MÁRIO ROBERTO BASTOS

**ARQUITETURA ODP-CIM APLICADA NA PREVISÃO DISTRIBUÍDA DA
CARGA DO SISTEMA ELÉTRICO DE POTÊNCIA**

Dissertação apresentada à Escola
Politécnica da Universidade de São Paulo
para obtenção do Título de Mestre em
Engenharia

São Paulo
2006

MÁRIO ROBERTO BASTOS

**ARQUITETURA ODP-CIM APLICADA NA PREVISÃO DISTRIBUÍDA DA
CARGA DO SISTEMA ELÉTRICO DE POTÊNCIA**

Dissertação apresentada à Escola
Politécnica da Universidade de São Paulo
para obtenção do Título de Mestre em
Engenharia

Área de Concentração:

Sistemas Digitais

Orientador:

Prof. Dr. Jorge Luis Risco Becerra

São Paulo
2006

Este exemplar foi revisado e alterado em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 22 de junho de 2006.

Assinatura do autor _____

Assinatura do orientador _____

FICHA CATALOGRÁFICA

Bastos, Mário Roberto

Arquitetura ODP-CIM aplicada na previsão distribuída da carga do sistema elétrico de potência / M.R. Bastos. -- ed.rev. -- São Paulo, 2006.

p.

Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Computação e Sistemas Digitais.

1.Arquitetura de software 2.Modelagem de dados 3.Sistemas distribuídos 4.Sistema de informação 5.Sistemas elétricos de potência I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Computação e Sistemas Digitais II.t.

À minha família, pelo incentivo, apoio e pela compreensão pelas horas ausentes.
À minha mãe e meu pai (em memória) por tudo que proporcionaram e a
Deus pela suavização da estrada trilhada.

**“Se queremos progredir,
não devemos repetir a história,
mas fazer uma história nova”
Gandhi**

AGRADECIMENTOS

Ao amigo e orientador professor Dr. Jorge Luiz Risco Becerra, pelo agradável convívio durante todo o período, pelo incentivo constante, pelas valiosas orientações e contribuições, pelos caminhos propostos com a liberdade para que eu os percorresse à minha maneira e, principalmente por acreditar que fosse possível.

À Companhia de Transmissão de Energia Elétrica Paulista, pelo incentivo à criatividade, ao conhecimento e capacidade de aprendizado, nas pessoas de seu Presidente Dr. José Sidnei Colombo Martini, de seu Diretor Técnico Dr. Celso Sebastião Cerchiari e de seu Gerente de Operação Dr João Carlos Faria Costa, cultivadores do comprometimento e capacidade de aprender dos seus profissionais, fazendo da educação uma estratégia da empresa e não apenas uma estratégia a ser alinhada à estratégia da empresa.

Ao Gerente da Divisão de Supervisão e Automação (TOA) da CTEEP, Eng. Paulo Roberto Pedroso de Oliveira, pela sua percepção do valor do conhecimento ao proporcionar aos profissionais as condições para o desenvolvimento de competências, novas habilidades e atitudes.

À Escola Politécnica da USP, pela cessão de suas instalações e aos seus professores, pela atenção e partilha de conhecimentos, imprescindíveis para a concretização deste trabalho.

Aos amigos pelo incentivo e interesse constantes, demonstrados durante todo o período de desenvolvimento deste trabalho, através de sugestões, debates, críticas e opiniões. Juntos, compomos uma grande equipe.

E a todos aqueles que contribuíram, diretamente ou indiretamente, para a realização deste trabalho.

RESUMO

O presente trabalho apresenta a utilização do modelo CIM – Common Information Model com uma abordagem ODP – Open and Distributed Processing na definição de uma arquitetura distribuída. A modelagem ODP-CIM efetuada busca validar formalmente a escolha da WBEM – Web Based Enterprise Management como arquitetura distribuída responsável pela interoperabilidade entre sistemas heterogêneos de uma empresa de energia elétrica. O contexto WBEM adicionalmente inclui a modelagem CIM das informações, de modo que o ambiente resultante pode ser visto como uma coleção distribuída de sistemas inter-relacionados.

De modo a se verificar a adequação da WBEM como provedora de interoperabilidade e distribuição, será implementado um protótipo; o SISDEC - Sistema Distribuído de Estudo de Carga, responsável pela previsão, através de técnicas de inteligência artificial, da carga nodal de curto prazo do sistema elétrico de potência. A implementação dos modelos resultantes é apresentada assim como os resultados dos testes. Enfatizando a disponibilidade aos vários domínios de gestão, das informações modeladas, validando a eficácia da modelagem e a adequação da arquitetura distribuída proposta. Não se visa aqui a apresentação ou definição de métricas específicas, sendo os resultados apresentados do tipo qualitativo.

O SISDEC se insere em uma arquitetura de informação corporativa e utilizará um middleware, originário de um projeto de P&D (Pesquisa e Desenvolvimento), como acesso à massa de dados históricos da carga nodal do sistema elétrico de potência, obtidos a partir de um sistema SCADA - Supervisory Control And Data Acquisition. O acesso corporativo às informações processadas pelo SISDEC é efetuado através da intranet da empresa.

ABSTRACT

The work presents the use of model CIM - Common Information Model with a boarding ODP - Open and distributed Processing in the definition of a distributed architecture. Effected modeling ODP-CIM search to validate formally the choice of the WBEM - Web Based Enterprise Management as architecture distributed responsible for the interoperation between heterogeneous systems in a company of electric energy. The contest of WBEM additionally includes modeling CIM of the information, in way that the resultant environment can be seen as a distributed collection of interrelated systems.

In order to verify the adequacy of the WBEM as supplier of the interoperation and distribution, an archetype will be implemented; the SISDEC - Distributed System of Load Study, responsible for the forecast through techniques of artificial intelligence, of the nodal load of short term of the electrical system of power. The implementation of the resultant models is presented, as well as the results of the tests focusing the availability to the some management dominion, of the shaped information, validating the effectiveness of the modeling and the adequacy of the proposed distributed architecture. The metric presentation or definition specific is not aimed here, as the presented results qualitative type.

The SISDEC inserts in an architecture of corporative information and it will use one middleware, originated of a R&D project (Research and Development), as access to a mass of nodal load historical data of the electrical system of power, gotten from a system SCADA - Supervisory Control and Date Acquisition. The corporative access to processed information from SISDEC is effected through the Intranet of the company.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE ABREVIATURAS E SIGLAS

RESUMO

ABSTRACT

1. INTRODUÇÃO.....	1
1.1 Apresentação.....	1
1.2 Objetivos.....	2
1.3 Motivação.....	2
1.4 Abrangência.....	8
1.5 Metodologia.....	9
1.6 Organização.....	10
2. FUNDAMENTAÇÃO CONCEITUAL.....	11
2.1 Modelo Comum de Informação (CIM).....	11
2.1.1 Esquema (Schema).....	12
2.1.2 Núcleo ou Modelo Central (Core Model).....	13
2.1.3 Modelo Comum (Common Model).....	15
2.1.4 Extensões (Extension Schema).....	15
2.1.5 Meta Esquema (Meta Schema).....	16
2.1.6 Implementações CIM.....	18
2.1.7 MOF – Managed Object Format.....	20
2.2 Modelo de Referência RM-ODP.....	21

2.2.1	Arquitetura do RM-ODP.....	22
2.2.2	Especificações de Ponto de Vista.....	22
2.3	Ambiente Corporativo do Sistema de Estudo da Carga.....	26
2.3.1	Sistema Elétrico de Potência.....	27
2.3.2	Curva de Carga.....	29
2.3.3	Sistema de Supervisão e Controle da CTEEP.....	31
2.3.3.1	Arquitetura Lógica.....	32
2.3.3.2	Arquitetura Física.....	33
2.3.4	Middleware CTEEP.....	34
2.4	Aplicação de Inteligência Artificial no Sistema de Estudo da Carga.....	36
2.4.1	Redes Neurais Artificiais.....	36
2.4.1.1	Base Biológica.....	36
2.4.1.2	Neurônio Artificial.....	37
2.4.1.3	Parâmetros de Redes Neurais.....	38
2.4.2	Mapa Auto Organizável de Kohonen.....	40
2.4.3	Perceptron de Múltiplas Camadas.....	41
2.4.4	Algoritmos Genéticos.....	42
2.4.5	Algoritmo Fuzzy-Isodata.....	43
3.	DEFINIÇÃO DA ARQUITETURA DO SISDEC.....	45
3.1	Arquitetura do SISDEC.....	46
3.1.1	Estrutura Básica.....	46
3.1.2	Modelagem RM-ODP.....	48
3.1.2.1	Ponto de Vista Empresa.....	49

3.1.2.2	Ponto de Vista Informação.....	56
3.1.2.3	Ponto de Vista Computação.....	61
3.1.2.4	Ponto de Vista Engenharia.....	66
3.1.2.5	Ponto de Vista Tecnologia.....	68
4.	IMPLEMENTAÇÃO.....	71
4.1	Introdução.....	71
4.2	Protótipo.....	72
4.2.1	Arquitetura do WMI.....	73
4.2.2	Módulos da Aplicação.....	74
4.2.3	Implementação SISDEC.....	76
4.2.4	Modelagem CIM.....	86
4.2.5	Experimento.....	90
5.	CONSIDERAÇÕES FINAIS.....	98
5.1	Conclusões.....	98
5.2	Contribuições Acadêmicas.....	99
5.3	Trabalhos Futuros.....	99
	ANEXOS.....	101
	Anexo A: WBEM - Web Based Enterprise Management.....	101
	Anexo B: Mapa Auto Organizável de Kohonen.....	105
	Anexo C: Algoritmos Genéticos.....	108
	Anexo D: Computação Fuzzy.....	114
	Anexo E: Casos de Uso do SISDEC.....	118
	REFÊRENCIAS BIBLIOGRÁFICAS.....	141

LISTA DE FIGURAS

FIGURA 2.1 Core Model.....	14
FIGURA 2.2 Meta Esquema.....	16
FIGURA 2.3 Modos de Troca de Informações CIM.....	18
FIGURA 2.4 Pontos de Vista RM-ODP.....	23
FIGURA 2.5 Contexto de Inserção SISDEC.....	27
FIGURA 2.6 Curva de Carga do Estado de São Paulo Em 25/10/2005.....	29
FIGURA 2.7 Estrutura Organizacional da CTEEP.....	31
FIGURA 2.8 Composição da Diretoria Técnica da CTEEP.....	32
FIGURA 2.9 Arquitetura Lógica do Sistema de Supervisão e Controle (SSC).....	33
FIGURA 2.10 Arquitetura do Middleware CTEEP.....	35
FIGURA 2.11 Neurônio Biológico.....	37
FIGURA 2.12 O Neurônio Artificial.....	38
FIGURA 2.13 Função Sigmóide.....	39
FIGURA 2.14 Mapa de Kohonen.....	40
FIGURA 2.15 Rede Neural Perceptron Múltiplas Camadas.....	41
FIGURA 2.16 Mínimos, Local e Global.....	42
FIGURA 3.1 Papel do Objetos Empresa Classificador, Treinador e Previsor.....	50
FIGURA 3.2 Papel do Objeto Empresa Gerenciador.....	50
FIGURA 3.3 Papel Objeto Empresa Usuário.....	51
FIGURA 3.4 Objetos Empresa e seus papeis.....	51
FIGURA 3.5 Modelo de Casos de Uso para o SISDEC.....	52

FIGURA 3.6 Caso de Uso UC01: Solicitar Carga Histórica.....	53
FIGURA 3.7 Caso de Uso UC02: Solicitar Perfil.....	54
FIGURA 3.8 Caso de Uso UC03: Classificar Perfil.....	54
FIGURA 3.9 Caso de Uso UC04: Treinar Rede.....	55
FIGURA 3.10 Caso de Uso UC05: Solicitar Rede Neural.....	55
FIGURA 3.11 Caso de Uso UC06: Solicitar Previsão.....	56
FIGURA 3.12 Pontos de Conformidade.....	57
FIGURA 3.13 Localização do SISDEC.....	58
FIGURA 3.14 Diagramas de Classes CIM do SISDEC.....	59
FIGURA 3.15 Esquemas Invariantes ODP.....	60
FIGURA 3.16 Esquemas Dinâmicos ODP.....	61
FIGURA 3.17 Caso de Uso UC01.1:Listar Relação de Carga.....	63
FIGURA 3.18 Requisito para o Caso de Uso Listar Relação de Carga.....	63
FIGURA 3.19 Caso de Uso UC01.2:Pesquisar Carga Histórica.....	64
FIGURA 3.20 Caso de Uso Pesquisar Carga Histórica.....	64
FIGURA 3.21 Caso de Uso UC01.3: Requisitar Carga Histórica.....	65
FIGURA 3.22 Caso de Uso Requisitar Carga Histórica.....	65
FIGURA 3.23 Caso de Uso UC01.4:Guardar Carga Histórica.....	66
FIGURA 3.24 Caso de Uso Guardar Carga Histórica.....	66
FIGURA 3.25 Objetos Engenharia.....	68
FIGURA 3.26 Ponto de Vista Tecnologia.....	69
FIGURA 4.1 Modelo de Objetos do SISDEC.....	75
FIGURA 4.2 Contexto Físico do SISDEC.....	77

FIGURA 4.3 Contexto Lógico do SISDEC.....	78
FIGURA 4.4 Diagrama de Classe Módulo Classificador.....	80
FIGURA 4.5 Diagrama de Classe Módulo Treinador.....	83
FIGURA 4.6 Diagrama de Classe Módulo Previsor.....	84
FIGURA 4.7 Configuração SISDEC para Testes.....	91
FIGURA 4.8 Ambiente de Testes.....	92
FIGURA 4.9 Tela de Usuário SISDEC – Funcionalidade Carga.....	93
FIGURA 4.10 Resultado do Experimento para o caso de uso UC01.....	94
FIGURA 4.11 Gráfico do Perfil obtido no caso de uso UC02.....	95
FIGURA 4.12 Resultado dos Testes.....	97
FIGURA A.1 Relacionamento entre as tecnologias padrão que compõem o WBEM..	101
FIGURA A.2 Arquitetura lógica de servidor WBEM.....	103
FIGURA B.1 Mapa de Kohonen.....	105
FIGURA C.1 Mínimos, Local e Global.....	109
FIGURA C.2 Cruzamento do Tipo de Um Ponto.....	111
FIGURA C.3 Cruzamento do Tipo de Dois Pontos.....	112
FIGURA C.4 Cruzamento Linear	112
FIGURA E.1 Modelo de Casos de Uso para o SISDEC.....	118
FIGURA E.2 Caso de Uso UC01: Solicitar Carga Histórica.....	119
FIGURA E.3 Caso de Uso UC01.1:Listar Relação de Carga.....	120
FIGURA E.4 Requisito para o Caso de Uso Listar Relação de Carga	120
FIGURA E.5 Caso de Uso UC01.2: Pesquisar Carga Histórica.....	121
FIGURA E.6 Requisito para o Caso de Uso Pesquisar Carga Histórica.....	121

FIGURA E.7 Caso de Uso UC01.3: Requisitar Carga Histórica.....	122
FIGURA E.8 Requisito para o Caso de Uso Requisitar Carga Histórica.....	122
FIGURA E.9 Caso de Uso UC01.4: Guardar Carga Histórica.....	123
FIGURA E.10 Requisito para o Caso de Uso Guardar Carga Histórica.....	123
FIGURA E.11 Caso de Uso UC02: Solicitar Perfil.....	124
FIGURA E.12 Caso de Uso UC02.1: Listar Relação Perfil.....	125
FIGURA E.13 Requisito para o Caso de Uso Listar Relação Perfil.....	125
FIGURA E.14 Caso de Uso UC02.2:Pesquisar Perfil.....	126
FIGURA E.15 Requisito para o Caso de Uso Pesquisar Perfil.....	126
FIGURA E.16 Caso de Uso UC03: Classificar Perfil.....	127
FIGURA E.17 Caso de Uso UC03.1: Informar Dias.....	127
FIGURA E.18 Requisito para o Caso de Uso Informar dias.....	128
FIGURA E.19 Caso de Uso UC03.2: Requisitar Classificação.....	128
FIGURA E.20 Requisito para o Caso de Uso Requisitar Classificação.....	129
FIGURA E.21 Caso de Uso UC04: Treinar Rede.....	129
FIGURA E.22 Caso de Uso UC04 – digrama de seqüência.....	130
FIGURA E.23 Requisito para o Caso de Uso Treinar Rede.....	130
FIGURA E.24 Caso de Uso UC05: Solicitar Rede Neural.....	131
FIGURA E.25 Caso de Uso UC05.1: Listar relação de redes.....	132
FIGURA E.26 Requisito para o Caso de Uso listar relação redes.....	132
FIGURA E.27 Caso de Uso UC05.2: Pesquisar Rede.....	133
FIGURA E.28 Requisito para o Caso de Uso Pesquisar Rede.....	133
FIGURA E.29 Caso de Uso UC05.3: Requisitar Rede.....	134

FIGURA E.30 Requisito para o Caso de Uso Requisitar Rede.....	134
FIGURA E.31 Caso de Uso UC06: Solicitar Previsão.....	135
FIGURA E.32 Caso de Uso UC06.1: Listar relação previsão.....	136
FIGURA E.33 Requisito para o Caso de Uso Listar relação previsão.....	136
FIGURA E.34 Caso de Uso UC06.2: Pesquisar Previsão.....	137
FIGURA E.35 Requisito para o Caso de Uso Pesquisar Previsão.....	137
FIGURA E.36 Caso de Uso UC06.3: Requisitar Previsão.....	138
FIGURA E.37 Requisito para o Caso de Uso Requisitar Previsão.....	138
FIGURA E.38 Caso de Uso UC06.4: Fazer Previsão.....	139
FIGURA E.39 Requisito para o Caso de Uso Fazer Previsão.....	140

LISTA DE ABREVIATURAS E SIGLAS

API	Application Program Interface
ATL	Active Template Library
BROKER	Componente de software que, em ambientes distribuidos, facilita a comunicação entre os objetos, tipicamente entre servidor e cliente e/ou repositório e solicitante
CIM	Commom Information Model
CIMOM	CIM Object Manager
COS	Centro de Operação do Sistema de Transmissão da CTEEP
CRO	Centro Regional de Operação do Sistema de Transmissão da CTEEP
CROB	Centro Regional de Operação Bauru, da CTEEP
CROC	Centro Regional de Operação Cabreúva, da CTEEP
CROS	Centro Regional de Operação São Paulo, da CTEEP
CTEEP	Companhia de Transmissão de Energia Elétrica Paulista
DMTF	Distributed Management Task Force
DTD	Document Type Definition
HTTP	Hipertext transfer Protocol
IDL	Interface Definition Language
IEC	International Electro Technical Commission
ISO	International Organization for Standardization
ITU	International Telecommunication Union
MOF	Managed Object Format

P&D	Pesquisa e Desenvolvimento
RANGER	Scada utilizado nos Centros de Operação da CTEEP
RM-ODP	Reference Model for Open and Distributed Processing
SISDEC	Sistema Distribuído de Estudo da Carga
SCADA	Supervisory Control And Data Acquisition
SOM	Self Organizing Map
TCP/IP	Transport Control Protocol/Internet Protocol
UML	Unified Modeling language (linguagem de Modelagem Unificada)
XML	Extensible Markup Language
WBEM	Web Based Enterprise Management
WMI	Windows Management Instrumentation

1. INTRODUÇÃO

1.1 Apresentação

O processo de tomada de decisões nas empresas é fortemente determinado pela informação disponível. Toda informação tem que estar disponível, de uma maneira inteligível, quando e onde for necessária.

Atualmente os requisitos de negócios da indústria de energia elétrica demandam modelos complexos de dados, acesso rápido e compartilhamento da informação através do ambiente de negócios como um todo. Este dogma do mundo dos negócios implica na facilidade de troca de informações entre sistemas heterogêneos e a facilidade de acesso, de todas as partes interessadas, a estas informações [Anderson, 00]. A interoperabilidade constitui um requisito básico para a competitividade das empresas no mundo de negócios atual.

Adicionalmente, o uso crescente de equipamentos digitais, na supervisão e controle dos sistemas elétricos de potência, dá origem a uma grande quantidade de dados. Observa-se uma tendência de se obter cada vez mais dados e a necessidade de torná-los disponíveis onde forem necessários [McMorran, 04].

Observa-se que os diversos e isolados sistemas de software, que suportam a operação e a manutenção das redes de energia elétrica, se expandiram nos últimos anos e tendem a serem integrados em um sistema de informação corporativo [Nordström, 05], mas tal integração ainda apresenta desafios a serem vencidos. Observa-se que a tecnologia de informação ainda tem um longo caminho a percorrer no sentido de atender aos desafios surgidos devido às mudanças nas empresas de energia, quando da abertura do mercado [Nordström, 05].

De modo a atender a estas necessidades de interoperabilidade e distribuição, apresenta-se neste trabalho a proposta de utilização de uma arquitetura distribuída, aberta e padronizada, a WBEM.

1.2 Objetivos

O primeiro objetivo deste trabalho é, através da utilização do método de modelagem ODP-CIM apresentada por [Rosa, 04], se efetuar o estudo da viabilidade da utilização da WBEM como arquitetura proposta para prover interoperabilidade e distribuição, dentro do contexto corporativo de uma empresa de energia elétrica. Busca-se com este estudo a validação da escolha desta arquitetura através da definição de uma arquitetura distribuída na qual a WBEM pôde ser mapeada.

O segundo objetivo é o estudo, também dentro do contexto corporativo de uma empresa de energia elétrica, da viabilidade da modelagem ODP-CIM das informações geradas por sistemas aplicativos heterogêneos. Esta modelagem, através do ponto de Vista Informação ODP, tem a responsabilidade pela interoperabilidade (arquitetura de interface) tanto interna, entre os componentes distribuídos do SISDEC, quanto externa, enfocando os diversos clientes corporativos deste sistema e fontes de dados do sistema elétrico de potência.

O terceiro objetivo será a demonstração prática da metodologia apresentada, através da implementação, dentro da arquitetura proposta, de um sistema distribuído a cujos produtos, os clientes corporativos terão disponibilidade de acesso, através de interfaces WBEM e dados CIM.

Busca-se, então neste trabalho, a convergência de domínios de gestão através da conversão dos formatos dos dados e da interoperabilidade entre os vários módulos distribuídos constituintes do SISDEC, de modo que a consulta aos dados e o acesso às funcionalidades sejam feitos de forma transparente, independentemente do sistema usuário e da localização dos mesmos.

1.3 Motivação

Energia é um ingrediente essencial para a vida humana e suas múltiplas atividades, uma

vez que para se assegurar os padrões adequados de qualidade de vida nas residências e nas atividades de comércio e serviços, consome-se energia elétrica dentre outras. A demanda por energia apresenta crescimento contínuo devido ao crescimento da economia global, como também pelo aumento da própria população mundial. Deste modo, energia deve ser fornecida em quantidade e qualidade adequadas, onde e quando necessária.

Geralmente a produção de energia elétrica é feita longe dos centros consumidores, o que implica na necessidade de seu transporte e distribuição a estes consumidores. Esta necessidade é suprida através da implementação de um grande e complexo sistema de geração, transmissão e distribuição de energia elétrica, o Sistema Elétrico de Potência.

Os sistemas grandes e historicamente crescentes, tais como o Sistema Elétrico de Potência, consistem de um conjunto heterogêneo de equipamentos produtores de dados, que permitem a tomada de decisões [Anderson, 00]. Mas tomada de decisões envolve a posse de informações, o que por sua vez implica na necessidade de uma linguagem comum para a transferência (comunicação) das informações geradas em domínios distintos.

Por outro lado, o gerenciamento do sistema elétrico de potência não se resume apenas na sua supervisão e controle em tempo real, ele é a base para uma gestão completa de todo o negócio, integrando o seu planejamento e operação com a análise dos resultados obtidos, objetivando alcançar a eficiência e eficácia na sua operação. O planejamento, a análise e a operação dos sistemas elétricos de potência requerem a antecipação da evolução da carga elétrica nos mesmos [Senjyu, 02] [Swarup, 02].

O desequilíbrio produção/consumo de energia constitui um dos maiores problemas em sistemas elétricos de potência, e de modo a se evitar descontinuidades no fornecimento de energia devido à incapacidade de atendimento a demandas de carga é necessária uma estimativa precisa, tanto desta carga quanto das necessidades futuras de energia.

Este conhecimento prévio da carga elétrica permite às empresas de geração de energia elétrica gerenciarem recursos de modo a otimizarem a sua produção reduzindo seus custos de geração, melhorarem a coordenação da geração e a programação de intercâmbios (compra e venda de energia) internos e/ou com outras empresas de energia [Chow, 96] [Mara, 99]. Adicionalmente, as empresas de transmissão de energia elétrica também utilizam este conhecimento prévio para coordenarem o uso de seus equipamentos, na monitoração de sobrecargas e redução de custos de transmissão.

Uma vez que a carga é estimada geralmente como resultado de um processo de previsão que utiliza dados reais de processo elétrico, observa-se neste contexto que a disseminação dentro da empresa, de informações oriundas do sistema elétrico de potência, de um modo inteligível e independente dos sistemas nos quais foram geradas e processadas, constitui uma das necessidades primárias das empresas de energia elétrica [Becerra, 04].

Verifica-se a existência de trabalhos versando sobre as necessidades de padronização das informações geradas e consumidas por diferentes sistemas de informação e controle dentro do contexto de empresas de energia elétrica. A grande maioria está voltada para a utilização da modelagem CIM como ferramenta de compatibilidade e interoperação.

Em [Becker et al, 00] trata-se das necessidades, relativas a integração das aplicações existentes, mostrando que para as empresas se adaptarem as condições dinâmicas do ambiente de negócios é necessária uma infraestrutura flexível de tecnologia de informação e disponibilidade de informações. Ressalta ainda a necessidade de um modelo coerente e unificado de dados, uma vez que a existência de sistemas com diferentes tecnologias, resultando em ilhas de informação, causa impactos na eficiência da empresa como um todo. A agregação de informações e o fluxo de mensagens através da corporação são passos críticos na eficiência e capacidade corporativas. A utilização da modelagem CIM é considerada como facilitador para trocas de informação e interoperabilidade, com o seu repositório de metadados fornecendo toda a informação necessária para se localizar qualquer tipo de informação dentro da corporação. É citado

ainda o uso crescente do XML como formato de codificação de mensagens e a sua utilização como meio de integração entre aplicações. Também é considerada a necessidade da integração dos vários sistemas não interferir de modo negativo nas aplicações já existentes nos mesmos.

No trabalho de [Dong et al, 02] voltado para o contexto de manutenção e diagnóstico de equipamentos do sistema elétrico de potência, também trata das dificuldades de acesso e de troca de informações de manutenção geradas por distintas aplicações e armazenadas em distintas bases de dados. Levanta a questão de que o tradicional gerenciamento de informações não consegue atender a coleta e armazenamento de informação relativas à manutenção e diagnósticos dos equipamentos. Propõe o uso do CIM como mecanismo para troca de informações, assim como para a organização, acesso, compartilhamento e armazenamento dos dados, de modo que as informações coletadas por uma empresa possam ser facilmente compartilhadas tanto por aplicações quanto por outras empresas geograficamente distantes. Baseado na extensão do CIM propõe uma biblioteca digital comum para uso da comunidade de manutenção de equipamentos, baseada na Internet. Esta biblioteca coletará, em escala mundial, informações de equipamentos e de testes, práticas de manutenção e casos de falhas, simplificando o acesso ao estado da arte em diagnósticos e à informação de manutenção, proporcionando maior velocidade de extração de informação e de transação, integrando conhecimento e informação relativos a rotinas de manutenção e provendo o diagnóstico remoto.

A integração de um modelo de segurança com o ambiente de gerência WBEM é apresentado em [Carrião et al, 04], onde na integração de vários sistemas distribuídos e heterogêneos surgem dificuldades de interoperabilidade. O CIM é utilizado como modelo de interoperabilidade, fornecendo as definições e estruturas consistentes de dados, descrevendo as características comuns de gerenciamento e componentes do servidor WBEM. Ressalta ainda que o WBEM através de interfaces comuns a todos os sistemas simplifica e reduz os custos ao se integrar facilmente com os sistemas existentes na corporação.

Os requisitos de gerenciamento de servidores baseados em WBEM e sua arquitetura são apresentados no trabalho de [Lee et al, 04], onde se considera um ambiente de computação distribuído. Este ambiente se caracteriza pela necessidade da disponibilidade de serviços de rede independentemente da localização física e temporal. Os servidores de processamento recebem e processam grandes quantidades de informações provenientes de vários sensores, sendo também responsáveis pelo fornecimento de vários serviços. São apresentados os requisitos funcionais e não funcionais do ambiente computacional distribuído, e o atendimento do WBEM aos mesmos. A capacidade do CIM em descrever a informação de gerenciamento é ressaltada, citando os esquemas utilizados, as classes que modelam os serviços fornecidos, a base de dados modelando os dados dos sensores e as métricas CIM para QoS. O aspecto de segurança também é considerado com o uso do esquema User, observando-se o controle de acesso aos recursos. Foi utilizado como protocolo de gerenciamento o mapeamento CIM to XML, permitindo o encapsulamento sobre HTTP e CIM operations over HTTP, o qual possibilita às implementações CIM operarem de modo aberto e padronizado. A arquitetura apresentada é composta por servidores (clientes WBEM) e agentes (servidores WBEM) que trocam informações CIM no formato XML sobre HTTPS, observando-se que nesta arquitetura proposta, as comunicações se restringem a transferência de resultados finais de processamento.

A utilização do CIM no processo de análise de arquiteturas de sistemas de informação de empresas concessionárias de energia é apresentada por [Nordström, 05] resultando em um método direto de se criar documentos consistentes, relativos ao uso de dados e funcionalidades espalhados pelos diversos sistemas. Conforme os autores, o uso do CIM na modelagem apresentada facilitou a análise de distribuição das funcionalidades pelos diversos sistemas, assim como a análise dos riscos de duplicação de dados. Ressaltam ainda que o entendimento da interdependência entre os processos de negócios das concessionárias e os sistemas de informação que suportam estes processos é primordial para a análise da arquitetura dos sistemas de informação. E de modo a poderem modelar os processos de negócios em UML, utilizaram a extensão EPBE-Eriksson-Penker

Business Extensions, a qual inclui novos diagramas à UML. O método apresentado utiliza a extensão EPBE de classes CIM e os componentes abstratos correspondentes ao conjunto de funções de negócios definidos na série de padrões IEC-61968, consistindo de três fases; definição das atividades do processo de negócio, identificação das classes de dados e mapeamento das funcionalidades. O resultado consiste de modelos UML e listagem de funcionalidades, nos quais a duplicação de dados e/ou funcionalidades, assim como sobreposições podem ser identificadas e documentadas. Este método, ao usar as classes padronizadas no CIM como framework para a modelagem, torna-se aderente a padrões atuais existentes.

Observa-se que nos vários trabalhos a incidência maior é no uso específico da modelagem CIM na padronização da informação, não se concentrando na descrição detalhada da arquitetura onde a modelagem será utilizada.

Dentro deste contexto, um dos motivos do trabalho proposto foi validar a escolha de uma arquitetura para atender a necessidade tanto do acesso a informações oriundas de sistemas heterogêneos, quanto do conhecimento prévio da carga (demanda) de energia, como subsídios para o processo de tomada de decisões de negócio da empresa de energia elétrica.

O acompanhamento de um projeto de P&D (Pesquisa e Desenvolvimento), voltado para o uso de técnicas de inteligência artificial na previsão de carga elétrica, propiciou subsídios para se implementar a funcionalidade de previsão de carga provida pelo SISDEC.

No contexto acadêmico, a verificação da viabilidade da modelagem CIM dentro do ambiente corporativo de uma empresa de energia elétrica surgiu do contato, na Escola Politécnica da USP, com as tendências mundiais na área de Tecnologia da Informação, ocorrido no período do curso de Tecnologia de Informação (MTI). Academicamente, o conhecimento aprendido durante o processo da modelagem e da implementação fornecerá subsídios aos estudos de aplicabilidade do padrão de modelagem CIM aos

processos de gerenciamento do sistema elétrico de potência, assim como na utilização da WBEM como provedora de distribuição e interoperabilidade dentro do ambiente corporativo.

1.4. Abrangência

A abrangência deste trabalho se restringe aos tópicos:

Validação, pela aplicação formal do modelo CIM com uma abordagem ODP, de uma arquitetura distribuída proposta, que provê a interoperabilidade entre as várias aplicações (domínios) constituintes do Sistema Distribuído de Estudo da Carga, e os sistemas corporativos, assim como a distribuição das suas funcionalidades pelo parque de microcomputadores constituintes da rede corporativa.

Modelagem CIM, das diversas instâncias das informações (produtos) geradas pelo Sistema Distribuído de Estudo da Carga.

Estudo da aplicabilidade da modelagem CIM e do uso da WBEM dentro do contexto de uma empresa de energia elétrica.

O Sistema Distribuído de Estudo da Carga, aqui proposto foi experimentado com uma massa de dados históricos, referentes a leituras de carga nodal do Sistema de Transmissão da CTEEP, relativas a um período de quatro anos (2002 a 2005).

Não se considerou, apesar de grande número de bibliografias assim o fazerem, o uso de variáveis climáticas na modelagem, implementação e testes, uma vez que o Sistema de Supervisão e Controle aqui utilizado, por não possuir medição destas variáveis, não disponibiliza o histórico das mesmas. Por outro lado, a inclusão destas informações posteriormente é facilitada devido à natureza orientada a objeto da modelagem CIM e a natureza modular da arquitetura proposta.

1.5. Metodologia

Para se atingir os objetivos deste trabalho, a metodologia aplicada compõe-se das seguintes fases:

- Pesquisa bibliográfica
- Aplicação formal da modelagem CIM-ODP
- Proposição da arquitetura
- Implementação do protótipo
- Avaliação da implementação

A pesquisa bibliográfica corresponde ao estudo das referências na área de previsão de carga, de métodos de classificação de dados, conceitos de arquitetura de software, técnicas de inteligência artificial, conceitos de padronização ODP e modelagem CIM.

A aplicação formal da modelagem CIM-ODP segue a metodologia definida em [Rosa, 04] aplicada na modelagem da aplicação SISDEC, com ênfase nos pontos de vista Engenharia, Tecnologia e Informação.

Na implementação do protótipo busca-se a validação da modelagem CIM das informações processadas pelo SISDEC.

Como avaliação da implementação efetuam-se os testes referentes à interoperabilidade e distribuição dos módulos distribuídos constituintes do SISDEC, através de caso de uso.

1.6. Organização

No primeiro Capítulo é feita a apresentação, relacionando a motivação e os objetivos, assim como a abrangência do trabalho desenvolvido.

O Capítulo 2 apresenta os conceitos fundamentais ao desenvolvimento do trabalho, descrevendo o Modelo Comum de Informação CIM, o modelo de referência RM-ODP, o middleware CTEEP, o Sistema Elétrico de Potência, o conceito de Curva de Carga, suas características e importância, assim como a descrição da arquitetura física e lógica do Sistema de Supervisão e Controle da CTEEP, assim como as técnicas de inteligência artificial utilizadas.

A modelagem, utilizada na definição da arquitetura proposta é apresentada no Capítulo 3, onde se detalha a aplicação do modelo CIM na abordagem ODP de definição desta arquitetura.

No Capítulo 4 trata-se da implementação destes modelos dentro do ambiente corporativo. Assim como dos testes efetuados utilizando-se um caso de uso.

As conclusões do trabalho desenvolvido são focalizadas no Capítulo 5, apontando-se adicionalmente futuros trabalhos a serem desenvolvidos, referentes à modelagem e metodologia aqui adotadas.

A bibliografia se encontra relacionada no último capítulo, o qual encerra o trabalho proposto.

2. FUNDAMENTAÇÃO CONCEITUAL

2.1 Modelo Comum de Informação

O objetivo deste capítulo é apresentar os conceitos fundamentais ao desenvolvimento do trabalho proposto. Apresenta-se o modelo comum de informação – CIM, o modelo de referência RM-ODP, a caracterização de um sistema elétrico de potência, a importância da curva de carga, assim como os conceitos de inteligência artificial, utilizados na implementação das funcionalidades da aplicação, além do middleware responsável por prover os dados a partir do sistema de supervisão e controle, o qual também é apresentado.

O Modelo Comum de Informação (CIM – Common Information Model) constitui a fundação central da iniciativa da DMTF (Distributed Management Task Force) no campo do gerenciamento corporativo baseado em WEB. É um modelo conceitual de informação que descreve entidades computacionais e de negócios dentro de ambientes corporativos e na Internet, com o enfoque conceitual buscando a redução da complexidade global dos sistemas, tornando-os mais gerenciáveis e adaptáveis aos requisitos dos negócios, sempre em alteração.

O esquema CIM fornece definições para servidores, desktops, periféricos, sistemas operacionais, aplicações, componentes de redes, usuários e outros, juntamente com os seus detalhes individuais. Define, ainda, a informação disponível para compartilhamento pelos vários processos corporativos e pelas várias aplicações que os suportam.

Uma das funções principais oferecidas pelo CIM é a habilidade para se definir associações entre componentes. O seu enfoque orientado a objeto facilita o entendimento das relações e interdependências dos objetos gerenciados, enquanto que sua natureza hierárquica baseada em herança (de classes) facilita sua extensão para atender e englobar as mais diferentes características dos vários sistemas e equipamentos que se deseje gerenciar.

Estas definições comuns são neutras em relação aos processos que produzem e utilizam as informações, às aplicações que acessam os dados que expressam estas informações e às tecnologias nas quais estas aplicações são implementadas.

Em uma arquitetura baseada em CIM, as aplicações mapeiam seus dados no modelo CIM e efetuam interface apenas com o middleware que implementa este modelo, não se preocupando com as demais aplicações.

Com isto, não somente se reduz o número de interfaces a serem construídas e mantidas, como também se propicia a base para a integração de aplicações ao reduzir o acoplamento entre elas. Atualizações ou mesmo substituições de aplicações podem ser efetuadas com mínimo impacto funcional nas demais aplicações.

Com o modelo CIM pode-se, então, isolar os detalhes de interoperabilidade entre aplicações, permitindo que cada uma delas possa evoluir e assim mesmo continuarem a se comunicar.

O CIM é composto por uma Meta Esquema e por Esquemas Padrões. O Meta Esquema corresponde a definição formal do modelo, definindo os termos que serão utilizados para expressar o modelo, incluindo a maneira apropriada de se usar estes termos e sua semântica, enquanto o Esquema Padrão corresponde a um conjunto padrão de classes com suas propriedades e associações.

2.1.1 Esquema (Schema)

Um Esquema representa um conjunto de modelos de dados que descrevem um conjunto de objetos a serem gerenciados.

Constituem os blocos construtivos para as aplicações e plataformas de gerenciamento, tais como configuração de dispositivos, gerenciamento de performance e gerenciamento de mudanças.

O Esquema CIM consiste dos modelos Modelo Central, Modelo Comum e Extensão.

2.1.2 Núcleo ou Modelo Central (Core Model)

É o modelo de informação que captura as noções que são aplicáveis para todas as áreas de gerenciamento. Constitui um pequeno conjunto de classes, associações e propriedades que propiciam o vocabulário básico para analisar e descrever sistemas gerenciados.

O Core Model estabelece a classificação básica dos elementos e associações do ambiente gerenciado [DMTF, 03b]. A hierarquia de classes se inicia com a classe abstrata Managed Element, da qual derivam as subclasses: Managed System Element, Product, setting e Configuration, Collection e Statistical Information.

A partir destas classes do Core Model, o modelo se expande em muitas direções, endereçando vários domínios de problema a relacionamentos entre entidades gerenciadas.

Os objetos no Core Model são definidos e associados conforme a figura 2.1, onde por facilidade de visualização, nem todas as classes foram mostradas. Os termos foram mantidos em inglês, uma vez que todas as referências aos mesmos dentro do contexto CIM são efetuadas neste idioma.

Observando esta figura, temos:

Classe Managed Element:	é a raiz da hierarquia de objetos CIM e atua como uma referência para as associações que se aplicam a todas as entidades na hierarquia.
Managed System Elements:	representam sistemas, componentes de sistemas, quaisquer tipos de serviços (funcionalidades), software e redes. A definição de Sistema no contexto CIM é muito ampla, variando desde sistemas de computadores e dispositivos dedicados, até sistemas de aplicação e domínios de redes.
Classes Logical e Physical Elements:	são subclasses da classe Managed System Element.

Figura 2.1 Core Model, adaptado de [DMTF, 03b]

- Classe Product:** representa contratos entre vendedores e consumidores, captura a informação sobre o produto (Product) adquirido, como ele é suportado e onde está instalado.
- Settings:** define dados de parametrização específicos a serem aplicados para um ou mais Managed System Elements.
- Classe Statistical Information:** é uma superclasse abstrata para qualquer tipo de dados estatísticos de um Managed Element.
- Collections:** representam “caixas” arbitrárias que agrupam Managed Elements. Membership pode ser descrito pela definição de classe e/ou indicado por uma instanciação explícita de uma associação Member Of Collection.

Associações Component:	estabelecem um relacionamento “parte de” entre Managed System Elements.
Associações Dependency:	descrevem dependências funcionais (um objeto não pode funcionar sem o outro) ou dependências existenciais (um objeto não pode existir sem o outro).

2.1.3 Modelo Comum (Common Model)

Modelo que captura as noções que são comuns a áreas particulares de gerenciamento, mas independentes de uma particular plataforma, tecnologia ou implementação. Estas áreas comuns são sistemas, aplicações, redes e dispositivos.

A área Sistemas se refere a dispositivos tais como computadores, hubs ou roteadores; Aplicações, modela as informações que descrevem os softwares aplicativos, assim como as que são utilizadas pelos mesmos; Redes, representa os vários modelos de rede incluindo topologias, conectividade e acesso fornecidos pelos protocolos e serviços necessários para prover acesso de rede; e Dispositivos, que contém as classes que representam os dispositivos que integram os componentes físicos do sistema.

Um modelo de informação é específico o suficiente para fornecer a base para o desenvolvimento da aplicação, fornecendo um conjunto de classes básicas para extensão em uma área de tecnologia específica.

2.1.4 Extensões (Extension Schema)

São extensões, do modelo comum, específicas da tecnologia considerada. Uma vez que os esquemas são por definição extensíveis, os fornecedores de tecnologia podem criar esquemas personalizados para descrever seus produtos.

São constituídos de classes representando objetos gerenciados específicos de uma tecnologia, que são adicionadas ao modelo comum. Estas classes são tipicamente

utilizadas para ambientes operacionais específicos tais como UNIX ou Microsoft Windows.

2.1.5 Meta-Esquemas (Meta-Schemas)

Especifica formalmente o modelo, definindo os termos usados para expressar, tanto o próprio modelo, quanto o seu uso e semântica. A Linguagem de Modelagem Unificada (UML) [Booch, 01] é utilizada para definir a estrutura de um meta-esquema (figura 2.2).

Figura 2.2 Meta Esquema [DMTF, 03b]

Os elementos constituintes de um Meta-Esquema são:

- **Esquema:** grupo de classes com um único proprietário. Os esquemas são usados para administração e nomeação de classes.
- **Classe:** coleção de instâncias que suportam o mesmo tipo (mesmos métodos e propriedades). As classes podem ser arranjadas em uma generalização hierárquica que represente sub tipos de relacionamentos entre classes. Classes têm métodos, os quais representam o comportamento relevante da classe. E

podem participar em associações como sendo o alvo (objeto) de referência de propriedade da associação. Classes também podem ter instâncias.

- **Propriedade:** valor usado para caracterizar instâncias de uma classe. A propriedade pode ser considerada como um par de funções GET e SET que, quando aplicadas em um objeto, retorna e força (determina) um estado, respectivamente.
- **Método:** declaração de uma assinatura (nome do método, tipo de retorno e parâmetros), e, no caso de uma classe concreta, pode implicar em uma implementação.
- **Trigger:** reconhecimento de uma variação de estado (tais como criação, atualização, acesso ou destruição) da instância de uma classe, e a atualização e acesso a uma propriedade.
- **Indicação:** objeto criado como resultado de um trigger. Uma vez que indicações são subtipos de classe, eles podem ter propriedades e métodos, e podem ser arranjados em uma hierarquia de tipo.
- **Associação:** classe que contém duas ou mais referências. Ela representa um relacionamento entre dois ou mais objetos. Pelo modo como as associações são definidas, é possível estabelecer-se o relacionamento entre classes sem afetar qualquer uma das classes relacionadas.
- **Referência:** define o papel que cada objeto exerce em uma associação. A referência representa o nome do papel de uma classe no contexto da associação. Associações suportam múltiplos

relacionamentos para um dado objeto, por exemplo, um sistema pode ser relacionado a muitos componentes.

- **Qualificador:** usado para caracterizar os elementos nomeados, por exemplo, a definição das características de uma propriedade ou a chave de uma classe. Os qualificadores propiciam um mecanismo que permite que um meta esquema seja extensível de um modo limitado e controlado.

2.1.6 Implementações CIM

Por ser um modelo conceitual independente de uma particular implementação, o CIM permite a troca de informação em uma variedade de modos. A figura 2.3 ilustra quatro destes modos, os quais podem ser utilizados de forma combinada pelas aplicações.

Figura 2.3 Modos de troca de informações CIM

Modo Repositório - armazena meta modelos de informação para acesso por programas

- As estruturas (“constructs”) definidas no modelo são armazenadas em uma base

de dados. Estas estruturas não constituem instâncias de objetos ou relacionamentos entre objetos, mas sim definições que podem ser utilizadas para estabelecer objetos e relacionamentos.

- O meta modelo usado pelo CIM é armazenado no repositório, que se torna a representação do meta modelo. Isto é obtido pelo mapeamento do meta modelo dos “constructs” em um esquema físico do repositório alvo. Após este mapeamento, preenche-se o repositório com as classes e propriedades expressas no Core Model, Common Model e Extension Schemas.

Aplicações de Sistemas de Gerenciamento de Base de Dados (DBMS) – transforma a definição conceitual em esquema físico tendo em vista uma tecnologia de DBMS em particular.

- O CIM é mapeado no esquema físico de um DBMS alvo (por exemplo, relacional). A informação armazenada na base de dados consiste de instâncias reais dos constructs. As aplicações podem trocar informação quando elas têm acesso a um DBMS comum e o mapeamento ocorre de modo previsível.

Objetos de aplicação – define um conjunto de objetos de dados orientados para a aplicação, que podem ser instanciados e estendidos na tecnologia desejada.

- O CIM é usado para criar um conjunto de objetos de aplicação em uma linguagem em particular. As aplicações podem trocar informações quando elas podem ligar-se aos objetos de aplicação.

Parâmetros de Troca – conteúdo do CIM é usado para estruturar instâncias passadas entre aplicações

- Neste modo, o CIM é uma forma neutra usada para troca de informações por meio de um conjunto padrão de objetos API. A troca de informação pode ser efetuada via chamada direta ao conjunto de API, ou pela troca orientada de API,

as quais podem criar o objeto apropriado, na tecnologia local implementada.

2.1.7 MOF – Managed Object Format

A informação gerenciada é descrita em uma linguagem de definição de interface (IDL – Interface Definition Language) denominada MOF - Managed Object Format.

A especificação MOF será aqui utilizada para se referir a uma coleção de informações descrita em conformidade com a sintaxe MOF.

A sintaxe MOF é uma maneira de descrever definições de objetos em forma textual [DMTF, 99], sendo seus componentes principais, as descrições textuais de declarações de classes, associações, propriedades, referências, métodos e instâncias, assim como seus qualificadores.

As descrições feitas em uma especificação MOF podem ser validadas dentro de um contexto definido (namespace).

Ao se introduzir uma especificação MOF, em uma implementação, dentro de um determinado contexto, se obtém uma validação tanto da correção da sintaxe desta especificação quanto da correção da sua semântica, dentro da implementação em particular.

A especificação MOF, também pode ser verificada quanto à correção de sua sintaxe utilizando-se um compilador MOF.

O código MOF provê diversas vantagens em relação aos métodos tradicionais [DMTF, 05], tais como:

Eficiente: A quantidade de código para se definir uma classe CIM em MOF é muito menor do que a necessária para definir a mesma classe em C++, reduzindo o tempo necessário para desenvolvimento.

Inteligível: O código MOF é mais fácil de ser lido e entendido, reduzindo futuros custos de manutenção.

Re-usável: Uma vez que o arquivo MOF não é parte direta da aplicação, o projetista pode criar e compartilhar arquivos MOF descrevendo uma tecnologia específica, facilitando a adoção de novas tecnologias pela corporação.

MOF possui componentes que são semelhantes aos encontrados em linguagens de programação, tais como C. A referência [DMTF, 99] pode ser consultada para uma visão completa e mais detalhada dos componentes MOF.

2.2 Modelo de Referência RM-ODP

Um modelo de referência consiste na decomposição padronizada do problema em partes conhecidas que cooperam entre si em prol de uma solução, originando-se do processo de amadurecimento da solução em função da necessidade de representações mais abstratas que caracterizam o domínio em estudo [Varoto, 02].

O modelo de referência para processamento distribuído e aberto (RM-ODP) é um padrão desenvolvido pela ISO e ITU-T, que define uma estrutura (framework) arquitetural que provê o suporte para a integração da distribuição, interoperabilidade e portabilidade [ISO, 98a].

Ele é baseado em conceitos precisos, derivados dos processos atuais de desenvolvimento de sistemas de processamento distribuído, e tanto quanto possível, no uso de técnicas de descrição formais para especificação de arquiteturas.

Podem ser ressaltadas as características do RM-ODP, de sintaxe de representação independente e linguagens de implementação, assim como os seus pontos de vistas arquiteturais. Estes cinco pontos de vistas arquiteturais, que englobam todo o contexto de interoperabilidade, ao considerarem desde as políticas e procedimentos até as

soluções de engenharia são: empresa, informação, computação, engenharia e tecnologia. Não há uma estruturação em camadas destes pontos de vistas, uma vez que constituem diferentes abstrações de um mesmo sistema construídas, considerando-se contextos específicos [Egyhazy, 04].

O objetivo do RM-ODP é o desenvolvimento de padrões que permitam que os benefícios dos serviços de processamento de informação distribuído possam ser obtidos em um ambiente de recursos de Tecnologia de Informação heterogêneos e múltiplos domínios organizacionais.

2.2.1 Arquitetura do RM-ODP

A padronização OPD possui quatro elementos fundamentais:

- enfoque de modelagem orientada a objeto, para especificação de sistemas
- especificação de um sistema em termos de especificações de pontos de vista separados, mas inter-relacionados.
- definição de uma infraestrutura que provê transparências de distribuição para aplicações de sistema
- estrutura para avaliação da aderência do sistema ao padrão

2.2.2 Especificações de Ponto de Vista

Um ponto de vista é uma abstração que resulta da divisão da especificação do sistema como um todo, em pontos de vista relevantes a uma particular área de interesse, durante o processo de projeto de um sistema.

Figura 2.4 Pontos de Vista do RM-ODP [Varoto, 02]

Ponto de Vista da Empresa

Este ponto de vista descreve o sistema de informações em termos de o que o mesmo deve fazer, capturando as necessidades e especificações administrativas e técnicas que guiam e justificam o projeto do sistema.

Trata dos objetivos, escopo e políticas que governam as atividades de um sistema especificado, na organização da qual ele faz parte. Define as políticas sociais e organizacionais em termos de objetos passivos e ativos, produzindo uma especificação que expressa as responsabilidades de todos os participantes do sistema distribuído.

É uma visão puramente funcional da arquitetura, definindo os objetivos gerais e o escopo do projeto, as entidades dentro do domínio do negócio e as regras e políticas que governam a interação entre estas várias entidades [Egyhazy, 04].

Ponto de Vista da Informação

Descreve o sistema de informações em termos de estruturas de informações, fluxo de informações e restrições relacionadas com a manipulação das mesmas.

Trata com os vários tipos de informação manipulada pelo sistema e restrições no uso e interpretação desta informação. Este ponto de vista captura a informação trocada entre as aplicações distribuídas em termos de metadados, estados e estrutura de um objeto.

Tem, portanto, como foco a semântica das informações, as classes de definição de objetos informação, esquemas invariante, estático e dinâmico e as regras associadas [Ferreira, 01].

A informação, neste ponto de vista, pode ser estática, por exemplo uma particular instância de um objeto; dinâmica, descrevendo possíveis alterações de estado e/ou estrutura de um objeto; ou invariante, descrevendo propriedades, de um objeto, que não devem ser alteradas.

Ponto de Vista da Computação

O sistema de informações, neste ponto de vista, é descrito em termos de operações e características computacionais do processo de mudança de informações, preocupando-se com a distribuição dos objetos dentro do sistema, mas não com sua infra-estrutura de comunicação.

Trata da decomposição funcional do sistema em um conjunto de objetos que interagem através de suas interfaces, habilitando a distribuição do sistema [Ferreira, 01]. Descreve a funcionalidade das aplicações ODP, de maneira transparente em relação à distribuição das mesmas, em termos de uma ou mais interfaces que os objetos oferecem para isolar seus dados privados e seus métodos (processos). A interface é, dentro da perspectiva de interoperabilidade, a chave para se entender os componentes [Egyhazy, 04].

Ponto de Vista da Engenharia

Descreve o sistema de informações em termos de recursos de engenharia para

suportar a natureza distribuída do processamento, preocupando-se com a maneira pela qual a interação entre objetos ocorre e os recursos necessários para que isso se torne possível.

Trata da infra-estrutura requerida para suportar a distribuição do sistema, considerando-se heterogeneidade de hardware e software. Descreve os elementos das aplicações distribuídas que suportam a distribuição do software e a infraestrutura geral. Define o conjunto de objetos que suportam os canais de comunicação, assim como a criação das estruturas de engenharia, tais como topologia da rede, padrões de comunicação e a infra-estrutura de segurança.

Ponto de Vista da Tecnologia

O sistema de informações é descrito, por este ponto de vista, em termos dos componentes que compõem o sistema em construção.

Trata da escolha da tecnologia para suportar a distribuição do sistema. Considera os elementos específicos de hardware e software da aplicação distribuída a ser implementada.

O RM-ODP não fornece muita orientação ou controle sobre este ponto de vista, uma vez que tenta se manter, o mais possível, tecnologicamente independente.

O RM-ODP não determina como o sistema de informações deva ser projetado e implementado, assim como não define quais as ferramentas ou técnicas devam ser utilizadas para a representação de cada um dos pontos de vista. Torna-se portanto necessária, além do RM-ODP, a aplicação de métodos formais que possam guiar o projeto e implementação de sistemas de software através de cada um dos pontos de vista.

Para se representar um sistema, em um determinado ponto de vista, torna-se necessário a definição de um conjunto estruturado de conceitos, em termos dos quais a especificação

do sistema possa ser expressa. Para cada ponto de vista existe uma linguagem de ponto de vista associada, que pode ser usada para expressar a especificação do sistema para aquele ponto de vista.

Os conceitos de modelagem de objetos fornecem uma base comum para as linguagens de ponto de vista e tornam possíveis, tanto a identificação dos relacionamentos entre as diferentes especificações de pontos de vistas, quanto verificação de correspondências entre as representações do sistema em diferentes pontos de vistas.

A UML, por definir um conjunto básico de diagramas e notações que permitem representar as múltiplas perspectivas (estruturais / estáticas e comportamentais / dinâmicas) do sistema sobre análise e desenvolvimento, é utilizada para a modelagem do sistema em seus vários pontos de vista.

2.3 Ambiente Corporativo do Sistema Distribuído de Estudo da Carga

O ambiente corporativo, onde se insere o SISDEC, pode ser visualizado através da ferramenta Sistema Aberto de Automação [Becerra, 98], a qual divide em níveis de informação, a estrutura gerencial de uma empresa. A localização do SISDEC, dentro deste contexto onde se concentrará a modelagem RM-ODP pode ser visualizada na figura 2.5, adaptada de [Rosa, 04].

Na camada de gestão estão localizados os sistemas que lidam com grande volume de informação, visando o planejamento da operação e projetos entre outros [Rosa, 04]. O SISDEC está localizado nesta camada, e utilizará dados oriundos da primeira camada (processo elétrico), mas transferidos para a segunda camada pelo middleware descrito em [Becerra, 04].

O Sistema de Supervisão e Controle está localizado na camada de controle, utilizando as informações em tempo real do processo elétrico, obtidas na primeira camada, para propiciar as tarefas de supervisão e controle do Sistema Elétrico de Potência.

Os canais de comunicação definem o meio comum de comunicação, definindo as regras de envio de informações, neste caso específico, através de servidores CIM.

Figura 2.5 Contexto de inserção SISDEC

2.3.1 Sistema Elétrico de Potência

O Sistema Elétrico de Potência Brasileiro é constituído por empresas federais, estaduais e privadas, cuja operação e expansão são efetuadas através de um planejamento centralizado [Filho, 01].

O objetivo de um Sistema Elétrico de Potência é gerar energia elétrica em quantidade e qualidade adequadas, transmiti-la e distribuí-la aos consumidores individuais, em forma e qualidade apropriadas, com eficiência econômica e um baixo impacto ambiental [Filho, 01].

O Sistema Elétrico de Potência pode ser caracterizado, de uma forma global, por três elementos básicos: geração, transmissão e distribuição [Dabbaghchi, 97].

O elemento Geração, constituído de equipamentos produtores de energia elétrica em escala industrial (usinas hidrelétricas, predominantemente) disponibiliza, através das linhas de Transmissão e subestações, a energia gerada, aos equipamentos de Distribuição (linhas de distribuição e subestações), os quais a distribuem aos consumidores finais [Dabbaghchi, 97].

O nível de qualidade do produto final (energia elétrica) implica em escalas de investimentos distintas para cada um destes elementos. De forma que a exploração econômica e confiável do Sistema Elétrico de Potência depende do grau de integração e coerência da gestão destas três atividades [Filho, 01].

Um Sistema de Potência apresentando uma geração apta a atender a demanda dos consumidores, tanto em condições normais quanto em situações emergenciais, é considerado um Sistema Seguro [Aboul, 01]. O processo de especificar quando um sistema é seguro ou inseguro chama-se “Security Assessment” (Avaliação de Segurança), enquanto que o conjunto de ações necessárias para restaurar a segurança de um sistema é denominado “Security Enhancement” (Otimização da Segurança) [Aboul, 01]. Tanto o “Security Assessment” quanto o “Security Enhancement” necessitam da previsão da carga de curto prazo [Aboul, 01].

Adicionalmente, a qualidade da previsão da carga contribui de forma significativa para a indicação mais precisa dos sinais de ordem econômica a partir dos estudos de planejamento energético da operação, além de permitir a determinação mais adequada da operação eletroenergética do Sistema e do dimensionamento da Rede Elétrica a partir dos estudos de ampliações e reforços e do planejamento da operação elétrica.

Portanto, a previsão da Carga de Curto Prazo é utilizada, pelas empresas de energia elétrica, para assegurarem eficiência energética e segurança da operação de um Sistema Elétrico de Potência [Piras, 96][Senjyu, 00].

2.3.2 Curva de Carga

A carga de um Sistema de Potência representa a demanda elétrica acumulada de todos os consumidores alimentados pelo mesmo, assim como os intercâmbios de energia com outros sistemas [Trudnowski, 01].

Os gráficos da carga, como uma função do tempo, são denominados Curvas de Carga (figura 2.6).

Figura 2.6 Curva de Carga do Estado de São Paulo em 15/12/2005

A utilização ótima dos geradores e subestações de energia é totalmente dependente da precisão na previsão da carga [Chow, 96] [Dabbaghchi, 97] [Tamimi, 00], sendo difícil a sua previsão, uma vez que o seu valor é baseado em decisões, individuais e distribuídas, de pessoas respondendo aos seus desejos de consumo de energia elétrica [Dabbaghchi, 97].

A caracterização desta carga, considerando-se o aspecto de curto prazo, possui várias nuances, uma vez que o período considerado na sua previsão varia de minutos a um dia [Aboul, 01], de um dia a uma semana [Marin, 02], de uma hora a uma semana e outros.

Na literatura, vários métodos tais como predição de séries temporais, regressão múltipla, interpolação exponencial, filtro de Kalman e outros, tem sido propostos para a previsão de carga de curto prazo [Tamimi, 00] [Trudnowsi, 01] [Piras, 96].

As curvas de carga do Sistema Elétrico de Potência apresentam similaridades entre dias equivalentes na semana, por exemplo, a carga da quarta feira desta semana é similar a da quarta feira da semana passada, assim como é similar a quarta feira de semana retrasada.

Os dias considerados especiais como, por exemplo, os feriados, representam uma exceção a esta regra, apresentando uma curva de carga atípica [Aboul, 01], e apesar de terem grande impacto no perfil da carga, são conhecidos a priori, de modo que providências, visando a diminuição de sua influência nos processos de previsão de carga, podem ser adotadas.

Comprova-se que as condições climáticas e o perfil da carga nos dias anteriores recentes apresentam uma grande influência na previsão desta carga de curto prazo, enquanto que os fatores socioeconômicos tendem a influenciar a previsão da carga de longo prazo (meses à frente) [Huang, 01].

Os fatores que mais influenciam o perfil da carga de curto prazo são as atividades sociais humanas e as variáveis ambientais, tais como temperatura, umidade relativa, velocidade de ventos e cobertura de nuvens [Trudnowsi, 01], sendo que o impacto destas variáveis climáticas na demanda de energia elétrica é complexo, altamente não linear e de difícil caracterização [Rashid, 96] [Senjyu, 00].

Existem trabalhos que utilizam conjuntamente vários métodos de previsão, tais como redes neurais, lógica fuzzy, séries temporais e regressão múltipla, dependendo do tipo de

dia considerado (fins de semana, feriados, dias especiais, dias normais ou dias de verão) [Tamimi, 00] [Kim, 00].

2.3.3 Sistema de Supervisão e Controle da CTEEP

A Companhia de Transmissão de Energia Elétrica Paulista – CTEEP (figura 2.7) é oriunda da cisão dos ativos de transmissão de energia elétrica da CESP e da Eletropaulo, ocorrida durante a reestruturação do setor energético brasileiro em abril de 1999. O objetivo da companhia é operar, manter, expandir e explorar sistemas de transmissão de energia elétrica.

Figura 2.7 Estrutura Organizacional da CTEEP

A organização da Diretoria Técnica da CTEEP pode ser visualizada na figura 2.8, onde se destaca o Departamento de Operação - TO, responsável pelas tarefas, dentre outras, de responder pela operação do sistema de transmissão em tempo real, de planejar a curto e médio prazo a operação elétrica do sistema, de efetuar estudos elétricos, de estabelecer critérios para operação em condições normais e de emergência e, de recomposição do sistema elétrico.

Figura 2.8 Composição da Diretoria Técnica da CTEEP

A CTEEP opera uma complexa infra-estrutura composta por mais de 11.780 quilômetros de linhas de transmissão que se estendem por todo o Estado de São Paulo, ultrapassando 18.266 quilômetros de circuitos e 102 subestações operadas pela empresa, com capacidade de transformação acima de 38.500 MVA.

A operação deste Sistema Elétrico de Potência efetuada pela CTEEP é coordenada a partir de um Centro de Operação do Sistema - COS, localizado na S/E de Bom Jardim, em Jundiaí, e de três Centros Regionais de Operação - CRO, localizados em Bauru - CROB, Cabreúva - CROC e São Paulo - CROS.

Estes centros, interligados por redes de computadores, têm a responsabilidade de coordenar, supervisionar e controlar as operações realizadas na malha elétrica de transmissão, incluindo ainda no caso do COS como centro contratado pelo Operador Nacional do Sistema Elétrico - ONS, a coordenação, supervisão e controle da geração das usinas localizadas no Estado de São Paulo. Esta atuação integrada resulta em altíssimos níveis de confiabilidade e qualidade para as tarefas de coordenação, supervisão e controle em tempo real do sistema elétrico de potência.

Toda esta operação é monitorada por um sistema integrado de coordenação, supervisão e controle, o SSC – Sistema de Supervisão e Controle.

2.3.3.1 Arquitetura Lógica

A arquitetura do Sistema de Supervisão e Controle apresenta uma concepção distribuída (figura 2.9), baseada no modelo cliente-servidor, com os servidores de aplicação

instalados no COS e os servidores de comunicação nos CROs.

Cabem aos servidores de comunicação as tarefas de varredura dos equipamentos de campo (Unidade Terminais Remotas - UTR), envio dos dados coletados para os servidores de aplicação e recebimento dos comandos a serem enviados aos equipamentos de campo.

Figura 2.9 Arquitetura Lógica do Sistema de Supervisão e Controle (SSC)

2.3.3.2 Arquitetura Física

O COS é composto por cinco redes locais a 100 Mbit/s, sendo duas redes redundantes reservadas para operação, duas redundantes para o sistema historiador e uma rede local para tarefas de suporte.

Nas redes de operação estão conectadas as consoles de operação do sistema (transmissão, supervisor e geração), de manutenção/desenvolvimento, de pós-operação, de tempo real e os servidores de aplicação, enquanto que nas redes do Sistema Historiador, estão conectados os servidores duais do Historiador. Na rede de suporte

estão conectados microcomputadores PC compatíveis, executando aplicativos que fornecem os dados históricos em formato gráficos, planilhas e texto.

Os servidores de aplicação processam as informações providas pelos servidores de comunicação, obtidas dos equipamentos de campo e fornecem os resultados e atuações necessárias aos Centros Regionais de Operação - CRO, através das redes de área expandidas dualizadas, utilizando roteadores redundantes.

Os CROs apresentam configuração composta de três redes locais, sendo duas redundantes para a operação e uma para suporte. Nas redes de operação estão conectadas consoles e servidores duais de comunicação bidirecional com os vários equipamentos de campo, os quais utilizam diferentes protocolos e/ou velocidades de comunicação. Na rede de suporte existem microcomputadores PC compatíveis executando aplicativos de acesso aos servidores do Historiador.

O Departamento de Operação – TO possui duas redes locais a 100 Mbit/s, sendo uma reservada à operação. A outra rede local do TO, a de suporte, possui microcomputadores PC compatíveis destinados a estudos e acompanhamento do sistema, sendo locados à alta gerência: Gerência do Departamento de Operação, Diretoria Técnica, Assessoria da Diretoria Técnica, Presidência e Secretaria de Energia, entre outros usuários.

2.3.4 Middleware CTEEP

Middleware é um nome genérico usado para se referir a uma classe de software cujo objetivo é servir como “cola” entre sistemas construídos separadamente [Sanz, 00]. Com a utilização de um middleware, que propicia uma plataforma homogênea para a interoperabilidade, obtém-se uma redução na complexidade de implementação de clientes e servidores, independentemente de suas localizações.

O SISDEC necessita do acesso aos dados históricos da carga do Sistema Elétrico de Potência, os quais são coletados do SSC e tornados disponíveis no ambiente corporativo

pelo middleware CTEEP [Becerra, 04].

Este middleware está inserido numa arquitetura de sistema de informação corporativa, cuja estrutura é composta pelos elementos: nível operacional, nível estratégico e rede corporativa [Becerra, 04].

No nível operacional se localizam os sistemas onde se realizam as funções de supervisão, controle e aquisição. Os sistemas corporativos e os sistemas técnicos que são de apoio complementar ao processo principal da transmissão e de apoio a decisões estratégicas da corporação, estão localizados no nível estratégico.

A rede corporativa, constituída de uma infra-estrutura de comunicação integra todos os sistemas da CTEEP, mas apesar desta rede permitir a integração entre sistemas do nível operacional e do nível estratégico, não permite a interoperabilidade entre os mesmos.

Figura 2.10 Arquitetura do middleware CTEEP [Becerra, 04]

O middleware busca esta interoperabilidade entre os sistemas da CTEEP, e apresenta a arquitetura mostrada na figura 2.10.

A interoperabilidade obtida com o uso deste middleware, aberto e modularizado, permite a escalabilidade posterior do mesmo [Becerra, 04].

2.4 Aplicação de Inteligência Artificial no Sistema de Estudo da Carga

Inteligência Artificial é um ramo da Ciência da Computação que trata da emulação de processos mentais humanos através de modelos computacionais, focalizando-se a representação do conhecimento e métodos de inferência.

Um sistema de Inteligência Artificial não é capaz somente de armazenamento e manipulação de dados, mas também da aquisição, representação, e manipulação de conhecimento. Esta manipulação inclui a capacidade de deduzir ou inferir novos conhecimentos - novas relações sobre fatos e conceitos - a partir do conhecimento existente e utilizar métodos de representação e manipulação para resolver problemas complexos que são freqüentemente não quantitativos por natureza.

A Inteligência Artificial utilizada, neste trabalho para a previsão da carga do Sistema Elétrico de Potência é representada por redes neurais artificiais.

2.4.1 Redes Neurais Artificiais

São sistemas de processamento de dados, constituídos por um grande número de simples elementos de processamento, os neurônios, altamente interconectados dentro de uma estrutura inspirada na estrutura do córtex cerebral humano, com altíssimo grau de paralelismo. Constituem um método de se solucionar problemas, usando-se técnicas computacionais que apresentam um modelo inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento através da experiência [Haykin, 02].

2.4.1.1 Base Biológica

O cérebro humano constitui um sistema extremamente complexo, capaz de pensar, lembrar e resolver problemas.

O neurônio é a unidade celular fundamental do sistema nervoso cerebral. Ele é um simples elemento de processamento que recebe e combina sinais de outros neurônios através de caminhos de entrada denominados dentritos (figura 2.11).

Se o sinal de entrada combinado for forte o suficiente, o neurônio dispara, produzindo um sinal de saída através do axônio, que se conecta a dentritos de vários outros neurônios. Cada sinal entrando no neurônio, por um dos dentritos, passa através de uma junção sinaptica (sinapse), que é um espaço infinitesimal preenchido com fluido neurotransmissor que acelera ou retarda o fluxo de cargas elétricas.

Figura 2.11 Neurônio Biológico

O ajuste da impedância ou condutância da sinapse caracteriza um processo extremamente crítico, uma vez que o mesmo lida com aprendizado e memorização. O cérebro aprende e armazena informações conforme as sinapses são ajustadas [Haykin, 02].

2.4.1.2 Neurônio Artificial

Um neurônio artificial é um modelo cujos componentes possuem uma analogia direta com os componentes do neurônio biológico, no qual os sinais de entrada, ao contrário dos sinais biológicos, são variáveis contínuas e não pulsos elétricos discretos. Cada um dos sinais de entrada é modificado por um peso w_{in} , cuja função é análoga a da junção sinaptica do neurônio biológico (figura 2.12). Estes pesos podem ser positivos ou

negativos, de modo a acelerar ou inibir o fluxo de sinais elétricos.

O neurônio artificial efetua a agregação (soma, produto ou outras funções) das entradas ponderadas e a filtragem não linear (função de ativação f) do resultado desta agregação.

Figura 2.12 O neurônio Artificial

2.4.1.3 Parâmetros de redes Neurais

As redes neurais podem ser caracterizadas pelos parâmetros:

Número de Camadas

Uma rede neural é composta por uma coleção de neurônios interconectados, agrupados em camadas, que são classificadas em três categorias: camada de entrada, camadas intermediárias (ocultas) e camada de saída.

A camada de entrada recebe os dados originais e envia informações para as camadas subsequentes. A camada de saída interpreta as informações das camadas anteriores produzindo um resultado, que é transferido para o exterior da rede, com sendo o “entendimento da rede neural” sobre os dados de entrada.

A(s) camada(s) oculta(s), apesar de não interagir (em) diretamente com o mundo externo à rede neural, agrega(m) complexidade, na forma de uma representação interna do problema, de modo a permitir que a rede neural possa ser utilizada em casos onde as soluções envolvam alta complexidade e não linearidades.

Pesos das Conexões

Os pesos das conexões expressam a intensidade relativa das várias conexões que transferem dados de uma camada para a outra e representam um parâmetro crítico para a rede neural, uma vez que através do seu ajuste repetitivo, que constitui um processo de aprendizado, a rede tende a produzir a saída (resultado) desejada. O objetivo do treinamento de uma rede neural é, portanto encontrar um conjunto de pesos que interpretará corretamente todos os conjuntos de valores de entrada considerados para um determinado problema [Haykin, 02].

Função de Agregação

Esta função determina o modo pelo qual os diversos sinais de entrada do neurônio são compostos para a formação da entrada equivalente. Neste trabalho se considerou apenas a Função Soma (Somatório).

Função de Ativação (Transformação) - f

A relação entre os níveis de ativação interna do neurônio, denominada função de ativação, define o nível de ativação ou estimulação (disparo) a partir do qual o neurônio pode gerar uma saída. A função de ativação mais utilizada em redes neurais é a função sigmóide (figura 2.13), contínua e sucessivamente diferenciável, cujo resultado varia gradualmente entre dois valores assintóticos (valores de saturação), tipicamente 0 e 1, ou -1 e $+1$.

Figura 2.13 Função Sigmóide

2.4.2 Mapa Auto Organizável de Kohonen

A estrutura do mapa de Kohonen consiste de uma camada de neurônios altamente interconectados, onde cada neurônio está conectado com todos os outros neurônios desta camada (inibição lateral) e com cada um dos neurônios da camada de entrada, (figura 2.14).

Figura 2.14 Mapa de Kohonen

O algoritmo de Kohonen [Kohonen, 90] é um eficiente processo de auto-organização, o qual produz, de modo não supervisionado, a distribuição de um espaço de entrada V_{in} em um outro espaço de menor dimensão V_{out} , preservando as relações topológicas entre os vetores de entrada [Marin, 02].

O espaço de saída é, portanto, representado por uma matriz bidimensional de neurônios apresentando relacionamento de vizinhança [Chicco, 01], de modo que vetores de entrada similares ativem neurônios vizinhos.

No final do processamento, o resultado é um tipo de mapa onde neurônios com funções similares estarão agrupados. Pode-se efetuar, deste modo, a classificação dos valores de entrada uma vez que valores similares serão representados em áreas específicas do mapa

da rede, formando classes ou agrupamentos (clusters).

2.4.3 Perceptron Múltiplas Camadas

O Perceptron é construído em torno de um neurônio não linear (modelo de McCulloch-Pitts) [Haykin, 02], consistindo de uma função de agregação linear seguida por uma função de ativação não linear. A função de agregação deste neurônio calcula a combinação linear das entradas aplicadas, sendo a soma resultante aplicada à função de ativação. O neurônio produz uma saída $+1$ se a entrada da função de ativação for positiva e -1 caso contrário.

O objetivo do perceptron é, portanto, classificar corretamente o conjunto de estímulos (entradas) aplicados externamente, em duas classes, sendo a regra de decisão para esta classificação, atribuir o ponto representado pelas entradas aplicadas à uma classe se a saída do perceptron for $+1$ e à outra classe se ela for -1 .

A rede Perceptron múltiplas camadas apresenta os seus neurônios distribuídos em uma camada de entrada, que recebe os sinais a serem processados, uma ou mais camadas intermediárias, conhecidas como camadas ocultas, e uma camada de saída (figura 2.15).

Figura 2.15 Rede Neural Perceptron Múltiplas Camadas (neste caso, quatro)

A quantidade de neurônios em cada uma das camadas depende em grande parte da modelagem do problema sendo tratado, uma vez que o tamanho do vetor de entradas é

proporcional ao número de componentes do sinal de entrada.

O vetor de saída é dependente da natureza da informação a ser provida pela rede neural, sendo então, o seu dimensionamento definido pelos resultados desejados.

O número de camadas internas pode ser determinado pelo algoritmo de treinamento adotado (sendo determinado pelo processo de treinamento), ou mesmo escolhido a priori.

2.4.4 Algoritmos Genéticos

Os Algoritmos Genéticos constituem técnicas de pesquisa, baseadas nos mecanismos de seleção utilizados pela natureza, de modo que os indivíduos mais aptos de uma população serão aqueles que sobrevivem ao adaptarem-se mais facilmente às variações ocorridas no seu meio ambiente [Rosa, 03].

Ao tratar com uma população de indivíduos bem adaptados estes algoritmos obtêm segurança na quantidade, minimizando desta forma, a possibilidade de se atingir um falso ponto ótimo (solução local) [He, 02], figura 2.16.

Figura 2.16 Mínimos, Local e Global

Através de mecanismos de troca de informações randômicos (seleção, mutação e cruzamento) os sobreviventes mais adaptados são combinados, de modo que em cada nova geração um novo elemento é criado combinando-se uma unidade de informação

com as partes mais adaptadas dos indivíduos da geração anterior, de modo que os cromossomos com melhores aptidões (melhor resposta ao problema) serão encontrados ao fim deste processo [Fregni, 97].

Cada solução do problema em estudo é mapeada no cromossomo de um indivíduo de uma determinada população, que por sua vez representa um subconjunto do espaço de soluções do problema em estudo. Procura-se, então, a cada iteração (geração) encontrar-se novos pontos de pesquisa onde são esperados melhores desempenhos.

Apesar de usarem processos de troca de informação aleatórios, os Algoritmos Genéticos não constituem um algoritmo de pesquisa aleatória, uma vez que utilizam as informações presentes em cada indivíduo para se investigar novos pontos de pesquisa, objetivando a maximização do rendimento [Bento, 04].

2.4.5. Algoritmo Fuzzy-Isodata

Um método alternativo de classificação de perfis implementado utiliza computação fuzzy na forma do algoritmo Fuzzy-Isodata, o qual objetiva encontrar partições fuzzy, a partir de um conjunto de dados de treinamento, através de uma generalização nebulosa da função de mínimos quadráticos [Bastos, 04] [Miyamoto, 98].

Esta generalização consiste na minimização de uma função objetivo de modo a organizar um conjunto de N objetos, de dimensão p , em subconjuntos, denominados clusters, de tal modo que os objetos em um mesmo subconjunto sejam semelhantes entre si, mas sejam diferentes, em algum grau, de outros objetos pertencentes a outros clusters. Baseia-se na medida de similaridade de modo a caracterizar o grau de semelhança entre os objetos [Velasquez, 01].

A justificativa do uso deste algoritmo vem da constatação de que os sistemas fuzzy são aderentes à solução do problema de determinar similaridades em massas imensas de dados, uma vez que estes dados possam ser representados por expressões lingüísticas

condicionais que caracterizem suas propriedades. Havendo disponibilidade de quantidade suficiente de dados históricos, as similaridades existentes podem ser identificadas.

A partir da conceituação apresentada neste capítulo, juntamente com o método descrito por [Rosa, 04], tem-se as informações, ferramentas e métodos necessários para se proceder à modelagem da arquitetura proposta, assim como da aplicação por esta suportada.

3. DEFINIÇÃO DA ARQUITETURA DO SISDEC

A facilidade de troca de informação entre todos os elementos de um sistema de gerenciamento e controle de energia e o acesso a estas informações por todas as partes interessadas é um requisito básico para a competitividade dos negócios de energia atualmente [Becker, 00] [Kim, 00] [McMorran, 04].

A forma como a informação é representada é de vital importância na interoperabilidade de sistemas [Tosic, 99], uma vez que a visão conjunta destes dados como sendo uma base de dados heterogênea e distribuída implica na conversão dos formatos utilizados por cada um dos sistemas.

Uma vez que a integração das informações dentro das empresas de energia elétrica aparece como fator primordial para o bom desempenho do negócio [Dong, 02] [Huang, 03] [Filho, 01], e buscando-se a disseminação de informações oriundas de uma grande diversidade de sistemas de informação, procura-se neste trabalho dar um passo inicial no sentido desta integração [Simunic, 04], ao se modelar, através de um modelo comum, as informações relativas à funcionalidade Previsão da Curva de Carga do Sistema Elétrico de Potência, assim como definir uma arquitetura distribuída que forneça o suporte à implementação desta funcionalidade.

A previsão da carga foi escolhida como uma funcionalidade a ser implementada, devido à necessidade desta informação para as tarefas de planejamento, análise e operação dos sistemas elétricos de potência [Senjyu, 02] [Suarup, 02]. A definição de uma arquitetura distribuída surge como atendimento à necessidade de integração, assim como de acesso a informações oriundas de diversos sistemas heterogêneos.

Considera-se, neste trabalho, o mapeamento entre o CIM e o RM-ODP proposto por [Rosa, 04], a partir do qual o modelo CIM foi usado com uma abordagem ODP, para se obterem as interfaces CIM correspondentes às funções e serviços definidos pela aplicação do RM-ODP ao problema do estudo da carga, assim como a definição da sua arquitetura.

Neste capítulo, usando-se os conceitos fornecidos no capítulo anterior apresenta-se a

modelagem na definição da arquitetura proposta, sendo detalhada a aplicação do método descrito por [Rosa, 04].

3.1 Arquitetura do SISDEC

O Sistema distribuído aqui proposto se insere no nível Gestão do modelo Sistema Aberto de Automação – SAA [Becerra, 98], contexto no qual a modelagem RM-ODP será efetuada.

As informações providas por este Sistema objetivam a alta gerência da empresa, uma vez que podem auxiliar nas decisões de investimento (a curto, médio e longo prazos) na rede de transmissão e subestações, gerenciamento dos impactos de restrições de fornecimento, acompanhamento de operação de equipamentos em regime de sobrecarga, dentre outras.

3.1.1 Estrutura Básica

O SISDEC é um sistema composto por vários módulos, que processam informações referentes à carga do sistema elétrico de potência, as quais apresentam similaridades entre dias equivalentes da semana. Seus módulos são: Classificador, Treinador, Previsor e Gerenciador.

A previsão de carga é fornecida pelo módulo Previsor, que utiliza redes neurais para proporcionar a informação da carga prevista. Dependendo do dia solicitado, a previsão de sua carga é efetuada por uma rede neural específica, a qual foi treinada para este dia.

Devido a similaridades, os dados de carga podem ser analisados e classificados em conjuntos, de modo que dados heterogêneos serão considerados homogêneos, com respeito a algum critério ou função objetivo [Palácio, 01], e passarão a constituir domínios locais de dados. Cada um destes domínios, doravante chamado de perfil de carga, será modelado por uma ou mais redes neurais.

Cada rede neural deve ser treinada para um determinado perfil de carga, através de um algoritmo de treinamento. No fim da fase de treinamento a rede apresenta mínimos erros e

grande generalização, significando que ao receber uma solicitação de um dia que não faz parte dos dados de treinamento, mas classificou-se dentro deste perfil, a rede neural consiga produzir o resultado adequado.

A classificação de perfil de carga utiliza como insumo, dados históricos de carga, a serem obtidos do sistema de supervisão e controle, através de interfaces de comunicação, pelo middleware descrito em [Becerra, 04]. Estes dados são armazenados no repositório CIM, ficando disponível para os diversos módulos do SISDEC, assim como os demais dados processados pelo SISDEC. A disponibilidade destes dados no formato CIM permite a exportação e utilização por parte de outras aplicações.

A partir do repositório, o módulo Classificador efetua a classificação dos vários perfis de carga em dias similares.

Uma vez armazenada no repositório CIM, esta informação poderá ser utilizada pelo módulo Treinador ou mesmo ser fornecida sob solicitação, para um cliente. A informação de perfil pode ser utilizada em aplicações que estudem a similaridade e comportamento sazonal do consumo de energia.

A partir das informações deste repositório, o módulo Treinador atua no módulo Previsor de forma a treina-lo para um destes perfis de carga CIM. A rede neural obtida após o treinamento é modelada em CIM e também armazenada no repositório CIM.

Estas redes assim armazenadas poderão ser utilizadas pelo módulo Previsor na tarefa de previsão da curva de carga para um dia solicitado pelo cliente.

Adicionalmente, pelo fato de estarem modeladas, estas estruturas neurais podem ser exportadas para outras aplicações (implementando-se interfaces CIM para as mesmas) para simulações, por exemplo, o MatLab. Ou seja, consegue-se exportar as informações da estrutura da rede, de modo que a mesma possa ser reconstituída em outros domínios.

O módulo Gerenciador é responsável pelo gerenciamento da interface entre o repositório e os demais módulos, exercendo as funções, entre outras, de iniciar e supervisionar

processos.

Os requisitos funcionais a serem atendidos para o SISDEC, são:

- Atender as solicitações dos diversos usuários
- Efetuar a previsão de carga
- Prover a interoperabilidade de informação no ambiente compartilhado
- Prover a distribuição das estruturas neurais e dos repositórios de informação
- Efetuar a classificação dos perfis de carga
- Tornar disponíveis as informações de perfil de carga
- Tornar disponíveis as informações de carga histórica
- Efetuar o treinamento das redes neurais
- Tornar disponíveis as redes neurais treinadas

3.1.2 Modelagem RM-ODP

A modelagem, segundo os padrões estabelecidos nos pontos de vista do modelo ODP, por considerar a integração do Sistema Distribuído de Estudo da Carga com os demais sistemas de informação da empresa, assim como a distribuição de suas funcionalidades, apresenta-se adequada na definição da arquitetura proposta.

Com base na descrição detalhada do sistema, enfocando as expectativas dos usuários em termos de o que o sistema deverá fazer, se extraem os seus casos de usos que constituem a primeira fase da definição deste sistema.

Uma vez que a ordem na qual se aplicam os pontos de vistas e o tipo de informação no que diz respeito as suas utilizações no processo de modelagem não é determinada pelo RM-ODP, será adotada a seqüência Empresa, Informação, Computação, Engenharia e Tecnologia.

3.1.2.1 Ponto de Vista Empresa

A especificação deste ponto de vista permite estabelecer o ambiente de negócios no qual o sistema irá atuar, assim como a definição dos objetos empresa, suas funcionalidades e responsabilidades no sistema distribuído [Filho, 01].

Neste ponto de vista é determinado o modelo do sistema distribuído, do ambiente com o qual interage, são estabelecidos os seus elementos básicos, assim como suas principais características [Filho, 01].

Os casos de uso UML, por representarem a funcionalidade do sistema serão aqui utilizados para a representação do ponto de vista Empresa.

A partir dos requisitos funcionais listados foram elaborados tanto os casos de uso do SISDEC, os quais definem a sua primeira representação e capturam as necessidades a serem atendidas, quanto à definição dos objetos Empresa correspondentes.

Os objetos Empresa, do SISDEC, resultantes dos requisitos funcionais são:

- Usuário
- Classificador
- Treinador
- Previsor
- Gerenciador
- Repositório

Estes objetos Empresa possuem papéis exercidos dentro da comunidade a que pertencem, sendo que os seguintes papéis podem ser caracterizados entre os objetos Empresa do SISDEC:

- Servidor
- Consultor
- Fornecedor

O papel Fornecedor, apresentado pelos objetos Empresa Classificador, Treinador e Previsor, é responsável por processar perfil, rede neural e carga prevista, efetuando, portanto, a classificação, o treinamento e a previsão de carga, figura 3.1.

O papel Servidor, apresentado pelo objeto Empresa Gerenciador, fornece informações dos recursos, efetuando a interface entre o objeto Empresa Repositório, provedores (objetos Empresa Previsor, Classificador e Treinador) e objeto Empresa Usuário, figura 3.2. O papel Consultor, caracterizado pelo objeto Empresa Usuário, solicita as várias informações (rede neural, carga, perfil), figura 3.3.

Figura 3.1 Papel dos objetos Empresa Classificador, Treinador e Previsor

Figura 3.2 Papel do objeto Empresa Gerenciador

Figura 3.3 Papel do objeto Empresa Usuário

A figura 3.4 mostra os objetos Empresa e seus respectivos papeis dentro do SISDEC.

Figura 3.4 Objetos Empresa e seus papeis

O modelo de casos de uso elaborado para o SISDEC pode ser visualizado na figura 3.5.

Figura 3.5 Modelo de Casos de Uso para o SISDEC

A partir deste modelo, foram expandidos cada um dos casos de uso do SISDEC, a seguir.

UC01: Solicitar Carga Histórica

O usuário poderá:

- Listar a relação de cargas disponíveis
- Efetuar a pesquisa (busca) de uma carga específica
- Requisitar carga não relacionada no repositório
- Guardar carga histórica no repositório

Figura 3.6 Caso de Uso UC01: Solicitar Carga Histórica

UC02: Solicitar Perfil

O usuário poderá:

- Listar a relação de perfis disponíveis
- Efetuar a pesquisa (busca) de um perfil específico

Figura 3.7 Caso de Uso C02: Solicitar Perfil

UC03: Classificar Perfil

O usuário poderá:

- Informar dias
- Requisitar classificação

Figura 3.8 Caso de Uso UC03: Classificar Perfil

UC04: Treinar Rede

O usuário poderá:

- Treinar rede neural

Figura 3.9 Caso de Uso UC04: Treinar Rede

UC05: Solicitar Rede Neural

O usuário poderá:

- Listar relação de redes disponíveis
- Efetuar a pesquisa (busca) de uma rede neural específica
- Requisitar uma rede desejada

Figura 3.10 Caso de Uso UC05: Solicitar Rede Neural

UC06: Solicitar Previsão

O usuário poderá:

- Listar relação de previsões disponíveis
- Efetuar pesquisa (busca) de uma determinada previsão
- Requisitar uma previsão específica
- Fazer a previsão

Figura 3.11 Caso de Uso UC06: Solicitar Previsão

3.1.2.2 Ponto de Vista Informação

O ponto de vista da Informação expressa o comportamento do sistema face às transformações das informações manipuladas e armazenadas pelos objetos Empresa, sendo obtido a partir dos componentes identificados na modelagem da empresa [Filho, 01]. Ele descreve a informação requisitada pelo sistema, através de representações, denominadas esquema, que especificam o estado e a estrutura do objeto [Varoto, 02].

Uma vez que este ponto de vista Informação deve capturar uma representação consistente

da estrutura das informações e que os diagramas de classe UML permitem a representação da estrutura interna dos objetos bem como seus relacionamentos, estes diagramas serão utilizados para a representação deste ponto de vista.

No SISDEC são definidos três objetos de Informação:

- Objeto Informação Carga (Prevista e Histórica)
- Objeto Informação Perfil de Carga
- Objeto Informação Rede Treinada

Cada objeto empresa utiliza um ou mais tipos de objetos informação, que correspondem aos pontos de conformidade entre o ponto de vista Empresa e o ponto de vista Informação, que estão representados na figura 3.12.

<u>Objeto Empresa</u>	<u>Objeto Informação</u>
Usuário	Carga Perfil Rede Neural
Treinador	
Classificador	Carga Perfil
Previsor	Carga
Gerenciador	Carga Perfil Rede Neural
Repositório	

Figura 3.12 – Pontos de conformidade

Por ser uma aplicação específica, o SISDEC não tem como ser mapeado diretamente em uma classe definida no esquema CIM, mas por outro lado o modelo CIM permite a integração de novos conceitos, como extensões de seu padrão [Tossic, 99].

Considerando que o SISDEC é um aplicativo de software, os seus objetos Informação

foram modelados como uma extensão da classe LogicalElement, pertencente ao modelo núcleo do CIM, figura 3.13.

Figura 3.13 Localização do SISDEC

Ao fazer esta extensão, os objetos do SISDEC herdam todas as propriedades e métodos desta classe mãe. A partir desta extensão são, então, definidos os métodos e as propriedades específicas do SISDEC, figura 3.14.

Figura 3.14 Diagramas de classes CIM do SISDEC (visão macro)

Os três esquemas ODP, invariantes, estáticos e dinâmicos, para o SISDEC podem ser obtidos a partir das classes e de seus relacionamentos.

Para o SISDEC temos as seguintes regras:

A classe **CteepSisdecPerfil** caracteriza-se por ser composta por pelo menos uma classe dia, seu valor característico de carga não pode ser nulo e é classe filha da classe CIM **LogicalElement**.

A classe **CteepSisdecCarga** inclui as classes **CteepSisdecCargaHistorica** e **CteepSisdecCargaPrevista**, as quais são compostas por classes **CteepSisdecValores**, e também é classe filha da classe CIM **LogicalElement**. Adicionalmente, as classes **CteepSisdecCargaHistorica** e **CteepSisdecCargaPrevista** não podem ter valores de carga nulos.

A classe rede neural é composta pelas classes **CteepSisdecCamadaEntrada**, **CteepSisdecCamadaOculta** e **CteepSisdecCamadaSaida**, sendo também filha da

classe CIM LogicalElement. Todas as três classes devem estar presentes de forma a constituírem a classe rede neural.

Estes relacionamentos constituem os esquemas invariantes ODP, para estes objetos [Rosa, 04], uma vez que restringem os estados e a estrutura do objeto Informação, ao longo de sua existência.

Os esquemas dinâmicos referem-se aos métodos implementados nas classes, que definem as mudanças permitidas nos estados e estruturas do objeto Informação. Toda ação que cause uma mudança no estado de um objeto é capturada neste esquema.

A captura dos estados e estruturas dos objetos Informação em um instante particular do tempo é efetuada pelos esquemas estáticos, os quais são representados pelos diagramas de classes, figura 3.14.

Os esquemas invariantes ODP para os objetos Informação Carga, Perfil e Rede Neural do SISDEC são apresentados na figura 3.15, e os esquemas dinâmicos na figura 3.16.

Figura 3.15 Esquemas Invariantes ODP

```

+ informaRecurso() : float
+ perfil() : int
+ redeNeural() : float
+ cargaPrevista() : float
+ cargaHistorica() : float

```

Figura 3.16 Esquemas Dinâmicos ODP

3.1.2.3 Ponto de Vista Computação

O ponto de vista da Computação está relacionado com os processos de alteração dos dados nos modelos de informações, capturando os aspectos dinâmicos do sistema em desenvolvimento e representando as operações responsáveis por tais alterações nos dados.

Os diagramas de seqüência UML por fornecerem uma representação de como os objetos podem interagir na realização de cada uma das funcionalidades do sistema e por representarem as seqüências de interação de cada um destes objetos, capturando o comportamento dinâmico de cada objeto, serão aqui utilizados para representarem o ponto de vista Computação, juntamente com seus casos de uso correspondentes.

Casos de Uso

Os Casos de Uso definidos no ponto de vista Empresa, são expandidos para representarem o ponto de vista Informação.

UC01: Solicitar Carga Histórica

- UC01.1: Listar Relação de Carga
- UC01.2: Pesquisar Carga Histórica

- UC01.3: Requisitar Carga Histórica
- UC01.4: Guardar Carga Histórica

UC02: Solicitar Perfil

- UC02.1: Listar Relação Perfil
- UC02.2: Pesquisar Perfil

UC03: Classificar Perfil

- UC03.1: Informar Dias
- UC03.2: Requisitar Classificações

UC04: Treinar Rede

UC05: Solicitar Rede Neural

- UC05.1: Listar Relação Redes
- UC05.2: Pesquisar Rede
- UC05.3: Requisitar Rede

UC06: Solicitar Previsão

- UC06.1: Listar Relação de Previsão
- UC06.2: Pesquisar Previsão
- UC06.3: Requisitar Previsão
- UC06.4: Fazer Previsão

Para efeito de discussão, somente o Caso de Uso UC01 será aqui apresentado, os demais casos de uso podem ser consultados no apêndice E.

UC01.1: Listar Relação de Carga

O usuário solicita a relação de cargas históricas disponíveis no repositório.

Figura 3.17 Caso de Uso UC01.1:Listar Relação de Carga

Para este caso de uso, tem-se como requisito a existência de uma relação de cargas históricas disponíveis no repositório.

Figura 3.18 Requisito para o Caso de Uso Listar Relação de Carga

UC01.2: Pesquisar Carga Histórica

O usuário fornece o critério de filtro (dia/mês/ano) para acesso às informações de carga desejada.

Figura 3.19 Caso de Uso UC01.2: Pesquisar Carga Histórica

O requisito para o UC01.2 é a existência da carga solicitada armazenada, figura 3.20.

Figura 3.20 Requisito para o Caso de Uso Pesquisar Carga Histórica

UC01.3: Requisitar Carga Histórica

O usuário fornece o critério de filtro (dia/mês/ano) para acesso às informações de carga

desejada, não existente no repositório. O módulo Gerenciador busca no Middleware CTEEP a informação solicitada, sendo a existência desta informação um requisito para este caso de uso, figura 3.22.

Figura 3.21 Caso de Uso UC01.3: Requisitar Carga Histórica

Figura 3.22 Requisito para o Caso de Uso Requisitar Carga Histórica

UC01.4: Guardar Carga Histórica

O usuário solicita o armazenamento da carga histórica no repositório, tendo como requisito a existência de uma carga histórica obtida pelo UC01.3, figura 3.24.

Figura 3.23 Caso de Uso UC01.4: Guardar Carga Histórica

Figura 3.24 Requisito para o Caso de Uso Guardar Carga Histórica

3.1.2.4 Ponto de Vista Engenharia

Neste ponto de vista, que trata da infraestrutura requerida para suportar a distribuição do sistema, definem-se os mecanismos e as funções necessárias para este suporte, resultando na infraestrutura de apoio ao Ponto de Vista Computação.

Uma vez que no SISDEC todo acesso ao repositório, assim como a comunicação entre

usuários, repositório e provedores são efetuados pelo módulo gerenciador, conforme definido pelos diagramas de sequência, o conceito cliente servidor pode ser aplicado ao mesmo.

O modelo de Engenharia, para o SISDEC, pode ser visualizado na figura 3.25 a qual relaciona os nós (elementos funcionais) e suas conexões, definindo um modelo de objetos constituídos pelos objetos básicos de Engenharia (Cliente e Servidor) e os que constituem o canal de comunicação entre os mesmos.

Neste modelo, observa-se que no SISDEC o conceito Cliente-Servidor pode ser aplicado em vários níveis de abstração.

Em uma primeira abstração os usuários seriam objetos Cliente e o SISDEC o objeto Servidor. No segundo nível de abstração, o objeto SISDEC Gerenciador será o objeto Cliente, com os objetos SISDEC Gerenciador de Repositórios, Classificador, Treinador e Previsão como objetos Servidor.

No terceiro nível de abstração os objetos Empresa Classificador, Treinador e Previsor seriam os objetos Cliente do objeto Empresa Gerenciador, agora como objeto Servidor.

O módulo servidor da figura 3.25 é responsável pelas funções de distribuição e interoperabilidade, apresentando protocolos de comunicação, conversores de protocolo (adaptadores de protocolo) e gerenciador de comunicação. Ele possibilita mecanismos de autenticação, assim como, pesquisa, por parte dos clientes, de funcionalidades e de mecanismos de comunicação [DMTF, 04a], constituindo a parte central da arquitetura aqui proposta.

Figura 3.25 – Objetos Engenharia

O SISDEC é um sistema com estilo arquitetural misto sendo parcialmente, um sistema baseado em repositórios, uma vez que seus diversos módulos (subsistemas) não interagem diretamente entre si, de modo que, com exceção do Módulo Gerenciador, a única interação possível entre eles, é a manipulação das informações compartilhadas através dos repositórios. Como o acesso ao repositório somente é efetuado pelo Módulo Gerenciador, O SISDEC também é um sistema baseado na arquitetura cliente servidor.

Este estilo arquitetural apresenta grande flexibilidade na medida que para os subsistemas que não têm dependências entre si, estes podem sofrer alterações sem interferirem no comportamento dos demais.

3.1.2.5 Ponto de Vista Tecnologia

A escolha da tecnologia para suportar a distribuição do sistema é efetuada neste ponto de vista, o qual descreve a implementação do sistema RM-ODP em termos de uma configuração de objetos Tecnologia representando os componentes de software e hardware da implementação.

Este ponto de vista estabelece uma ligação entre o conjunto de especificações dos modelos Empresa, Informação, Computação e Engenharia, e a implementação real do sistema distribuído [Filho, 01].

Será utilizada a arquitetura voltada para a unificação dos mecanismos que descrevem e compartilham informações, conhecida como WBEM (Web Based Enterprise Management), como tecnologia de middleware, responsável pela implantação das funções que suportem as interações dos objetos distribuídos.

Utilizou-se WBEM como tecnologia de middleware uma vez que a mesma já utiliza o CIM em sua implementação, além de apresentar características de interoperabilidade, repositório de tipos, registro de nomes e autenticação.

O SISDEC sob o Ponto de Vista Tecnologia pode ser visto na figura 3.26.

Figura 3.26 Ponto de Vista Tecnologia

O componente de software que facilita a comunicação entre objetos (conhecido genericamente por broker), é dentro do contexto SISDEC representado pelo módulo Gerenciador. Este módulo constitui o broker entre os usuários, os módulos Classificador, Treinador, Previsor e o Repositório, sendo também responsável pelas funções de

notificação de eventos, registro de nomes e autenticação [DMTF, 04a].

Neste capítulo procedeu-se a aplicação da modelagem CIM com abordagem ODP de forma a se obter uma arquitetura distribuída que permite a interoperabilidade e distribuição dos diversos módulos constituintes do SISDEC, com a padronização das interfaces dos objetos e da informação compartilhada pelos mesmos.

4. IMPLEMENTAÇÃO

4.1 Introdução

O modelo de referência ODP foi utilizado com uma abordagem CIM para a definição da arquitetura distribuída na qual o Sistema Distribuído de Estudo da Carga foi implementado.

O Modelo Comum de Informação CIM foi também usado para se modelarem os perfis de carga, a estrutura neural das redes já treinadas, as cargas históricas e por fim a informação principal gerada pela aplicação, a curva de carga prevista.

Usou-se Diagramas UML para as definições de classe, casos de uso e diagramas de interação, visando à modelagem funcional dos elementos da arquitetura proposta.

A modelagem assim efetuada permitiu se mapear a arquitetura distribuída na arquitetura WBEM (Web-Based Enterprise Management), a qual está voltada para a unificação dos mecanismos que descrevem e compartilham informações.

WBEM é um conjunto de padrões, desenvolvido pela DMTF, para unificar o gerenciamento dos ambientes computacionais corporativos, permitindo que uma variedade de elementos de processamento de informação, de diferentes fornecedores, seja gerenciada de uma maneira uniforme [HP, 03]. Constitui uma arquitetura de gerenciamento com uma linguagem para descrever as funcionalidades de gerenciamento tanto de dispositivos quanto de serviços, com um conjunto de interfaces para acesso às informações de gerenciamento e um extenso conjunto de modelos para aplicações (telecomunicações, armazenamento, computação, entre outras).

4.2 Protótipo

O protótipo utilizou a implementação WBEM da Microsoft, WMI (Windows Management Instrumentation), uma vez que a maioria dos computadores da Transmissão Paulista é baseada no sistema operacional Windows, o qual já possui o WMI instalado (Windows 2000 e XP) ou passível de instalação gratuita (Windows 98).

Devido ao fato da estrutura computacional corporativa da Cteep ser baseada, em sua maioria, em computadores padrão PC executando sistema operacional Windows interligados por uma rede Intranet, a seguinte tecnologia foi adotada neste trabalho:

Sistema Operacional:

Microsoft Windows 98/XP/2003

Clientes Windows:

Com navegadores (browsers)

Aplicativos dedicados

Arquitetura WBEM:

WMI – Windows Management Instrumentation – Microsoft

Infraestrutura de comunicação:

Rede corporativa da CTEEP – Transnet

A não necessidade de instalação de módulos de software, estranhos ao ambiente corporativo, e ausência de impactos negativos em relação à modelagem ODP-CIM também nortearam a escolha do WMI neste trabalho inicial.

4.2.1 Arquitetura do WMI

Mantendo o paralelo com o WBEM, o WMI foi elaborado considerando-se quais elementos do esquema CIM aplicam-se a computadores com o sistema operacional Windows e a partir destes fez-se uma extensão deste esquema, criando classes que exponham informações adicionais, disponíveis sob o Windows.

A capacidade de gerenciamento é provida por um módulo de software (CIMOM), que mantém uma lista das classes suportadas e que fornece dados atualizados quando instâncias das classes são solicitadas. Adicionalmente, este módulo responde a chamadas de métodos em objetos implementando as ações requisitadas [MS, 05]. O CIMOM reside no serviço WinMgmt, sendo implementado por WinMgmt.exe.

Uma API deve ser usada para efetuar a comunicação com o CIMOM, de modo a ser possível, a execução de consultas do esquema, listagem das instâncias disponíveis e chamadas de métodos.

No WMI, a Microsoft adotou um conjunto de interfaces DCOM para tornar disponível a funcionalidade do CIMOM, de modo que a mesma seja, também, acessível a partir de computadores remotos sem a necessidade de código adicional.

Uma base de dados, denominada Repositório de Objetos CIM, contendo as definições de classes e demais informações estáticas, é mantida pelo serviço WinMgmt.

Para as informações sobre objetos dinâmicos, o WinMgmt suporta o registro de provedores, que constituem servidores COM suportando um conjunto de interfaces definidas. Os provedores podem ser consultados sob demanda pelo CIMOM para informações dinâmicas, ou estes podem chamar o CIMOM para informar alteração de dados, assim como dinamicamente, descreverem novas classes no esquema [MS, 05].

4.2.2 Módulos da Aplicação

Devido a sua complexidade inerente, um Sistema Elétrico de Potência, que é composto por diversos componentes com comportamento não linear, pode apresentar problemas com características importantes diversas, exigindo técnicas sofisticadas de análise.

Esta não linearidade, a grande quantidade de interações entre as partes do sistema, as imprecisões na sua modelagem, as variações aleatórias de carga e a grande quantidade de dados históricos [Zang, 98] tornam tanto os problemas quanto à operação do Sistema Elétrico de Potência fortes candidatos ao uso de técnicas de Inteligência Artificial [Dabbaghchi, 97].

O procedimento tradicional para se elaborar um modelo de previsão de dados resume-se na coleta de dados históricos, na identificação de padrões ou tendências nestes dados, na modelagem matemática destes padrões e ou tendências e finalmente no teste da eficiência das previsões [Palácio, 01].

A identificação de padrões ou tendências foi implementado utilizando-se redes neurais de Kohonen e algoritmo Fuzzy-Isodata, enquanto que a previsão de carga de curto prazo foi implementada utilizando-se técnicas de inteligência artificial, nomeadamente redes neurais artificiais treinadas por algoritmo genético.

As vantagens do uso de redes neurais para a previsão de carga são: a não necessidade de um modelo da carga [Zang, 98], a alta precisão [Winkler, 01] e a maior eficiência na aprendizagem de padrões em dados com ruído, incompletos e/ou contendo valores contraditórios.

O modelo de objetos da aplicação SISDEC pode ser visualizado na figura 4.1, constituindo-se de três fases distintas de operação: classificação, treinamento e previsão.

Na fase de treinamento, várias redes são configuradas e treinadas. Os dados históricos a serem utilizados para a fase de treinamento são processados previamente de modo a se

tratarem ruídos e inconsistências, normaliza-los e classifica-los em similaridades (perfis diários), o que constitui a fase de classificação.

Figura 4.1 Modelo de Objetos do SISDEC

Os dados históricos processados são armazenados com uma etiqueta na qual são inseridas informações relativas ao dia do ano, dia da semana, se é feriado, se o dia anterior foi feriado, se o próximo dia será feriado e se está em horário de verão.

A normalização dos dados é um importante estágio do processo de treinamento da rede neural que facilita o treinamento ao adaptar as entradas e saídas a uma escala mais adequada, de modo que estes valores estejam sempre em um mesmo intervalo, permite ainda a aderência à simetria das funções de ativação dos neurônios, faz com que a influência dos valores mais altos não suprimam a influência dos valores menores [Swarup, 02], e influencia positivamente tanto a taxa de classificação quanto o erro

médio quadrático [Zang, 98].

Foi utilizada a normalização conhecida como Normalização Externa, na qual todos os dados de treinamento são normalizados para uma faixa específica antes de serem utilizados [Zang, 98].

A normalização adotada foi transformação linear, para a faixa [a,b]:

$$X_n = \frac{((b-a) \times (X_o - X_{\min}))}{((X_{\max} - X_{\min}) + a)} \quad (4.1)$$

com $a = -1$ e $b = 1$

Onde:

X_n	entrada normalizada
X_o	entrada não normalizada (original)
$X_{\max}$	maior valor de entrada original do conjunto de dados
$X_{\min}$	menor valor de entrada original do conjunto de dados

4.2.3 Implementação SISDEC

A implementação do SISDEC pode ser vista de forma macro, a partir das figuras 4.2 e 4.3, nas quais observa-se o caráter distribuído das informações e funcionalidades, de forma transparente em relação à suas localizações. Estas figuras são de caráter informativo, uma vez que a distribuição dos módulos pode ser efetuada conforme a necessidade.

Figura 4.2 Contexto Físico do SISDEC

As diversas instâncias do Gerenciador atuam como Brokers entre os usuários e as informações solicitadas pelos mesmos, permitindo uma considerável combinação de usuários e provedores processando em um mesmo computador ou distribuídos por vários computadores conectados em rede.

Seja um usuário solicitando informações que são gerenciadas por um Gerenciador remoto. O Gerenciador local, ao receber a solicitação do usuário, verifica que estas informações não são de sua responsabilidade e solicita ao Gerenciador correspondente que responda a consulta feita. Neste caso, o Gerenciador local passa a ser um cliente WBEM de um Gerenciador remoto.

Se a informação residir no seu repositório, o Gerenciador remoto responde diretamente com estes dados, caso contrário solicitada ao(s) provedor(es) que informem os dados solicitados.

As informações geradas pelos módulos, Classificador e Treinador, são enviadas ao Gerenciador para que este atualize o repositório CIM. As estruturas destas informações são mantidas no repositório como estáticas, enquanto que os seus conteúdos são de caráter dinâmico.

Figura 4.3 Contexto Lógico do SISDEC

Na figura 4.3 pode-se visualizar o contexto lógico do SISDEC, onde podem ser identificados barramentos virtuais: entre provedores e Gerenciador, entre usuários e um mesmo Gerenciador, entre Repositório e Gerenciador e por fim entre os Gerenciadores.

Conforme a localização física dos provedores e usuários, os diversos barramentos são mapeados na infraestrutura da rede corporativa, com exceção da ligação Gerenciador-repositório. Observe que podem existir computadores sem o módulo Gerenciador, mas todo acesso se efetuará por pelo menos um Gerenciador.

Utilizou-se para a implementação dos provedores de instâncias e de métodos do SISDEC, programação em C++, com o ambiente Visual Studio e ATL (Active Template Library), enquanto que os arquivos MOF correspondentes às classes e métodos foram gerados através do ambiente visual CIM-Studio, todos da Microsoft. Estes arquivos MOF constituem um item de portabilidade, uma vez que descrevem e definem os módulos da aplicação SISDEC, independentemente da plataforma utilizada.

A funcionalidade do SISDEC é provida pela composição dos seus módulos, com os seguintes aspectos funcionais.

Módulo Classificador

O objetivo da classificação é separar os dados de carga de forma que, dados heterogêneos possam ser considerados homogêneos, com respeito a algum critério ou função objetivo, e venham a constituir domínios (conjuntos/clusters) locais de dados, que serão por sua vez, modelados por redes neurais distintas. A figura 4.4 apresenta o diagrama de classe para este módulo.

Importante se observar que, conforme [Marin, 02], neste processo a influência das variáveis exógenas, tais como temperatura, nebulosidade e outras, acaba sendo incorporada de modo indireto.

Este módulo possui duas versões de implementação, sendo uma delas por meio de uma rede neural auto-organizável de Kohonen (mapa SOM) [Kohonen, 90], cuja estrutura é composta de uma rede plana com formato quadrado, com 13 X 13 neurônios (169), podendo ser alterada, conforme solicitação de classificação, dependendo de quão

dispersos podem ser os perfis, ou seja, do número desejado de perfis finais (clusters) de carga. O resultado desta classificação constitui a separação dos dias em função dos perfis de carga.

Figura 4.4 Diagrama de Classe Módulo Classificador

A segunda implementação usa o algoritmo Fuzzy-Isodata, descrito em [Bastos, 04], para efetuar a classificação dos perfis de carga. As duas implementações permanecerão disponíveis no SISDEC por um período de testes e avaliações, permanecendo posteriormente no sistema apenas a que apresentar melhor desempenho.

Seja qual for a implementação utilizada, no fim do processo a massa de dados classificada é dividida, posteriormente, em duas bases de dados, uma utilizada como entrada das estruturas neurais durante a sua fase de treinamento, e a outra, para a fase de

testes pós-treinamento.

A classificação de perfis de carga é solicitada pelo usuário ao Gerenciador, que solicita ao Classificador, provedor de método e instância, as informações dinâmicas correspondentes aos perfis obtidos.

O Classificador solicita ao Gerenciador os dados históricos da carga, o qual verifica se os mesmos existem no repositório. Caso afirmativo estes dados solicitados são enviados ao provedor Classificador.

Uma vez tendo recebido os dados históricos, o módulo Classificador efetua a classificação dos mesmos em conjuntos com alta similaridade. Ao término desta tarefa, este provedor envia as informações resultantes ao Gerenciador, para que este as insira no repositório.

Caso não existam no repositório, dados históricos de carga para serem classificados, o provedor Classificador sinaliza o ocorrido, e o Gerenciador comunicará ao usuário a indisponibilidade do serviço.

Módulo Treinador

A fase de treinamento envolve a determinação, redução (regularização) e treinamento das redes associadas com cada uma dos perfis obtidos na fase de classificação, produzindo mínimos erros para o treinamento e a generalização.

A generalização de uma rede neural é a medida de quão bem a rede processa dados não pertencentes ao conjunto de dados usados no seu treinamento, o que depende consideravelmente do método de treinamento adotado [Zang, 98], da complexidade do problema a ser tratado e do número de amostras usado na fase de treinamento (quanto maior a rede, maior o conjunto de amostras a ser usado no seu treinamento e mais demorado este será) [Osowsk, i02].

Utilizam-se várias redes neurais, uma para cada perfil obtido no processo de classificação, uma vez que se observa, que os modelos de previsão de carga utilizando uma única rede neural não apresentam bons resultados quando tratando com todos os dias do ano, independentemente de seus perfis de carga [Marin, 02].

A determinação do tamanho ótimo de uma rede neural, também não é uma tarefa trivial, uma vez que com tamanho insuficiente, a mesma não poderá modelar adequadamente o problema a ser resolvido enquanto que ser for excessivamente grande, ela tenderá a “memorizar” os dados e não a “aprender” as características estatísticas do processo representado pelos mesmos [Osowski, 02].

Observa-se que, geralmente existe uma redundância considerável de informação contida nos pesos de uma rede neural totalmente conectada, podendo-se eliminar (regularizar) alguns destes, de modo a se diminuir o tempo necessário para o treinamento, e ainda assim manter-se a capacidade funcional necessária para solucionar-se o problema [Osowski, 02].

O treinamento de uma rede neural é solicitado pelo Usuário ao Gerenciador, o qual verifica no repositório CIM se as informações de perfil correspondente a esta rede neural existe no mesmo. Caso a informação seja encontrada no repositório, o Gerenciador solicita ao provedor Treinador, as informações da rede correspondente treinada.

O provedor Treinador, provedor de método e instância, solicita ao Gerenciador as informações de perfil, e ao recebe-las efetua o treinamento da rede considerada. Ao término do processo são efetuados testes para verificação da eficácia do treinamento efetuado, sendo as informações resultantes modeladas em CIM e enviadas ao Gerenciador, para que este as insira no repositório.

O Gerenciador, após receber as informações relativas à rede treinada e inseri-las no repositório, sinaliza para o Usuário a disponibilidade das informações.

O diagrama de classe para este módulo pode ser visto na figura 4.5.

Figura 4.5 Diagrama de Classe Módulo Treinador

Módulo Previsor

A previsão de carga é solicitada pelo Usuário ao Gerenciador, o qual verifica no repositório CIM se a carga solicitada já não existe no mesmo. Caso a carga seja encontrada no repositório ela será retornada ao usuário, e caso contrário, o Gerenciador envia ao Previsor, provedor de método e instância, a solicitação das informações desejadas pelo Usuário.

O Previsor efetua o processo de previsão, buscando através do Gerenciador as informações relativas à rede neural que é responsável pela previsão da carga da data

solicitada. Estas informações também são consultadas pelo Gerenciador no repositório CIM.

No caso das informações residirem no repositório, elas são enviadas ao provedor Previsor, o qual gera então a previsão da carga. Após a geração da previsão, o provedor envia as informações resultantes modeladas em CIM para o Gerenciador, que as insere no repositório e posteriormente as envia para o Usuário solicitante.

Caso não exista uma rede neural correspondente ao dia da carga solicitada, o provedor Previsor sinaliza o ocorrido, e o Gerenciador comunicará ao Usuário a indisponibilidade do serviço.

O diagrama de classe deste módulo pode ser visto na figura 4.6.

Figura 4.6 Diagrama de Classe Módulo Previsor

Módulo Gerenciador

As informações relativas aos perfis de carga classificados, à estrutura e aos parâmetros das redes neurais já treinadas, às cargas previstas e ao histórico de cargas são modeladas no padrão CIM e tornadas disponíveis, em qualquer ponto da corporação, como produtos das aplicações.

O acesso transparente a estas diversas informações, através do gerenciamento do repositório CIM pelo Gerenciador, caracteriza uma interoperabilidade entre vários aplicativos distribuídos e informações distintas em sua origem e tipicidade.

Este módulo é, portanto responsável pela integração funcional dos diversos módulos do SISDEC, sendo suas funções macros, para cada uma das funcionalidades:

Previsão de Carga

Ao receber uma solicitação de previsão de carga, este módulo verifica no Repositório CIM se já existe a carga solicitada. Caso afirmativo informa ao usuário, e solicita orientação no sentido de usar a carga prevista existente ou de fazer uma nova previsão.

Se a carga prevista for aceita, a mesma é enviada ao usuário, encerrando-se o processo, caso contrário, este módulo busca, no repositório CIM, a rede Neural Treinada correspondente ao dia desejado e a envia ao módulo Previsor para que este efetue a previsão.

Ao término da previsão, a Carga Prevista recebida do Módulo Previsor é enviada ao usuário solicitante.

Treinamento da Rede Neural

Ao receber uma solicitação de treino de uma estrutura neural, este módulo

dispara o Módulo Treinador, para que este efetue o treinamento solicitado.

Tendo a rede treinada, o Módulo Treinador a modela em CIM e envia-a para o Repositório e para o Módulo Gerenciador, o qual a envia para o usuário solicitante, encerrando o processo de treinamento.

Classificação de Perfis de Carga

Este Módulo, ao receber solicitação para se efetuar uma classificação de dados de carga, busca no Repositório CIM, os dados de carga necessários para a classificação.

Tendo estes dados, o Módulo Gerenciador envia-os para o Módulo Classificador, disparando o processo de classificação.

O resultado do processo de classificação é modelado em CIM, e enviado, tanto para o Repositório, quanto para o Módulo Gerenciador, que o repassará para o usuário solicitante.

4.2.4 Modelagem CIM

A modelagem CIM das informações perfil de carga, estruturas neurais e carga é apresentada a seguir.

Perfil de Carga

As informações relativas ao objeto Perfil de Carga são estruturadas de modo a representarem os diversos clusters (conjuntos) obtidos pelo processo de classificação.

Após classificação, os perfis devem conter as informações de identificação de cada dia pertencente ao cluster.

Esta informação, em cada perfil, tem a seguinte estrutura:

Dia Ano	Dia Semana	Tipo Dia	Horario Vero
---------	------------	----------	--------------

Dia Ano = Indica o dia em relação ao ano (1 a 366)

Dia Semana = Indica o dia em relação à semana (1 a 7, domingo a sábado)

Tipo Dia = Indica se o dia é feriado, véspera de feriado ou pós-feriado (1 a 3)

Horário Verão = Indica se o dia selecionado está em horário de verão (sim, não)

Estas são as informações a serem modeladas em CIM-MOF.

```
Class CteepSisdecPerfil: CIM_LogicalElement
{
  [key]
  uint8 IDperfil;
  uint8 diaAno[];
  uint8 diaSemana[];
  uint8 tipoDia[];
  boolean horaVerão[];
};
```

Rede Neural

As redes neurais apresentam a maior estrutura de informações necessárias para a sua correta caracterização.

A estrutura abriga as informações de número de camadas, quantidade de neurônios em cada camada, conexões e seus pesos correspondentes, tipo da rede, funções de agregação e ativação (e seus parâmetros) e identificação dos dias aos quais a rede foi treinada.

redeTipo	nPerfil	nCamada	nNeuronio	pesoConexao	tipoAtivacao	funcaoAgregacao
----------	---------	---------	-----------	-------------	--------------	-----------------

parAtivacao1	parAtivacao2
---------------------	---------------------

RedeTipo	=	Indica o tipo da rede. Perceptron = 1
nPerfil	=	Indica para quais perfis a rede está devidamente treinada
nCamada	=	Indica o número de camadas da rede
nNeurônio	=	Indica o número de neurônios em cada camada
pesoConexão	=	Indica os valores dos pesos da conexões de todos os neurônios da rede
tipoAtivação	=	Indica o tipo de função de ativação, sendo 1 = Sigmoid
parAtivação1	=	Indica o valor do primeiro parâmetro da função de Ativação
parAtivação2	=	Indica o valor do segundo parâmetro da função de Ativação
funcaoAgregacao	=	Indica o tipo de função de agregação, sendo 1 = Somatório.

Estas são as informações a serem modeladas em CIM-MOF.

```

Class CteepSisdecRedeNeural: CIM_LogicalElement
{
[key]
uint8 IDrede;
uint8 RedeTipo;
uint8 nPerfil[];
uint8 nCamada;
uint8 nNeurônio;
real32 pesoConexão[];
uint8 tipoAtivação;
real32 parAtivação1;
real32 parAtivação2;
uint8 funcaoAgregacao;
};

```

Carga

As informações, relativas ao objeto Carga Prevista, são estruturadas de modo a representarem os diversos valores de consumo diário de energia; previstos pelo processo de previsão.

A Carga Prevista é composta por 96 valores de energia, correspondentes a intervalos horários de 15 minutos dentro do dia considerado. A visualização destes valores em um gráfico representa a Curva de Carga Prevista.

A estrutura desta informação é, então, composta pelas informações de dia/mês/ano e os 96 valores de energia, correspondentes ao dia considerado.

Esta informação tem a seguinte estrutura:

Ano	Dia ano	Carga1	Carga2		Carga96
------------	----------------	---------------	---------------	--------------	----------------

Ano = Indica o ano, ao qual se refere o dia desejado

Dia Ano = Indica o dia em relação ao ano (1 a 366)

$Carga_n$ = Indica o valor da energia em cada intervalo de 15 minutos dentro do dia

Estas são as informações a serem modeladas em CIM-MOF.

```

Class CteepSisdecCarga: CIM_LogicalElement_
{
  [key]
  uint8 IDCarga;
  uint8 ano;
  uint8 diaAno;
  real32 carga[96];
};

```

4.2.5 Experimento

Para se verificar a eficácia da modelagem, foi efetuado um experimento, no qual são utilizados três computadores, PC compatíveis, interligados em um ramo local da rede corporativa cuja configuração SISDEC de aplicativos e informações pode ser visualizada na figura 4.7.

O protótipo implementado considera que as redes neurais estejam treinadas, os perfis de carga já determinados e que estas estruturas já estão armazenadas e disponíveis através dos provedores CIM.

Fica como verificação da eficácia da modelagem, o acesso às funcionalidades e informações disponíveis para os usuários clientes. Estas informações são perfis de carga e carga prevista, sendo as funcionalidades consideradas previsão da carga e consultas a perfil.

Figura 4.7 Configuração SISDEC para Testes

Os resultados obtidos podem ser extrapolados para as funcionalidades: treinamento das redes neurais, classificação de perfis, consultas a carga histórica e redes treinadas.

O ambiente de testes montado para verificação da modelagem proposta pode ser visto na figura 4.8, com a seguinte descrição funcional.

Figura 4.8 Ambiente de Testes

Foi efetuada a solicitação de carga prevista e perfil, conforme os seguintes casos de uso:

UC01:

Solicitar previsão de carga para três redes neurais localizadas em máquinas distintas (treinadas para o mesmo perfil)

Considerar apenas trechos da carga prevista em duas das redes neurais

Compor a carga prevista com estes trechos obtidos

Comparar a carga prevista montada com carga prevista completa

UC02:

Solicitar perfil de carga, existente no Repositório da máquina local

A figura 4.9 apresenta uma das telas de usuário do SISDEC, onde se selecionou dentre as diversas cargas previstas disponíveis no(s) Repositório(s) a do dia 26 de setembro de 2005. A comparação entre as cargas previstas (máquina local e máquinas remotas) foi efetuada após a exportação (opção oferecida pelo SISDEC) dos dados de carga para uma planilha Excel.

Os resultados apresentados na figura 4.10 confirmam, para o caso de uso UC01, que a previsão de carga, buscando dados localizados em máquinas distintas apresenta-se funcional em relação à previsão efetuada pela rede localizada próxima ao usuário.

Figura 4.9 Tela de Usuário do SISDEC – Funcionalidade Carga

Figura 4.10 Resultado do Experimento para o caso de uso UC01

Para o caso de uso UC02, dados existentes no Repositório foram obtidos de forma transparente para o usuário. Observa-se na figura 4.11 as curvas de carga obtidas ao se solicitar o perfil relativo às segundas feiras (Sem Horário de Verão, Não feriados e meses: Abril, Maio, Junho, Julho, Agosto, Setembro e Outubro – 2002 a 2005).

Figura 4.11 Gráfico do Perfil obtido no caso de uso UC02.

Uma vez que o funcionamento do SISDEC está baseado na troca de informações, o protótipo implementado verificou também o resultado final em relação ao usuário, do fluxo de informações, conforme várias possibilidades de combinação Usuário-CIMOM-Provedores.

Os provedores são responsáveis pela disponibilidade das informações solicitadas, efetuando a interface entre os dados e o CIMOM. Nesta implementação eles constituem processos que ficam aguardando solicitações dos usuários repassadas pelo CIMOM.

O seguinte fluxo de informação foi considerado nos testes efetuados.

Usuário solicita

perfil de carga

provedor local

CIMOM local envia a solicitação do usuário ao provedor.

Provedor passa ao CIMOM local a informação solicitada.

CIMOM local repassa ao usuário, a informação solicitada.

provedor remoto

CIMOM local envia a solicitação do usuário ao CIMOM da máquina remota, onde o provedor necessário se encontra.

Provedor remoto passa ao CIMOM remoto a informação solicitada.

CIMOM repassa ao CIMOM local, que envia para o cliente a informação solicitada.

carga prevista

provedor local

CIMOM local envia a solicitação do usuário ao provedor.

Provedor passa ao CIMOM local a informação solicitada.

CIMOM local repassa ao usuário, a informação solicitada.

provedor remoto

CIMOM local envia a solicitação do usuário ao CIMOM da máquina remota, onde o provedor necessário se encontra.

Provedor passa ao CIMOM a informação solicitada.

CIMOM remoto repassa ao CIMOM local, que comunica ao cliente a informação solicitada.

A figura 4.12 retrata os resultados dos testes efetuados conforme o fluxo de informação descrito.

Informação Distribuída	Funcionalidade Distribuída	Transparência
Perfil de Carga	Buscar Perfil	Localização e Acesso
Carga Prevista	Previsão de Carga	Localização e Acesso

Figura 4.12 Resultado dos Testes

Desta figura, observa-se que dentro do contexto do usuário, a utilização da tecnologia WBEM propicia as características de distribuição e transparência de localização das informações: carga prevista e perfil de carga, assim como das funcionalidades previsão de carga e consultas de perfil. O acesso à funcionalidade e aos dados independe da localização do mesmo, enquanto que a utilização de dados modelados em um modelo comum permitiu a interoperabilidade.

A transparência de acesso tanto às funcionalidades, quanto às informações do SISDEC, são suportadas tanto pela infraestrutura de redes (meios físicos, serviços de protocolos e sistemas operacionais de rede), como pelos processos de segurança (perfis de acesso, senhas) já existentes no ambiente corporativo de Tecnologia de Informação.

5. CONSIDERAÇÕES FINAIS

Este capítulo apresenta considerações baseadas nos resultados do trabalho efetuado, propõem-se novos trabalhos.

O trabalho efetuado é considerado um passo inicial, uma vez que se usou, na sua implementação, um Gerenciador de objetos CIM, cujo código não é aberto. Atualmente, existem gerenciadores CIMOM com código aberto, em versões comerciais e na modalidade de software livre.

5.1 CONCLUSÕES

A implementação do protótipo, resultante deste trabalho, confirma a viabilidade da modelagem CIM na área corporativa de uma empresa de Energia Elétrica, assim como valida a proposta da arquitetura distribuída.

O acesso, de modo transparente da localização, às informações e funcionalidades do SISDEC distribuídas pela rede de computadores, é possibilitado pela arquitetura proposta validando a modelagem efetuada neste trabalho.

O trabalho atende aos estudos iniciais de utilização de modelos comuns para troca de informações em ambientes heterogêneos, dentro da corporação.

Observa-se que a integração das informações foi efetuada sem impactos negativos nos processos dos Sistemas da Operação da Transmissão de Energia, assim como nos demais Sistemas utilizados pela empresa.

A utilização de um gerenciador de objetos CIM (CIMOM) permitiu uma maior flexibilidade e funcionalidade à arquitetura distribuída originalmente proposta, que se restringia à modelagem da informação, apenas. Esta maior flexibilidade e funcionalidade decorrem da utilização do CIMOM como broker entre os vários módulos

e usuários do SISDEC.

Os arquivos MOF, obtidos na implementação do SISDEC, permitirão a portabilidade para outras plataformas de software e hardware, possibilitando a realização de testes comparativos, qualitativos e quantitativos de desempenho das várias implementações WBEM disponíveis atualmente.

Adicionalmente ressalta-se que o WMI não tem aderência total à padronização WBEM, uma vez que não utiliza CIM-XML sobre HTTP, conforme preconizado pela DMTF, para codificação e transporte entre Sistemas WBEM. A tecnologia WMI utiliza COM/DCOM em substituição ao CIM-XML sobre HTTP.

5.2 CONTRIBUIÇÕES ACADÊMICAS

O conhecimento aprendido durante o processo de modelagem e da implementação fornecerá subsídios aos estudos de aplicabilidade do padrão de modelagem CIM nos processos de gerenciamento do sistema elétrico de potência. Adicionalmente aponta a viabilidade do uso da WBEM como tecnologia provedora de distribuição e interoperabilidade, dentro do ambiente corporativo.

5.3 TRABALHOS FUTUROS

Trabalhos posteriores serão elaborados visando o mapeamento das informações MOF para XML [McMorran, 04], juntamente com a utilização de implementações WBEM outras que o WMI, o que permitirá uma maior facilidade na transferência de informações entre aplicativos distintos, resultando em uma maior interoperabilidade.

Observa-se que o uso de XML como transporte de informações em várias aplicações distintas, já é bem difundido [deVos, 01][McMorran, 04][Huang, 03] [Widergren, 99], [Oliveira, 05], [Singhal, 05], tendo já sido publicados pela DTMF, os documentos especificando a representação CIM em XML e as operações CIM sobre HTTP .

A utilização de outras implementações WBEM deverá ser estudada uma vez que neste trabalho se usou um Gerenciador de objetos CIM, cujo código não é aberto, e não foram efetuadas comparações com outros gerenciadores. Atualmente, existem gerenciadores CIMOM com código aberto, em versões comerciais e na modalidade de software livre.

A modelagem e implantação de um protótipo, buscando a inserção do SISDEC como uma tarefa dentro do middleware CIM, no alinhamento com a modelagem de processos no CIM, conforme a proposta de [Kumar, 05], apresenta-se como uma vertente a ser trabalhada. Adicionalmente esta inserção, conforme o autor, desponta como um modo de se conseguir reuso de código.

Por outro lado, a utilização de dados de tempo real, obtidos do Sistema Elétrico de Potência, poderia ser futuramente considerada na correção automática da curva de carga prevista pelo SISDEC, utilizando-se uma interface CIM, como middleware.

Ainda considerando a modelagem CIM, o enfoque do gerenciamento de mudanças de aplicativos (software) instalados pode ser desenvolvido fazendo-se também o mapeamento do SISDEC no objeto Product definido no Core Model CIM, o qual possibilita a definição, consistência e a disponibilidade das várias informações necessárias a este gerenciamento centralizado.

ANEXO A

WBEM - Web-Based Enterprise Management

O WBEM surgiu na década de 90, definida por um consórcio de empresas (BMC Software, Cisco Systems, Compaq, Intel e Microsoft), como um modo de se gerenciar computadores “desktop”, evoluindo posteriormente para uma ferramenta de gerenciamento de uso geral e com um número crescente de implementações [Hobbs, 04]. A partir de 1998, a responsabilidade de manter a WBEM foi passada para a DMTF.

O objetivo foi o desenvolvimento de uma arquitetura aberta para o gerenciamento, baseado na WEB, de toda a infra-estrutura corporativa de uma organização [Teixeira, 00].

WBEM compreende um conjunto de tecnologias padronizadas (Figura A.1), que incluem [Hobbs, 04]:

Figura A.1 – Relacionamento entre as tecnologias padrão que compõem o WBEM

- CIM linguagem e metodologia para descrever dados gerenciais [Lee, 04]. É um modelo orientado a objeto para descrição de informações de gerenciamento (modelagem dos aspectos e comportamentos dos objetos gerenciados) [Teixeira, 00], que inclui um conjunto de modelos padrões (esquemas) para

sistemas, aplicações, redes, dispositivos e outros componentes, expressos na linguagem MOF [Hobbs, 04], a qual fornece uma representação textual das informações modeladas pelo CIM [Teixeira, 00].

Define, por meio do CIM Operations, um conjunto extensível de operações permitidas para os objetos gerenciados.

- **XmlCIM** especificação de codificação, para codificar comandos e respostas, a qual pode ser utilizada para representar entidades WBEM [Lee, 04]. É uma especialização do XML – eXtensible Markup Language, que define a codificação que os clientes e servidores CIM utilizam para comunicação sobre objetos CIM, e as operações que podem ser efetuadas com os mesmos [HP, 04].
 - A definição dos elementos XML na DTD (Document Type Definition)
 - A representação do CIM na especificação XML
- **HTTP** HyperText Transfer Protocol é utilizado para transportar mensagens xmlCIM pela rede de computadores, constituindo o mecanismo de transporte para comandos e respostas através da rede [Hobbs, 04]. É um protocolo do nível de aplicação que implementa conceitos de orientação a objetos genéricos, permitindo que sistemas distribuídos sejam construídos independentemente dos dados a serem transferidos [Teixeira, 00].
Ao definir o mapeamento das operações CIM em HTTP, o WBEM permitiu às implementações CIM interoperarem de modo padronizado e aberto [Lee, 04].

A arquitetura lógica do WBEM (Figura A.2) é composta por vários componentes [Carrião, 04], sendo uma arquitetura padronizada para sistemas de gerenciamento hierarquizados, incluindo a definição do servidor WBEM e suas interfaces. Nesta arquitetura, o servidor WBEM atua como um broker entre o usuário e os complexos sistemas que o mesmo utiliza [Hobbs, 04].

Um servidor WBEM pode operar diretamente sobre o sistema chamando funções, serviços e/ou comandos do próprio sistema, ou passando as solicitações dos clientes diretamente para os módulos Provedores. Ele mantém as informações necessárias para atendimento a estas solicitações dos clientes, assim como as informações necessárias ao seu próprio funcionamento, em um tipo de base de dados denominada Repositório [HP, 03].

A interface entre o servidor WBEM e os provedores é denominada API - Provider Application Programming Interface. O servidor chama as funções da API do provedor quando ele necessita que o provedor efetue uma operação solicitada por um cliente.

Os Provedores efetuam a conversão entre CIM e os comandos, serviços e formatos de dados específicos da plataforma utilizada, uma vez que as aplicações clientes WBEM tratam apenas com representação CIM.

O cliente WBEM requisita acesso aos objetos gerenciados, através de uma operação XML, que é traduzida pelo adaptador de protocolo de XML para CIM e entregue ao Gerenciador de Objetos CIM (CIMOM).

Figura A.2 - Arquitetura lógica de servidor WBEM

O CIMOM constitui o componente central do servidor WBEM, sendo responsável pelo tratamento das solicitações feitas pelo cliente e pela comunicação com os componentes

do servidor [Carrião, 04], enquanto que o Manipulador de Indicação WBEM efetua o roteamento e a entrega de mensagens do CIMOM para os respectivos destinos.

O gerenciador de objetos CIM responde apenas às operações definidas na especificação CIM, tais como criar, modificar e excluir, assim como também verifica a sintaxe e semântica das mensagens, além de proporcionar infraestrutura de segurança às operações efetuadas [Lee, 04]. O CIMOM direciona aos provedores as ações referentes a dados dinâmicos, e no caso de dados estáticos efetua por si mesmo as ações necessárias em seu repositório.

O Repositório do CIMOM armazena todos os esquemas CIM, que servem para verificação dos dados enviados pelos clientes e provedores e, pode também ser utilizado para a criação de instâncias de dados CIM a partir de clientes ou provedores, sendo estas instâncias, neste caso, denominadas dados estáticos [Carrião, 04].

Os dados dinâmicos são aqueles coletados pelos provedores a partir de elementos gerenciados. Os provedores representam os agentes de gerenciamento externos ou o interno (CIM), que monitoram elementos gerenciados.

ANEXO B

Mapa Auto Organizável de Kohonen

O Mapa de Kohonen constitui um método não hierárquico de estatística multivariada utilizado para organizar um conjunto de N objetos em subconjuntos mutuamente exclusivos (clusters), de forma que objetos em um mesmo subconjunto sejam semelhantes entre si, mas diferentes de objetos pertencentes aos outros subconjuntos [Velasquez, 01].

A estrutura do mapa de Kohonen consiste de uma camada de neurônios altamente interconectados, de modo que cada neurônio está conectado com todos os outros neurônios desta camada (inibição lateral) e com cada um dos neurônios da camada de entrada, a qual faz o papel de um buffer de entrada, figura B.1.

O algoritmo de Kohonen [Kohonen, 90] é um eficiente processo de auto-organização, o qual produz, de modo não supervisionado, a distribuição de um espaço de entrada V_{in} em um outro espaço de menor dimensão V_{out} , preservando as relações topológicas entre os vetores de entrada [Marin, 02]. Seu objetivo final é efetuar uma transformação adaptativa e ordenada de um conjunto de dados de entrada em um mapa bi-dimensional de saída.

Figura B.1 Mapa de Kohonen

O espaço de saída é, portanto, representado por uma matriz bidimensional de neurônios, apresentando relacionamento de vizinhança, de modo que vetores de entrada similares ativem neurônios vizinhos.

Durante o processo de aprendizado, cada vetor de entrada X é comparado com os vetores de peso W_i de cada neurônio na rede. O neurônio cujo vetor peso for o mais próximo do vetor X será selecionado, modificando o seu peso e os dos seus vizinhos, obedecendo a equação B.1:

$$W_i(t+1) = W_i(t) + l_c h_{iv}(X - W_i(t)) \quad (B.1)$$

O termo l_r representa a taxa dinâmica de aprendizado, a qual altera-se durante o aprendizado, conforme a equação B.2:

$$l_c(t) = \frac{l_{ro}}{(1 + \frac{c \times t}{n})} \quad (B.2)$$

Onde:

l_{r0} é a taxa inicial de aprendizado; c é uma constante, t é a iteração atual e n o numero de neurônios.

A função de vizinhança $h_{iv}(x)$ determina o incremento do peso de cada neurônio como uma função de proximidade ao neurônio selecionado, conforme equação. B.3.

$$h_{iv}(x) = \exp\left(\frac{(-|x_c - x_i|^2)}{(2 \times sig^2)}\right) \quad (B.3)$$

Onde:

x_c é a posição do neurônio selecionado,
 x_i é a posição dos demais neurônios na rede,
sig é a área de ativação.

No início do processamento, a área de ativação é ampla, de modo a abranger muitos neurônios da rede, resultando, portanto na ativação de neurônios em uma vasta área em torno do neurônio selecionado.

Conforme o aprendizado evolui, a área de ativação é constantemente diminuída de modo que, apenas os neurônios mais próximos ao centro da área de ativação são influenciados pelo neurônio mais ativado.

O processo termina quando o tamanho da área de ativação for menor que um determinado valor desejado. O resultado é um tipo de mapa, onde neurônios com funções similares estarão agrupados. Pode-se efetuar, deste modo, a classificação dos valores de entrada uma vez que valores similares de entrada serão representados em áreas específicas do mapa da rede, formando classes ou agrupamentos (clusters).

Portanto, a localização espacial dos neurônios auto-organizados indicará características intrínsecas aos padrões de entrada.

ANEXO C

Algoritmos Genéticos

Algoritmos Genéticos representam uma classe de algoritmos de otimização que empregam um mecanismo de pesquisa probabilístico de soluções, baseando-se em um processo de evolução artificial (semelhante ao biológico) para combinação e seleção de soluções para o problema em análise [Bento, 04] [Fregni, 97].

Dado um problema, é produzida aleatoriamente uma grande quantidade de candidatos (população) à solução do mesmo. Cada uma destas soluções é mapeada no cromossomo de um indivíduo, sendo, portanto cada população um subconjunto do espaço de soluções do problema em estudo.

Estas soluções são, em seguida, avaliadas quanto à sua capacidade de resolver este problema. As melhores soluções são selecionadas para reproduzirem a nova população de candidatos à solução almejada, de modo que em cada nova geração um novo elemento é criado combinando-se uma unidade de informação com as partes mais adaptadas dos indivíduos da geração anterior.

Os Algoritmos Genéticos utilizam operadores de seleção, mutação e cruzamento, sobre os cromossomos, de modo que os cromossomos com melhores aptidões (melhor resposta ao problema) serão encontrados ao fim deste processo [Fregni, 97].

Procura-se, então, a cada iteração (geração) encontrar-se novos pontos de pesquisa onde são esperados melhores desempenhos.

Ao tratar com uma população de indivíduos bem adaptados estes algoritmos obtém segurança na quantidade, minimizando desta forma, a possibilidade de se atingir um falso ponto ótimo (solução local), figura C.1.

Figura C.1 Mínimos, Local e Global

Apesar de usarem processos de troca de informação randômicos, os Algoritmos Genéticos não constituem um algoritmo de pesquisa aleatória, uma vez que utilizam as informações presentes em cada indivíduo para se investigar novos pontos de pesquisa, objetivando a maximização do rendimento.

Parâmetros Genéticos

O comportamento dos Algoritmos Genéticos é influenciado por alguns parâmetros, e de modo a defini-los conforme as necessidades do problema em estudo e dos recursos disponíveis, é necessária uma análise dos mesmos.

Tamanho da População

Este parâmetro determina a quantidade de cromossomos na população, afetando diretamente o desempenho global e a eficiência do algoritmo.

Uma grande população fornece uma maior cobertura do domínio do problema, diminuindo a possibilidade de convergências para mínimos locais, mas exigindo maiores recursos computacionais e/ou tempos de processamento.

Por outro lado, uma pequena população não constitui uma representativa cobertura do espaço de pesquisa do problema, influenciando negativamente o desempenho do algoritmo.

Taxa de Cruzamento

Este parâmetro determina a velocidade com a qual novos cromossomos serão introduzidos na população.

Com altas taxas de cruzamento, a maior parte da população será substituída, podendo ocorrer perda de cromossomos apresentando alta aptidão, enquanto que com baixas taxas, o algoritmo pode tornar-se muito lento.

Tipo de Cruzamento

Determina a forma a ser utilizada para se efetuar a troca de segmentos de informação entre os pares de cromossomos selecionados para cruzamento.

Taxa de Mutação

Define a probabilidade de ocorrência de uma mutação nos cromossomos, seu valor ótimo depende da aplicação considerada.

A mutação é necessária para se criar nova informação para a população e prevenir a saturação, da mesma, com cromossomos semelhantes, fato caracterizado pela convergência prematura.

Baixos valores de mutação evitam a estagnação dos cromossomos, possibilitando adicionalmente o acesso a qualquer ponto no espaço de pesquisa do problema.

Com valores excessivos, a busca torna-se essencialmente aleatória, aumentando adicionalmente a possibilidade de que soluções ótimas sejam destruídas.

Operadores Genéticos

São os elementos responsáveis pelas transformações impostas à população durante o ciclo de cada geração.

Seleção

Constitui o processo de adaptação e sobrevivência dos cromossomos ao oferecer aos melhores indivíduos da população atual, preferência para o processo de reprodução, permitindo que estes indivíduos possam passar as suas características às próximas gerações.

Cruzamento

O objetivo deste operador é efetuar a propagação das características dos indivíduos mais aptos da população através da troca de segmentos de informações entre um par de indivíduos selecionados, resultando na criação de novos indivíduos.

Consideraremos três tipos de cruzamento: um ponto, dois pontos e uniforme.

O cruzamento tipo um ponto caracteriza-se pela escolha aleatória do ponto de quebra do cromossomo, e a partir deste ponto efetua-se a troca de genes entre os dois indivíduos (figura C.2).

Figura C.2 Cruzamento do tipo de um ponto

O cruzamento de dois pontos é similar ao de um ponto, mas são escolhidos aleatoriamente dois pontos de quebra para efeito de troca de genes (figura C.3).

Figura C.3: Cruzamento do tipo dois pontos

O cruzamento uniforme utiliza uma máscara de cruzamento gerada aleatoriamente, de modo que cada gene do descendente é criado copiando o gene de um dos pais, baseado no valor do gene correspondente na máscara de cruzamento (figura C.4).

Após gerar um descendente os pais são trocados de lado e a mesma máscara é novamente aplicada gerando o segundo descendente. A cada par de pais, uma nova máscara de cruzamento deverá ser criada.

Figura C.4 Cruzamento linear

Mutação

Este operador é responsável pela introdução e manutenção da diversidade genética na população, assegurando desta forma que a probabilidade de alcançar qualquer ponto do espaço de busca nunca seja igual a zero.

A mutação é aplicada aos indivíduos com uma probabilidade dada pelo parâmetro denominado taxa de mutação.

ANEXO D

Computação Fuzzy

Zadeh [Zadeth, 65] define formalmente conjuntos compostos de classes de objetos com funções de pertinência contínuas, conjuntos estes denominados por ele, como conjuntos fuzzy.

Ao definir estes conjuntos, Zadeh buscou definir um conceito que possa ser utilizado para tratar formalmente as classes de objetos encontrados no mundo real.

Um método formal para lidar com este tipo de classificação seria de suma importância, uma vez que, os seres humanos em seus processos de abstração, de análise, de comunicação e de reconhecimento de padrões, utilizam tais classes imprecisamente (ou vagamente) definidas.

Zadeh propôs, de modo preliminar, um conceito (os conjuntos fuzzy) que poderia se utilizado para se tratar com estas classes vagamente definidas.

Importante observar que, este conceito conforme desenvolvido por Zadeh, representa uma generalização do conceito usual de conjuntos, o qual se torna um caso particular do mesmo.

A definição formal de um conjunto fuzzy é:

“Conjunto fuzzy A em x é caracterizado por uma função de pertinência $f_r(x)$, a qual associa a cada ponto em x um número real no intervalo $[0,1]$, com o valor da função $f_r(x)$ em x representando o grau de pertinência de x em A” [Zadeth, 65].

Nota-se que nos casos em que A é um conjunto, no sentido usual do termo, sua função de pertinência pode tomar apenas dois valores (1 ou 0), com $f_r(x)=1$ ou $f_r(x)=0$ representando o fato do objeto x, em estudo, pertencer ou não ao conjunto A.

Zadeh observa e demonstra que a função de pertinência não tem caráter probabilístico e que a noção de conjuntos fuzzy é completamente não estatística por natureza.

As definições usuais dos conjuntos são estendidas para os conjuntos fuzzy, tais como: conjunto vazio, igualdade de conjuntos, conjunto complemento, união de conjuntos, interseção de conjuntos, etc; assim como se definem operações algébricas sobre este tipo de conjunto.

Com esta formulação propôs-se um conceito que resulta em um modo natural de se tratar com os problemas nos quais a fonte de imprecisão caracteriza-se mais pela ausência de critérios bem definidos de classificação e não tanto pela presença de variáveis de natureza aleatória, o que é o caso do problema de classificação dos perfis de carga.

Algoritmo Fuzzy-Isodata

O algoritmo Fuzzy-Isodata, também referenciado com C-Means, é um método de classificação utilizando computação fuzzy, o qual objetiva encontrar partições fuzzy, a partir de um conjunto de dados de treinamento, através de uma generalização nebulosa da função de mínimos quadráticos.

Esta generalização consiste na otimização de uma função objetivo de modo a organizar um conjunto de N objetos, de dimensão p , em subconjuntos, denominados clusters, de tal modo que os objetos em um mesmo subconjunto sejam semelhantes entre si, mas sejam diferentes, em algum grau, de outros objetos pertencentes a outros clusters [Miyamoto, 98]. Baseia-se na medida de similaridade de modo a caracterizar o grau de semelhança entre os objetos [Velasquez, 01].

Função Objetivo

Função objetivo do algoritmo Fuzzy-Isodata

$$J_m(\mathbf{U}; \mathbf{Y}) = \sum_{k=1}^n \sum_{j=1}^c (u_{jk})^m E_j(\mathbf{x}_k)$$

Onde:

- conjunto de treinamento, contendo n amostras não classificadas

$$\mathbf{X} = \{ \mathbf{x}_k \mid k \in [1; n] \}$$

- conjunto contendo os centros dos clusters

$$\mathbf{Y} = \{ \mathbf{y}_j \mid j \in [1; c] \}$$

- medida de dissimilaridade (distorção) entre a amostra $\mathbf{x}_k$ e o centro $\mathbf{y}_k$ do cluster j especificado

$$E_j(\mathbf{x}_k)$$

Usada distância Euclidiana:

$$E_j(\mathbf{x}_k) = \| \mathbf{x}_k - \mathbf{y}_j \|^2$$

- matriz, $c \times n$, de partições fuzzy c-partition, contendo os valores de pertinência de todas as amostras em relação a todos os clusters

$$\mathbf{U} = [u_{jk}]$$

- parâmetro que controla o grau de nebulosidade (fuzziness)

$$m \in (1 ; \infty)$$

A minimização da função J_m , sob a restrição probabilística

$$\sum_{j=1}^c (u_{jk}) = 1$$

pode ser efetuada com a iteração das fórmulas:

para $E_j(x_k) > 0$

$$y_j = \frac{\sum_{j=1}^c (u_{jk})^m \cdot x_k}{\sum_{j=1}^c (u_{jk})^m} \quad u_{jk} = \left(\sum_{l=1}^c \left(\frac{E_l(x_k)}{E_l(x_k)} \right)^{\frac{2}{m-1}} \right)^{-1}$$

para $E_j(x_k) = 0$

$$y_j = \frac{\sum_{j=1}^c (u_{jk})^m \cdot x_k}{\sum_{j=1}^c (u_{jk})^m} \quad u_{jk} = 1$$

A justificativa do uso deste algoritmo vem da constatação de que os sistemas fuzzy são aderentes à solução do problema de determinar similaridades em massas imensas de dados, uma vez que estes dados possam ser representados por expressões linguísticas condicionais que caracterizem suas propriedades. Havendo disponibilidade suficiente de dados históricos, as similaridades existentes podem ser identificadas.

ANEXO E

Casos de Uso do SISDEC

O SISDEC é caracterizado pelos casos de uso a seguir apresentados, juntamente com os seus diagramas de seqüência. Esta caracterização engloba os modelos utilizados nos pontos de vista ODP Empresa e Informação.

O modelo global de casos de uso é mostrado na figura E.1, o qual norteia toda a modelagem posterior.

Figura E.1 Modelo de Casos de Uso para o SISDEC

UC01: Solicitar Carga Histórica

O usuário poderá:

- Listar a relação de cargas disponíveis
- Efetuar a pesquisa (busca) de uma carga específica
- Requisitar carga não relacionada no repositório
- Guardar carga histórica no repositório

Figura E.2 Caso de Uso UC01: Solicitar Carga Histórica

UC01.1: Listar Relação de Carga

O usuário solicita a relação de cargas históricas disponíveis no repositório.

Figura E.3 Caso de Uso UC01.1:Listar Relação de Carga

Para este caso de uso, tem-se como requisito a existência de uma relação de cargas históricas disponíveis no repositório.

Figura E.4 Requisito para o Caso de Uso Listar Relação de Carga

UC01.2: Pesquisar Carga Histórica

O usuário fornece o critério de filtro (dia/mês/ano) para acesso às informações de carga desejada.

Figura E.5 Caso de Uso UC01.2: Pesquisar Carga Histórica

O requisito para o UC01.2 é a existência da carga solicitada armazenada, figura E.6.

Figura E.6 Requisito para o Caso de Uso Pesquisar Carga Histórica

UC01.3: Requisitar Carga Histórica

O usuário fornece o critério de filtro (dia/mês/ano) para acesso às informações de carga desejada, não existente no repositório. O módulo Gerenciador busca no Middleware CTEEP a informação solicitada, sendo a existência desta informação um requisito para este caso de uso, figura E.8.

Figura E.7 Caso de Uso UC01.3: Requisitar Carga histórica

Figura E.8 Requisito para o Caso de Uso Requisitar Carga Histórica

UC01.4: Guardar Carga Histórica

O usuário solicita o armazenamento da carga histórica no repositório, tendo como requisito a existência de uma carga histórica obtida pelo UC01.3, figura E.10.

Figura E.9 Caso de Uso UC01.4: Guardar Carga Histórica

Figura E.10 Requisito para o Caso de Uso Guardar Carga Histórica

UC02: Solicitar Perfil

O usuário poderá:

- Listar a relação de perfis disponíveis
- Efetuar a pesquisa (busca) de um perfil específico

Figura E.11 Caso de Uso UC02: Solicitar Perfil

UC02.1: Listar Relação Perfil

O usuário solicita a relação de perfis disponíveis no repositório.

Figura E.12 Caso de Uso UC02.1: Listar Relação Perfil

Como requisito tem-se a existência de perfis disponíveis no repositório.

Figura E.13 Requisito para o Caso de Uso Listar Relação Perfil

UC02.2: Pesquisar Perfil

O usuário pesquisa (busca) um perfil específico.

Figura E.14 Caso de Uso UC02.2: Pesquisar Perfil

Requisito: existência do perfil disponível no repositório

Figura E.15 Requisito para o Caso de Uso Pesquisar Perfil

UC03: Classificar Perfil

O usuário poderá:

- Informar dias
- Requisitar classificação

Figura E.16 Caso de Uso UC03: Classificar Perfil

UC03.1: Informar dias

O usuário fornece a relação dos dias a serem classificados.

Figura E.17 Caso de Uso UC03.1: Informar Dias

Requisito: dias informados não podem estar já classificados e devem existir no repositório.

Figura E.18 Requisito para o Caso de Uso Informar dias

UC03.2: Requisitar classificação

O usuário solicita a classificação dos dias informados.

Figura E.19 Caso de Uso UC03.2: Requisitar classificação

Este caso de uso tem como requisito os dias fornecidos pelo usuário, no caso de uso UC03.1.

Figura E.20 Requisito para o Caso de Uso Requisitar classificação

UC04: Treinar Rede

O usuário poderá:

- Treinar rede neural

Figura E.21 Caso de Uso UC04: Treinar Rede

O usuário solicita o treinamento de uma rede neural.

Figura E.22 Caso de Uso UC04 diagrama seqüência

Tem como requisito a existência, no repositório, do perfil a ser utilizado.

Figura E.23 Requisito para o Caso de Uso Treinar rede

UC05: Solicitar Rede Neural

O usuário poderá:

- Listar relação de redes disponíveis

- Efetuar a pesquisa (busca) de uma rede neural específica
- Requisitar uma rede desejada

Figura E.24 Caso de Uso UC05: Solicitar Rede Neural

UC05.1: Listar relação de redes

O usuário solicita a relação de redes disponíveis no repositório.

Figura E.25 Caso de Uso UC05.1: Listar relação de redes

Requisito a ser cumprido para este caso de uso é a disponibilidade de redes no repositório.

Figura E.26 Requisito para o Caso de Uso listar relação redes

UC05.2: Pesquisar rede

O usuário pesquisa (busca) uma rede neural específica.

Figura E.27 Caso de Uso UC05.2: Pesquisar rede

Requisito a ser cumprido para este caso de uso é a existência da rede pesquisada, no repositório.

Figura E.28 Requisito para o Caso de Uso Pesquisar rede

UC05.3: Requisitar rede

O usuário solicita a rede desejada.

Figura E.29 Caso de Uso UC05.3: Requisitar rede

A rede desejada deve estar disponível no repositório, sendo sua existência o requisito a ser obedecido para este caso de uso.

Figura E.30 Requisito para o Caso de Uso Requisitar rede

UC06: Solicitar Previsão

O usuário poderá:

- Listar relação de previsões disponíveis
- Efetuar pesquisa (busca) de uma determinada previsão
- Requisitar uma previsão específica
- Fazer a previsão

Figura E.31 Caso de Uso UC06: Solicitar Previsão

UC06.1: Listar relação de Previsão

O usuário solicita a relação de previsões disponíveis no repositório.

Figura E.32 Caso de Uso UC06.1: Listar relação de previsão

A existência de previsões disponíveis no repositório é um requisito para este caso de uso.

Figura E.33 Requisito para o Caso de Uso Listar relação previsão

UC06.2: Pesquisar previsão

O usuário solicita a busca de uma determinada previsão desejada.

Figura E.34 Caso de Uso UC06.2: Pesquisar previsão

Tem como requisito a existência da previsão no repositório.

Figura E.35 Requisito para o Caso de Uso Pesquisar previsão

UC06.3: Requisitar Previsão

O usuário efetua a solicitação da previsão específica.

Figura E.36 Caso de Uso UC06.3: Requisitar previsão

Seu requisito é a existência da previsão no repositório.

Figura E.37 Requisito para o Caso de Uso Requisitar previsão

UC06.4: Fazer Previsão

O usuário solicita a previsão da carga do dia seleccionado.

Figura E.38 Caso de Uso UC06.4: Fazer previsão

O seu requisito é a existência no repositório, da rede neural específica para o dia desejado.

Figura E.39 Requisito para o Caso de Uso Fazer previsão

REFERÊNCIAS BIBLIOGRÁFICAS

- [Aboul, 01] Aboul-Magd, M.A., Ahmed, E. E. E., An Artificial Neural Network Model For Electrical Daily Peak Load Forecasting With An Adjustment For Holidays, Power Engineering, 2001. LESCOPE '01. 2001 Large Engineering Systems Conference on, 2001 pp. 93-97
- [Anderson, 00] Anderson, L., Brand, K.P., The Benefits of the Coming Standard IEC61850 for Communication in Substations, Southern African power System Protection Conference, Johannesburg, 8-9 Nov 2000.
- [Bastos, 04] Bastos, M. R., Andrade, M.T.C, Becerra, J.L.R., Uso da Computação Fuzzy na Classificação do Perfil de Carga de Um Sistema Elétrico de Potência, Congresso e Exposição Internacional de Automação - CONAI-2004 São Paulo, Brasil.
- [Becerra, 04] Becerra, J.L.R. et al, Arquitetura de um Middleware Corporativo na Companhia de Transmissão de Energia Elétrica Paulista, Projeto P&D 2004.
- [Becerra, 98] Becerra, J.L.R., Aplicabilidade do Padrão de Processamento Distribuído e Aberto nos Projetos de Sistemas Abertos de Automação, Tese de Doutorado – Escola Politécnica da Universidade de São Paulo, 1998.
- [Becker, 00] Becker, D. et al, Standards-Based Approach Integrates Utility Applications, Computer Applications in Power, IEEE , Volume: 13 , Issue: 4 , Oct. 2000 Pages:13 – 20

- [Bento, 04] Bento, E. P., Kagan, N., Algoritmos Genéticos e Variantes na Solução de Problemas de Configuração de redes de Distribuição, Boletim Técnico da escola Politécnica da Universidade de São Pulo, BT/PEA/0424, 2004

- [Booch, 01] Booch, G., Rumbaugh, J., Jacobson, I., The Unified Modeling Language User Guide, Addison-Wesley, 2001

- [Carrião, 04] Carrião, D. P. S., Santin, A. O., Maziero, C. A., Integrando o Modelo de Segurança SPKI/SDSI ao Ambiente de Gerência WBEM, IV Workshop em Segurança de Sistemas Computacionais – SBRC, Gramado, Brasil, 2004

- [Chicco, 01] Chicco, G. et al, Load Pattern Clustering for Short-Term Load Forecasting of Anomalous Days, IEEE Power Tech Conference, Porto, Portugal, 2001

- [Chow, 96] Chow, T.W.S., Leung, C.T., Neural Network Basead Short-Term Load Forecasting Using Weather Compensation, IEEE Transactions on Power Sytems, Vol. 11, n° 4, November 1996

- [Dabbaghchi, 97] Dabbaghchi, I. et all., AI Application Areas In Power Systems, IEEE Expert, Jan-Feb 1997

- [deVos, 01] deVos, A.; Rowbotham, C.T.; Knowledge Representation for Power System Modelling, Power Industry Computer Applications, 2001. PICA 2001. Innovative Computing for Power - Electric Energy Meets the Market. 22nd IEEE Power Engineering Society International Conference on , 20-24 May 2001 Pages:50 – 56

- [DMTF, 99] DMTF, Common Information Model (CIM) Specification, version 2.2, document version 1.2, June 14, 1999
- [DMTF, 03a] DMTF, Understanding the Application management Model CIM, version 2.7, document version 1.2, June 17, 2003
- [DMTF, 03c] DMTF, Common Information Model (CIM) Core Model, version 2.4, August 30, 2003
- [DMTF, 04a] DMTF, Specification for the Representation of CIM in XML, Version 2.2, December 09, 2004
- [DMTF, 04b] DMTF, Specification for CIM Operations over HTTP, version 1.2, December 09, 2004
- [DMTF, 03b] DMTF, CIM Interop Model White Paper, CIM version 2.7, version 0.9, June 19, 2003
- [Dong, 02] Dong, X. et al, Information Model for Power Equipment Diagnosis and Maintenance, Power Engineering Society Winter Meeting, 2002. IEEE , Volume: 1 , 27-31 Jan. 2002 Pages:701 - 706 vol.1
- [Egyhazy, 04] Egyhazy, C., Mukherji, R., Interoperability Architecture Using RM-ODP, Communications of The ACM, february 2004, Vol.47 nº 2, pg 93-97
- [Ferreira, 01] Ferreira, C.L.P., Maestro:Um Middleware para Suporte a Aplicações Distribuídas Baseadas em Componentes de Software, Dissertação de Mestrado, Escola Politécnica da Universidade de São Paulo, 2001
- [Filho, 01] Filho, A. S., Especificação de um Sistema de Informação Integrado para o Planejamento Operacional em Empresas de Energia, Tese de

- Doutorado, Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia de Computação e Sistemas Digitais, 2001
- [Fregni, 97] Fregni, E., Arquitetura Híbrida na Solução do Problema de Escalação da Manufatura, Tese de Doutorado, Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia de Computação e Sistemas Digitais, 1997
- [Galasini, 04] Galasini, M., O Método ODP-UP pra a Definição de Arquiteturas de Sistemas Distribuídos, Dissertação de Mestrado, Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia de Computação e Sistemas Digitais, 2004
- [Haykin, 02] Haykin, S., Redes Neurais – Princípios e Prática, Bookman, 2ª Edição, 2002.
- [He, 02] He, Y. et al. A Hierarchical Evolutionary Algorithm for Constructing and Training Wavelet Networks, Neural Computing and Applications, 2002
- [Hobbs, 04] Hobbs, C., A Practical Approach to WBEM/CIM Management, Auerbach Publications, 2004
- [HP, 03] HP, WBEM Services Software Developer's Kit for HP-UX Provider and Client Developer's Guide, Hewlett-Packard, 2003
- [Huang, 01] Hwang, K., Kim, G., A Short-Term Load Forecasting Expert System, Science and Technology, 2001. Korus'01. Proceedings. The Fifth Russian-Korean International Symposium On, Vol 1, 2001, 112-116

- [Huang, 03] Huang, G.M.; Nair, N.-K.C.; Static and Dynamic CIM-XML Documents for Proprietary EMS, Power Engineering Society General Meeting, 2003, IEEE , Volume: 2 , 13-17 July 2003 Pages: 1086 Vol. 2
- [ISO, 98a] ISO Recommendation X.901/ISO/IEC 10746-1: Information Technology Open Distributed Processing, Reference Model: Overview, 1998.
- [ISO, 96a] ISO Recommendation X.901/ISO/IEC 10746-2: Information Technology Open Distributed Processing, Reference Model: Foundations, 1996.
- [ISO, 96b] ISO Recommendation X.901/ISO/IEC 10746-3: Information Technology Open Distributed Processing, Reference Model: Architecture, 1996.
- [ISO, 98] ISO Recommendation X.901/ISO/IEC 10746-4: Information Technology Open Distributed Processing, Reference Model: Architectural Semantics, 1998.
- [IEC, 04] IEC Technical Specification IEC TS 61970-2: Energy Management System Application Program Interface (EMS-API) – part 2: Glossary, 2004
- [IEC, 03] IEC International Standard IEC 61970-301: Energy Management System Application Program Interface (EMS-API) – part 301: Common Information Model (CIM) Base, 2003.

- [Jones, 04] Jones, D., Managing Windows With VBScript and WMI, Addison Wesley, 2004

- [Kim, 00] Kim, K. et al, "Short Term Load Forecasting For Special Days in Anomalous Load Conditions Using Neural Networks and Fuzzy Inference Method", IEEE Transactions on Power Systems, Vol 15, n° 2, May 2000, 559-565

- [Kohonen, 90] Kohonen, T., Self-organized Map, Proc. IEEE, 1990, 78, (9), pp. 1464-1480

- [Kumar, 05] Kumar, A., Karnik N., Ravindranath, C. K., Moving from Data Modeling to Process Modeling in CIM, Integrated Network Management, 2005 9th IFIP/IEEE International Symposium on, 15-19 May, 2005 pges 763-686

- [Lee, 04] Lee, S. et al, Design of a WBEM-based Management System for Ubiquitous Computing Servers, POSTECH/ETRI, 2004

- [Mara, 99] Marra W., Lage W. F., Miranda C. A. S., Previsão de Carga de Curto Prazo, XV SNPTEE – Seminário Nacional de Produção e Transmissão de Energia Elétrica, GOP/19, Foz do Iguaçu, Paraná, Brasil, 17 a 22 de Outubro de 1999

- [Marin, 02] Marin, F.J., et all, Global Model For Short-Term Load Forecasting Using Artificial Neural Networks, IEE Proc. – Gener. Transm. Distrib., Vol 149, n° 2, March 2002

- [McMorran, 04] McMorran, A.W. et al, Translating CIM XML Power System Data to a Proprietary Format for System Simulation, Power Systems, IEEE Transactions on , Volume: 19 , Issue: 1 , Feb. 2004 Pages:229 – 235

- [Miyamoto, 98] Miyamoto, S., Katoh, S., Crisp and Fuzzy Methods of Optimal Clustering on Networks of Objects, IEEE Second International Conference on Knowledge-based Intelligent Electronic Systems, 21-23 April, Adelaide, Australia, 1998

- [MS, 05] Microsoft, Windows Management Instrumentation: Administering Windows and Applications Across Your Enterprise, <http://msdn.microsoft.com>, ultimo acesso em 12/03/2005

- [Nordström, 05] Nordström, L., Cegrell, T., Analyzing Utility Information System Architectures using the Common Information Model, CIGRE/IEEE PES, 2005 International Symposium, 5-7 October 2005, pages 274-281

- [Oliveira, 05] Oliveira, J.L, et al, Policy-Based Network Management in an Integrated Mobile Network, Proceedings of the Advanced Industrial Conference on Telecommunications/Service Assurance with Partial and Intermittent Resources Conference/Elearning on Telecommunications Workshop, 2005, 17-20 july 2005, pages 59-64

- [Osowski, 02] Osowski, S.; Siwek, K., Regularization of Neural Networks for Improved Load Forecasting in the Power System, IEE- Proc. – Generation Transm. Distrib., vol:149, n° 3, May 2002

- [Palacio, 01] Palacio, D.R., Simple Model For Load Forecast Weather Sensitive, Electricity Distribution, 2001. Part 1: Contributions. CIRED. 16th

International Conference and Exhibition on (IEEE Conf. Publ n° 482), Vol. 4, 2001

- [Piras, 96] Piras, A. et al, Heterogeneous Artificial Neural Network for Short Term Electrical Load Forecasting, IEEE Tansactions on Power Systems, vol 11, n° 1, Feb 1996

- [Putman, 01] Putman, J.R., Architecting with RM-ODP, Prentice Hall, 2001.

- [Rashid, 96] Al-Rashid, Y., Paarmann, L. D., Short-Term Electric Load Forecasting Using Neural Network Models, Circuits and Systems , 1996, IEEE 39th Midwest Symposium On, Vol 3, 1996, 1436-1439

- [Rosa, 04] Rosa, P. S., Integração ODP-CIM para Definição de Arquitetura de Sistemas Informação: Uma Aplicação na Qualidade de Distribuição de Energia Elétrica, Dissertação de Mestrado, Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia de Computação e Sistemas Digitais, 2004

- [Rosa, 03] Rosa, J. L. A., Classificação de Dados Através da Otimização do Método KNN-Fuzzy em Ambiente de Computação Paralela, Tese de Doutorado, Universidade Federal do Rio de Janeiro, 2003

- [Sanz, 01] Sanz, R., Corba for Control Systems, Annual Reviews in Control, Vol. 25, 2001, pp 169-181

- [Senjyu, 00] Senjyu, T., et al, Next Day load Curve Forecasting Using Self Organizing Map, Power System – Technology, 2000. Proceedings. POWERCON 2000. International Conference On, Vol. 2, 2000, 1113-1118

- [Senjyu, 02] Senjyu, T. et al, One-Hour-Ahead Load Forecasting Using Neural Network, IEEE Transactions on Power Systems, Vol 17, nº 1, Feb 2002

- [Simunic, 04] Simunic, J.; Zubrinic-Kostovic, K.; Dobras, B.; Process Information Model for Transmission System Operator Following IEC Model, Electrotechnical Conference, 2004. MELECON 2004. Proceedings of the 12th IEEE Mediterranean, Volume: 3, 12-15 May 2004 Pages:851 - 854 Vol.3

- [Singhal, 05] Singhal, S. et al, Quartermaster- A Resource Utility System, Integrated Network Management, 2005 9th IFIP/IEEE International Symposium on, 15-19 May, 2005 pges 265-278

- [Swarup, 02] Swarup, K. S., Satish, B., Integrated ANN – Approach to Forecast Load, IEEE Computer Applications in Power, April 2002, 46-51

- [Tamimi, 00] Tamimi, M., Egbert R., Short Term Electric Load Forecasting via Fuzzy Neural Collaboration, Electric Power Systems Resarch 56 (2000) 243-248

- [Teixeira, 00] Teixeira, S. R., Moraes, L. F. M., Teixeira Jr, J. H., Gerência baseada na Web, Relatório Técnico, Laboratório de Redes de Alta Velocidade – RAVEL, Universidade Federal do Rio de Janeiro

- [Tosic, 99] Tosic, V.; Dordevic-Kajan, S., The Common Information Model (CIM) standard - an analysis of features and open issues, Telecommunications in Modern Satellite, Cable and Broadcasting Services, 1999. 4th International Conference on, Volume: 2 , 13-15 Oct. 1999 Pages:677 - 680 vol.2

- [Trudnowski, 01] Trudnowski, D. J. et al, Real Time Very Short-Term Load Prediction For Power-System Automatic Generation Control, IEEE Transactions On Control Systems Technology, Vol 9, nº 2, March 2001
- [Tunstall, 03] Tunstall, C., Cole, G.; A Guide to Windows Management Instrumentation, Addison-Wesley, 2003
- [Varoto, 02] Varoto, A. C., Visões em Arquitetura de Software, Dissertação de Mestrado, Instituto de Estatística da Universidade de São Paulo, 2002
- [Velasquez, 01] Velasquez, R. M. G. et al, Técnicas de Classificação para Caracterização da Curva de Carga de Empresas de Distribuição de Energia – Um Estudo Comparativo, Proceedings of the V Brazilian Conference on Neural Networks, pp 133-138, April 2-5, Rio de Janeiro, RJ, 2001
- [Westerinen, 00] Westerinen A. R., Williams R. C., Common Information Model, John Wiley & Sons, 2000
- [Widergren, 99] Widergren, S.; deVos, A.; Jun Zhu; XML for Data Exchange, Power Engineering Society Summer Meeting, 1999. IEEE , Volume: 2 , 18-22 July 1999 Pages:840 - 842 vol.2
- [Winkler, 01] Winkler, G. et al, Intelligente Energy Management of Electrical Power Systems With Distributed Feeding on The Basis of Forecasts of Demand and Generation, CIRED2001, Conference Publication 482, June 2001, 18-21
- [Zadeth, 65] Zadeth, L.A., Fuzzy Sets, Information and Control, 8:338-353, 1965

- [Zang, 98] Zang, G., Patuwo, B. E., Hu, M. Y., Forecasting With Artificial Neural Networks: The State Of The Art, International Journal Of Forecasting, 14 (1998) 35-62