineares, ou outros modelos matemáticos que refletem o comportamento do sistema o qual possibilita a análise formal do mesmo (Zurawski; Zhou, 1994). Esta característica permite realizar a verificação formal das propriedades comportamentais do sistema.

B.1.2 ELEMENTOS DA REDE DE PETRI E SUA REPRESENTAÇÃO GRÁFICA

A representação gráfica de uma RP consiste de elementos conectados por segmentos orientados (Peterson, 1981); (Murata, 1989).

Os elementos estruturais das Redes de Petri são (vide Figura 3.1):

- * **TRANSIÇÕES (T):** correspondem a eventos que causam a mudança de estado do sistema. Podem ser temporizadas e graficamente são representadas por barras.
- * **LUGARES (P):** representam condições (pré e pós-condições) relacionadas com a ocorrência de eventos que podem estar associadas ao modo de operação ou à disponibilidade de um recurso no sistema. Sua representação gráfica é através de círculos.
- * **MARCAS:** cuja função é representar a manutenção de uma determinada condição. Graficamente estas marcas são modeladas por pontos pretos no interior dos lugares. A existência de uma MARCA em um LUGAR equivale ao valor binário “1” e é “0” quando não existe MARCA no LUGAR.
- * **ARCOS ORIENTADOS:** estabelecem relações causais entre os eventos e as condições e vice-versa. São representados por setas que conectam de forma alternada LUGARES e TRANSIÇÕES.
- * **PORTAS:** que habilitam ou inibem a ocorrência dos eventos correspondentes às transições, sendo denominadas habilitadoras ou inibidoras, conforme sua natureza. Para a PORTA habilitadora, quando o sinal de origem for equivalente ao valor binário “1” esta PORTA habilita a TRANSIÇÃO à qual está conectada. Para a PORTA inibidora, quando o sinal de origem for equivalente ao valor binário “1”, esta PORTA inibe a TRANSIÇÃO à que está conectada.

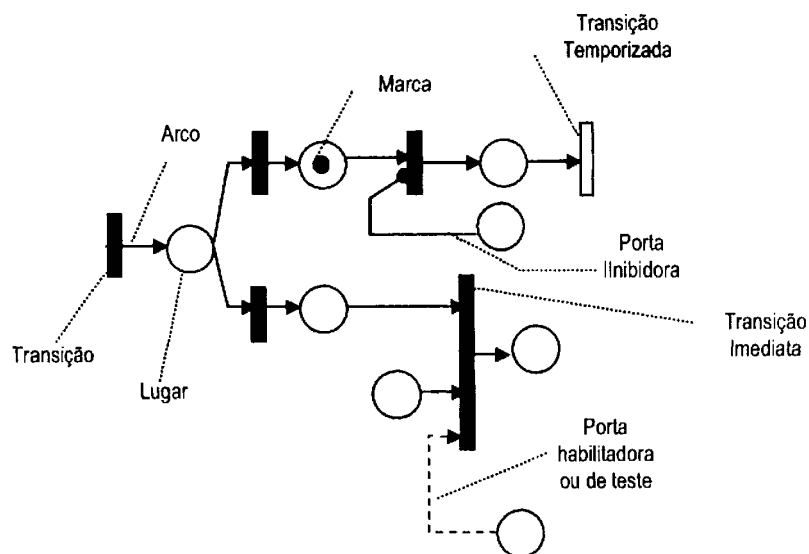


Fig. B.1 Exemplo de um modelo em redes de Petri com a identificação de seus elementos

Uma RP é um multigrafo já que permite múltiplos arcos de um elemento do grafo até outro. É bipartido, pois os elementos do grafo são particionados em dois conjuntos: LUGARES e TRANSIÇÕES; e os ARCOS conectam elementos de diferentes conjuntos. É direcionado, porque os arcos são orientados. O número de LUGARES e TRANSIÇÕES é finito e não nulo.

B.1.3 ESTRUTURA DE UMA REDE DE PETRI

A estrutura de uma Rede de Petri pode ser definida formalmente como uma quádrupla (P, T, I, O) onde:

- $P = \{p_1, p_2, \dots, p_m\}$ é o conjunto de LUGARES da rede, em que $m \in \mathbb{N}$ é o número total de LUGARES da rede.
- $T = \{t_1, t_2, \dots, t_n\}$ é o conjunto de TRANSIÇÕES, em que $n \in \mathbb{N}$ é o número de TRANSIÇÕES da rede.
- $I : (P, T)$ é a função que define os ARCOS de entrada em relação às TRANSIÇÕES.

Se $I(p_j, t_i) = k$, $k \in \mathbb{N}$ existem k ARCOS orientados do LUGAR p_j para a TRANSIÇÃO t_i .

Se $I(p_j, t_i) = 0$, não existe ARCO orientado do LUGAR p_j para a TRANSIÇÃO t_i .

- $O : (T,P)$ é a função que define os ARCOS de saída em relação às TRANSIÇÕES.

Na representação gráfica a existência de múltiplos ARCOS conectando LUGARES e TRANSIÇÕES, pode ser indicada de forma compacta por um único ARCO rotulado com seu peso ou multiplicidade k (vide Figura 3.2)

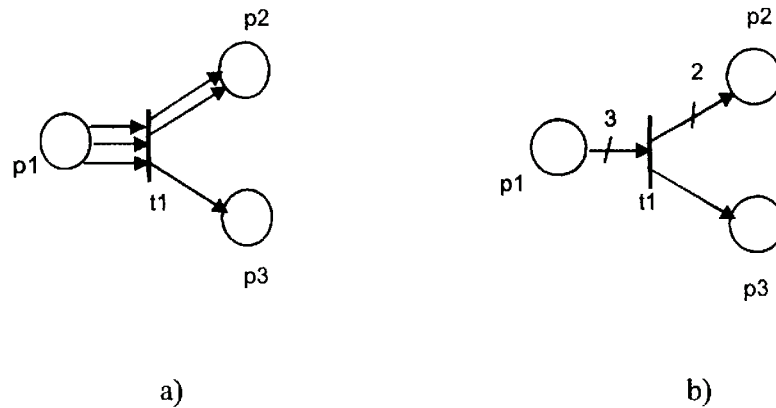


Fig. B.2 a) Arcos múltiplos. b) Representação compacta.

B.1.4 MARCAÇÃO DE UMA REDE DE PETRI E SUA EVOLUÇÃO

Uma RP com MARCAS em seus LUGARES é chamada de Rede de Petri Marcada $N = (P,T,I,O,M_0)$, em que M_0 é a MARCAÇÃO inicial. Uma distribuição particular das MARCAS nos LUGARES da rede representa um estado do sistema.

$M = (M(p_1), M(p_2), \dots, M(p_m))$ é a MARCAÇÃO em que $m \in \mathbb{N}$ é o número total de LUGARES. $M(p_i)$ é o número de MARCAS do LUGAR p_i ($M(p_i) \in \mathbb{N}$).

O disparo de uma TRANSIÇÃO na RP corresponde a ocorrência de um evento que altera o estado do sistema, isto é, altera a MARCAÇÃO atual (M_i) para uma nova MARCAÇÃO (M_{i+1}). Uma TRANSIÇÃO ($t_i \in T$) está habilitada para disparar se e somente se cada LUGAR de entrada p_j (tal que $I(p_j, t_i) > 0$) contém pelo menos um número de MARCAS igual ao peso do ARCO (p_j, t_i), ou seja: $M(p_j) \geq I(p_j, t_i)$ para qualquer LUGAR $p_j \in P$.

O disparo de uma TRANSIÇÃO t_i habilitada remove de cada LUGAR de entrada p_j (pré-condição) da TRANSIÇÃO t_i um número de MARCAS igual ao peso do ARCO $I(p_j, t_i)$ que conecta p_j a t_i . Depositando, também, em cada LUGAR de saída p_q (pós-condição) da TRANSIÇÃO t_i um número de MARCAS igual ao peso do ARCO $O(t_i, p_q)$ que conecta t_i

a p_q . A Figura 3.3 ilustra um exemplo da alteração da MARCAÇÃO de uma rede após o disparo de uma TRANSIÇÃO.

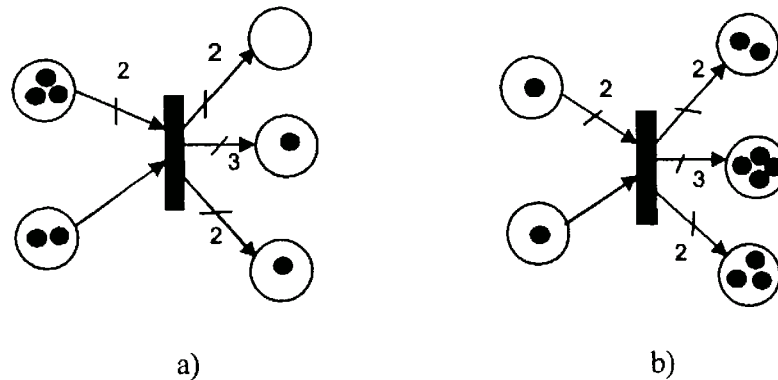


Fig. B.3 a) TRANSIÇÃO habilitada. b) MARCAÇÃO após o disparo.

A evolução da MARCAÇÃO (de acordo com regras de disparo de TRANSIÇÕES) simula o comportamento dinâmico do sistema.

Um exemplo da representação gráfica de uma RP é dado pela Figura 3.4, através do modelo simples de um sistema computacional. Na MARCAÇÃO inicial, pode-se observar que uma tarefa encontra-se em espera, e como o processador está disponível, a TRANSIÇÃO “início do processamento da tarefa” está habilitada. O comportamento dinâmico do sistema é observado ao aplicar-se a regra de disparo.

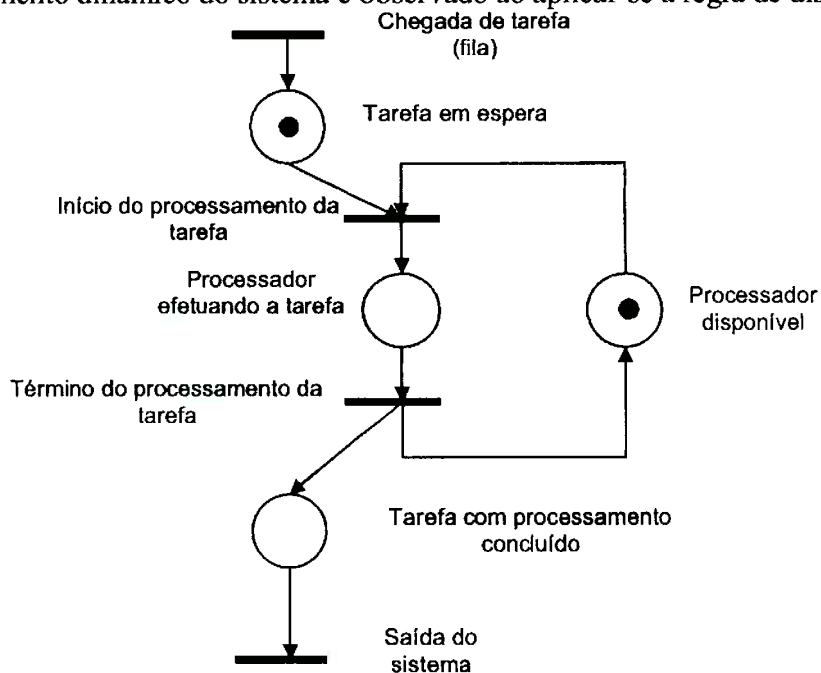


Fig. B.4 Modelo em RP de um sistema computacional

B.1.5 CLASSIFICAÇÃO DAS REDES DE PETRI

As abordagens relacionadas com Redes de Petri encontradas na literatura podem ser organizadas em três classes fundamentais (Santos, 1998):

- **Rede de Petri básica ou ordinária:** se constitui em um modelo elementar adequado para visualizar comportamentos que envolvem paralelismo, sincronização de processos e compartilhamento de recursos. Este é o modelo básico de rede que permite extensões segundo a aplicação para a qual se utiliza. São as RP apresentadas nos itens anteriores.
- **Reduções de Rede de Petri ordinárias:** são descrições simplificadas que procuram sintetizar a apresentação gráfica dos modelos, mas que ainda podem ser representadas adequadamente por Redes de Petri ordinárias. Entre estas estão: Redes de Petri generalizadas, Redes de Petri de capacidade finita, Redes de Petri coloridas, Redes de Petri orientadas a objetos.
- **Extensões das Redes de Petri ordinárias:** correspondem a modelos que incorporam regras adicionais de funcionalidade para enriquecer o poder da modelagem. São consideradas três subclasses (David; Alla, 1994): modelos equivalentes a máquinas de Turing, modelos para sistemas contínuos e híbridos e modelos de sistemas que evoluem em função da ocorrência de eventos externos ou do tempo.

B.1.6. PROPRIEDADES DAS REDES DE PETRI

As Redes de Petri possuem propriedades que permitem a análise detalhada da estrutura e do desempenho do sistema modelado. Dois tipos de propriedades podem ser distinguidas nas redes de Petri:

A. PROPRIEDADES PROCEDURAIS OU DINÂMICAS

São as que dependem da MARCAÇÃO inicial e estão ligadas à evolução da rede (são chamadas também de “boas” propriedades), isto é, dos estados do sistema. Sua verificação se faz geralmente pela construção do grafo de MARCAÇÕES acessíveis, isto é, estados alcançáveis do sistema.

As principais propriedades procedurais ou dinâmicas das redes de Petri são:

Alcançabilidade: refere-se à possibilidade de um sistema atingir um determinado estado (ou conjunto de estados). Uma MARCAÇÃO M_i é *alcançável* a partir de M_0 (marcação inicial) se existir uma seqüência de disparos de TRANSIÇÕES que leve uma rede com MARCAÇÃO M_0 a uma outra com MARCAÇÃO M_i . O conjunto de alcançabilidade R , é o conjunto de todas as MARCAÇÕES alcançáveis a partir de M_0 através de seqüências válidas de disparo das TRANSIÇÕES

Limitação (*boundedness*): uma rede de Petri é *limitada* se e somente se o número de MARCAS de todos os LUGARES da rede não superam um limite definido B , isto é, $M(p_i) \leq B$ para qualquer marcação *alcançável* ($B \in \mathbb{N}$).

Uma rede marcada N é k -limitada se e somente se todos os seus LUGARES são k -limitados.

Segurança (*safety*): esta propriedade é uma caso especial da propriedade anterior. Neste caso, uma rede é dita ‘segura’ quando ela é 1-limitado, isto é o número de MARCAS contidas em qualquer LUGAR é sempre 0 ou 1. Uma rede marcada N é ‘segura’ se e somente se todos os seus lugares são ‘seguros’. Esta rede também é denominada de binária.

Reversibilidade (*reversibility*): uma Rede de Petri é dita reversível, se a partir de uma MARCAÇÃO qualquer (estado qualquer) ela pode retornar à MARCAÇÃO inicial (estado inicial do sistema).

Vivacidade (*liveness*): uma Rede de Petri é *viva* se e somente se para toda TRANSIÇÃO (t_i) e para qualquer MARCAÇÃO (M_i), existe uma seqüência de disparos de transições a partir de M_i que habilita t_i . A vivacidade de uma RP indica a ausência da possibilidade de ‘bloqueios’ (*deadlock*), garantindo que todas os estados sejam acessíveis e assim, que todos os processos modelados possam ser executados. Uma das principais razões para que ocorra um “bloqueio” é o compartilhamento indevido ou sem controle de recursos. Pode-se demonstrar que quatro condições devem ser observadas simultaneamente para que ocorra um ‘bloqueio’, ou em outras palavras, para evitar a ocorrência de um ‘bloqueio’ é suficiente garantir que pelo menos uma destas condições nunca seja satisfeita (Hasegawa, 1996):

- exclusão mútua por compartilhamento de recursos.
- ocupação do recurso enquanto aguarda a liberação de outros recursos para executar um processo que requer todos estes recursos conjuntamente.

- não liberação de um recurso utilizado em um processo por outro processo.
- existência de um caminho circular.

B. PROPRIEDADES ESTRUTURAIS

São aquelas que dependem da topologia das redes (Cardoso; Valette, 1997); (Murata 1989). Elas são independentes da marcação inicial, isto é, das condições iniciais do sistema. Assim, sua análise pode ser baseada na álgebra linear aplicada à estrutura do grafo. Estas propriedades podem ser caracterizadas em termos da matriz de incidência que é a representação matricial da estrutura do sistema.

As propriedades estruturais são definidas através dos componentes conservativos de LUGAR e dos componentes repetitivos estacionários (Cardoso; Valette, 1997). A partir destes elementos estruturais pode-se utilizar também a informação sobre a MARCAÇÃO, definindo-se assim, os invariantes de LUGAR e de TRANSIÇÃO.

Componente conservativo ou invariante de lugar: um invariante de LUGAR é uma função linear da marcação dos LUGARES cujo valor é uma constante e que depende apenas da marcação inicial da rede. Ele corresponde a uma restrição sobre os estados e as atividades do sistema que será sempre verificada, quaisquer que sejam suas evoluções. O conjunto de LUGARES envolvidos em uma invariante de LUGAR é chamado de componente conservativo de LUGAR e é dado pela solução da equação:

$$\mathbf{f}^T * \mathbf{C} = 0,$$

Em que:

\mathbf{f}^T é um vetor de LUGARES,

\mathbf{C} é a matriz de incidência (nxm) definida por $\mathbf{C} = \mathbf{O}(\mathbf{P},\mathbf{T}) - \mathbf{I}(\mathbf{P},\mathbf{T})$

$\mathbf{I}(\mathbf{P},\mathbf{T})$: Função que define os ARCOS de entrada das transições.

$\mathbf{O}(\mathbf{P},\mathbf{T})$: Função que define os ARCOS de saída das transições.

A matriz de incidência de uma Rede de Petri de n TRANSIÇÕES e m LUGARES corresponde a uma matriz de nxm números inteiros. Cada elemento desta matriz representa a forma que os LUGARES estão conectados com as TRANSIÇÕES em uma

rede de Petri. As colunas correspondem aos LUGARES e as linhas às TRANSIÇÕES, desta maneira, o número negativo indica que é um LUGAR de entrada, o positivo indica que é um LUGAR de saída e zero indica que não há relação entre o LUGAR e a TRANSIÇÃO correspondente.

Componente repetitivo ou invariante de transição: um invariante de TRANSIÇÃO é uma seqüência de disparos que não altera a MARCAÇÃO da Rede de Petri. O conjunto de TRANSIÇÕES de um invariante de TRANSIÇÃO é chamado componente repetitivo estacionário da rede e é dado pela solução da equação:

$$C * s = 0,$$

Em que:

C: matriz de incidência.

s: vetor característico, isto é, o vetor onde cada componente $s(t)$ representa o número de ocorrência da TRANSIÇÃO t em uma seqüência de disparo.

As propriedades procedurais e as propriedades estruturais estão interligadas, já que a dinâmica de uma Rede de Petri depende diretamente de sua estrutura.

B.1.7. MÉTODOS DE ANÁLISE DE UMA REDE DE PETRI

Uma vantagem de aplicar as Redes de Petri ao estudo de SED é a sua formalização matemática para a verificação das propriedades de um sistema modelado com esta técnica.

Uma vez que o modelo em rede de Petri é construído, este pode ser usado em duas formas de avaliação:

- **Lógica:** valida a lógica do modelo. Esta é uma visão estática que descreve a estrutura, os dados e as regras que manipulam os dados e executam tarefas. Assim é necessário verificar que a combinação de regras, dados e estrutura funcionem, isto é, que as regras sejam consistentes e completas. Isto pode ser realizado pela execução do modelo para garantir que este descreve apropriadamente a estrutura do sistema. Pode-se dizer que se realiza, neste caso, um “*debugging*” no modelo. Qualquer erro encontrado

deve ser corrigido na apropriada visão estática para preservar rastreabilidade.

- **Comportamental:** aqui examina-se o comportamento dinâmico do modelo, isto é a sua funcionalidade. A avaliação do comportamento compreende a resposta das seguintes perguntas:
 - O modelo produz todas as saídas devidas para um dado estímulo?
 - A informação chega às funções corretas e na seqüência correta? Isto é, as entradas são processadas no modo requerido?
 - O comportamento do modelo pode ser comparado com os requerimentos dos usuários?

Para avaliar o comportamento existem muitas técnicas entre elas estão: a simulação e a análise do espaço de estados

A análise das propriedades de um sistema é realizada através de 4 métodos (Zurawski; Zhou 1994); (Cardoso; Valette, 1997) a seguir explicados:

A. GERAÇÃO DA ÁRVORE OU GRAFO DE ALCANÇABILIDADE

Este método envolve a enumeração de todas as MARCAÇÕES (estados de um sistema) possíveis ou alcançáveis a partir da MARCAÇÃO inicial M_0 , a qual é a raiz da árvore (estado inicial). Cada nó representa uma MARCAÇÃO alcançável e cada ligação representa o disparo de uma TRANSIÇÃO (ocorrência de um evento) (Murata, 1989).

Um grande número de propriedades procedurais podem ser estudadas utilizando este método, entretanto, por ser um método exaustivo torna-se pouco efetivo para a análise de redes com um número considerável de LUGARES e TRANSIÇÕES.

B. ANÁLISE DA EQUAÇÃO DE ESTADO

Neste método, o comportamento do sistema em estudo é analisado através das equações algébricas da rede de Petri (Zurawski; Zhou, 1994).

Nesta abordagem, a matriz de incidência, que indica as conexões entre LUGARES e TRANSIÇÕES é utilizada para representar a estrutura do sistema. A equação de estado representa uma alteração na distribuição das MARCAS (estados) como

resultado do disparo de uma TRANSIÇÃO (ocorrência de um evento). A equação é definida da seguinte forma:

$$\mathbf{m}_k = \mathbf{m}_{k-1} + \mathbf{C}^T \mathbf{u}_k$$

Onde:

\mathbf{m}_k é um vetor coluna ($m \times 1$) que representa a marcação imediatamente alcançável a partir de \mathbf{m}_{k-1} após o disparo da TRANSIÇÃO t_i .

\mathbf{u}_k é um vetor ($n \times 1$) com grandeza de 1 na posição i que representa o disparo da TRANSIÇÃO t_i e zero nas outras posições.

C. SIMULAÇÃO

Esta técnica é usada quando o sistema é relativamente complexo e intratável utilizando métodos analíticos. Também é efetiva no estágio final do projeto para propósitos de validação. A maior parte das análises por simulação são realizadas para estudos de desempenho. Aqui o comportamento do modelo pode ser examinado executando este com entradas consistentes.

No entanto, a simulação só pode assegurar resultados para os casos simulados. Assim, para garantir uma generalização é necessário considerar uma estratégia de simulação/teste que evite a necessidade de simular todos os estados pois, isto resultaria em problemas relacionados com a capacidade de computação e com a alcançabilidade de estados que seria impraticável em modelos complexos.

D. TÉCNICAS DE REDUÇÃO

Este método consiste na redução de um modelo complexo em um modelo mais simples e tratável porém, sem perder as propriedades principais. A técnica de transformar modelos abstratos em modelos mais refinados (refinamentos sucessivos) de maneira hierárquica é um procedimento também utilizado para a síntese das redes.

7. REFERÊNCIAS BIBLIOGRÁFICAS

- ABRAMSON, A. B. The intelligent Building Evolution. In: INTELLIGENT BUILDING CONGRESS, 1995, Tel-Aviv. **Proceedings...** Tel-Aviv, Israel, 1995. p.309-318.
- AKEHURST, D.H.; DERRICK, J.; WATERS, A.G. **Addressing computational viewpoint design**. In: Enterprise Distributed Object Computing Conference, 2003. Proceedings. Seventh IEEE International 16-19 Sept. 2003 Page(s):147 – 158.
- ANANTARAM, C.; NAGARAJA, G.; NORI, K.V. Verification of Accuracy of rules in a Rule Based System. **Data & Knowledge Engineering**, v.27, n.2, p.115 – 138, Sep. 1998.
- ARKIN, H.; PACIUK, M. Service system integration in intelligent buildings. In: INTELLIGENT BUILDING CONGRESS, 1995, Tel-Aviv. **Proceedings...** Tel-Aviv, Israel, 1995. p.19-30.
- ABCI - ASSOCIAÇÃO BRASILEIRA DA CONSTRUÇÃO INDUSTRIALIZADA- As Edificações do Terceiro Milênio. **Construção**. n.2355, p.25-26, Março 1993.
- ASATO, O.; KATO, E.; INAMASU, R.; PORTO, A. Analysis of Open CNC Architecture for Machine Tools. **Journal of the Brazilian Society of Mechanical Sciences**, Rio de Janeiro v.24, n.3, 2002.
- AZEGAMI M.; FUJIYOSHI, H. A Systematic Approach to Intelligent Building Design. **IEEE Communications Magazine**, v.31, n.10, Oct. 1993.
- BARESI, L.; PEZZÈ, M. On Formalizing UML with High-Level Petri Nets. In: AGHA, G.A.; DE CINDIO, F.; ROZENBERG, G. (eds.). **Concurrent Object-Oriented Programming and Petri Nets - Lecture Notes in Computer Science**. Springer-Verlag, 2001. p.276-304.
- BASTIDE, R. Approaches in unifying Petri Nets and the Object-Oriented Approach. In: 16 TH INTERNATIONAL CONFERENCE ON APPLICATIONS AND THEORY OF PETRI NETS, 1ST. WORKSHOP ON OBJECT-ORIENTED PROGRAMMING AND MODELS OF CONCURRENCY, 1995, Turin. **Proceedings...** Turin, Italy, 1995.
- BASTIDE, R.; PALANQUE, P.; SY, O.; LE, D.; NAVARRE, D. Petri Net Based Behavioural Specification of CORBA Systems. **Lecture Notes in Computer Science: Application and Theory of Petri Nets 1999**, n.1639, p.66-85, 1999. / Apresentado a 20th. International Conference on Applications and Theory of Petri Nets, ICATPN'99, Williamsburg, USA, 1999.
- BARR, W.; BOYD, T.; INOUE, Y. The TINA initiative. **IEEE Comuncations Magazine**, v.31, n.3, p.70- 76, March 1993.

- BAX, M.; GIAMBIASI, N. OPENRT: A formalism for modeling, and simulating distributed real time systems. In: 7th EUROPEAN SIMULATION SYMPOSIUM, Erlangen -Nuremberg, 1995. **ESS'95**.
- BECERRA, J. **Aplicabilidade do Padrão de Processamento Distribuído e Aberto nos Projetos de Sistemas Abertos de Automação**. 1998. 162p. Tese (Doutorado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 1998.
- BECERRA, J. **Especificação, Projeto e Implementação de um Controlador de Célula flexível Aplicado na Manufatura**. 1993. 166p. Dissertação (Mestrado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 1993.
- BECKER, R. What is an Intelligent Building?. In: INTELLIGENT BUILDINGS CONGRESS, 1995, Tel-Aviv. **Proceedings...** Tel-Aviv, 1995. p.320.
- BETTONI, R. **Projeto de um Sistema de Supervisão e Controle Predial**. 1985. Dissertação (Mestrado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 1985.
- BLANC, X.; GERVAIS, M.-P.; LE DELLIU, J. **The specifications exchange service of an RM-ODP framework**. In: Enterprise Distributed Object Computing Conference, 2000. EDOC 2000. **Proceedings**. Fourth International 25-28 Sept. 2000 Page(s):86 – 90.
- BLANC, X.; GERVAIS, M.-P.; LE DELLIU R. Using the UML language to Express the ODP Enterprise Concepts. In: 3rd INTERNATIONAL ENTERPRISE DISTRIBUTING OBJECT COMPUTING CONFERENCE, 1999, Mannheim. **Proceedings...** IEEE Press (Ed). Mannheim, Germany, 1999. p.50-59. **EDOC'99**.
- BOITEN, E.; Bowman, H.; Derrick, J.; Linington, P.F.; Steen, M.W.A. Viewpoint consistency in ODP. **Computer Networks**. v.34, n.3, p.503-537, August 2000.
- BOOCH, G.; RUMBAUGH J.; JACOBSON, I. **The Unified Modeling Language User Guide**. Reading, Massachusetts: Addison-Wesley, 1999.
- BROCK, L.D. **Transition to an open system architecture [for weapon systems]**. In: Digital Avionics Systems Conferences, 2000. **Proceedings**. DASC. The 19th Volume 1, 7-13 Oct. 2000 Page(s):4D5/1 - 4D5/8 vol.1.
- BUCHS, D.; GUELF, N. **A formal specification framework for object-oriented distributed systems**. In: Software Engineering, IEEE Transactions on Volume 26, Issue 7, July 2000 Page(s):635 – 652.
- CALVEZ, J. P. **Embedded Real-Time Systems**. New York: John Wiley & Sons, 1993.
- CALLAGHAN, V.; CLARKE, G.; SHARPLES, S. *A Multi-Agent for Intelligent Building Sensing and Control*. *Internacional Sensor Review Journal*. Maio, 1999.
- CANTÚ, M. **Dominando o Delphi – “A bíblia”**. São Paulo. Makron Books do Brasil, 1995.
- CHANG, W-T; TSENG, C-C.; CHOU, W-K. Petri Net Based Analysis on Object Assignment in Distributed Object-Oriented Systems. **Journal of Systems Architecture**. v.44, n.12, p.955-970, 1998.
- CHEN, J.; TAYLOR, V.E. **Mesh partitioning for efficient use of distributed systems**. In: Parallel and Distributed Systems, IEEE Transactions on Volume 13, Issue 1, Jan. 2002 Page(s):67 – 79.
- CARDOSO, J.; VALETTE, R. **Redes de Petri**. Florianópolis: Editora da UFSC, 1997.

- CARDOSO, J.; SIBERTIN-BLANC. Ordering actions in Sequence Diagrams of UML. In: 23th INTERNATIONAL CONFERENCE ON INFORMATION TECHNOLOGY INTERFACES, 2001, Pula. **Proceedings...** Pula, Croatia, 2001.
- CHANG, W.; TSENG, C.; CHOU, W. Petri net based analysis on object assignment in distributes object-oriented systems. **Journal of systems architecture**. v.44, n.12, p. 995-970, 1998.
- CIC. **HomePnP Specification**: version 1.0. CEBus Industry Council. 1998. 369p. Disponível em: <<http://www.cebus.org/Files/hpnp10.zip>>. Acesso em: dic 2002.
- CIC, CEBus Industry Council Inc. **Home Plug & Play Specification**. 1998. 369p. Disponível em: < <http://www.cebus.org/News&Events.html>>. Acesso em: dez. 2002.
- CLARK G.; MEHTA, P. PROWSE, R. Intelligent Integrated Building Management Systems. In: INTELLIGENT BUILDING CONGRESS, 1995, Tel-Aviv. **Proceedings...** Tel-Aviv, Israel, 1995. p.9-18.
- COSTA, C.A.; HARDING, J.A.; YOUNG, R.I.M. The application of UML and an Open Distributed Process framework to information system design. In: **Computers in Industry**. v.46, n.1, p.33-48, August 2001.
- DAVID, R.; ALLA, H. Petri Nets for Modeling of Dynamic Systems: A Survey. **Automatica**, v.30, n.2, p.175-201, 1994.
- DEBONI, J. E. Modelagem orientada a objetos com a UML. Editora Futura, São Paulo, 2003
- DIAGNE, A.; ESTRAILLIER P. **A Component-based Framework for the Specification, Verification and Validation of Open Distributed Systems. Rapport de Recherche, 18 p. Dic 1997.**
- DIAGNE, A. **Une approche Multi-formalismes de Specification de Systèmes Répartis: Transformation de Composants Modulaires en Réseaux de Petri.** 1996. Tèse (Docteur) - Université Paris 6. Paris, 1996.
- DILONARDO, L. **Avaliação do Uso de Tecnologias Passivas visando a Eficiência Energética em Edifícios de Escritórios.** 2001. 250p. Dissertação (Mestrado) – Faculdade de Arquitetura e Urbanismo, Universidade de São Paulo. São Paulo, 2001.
- DOMINGUEZ, E.; PINOTTI JR. M.; PACHECO, A. Proposta de uma Nova Estratégia em Redes Fieldbus Usando Controle Adaptativo por Modelo de Referência. In: II CONGRESSO NACIONAL DE ENGENHARIA MECÂNICA, João Pessoa, 2002. **Anais.** João Pessoa, Brasil, 2002.
- DOUGLASS, P.D. **Real Time UML**: developing efficient objects for embedded systems. Harlow England: Addison- Wesley Longman, Inc., 1998.
- DURAN, F.; HERRADOR, J.; VALLECILLO, A. **Using UML and Maude for writing and reasoning about ODP policies.** In: Policies for Distributed Systems and Networks, 2003. Proceedings. POLICY 2003. IEEE 4th International Workshop on
4-6 June 2003 Page(s):15 - 25.
- ECKERT, K.P. From OSI to OMG: Experiences from the Port of an ISODE-Based Application to OMG Concepts. In: COMPUTER COMMUNICATIONS (JOURNAL) ON THE 1ST ST. PETERSBURG WORKSHOP, Thompson, J.

- (ed.), Elsevier Science Publishers B.V., North-Holland, Amsterdam (NL), 1996. p.4-12.
- ELKOUTBI, M.; KELLER, R.K. User Interface Prototyping based on UML Scenarios and High-level Petri Nets. In: 21st INTERNATIONAL CONFERENCE ON APPLICATION AND THEORY OF PETRI NETS, 2000, Aarhus. **Proceedings...** Aarhus, Denmark, 2000. p.166-186.
- FAROOQUI, K.; LOGRIPPO, L.; MEER, J. The ISO Reference Model for Open Distributed Processing: an introduction. **Computer Networks and ISDN Systems**. v.27, n.11, p. 1215-1229. July 1995.
- FAROOQUI, K.; LOGRIPPO, L. **On the Notions of Abstraction, Consistency, and Design in the ODP Framework of Viewpoints**. Ottawa, University of Ottawa. Disponível em: <<http://lotos.site.uottawa.ca/ftp/pub/Lotos/TechRep/>>. Acesso em: maio 2002.
- FLAX, B. Intelligent Buildings. **IEEE Communications Magazine**. v.29, n.4, p.24-27. April 1991.
- FINLEY, M.R.; KARAKURA, A.; NBOGNI, R. Survey of Intelligent Buildings Concepts. **IEEE Communications Magazine**. v.29, n.4, p.18-23. April 1991.
- FU, L.C.; SHIH, T.J. Holonic supervisory control and data acquisition kernel for 21st century intelligent building system, In: Proceedings of the 2000 IEEE International Conference on Robotics and Automation, San Francisco, CA, IEEE Computer Society Press, Washington DC, 2000, pp. 2641– 2646.
- FUJIE, S.; MIKAMI, Y. Construction Aspects of Intelligent Buildings. **IEEE Communications Magazine**. v.29, n.4, p.50-57. April 1991.
- FURLAN, J. D. **Modelagem de Objetos através de UML**. São Paulo: Makron Books. 1998.
- GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J. **Design Patterns –Elements of Reusable Object-Oriented Software**. Massachusetts: Addison-Wesley Publishing Company, Reading, 1995.
- GASPOZ, J. P. Methodology for the Development of Distributed Telecommunications Services. **Journal of Systems and Software**, Elsevier Science. v.33, n.3, p.253-271. June 1996.
- GEHRKE, T.; GOLTZ, U.; WEHRHEIM, H. The Dynamic Models of UML: Towards a semantic and its application in the Development Process, **Hildesheimer Informatik-Bericht 11/98**, Institut für Informatik, Universität Hildesheim, Germany, 1998.
- GIESE, H.; GRAF, J.; WIRTZ, G. Closing the gap between object-oriented modeling of structure and behaviour. In: 2nd INTERNATIONAL CONFERENCE ON THE UNIFIED MODELING LANGUAGE, 1999, Colorado. **Proceedings**. Colorado, USA, 1999.
- GIESE, H.; WIRTZ, G. Using UML and Object-Coordination-Nets for Workflow Specification., In: 2nd INTERNATIONAL CONFERENCE ON THE UNIFIED MODELING LANGUAGE, 1999, Colorado. **Proceedings**. Colorado, USA, 1999.
- GIESE, H.; WIRTZ, G. Visual Modeling of Object-Oriented Distributed Systems. **Journal of Visual Languages and Computing**. v.12, n.2, p.183-202. April 2001.
- GOMEZ, L F. **Redes de Petri Reactivas e Hierárquicas - integração de formalismos no projecto de sistemas reactivos de tempo-real**. 1997. 240p. Tese (Doutorado) - Universidade Nova de Lisboa. Lisboa, 1997.

- GUSTIN BASTIDAS, G. D. Development of an Open Distributed Approach for Building Automation. Artigo submetido ao 8TH INTERNATIONAL CONGRESSO F MECHANICAL ENGINEERING, Ouro Preto, 2005. **Proceedings...** Ouro Preto, Brasil, 2005. **COBEM 2005**. não publicado.
- GUSTIN BASTIDAS, G. D. Modelagem de Sistemas Abertos e Distribuídos para Automação Predial. In: XV CONGRESSO BRASILEIRO DE AUTOMÁTICA, Gramado, 2004. **Anais...** Gramado, Brasil, 2004. **CAB 2004**.
- GUSTIN BASTIDAS, G. D. An Open Distributed Systems Approach for Building Automation. In: IX MECHATRONICS FORUM INTERNATIONAL CONFERENCE 2004, Ankara, 2004.. **Proceedings...** Ankara, Turquia, 2004. **MECHATRONICS 2004**. GUSTIN BASTIDAS, G. D. Open Distributed Supervisory System Design using Petri Nets,; IEEE INTERNATIONAL SYMPOSIUM ON INDUSTRIAL ELECTRONICS.). Rio de Janeiro, 2003. **Proceedings...** Rio de Janeiro, Brasil, 2003. **ISIE 2003**.
- GUSTIN BASTIDAS, G. D. Petri Net Approach for Modelling System Integration in Intelligent Buildings. **Journal of the Brazilian Society of Mechanical Sciences**. v. XXIV, p.309 – 318, 2002. Rio de Janeiro, 2002.
- GUSTIN BASTIDAS, G. D. Modelagem de um sistema aberto para a integração de sistemas em edifícios Inteligentes. In: II CONGRESSO NACIONAL DE ENGENHARIA MECÂNICA 2002.). João Pessoa, 2002. **Anais...** João Pessoa, Brasil, 2002. **CONEM 2002**.
- GUSTIN BASTIDAS, G. D. Modelagem de um Sistema Aberto e Distribuído para Automação Predial. In: II INTERNATIONAL CONFERENCE ON AUTOMATIC CONTROL 2002. Santiago de Cuba, 2002. **Anais...** Santiago de Cuba, Cuba 2002. **AUT 2002**.
- GUSTIN BASTIDAS, G. D. Modeling Integration of Systems in Intelligent Buildings through Mechatronics Approach. In: VIII MECHATRONICS FORUM INTERNATIONAL CONFERENCE 2002. Enschede, 2002. **Proceedings...** Enschede, Holanda, 2002. **MECHATRONICS 2002**.
- GUSTIN BASTIDAS, G. D. Metodologia para a Modelagem, de Sistemas de Elevadores em Edifícios Inteligentes. In: I CONGRESSO NACIONAL DE ENGENHARIA MECÂNICA 2000. Natal/RN, 2000. **Anais...** Natal/RN, Brasil, 2000. **CONEM 2000**.
- GUSTIN BASTIDAS, G. D. **Aplicação de Redes de Petri Interpretadas na Modelagem de Sistemas de Elevadores em Edifícios Inteligentes**. 1999. 164p. Dissertação (Mestrado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 2000.
- HAN, C.Y. An OOP Based Intelligent Building Information System in New Methods and Technologies in Planning and Construction of Intelligent Building. In: 2nd. IB/IC INTELLIGENT BUILDING CONGRESS, 1997, Tel-Aviv. **Proceedings**. Tel-Aviv, Israel, 1997.
- HASEGAWA, K.; MIYAGI, P.E.; TAKAHASHI, K. Mark Flow Graph (MFG) para modelamento e controle de sistemas de eventos discretos. **Monografias em Automação e Inteligência Artificial**. PMC/PEL/EPUSP, São Paulo. v.1, n.1, p.1-10. 1989.

- HASEGAWA, K. Modeling, Control and Deadlock Avoidance of FMS. In: XI CONGRESSO BRASILEIRO DE AUTOMÁTICA: Conferências Plenárias, 1996, São Paulo. p.37-51. **XI CBA**.
- HO, Y. **Discrete Event Dynamic Systems: Analyzing Complexity and Performance in the Modern World**. New York: Institute of Electrical and Electronics Engineers. 1992.
- HOLLOWAY, E.L.; KROGH, B.H.; GIUA, A. A Survey of Petri Net Methods for Controlled Discrete Event Systems. **Journal of Discrete Event Dynamic Systems**. v. 7, n.2, 1997.
- HOLVET, T.; VERBAETEN, P. Using Petri nets for specifying Active Objects and Generative Communication. **Lecture Notes in Computer Science: Concurrent Object-Oriented Programming and Petri Nets**. Advances in Petri nets 1999, n.1639, p.66-85, 1999. Gul A Agha, De Cindio, Rozenberg, editores.
- HOLVET, T.; VERBAETEN, P. Using Petri nets for specifying Active Objects and Generative Communication. **Lecture Notes in Computer Science: Concurrent Object-Oriented Programming and Petri Nets**. Advances in Petri nets 1999, n.1639, p.66-85, 1999. Gul A Agha, De Cindio, Rozenberg, editores.
- ISO/IEC 10746-1: Information Technology – Open Distributed Processing – Reference Model: Overview, 1998.
- ISO/IEC 10746-2: Information Technology – Open Distributed Processing – Reference Model: Foundations. 1996.
- ISO/IEC 10746-3: Information Technology – Open Distributed Processing – Reference Model: Architecture. 1996.
- ISO/IEC 10746-4: Information Technology – Open Distributed Processing – Reference Model: Architectural Semantics, 1998.
- JENSEN, K. Coloured Petri nets: a high level language for system design and analysis. **Lecture Notes in Computer Science: Advances in Petri Nets 1990**. Berlin: G. Rozenberg, Ed.; Springer-Verlang. v.483, p.342-416, 1996.
- KANDÉ, M.M.; MAZAHER, S.; PRNJAT, O.; SACKS, L.; WITTING, M. Applying UML to Design an Inter-Domain Service Management Application. In: UML'98: BEYOND THE NOTATION - INTERNATIONAL WORKSHOP, 1998, Mulhouse. **Proceedings...** Mulhouse, France, 1998.
- KAM Hay Fung; LOW, G.; BAY, P.K. **Embracing dynamic evolution in distributed systems Software**, IEEE Volume 21, Issue 2, March-April 2004 Page(s):49 – 55.
- KATCHABAW, M.J.; HOWARD S.L.; LUTFIYYA H.L.; MARSHALL A.D.; BAUER M.A. Making distributed applications manageable through instrumentation. **The Journal of Systems and Softwares**. v.45, n.2, p.81-97, 1999.
- KILOV, H. Precise specification of behavior in object-oriented standardization activities. **Computer Standards and Interfaces**, v.15, n.2-3, p.275-285, 1993.
- KOKAI, Y.; MASUDA, F.; HORIIKE, S.; SEKINE, Y. Recent development in open systems for EMS/SCADA. **Internacional Journal of Electrical Power & Energy Systems**. v.20, n.2, p.111-123, February 1997.
- KRONER, W.M. An intelligent and responsive architecture. **Automation in Construction**. v.6, n.5-6 p. 381-393, September 1997.
- KRONER, W. Towards an Intelligent Architecture. In: INTELLIGENT BUILDING CONGRESS, 1995, Tel-Aviv. **Proceedings**. Tel-Aviv, Israel, 1995. p.60-73.

- LAKOS C. The Object Orientation of Object Petri Nets. In: 1st WORKSHOP ON OBJECT-ORIENTED PROGRAMMING AND MODELS OF CONCURRENCY, 1995, Turin. **Proceedings**. Turin, Italy, 1995.
- LAKOS C. From Coloured Petri Nets to Object Petri Nets. In: 16th INTERNATIONAL CONFERENCE ON THE APPLICATION AND THEORY OF PETRI NETS, 1995, Turin. **Proceedings**. Turin, Italy, 1995.
- LAKOS C. The consistent use of names and polymorphism in the definition of Object Petri Nets. In: 17th INTERNATIONAL CONFERENCE ON THE APPLICATION AND THEORY OF PETRI NETS, 1996, Osaka. **Proceedings**. Osaka, Japan, 1996. p.380-399.
- LAM-SON Le; WEGMANN, A. **Definition of an Object-Oriented Modeling Language for Enterprise Architecture System Sciences**, 2005. In: HICSS '05. Proceedings of the 38th Annual Hawaii International Conference on 03-06 Jan. 2005 Page(s):222a - 222a.
- LIU, Z. G.; MAKAR, J. M.; A. K. Developmente of fire detection systems in the intelligent buildings. Construction. Institute for Research in Construction 2001. In: 12th INTERNATIONAL CONFERENCE ON AUTOMATIC FIRE DETECTION, 2001, Gaithersburg, MD **Proceedings**. Osaka, Japan, 1996. p561-573.
- MACARTNEY, A.J.; BLAIR, G.S. Flexible trading in distributed multimedia systems. **Computer Networks and ISDN Systems**. v.25, n.2, p.145-157, 1992.
- MAEDA, S. **Intelligent Buildings: a key solution for the 21st century office**. 1993. 185p. Tese (doutorado) - Stanford University. Stanford, 1993.
- MARTE, C.L. **Estudo e Análise das Unidades Funcionais Remotas no Controle Distribuído em Processo de Automação Predial**. 1994. 171p. Dissertação (Mestrado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 1994.
- MARTE, C.L. **Sistemas Computacionais distribuídos aplicados em automação dos transportes**. 2000. 324p. Tese (Doutorado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 2000.
- MARTUCCI JÚNIOR, M. **Estudos de Estruturas de Sistemas de Automação**. 1992. 190p. Tese (Livre Docência) - Escola Politécnica, Universidade de São Paulo. São Paulo, 1992.
- MATSUSAKI, C. **Redes F-MFG (Functional Mark Flow Graph) e sua Aplicação no Projeto de Sistemas Antropocentricos**. 1998. 89p. Dissertação (Mestrado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 1998.
- MATSUYAMA, F. **Generalização dos níveis de controle e instrumentação de um sistema aberto de automação**. 1997. 251p. Tese (Doutorado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 1997.
- MEHTA, P.; CLARK, G.; PROWSE, R. Intelligent Integrated Buildings Management Systems. In: INTELLIGENT BUILDING CONGRESS, 1995, Tel-Aviv. **Proceedings**. Tel-Aviv, Israel, 1995. p.9-18.
- MAIER, CH., MOLDT, D. Object Coloured Petri Nets – A Formal Technique for Object Oriented Modelling. **Lecture Notes in Computer Science: Advances in Petri Nets 2001**. Berlin: G. Agha et al., Ed.; Springer-Verlang. v.2001, p.406-427, 2001
- MELLO FILHO H. B. Sistemas Abertos – Um nova era na informática. **Bytr Brasi**;, p. 93-4, maio 1992:

- MELLO FILHO H. B. *Sistemas Abertos – Um nova era na informática*. **Byte Brasil**. v.1, n.5, p.93-94, Maio 1992.
- MICROSOFT. **Understanding Universal Plug and Play: White Paper**. Microsoft Corporation 2000, Redmond. 45p. Disponível em: <http://www.upnp.org/download/UPNP_UnderstandingUPNP.doc>. Acesso em: jan. 2003.
- MIYAGI, E.P.; GUSTIN, G.; ARAKAKI, J.; VILLANI, E.; MIYAGI, M.; KISIL, M. *Integração de Atividades e Serviços em Edifícios Inteligentes - Aplicação da Metodologia PFS/MFG*. In: XII ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 1998, Rio de Janeiro. **Anais... Rio de Janeiro, Brasil, 1998. XII ENEGEP'98**.
- MIYAGI, E.P. **Controle Programável - fundamentos do controle de sistema a eventos discretos**. São Paulo: Editora Edgard Blücher, 1996.
- NANKMAN, M.A.; NIEUWENHUIS, L.J.M. Specification of a distributed storage system. **Computer Communications**, v.19, n.1, p.30-38, 1996.
- OBJECT MANAGEMENT GROUP (OMG). *The Common ObjectRequest Broker: Architecture and specification*. Revision 2.0. 1995.
- OMG UML 2 Infrastructure Final Adopted Specification. Object Management Group, December 2003.
- OLIVEIRA, H. **Uma Metodologia para a integração da Informação em Sistemas abertos**. 1998. 264p. Tese (Doutorado) – Escola Politécnica, Universidade de São Paulo. São Paulo, 1998.
- PALUDETTO, M. **Sur la commande des procédés industriels: une méthodologie basée objects et réseaux de Petri**. 1991. 270p. Thèse (Doctorat), Université Paul Sabatier. Toulouse, France, 1991.
- PALUDETTO, M.; DELATOUR, J. UML et les réseaux de Petri – vers une sémantique des modèles dynamiques et une méthodologie de développement des systèmes temps réel. **L'object**. v.5, n.3-4, p.443-468, Décembre 1999.
- PALUDETTO, M. Petri Nets and Girard's Linera Logic as a semantics for RT-UML dynamic Diagrams. In: WORLD MULTICONFERENCE ON SYSTEMS, CYBERNETICS AND INFORMATICS, 2001, Orlando. **Proceedings... Orlando, Estados Unidos, 2001. SCI'2001**.
- PERKUSICH, A.; PERKUSICH, M.L.B.; CHANG, S.K. Object oriented design, modular analysis, and fault-tolerance of real-time control *software* systems. **International Journal of Software Engineering and Knowledge Engineering**. v.6, n.3, p.447-476, 1996.
- PERTUSIER, M.R.O. **Implantação, Operação e Manutenção de Edifícios de Alta Tecnologia. Curso sobre Edifícios Inteligentes**. Resumo Faculdade de Tecnologia de São Paulo – FATEC. São Paulo, 1993.
- PIANEGIANI, F.; MACII, D.; CARBONE, P. **Open distributed control and measurement system based on an abstract client-server architecture**. In: Virtual and Intelligent Measurement Systems, 2002. VIMS '02. 2002 IEEE International Symposium on 19-20 May 2002 Page(s): 63 – 6?.
- PINTO, S. **Hospitais**. Brasília: Thesaurus, 1996.
- PIZZICA, S. **Open systems architecture solutions for military avionics testing**. In: Aerospace and Electronic Systems Magazine, IEEE Volume 16, Issue 8, Aug. 2001 Page(s):4 – 9.

- PUTMAN, J. **Architecting with RM-ODP**. New Jersey: Prentice Hall, 2001.
- RAMAGDE, P.J.; WONHAM, W.M. The Control of Discrete Event Systems. **Proceedings of the IEEE**, v.77, n.1, p.81-98, 1989.
- RIASCOS, L. **Modelagem do Tratamento de Falhas em Sistemas de Manufatura Através de Rede de Petri**. Dissertação (Mestrado) – Escola Politécnica, Universidade de São Paulo São Paulo, 1998.
- ROMERO, R.; VALLECILLO, A. **Formalizing ODP computational viewpoint specifications in Maude**. In: Enterprise Distributed Object Computing Conference, 2004. EDOC 2004. Proceedings. Eighth IEEE International 20-24 Sept. 2004 Page(s):212 – 223.
- ROSA, P.F. **Um ambiente para Desenvolvimento de Sistemas Distribuídos**. 1995. 167p. Tese (Doutorado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 1995.
- ROSENBERRY, W.; KENNEY, D.; FISHER, G. **Understanding DCE**. Sebastopol, CA: O'Reilly & Associates, Inc., 1993.
- ROTH, B. Administración, Gestión, Control y Servicio de Edificios. **Desarrollo Nacional**. v.35, n.9, Noviembre 1997.
- SANCHEZ, A, TORRES, D., FERNÁNDEZ E., HABER, R. Apuntes para una metodología de apoyo al diseño del sistema de comunicaciones en un edificio inteligente. In: III WORKSHOP SINTED SOBRE EDIFICIOS INTELIGENTES, 1998, Cancun. **Memórias...** Cancun, México, 1998. p.163-226.
- SANTOS FILHO, D.J. **Controle de Sistemas Antropocêntricos de Produção baseado em Redes de Petri Interpretadas**. 1998. 287p. Tese (Doutorado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 1998.
- SANTOS FILHO, D.J. **Proposta do Mark Flow Graph estendido para a Modelagem e Controle de Sistemas Integrados de Manufatura**. 1993. 155p. Dissertação (Mestrado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 1993.
- SARAMAGO, M.A.P. **Integração de Dispositivos Inteligentes utilizando Conceitos de Domótica direcionados a Automação Hospitalar**. 2002. 224p. Tese (Doutorado) – Universidade de Campinas. Campinas-SP, 2002.
- SCHERER, D; MURER, T.; WÜRTZ A. Designing the Distributed Architecture DIPS for Cooperative Software Engineering. **Distributed Systems Engineering**. v.4, n.3, p.160 -168, September 1997.
- SCHOFIELD, S. **Open Architecture Controller For Advanced Machine tools**. 1995. Thesis (PHD) - University of Califórnia. Berkeley, 1995.
- SHANKARAN, N.; KLEFSTAD, R. **ZEUS: a CORBA framework for service location and creation**. In: Applications and the Internet Workshops, 2004. SAINT 2004. Workshops. 2004 International Symposium on 26-30 Jan. 2004 Page(s):382 - 387.
- SILVA, M.; FERREIRA, D. **Especificação e Desenvolvimento de um controlador E-MFG para Sistema Flexíveis de Produção**. 1998. 356p. Trabalho de Formatura – Escola Politécnica, Universidade de São Paulo. São Paulo, 1998.
- SINNOTT, R.O.; TURNER, K.J. Applying the architectural semantics of ODP to develop a trader specification. **Computer Networks and ISDN Systems**, v.29, n.4, p.457-471, 1997.
- SCHÖF, S.; SONNENSCHNEIN, M; WIETING, R. High-Level Modeling with Thor nets. In: 14TH INTERNATIONAL CONGRESS ON CYBERNETICS, 1995, Namur. **Proceedings...** Namur, Belgica, 1995.

- SO, A., WONG, A., WONG, K. A New definition of intelligent buildings for Asi. In: *The Intelligent Building Index Manual*, 2nd edition, Asian Institute of Intelligent Buildings, Hong Kong, 2001.
- STEEN, M.W.A.; DERRICK, J. **Formalising ODP enterprise policies**. In: *Enterprise Distributed Object Computing Conference*, 1999. EDOC '99. Proceedings. Third International 27-30 Sept. 1999 Page(s):84 – 93.
- SUN MICROSYSTEMS. **Jini™ Technology Surrogate Architecture Specification**. Palo Alto, 2001. Disponível em: <<http://www.jini.org/standards/sa.pdf>>. Acesso em: mar. 2003.
- TAIANI, F.; PALUDETTO, M.; DELATOUR, J.; CROS, T. Vérification d'objects temps réel à l'aide des reseaux de Petri et de la logique Linéaire. In: *9th CONFERENCE ON REAL TIME AND EMBEDDED SYSTEMS*, 2001, Paris. **RTS'2001**. p. 65 – 77.
- TANAKA, A.; NAGASE, Y.; KIRYU, Y.; NAKAI, K. **Applying ODP Enterprise Viewpoint Language to hospital information systems**. In: *Enterprise Distributed Object Computing Conference*, 2001. EDOC '01. Proceedings. Fifth IEEE International 4-7 Sept. 2001 Page(s):188 – 192.
- TAKIUCHI, M.; MELO, E.; TONIDANDEL F. *Domótica Inteligente: Automação Baseada em Comportamento*. CBA 2004.
- TOKMAKOFF, A. A. **Modelling, Analysis and Prototyping of teh ODP trader using Coloured Petri Nets and Java**. 1998. 481p. Tese (Doutorado) – University of South Australia. Warrendi Road, 1998.
- TSCHAMMER V.; MAGEDANZ, T.; TSCHICHHOLZ, M.; WOLISZ, A. Cooperative Management in Open Distributed Systems. **Computer Communications**, v.17, n.17, p.7171-728. 1994.
- VALK, R. Self-modifying nets: a natural extension of Petri nets. **Lecture notes in computer science: Proceedings of ICALP78**. v. 62, p.464-476, 1978.
- VALK, R. Petri Nets as Token Objects: An Introduction to Elementary Object Nets. **Lecture Notes in Computer Science: Application and Theory of Petri Nets**. n.1420, p.1-25, 1998.
- VALK, R. **Relating Different Semantics for Object Petri Nets, Formal Proofs and Examples**. Report *FBI-HH-B-226/00*, Universität Hamburg. April 2000. p.1-50.
- VILLANI, E. **Aplicação de Redes de Petri Interpretadas na Modelagem de Sistemas de Elevadores em Edifícios Inteligentes**. 1999. 164p. Dissertação (Mestrado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 1999.
- VISCONDE, M.G. **Programação de Projetos Hospitalares** 1999. 164p. Dissertação (Mestrado) – Faculdade de Arquitetura e Urbanismo, Universidade de São Paulo. São Paulo, 1999.
- WANG, L. Object-oriented Petri Nets for modelling and analysis of automated manufacturing systems. **Computer Integrated Manufacturing Systems**, v.26, n.2, p.111-125, 1996.
- WANG, S.; XIE, J. **Conceptual modeling of complex systems using an RM-ODP based ontology**. In: *Integrating Building Management System and facilities management on the Internet • ARTICLE Automation in Construction*, Volume 11, Issue 6, October 2002, Pages 707-715 Shengwei WANG and Junlong XIE.

- WEGMANN, A.; NAUMENKO, A. Enterprise Distributed Object Computing Conference, 2001. In: EDOC '01. Proceedings. Fifth IEEE International 4-7 Sept. 2001 Page(s):200 – 211.
- WEN-TSUNG, C.; CHIEN-CHAO T.; WEN-KUANG, C. Petri Net Based Analysis on Object Assignment in Distributes Objetc-Oriented Systems. **Journal of Systems Architecture**. v.44, n.12, p.955-970, 1998.
- WHELER, T. **O Manual dos Sistemas Abertos**. (Tradução de Open Systems Handbook/ Cristina Bazán). Rio de Janeiro: Editora Campus, 1994.
- WIGGINTON, M., HARRIS, J. **Intelligent Skin**. Oxford, UK: Architectural, 2002.
- WOLISZ, A.; TSCHAMMER, V. Performance aspects of trading in open distributed systems. **Computer Communications**. v.16, n.5, p.277-287, May 1993.
- WONG, J. K. W.; LI, H.; WANG, S. W. **Intelligent building research: a review • HORT SURVEY** Automation in Construction, Volume 14, Issue 1, January 2005, Pages 143-159. J. K. W. WONG, H. LI and S. W. WANG
- WRIGHT, P.; SCHOFIELD, S.; WANG, F.C. **Open architecture control for machine tools**. Integrated Manufacturing Laboratory, University of California, Berkeley. Nov. 1996.
- ZAPF, M.; HEINZL, A. **Techniques for Integrating Petri Nets and Object-Oriented Concepts**. Working Papers in Information Systems. Universität Bayreuth. 1999.
- ZHA, X.F. An object-oriented knowledge based Petri net approach to intelligent integration of design and assembly planning. **Artificial Intelligence in Engineering**. v.14, n.1, p.83-112, 2000.
- ZURAWSKI, R.; ZHOU, M. Petri nets and industrial applications: a tutorial. **IEEE Transactions on Industrial Electronics**, v.41, n.6, p.567-583, December 1994.