



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA**

**MÁRIO BARRETO DE MOURA NETO**

**APLICAÇÃO DA NORMA IEC 61970 PARA PADRONIZAÇÃO DE DADOS E  
INTEROPERABILIDADE DE REDES ELÉTRICAS INTELIGENTES**

**FORTALEZA**

**2014**

**MÁRIO BARRETO DE MOURA NETO**

**APLICAÇÃO DA NORMA IEC 61970 PARA PADRONIZAÇÃO DE DADOS E  
INTEROPERABILIDADE DE REDES ELÉTRICAS INTELIGENTES**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: Sistema de Energia Elétrica.

Orientadora: Prof<sup>ª</sup>. Ruth Pastôra Saraiva Leão, PhD.

Coorientador: Prof. Giovanni Cordeiro Barroso, Dr.

**FORTALEZA**

**2014**

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Biblioteca de Pós-Graduação em Engenharia - BPGE

- 
- M888a Moura Neto, Mário Barreto de.  
Aplicação da norma IEC 61970 para padronização de dados e interoperabilidade de redes elétricas inteligentes / Mário Barreto de Moura Neto. – 2014.  
102 f. : il., enc. ; 30 cm.
- Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Tecnologia, Departamento de Engenharia Elétrica, Programa de Pós-Graduação em Engenharia Elétrica, Fortaleza, 2014.  
Área de Concentração: Sistema de Energia Elétrica.  
Orientação: Profª. Dra. Ruth Pastôra Saraiva Leão.  
Coorientação: Prof. Dr. Giovanni Cordeiro Barroso.
1. Engenharia elétrica. 2. Dados - Padronização. 3. Sistemas de potência – Automação. I. Título.

## **AGRADECIMENTOS**

A realização desse trabalho contou com o apoio indispensável de alguns amigos a quem venho expressar minha gratidão.

À minha orientadora, professora Ruth Pastora Saraiva Leão, pela oportunidade de ser seu orientando, por cada ensinamento, pelo entusiasmo e excelência que plantaram em mim o desejo de me tornar professor.

Ao meu coorientador, professor Giovanni Cordeiro Barroso, pela valorosa contribuição na compreensão do tema desse trabalho, e pela constante disponibilidade em me ajudar a resolver as persistentes dificuldades.

Ao grupo de pesquisa em Redes Elétricas Inteligentes do Departamento de Engenharia Elétrica da UFC, professores e alunos, pelo estímulo e pelo compartilhar de conhecimentos.

Ao professor José Carlos Teles Campos, meu primeiro orientador durante a graduação, pelos ensinamentos que me serviram de fundamentos na vida acadêmica.

Aos meus pai, Dulce e Mário, pelo amor e apoio incondicionais, meus primeiros e principais entusiastas.

À minha noiva, Babi, pela alegria que me contagia e que tornará o resto dos meus dias mais felizes.

A Deus, sempre perto e presente, pela saúde, pelo prazer dos estudos, pelas dificuldades, pelos êxitos, e por colocar cada pessoa supramencionada em meu caminho com um propósito eterno.

“Seria possível, eu me pergunto, estudar um pássaro tão intimamente, observar e catalogar suas peculiaridades nos mais diminutos detalhes, a ponto de ele se tornar invisível? Seria possível, enquanto calculando meticulosamente a envergadura de suas asas ou a extensão de seu tarso, de alguma forma perdermos de vista sua poesia? ... Eu acredito que sim. Acredito que, ao nos aproximarmos de nossos objetos de estudo com a sensibilidade de estatísticos e disseccionistas, nós nos distanciamos cada vez mais do maravilhoso e fascinante planeta da imaginação cuja gravidade nos atraiu, antes de tudo, aos nossos estudos.” (Alan Moore)

## RESUMO

No processo atual de modernização pelo qual passam os sistemas de energia elétrica, o conceito de Redes Elétricas Inteligentes e seus fundamentos têm servido de diretrizes. Na busca pela interoperabilidade, a comunicação entre sistemas heterogêneos tem sido objeto de constantes e crescentes avanços. Este trabalho tem como objetivo o estudo e a aplicação do modelo de dados da série de normas IEC 61970, denominado Common Information Model (CIM). Com esse objetivo, os aspectos gerais da norma são apresentados, auxiliados pelos conceitos de UML (Unified Modeling Language) e XML (eXtensible Markup Language), essenciais para a compreensão integral do modelo. Determinadas características do modelo CIM, como sua extensibilidade e generalidade, são enfatizadas, as quais o credenciam como modelo com excelentes características para o estabelecimento da interoperabilidade. Com o intuito de exemplificar a utilização do modelo, realizou-se um estudo de caso em que se modelou uma rede elétrica de distribuição em média tensão de maneira a torná-la própria para integração com um sistema multiagente em um formato padronizado e, conseqüentemente, adequado à interoperabilidade. O processo completo de modelagem da rede elétrica utilizando o CIM foi demonstrado. Por fim, uma interface foi desenvolvida como mecanismo de manipulação dos dados nos documentos XML que possam fazer parte do fluxo de informações. A utilização do PHP, juntamente com um banco de dados MySQL, é justificada em decorrência de sua adequação de uso em ambientes diversos. Os conjuntos formados pela interface, simulador da rede elétrica e sistema multiagente para recomposição automática, constituíram um sistema cujas informações foram plenamente integradas.

**Palavras-chave:** Modelo de Dados. Interoperabilidade. IEC 61970. Common Information Model. Redes Elétricas Inteligentes.

## ABSTRACT

In the context of the current modernization process through which the electrical power systems go through, the concept of Smart Grids and their foundations serve as guidelines. In the search for interoperability, communication between heterogeneous systems has been the subject of constant and increasing developments. Under this scenario, the work presented in this dissertation focuses primarily on the study and application of the data model contained in the IEC 61970 series of standards, best known as the Common Information Model (CIM). With this purpose, the general aspects of the standard are exposed and assisted by the concepts of UML and XML, which are essential for a complete understanding of the model. Certain features of the CIM, as its extensibility and generality are emphasized, which qualify it as ideal data model for the establishment of interoperability. In order to exemplify the use of the model, a case study was performed which modeled an electrical distribution network in medium voltage so as to make it suitable for integration with a multi-agent system in a standardized format and, consequently, adequate to interoperability. The complete process of modeling an electrical network using the CIM is shown. Finally, the development of an interface is proposed as a mechanism that enables human intervention in the data flow between the integrated systems. The use of PHP with a MySQL database, are justified because of their suitability in various usage environments.

**Keywords:** Data Model. Interoperability. IEC 61970. Common Information Model. Smart Grids.

## LISTA DE FIGURAS

Figura 1.1 – Estrutura conceitual das REI .....	19
Figura 1.2 – Domínios de aplicação e relevância de normas IEC para REI.....	22
Figura 2.1 – Normas IEC de comunicação e seus domínios de aplicação .....	26
Figura 2.2 – UML: exemplo de classe .....	30
Figura 2.3 – UML: exemplo de associação .....	31
Figura 2.4 – UML: exemplo de generalização (ou herança) .....	32
Figura 2.5 – UML: exemplo de agregação .....	33
Figura 2.6 – UML: exemplo de composição .....	33
Figura 2.7 – UML: generalizações do elemento disjuntor .....	35
Figura 2.8 – Exemplo de conectividade .....	39
Figura 2.9 – Topologia com associação direta entre equipamentos .....	40
Figura 2.10 – <i>ConnectivityNode</i> conectando elementos .....	40
Figura 2.11 – UML: Diagrama de conectividade .....	41
Figura 2.12 – Exemplo de terminais .....	42
Figura 2.13 – Exemplo de mapeamento .....	44
Figura 2.14 – Diagrama de mapeamento .....	45
Figura 3.1 – Exemplo de elemento em XML .....	47
Figura 3.2 – Exemplo de elemento com atributo em XML .....	48
Figura 3.3 – Documento XML com dados da Tabela 3.1 .....	49
Figura 3.4 – Exemplo de XSD .....	50
Figura 3.5 – Documento RDF/XML com dados da Tabela 3.2 .....	52
Figura 3.6 – Exemplo de utilização do CIMTool .....	55
Figura 3.7 – CIMTool com a adição de três classes .....	56
Figura 3.8 – Adição de atributo no CIMTool .....	56
Figura 3.9 – Lista de classes e atributos do sistema modelado no CIMTool .....	57
Figura 3.10 – Geração do XSD no CIMTool .....	58
Figura 3.11 – Estrutura do XSD no CIMTool .....	58
Figura 3.12 – XML com informações do exemplo .....	59
Figura 3.13 – Tela de validação do exemplo de documento XML .....	60
Figura 4.1 – CIM e barramento de dados .....	61
Figura 4.2 – SFRA com topologia de uma rede de distribuição em média tensão..	64
Figura 4.3 – Mapeamento dos elementos da subestação de Messejana .....	66

Figura 4.4 – Parte do arquivo XML das subestações .....	66
Figura 4.5 – Validação do documento XML com dados das subestações .....	68
Figura 4.6 – Mapeamento das chaves da rede elétrica .....	69
Figura 4.7 – XML com dados sobre a chave RS1 .....	70
Figura 4.8 – Validação do documento XML com dados do centro .....	70
Figura 4.9 – Mapeamento da topologia da rede teste .....	71
Figura 4.10 – XML com parte da topologia da rede teste .....	72
Figura 4.11 – Validação do documento XML com dados da topologia do centro ..	72
Figura 4.12 – Modelagem dos trechos .....	74
Figura 4.13 – XML com parte das informações dos trechos .....	74
Figura 4.14 – Validação do documento XML com dados dos trechos .....	75
Figura 4.15 – Nós topológicos com mudança no estado de chave .....	76
Figura 4.16 – Parte do documento XML dos trechos após mudança .....	76
Figura 4.17 – Estrutura do XSD dos trechos .....	77
Figura 4.18 – Topologia após retirada de chave .....	78
Figura 4.19 – XML com topologia após retirada da chave RP1 .....	79
Figura 4.20 – XML com nós topológicos após retirada da chave RP1 .....	80
Figura 4.21 – Lógica de funcionamento da SMRA, SFRA e interface web .....	83
Figura 4.22 – Interface de escrita e leitura de dados .....	84

## LISTA DE TABELAS

Tabela 3.1 – Dados de estudantes para exemplo de uso do XML .....	48
Tabela 3.2 – Dados de exemplo de uma família .....	52
Tabela 4.1 – Parâmetros das subestações .....	65
Tabela 4.2 – Parâmetros das chaves .....	68
Tabela 4.3 – Parâmetros dos trechos .....	73
Tabela 4.4 – Tabelas e parâmetros da Rede Aquiraz .....	84

## LISTA DE ABREVIATURAS E SIGLAS

AMI	Advanced Metering Infrastructure
CCAPI	Control Center Application Programming Interface
CIM	Common Information Model
CIMug	CIM User Group
DER	Decentralized Energy Resources
DMS	Distribution Management System
DTD	Document Type Definition
EMS	Energy Management System
EPRI	Electric Power Research Institute
FACTS	Flexible Alternating Current Transmission System
HVDC	High Voltage Direct Current
IEC	International Electrotechnical Commission
IED	Intelligent Electronic Device
JADE	Java Agent Development Framework
MAMP	Macintosh, Apache, MySQL and PHP
NIST	National Institute of Standards and Technology
RDF	Resource Description Framework
REI	Redes Eléctricas Inteligentes
SCADA	Supervisory Control and Data Acquisition
SFRA	Simulador de Falta de Recomposição Automática
SGML	Standard Generalized Markup Language
SMRA	Sistema Multiagente de Recomposição Automática
UML	Unified Modeling Language
W3C	World Wide Web Consortium
WAMP	Windows, Apache, MySQL and PHP
XML	eXtensible Markup Language
XSD	XML Schema Definition

## SUMÁRIO

<b>1. INTRODUÇÃO</b>	<b>15</b>
<b>1.1. Fundamentos das Redes Elétricas Inteligentes</b>	<b>16</b>
1.1.1. <i>Geração distribuída</i>	16
1.1.2. <i>Medição inteligente</i>	17
1.1.3. <i>Automação da distribuição</i>	18
1.1.4. <i>Comunicação: o elemento integrador</i>	19
1.1.5. <i>Padronização</i>	20
<b>1.2. Justificativa</b>	<b>22</b>
<b>1.3. Objetivos</b>	<b>23</b>
<b>1.4. Organização do trabalho</b>	<b>23</b>
<b>2. MODELO CIM: THE COMMON INFORMATION MODEL</b>	<b>25</b>
<b>2.1. Histórico de criação</b>	<b>25</b>
<b>2.2. Domínio de aplicação do CIM</b>	<b>26</b>
<b>2.3. IEC 61970-301: A base do modelo</b>	<b>28</b>
2.3.1. <i>Fundamentos da linguagem UML</i>	29
2.3.1.1. <i>Classes</i>	30
2.3.1.2. <i>Associação</i>	30
2.3.1.3. <i>Generalização (ou Herança)</i>	31
2.3.1.4. <i>Agregação</i>	32
2.3.1.5. <i>Composição</i>	33
2.3.2. <i>A norma IEC 61970-301</i>	34
2.3.2.1. <i>Domain</i>	35
2.3.2.2. <i>Core</i>	36
2.3.2.3. <i>OperationalLimits</i>	36
2.3.2.4. <i>Topology</i>	36
2.3.2.5. <i>Wires</i>	37
2.3.2.6. <i>Generation.Production</i>	37
2.3.2.7. <i>Generation.Dynamics</i>	37
2.3.2.8. <i>LoadModel</i>	37
2.3.2.9. <i>Outage</i>	37
2.3.2.10. <i>Protection</i>	38
2.3.2.11. <i>Equivalentents</i>	38

2.3.2.12. Meas .....	38
2.3.2.13. SCADA .....	38
2.3.2.14. Control Area .....	38
2.3.3. Modelo de Conectividade .....	39
2.3.4. Extensibilidade .....	42
<b>2.4. IEC 61968-11 e IEC 62325-301: extensões oficiais .....</b>	<b>43</b>
<b>2.5. Utilização da norma .....</b>	<b>43</b>
<b>3. EXTENSIBLE MARKUP LANGUAGE (XML) .....</b>	<b>46</b>
<b>3.1. Características da linguagem XML .....</b>	<b>46</b>
<b>3.2. Sintaxe da XML .....</b>	<b>47</b>
3.2.1. Elementos .....	47
3.2.2. Atributos .....	48
<b>3.3. XML Schema Definition (XSD) .....</b>	<b>49</b>
<b>3.4. Modelo CIM e Linguagem XML .....</b>	<b>51</b>
3.4.1. Resource Description Framework (RDF) .....	52
3.4.2. CIMTool .....	54
<b>4. APLICAÇÃO DO MODELO CIM .....</b>	<b>61</b>
<b>4.1. Estudo de caso .....</b>	<b>61</b>
4.1.1. Sistema Multiagente de Recomposição Automática (SMRA) .....	62
4.1.2. Simulador de Falhas e Recomposição Automática (SFRA) .....	63
4.1.3. Definição e mapeamento dos dados .....	64
4.1.3.1. Modelagem das subestações .....	65
4.1.3.2. Modelagem das chaves de centro .....	68
4.1.3.3. Modelagem da topologia da rede em 13,8 kV .....	71
4.1.3.4. Modelagem dos trechos .....	73
4.1.4. Mudança na topologia .....	77
4.1.5. Interface web .....	81
4.1.5.1. Formulário PHP .....	81
4.1.5.2. Banco de dados MySQL .....	83
<b>5. CONCLUSÃO GERAL .....</b>	<b>86</b>
<b>5.1. Trabalhos futuros .....</b>	<b>87</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>88</b>
<b>ANEXO 1 – DIAGRAMAS UML DA NORMA 61970-301 MAIS UTILIZADOS.....</b>	<b>92</b>
a) Dependência entre os pacotes .....	92
b) Contêineres de equipamentos .....	93

c) Classes de conectividade .....	94
d) Generalizações de equipamentos .....	95
e) Classes de medidas .....	96
f) Tensão de base .....	97
g) Nó topológico .....	98
h) Cabos e condutores .....	99
<b>ANEXO 2 – EXEMPLOS DE TELAS DA INTERFACE PHP .....</b>	<b>100</b>
a) Tela de acesso aos dados das chaves e dos trechos do Centro .....	100
b) Tela de acesso aos dados da SE Aquiraz .....	100
c) Tela de preenchimento dos dados de duas chaves do Centro .....	101
d) Tela de acesso aos dados das subestações .....	102
e) Tela seleção de arquivo para leitura de dados das subestações .....	102

---

**CAPÍTULO 1.****INTRODUÇÃO**

---

A comunicação é um dos elementos que têm revolucionado o mundo. Ao possibilitar o contato com informações provenientes dos cantos mais remotos do globo, os meios de comunicação têm efetuado uma espécie de encurtamento das distâncias. Assim como os avanços no meios de transportes, a partir de meados do século XIX, os avanços atuais nos meios de comunicação têm modificado inúmeros aspectos na forma como a humanidade vive e interage.

Com o desenvolvimento contínuo e abrangente nos sistemas computacionais, os indivíduos passaram a ter um novo modelo de receber e armazenar informações. A facilidade e a conveniência de acesso, proporcionados pelos computadores pessoais e pela internet, tornaram obsoletos os instrumentos tradicionais de comunicação e armazenamento. Além disso, a velocidade de envio e recebimento de dados torna as pessoas cada vez mais dependentes da qualidade tanto dos mecanismos de fluxo como da informação propriamente dita.

Paralelamente, o crescimento global da população implica em um aumento ainda maior na demanda por energia elétrica. Hoje, as redes elétricas existentes estão sob pressão para entregar a crescente demanda por energia, bem como fornecer um suprimento estável e sustentável de energia elétrica. Estes desafios complexos conduzem à evolução das tecnologias de redes elétricas inteligentes (REI).

As redes inteligentes visam aumentar a capacidade da geração, transmissão e distribuição dos sistemas elétricos, bem como a confiabilidade, eficiência e sustentabilidade. As REI serão capazes de integrar uma ampla variedade de tecnologias de geração com diferentes características, como grandes usinas, geração distribuída, geração intermitente, bem como integrar recursos de armazenamento de energia, e adaptar-se a novos tipos de demanda do consumidor, como carga e descarga de veículos elétricos, participação ativa do consumidor no gerenciamento do consumo e produção *on-site* de eletricidade.

As REI produzirão uma enorme quantidade de dados. Para gerenciar, armazenar e analisar esses dados de forma eficaz, o sistema de potência, a arquitetura de comunicação, e as tecnologias de informação de alto nível precisam

ser coordenados como um sistema integrado de sistemas (OPTICAL ZEITGEST LABORATORY).

Interoperabilidade, sincronismo, confiabilidade, segurança e modularidade estão entre os atributos necessários para integração eficaz dos domínios: sistemas elétricos, arquitetura de comunicação e tecnologia da informação.

Da perspectiva da tecnologia da informação, os requisitos de comunicação de dados e a integração entre sistemas e aplicativos são questões relevantes dentro do conceito de REI. Um modelo de dados bem definido é a chave para facilitar ajustes em programas. O modelo CIM (*Common Information Model*) é proposto para permitir a troca de informações entre aplicativos para os sistemas de distribuição de energia elétrica. A linguagem XML (*eXtensible Markup Language*), uma linguagem de marcação genérica usada amplamente na ciência da computação, fornece grande interoperabilidade (FAWCETT *et al.*, 2012, p. 12). O modelo CIM e a linguagem XML serão abordados em seções posteriores.

Considerando-se os diversos dispositivos elétricos presentes em uma rede elétrica, responsáveis por diferentes tarefas e construídos com diferentes tecnologias, percebe-se que a construção de um sistema interoperável é um grande e importante desafio. O desenvolvimento de interfaces que possibilitem a comunicação entre os componentes de uma rede é, assim, fundamental na construção das REI. Os diversos aspectos envolvidos na construção dessas interfaces, como o formato com que os dados serão transmitidos, são de grande relevância para o funcionamento das redes inteligentes.

## **1.1. Fundamentos das Redes Elétricas Inteligentes**

É possível estabelecer três grandes áreas de desenvolvimento no domínio de sistemas de potência, relacionadas às Redes Elétricas Inteligentes (REI): geração distribuída, medição inteligente e automação da distribuição. As Tecnologias de Comunicação e Informação servirão de suporte no processo de integração.

### *1.1.1. Geração distribuída*

Chama-se geração distribuída o processo de descentralização das fontes geradoras de energia elétrica. Diversamente do que se encontra nos sistemas

tradicionais, nos quais as fontes de produção são centralizadas e de grande porte, no modelo das REI visa-se flexibilizar a hierarquia produtor/consumidor (IEC, 2010, p. 11) de forma a diversificar as fontes e suas localizações.

Uma das principais consequências desse novo modelo é a aproximação entre a fonte de geração e sua respectiva unidade consumidora. Essa proximidade não se dá apenas no aspecto físico, mas também no relacional. A maior interação entre o agente consumidor e seu sistema de geração de energia, além de representar um incremento na complexidade estrutural do sistema, permite que haja um maior controle sobre as diversas variáveis que determinam a qualidade da energia gerada (IEC, 2010, p. 11).

O incentivo à micro e mini geração de energia elétrica é uma outra consequência da geração distribuída. Pela maior adequação aos centros residenciais/comerciais, as fontes fotovoltaicas e eólicas de pequeno e médio porte mostram-se ideais para esse tipo de cenário, além de serem fontes limpas e renováveis. A sustentabilidade, assim, é uma das principais características desse modelo de geração (IEC, 2010, p. 12).

O desenvolvimento e a implantação da geração distribuída, entretanto, exigem uma série de inovações e desenvolvimentos (EPRI, 2009, p. 10). O fluxo bidirecional de energia, por exemplo, faz-se imprescindível para a completa viabilização desse novo conceito. A capacidade de uma unidade consumidora que possui um sistema de micro-geração de poder não apenas consumir energia da rede, mas também disponibilizar seu excedente pressupõe a existência de um arcabouço tecnológico e regulamentar que oriente seu funcionamento. A resolução n° 482 da ANEEL (Agência Nacional de Energia Elétrica), por exemplo, publicada em abril de 2012, é um marco no sistema regulatório da geração distribuída no Brasil. Pelo aspecto instrumental, a habilidade de supervisionar e controlar os parâmetros de microrredes torna necessário o desenvolvimento de dispositivos de medição inteligentes, que é o próximo tópico a ser explorado.

### *1.1.2. Medição inteligente*

O conceito de medição inteligente refere-se à capacidade, por parte das concessionárias e dos consumidores de energia, de monitorar e controlar as diversas variáveis em um sistema elétrico (IEC, 2010, p. 16).

Supervisionar os dados provenientes de um sistema de energia é uma atividade que possui vantagens de diferentes naturezas. Pode-se mencionar, por exemplo, a eficiência econômica como resultado do controle sobre os horários de suprimento de determinado tipo de fonte geradora (EPRI, 2009, p. 7). A energia, igualmente, pode ter sua qualidade aprimorada mediante o conhecimento do consumo de reativo e de ocasionais quedas de tensão.

O acesso em tempo real aos valores dos parâmetros e aos estados dos elementos que compõem uma rede representa um grande avanço na maneira como a energia elétrica é utilizada. Ao consumidor, antes numa posição exclusivamente passiva, é permitido, além de produzir sua própria energia mediante a micro-geração, acompanhar e interferir no comportamento e no desempenho de seu sistema (IEC, 2010, p. 13). Percebe-se, então, que o modelo de Redes Elétricas Inteligentes sugere não apenas a descentralização e a bidirecionalidade do fluxo de energia, mas também a medição avançada para obtenção de informações de interesse relacionadas à energia elétrica. Para possibilitar tal interação, o processo de automação da distribuição de energia é indispensável.

### *1.1.3. Automação da distribuição*

O desenvolvimento tecnológico inerente à modernização dos sistemas de energia reflete-se no crescente grau de automação dos sistemas elétricos de distribuição. Desde aspectos na concepção de Dispositivos Eletrônicos Inteligentes (do inglês *Intelligent Electronic Device*, IED), até as estratégias de planejamento energético, uma rede elétrica automatizada modifica toda a estrutura de operação de um sistema de energia (IEC, 2010, p. 15).

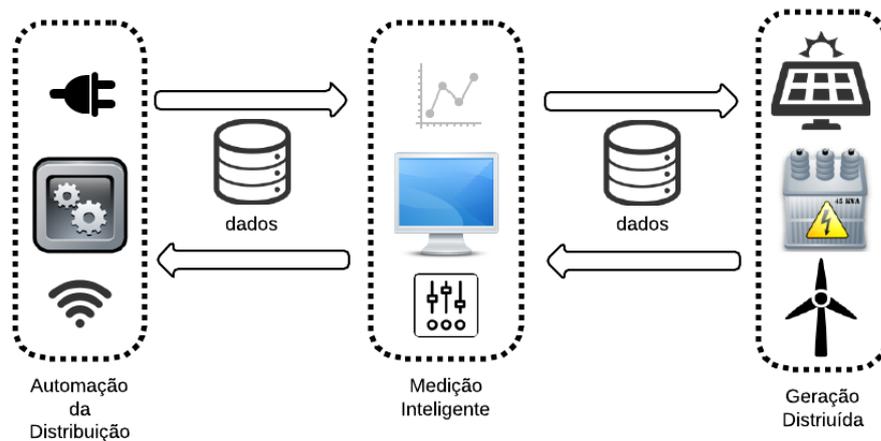
Como principal consequência desse processo, observa-se uma gradual horizontalização na hierarquia decisória de uma rede. A clássica organização centralizada, gerida pelos centros de controle das agências distribuidoras e das operadoras, dá lugar aos chamados sistemas distribuídos. Ao basear-se no desenvolvimento de *softwares* customizados para cada tipo de rede, a aplicação de sistemas multiagentes possibilita a divisão de tarefas nos diferentes níveis da topologia das malhas elétricas. Dessa maneira, há um consequente aumento na velocidade de ação de atuadores e uma maior eficiência dos sistemas de controle e proteção.

O acesso remoto aos pontos de atuação e medição por parte dos consumidores pode ser mencionado como uma outra consequente vantagem da automação da distribuição.

#### 1.1.4. Comunicação: o elemento integrador

A partir de uma perspectiva integral do conceito de Redes Elétricas Inteligentes, é possível perceber a importância que a comunicação representa para o seu completo funcionamento. A cooperação entre os diversos domínios e os vários participantes, cerne do modelo, exige que haja um substancial e contínuo desenvolvimento nas tecnologias relacionadas à troca de informação. Na Figura 1.1, é representada a estrutura conceitual das REI.

Figura 1.1. Estrutura conceitual das REI.



Entre inúmeros desafios técnicos que devem ser transpostos para a construção de Redes Elétricas Inteligentes, encontra-se o desenvolvimento de sistemas de comunicação e de manipulação de dados (EPRI, 2009, p. 10). O domínio conhecido como Tecnologia da Informação e Comunicação ocupa uma posição de destaque no progresso e na construção das REI. O aprimoramento dos meios que viabilizem o fluxo de dados, para torná-los mais fluídos e operacionais, mostra-se indispensável para o efetivo funcionamento de todos os aspectos das Redes Inteligentes. Simultaneamente, o avanço nos métodos de análise e processamento de informação deve ser aprimorado de maneira a incrementar a eficiência dos sistemas.

### 1.1.5. Padronização

Tradicionalmente, o desenvolvimento de *softwares* de suporte à operação das redes de energia não priorizava a necessidade de integração com outros sistemas. No entanto, com a gradual abertura dos mercados de eletricidade às empresas privadas, sobretudo a partir da década de 1990, diversas linguagens e estruturas proprietárias passaram a ser utilizadas nas arquiteturas de programação dos sistemas de potência. Nessas circunstâncias, em que operadoras diferentes utilizavam sistemas proprietários, observou-se a necessidade de desenvolver meios que possibilitassem a comunicação entre ambientes diversos (EPRI, 2011).

A definição de padrões mostrou-se um avanço de grande importância no processo de integração de sistemas computacionais. Sem o uso de padrões, o processo de integração tornar-se-ia economicamente muito dispendioso, visto que seria necessário criar interfaces específicas entre cada um dos diferentes aplicativos que precisam ser integrados, podendo causar impacto na velocidade de resposta dos aplicativos. Além disso, os trabalhos de organizações especializadas em orientar os esforços na criação e aprimoramentos dos padrões e normas são propulsores da inovação tecnológica. Como exemplos desse tipo de organização, pode-se destacar a *International Electrotechnical Commission* (IEC) e o *National Institute of Standards and Technology* (NIST).

Por ser a interoperabilidade um elemento indispensável para o funcionamento das REI, os padrões e normas constituem instrumentos essenciais no processo de integração de sistemas de energia (IEC, 2010, p. 11). O papel fundamental de padrões para as REI é garantir os investimentos em novas tecnologias, diminuindo os riscos de essas tecnologias tornarem-se prematuramente obsoletas; catalisar inovações, assegurando o uso de novas tecnologias em toda a rede; criar economia de escala para reduzir custos; evidenciar as melhores práticas; e abrir mercado para sistemas e dispositivos de REI. Os padrões, no entanto, não são estáticos, mas evoluem à medida que ganham maturidade, passando por revisões para adicionar novas funcionalidades e remediar alguma limitação revelada durante sua aplicação (NIST, 2012).

Nesse contexto, o objeto de estudo do presente trabalho é o uso do padrão apresentado na série de normas IEC 61968/61970. Tal padrão, conhecido como *Common Information Model* (CIM), consiste em um modelo cuja ideia-chave é definir

uma linguagem comum que facilite a troca de mensagens entre sistemas independentes (USLAR *et al.*, 2011, p. IX). Em outras palavras, o CIM é um dos instrumentos por meio do qual se efetua a comunicação e integração de sistemas de monitoramento de dados de redes elétricas.

O CIM é considerado um dos elementos basilares no desenvolvimento das REI, estando presente em documentos oficiais tais como *Japanese METI International Standardization Roadmap for Smart Grid* (2009), *Microsoft Smart Energy Reference Architecture* (2009), *The German Standardization Roadmap E-Energy / Smart Grid* (2010) e *The State Grid Corporation of China (SGCC) Framework and Roadmap for Strong and Smart Grid Standards* (2010). Assim, o conhecimento e aplicação desse modelo é de grande relevância para a construção de uma REI.

Além do CIM, a IEC tem desenvolvido, no decorrer dos últimos decênios, uma série de normas e padrões dedicados à implementação de diferentes aspectos das REI. Visto que o conceito de redes inteligentes envolve mudanças significativas em praticamente todas as dimensões dos sistemas de energia elétrica, é importante que haja um planejamento de qual a melhor maneira de efetuar as alterações necessárias para o desenvolvimento de uma REI. Para auxiliar nesse planejamento, a Figura 1.2 apresenta os domínios de aplicação de diversas normas da IEC voltadas ao desenvolvimento de redes inteligentes. Observa-se, pela análise da coluna mais à direita, que as séries de normas 61970 e 61968, ambas responsáveis pelo CIM, são consideradas como “núcleo” em sua relevância para a construção das REI.

Com base na importância no domínio e aplicação de um padrão internacional para troca de dados nos sistemas de potência, o grupo de pesquisa em Redes Elétricas Inteligentes do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Ceará constituiu, como ação prioritária e necessária para o avanço da pesquisa na área, o estudo e aplicação do padrão CIM em um protótipo de rede elétrica. Nesse contexto, o estudo do CIM apresenta-se como padrão a ser utilizado na integração de diferentes iniciativas em curso, baseadas no conceito de redes inteligentes.

Figura 1.2. Domínios de aplicação e relevância de normas IEC para REI.

	HVDC/FACTS	Blackout Prevention / EMS	DMS	Distribution Automation	Substation Automation	DER	AMI	DR	Smart Home	Electric Storage	Electromobility	Relevance for Smart Grid
SOA – IEC 62357		x	x									Core
CIM – IEC 61970-301		x	x	x	x	x	x	x		x		Core
ISO/IEC 14543									x			Low
ISO/IEC 27001									x			Low
IEC 60255			x	x	x							Low
IEC 60364						x			x			Medium
IEC 60495							x		x			Low
IEC 60633	x											Low
IEC 60834		x	x		x							Low
IEC 60870-5		x	x	x	x							High
IEC 60870-6		x	x									High
IEC 60904						x		x	x			Medium
IEC/TR 61000						x	x		x	x	x	Low
IEC/TS 61085												
IEC 61140									x		x	Medium
IEC/TR 61158 / 61784					x							Medium
IEC/TR 61334							x					High
IEC 61400		x	x			x						High
IEC 61508												
IEC 61850		x	x	x	x	x	x			x	x	Core
IEC 61850-7-410		x	x	x	x	x						High
IEC 61850-7-420			x	x	x	x						High
IEC 61851									x		x	High
IEC 61869				x	x							Medium
IEC 61954	x											Low
IEC 61968			x			x	x	x				Core
IEC 61970		x	x		x							Core
IEC 61982											x	Low
IEC 62051-54 / 58-59			x			x	x	x	x	x	x	High
IEC 62056			x			x	x	x	x	x	x	High

Fonte: (IEC, 2010, p. 106).

## 1.2. Justificativa

A análise preliminar de artigos científicos e documentos foi fundamental para definir as diretrizes que orientaram a presente dissertação. De uma forma geral, os trabalhos apresentados em Andren *et al.* (2012), Cauchon *et al.* (2011), Guo *et al.* (2011), Haifeng *et al.* (2013), Hargreaves *et al.* (2012), Ilich *et al.* (2008), Lendak *et al.* (2010), Li *et al.* (2013), Mercurio *et al.* (2009), Moseley *et al.* (2012), Rein *et al.* (2013), União Europeia (2007), Uslar *et al.* (2012), Wang *et al.* (2011), Wuergler *et al.* (2012) e Zhang *et al.* (2012) orbitam em torno de um dos dois extremos: aplicações do modelo CIM em casos específicos, nos quais o modelo é apresentado como um acessório que aprimora um projeto principal; ou estudos conceituais sobre o CIM, em que se evidenciam as potencialidades do modelo. Havia, assim, uma

espécie de lacuna entre o conhecimento e compreensão do modelo e a forma como o padrão deve ser utilizado em um caso concreto. O documento CIM Primer, elaborado pelo EPRI (*Electric Power Research Institute*) e utilizado como referência nesse trabalho, inclui alguns exemplos que, em decorrência da natureza abrangente do trabalho, acabam sendo breves e superficiais.

Como pode-se observar pela análise das referências bibliográficas utilizadas nesse trabalho, a utilização do CIM tem-se tornado recorrente em projetos envolvendo fluxo de dados entre aplicativos nos últimos 3 anos. No entanto, a compreensão da norma ainda exige um esforço considerável na busca por fontes que abordem a utilização prática e direta da norma. É facilmente percebida a carência que há na literatura especializada de documentos dedicados aos aspectos funcionais do modelo CIM.

Por ser considerado um dos padrões fundamentais para a construção das REI, julgou-se oportuno desenvolver um material que servisse de instrumento para iniciar o uso do modelo CIM. Dessa maneira, a presente dissertação figura como um manual introdutório da utilização do modelo e sua aplicação num caso-teste. Com uma abordagem pragmática, esse trabalho constitui uma ferramenta útil para a compreensão e uso do modelo de dados constante da norma IEC 61970.

### **1.3. Objetivos**

Tendo em conta as considerações expostas acima, essa dissertação tem como objetivos:

- a) Apresentar o modelo CIM, suas características gerais e os conceitos necessários envolvidos na sua compreensão;
- b) Descrever, utilizando um estudo de caso para exemplificar a aplicação do modelo, o processo completo de modelagem de um sistema elétrico com o modelo de dados CIM;
- c) Desenvolver uma ferramenta que torne a manipulação dos dados rápida e adequada a um ambiente de pesquisa e desenvolvimento.

### **1.4. Organização do trabalho**

Essa dissertação é composta por cinco capítulos e dois anexos.

No Capítulo 2 os conceitos gerais das normas IEC 61970, 61968 e 62325, que compõem o CIM, são apresentadas. Para a compreensão do modelo, a linguagem UML (*Unified Modeling Language*) é introduzida. A parte 61970-301 da norma é apresentada, com uma descrição breve dos pacotes em que são divididos os diversos diagramas UML que a compõem. O processo de mapeamento dos dados é exemplificado.

No Capítulo 3 é apresentada a linguagem XML (*eXtensible Markup Language*), que será utilizada para codificar as informações modeladas com o CIM. Após a descrição das suas características gerais, introduz-se o conceito de XSD (*XML Schema Definition*), utilizado para validação de documentos XML. A diferença entre RDF (*Resource Description Framework*) e XML básica é apresentada, e a escolha pela última para a modelagem do estudo de caso é justificada. O aplicativo CIMTool é utilizado para a geração do XSD.

No Capítulo 4, um estudo de caso de integração de dois sistemas heterogêneos é apresentado e utilizado para efetuar a modelagem mediante o CIM. Uma rede elétrica de distribuição em média tensão, com parâmetros reais, sobre a qual operam os sistemas a serem integrados, é apresentada, e as etapas necessárias para a modelagem são descritas. Uma interface *web* é desenvolvida para facilitar a intervenção do operador do sistema integrado nos dados trocados.

No Capítulo 5, chega-se a uma conclusão do trabalho, apresentando os resultados e sugestões de trabalhos futuros.

O Anexo 1 apresenta alguns dos diagramas UML retirados diretamente da parte da norma IEC 61970-301, com o objetivo de auxiliar na compreensão da lógica do modelo.

O Anexo 2 apresenta imagens de algumas das telas da interface desenvolvida em PHP para o gerenciamento e consulta dos arquivos XML.

### 2.1. Histórico de criação

Com o processo de desregulamentação do mercado de energia elétrica, sobretudo a partir do início dos anos 1990, o compartilhamento de dados entre empresas envolvidas no setor elétrico tornou-se uma tarefa bastante complexa. O grande número de companhias de geração, transmissão e distribuição de energia, juntamente com a criação e o desenvolvimento de protocolos proprietários na manipulação e gerenciamento de dados internos, dificultou o estabelecimento da troca de informação entre sistemas diferentes.

A dificuldade de comunicação, além disso, não se dá apenas entre empresas diferentes. No interior de uma mesma organização há, frequentemente, o emprego de múltiplos aplicativos de estrutura proprietária. Assim, as tradicionais soluções empregadas pelas antigas operadoras, que consistiam em construir seu próprio formato padrão ou adquirir os diversos formatos de um mesmo vendedor, tornaram-se inviáveis e ineficientes.

Uma solução alternativa para esse problema seria providenciar interfaces entre cada um dos formatos proprietários a fim de viabilizar a troca de informação. Esse recurso, entretanto, além de oneroso pode ser ineficaz e com pouca flexibilidade de adaptação. A crescente complexidade dos sistemas utilizados no setor elétrico, e os diversos *softwares* e aplicativos empregados, tornam essa via pouco atrativa.

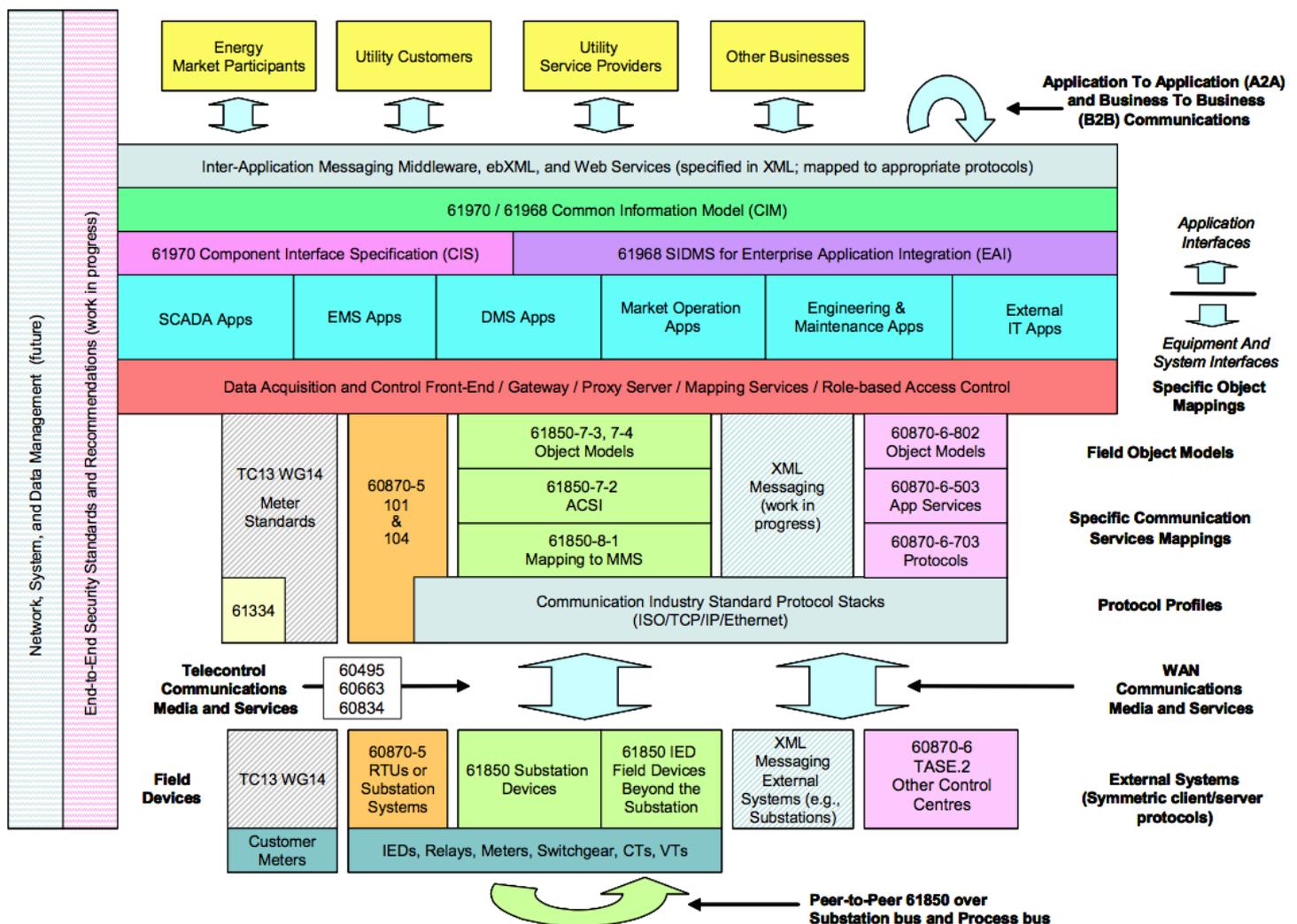
O *Common Information Model*, modelo que visa transpor essa dificuldade de comunicação imposta pela utilização de diferentes modelos proprietários, tem sua origem no projeto chamado *Control Center Application Programming Interface* (CCAPI) da organização estadunidense *Electric Power Research Institute* (EPRI), em meados dos anos 1990 (USLAR *et al.*, 2011, p. 23). O objetivo principal desse projeto era criar uma definição comum dos componentes elétricos presentes em sistemas de potência para ser utilizada na programação de interfaces entre aplicativos do Sistema de Gerenciamento de Energia (do inglês, *Energy Management System* (EMS)). O CIM visa, assim, definir a melhor maneira de se formatar os dados que precisam ser compartilhados entre sistemas.

Posteriormente, após a relevância do modelo ter sido percebida, o padrão CIM passou a ser mantido e desenvolvido pelo IEC – TC 57, Comitê Técnico 57 da IEC, uma organização europeia de alcance mundial.

### 2.2. Domínio de aplicação do CIM

Antes de iniciar-se o estudo dos conceitos técnicos envolvidos na compreensão do CIM, é fundamental analisar e definir o domínio sobre o qual o CIM é aplicado. Para auxiliar nesse entendimento, a Figura 2.1 apresenta algumas normas da IEC para comunicação e seus domínios de aplicação.

Figura 2.1. Normas IEC de comunicação e seus domínios de aplicação.



\*Notes: 1) Solid colors correlate different parts of protocols within the architecture.  
 2) Non-solid patterns represent areas that are future work, or work in progress, or related work provided by another IEC TC.

Observa-se, na parte mais superior do diagrama da Figura 2.1, que o modelo de dados apresentado na série de normas IEC 61970/61968 (CIM) é utilizado para o estabelecimento de interfaces do tipo Aplicação-Aplicação, ou seja, entre *software*. Salienta-se, assim, que o CIM não define a maneira como os aplicativos estão interligados. O tipo e o vocabulário das mensagens trocadas entre aplicativos são definidas, sim, pelo modelo objeto de nosso estudo, todavia o meio físico que será utilizado como transporte dessas mensagens deve ser o que for considerado o mais adequado para a integração. A indiferença, por parte do CIM, quanto à escolha do mecanismo de fluxo de dados, revigora seu potencial como o modelo de dados mais adequado na busca pela interoperabilidade.

O CIM não se confunde, dessa maneira, com as normas IEC 61850 e 60870-6 que são responsáveis, respectivamente, pelo protocolo de comunicação em subestações e entre centros de controle. Como pode ser observado na parte mais inferior da Figura 2.1, tais normas atuam entre equipamento e estabelecem os meios que viabilizam a transmissão física de dados.

A despeito dessa diferença fundamental, as normas IEC 61850 e 60870-6 possuem seus próprios modelos de dados, necessários para o funcionamento dos seus respectivos protocolos. Tais modelos de dados, entretanto, apresentam as classes úteis para a modelagem dos aspectos mais específicos e particulares dos equipamentos envolvidos na comunicação. O CIM é um modelo relacionado aos recursos de uma rede elétrica, que consegue modelar que um transformador possui dados de tensão/corrente, mas que não pode discernir, por exemplo, o dado proveniente de um relé da Schweitzer. Já a norma IEC 61850 apresenta um modelo de dados relacionado a um tipo específico de equipamento, que pode modelar a tensão medida por um relé da Schweitzer, por exemplo, mas não modela que tal medida pertence a um transformador presente na rede. O valor do CIM está em sua abrangência e extensibilidade, enquanto a IEC 61850 modela detalhes de uma subestação que o CIM não modela (AYERS, 2011). Pode-se afirmar, assim, que a diferença entre os dois modelos de dados está em que possuem perspectivas diferentes: enquanto a IEC 61850 possibilita a modelagem de dados no âmbito de controle de uma subestação, o CIM apresenta um modelo útil para a troca de dados entre aplicativos de alto-nível (Sučić *et al*, 2011).

A norma IEC 60870-6, que define o *Inter-Control Center Communications Protocol* (ICCP), conhecido também como *Telecontrol Application Service Element*

(TASE.2), apresenta uma norma análoga à IEC 61850 para o caso do estabelecimento da comunicação entre centros de controle. Além de ser um protocolo de comunicação, como o próprio nome especifica, a norma apresenta um modelo de dados específico para representação dos elementos necessários para a comunicação entre centros de controle.

A despeito de em DA CUNHA *et al* (2004) afirmar-se que o ICCP é mais adequado do que o CIM para a representação de dados em tempo real, no presente trabalho não se efetuou um estudo comparativo entre os modelos de dados. Entretanto, observa-se que o objetivo do CIM é possibilitar a troca de dados entre aplicativos independentemente da procedência da informação. Além disso, os trabalhos realizados no âmbito do Comitê Técnico 57 da IEC na busca da harmonização do modelo de dados de ambas as normas, IEC 61850 e IEC 60870-6, com o CIM indicam que o CIM estabelece, em seu domínio de aplicação, o modelo a ser utilizado.

### **2.3. IEC 61970-301: A base do modelo**

O CIM é um padrão aberto destinado a representar componentes de sistemas de potência e mantido como um diagrama em UML (*Unified Modeling Language*). O diagrama propriamente dito, que atualmente é disponibilizado no *site* do grupo dos usuários do CIM (*CIM User Group*, CIMug), é visualizado e manipulado por meio do aplicativo *Enterprise Architect*, cuja extensão dos arquivos é “.eap”.

Como será visto posteriormente, dentre a série de normas que integram o CIM, a IEC 61970-301 é a base do modelo. Nela, estão representados diversos componentes presentes num sistema elétrico como classes de um diagrama UML, com seus atributos e associações. Essa parte da norma deve ser entendida como o instrumento que possibilita a integração, mediante o estabelecimento do modelo comum, dos diversos domínios (mercado, controle e operação da geração, transmissão e distribuição, e prestação de serviço) envolvidos nos sistemas de potência (USLAR *et al.*, 2011, p. 80).

Para compreender corretamente a lógica de utilização do modelo presente nessa norma, é fundamental que se conheça as bases teóricas da UML. Entretanto, somente o conhecimento necessário para a compreensão e utilização do CIM será

aqui explorado, visto que uma abordagem mais abrangente sairia do escopo desse trabalho.

### 2.3.1. Fundamentos da linguagem UML

A UML é uma família de notações gráficas que ajuda na descrição e no projeto de sistemas de *software*, particularmente daqueles orientados a objetos. Assim como outras linguagens gráficas de modelagem, a motivação principal para a criação da UML, a partir da unificação de diversas outras, é o fato de que as linguagens de programação não estão em um nível de abstração suficientemente alto para facilitar as discussões sobre projeto (FOWLER, 2004, p. 25).

Em decorrência da alta complexidade dos sistemas reais que precisam ser modelados, o uso de uma ferramenta como a UML é extremamente útil no processo de discussão e representação de projetos. Ao examinar um diagrama UML, não é possível dizer *exatamente* como seria o código equivalente. Todavia, é possível ter uma *ideia aproximada* de como ficaria o código, o que é suficiente para ser útil.

Outra vantagem da UML é que, mesmo ao representar objetos reais, o grau de detalhamento pode ser modulado de acordo com a necessidade da modelagem que se faz. Essa característica é importante por possibilitar a análise do projeto modelado por pessoas que não tenham o conhecimento específico sobre a linguagem de programação utilizada.

Além disso, a utilização da UML para modelagem de sistemas não vincula o modelador ao uso de alguma linguagem de programação específica. O programador, assim, tem total liberdade para escolher a linguagem que melhor lhe convier. O CIM, por exemplo, e como será visto posteriormente nesse trabalho, utiliza principalmente (mas não exclusivamente) o XML como escolha de serialização<sup>1</sup> de seus modelos.

---

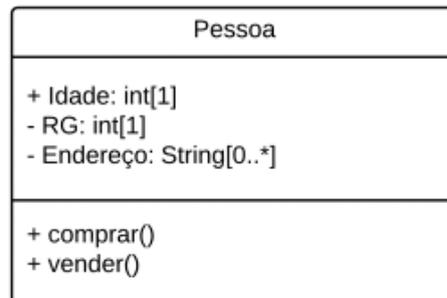
<sup>1</sup> Serialização - termo utilizado no ramo da Ciência da Computação para definir o processo de transformar um objeto abstrato num formato armazenável, transmissível e manipulável.

### 2.3.1.1. Classes

Os diagramas de classes são as formas mais simples de representação em UML. Eles descrevem os tipos de objetos presentes no sistema e os relacionamentos que há entre eles.

Uma classe é uma abstração de um tipo de objeto ou de um conjunto de objetos que compartilham características comuns. As classes são representadas por retângulos que, além do nome da classe, podem conter outras informações sobre a classe, tais como atributos e ações da classe. Um exemplo de classe é representado na Figura 2.2.

Figura 2.2. UML: exemplo de classe.



Nesse exemplo, a classe Pessoa possui três atributos (Idade, RG e Endereço) e duas operações (comprar e vender). Os símbolos “+” e “-” designam a visibilidade da propriedade a que se refere, sendo público e privado, respectivamente. O índice ao lado do tipo de atributo (int, String) refere-se à sua multiplicidade, que estabelece a quantidade que um determinado atributo deve ou pode (no caso da multiplicidade ser um intervalo) aparecer. O símbolo “\*” é usado para designar um valor ilimitado.

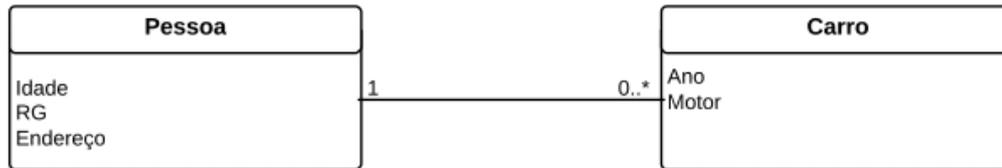
As operações, por sua vez, definem as ações e comportamentos que a classe pode apresentar. No entanto, como será visto posteriormente, o diagrama de classes do CIM não utiliza esse tipo de operação para caracterizar as classes presentes no modelo.

### 2.3.1.2. Associação

As associações são os tipos mais básicos de relacionamento entre classes representado pela UML. Representada por uma linha cheia ligando duas classes, as

associações estabelecem uma relação semântica entre as classes conectadas. Por exemplo, na Figura 2.3 representa-se o relacionamento entre a classe Pessoa e a classe Carro.

Figura 2.3. UML: exemplo de associação.



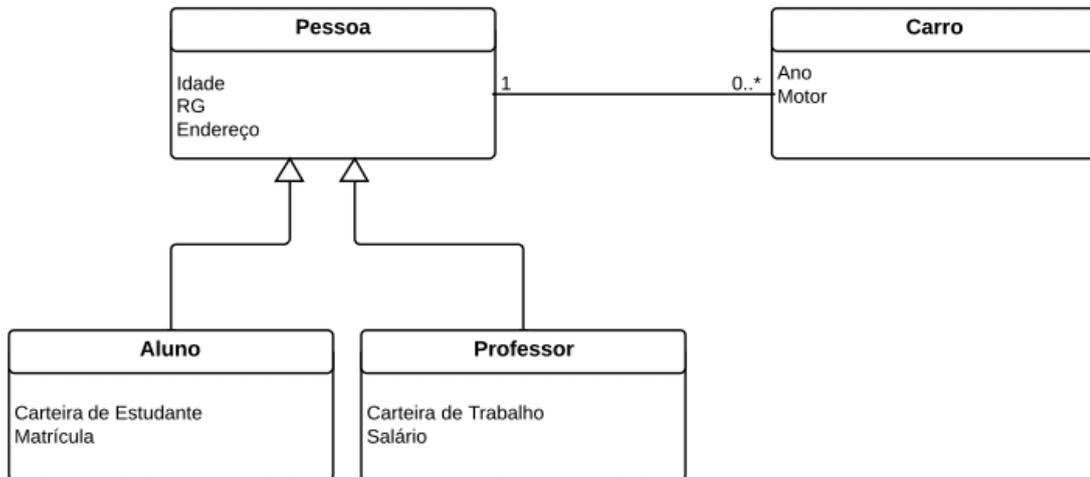
Assim como os atributos, as associações possuem multiplicidade em cada uma de suas extremidades para especificar a natureza quantitativa do relacionamento. É usual utilizar uma seta em uma das extremidades da linha de ligação quando a associação assim for melhor representada. Além disso, colocar o verbo que explique a associação acima de cada uma das extremidades também pode ser útil. Todas essas possibilidades são livres para serem utilizadas pelo responsável pela modelagem para tornar o diagrama o mais claro possível.

Quando a associação é bidirecional, é habitual (mas não obrigatório) omitir as setas das duas extremidades, assim como na figura acima. Presume-se que a associação seja compreensível dessa maneira.

### 2.3.1.3. Generalização (ou Herança)

A generalização é um tipo de relacionamento entre uma primeira classe mais genérica e uma segunda que é uma subclasse mais específica que a primeira. A subclasse herda todas as propriedades da classe-mãe, tanto os atributos, quanto as associações. A relação de herança é representada por uma linha cheia com uma seta vazada na extremidade da classe mais genérica. A Figura 2.4 apresenta um exemplo de generalização, na qual as classes Alunos e Professores são subclasses de Pessoa.

Figura 2.4. UML: exemplo de generalização (ou herança).



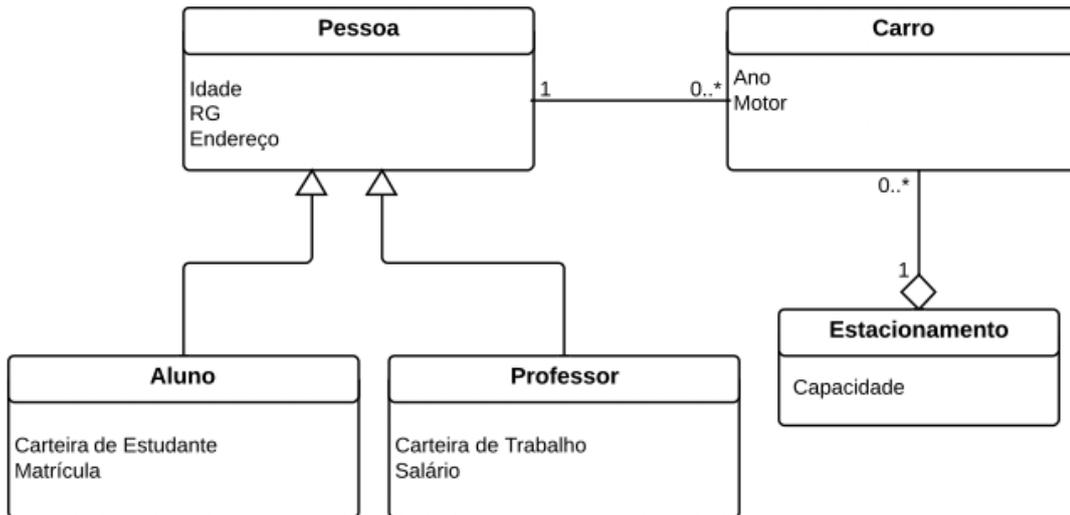
No diagrama da Figura 2.4, tanto a classe Aluno quanto a classe Professor são tipos da classe Pessoa, mas cada uma delas possui suas próprias propriedades (atributos e associações). Assim, as classes Aluno e Professor possuem os atributos Idade, RG, Endereço e a associação com a classe Carro.

A generalização é um tipo de relacionamento extremamente útil para simplificar e estruturar hierarquicamente a organização das classes. Além de evitar uma enorme quantidade de repetições, a herança cria uma relação de pai-filho entre as classes que auxilia no processo de organização da modelagem de sistemas.

#### 2.3.1.4. Agregação

Um tipo de relacionamento mais forte do que a associação é a agregação, que indica que uma classe está contida em outra. Essa relação estabelece um tipo de pertencimento. A agregação é representada por uma linha cheia com um diamante vazado na extremidade da classe que contém a outra. A Figura 2.5 representa uma relação de agregação entre a classe Carro e a classe Estacionamento.

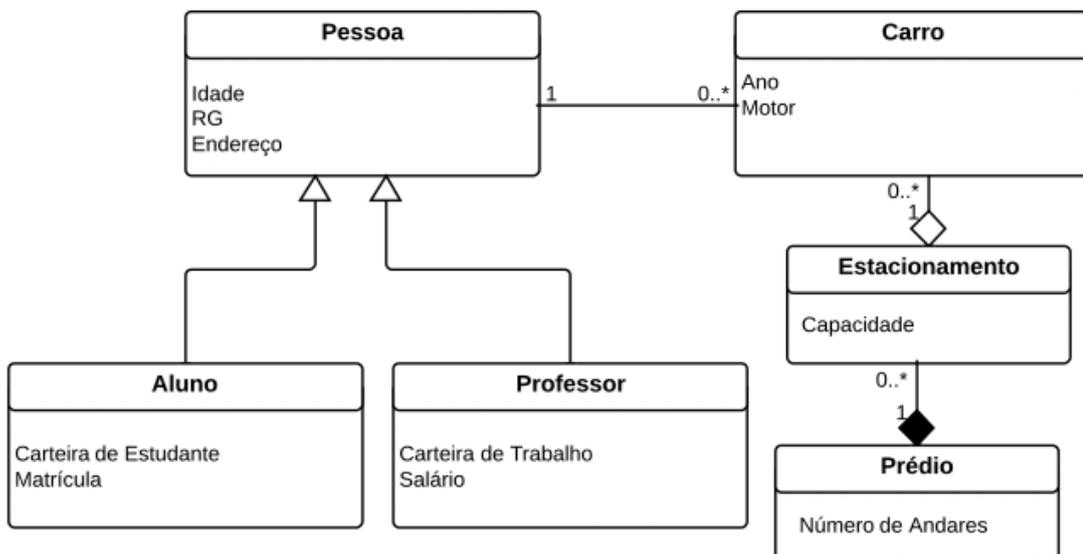
Figura 2.5. UML: exemplo de agregação.



### 2.3.1.5. Composição

A composição é um tipo ainda mais forte de associação do que a agregação. Assim como esta última, a composição também estabelece uma relação de pertencimento. No entanto, a diferença está no fato de que na composição a existência da classe-conteúdo é totalmente condicionada à existência da classe-container. A relação de composição é representada por uma linha cheia que possui um diamante preenchido no lado da classe-container. A Figura 2.6 apresenta um exemplo.

Figura 2.6. UML: exemplo de composição.



### 2.3.2. A norma IEC 61970-301

A norma IEC 61970-301 apresenta a base funcional do CIM. Nela, apresentam-se os principais elementos que compõem um sistema elétrico em diversos diagramas de classe, que usa o inglês britânico como idioma padrão. Apesar de poder parecer, inicialmente, uma maneira confusa de representar os componentes de um sistema de potência, essa representação mostra-se, ao contrário, bastante intuitiva e simplificadora.

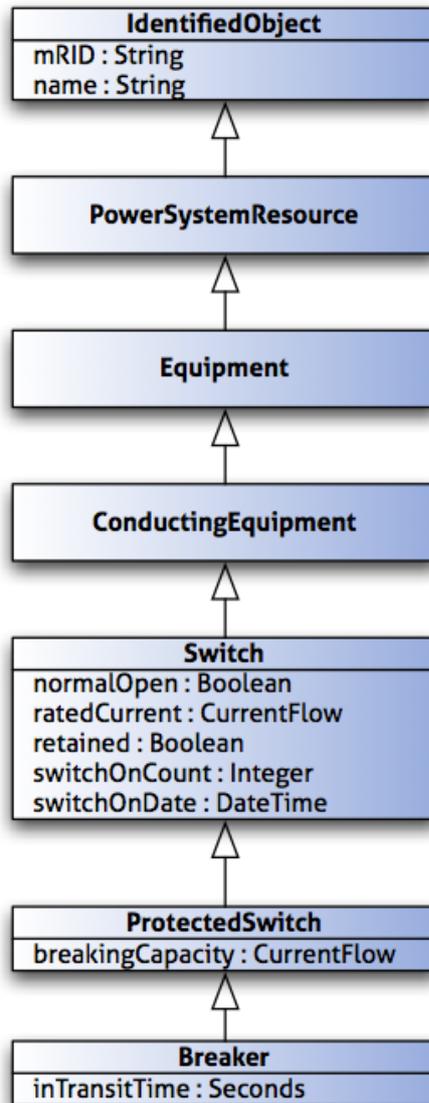
Suponha-se, por exemplo, que se queira modelar um disjuntor (*Breaker*), um dos elementos mais comuns na topologia de redes elétricas. No seu nível básico mais especializado, o disjuntor é um elemento mecânico responsável por conduzir ou bloquear a corrente que o percorre sob determinadas condições do circuito. O disjuntor pode ser considerado um tipo de chave protegida (*ProtectedSwitch*). Ele também pode ser caracterizado como um tipo de equipamento que conduz eletricidade (*ConductingEquipment*) e, mais genericamente, um equipamento (*Equipment*). Por ser um equipamento, o disjuntor pode, igualmente, ser um tipo de recurso do sistema (*PowerSystemResource*) e um objeto identificado (*IdentifiedObject*) (EPRI, 2011, p. 5-3).

Pode-se dizer, assim, que o disjuntor é um objeto identificado, um tipo de recurso do sistema, um tipo de equipamento, um tipo de equipamento condutor, um tipo de chave e um tipo de chave protegida. Isso pode ser expresso de outra forma, como uma série de generalizações presente na Figura 2.7.

Seguindo essa ideia, os diagramas de classes foram estruturados de forma a criar uma lógica que facilitasse na busca pelos diferentes componentes que precisam ser modelados.

Para organizar as cerca de 900 classes presentes na versão mais atual do CIM (v. 15), o modelo foi dividido em pacotes. A divisão das classes em pacotes teve o objetivo único de organizar o modelo de modo a facilitar seu uso, não devendo ser entendida como uma barreira que dificulte de alguma forma a relação entre as classes. Há inúmeras e importantes associações entre classes pertencentes a pacotes diferentes.

Figura 2.7. UML: generalizações do elemento disjuntor.



Fonte: (EPRI, 2011, p. 5-4).

O número de pacotes tem aumentado à medida que o CIM tem se desenvolvido. Aqui, serão mencionados os principais, que aloca as classes que representam elementos fundamentais e mais comuns dos sistemas de energia.

#### 2.3.2.1. Domain

O pacote Domain é um dicionário de unidades usado para definir os atributos dos tipos de dados utilizados em todos os outros pacotes. Há, usualmente, um atributo para designar o valor dos dados que são quantificáveis. Esse atributo possui

um tipo de dado correspondente (metro, caloria, corrente, tensão, temperatura, etc.) presente no pacote Domain.

#### 2.3.2.2. Core

Esse pacote contém duas das principais classes utilizadas na modelagem de sistemas elétricos: *PowerSystemResource* e *ConductingEquipment*. A grande maioria das classes presentes nos outros pacotes possui algum tipo de associação com uma dessas classes.

Além disso, esse pacote apresenta as principais classes que auxiliam na representação completa dos componentes físicos dos sistemas elétricos. Como exemplo, mencionam-se as classes *Terminal* e *BaseVoltage*, que são necessárias para representar conexões entre equipamentos condutores e seus níveis de tensão, respectivamente.

#### 2.3.2.3. OperationalLimits

O pacote *OperationalLimits* apresenta classes que modelam os limites operacionais que podem ser associados a determinadas classes de equipamentos. As classes *CurrentLimit* e *ActivePowerLimit* são exemplos.

#### 2.3.2.4. Topology

O pacote *Topology* contém as classes responsáveis por possibilitar a modelagem da conectividade física dos equipamentos presentes num sistema. Pode ser considerado uma extensão do pacote *Core*, visto que por meio de associações com sua classe *Terminal*, as classes de *Topology* são extremamente úteis para representar o modo como os equipamentos estão conectados entre si. Diversos aspectos da topologia de uma rede podem ser representados fazendo uso de suas classes. A classe *TopologicalNode* é um exemplo importante desse pacote.

#### 2.3.2.5. *Wires*

Extensão dos pacotes *Core* e *Topology*, o pacote *Wires* traz as classes que representam a maior parte dos equipamentos elétricos. A maior parte dos dados referentes à transmissão e distribuição de energia, como fluxo de carga e estimação de estado, são provenientes de propriedades das classes desse pacote.

#### 2.3.2.6. *Generation.Production*

Subdivisão do pacote *Generation*, o pacote *Generation.Production* contém as classes utilizadas para representar diversos tipos de geradores. Além dos aspectos técnicos importantes para sua descrição, as classes também possuem propriedades úteis para possíveis análises econômicas e de qualidade de energia, como custo de geração e taxa de emissão. *HydroGeneratingUnit* é um exemplo de classe.

#### 2.3.2.7. *Generation.Dynamics*

A outra subdivisão do pacote *Generation*, *Generation.Dynamics* apresenta as classes dos elementos móveis, como turbinas e câmaras de ebulição, envolvidos no processo de geração de energia. *SteamTurbine* e *HeatRecoveryBoiler* são exemplos.

#### 2.3.2.8. *LoadModel*

Esse pacote contém as classes necessárias para a modelagem de dados referentes à demanda de energia, como informações sobre comportamento sazonal e áreas críticas.

#### 2.3.2.9. *Outage*

Extensão dos pacotes *Core* e *Wires*, o pacote *Outage* apresenta classes necessárias para a efetuação de operações programadas em determinados equipamentos. As classes *OutageSchedule* e *CleranceTag*, por exemplo, são

utilizadas quando se deseja especificar o período agendado em que um determinado equipamento estará fora de serviço (para manutenção ou teste).

#### 2.3.2.10. *Protection*

Também uma extensão de *Core* e *Wires*, *Protection* traz classes úteis para modelar informações sobre equipamentos de proteção de redes elétricas. Esse pacote é especialmente importante para sistemas de simulação de falta e de recomposição. A classe *CurrentRelay* é um exemplo importante.

#### 2.3.2.11. *Equivalentents*

Pacote que contém classes necessárias para a modelagem de esquemas equivalentes de redes, trechos ou equipamentos.

#### 2.3.2.12. *Meas*

O pacote *Meas* apresenta as classes que possibilitam a troca de dados sobre diversos tipos de medidas, sobretudo de natureza elétrica. Mediante a associação da classe *Measurement* com a classe *Terminal* (do pacote *Core*), é possível transmitir medidas analógicas e digitais de diversos tipos (corrente, tensão, potência, energia, etc.).

#### 2.3.2.13. *SCADA*

*SCADA*, como o próprio nome sugere, é o pacote que contém as classes que auxiliam na modelagem do fluxo de dados referentes aos sistemas *SCADA* (*Supervisory Control and Data Acquisition*). Sistemas de aquisição de dados e supervisão reúnem uma série de informações provenientes de dispositivos remotamente localizados. Por isso, as classes *RemoteSource* e *RemotePoint* são bons exemplos desse pacote.

#### 2.3.2.14. *ControlArea*

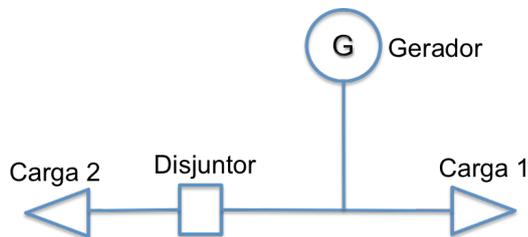
O pacote *ControlArea* apresenta algumas classes que podem ser utilizadas quando se deseja agregar determinados grupos funcionais, como algumas unidades geradoras, cargas ou trechos.

### 2.3.3. Modelo de Conectividade

Um dos grandes desafios quando se deseja transmitir dados sobre um sistema elétrico é poder fornecer informações a respeito da topologia da rede. Seja qual for o vocabulário usado, descrever de forma escrita a maneira como os equipamentos elétricos estão conectados entre si representa uma tarefa difícil de ser realizada.

No CIM, ao invés de haver uma associação direta entre as classes dos componentes que estão conectados entre si, utilizam-se as classes *Terminal* e *ConnectivityNode* (EPRI, 2011, p. 5-7). Para compreender sua utilização, considere o exemplo da Figura 2.8.

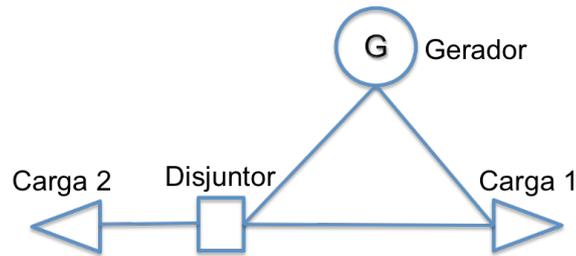
Figura 2.8. Exemplo de conectividade.



O circuito acima contém um gerador, um disjuntor e duas cargas, que possuem classes correspondentes no CIM: *AssynchronousMachine*, *Breaker* e *EnergyConsumer*, respectivamente.

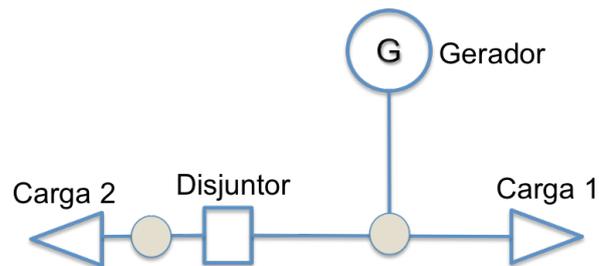
Se a conexão entre o gerador, o disjuntor e a Carga 1 fosse descrita como uma relação direta entre eles, a leitura dos dados poderia gerar mal entendimento. Por exemplo, se num arquivo de dados qualquer fosse dito que o Disjuntor está ligado ao Gerador e à Carga 1, que o Gerador está ligado à Carga e ao Disjuntor e que a Carga 1 está ligada ao Gerador e ao Disjuntor, uma possível topologia que está de acordo com essas informações seria a da Figura 2.9.

Figura 2.9. Topologia com associação direta entre equipamentos.



Para evitar esse problema, o CIM utiliza a classe *ConnectivityNode* (nó conectivo). Essa classe funciona como um ponto de conexão de 2 ou mais componentes. Quando se descreve a topologia da rede da Figura 2.8 como um único ponto de conexão interligando os 3 elementos, há apenas uma forma de interpretar tais informações, conforme a Figura 2.10.

Figura 2.10. *ConnectivityNode* conectando elementos.



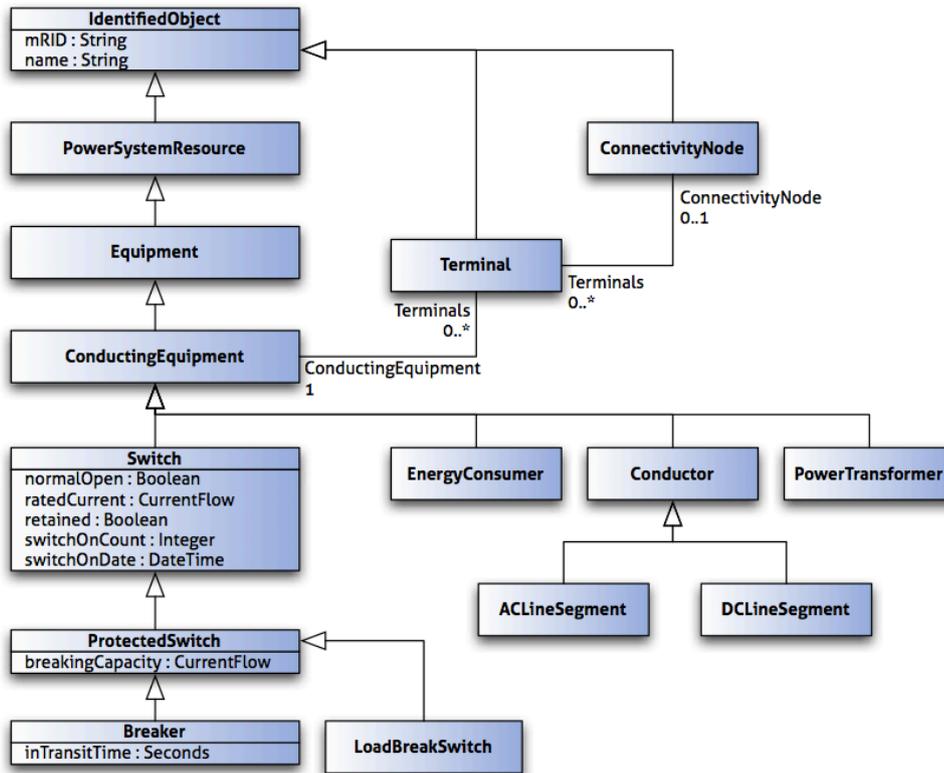
No entanto, a classe *ConnectivityNode* não se associa diretamente às classes dos equipamentos, e sim à classe *Terminal*, que representa os terminais que o equipamento pode ter para se conectar a outros. Para compreender corretamente como se dão as associações entre as classes *Terminal* e *ConnectivityNode*, a análise do diagrama UML dessa parte do CIM, na Figura 2.11, pode facilitar.

Primeiramente, observa-se que a classe *Terminal* possui uma associação com *ConductingEquipment* de multiplicidade 1 e [0..\*]. Isso significa dizer que cada terminal específico só pode ser associado a um único equipamento condutor, enquanto cada equipamento condutor pode ter de zero (no caso de estar eletricamente isolado) a inúmeros terminais, dependendo de a quantos outros equipamentos condutores ele está conectado.

Além disso, nota-se, como mencionado anteriormente, que a classe *ConnectivityNode* não possui uma associação direta com *ConductingEquipment*. Sua associação com a classe *Terminal*, por sua vez, possui multiplicidade [0..\*] e

[0..1]. Ou seja, cada nó conectivo pode estar associado a zero ou inúmeros terminais, enquanto cada terminal pode estar associado a zero ou a apenas um nó conectivo.

Figura 2.11. UML: Diagrama de conectividade.



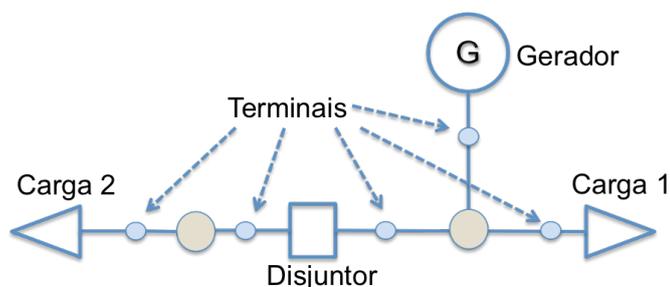
Fonte: (EPRI, 2011, p. 5-9).

Apesar de simplificar e facilitar a transmissão de dados sobre a topologia, a existência dos terminais pode parecer dispensável para descrever a maneira como os equipamentos estão conectados. Transmitir a informação de que um determinado nó conectivo está conectando um disjuntor, um gerador e uma carga seria suficiente para a correta interpretação da topologia da rede. No entanto, a classe *Terminal* foi inserida no CIM por ser necessária para a transmissão correta de dados de medição. Para exemplificar essa necessidade, considere-se a extensão do circuito representada na Figura 2.12.

Se o Disjuntor da figura estiver conduzindo, todos os parâmetros medidos em ambos os seus terminais serão idênticos. Nesse caso, transmitir a informação “Disjuntor está submetido a uma tensão V, uma corrente I e uma potência P” seria interpretada de maneira correta. No entanto, se o Disjuntor estiver em estado de

não-condução, tal afirmativa não será correta, visto que os parâmetros em ambos os seus terminais possivelmente não serão os mesmos. Dessa forma, faz-se indispensável a utilização dos terminais.

Figura 2.12. Exemplo de terminais.



Outra observação que se faz a respeito do diagrama UML da Figura 2.11 é que a classe *Terminal* associa-se apenas a *ConductingEquipment*, e não a *Equipment*. Esse fato é coerente com a realidade física dos circuitos elétricos, visto que apenas os equipamentos condutores de eletricidade podem se conectar a outros. As classes de equipamento que não são generalizações de *ConductingEquipment*, dessa forma, não podem ser associadas a terminais.

#### 2.3.4. Extensibilidade

Por tratar-se de um modelo aberto e que visa servir como base de dados para diferentes tipos de sistemas elétricos, o diagrama de classes que compõe o CIM busca ser o mais genérico e abrangente possível. No entanto, o CIM não deve ser entendido como um modelo fixo e rígido. Em decorrência do contínuo processo de desenvolvimento dos sistemas elétricos e dos inúmeros tipos de aplicações às quais o CIM pode ser utilizado, é possível que se efetuem extensões no modelo.

Além das extensões efetuadas e publicadas pela própria IEC (como será visto na próxima seção), é possível que os próprios usuários do CIM façam extensões do modelo de acordo com as necessidades de cada caso (EPRI, 2011, p. 10-1). Para efetuar as extensões desejadas, basta seguir as regras básicas da UML de acordo com a lógica de organização do CIM.

Há um crescente número de trabalhos que visam gerar blocos funcionais de extensão do CIM. Em Ding *et al.* (2009) e Wang *et al.* (2012), por exemplo, tem-se

extensões do CIM que visam auxiliar na modelagem de sistemas de microgeração, apresentando classes de célula-combustível, painéis fotovoltaicos, etc. Em Lijun *et al.* (2011), sugere-se uma extensão para informações mais precisas sobre diagnóstico de faltas em sistemas elétricos. Em Pradeep *et al.* (2011), há uma proposta de extensão na qual se cria uma nova classe para simplificar a representação de topologias a nível de Barramento/Trecho.

#### **2.4. IEC 61968-11 e IEC 62325-301: extensões oficiais**

O diagrama UML presente na norma IEC 61970-301 possui duas extensões oficiais, também elaboradas pelo IEC.

A norma IEC 61968-11 apresenta classes que auxiliam na modelagem de dados utilizados entre companhias de distribuição de energia. Tanto classes necessárias para especificações técnicas, como *Asset* (classe responsável pelas especificações de um determinado equipamento físico, como número de série, datas de funcionamento, preço, vida útil, etc.), até dados concernentes aos consumidores, como *Customer* (número de identificação, tipo de consumidor, tipo de contrato com a companhia, necessidades especiais, tipo de tarifa, etc.).

A segunda extensão, a norma IEC 62325-301, apresenta classes que podem auxiliar na troca de dados entre companhias envolvidas no mercado de energia. Classes como *BidTimeSeries* e *Auction* são úteis para transmitir, respectivamente, dados sobre lances e leilões.

Nesse trabalho, no entanto, foram utilizadas apenas as classes presentes na norma IEC 61970-301. Como o objeto de aplicação da norma foi a integração de sistemas responsáveis pela supervisão de um típico circuito de distribuição em alta e média tensão, todos os componentes presentes na rede (transformadores, disjuntores/religadores, barramentos) possuem classes correspondentes no modelo principal. Assim, não foi necessário criar nenhuma extensão e nem utilizar nenhuma das duas normas de extensões existentes.

#### **2.5. Utilização da norma**

Após conhecer e se familiarizar com a estrutura e a composição dos diagramas UML que integram o CIM, é possível implementar a primeira etapa na utilização do

modelo. Deve-se ter em mente constantemente, entretanto, o foco principal da utilização do CIM: utilizar um modelo de dados comuns para a modelagem de sistemas que viabilize a integração de aplicações heterogêneas.

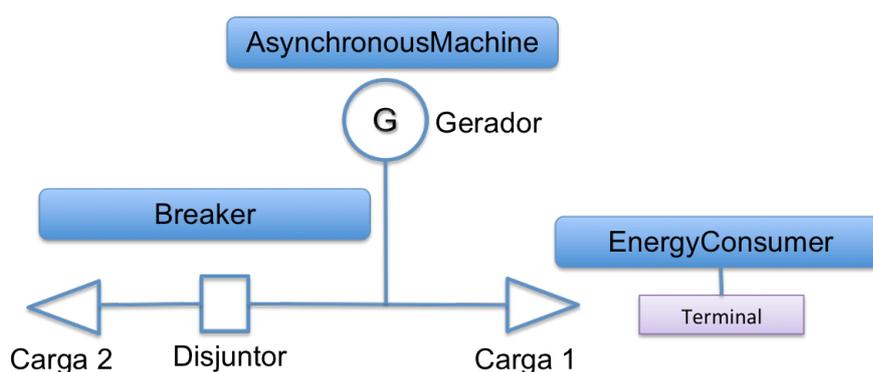
O primeiro passo para alcançar essa meta é, após definir os objetos e parâmetros que se deseja compartilhar, encontrar suas classes correspondentes nos diagramas do CIM. Tal processo, denominado mapeamento (USLAR *et al.*, 2011, p. 80) (EPRI, 2011, p. 2-4), deve ser primeiro feito como uma forma de “rascunho”, de forma a deixar claro quais dados deseja-se transmitir e, principalmente, a maneira como esses dados estão organizados. No Capítulo 3, será abordado como esse procedimento deve ser efetuado para auxiliar na geração do arquivo de dados.

Para a correta compreensão de como efetuar o mapeamento, considere-se novamente o exemplo ilustrado na Figura 2.8. Hipoteticamente, deseja-se transmitir os seguintes dados:

- a) Potência nominal do Gerador;
- b) Estado do Disjuntor;
- c) Corrente na Carga 1.

Encontram-se, primeiramente, as classes que modelam os objetos cujos parâmetros deseja-se transmitir. A Figura 2.13 ilustra essa etapa.

Figura 2.13. Exemplo de mapeamento.

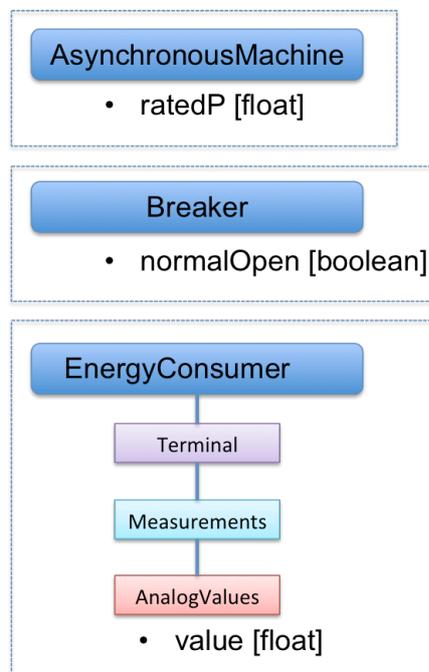


Em seguida, mapeiam-se os parâmetros pertencentes a cada uma das classes. O estado do disjuntor, por exemplo, é representado, no CIM, pelo parâmetro “normalOpen” pertencente à classe *Switch*. Já a corrente que chega à carga é representada pelo atributo “value” pertencente à classe *AnalogValues*, que possui associação com a classe *Measurements*, que possui associação com a classe

*Terminal*. Ou seja, os dados de medidas sempre estão associados aos terminais, nunca a um elemento específico. Visto que somente os elementos condutores possuem associação com o elemento Terminal (diagrama da Figura 2.11), só é possível representar dados de medidas de elementos condutores, como *EnergyConsumer*.

Ao final do mapeamento, tem-se um diagrama UML simplificado com as classes e os parâmetros. Uma das características mais importantes desse diagrama é que, por meio dele, é possível visualizar a forma como as informações estão organizadas. A Figura 2.14 apresenta o resultado do processo de mapeamento.

Figura 2.14. Diagrama de mapeamento.



Finalizado o processo de mapeamento, na próxima etapa deve-se definir um formato para “traduzir” o diagrama mapeado para uma forma escrita e transmissível. Visto que há diversas possibilidades de linguagem e que a UML não condiciona a escolha a nenhuma específica, o CIM, que pretende ser o mais genérico e interoperável possível, faz essa escolha. Esse é o tema do próximo capítulo.

---

**CAPÍTULO 3. EXTENSIBLE MARKUP LANGUAGE**

---

Como visto no capítulo anterior, a norma IEC 61970-301, suas extensões e os diagramas UML que as compõem são fundamentais para definir os dados que devem ser transmitidos e a forma como esses dados devem ser estruturados. Para adquirir uma utilidade prática no processo de integração de sistemas, no entanto, é preciso estabelecer uma maneira de converter os diagramas para formatos transmissíveis. O processo de codificar um objeto, de maneira a representá-lo em um formato transmissível e armazenável, é chamado de serialização.

A UML não condiciona a escolha do usuário a que tipo de formato deve ser utilizado para serializar os dados representados. Tendo em consideração o objetivo do CIM de ser utilizado para integração de sistemas, a busca por interoperabilidade sugere o uso de um formato que seja o mais genérico possível. Assim, consta da série de normas do CIM que a linguagem que deve ser empregada para a serialização dos objetos é o XML (*eXtensible Markup Language*).

### **3.1. Características da linguagem XML**

A XML, que foi criada em 1996 como um formato bem mais simplificado do SGML (*Standard Generalized Markup Language*) (FAWCETT *et al.*, 2012, p. 6), é, atualmente, uma das linguagens mais utilizadas quando se deseja transmitir e receber informações em redes de computadores. Seu potencial é tão expressivo que apenas 2 anos após sua criação, a W3C (*World Wide Web Consortium*) elaborou um documento, em 1998 (FAWCETT *et al.*, 2012, p. 6), recomendando a utilização da XML como formato a ser utilizado para o estabelecimento de fluxo de dados na internet.

A XML possui características que tornam seu uso extremamente vantajoso. Por ser representada num formato textual e estruturado, a representação de dados em XML pode ser interpretada tanto por computadores quanto por usuários. Além disso, por já ser uma linguagem padrão para fluxo de informação, há inúmeros aplicativos *open-source* de escrita/leitura e de avaliação de arquivos XML. O seu uso

independe, assim, da plataforma utilizada pelos aplicativos computacionais. Dessa maneira, a principal vantagem em se utilizar a XML é a sua consequente interoperabilidade (FAWCETT *et al.*, 2012, p. 12).

Outra vantagem da XML é que ela é extensível. Diferentemente da maior parte das outras linguagens, que possuem a grande parte do seu vocabulário rígido, a XML permite a livre utilização de palavras para nomear seus componentes da maneira como melhor convier ao programador. Essa característica torna esse formato idealmente flexível para transmitir informações de qualquer natureza.

## 3.2. Sintaxe

Há duas unidades básicas que compõem os documentos XML: elementos e atributos.

### 3.2.1. Elementos

Os documentos XML são estruturados em elementos. Assim como em outras linguagens de marcação, cada elemento deve possuir um nome e estar demarcado por *tags* iniciais (<nome\_da\_tag>) e finais (</nome\_da\_tag>). A Figura 3.1 apresenta um exemplo de como os elementos são representados em XML.

Figura 3.1. Exemplo de elemento em XML.

```
<elemento>  
    ... conteúdo ...  
</elemento>
```

Quando há um elemento no interior de outro, diz-se que ele é um elemento-filho do elemento mais externo.

### 3.2.2. Atributos

Os atributos são características dos elementos aos quais pertencem e são representados (quando existem) no interior da *tag* inicial do elemento, do lado do seu nome. A Figura 3.2 traz um exemplo.

Figura 3.2. Exemplo de elemento com atributo em XML.

```
<elemento atributo="valor_do_atributo">
    ... conteúdo ...
</elemento>
```

Utilizando essas 2 unidades que compõem os arquivos XML, é possível construir documentos que contenham informações. Os nomes dos elementos e dos atributos, em decorrência da extensibilidade da XML, podem ser designados da maneira que melhor se enquadrar à natureza da informação e à escolha do programador.

Para exemplificar o uso da XML para representar informação, considere-se o exemplo de um sistema de armazenamento de dados dos alunos de uma universidade. Deseja-se transmitir as seguintes informações de 3 estudantes: seu número de identidade e as matérias em que estão matriculados. Os dados são apresentados na Tabela 3.1.

Tabela 3.1. Dados de estudantes para exemplo de uso do XML.

Aluno 1	Aluno 2	Aluno 3
RG: 1234567	RG: 9876543	RG: 56789
Matemática	Química	Física
Física	Biologia	Química

Uma das possíveis maneiras de representar-se esses dados em XML é ilustrada na Figura 3.3.

Figura 3.3. Documento XML com dados da Tabela 3.1.

```
<alunos>
  <aluno RG="1234567">
    <disciplina1>Matematica</disciplina1>
    <disciplina2>Fisica</disciplina2>
  </aluno>
  <aluno RG="9876543">
    <disciplina1>Quimica</disciplina1>
    <disciplina2>Biologia</disciplina2>
  </aluno>
  <aluno RG="56789">
    <disciplina1>Fisica</disciplina1>
    <disciplina2>Quimica</disciplina2>
  </aluno>
</alunos>
```

Há inúmeras outras formas de representar os mesmos dados em XML. O atributo RG, por exemplo, poderia ser transformado em um elemento-filho e a informação estaria igualmente completa. Não há apenas uma estrutura correta. A forma como se estruturam as informações deve levar em consideração a maneira que melhor traduz os objetivos e as características, tanto do remetente quanto do receptor do arquivo contendo a mensagem.

Em decorrência dessa “maleabilidade” das estruturas dos documentos XML, é importante que se tenha uma maneira de averiguar que a estrutura e o conteúdo do arquivo estão no formato que foi estabelecido. Essa é a função do XML Schema Definition (XSD).

### 3.3. XML Schema Definition

A extrema flexibilidade com que se pode construir documentos em XML pode gerar um problema fundamental. Visto que há inúmeras maneiras de estruturar corretamente as mesmas informações, é preciso encontrar uma maneira de garantir que determinado arquivo está construído conforme o que o remetente e o receptor da mensagem acordaram.

Essa é a função do XML Schema Definition (XSD). Como o nome sugere, o XSD apresenta um esquema que define como determinado documento deve ser estruturado. Entre outros fatores, o esquema pode estabelecer os elementos e

atributos que devem estar presentes na mensagem, a quantidade e a ordem de cada um, e os tipos de variáveis que cada atributo ou elemento deve possuir.

Antes da criação do XSD, utilizava-se um arquivo chamado *Document Type Definition* (DTD) para validar arquivos XML. Apesar de terem a mesma função, o DTD possuía duas desvantagens principais: possuía uma sintaxe própria e não contemplavam os *namespaces*. *Namespaces* são usados para agrupar determinados elementos e atributos de forma a diferenciá-los de elementos e atributos que possuam o mesmo nome (FAWCETT *et al.*, 2012, p. 54). Assim, além de o XSD usar a própria linguagem XML para sua construção, ele permite a utilização de *namespaces*.

Para demonstrar como o XSD é usado para determinar a estrutura do documento XML, a Figura 3.4 apresenta o arquivo .xsd que indica a estrutura apresentada na Figura 3.3.

Figura 3.4. Exemplo de XSD.

```
1 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
2 <xs:element name="alunos">
3 <xs:complexType>
4 <xs:sequence>
5 <xs:element maxOccurs="unbounded" ref="aluno"/>
6 </xs:sequence>
7 </xs:complexType>
8 </xs:element>
9 <xs:element name="aluno">
10 <xs:complexType>
11 <xs:sequence>
12 <xs:element ref="disciplina1"/>
13 <xs:element ref="disciplina2"/>
14 </xs:sequence>
15 <xs:attribute name="RG" use="required" type="xs:integer"/>
16 </xs:complexType>
17 </xs:element>
18 <xs:element name="disciplina1" type="xs:NCName"/>
19 <xs:element name="disciplina2" type="xs:NCName"/>
20 </xs:schema>
```

Entre outras instruções constantes nesse código, verifica-se, entre as linhas 2 e 8, que o elemento “alunos” possui uma sequência de elementos do tipo “alunos”. Entre as linhas 9 e 17, verifica-se que cada elemento “aluno” deve possuir uma sequência com os elementos “disciplina1” e “disciplina2”, além do atributo “RG”.

Na linha 5, especificamente, o atributo “maxOccurs” estipula que pode haver inúmeros elementos “aluno” no interior do elemento “alunos”. Pode-se, dessa forma,

conferir uma margem de mudança permissível para os documentos XML que devem estar de acordo com o esquema. Assim, o uso do XSD não significa que a mensagem enviada/recebida deve seguir um modelo imutável. O arquivo XML pode sofrer determinadas mudanças em sua estrutura que são permitidas pelo esquema.

Percebe-se, então, que o XSD é um item fundamental quando se deseja estabelecer a interoperabilidade de sistemas. Ele funciona como uma espécie de “gabarito” que certifica que os sistemas remetente e receptor estão trocando mensagens cujas informações são fidedignas. Assim, há diversas funções em linguagens de programação e até aplicativos que são utilizados para validar documentos XML de acordo com determinado XSD.

### **3.4. Modelo CIM e linguagem XML**

Para desenvolver um modelo que seja interoperável, o CIM, em sua série de normas, aconselha o uso do XML como formato padrão a ser utilizado no estabelecimento da comunicação entre sistemas heterogêneos. A flexibilidade dessa linguagem, somada ao seu uso consolidado no processo de transmissão de dados, confere ao CIM um elevado potencial de integração.

No entanto, antes de se analisar a maneira como o modelo orienta a utilização da XML, é preciso destacar, assim como as próprias normas que definem o CIM fazem, uma limitação fundamental na XML: o único tipo de relacionamento entre elementos que se pode representar é do tipo hierárquico “pai-filho”. Para transpor essa limitação, há a alternativa de se utilizar o chamado *Resource Description Framework* (RDF), uma espécie de sub-linguagem derivada da XML que será explicada na subseção seguinte.

A escolha entre a XML básica e a RDF deve ser feita pelo programador incumbido pela modelagem do sistema, visto que a utilização do CIM não está condicionada a nenhuma das duas opções. Logo, como já foram explicados os fundamentos da XML, resta conhecer as características da RDF e fundamentar a escolha feita nesse trabalho para a aplicação do CIM.

### 3.4.1. Resource Description Framework (RDF)

A RDF é uma linguagem que foi desenvolvida para representar metadados sobre recursos na internet (W3C, 2004). Apesar de possuir uma definição gráfica formalizada, considera-se como essencial para a compreensão do presente trabalho unicamente o conhecimento da representação da RDF em XML, denominada RDF/XML.

O objetivo que se busca ao se utilizar a RDF/XML para representar dados, é inserir, no próprio documento, informações sobre os próprios dados. Mais especificamente, o uso da RDF/XML torna possível o estabelecimento de tipos diferentes de relacionamento entre os elementos que não a relação “pai-filho”.

Para entender corretamente como isso funciona, considere-se o exemplo dos dados de uma determinada família, representados na Tabela 3.2.

Tabela 3.2. Dados de exemplo de uma família.

Família da Silva		
Irmão	Pai	Irmã
CPF: 001948	CPF: 641927	CPF: 875131
21 anos	55 anos	25 anos
Engenheiro	Médico	Advogada

As informações que constam na tabela, além da relação de parentesco entre os elementos, são representadas no código em RDF/XML da Figura 3.5.

Figura 3.5. Documento RDF/XML com dados da Tabela 3.2.

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <familia rdf:ID="F1">
    <familia.nome>da Silva</familia.nome>
  </familia>
```

```

<pai rdf:ID="641927">
  < Pessoa.CPF>641927</ Pessoa.CPF>
  < Pessoa.idade>55</ Pessoa.idade>
  < Pessoa.profissao>Medico</ Pessoa.profissao>
  < Pessoa.familia rdf:resource="F1" />
  < Pessoa.parentes>
    < parentes>
      < parentes.filho rdf:resource="001948" />
      < parentes.filha rdf:resource="875131" />
    </ parentes>
  </ Pessoa.parentes>
</ pai>

<irmao rdf:ID="001948">
  < Pessoa.CPF>001948</ Pessoa.CPF>
  < Pessoa.idade>21</ Pessoa.idade>
  < Pessoa.profissao>Engenheiro</ Pessoa.profissao>
  < Pessoa.familia rdf:resource="F1" />
  < Pessoa.parentes>
    < parentes>
      < parentes.pai rdf:resource="641927" />
      < parentes.irma rdf:resource="875131" />
    </ parentes>
  </ Pessoa.parentes>
</ irmao>

<irma rdf:ID="875131">
  < Pessoa.CPF>875131</ Pessoa.CPF>
  < Pessoa.idade>25</ Pessoa.idade>
  < Pessoa.profissao>Advogada</ Pessoa.profissao>
  < Pessoa.familia rdf:resource="F1" />
  < Pessoa.parentes>
    < parentes>
      < parentes.pai rdf:resource="641927" />
      < parentes.irma rdf:resource="001948" />
    </ parentes>
  </ Pessoa.parentes>
</ irma>
</rdf:RDF>

```

Pode-se observar que uma das características da RDF/XML é que não há, como na XML básica, a estrutura puramente hierárquica entre os elementos presentes no documento. A RDF/XML torna possível representar relacionamentos entre classes de uma maneira que não envolva, necessariamente, algum tipo de nível ou graduação.

A despeito dessa vantagem, considerou-se, nesse trabalho, as características estruturais da XML básica mais relevantes na utilização do CIM do que as da RDF/XML. Como será visto posteriormente, os elementos fundamentais que serão modelados no estudo de caso em que o CIM foi utilizado são disjuntivos,

transformadores, linhas de transmissão, ou seja, típicos elementos condutores de eletricidade. Por isso, os relacionamentos mais frequentes são entre os elementos condutores e as classes responsáveis pela conectividade e pelos valores de medidas. Assim, seria de pouco proveito representar terminais, nós conectivos ou medidas como elementos num mesmo nível hierárquico que os próprios elementos. Nesse caso, a estrutura hierárquica da XML básica parece ser mais intuitiva e clara, fatores fundamentais quando se busca integração e comunicação entre sistemas.

Ressalta-se, novamente, que a utilização do CIM como modelo de dados não condiciona a escolha entre RDF/XML e XML básica, ficando a critério do responsável pela integração qual dos dois formatos adequa-se melhor às informações que estão sendo transmitidas.

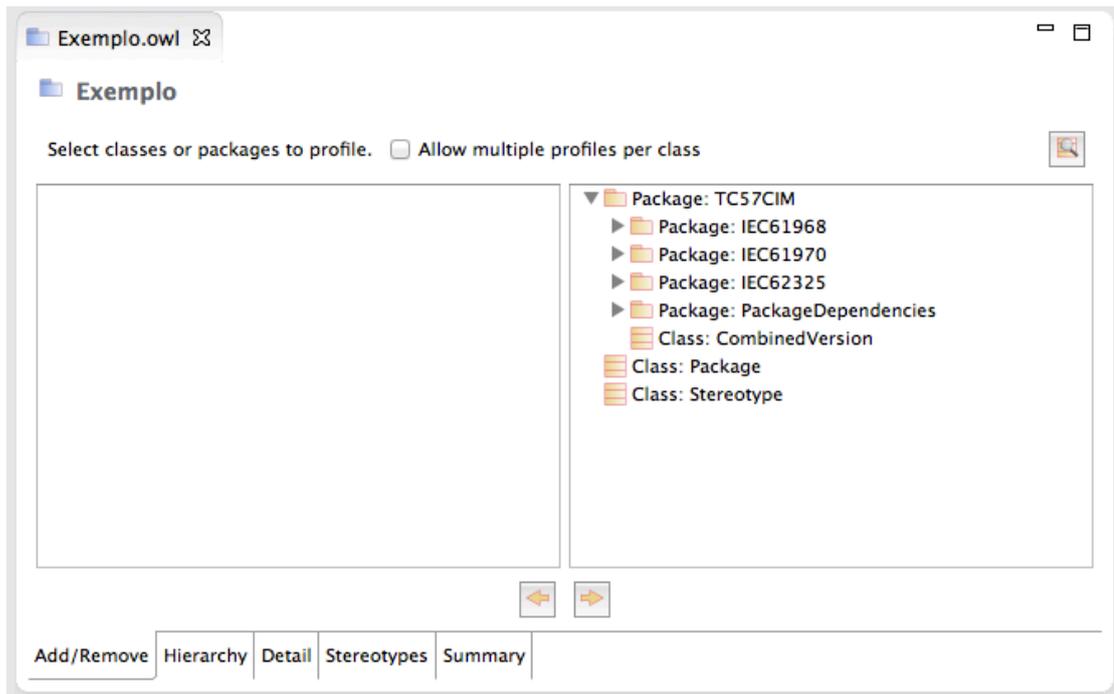
#### 3.4.2. CIMTool

Como explicado no capítulo anterior, o processo de mapeamento dos dados com auxílio dos diagramas UML resulta num modelo simplificado em que se encontram os dados a serem compartilhados e a sua estrutura. Esse modelo contextual, no entanto, possui baixo potencial operacional, visto que se trata de uma representação gráfica. Nessa segunda etapa da modelagem, é recomendável que se utilize um aplicativo que auxilie no processo de representação dos dados num formato mais adequado.

Dentre os aplicativos disponíveis para efetuar essa tarefa, utilizou-se, nesse trabalho, o CIMTool. Desenvolvido pela Langdale Consultants, o CIMTool é um *software open source* utilizado como *plug-in* da plataforma Eclipse e que utiliza como arquivo-base o documento de extensão “.xmi”, de *XML Metadata Interchange*. Arquivos “.xmi” constituem meios de transformar metadados em formato XML. O caso mais frequente do uso desse tipo de arquivo é quando se deseja transformar informações contidas num diagrama UML em XML, que é precisamente o caso que se busca nesse trabalho.

Para compreender como o CIMTool é utilizado, apresenta-se um documento de nome “Exemplo” na Figura 3.6.

Figura 3.6. Exemplo de utilização do CIMTool.

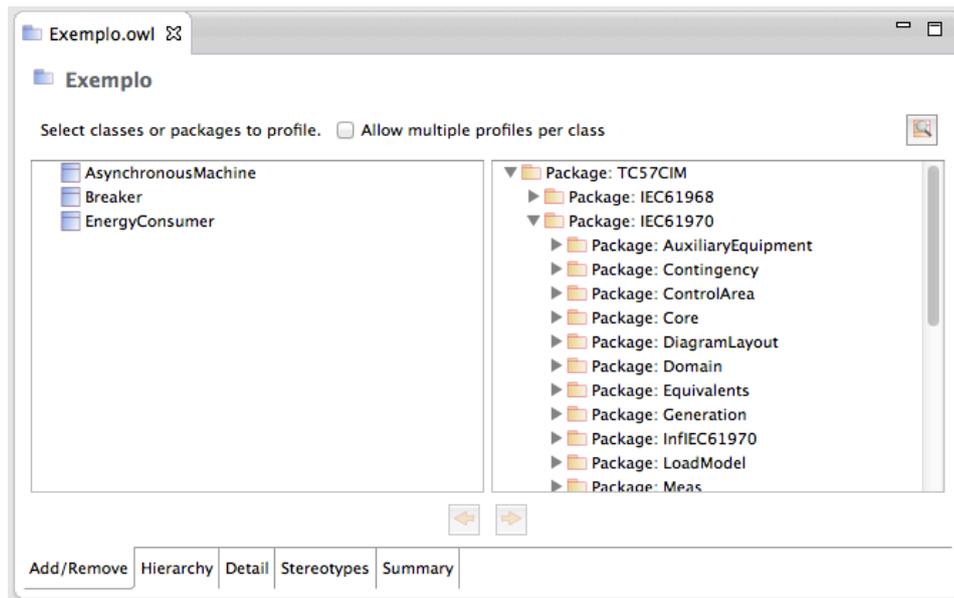


Na parte superior da tela encontra-se o título do arquivo, “Exemplo.owl”. A extensão “.owl” é usada para designar modelos de dados (ontologias) utilizadas na web. É nesse formato que o modelo contextual, mapeado anteriormente, deve ser reproduzido no CIMTool.

Para que isso seja efetuado, encontra-se, na parte lateral direita, os três pacotes que compõem o CIM (IEC61968, IEC61970 e IEC62325) na versão mais atual disponível, “iec61970cim15v33\_iec61968cim11v13\_iec62325cim01v07-cimtool.xmi”. Nesses pacotes, encontram-se as centenas de classes que compõem os diagramas UML do modelo. Assim, deve-se encontrar e selecionar as classes e atributos que compõem os dados mapeados e adicioná-las, por meio da seta direcionada para a esquerda, ao modelo contextual “.owl”.

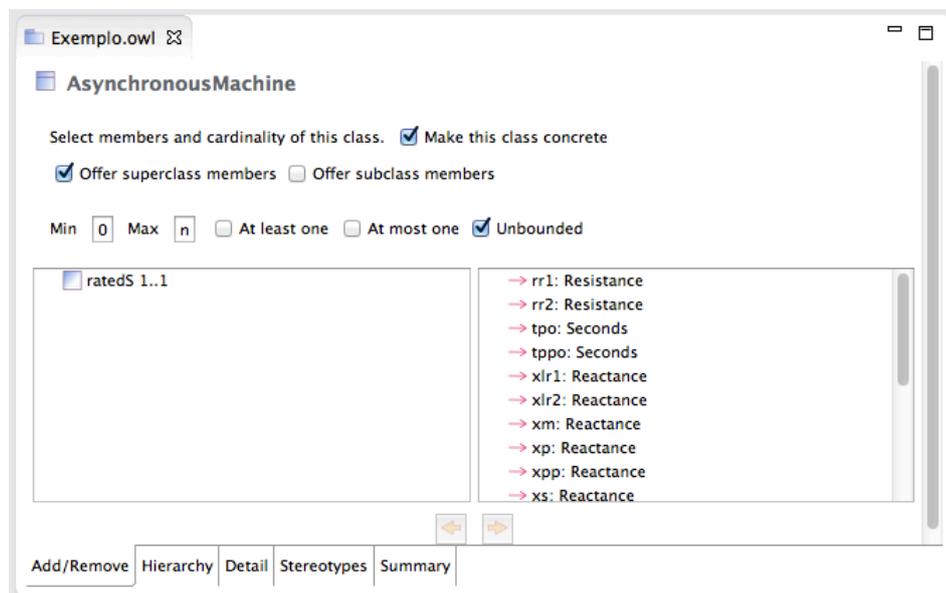
Para exemplificar a utilização prática do CIMTool, utiliza-se o modelo mapeado da Figura 2.14 que se refere ao circuito da Figura 2.8. A Figura 3.7 apresenta a tela do CIMTool após a adição das classes e atributos ao modelo contextual.

Figura 3.7. CIMTool com a adição de três classes.



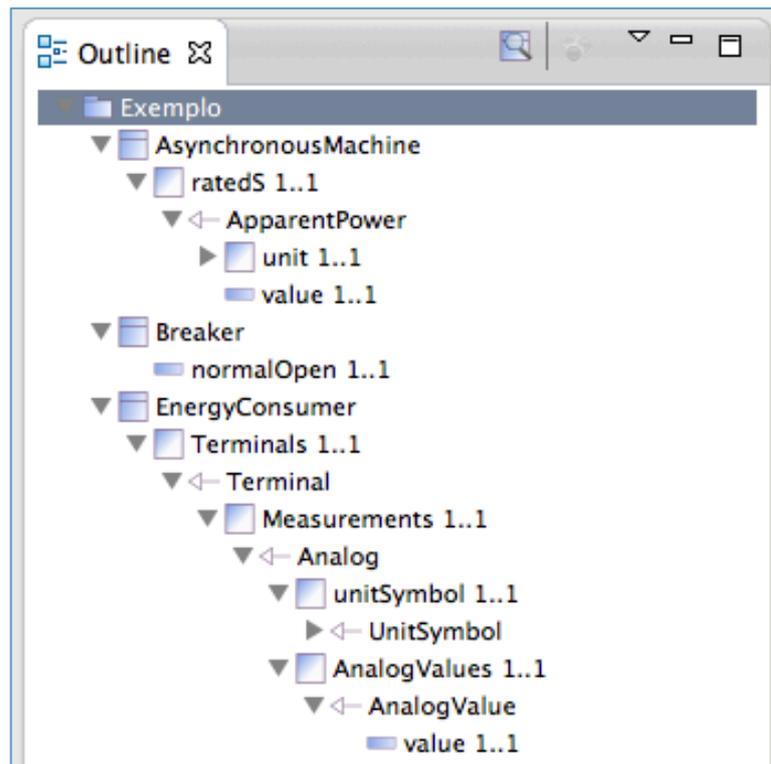
Observa-se que as classes *AsynchronousMachine*, *Breaker* e *EnergyConsumer* foram adicionadas e estão na parte lateral esquerda da tela. Para selecionar os atributos desejados, seleciona-se cada classe individualmente e os atributos a ela relacionados serão vistos na parte lateral direita da tela. A Figura 3.8 representa a adição do atributo “potência aparente” (*ratedS*) da classe *AsynchronousMachine*. Os outros atributos presentes no modelo (estado do disjuntor e corrente na carga) são adicionados analogamente.

Figura 3.8. Adição de atributo no CIMTool.



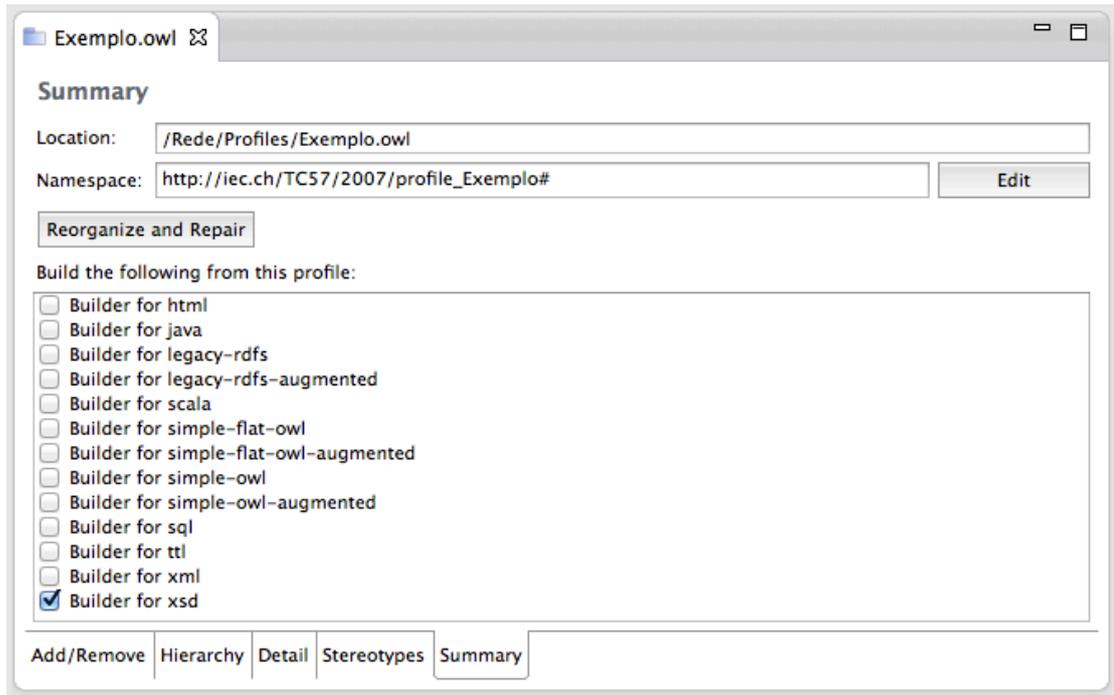
Esse procedimento, de busca e adição das classes e atributos, deve ser feito para elaborar o modelo contextual dos sistemas que se deseja modelar. A lista completa das classes e atributos que foram adicionados ao modelo é representada em uma das telas do CIMTool, que está representada na Figura 3.9.

Figura 3.9. Lista de classes e atributos do sistema modelado no CIMTool.



Terminado esse processo de encontrar e adicionar os elementos em que consistem as informações do sistema, resta obter o XSD que será gerado com base no modelo construído. Para isso, seleciona-se a aba “Summary” na tela central do CIMTool e seleciona-se a opção “Builder for xsd”. Caso se esteja utilizando o RDF/XML, deve-se selecionar a opção “Builder for legacy-rdfs”, que irá gerar o RDF Schema correspondente. A Figura 3.10 apresenta a tela do CIMTool na aba “Summary” com a opção para geração do XSD.

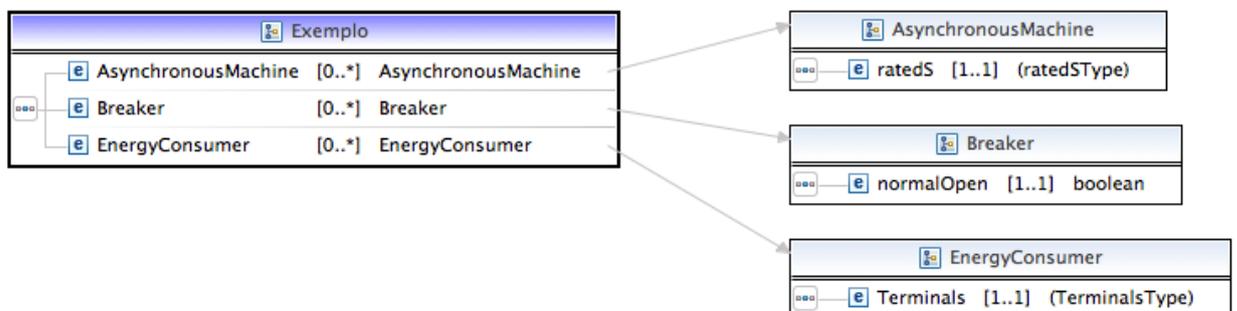
Figura 3.10. Geração do XSD no CIMTool.



Assim, o CIMTool gera o XSD que contém as instruções que orientam a construção manual do documento XML que conterá as informações a serem transmitidas. É possível, finalmente, construir um arquivo que contenha as informações desejadas e estruturadas no modelo CIM num formato que seja transmissível, armazenável e manipulável.

Visto que o código do XSD mesmo para esse exemplo simples é bastante extenso (368 linhas), a Figura 3.11 apresenta outra forma do CIMTool de representar um diagrama XSD, por meio de sua estrutura.

Figura 3.11. Estrutura do XSD no CIMTool.



O CIMTool gera o XSD, e não o próprio XML com as informações, porque há inúmeras maneiras de se construir um documento XML que obedeça ao mesmo

XSD. No exemplo acima, a multiplicidade [0..\*] de todos os três elementos torna possível que o arquivo XML possua de 0 a inúmeros elementos de cada um dos tipos.

Finalmente, seguindo as instruções do XSD, chega-se a uma possível configuração do arquivo XML contendo as informações desejadas. Na Figura 3.12, encontra-se o documento XML pronto para transmitir as seguintes informações: a potência aparente do gerador é de 20 watts, o disjuntor está no estado aberto (valor booleano 1) e a corrente consumida pela carga é de 10 ampères.

Figura 3.12. XML com informações do exemplo.

```

<Exemplo
  xmlns="http://iec.ch/TC57/2007/profile_Exemplo#"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://iec.ch/TC57/2007/profile_Exemplo# Exemplo.xsd">
  <AsynchronousMachine>
    <ratedS>
      <unit>W</unit>
      <value>20</value>
    </ratedS>
  </AsynchronousMachine>
  <Breaker>
    <normalOpen>1</normalOpen>
  </Breaker>
  <EnergyConsumer>
    <Terminals>
      <Measurements>
        <unitSymbol>A</unitSymbol>
        <AnalogValues>
          <value>10</value>
        </AnalogValues>
      </Measurements>
    </Terminals>
  </EnergyConsumer>
</Exemplo>

```

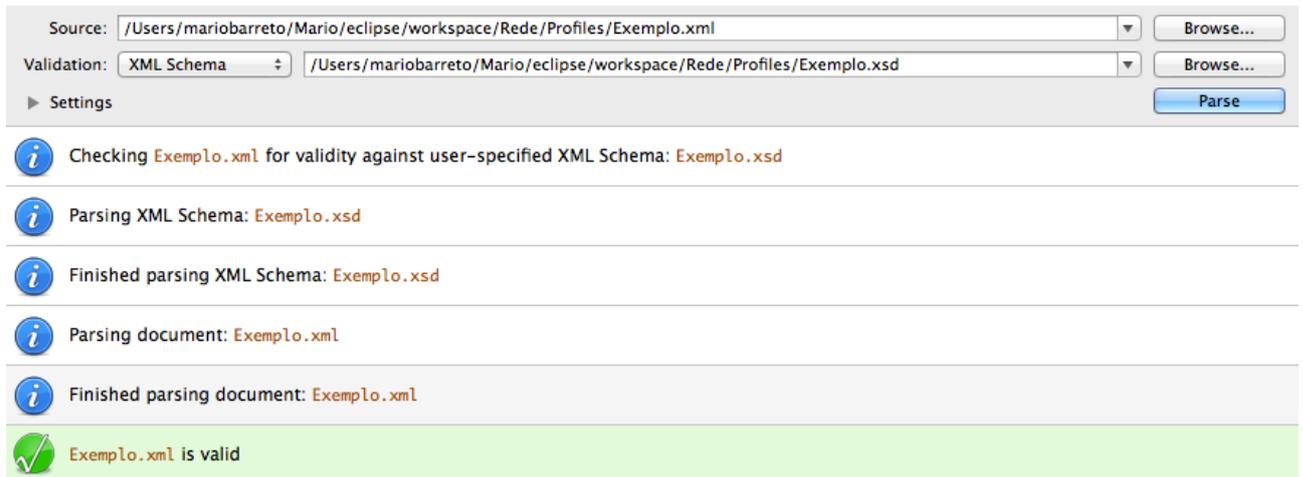
O arquivo representado na Figura 3.12 encontra-se no formato e na estrutura ideais para o estabelecimento de comunicação e a consequente interoperabilidade entre sistemas.

A utilização do CIM enseja o emprego de uma estrutura de dados que abrange boa parte das informações que são tradicionalmente necessárias em sistemas de energia e uma linguagem básica própria para a troca de dados. A importância de se trabalhar com um modelo comum de dados pode ser vislumbrada quando se considera a quantidade de aplicativos que têm a necessidade de trocar informações

entre si num ambiente integrado, típico das REI. Ainda que houvesse, num caso hipotético, a concordância pelo uso do XML como linguagem a ser utilizada, seria provável que cada responsável pela transmissão de seus respectivos dados os enviasse numa estrutura diferente.

Há vários aplicativos que se destinam a conferir se um determinado documento XML está de acordo com o XSD especificado. A mesma empresa que desenvolveu o CIMTool, inclusive, disponibiliza um aplicativo chamado CIM Validator, que tem a função de validar um arquivo XML em relação ao XSD gerado pelo CIMTool. Nesse trabalho, entretanto, foi utilizado o aplicativo XML Nanny. A Figura 3.13 apresenta a tela de validação do XML Nanny do documento XML do exemplo anterior.

Figura 3.13. Tela de validação do exemplo de documento XML.



Após a apresentação introdutória do CIM e dos aplicativos necessários para a utilização do modelo, no próximo capítulo será apresentada uma aplicação do CIM num caso teste visando a integração de dois sistemas.

---

**CAPÍTULO 4. APLICAÇÃO DO MODELO CIM**

---

A aplicação do CIM como modelo de dados que viabilize a interoperabilidade entre sistemas heterogêneos exige o mapeamento dos dados de cada um dos sistemas para o CIM. A comunicação entre os sistemas é efetivada mediante a utilização de um barramento de dados, conhecido na literatura como ESB (*Enterprise Service Bus*) (EPRI, 2011, p. 8-1) (USLAR *et al.*, 2011, p. 130), e não diretamente entre os sistemas. O emprego dessa estratégia evita que haja a necessidade de construir ligações dedicadas entre os sistemas comunicantes, tornando a arquitetura de comunicação mais simples e eficiente.

A Figura 4.1 ilustra a ideia que fundamenta o modo de utilização do CIM.

Figura 4.1. CIM e barramento de dados.



#### 4.1. Estudo de caso

O CIM será utilizado, nesse trabalho, para efetuar a modelagem de uma rede de distribuição em média tensão de forma a tornar seus dados próprios para serem integrados com um sistema multiagente num formato adequado à interoperabilidade. Antes de se dedicar aos detalhes do processo de modelagem propriamente dito, é fundamental conhecer as características gerais de ambos os sistemas e entender, particularmente, os tipos de dados que deverão ser trocados.

É importante salientar, previamente, o valor da utilização do modelo CIM para a integração proposta nesse estudo de caso, visto que os sistemas apresentados já haviam sido integrados mediante um modelo de dados proprietário. A recomendação

que é feita no presente trabalho pela utilização do CIM fundamenta-se na conveniência do uso de um modelo padronizado quando se deseja desenvolver um ambiente interoperável. A escolha pelo CIM indica que, além de possibilitarem a troca de informações entre os sistemas integrados (alcançada também por um modelo proprietário), os aplicativos possuem um melhor potencial de serem integrados com outros que venham a ser desenvolvidos. Dessa maneira, há, na utilização do CIM, um avanço de natureza mais preventiva do que corretiva, visto que se evitam os custos de tempo, investimento e tecnologia que ter-se-iam caso não se utilizasse essa norma.

O estudo de caso realizado no presente trabalho reflete uma circunstância típica das REI, na qual o processo de modernização exige um contínuo esforço de integração de aplicativos que precisem ser inseridos num ambiente integrado. Por isso, o uso do CIM representa um meio eficiente de facilitar a troca de dados entre aplicativos, visto que dispensa a utilização de modelos proprietários ao definir um modelo de dados único a ser utilizado na troca de dados entre aplicativos.

#### *4.1.1. Sistema Multiagente de Recomposição Automática*

O processo de modernização dos sistemas de energia elétrica implica numa crescente descentralização, tanto da geração de eletricidade quanto da estrutura de supervisionamento e manipulação de dados. Nesse contexto, os Sistemas Multiagentes (SMA) têm sido cada vez mais empregados na modelagem de aplicativos para redes elétricas inteligentes.

Os SMA são agentes computacionais que possuem as características de reatividade, pró-atividade e sociabilidade (MCARTHUR et al, 2007). Dessa maneira, os agentes ditos inteligentes são programados para atuarem sem a intervenção humana. No caso particular desse trabalho, descrito em Melo (2013), utilizou-se a tecnologia de multiagentes para construir um Sistema Multiagente para Recomposição Automática (SMRA), responsável pela recomposição de trechos desenergizados numa determinada rede de distribuição de energia elétrica.

O SMRA (Sistema Multiagente de Recomposição Automática) é implementado de forma a avaliar a situação da rede elétrica monitorada e atuar, quando necessário, no isolamento de faltas e recomposição de trechos, com o objetivo de minimizar o impacto de possíveis falhas nos sistemas de distribuição. Além disso, a

busca por um método de recomposição cada vez mais rápido torna a escolha pela utilização do SMRA uma opção cada vez mais atraente (SOLANKI et al, 2007).

Para realizar suas funções, o SMRA precisa ter acesso aos parâmetros de tensão, corrente, estado de disjuntores e religadores que compõem a rede elétrica supervisionada. O CIM, nesse contexto, é necessário para disponibilizar as informações necessárias para o funcionamento do SMRA num formato que, além de possibilitar a troca de informações, é padronizado. Ainda que no presente trabalho seja efetuada a modelagem da rede elétrica visando apenas sua ocasional integração com o SMRA, a modelagem com o CIM permite que as informações do sistema modelado sejam compartilhadas numa estrutura padrão, mediante um barramento de dados, com qualquer outro sistema que venha a necessitar dos dados modelados da rede.

Outra vantagem do emprego do CIM para o SMRA é a forma conveniente como os dados de topologia de trechos são estruturados no documento XML. Como será mencionado posteriormente, o CIM possui classes que são extremamente úteis quando se deseja representar não só a topologia estática de redes elétricas, como também a estrutura dinâmica, típica de redes submetidas a faltas e recomposições.

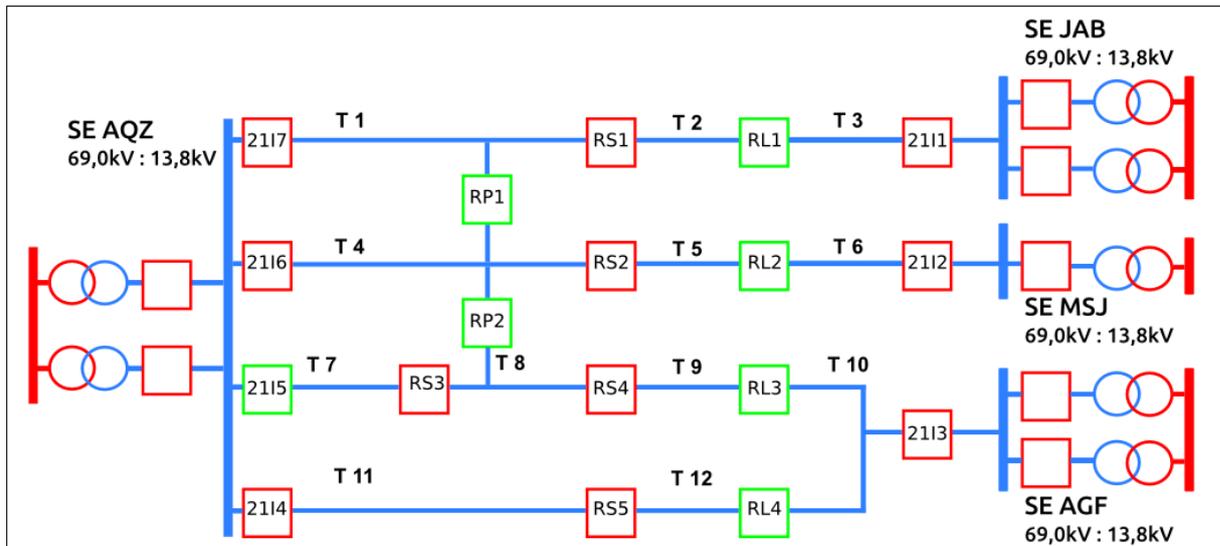
Para o desenvolvimento do multiagente, foi utilizada a plataforma JADE (*Java Agent Development Framework*), que possui seus protocolos de ação elaborados pela FIPA (*Foundation for Intelligent Physical Agents*). Por ser um aplicativo livre e basear-se numa linguagem que é frequentemente utilizada para a construção de aplicativos, esse exemplo enquadra-se bem na filosofia do presente trabalho de buscar desenvolver a integração e a interoperabilidade em ambientes os mais genéricos possíveis.

#### *4.1.2. Simulador da Rede Elétrica e Recomposição Automática*

Um simulador de uma rede elétrica foi desenvolvido em Melo (2013), designado Simulador de Falta e Recomposição Automática (SFRA), capaz de representar condições de faltas elétricas, e a partir do SMRA medidas são tomadas visando minimizar os trechos interrompidos pela falta elétrica. A rede elétrica em 13,8 kV, 60 Hz, que correspondia à antiga rede de distribuição da cidade de Aquiraz/CE, é suprida a partir de quatro subestações (SE AQZ, SE JAB, SE MSJ e SE AGF) em 69/13,8 kV de tensão nominal. A rede apresenta sete religadores de

saída de alimentadores (2111, 2112, 2113, 2114, 2115, 2116 e 2117), seis religadores normalmente abertos (RL1, RL2, RL3, RL4, RP1 e RP2) e doze trechos de alimentadores (T1 a T12). A Figura 4.2 apresenta a topologia da rede modelada no simulador.

Figura 4.2. SFRA com topologia de uma rede de distribuição em média tensão.



As características, medidas e limites reais da rede-teste foram consideradas no desenvolvimento do sistema multiagente responsável pela recomposição dos trechos.

Nas próximas subseções, será descrito o processo de modelagem da rede, em termos de seus dados e informações, com o uso do CIM. Em seguida, uma interface *web* é desenvolvida para facilitar a escrita e a leitura dos dados inseridos na rede.

#### 4.1.3. Definição e mapeamento dos dados

A definição dos dados que se deseja transferir entre os sistemas deve fazer parte de um levantamento elaborado pelo(s) operador(es) responsável(is) pelo controle e funcionamento de cada um dos sistemas. É fundamental compreender, previamente à modelagem, as características gerais dos aplicativos que estão sendo integrados. O CIM apresenta várias maneiras de organizar as informações de forma a tornar sua representação mais útil e conveniente. Por isso, torna-se indispensável não só a definição de quais dados devem ser modelados, mas também a compreensão da natureza de cada dado.

De posse desses dados, efetuou-se o mapeamento de cada um deles utilizando os diagramas de classes que compõem o CIM. Para tornar mais eficiente o acesso às informações, decidiu-se dividir os dados de acordo com dois critérios: a região funcional onde a informação era necessária e se o arquivo seria manipulado para escrita/leitura ou somente para leitura. Dessa maneira, obter-se-á, ao final do processo, quatro documentos distintos:

- a) 1 para os dados das quatro subestações (leitura/escrita);
- b) 1 para os dados das chaves do centro da rede (leitura/escrita);
- c) 1 para a topologia do centro da rede (leitura);
- d) 1 para os dados dos trechos (leitura/escrita).

Nas próximas subseções, será descrito o processo de elaboração de cada um desses arquivos utilizando o CIM como modelo de dados.

#### 4.1.3.1. Modelagem das subestações

Os elementos básicos que compõem as quatro subestações são barramentos, transformadores e disjuntores. A Tabela 4.1 apresenta os dados que precisam ser modelados para representar as subestações:

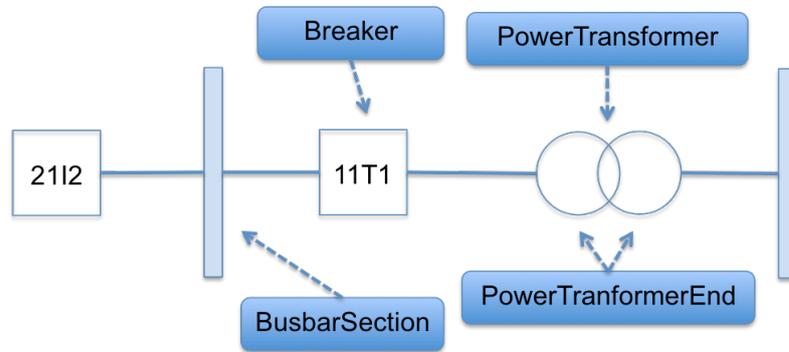
Tabela 4.1. Parâmetros das subestações.

Subestações	
Elemento	Parâmetro
Disjuntor	<ul style="list-style-type: none"> <li>• Estado</li> </ul>
Transformador	<ul style="list-style-type: none"> <li>• Tensão nominal nos enrolamentos</li> <li>• Corrente</li> </ul>
Barramento	<ul style="list-style-type: none"> <li>• Tensão nominal</li> </ul>

Visto que a estrutura das quatro subestações são bastante similares, será descrito apenas o mapeamento da subestação de Messejana, SE MSJ. O processo é análogo para todas as outras unidades, alterando, unicamente, a quantidade dos

elementos que as compõem. Apresenta-se, na Figura 4.3, o mapeamento de cada um dos elementos presentes nessa subestação.

Figura 4.3. Mapeamento dos elementos da subestação de Messejana.



Cada um dos elementos possui um atributo, “name” ou “mRID” (*master Resource Identifier*), para identificá-lo. Para representar a medida de corrente que flui pelo transformador, foi adicionado o elemento “Terminal” ao qual associa-se um elemento de medida de corrente.

Após a inserção dos dados no CIMTool, o XSD resultante orienta a construção do XML com os dados prontos a serem manipulados e compartilhados. A Figura 4.4 apresenta a parte do XML das subestações com as informações referentes à subestação de Messejana.

Figura 4.4. Parte do arquivo XML das subestações.

```
<Substation>
  <name>MSJ</name>
  <Breaker>
    <mRID>11T1</mRID>
    <normalOpen>0</normalOpen>
  </Breaker>
  <Breaker>
    <mRID>2112</mRID>
    <normalOpen>0</normalOpen>
  </Breaker>
  <BusbarSection>
    <mRID>B5</mRID>
    <BaseVoltage>
      <nominalVoltage>
        <multiplier>k</multiplier>
        <unit>V</unit>
        <value>0</value>
      </nominalVoltage>
    </BaseVoltage>
  </BusbarSection>
```

```

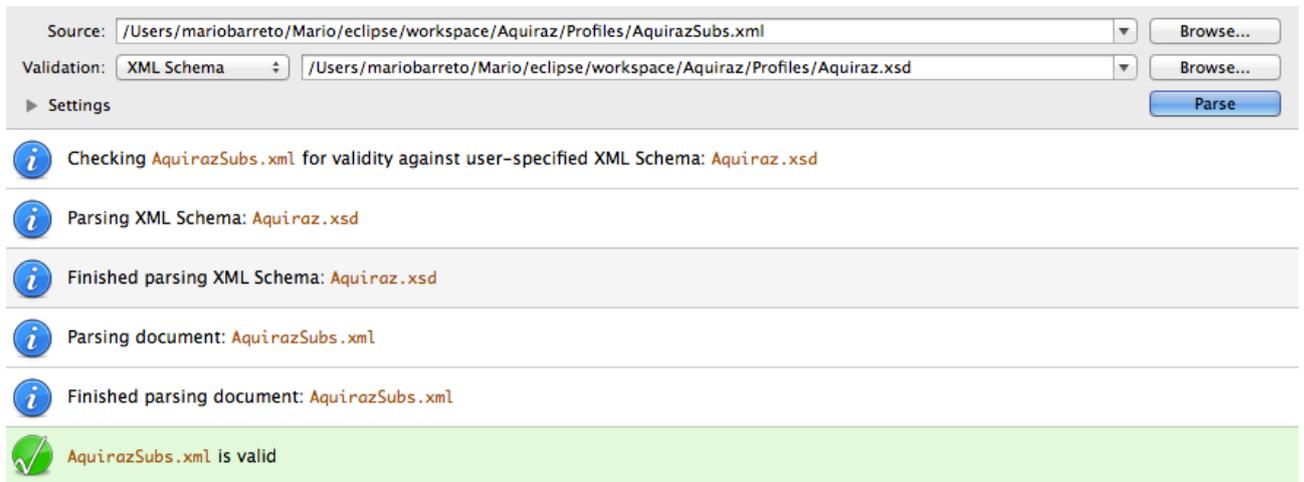
<BusbarSection>
  <mRID>B6</mRID>
  <BaseVoltage>
    <nominalVoltage>
      <multiplier>k</multiplier>
      <unit>V</unit>
      <value>0</value>
    </nominalVoltage>
  </BaseVoltage>
</BusbarSection>
<PowerTransformer>
  <mRID>02T1</mRID>
  <PowerTransformerEnd>
    <endNumber>1</endNumber>
    <BaseVoltage>
      <nominalVoltage>
        <multiplier>k</multiplier>
        <unit>V</unit>
        <value>0</value>
      </nominalVoltage>
    </BaseVoltage>
  </PowerTransformerEnd>
  <PowerTransformerEnd>
    <endNumber>2</endNumber>
    <BaseVoltage>
      <nominalVoltage>
        <multiplier>k</multiplier>
        <unit>V</unit>
        <value>0</value>
      </nominalVoltage>
    </BaseVoltage>
  </PowerTransformerEnd>
  <Terminals>
    <Measurements>
      <unitSymbol>A</unitSymbol>
      <AnalogValues>
        <value>0</value>
      </AnalogValues>
    </Measurements>
  </Terminals>
</PowerTransformer>
</Substation>

```

Informações sobre a configuração estrutural e topológica das subestações não foram utilizadas pelo multiagente construído em Melo (2013). Por isso, não foi necessário modelar a topologia dessa parte da rede.

A Figura 4.5 apresenta a tela do XML Nanny com a validação do arquivo “AquirazSubs.xml” com o XSD gerado pelo CIMTool, “Aquiraz.xsd”.

Figura 4.5. Validação do documento XML com dados das subestações.



#### 4.1.3.2. Modelagem das chaves de centro

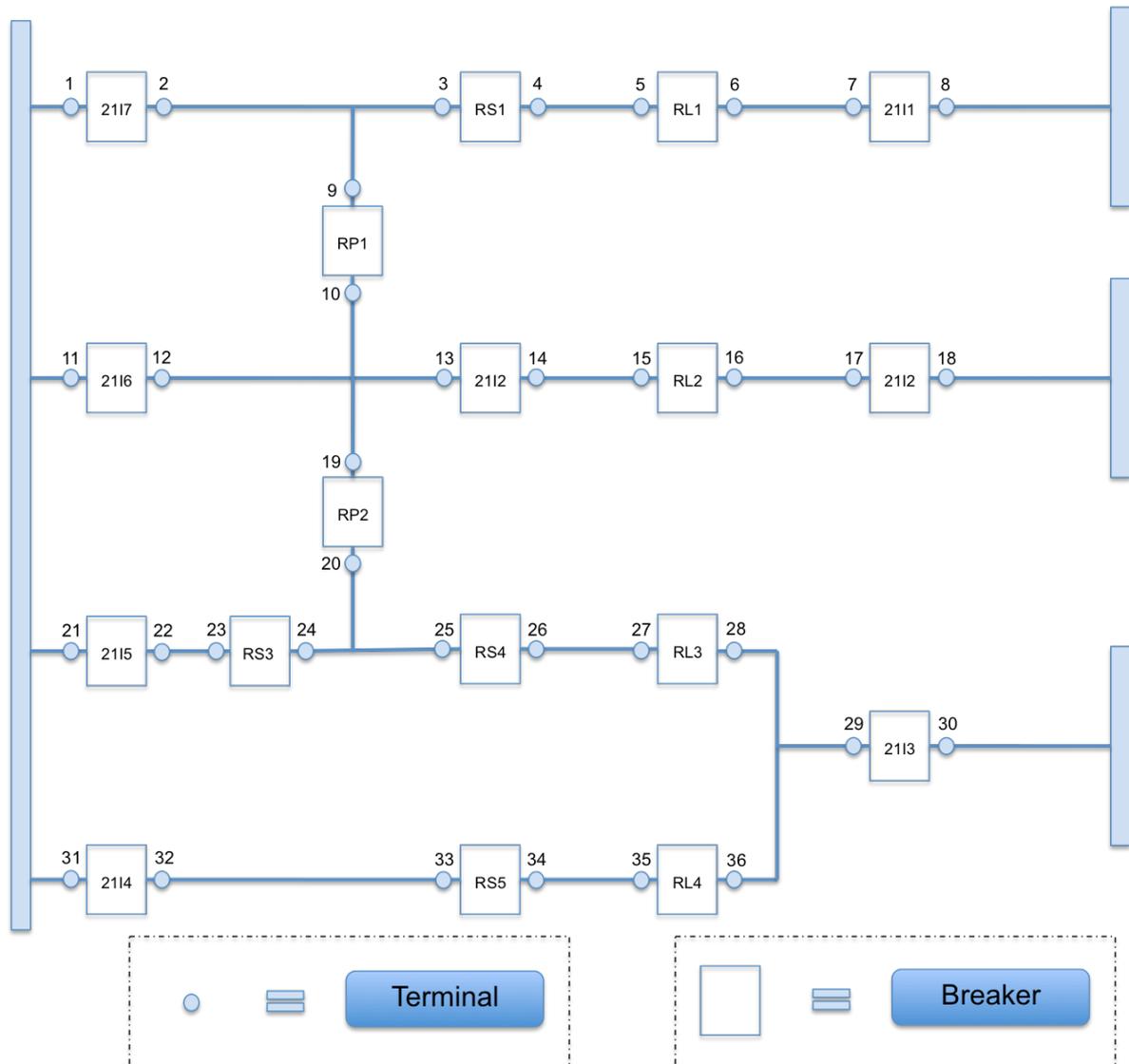
Os elementos que compõem a parte central da rede Aquiraz são disjuntores e religadores. A Tabela 4.2 apresenta os dados que precisam ser modelados para as chaves presentes no centro da rede elétrica.

Tabela 4.2. Parâmetros das chaves.

Centro	
Elemento	Parâmetro
Disjuntor	<ul style="list-style-type: none"> <li>Estado</li> <li>Corrente em cada terminal</li> <li>Tensão em cada terminal</li> </ul>
Religador	<ul style="list-style-type: none"> <li>Estado</li> <li>Corrente em cada terminal</li> <li>Tensão em cada terminal</li> </ul>

A despeito das diferenças operacionais que há entre disjuntores e religadores, no CIM, todavia, ambos são modelados como o mesmo elemento, “Breaker”. A Figura 4.6 apresenta o mapeamento dos elementos dessa parte da rede Aquiraz.

Figura 4.6. Mapeamento das chaves da rede elétrica.



As informações concernentes aos relgadores “2111”, “2112”, “2113”, “2114”, “2115”, “2116”, “2117”, adjacentes aos barramentos, apesar de fazerem parte da estrutura física das subestações, foram incluídas no documento da parte central da rede Aquiraz, por serem consideradas importantes para as manobras de recomposição efetuadas nessa região da rede. Os dados dos barramentos, por outro lado, constam somente no documento com as informações sobre as subestações, e estão representados na figura acima apenas para fins ilustrativos.

A Figura 4.7 apresenta a parte do arquivo XML dos dados dos relgadores que contém as informações referentes ao relgador RS1.

Figura 4.7. XML com dados do religador RS1.

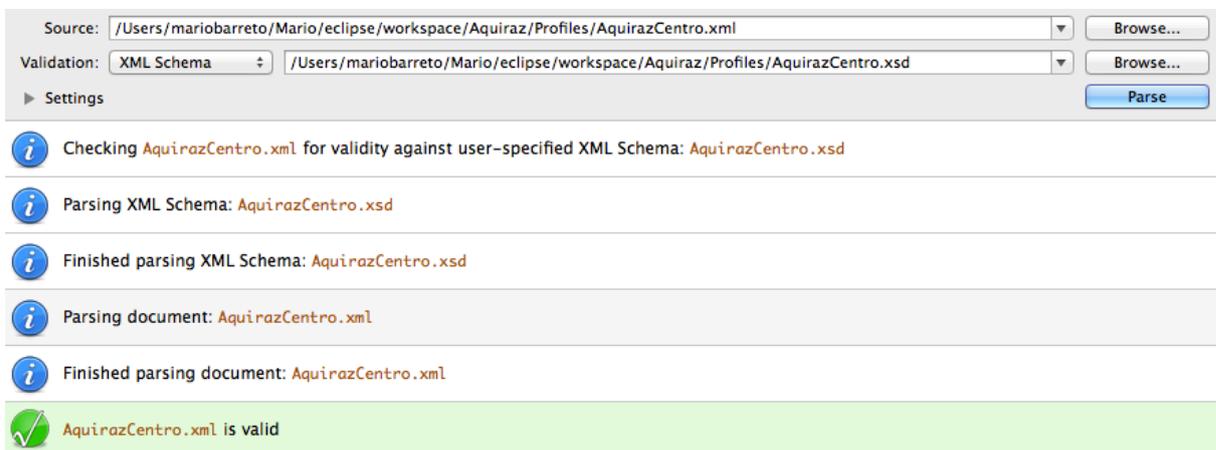
```

<Breaker>
  <mRID>RS1</mRID>
  <normalOpen>0</normalOpen>
  <Terminals>
    <sequenceNumber>3</sequenceNumber>
    <Measurements>
      <unitSymbol>A</unitSymbol>
      <AnalogValues>
        <value>0</value>
      </AnalogValues>
    </Measurements>
    <Measurements>
      <unitSymbol>V</unitSymbol>
      <AnalogValues>
        <value>0</value>
      </AnalogValues>
    </Measurements>
  </Terminals>
  <Terminals>
    <sequenceNumber>4</sequenceNumber>
    <Measurements>
      <unitSymbol>A</unitSymbol>
      <AnalogValues>
        <value>0</value>
      </AnalogValues>
    </Measurements>
    <Measurements>
      <unitSymbol>V</unitSymbol>
      <AnalogValues>
        <value>0</value>
      </AnalogValues>
    </Measurements>
  </Terminals>
</Breaker>

```

A Figura 4.8 apresenta a tela de validação do documento “AquirazCentro.xml” com o XSD gerado pelo CIMTool “AquirazCentro.xsd”.

Figura 4.8. Validação do documento XML com dados do centro.

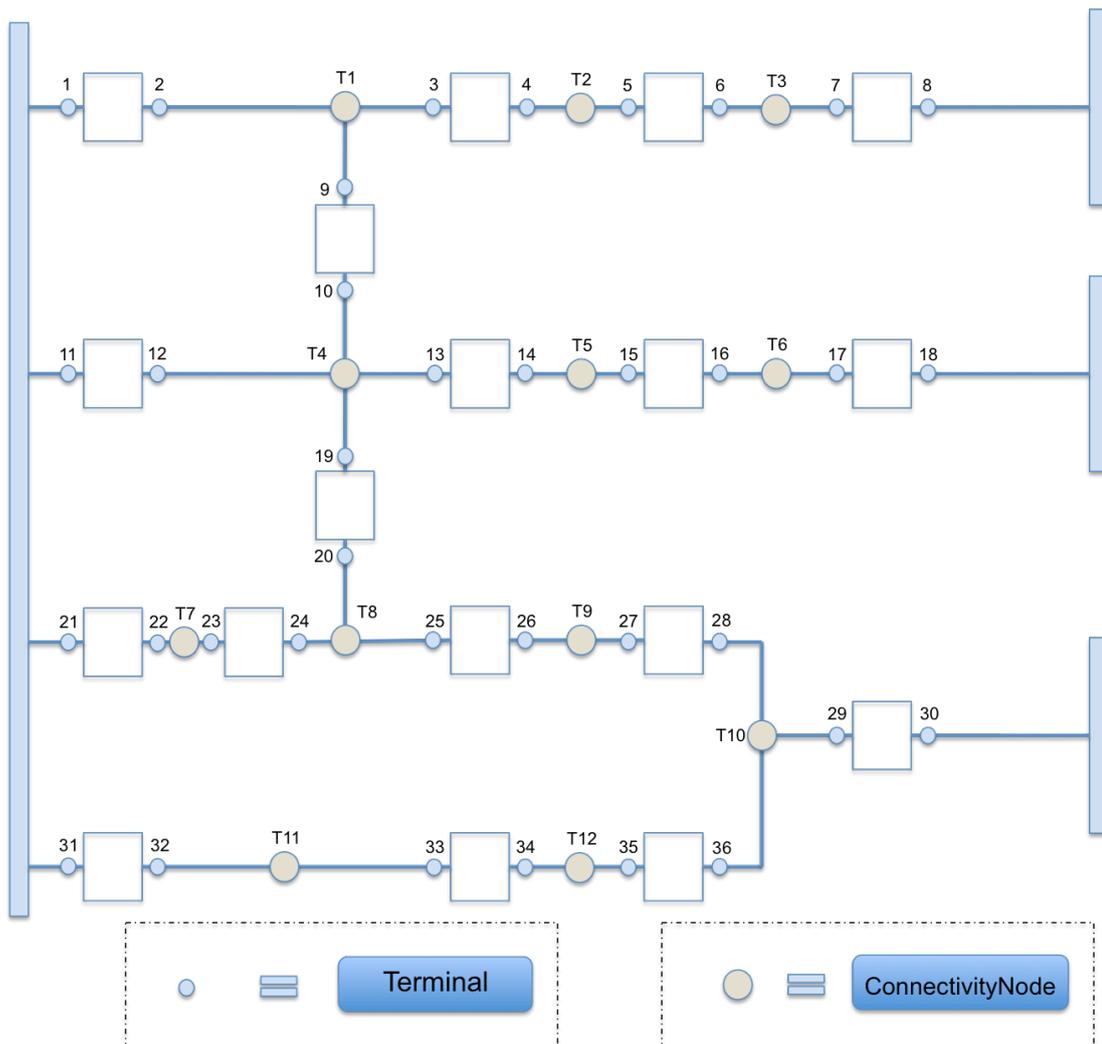


#### 4.1.3.3. Modelagem da topologia da rede em 13,8 kV

Os elementos necessários para representar a topologia da rede em 13,8 kV são “ConnectivityNode” e “Terminal”. O arquivo XML gerado a partir da modelagem da topologia da parte central da rede é o único, dentre os quatro, que não sofrerá nenhum tipo de alteração para seu uso, visto que a maneira como os religadores estão ligados entre si, independe das manobras necessárias para eventuais recomposições. No entanto, a despeito de sua natureza estática, os dados de conectividade contidos nesse arquivo são fundamentais para definir a estrutura básica da rede e, conseqüentemente, definir os limites operacionais que o SMRA deve obedecer em sua atuação.

A Figura 4.9 apresenta a modelagem da topologia dos religadores que compõem a parte central da rede teste.

Figura 4.9. Mapeamento da topologia da rede-teste.



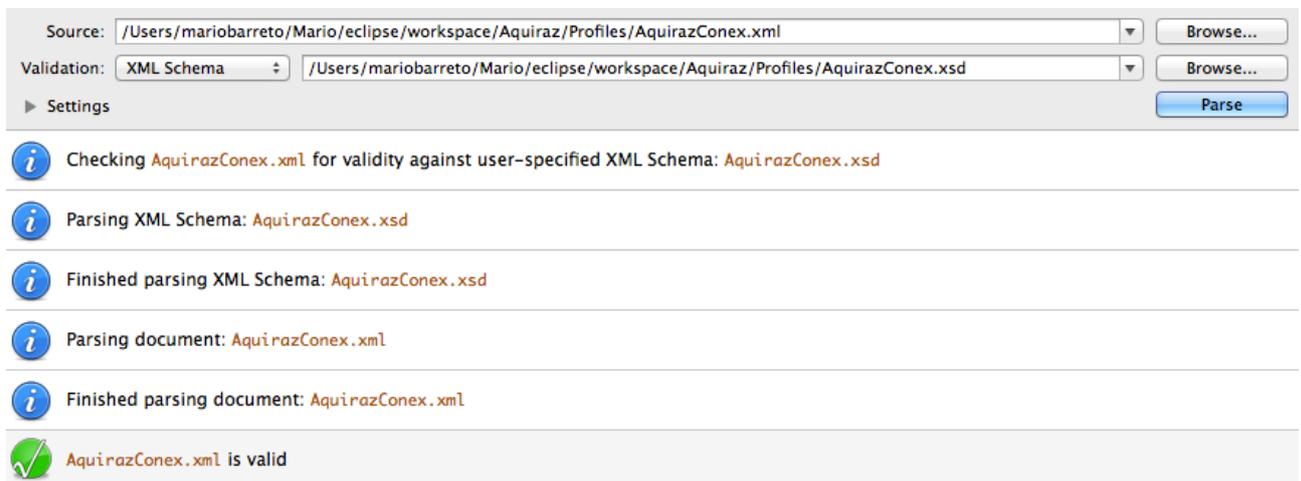
Os nós conectivos recebem a mesma nomenclatura dos trechos aos quais pertencem para facilitar sua identificação. Uma parte do documento da topologia das chaves, contendo a configuração dos nós T7 e T8, é apresentada na Figura 4.10.

Figura 4.10. XML com parte da topologia da rede-teste.

```
<ConnectivityNode>
  <name>T7</name>
  <Terminals>
    <sequenceNumber>22</sequenceNumber>
  </Terminals>
  <Terminals>
    <sequenceNumber>23</sequenceNumber>
  </Terminals>
</ConnectivityNode>
<ConnectivityNode>
  <name>T8</name>
  <Terminals>
    <sequenceNumber>20</sequenceNumber>
  </Terminals>
  <Terminals>
    <sequenceNumber>24</sequenceNumber>
  </Terminals>
  <Terminals>
    <sequenceNumber>25</sequenceNumber>
  </Terminals>
</ConnectivityNode>
```

A Figura 4.11 apresenta a tela do XML Nanny com a validação do documento “AquirazConex.xml” com o XSD gerado pelo CIMTool “AquirazConex.xsd”.

Figura 4.11. Validação do documento XML com dados da topologia do centro.



#### 4.1.3.4. Modelagem dos trechos

Os elementos que compõem os trechos são alimentadores em 13,8 kV. Além das informações a respeito das características físicas dos cabos, como capacidade e comprimento, necessita-se transmitir os dados sobre a topologia dos trechos. Na Tabela 4.3 representam-se as informações que precisam estar contidas no arquivo XML, resultante da modelagem dos trechos.

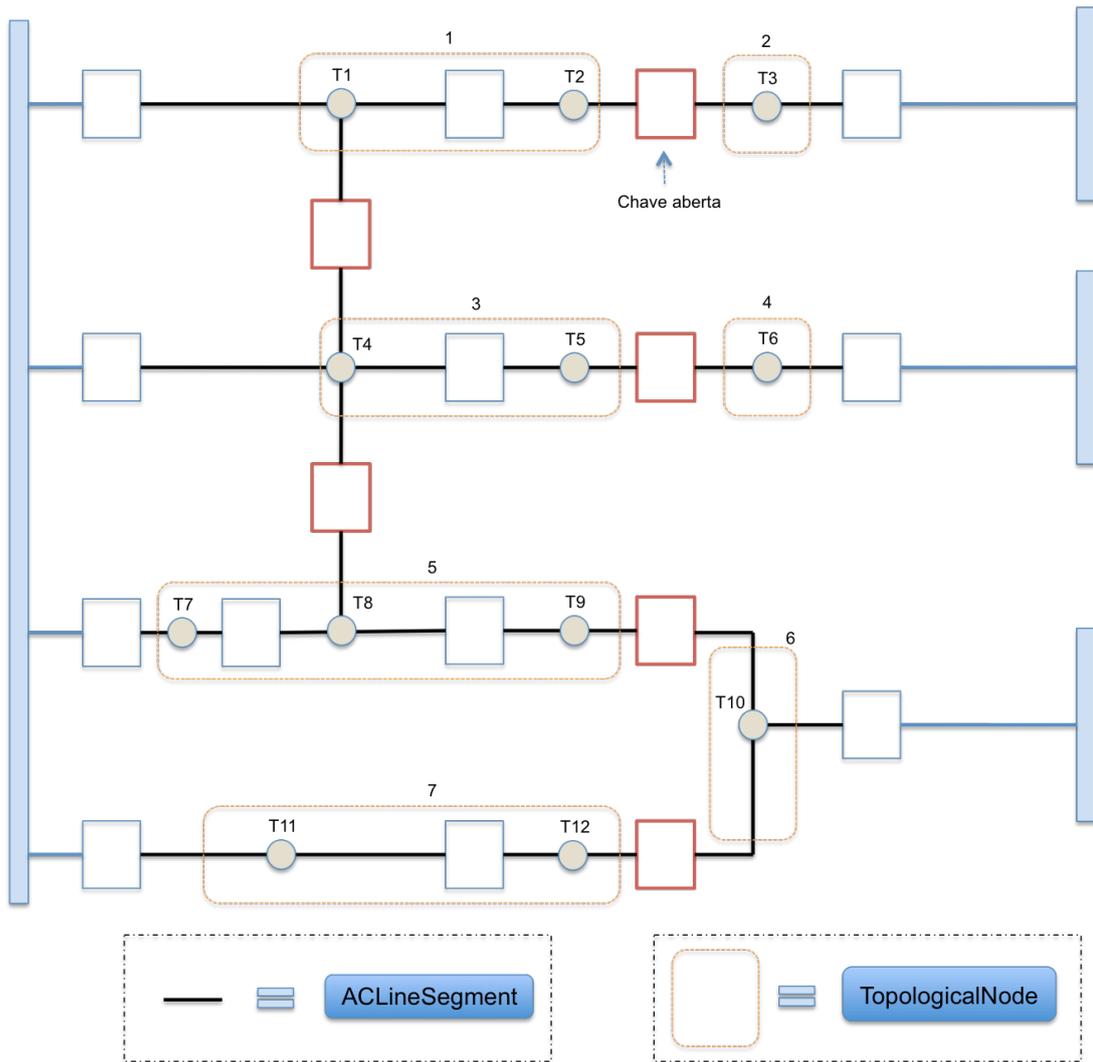
Tabela 4.3. Parâmetros dos trechos.

Trechos	
Elemento	Parâmetro
Trecho	<ul style="list-style-type: none"> <li>• Comprimento</li> <li>• Impedância</li> <li>• Topologia</li> </ul>

A conectividade entre os trechos, diferentemente da topologia dos religadores, representada na subseção anterior, altera-se de acordo com os estados dos disjuntores e religadores. Para transmitir essa informação, lançou-se mão da classe “TopologicalNode” do CIM. Trata-se de uma classe de natureza abstrata que relaciona os nós conectivos que estão, em um momento específico, eletricamente conectados. A Figura 4.12 representa a modelagem dos trechos.

Os nós topológicos relacionam, como dito anteriormente, os nós conectivos que estão eletricamente conectados. Visto que cada trecho possui um único nó conectivo, essa informação é útil para conhecer a maneira como cada trecho está, momentaneamente, conectado aos outros. Se, por exemplo, o religador entre os nós topológicos 1 e 2 vier a conduzir, altera-se a configuração topológica, passando a existir, ao invés de dois, um único nó topológico contendo os nós conectivos T1, T2 e T3. Recorda-se, todavia, que a lógica de manipulação dos nós topológicos é definida pelo sistema multiagente, ficando a cargo do CIM apenas a forma como a informação é representada.

Figura 4.12. Modelagem dos trechos.



A Figura 4.13 apresenta parte do XML com informações sobre as especificações físicas e a conectividade dos trechos.

Figura 4.13. XML com parte das informações dos trechos.

```

<ACLineSegment>
  <name>T1</name>
  <length>
    <unit>m</unit>
    <value>0</value>
  </length>
  <PerLengthImpedance>
    <r>
      <denominatorUnit>m</denominatorUnit>
      <unit>ohm</unit>
      <value>0</value>
    </r>
  </PerLengthImpedance>
</ACLineSegment>

```

```

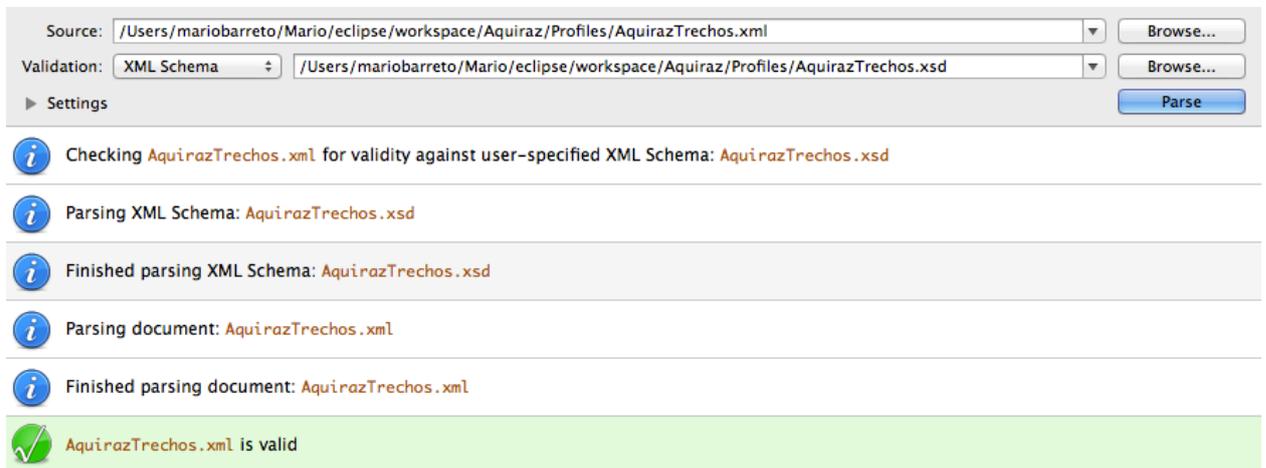
<x>
  <denominatorUnit>m</denominatorUnit>
  <unit>ohm</unit>
  <value>0</value>
</x>
</PerLengthImpedance>
</ACLineSegment>

<TopologicalNode>
  <mRID>1</mRID>
  <ConnectivityNodes>
    <name>T1</name>
  </ConnectivityNodes>
  <ConnectivityNodes>
    <name>T2</name>
  </ConnectivityNodes>
</TopologicalNode>

```

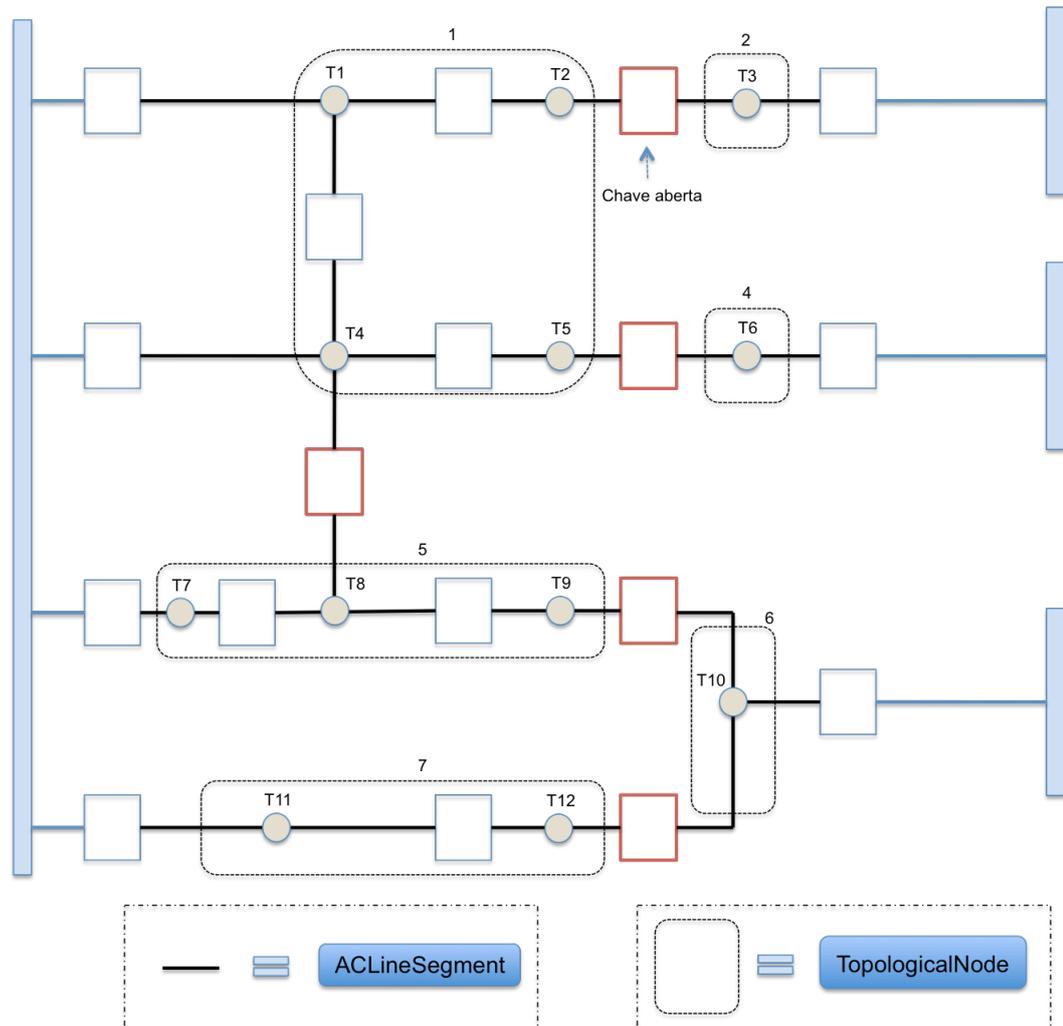
A Figura 4.14 apresenta a tela do XML Nanny com a validação do documento “AquirazTrechos.xml” com o documento XSD gerado pelo CIMTool “AquirazTrechos.xsd”.

Figura 4.14. Validação do documento XML com dados dos trechos.



No caso de o religador entre os nós T1 e T4 vier a conduzir, a nova topologia da rede é representada na Figura 4.15.

Figura 4.15. Nós topológicos com mudança no estado de chave.



Observa-se que os nós topológicos 1 e 3 fundiram-se, formando um único nó topológico 1 (a nomenclatura do nó resultante é livre e deve ser a que mais convier ao entendimento do modelo). O trecho do documento XML com as informações sobre a nova topologia é apresentada na Figura 4.16.

Figura 4.16. Parte do documento XML dos trechos após mudança.

```
<TopologicalNode>
  <mRID>1</mRID>
  <ConnectivityNodes>
    <name>T1</name>
  </ConnectivityNodes>
  <ConnectivityNodes>
    <name>T2</name>
  </ConnectivityNodes>
```

```

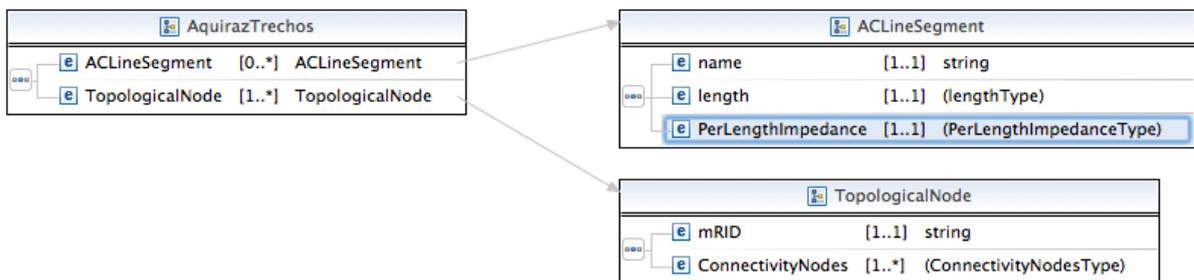
<ConnectivityNodes>
  <name>T4</name>
</ConnectivityNodes>
<ConnectivityNodes>
  <name>T5</name>
</ConnectivityNodes>
</TopologicalNode>

<TopologicalNode>
  <mRID>2</mRID>
  <ConnectivityNodes>
    <name>T3</name>
  </ConnectivityNodes>
</TopologicalNode>

```

Visto que tal mudança no estado das chaves pode ocorrer com uma certa frequência em decorrência das manobras efetuadas pelo sistema de recomposição, é interessante, no momento da modelagem, criar um XSD que seja flexível o suficiente para permitir que o documento XML aceite esse tipo de alteração. Dessa maneira, nesse caso em particular, o mesmo *schema* é utilizado para validar os dados contidos no XML mesmo após alteração em sua estrutura. A Figura 4.17 apresenta a maneira como o XSD dos trechos foi construído, na qual se observa que a multiplicidade dos *ConnectivityNodes* em cada *TopologicalNode* é de [1..\*], ou seja, cada nó topológico pode possuir no mínimo 1 nó conectivo, sem limite máximo.

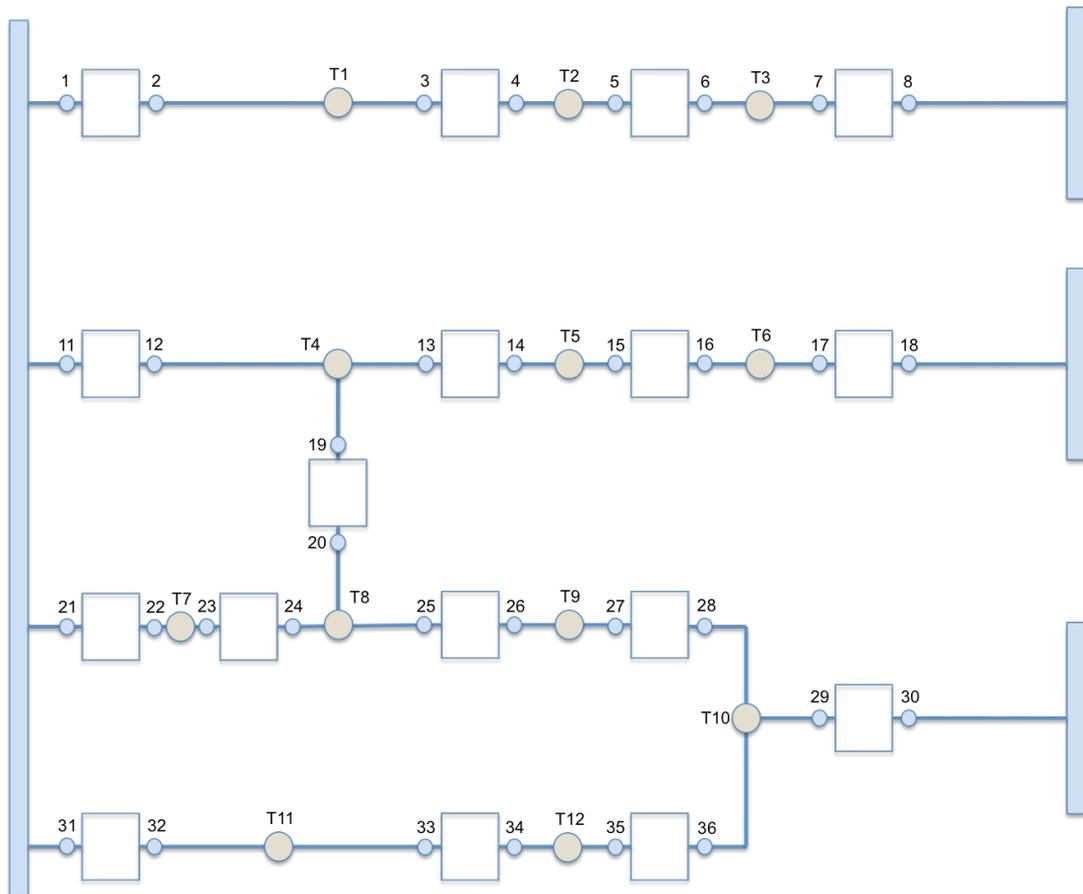
Figura 4.17. Estrutura do XSD dos trechos.



#### 4.1.4. Mudança na topologia

Suponha-se que a chave RP1, que conecta o trecho T1 ao trecho T4 na topologia original da rede, seja removida da rede elétrica. A Figura 4.18 apresenta a topologia da rede após essa mudança.

Figura 4.18. Topologia após retirada de chave.



No entanto, antes de se demonstrar trechos dos documentos com a nova configuração da rede, é fundamental ter-se em mente que o CIM não apresenta qualquer tipo de algoritmo que defina a maneira como o arquivo XML deve ser manipulado quando há adição ou exclusão de elementos na rede elétrica. O CIM determina, unicamente, o formato como a informação é estruturada no interior de cada arquivo. Dessa maneira, os sistemas responsáveis pela manipulação do documento XML devem ser configurados para efetuar a mudança e, no caso de inserção de novos elementos, inserir as novas linhas de informação no formato estipulado pelo CIM.

No caso da alteração sugerida acima, com exceção do arquivo XML contendo as informações sobre as subestações, os três documentos (topologia do centro, medidas/estados das chaves do centro e nós topológicos e trechos) gerados a partir da modelagem da rede precisam ser alterados.

Trecho do arquivo XML com a topologia das chaves do centro após a retirada da chave é apresentado na Figura 4.19, no qual se observa que os nós conectivos T1 e T4 tiveram suas configurações alteradas.

Figura 4.19. XML com topologia após retirada da chave RP1.

```

<ConnectivityNode>
  <name>T1</name>
  <Terminals>
    <sequenceNumber>2</sequenceNumber>
  </Terminals>
  <Terminals>
    <sequenceNumber>3</sequenceNumber>
  </Terminals>
</ConnectivityNode>

<ConnectivityNode>
  <name>T4</name>
  <Terminals>
    <sequenceNumber>12</sequenceNumber>
  </Terminals>
  <Terminals>
    <sequenceNumber>13</sequenceNumber>
  </Terminals>
  <Terminals>
    <sequenceNumber>19</sequenceNumber>
  </Terminals>
</ConnectivityNode>

```

A alteração efetuada no arquivo com os dados de medidas e estados das chaves do centro é de exclusão das linhas com as informações da chave RP1 retirada. Dessa forma, como se trata de uma exclusão que não altera o restante do arquivo, não será apresentado trecho do documento resultante após alteração.

Finalmente, o arquivo que contém os nós topológicos e os dados dos trechos pode ou não sofrer alteração. Se a chave RP1 estiver aberta no momento de sua retirada, os nós conectivos T1 e T2 não pertenciam a um mesmo nó topológico, assim como demonstrado na Figura 4.12. Nessa circunstância, não haverá nenhuma modificação estrutural no documento. Diferentemente, caso a chave RP1 esteja fechada (conduzindo), os nós conectivos T1, T2, T4 e T5 pertencem ao mesmo nó topológico, como é ilustrado na Figura 4.15. Assim, se a chave é retirada nessa circunstância, haverá uma alteração na estrutura do arquivo XML que descreve os nós topológicos. Um trecho do documento XML dos nós topológicos e trechos é apresentado na Figura 4.20.

Figura 4.20. XML com nós topológicos após retirada da chave RP1.

```
<TopologicalNode>
  <mRID>1</mRID>
  <ConnectivityNodes>
    <name>T1</name>
  </ConnectivityNodes>
  <ConnectivityNodes>
    <name>T2</name>
  </ConnectivityNodes>
</TopologicalNode>

<TopologicalNode>
  <mRID>3</mRID>
  <ConnectivityNodes>
    <name>T4</name>
  </ConnectivityNodes>
  <ConnectivityNodes>
    <name>T5</name>
  </ConnectivityNodes>
</TopologicalNode>
```

Um aspecto fundamental a ser considerado no caso de mudança da estrutura dos arquivos XML contendo as informações da rede é a maneira de validá-los, visto que o esquema foi gerado originalmente para validar os documentos com a estrutura anterior. Como já foi mencionado, o XSD pode ser construído de maneira a permitir determinado grau de flexibilidade quanto às mudanças que o seu respectivo arquivo XML pode ser submetido. Por outro lado, se o esquema deixa o documento muito “livre” ao permitir muitas alterações, o XSD tem seu objetivo principal (verificar que o arquivo XML encontra-se num formato adequado) comprometido. Dessa maneira, é importante elaborar um XSD que concilie flexibilidade e segurança na medida que se achar conveniente.

Como regra geral, recomenda-se que o XSD seja flexível apenas na medida de validar mudanças consideradas comuns ou frequentes. Como exemplo desse tipo de mudança, menciona-se a configuração dos nós topológicos. Nesse caso, o esquema que valida o arquivo XML contendo tais informações deveria ser construído de maneira a permitir tais mudanças, assim como foi explicado na seção 4.1.3.4. Dessa maneira, o XSD é rígido o suficiente para garantir que o arquivo XML está no formato correto e, ao mesmo tempo, possui a flexibilidade necessária para permitir a mudança de nós topológicos.

Para mudanças não-programadas, como o exemplo da retirada da chave RP1, sugere-se que seja elaborado um novo XSD contendo a nova estrutura que o XML deve apresentar. Se o XSD inicial permitisse esse tipo de mudança no XML, a segurança e fidedignidade das informações estaria comprometida. Por isso, o mais indicado é gerar um outro XSD com as novas especificações por meio do CIMTool ou outro programa análogo, que torne a elaboração do XSD simples e rápida.

#### *4.1.5. Interface web*

Finalizado o processo de modelagem da rede elétrica do caso teste utilizando o modelo de dados CIM, tem-se quatro arquivos XML prontos para serem utilizados para a troca de informações entre o SMRA e o simulador da rede elétrica.

Mediante a conexão estabelecida entre os dois sistemas, o simulador é programado para manipular os arquivos de dados que contêm informações sobre os parâmetros da rede no caso de falta em algum trecho. O SMRA, por sua vez, recebe os dados, identifica e decide quais são as manobras necessárias para a recomposição da rede, e envia uma resposta ao simulador para que este efetue as mudanças nos estados das chaves orientadas pelo SMRA.

No entanto, a leitura e a alteração dos arquivos XML são processos fundamentais para efetuar o controle e a supervisão do fluxo de dados entre os sistemas. Visto que a leitura e alteração manual de arquivos XML é um processo temporalmente dispendioso e que pode produzir frequentes erros, decidiu-se pelo desenvolvimento de um mecanismo que facilitasse a realização desses procedimentos.

##### *4.1.5.1. Formulário PHP*

Para criar uma ferramenta que fosse independente de plataforma e pudesse ser utilizada tanto com conexão à internet como *off-line*, concluiu-se pela construção de um aplicativo que pudesse ser acessado mediante qualquer navegador comum utilizado para acesso à internet. Dessa forma, o aplicativo serviria como interface por meio da qual o usuário responsável pelo monitoramento da rede teria seu acesso aos dados facilitado.

Inicialmente, ao serem consideradas as possibilidades de linguagem a ser empregada na construção da interface, foram levados em consideração os seguintes critérios:

- a) Adequabilidade ao ambiente *web*;
- b) Possibilidade de manipulação de arquivos XML;
- c) Integração com banco de dados;
- d) Simplicidade na construção de formulários.

A despeito de não ter havido um levantamento comparativo entre as funcionalidades das linguagens que poderiam ter sido utilizadas, a linguagem PHP atende aos objetivos que se buscava. Por ter sua origem, em 1994, motivada pela intenção de construir páginas dinâmicas acessadas pela internet (DALL'OGGIO, 2009), o PHP adequa-se muito bem tanto à criação de formulários quanto ao ambiente *web*, podendo seu código ser inserido em outros próprios para o design de *sites* como HTML e CSS.

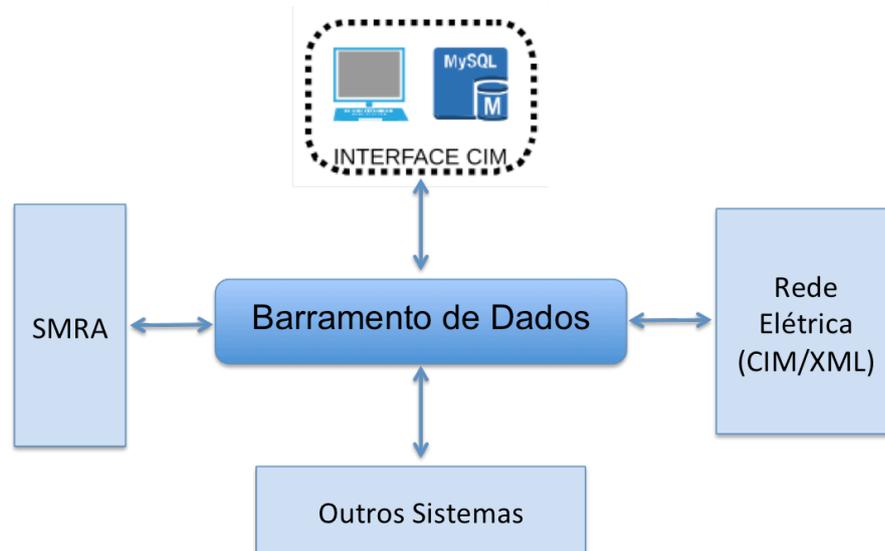
Outra vantagem importante na utilização do PHP é ele já possuir funções especializadas para o acesso ao banco de dados MySQL. O MySQL é um servidor e gerenciador de banco de dados relacional, de licença dupla (sendo uma delas de software livre), projetado inicialmente para ser utilizado com aplicações de pequeno e médio porte, mas que hoje já atende a aplicações de grande porte. Além disso, o MySQL possui todas as características que um banco de dados de grande porte precisa, sendo reconhecido como o banco de dados *open source* com maior capacidade para concorrer com programas similares de código fechado, tais como Oracle e SQL Server (MILANI, 2006).

Finalmente, o PHP também possibilita a manipulação de documento XML via funções especializadas e pré-definidas. O arquivo XML não é tratado pelo PHP como um simples arquivo de texto, havendo funções que permitem a leitura e a escrita de dados com acesso direto aos espaços dos elementos e atributos.

Com todas essas características em evidência, e tendo em perspectiva os procedimentos explicados anteriormente, a ideia fundamental da interface *web* é configurá-la para que ela permita ao usuário a manipulação das informações contidas nos dados dos arquivos XML no formato CIM, elaborados de acordo com o XSD gerado pelo CIMTool que, por sua vez, foi construído tendo como base a modelagem feita sobre os dados levantados. Os sistemas que estão sendo

integrados, SMRA e rede elétrica, utilizam-se desses aparatos para incrementar sua interoperabilidade. A Figura 4.21 apresenta uma ilustração desse cenário.

Figura 4.21. Lógica de funcionamento do SMRA, rede elétrica e interface web.



#### 4.1.5.2. Banco de dados MySQL

Um dos aspectos fundamentais da utilização da interface é a possibilidade de armazenar as informações num banco de dados. A análise do comportamento de cada parâmetro no decorrer de um determinado intervalo de tempo pode ser extremamente útil para o diagnóstico de falhas crônicas num sistema de energia.

Para a utilização do banco de dados, é preciso haver um planejamento de como as variáveis devem ser armazenadas. Assim como em outros bancos, o MySQL trabalha com o conceito de tabelas para armazenar as informações. Cada tabela possui um determinado número de linhas e colunas, em que cada célula representa uma parâmetro. A Tabela 4.4 apresenta como os parâmetros de cada tipo de elemento da rede teste foram estruturados. A quantidade de parâmetros varia de acordo com a região da rede.

Tabela 4.4. Tabelas e parâmetros da rede-teste.

Banco de Dados	
Tabela	Parâmetro
Subestação	<ul style="list-style-type: none"> <li>• Estado Chave: booleano</li> <li>• Tensão Barramento: float</li> <li>• Tensão Transformador: float</li> <li>• Corrente Transformador: float</li> </ul>
Chave Centro	<ul style="list-style-type: none"> <li>• Estado Chave: booleano</li> <li>• Tensão nos terminais: float</li> <li>• Corrente nos terminais: float</li> </ul>
Trecho	<ul style="list-style-type: none"> <li>• Comprimento do cabo: float</li> <li>• Resistência: float</li> <li>• Reatância: float</li> </ul>

Tem-se, ao final de todo o processo, uma interface utilizada por meio de um *browser* (Firefox, Google Chrome, etc.) que, em conjunto com o banco de dados construído, permite a intervenção, tanto para leitura quanto para escrita, do operador da rede na comunicação entre os sistemas. A Figura 4.22 apresenta uma ilustração da tela inicial interface desenvolvida. Mais imagens da interface estão presentes no Anexo 2.

Figura 4.22. Interface de escrita e leitura de dados.



A utilização da interface em modo *off-line* (*desconectado da internet*) foi realizada por meio de aplicativos como MAMP (Macintosh, Apache, MySQL and PHP) ou WAMP (Windows, Apache, MySQL and PHP). Tais programas, que podem ser baixados gratuitamente, criam um servidor local no computador em que ele está sendo executado. Dessa forma, é possível utilizar a interface criada em PHP utilizando qualquer navegador. Além disso, ambos os softwares disponibilizam a utilização da ferramenta “phpMyAdmin”, que constitui uma maneira fácil de configurar as tabelas nos bancos de dados. No caso específico da interface criada na integração com o SMRA e a rede elétrica, utilizou-se o MAMP.

No estudo de caso representado nesse capítulo foram apresentadas e descritas as etapas envolvidas no processo de modelagem utilizando o modelo de dados contido no CIM. Foi possível analisar o levantamento e o mapeamento dos dados, assim como a geração do arquivo XSD, num cenário típico de sistemas de potência. Os documentos XML contendo as informações da rede foram elaborados e validados, conferindo segurança e qualidade dos dados trocados. Dessa maneira, estabeleceu-se a integração do aplicativo SMRA à rede elétrica por meio da descrição de seus dados utilizando o modelo comum de dados.

O desenvolvimento da interface para leitura e escrita de dados conferiu ao sistema integrado uma nova dinâmica. Além de possibilitar ao responsável pelo monitoramento do sistema um mecanismo de intervenção direta, a utilização do banco de dados MySQL auxilia na identificação de comportamentos atípicos e na definição de diagnósticos.

---

**CAPÍTULO 5. CONCLUSÃO GERAL**

---

Essa dissertação teve como foco principal o estudo e a aplicação do modelo constante da norma IEC 61970 (*Common Information Model*).

As características gerais do CIM foram apresentadas, assim como a lógica de sua utilização. Os conceitos necessários para a compreensão integral do modelo, como a UML e a XML, tiveram seus fundamentos apresentados. Observou-se que a escolha pelos diagramas UML para representar o modelo de dados simplifica significativamente a utilização do modelo, criando uma estrutura que obedece a uma lógica bem definida e que impede o aparecimento de repetições desnecessárias. O uso da linguagem XML como formato ideal para o transporte dos dados é justificado, ao serem explicadas sua extensibilidade e a garantia da segurança da informação mediante o uso do XSD.

Uma proposta de integração de dois sistemas heterogêneos, um multiagente e uma rede elétrica teste, foi apresentada como estudo de caso para exemplificar o emprego do CIM num caso concreto. Os processos de levantamento e mapeamento dos dados foram descritos em cada etapa, dividindo-se a rede elétrica modelada em regiões funcionais de modo a facilitar o uso dos dados. Ressaltou-se, dessa forma, a conveniência do modelo comum de dados no estabelecimento da interoperabilidade entre sistemas heterogêneos, num contexto em que o conceito de Redes Elétricas Inteligentes tem-se tornado a diretriz principal no processo de modernização dos sistemas de potência.

O desenvolvimento de uma interface para permitir a intervenção direta do operador na troca de informações dos sistemas integrados visa alcançar um outro fundamento das redes inteligentes: a descentralização e a bidirecionalidade do fluxo de dados. A interface, construída em PHP e utilizando o banco de dados MySQL, foi desenvolvida de forma a se adequar a ambientes diversos. Dessa maneira, o conjunto formado pela interface, pelo multiagente SMRA e pelo simulador da rede elétrica, compõem uma rede cujas informações estão plenamente integradas.

## 5.1. Trabalhos futuros

Como sugestões de trabalhos futuros, tendo como objetivo incrementar o processo de integração de sistemas, propõe-se:

- a) Criação de novas funcionalidades para a interface, como rotinas que executem análises dos dados, geração de gráficos e índices de qualidade, entre outros;
- b) Desenvolvimento de simulador com interface gráfica que auxilie na construção dos arquivos XML no formato do padrão CIM;
- c) Aplicação da metodologia utilizada nessa dissertação na integração de microrredes, com as extensões no CIM necessárias para a representação de novas classes.

## REFERÊNCIAS BIBLIOGRÁFICAS

AYERS, L. M. Implementing Smart Grid standards: A letter from the trenches. *In: Innovative Smart Grid Technologies Asia*, 2011, Austrália. IEEE Power & Energy Society, 2011.

CAUCHON, L.; BOUFFARD, A.; DOLAN, D.; PELOQUIN, M.; MICHAUD, C. Real-Time IEC 61970 based System for Bulk Power System Restoration at Hydro-Quebec. *In: International Conference on Communication Software and Networks*, 3., 2011, China. IEEE Power & Energy Society, 2011. p. 100-104.

CIM Users Group (CIMug): <http://cimug.ucaiug.org>. Acessado em 19/12/2013.  
DALL’OGLIO, Paulo. **PHP: Programando com Orientação a Objetos**. 2. ed. Brasil: Novatec. 2009. p. 20.

DA CUNHA, C. A. S., Jr., REIN, O., JARDINI, J. A., MAGRINI, L. C. **Electrical utilities control center data exchange with ICCP and CIM/XML**. *In: Transmission and Distribution Conference and Exposition: Latin American*. IEEE Power & Energy Society, 2004. p. 260 – 265.

DING, M.; ZHANG, Z.; GUO, X. CIM Extension of Microgrid Energy Management System. *In: Power and Energy Engineering Conference*, 2009, China. China: IEEE Power & Energy Society, 2009.

Eclipse IDE: [www.eclipse.org](http://www.eclipse.org). Acessado em 19/12/2013.

ELECTRIC POWER RESEARCH INSTITUTE (EPRI). **CIM Primer**. EUA, 2011.

ELECTRIC POWER RESEARCH INSTITUTE (EPRI). **Report to NIST on the Smart Grid Interoperability Standards Roadmap**. EUA, 2009. p. 10.

FAWCETT, Joe; QUIN, Lian R. E.; AYERS, Danny. *Beginning XML*. 5. ed. EUA: Wiley, 2012.

FOWLER, Martin. *UML Essencial*. 3. ed. Brasil: Bookman, 2004. p. 25.

Guo, L.; Li-Ke, G.; Bo, C.; Hai-Long, H.; Yong-Gang, L.; Zhi-Gang, X.; et al. Design of IEC-61968-Based Distribution Network Information Exchange Interface. **Procedia Engineering**, v. 24, p. 69–77, 2011.

HAIFENG, H; JINHU, Z; YANG, C; TAO, H.; XIN, S; YAPING, L. The Study of Data Exchange Technology Among Control Center Systems. *In: International Conference on Power System Technology*, 2006, China: IEEE Power & Energy Society, 2013.

HARGREAVES, N.; PANTEA, S.; TAYLOR, G.; IRVING, M. A Critical Comparison of Approaches to Resource Name Management within the IEC Common Information Model. *In: Innovative Smart Grid Technologies Europe*, 3., 2012, Alemanha. IEEE Power & Energy Society, 2012.

ILICH P. R.; RIDDLES, R.; HAAK, W.; FROWD, R. Application of CIM Model for Enterprise Wide Power System Model for Planning, Protection and Operations. *In: Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, 2008, EUA. IEEE Power & Energy Society, 2008.

INTERNATIONAL ELECTROTECHNICAL COMISSION (IEC). **IEC Smart Grid Standardization Roadmap**. 1. ed. EUA, 2010.

Langdale Consultants: [www.langdale.com.au](http://www.langdale.com.au). Acessado em 19/12/2013.

LENDAK, I.; VARGA, E.; ERDELJAN, A; GAVRIÉ, M. RESTful Web Services and the Common Information Model (CIM). *In: International Energy Conference*, 2010, Bahrein. IEEE Power & Energy Society, p. 716-721.

LI, Y.; JUN, J.; LING, L. Study of Real-time Database Based on Common Information Model. *In: International Conference on Intelligent System Design and Engineering Applications*, 3., 2013, Hong Kong. IEEE Power & Energy Society, 2013. p. 1312-1315.

LIJUN, Q.; CUIJUAN, H.; HUAWEI, J.; MENG, L. Information Model for Power Grid Fault Diagnosis Based on CIM. *In: International Conference on Electric Utility Deregulation and Restructuring and Power Technologies*, 4., 2011, China. IEEE Power & Energy Society, 2011.

MAMP: [www.mamp.info](http://www.mamp.info). Acessado em 19/12/2013

McArthur, S. D. J.; Davidson, E. M.; Catterson, V. M.; Dimeas, A. L.; Hatziargyriou, N. D.; Ponci, F.; Funabashi, T. Multi-Agent Systems for Power Engineering Applications - Part I: Concepts, Approaches, and Technical Challenges. **IEEE Transactions on Power Delivery**, v. 22, n. 4, p. 1743–1752, Nov. 2007.

MELO, L. S. **Abordagem para Desenvolvimento de Sistema de Recomposição Automática para Rede de Distribuição de Energia Baseado em Sistemas Multiagentes**. Trabalho de Conclusão de Curso. Universidade Federal do Ceará – UFC, 2013, 79p.

MERCURIO, Andrea; DI GIORGIO, Alessandro; CIOCI, Pierfrancesco. Open-Source Implementation of Monitoring and Controlling Services for EMS/SCADA Systems by Means of Web Services - IEC 61850 and IEC 61970 Standards. **IEEE Transactions on Power Delivery**, v. 24, n. 3, p. 1148–1153, Jul. 2009.

MILANI, André. **MySQL – Guia do Programador**. 1. ed. Brasil: Novatec. 2006. p. 22.

MOSELEY, J.D.; MAGO, N.V.; LEGATT, M.E. A method for automatic creation of contingency definitions utilizing standard IEC 61968/70. *In: Transmission and Distribution Conference and Exposition (T&D)*, 2012, EUA. **Conferência...** EUA: IEEE Power & Energy Society, 2012.

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY (NIST). **Framework and Roadmap for Smart Grid Interoperability Standards, Release 2.0. 2.** ed. EUA, 2012.

Optical Zeitgeist Laboratory, Guide for Smart Grid Interoperability: <http://zeitgeistlab.ca/doc/Guide-for-Smart-Grid-interoperability.html>. Acessado em 19/12/2013.

PRADEEP, Y.; SESHURAJU, P.; KHAPARDE, S.A.; JOSHI, R.K. CIM-Based Connectivity Model for Bus-Branch Topology Extraction and Exchange. **IEEE Transactions on Smart Grid**, v. 2, n. 2, p. 244-253, Mar. 2011.

REIN, O.; JARDINI, J.A.; SEIZO HOKAMA, W.; MAGRINI, L.C. Advantages of Adopting Web Services in Smart Grids. *In: Innovative Smart Grid Technologies Latin America*, 2013, Brasil. IEEE Power & Energy Society, 2013.

Solanki, J. M.; Khushalani, S.; Schulz, N. N. A Multi-Agent Solution to Distribution System Restoration. **IEEE Transactions on Power Delivery**, v. 22, n. 3, p. 1026 – 1034, Aug. 2007.

Sučić, S., Dragičević, T., Capuder, T., Delimar, M. Economic dispatch of virtual power plants in an event-driven service-oriented framework using standards-based communications. **Electric Power System Research**, v. 81, n. 12, p. 2108 – 2119, Dez. 2011.

UNIÃO EUROPEIA. **Advanced Architectures and Control Concepts for More Microgrids**. UE, 2007.

USLAR, M.; ANDREN, F.; MAHNKE, W.; ROHJANS, S.; STIFTER, S.; STRASSER, T. Hybrid Grids: ICT-based Integration of Electric Power and Gas Grids – A Standards Perspective. *In: Innovative Smart Grid Technologies Europe*, 3., 2012, Alemanha. **Conferência...** Alemanha: IEEE Power & Energy Society, 2012.

Uslar, M.; Rohjans, S.; Specht, M. Technical Requirements for DER Integration Architectures. **Energy Procedia**, v. 20, p. 281–290, 2012.

USLAR, Mathias; SPECHT, Michael; ROHJANS, Sebastians; TREFKE, Jörn; GONZÁLEZ, José Manuel. **The Common Information Model CIM: IEC 61978/61970 and 62325 – A Practical Introduction to the CIM**. 1. ed. Alemanha: Springer, 2011. p. IX

WAMP: [www.wampserver.com](http://www.wampserver.com). Acessado em 19/12/2013.

WANG, C.; LIU, H.; WU, F. The Extend CIM for MicroGrid. *In: China International Conference on Electricity Distribution*, 2012, China.

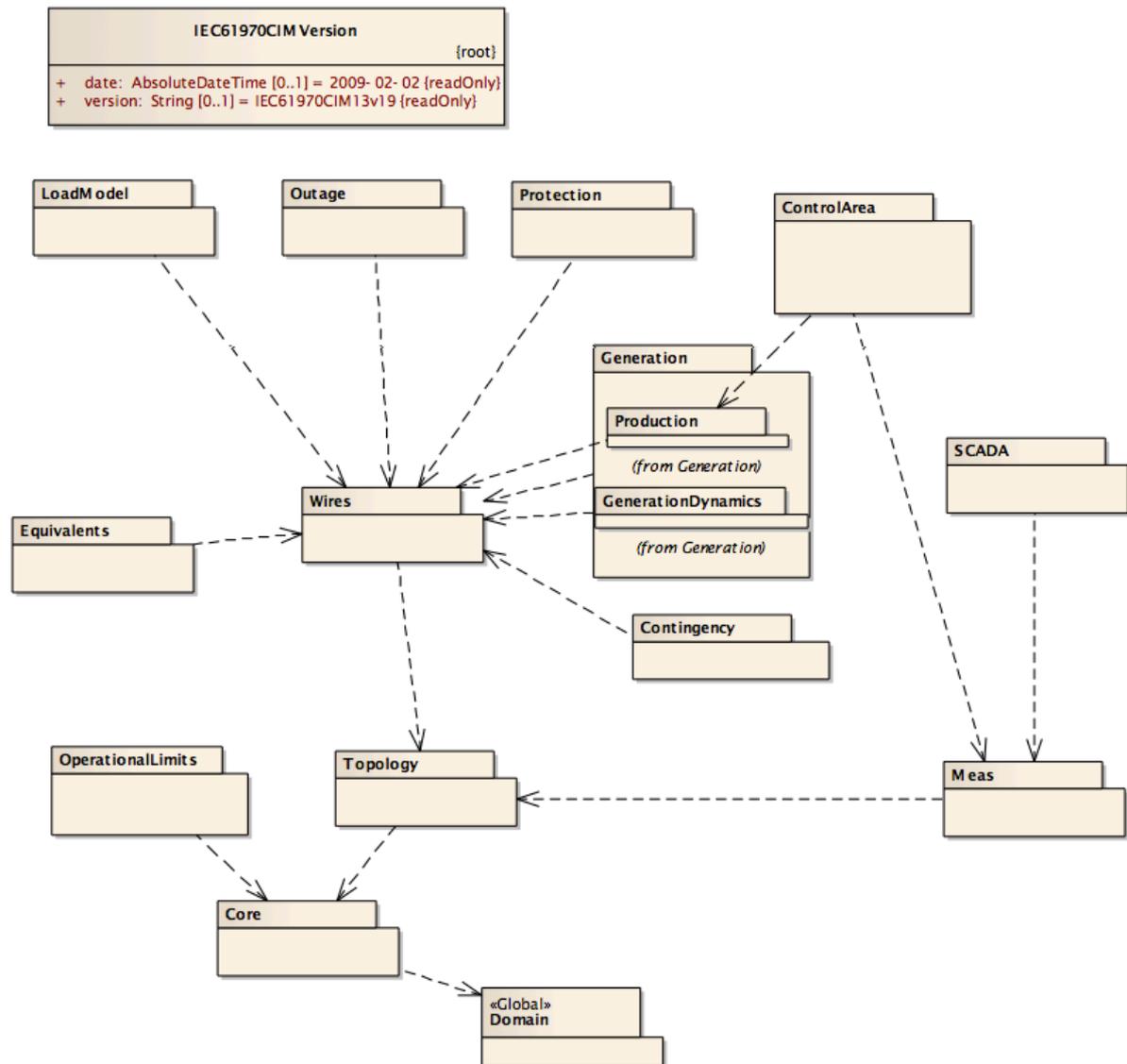
Wang, L.; Hosokawa, A.; Murayama, H. An Evolutive and Multilingual CIM Ontology Management System. **Energy Procedia**, v. 12, p. 18–26, 2011.

WORLD WIDE WEB CONSORTIUM (W3C). **RDF Primer**. EUA, 2004.

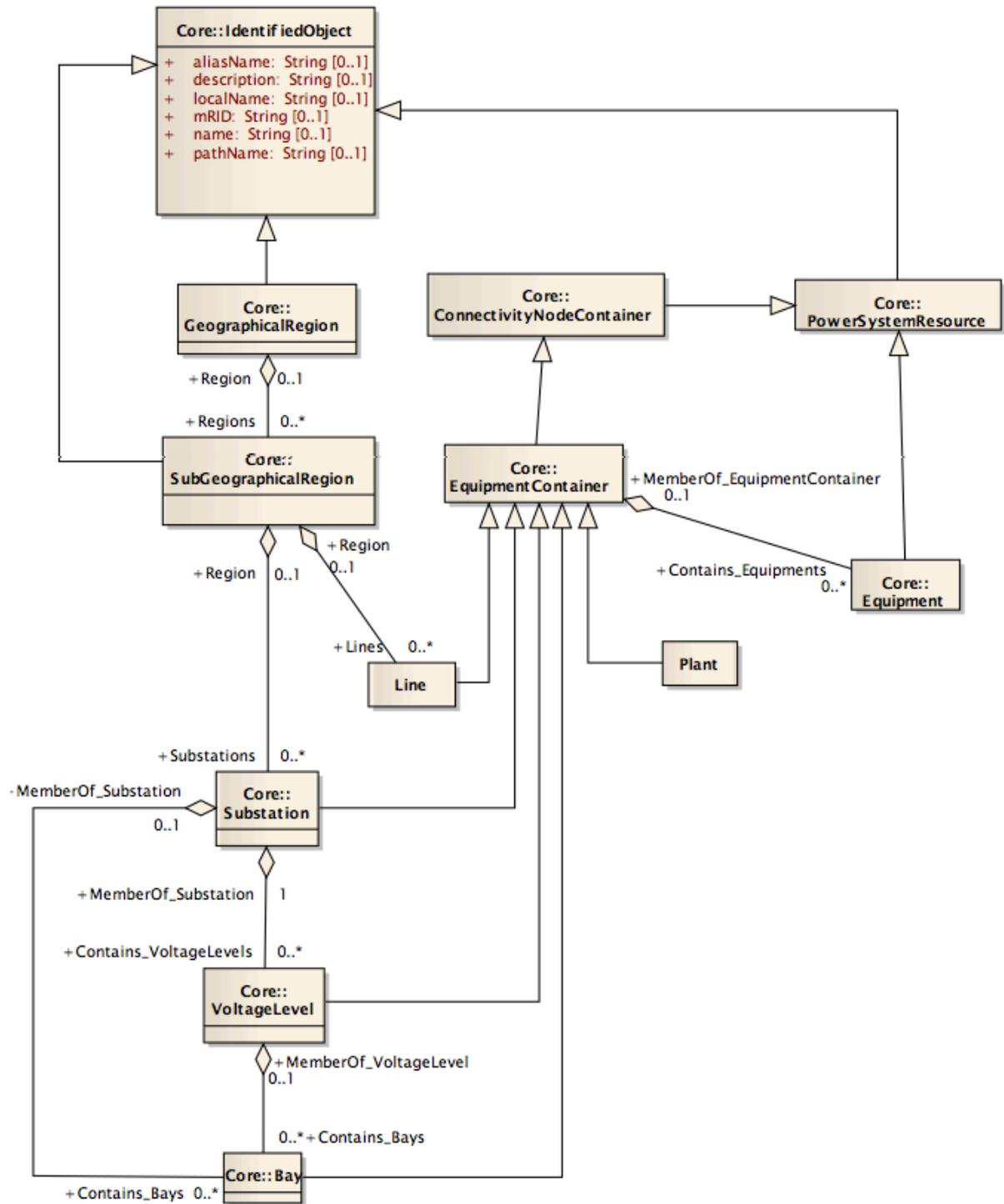
WUERGLER, E.; VANHEMELRYCK, C. Use of CIM for workflows across network operation, asset management and Network planning systems at a distribution utility. *In: Power and Energy Society General Meeting, 2012, EUA. IEEE Power & Energy Society, 2012.*

Zhang, Y.; Zhu, Y.; Bai, X.; Wei, H. CIM-based Data-sharing Scheme for Online Calculation of Theoretical Line Loss. **Energy Procedia**, v. 16, p. 1619–1626, 2012.

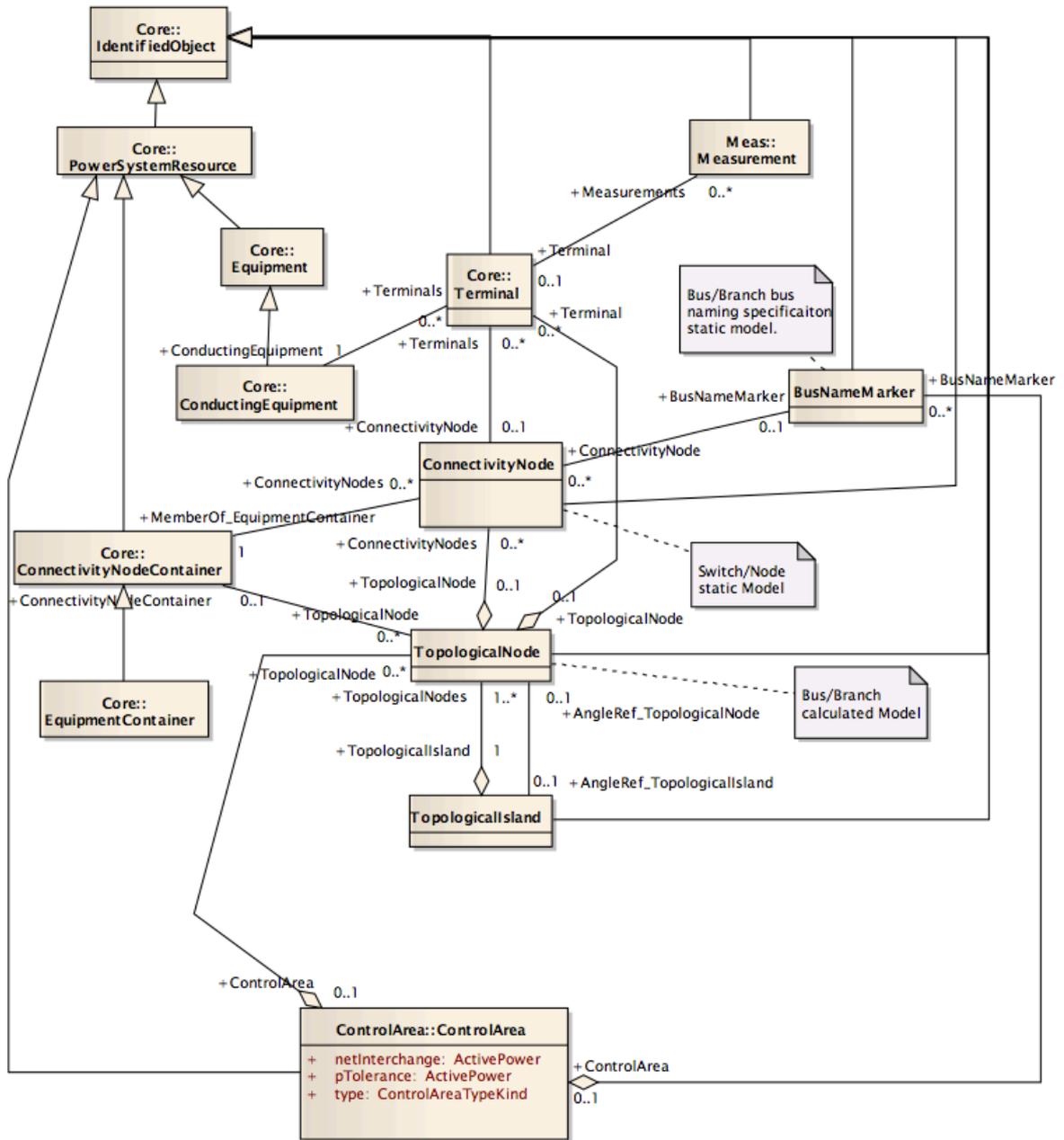
## a) Dependência entre os pacotes



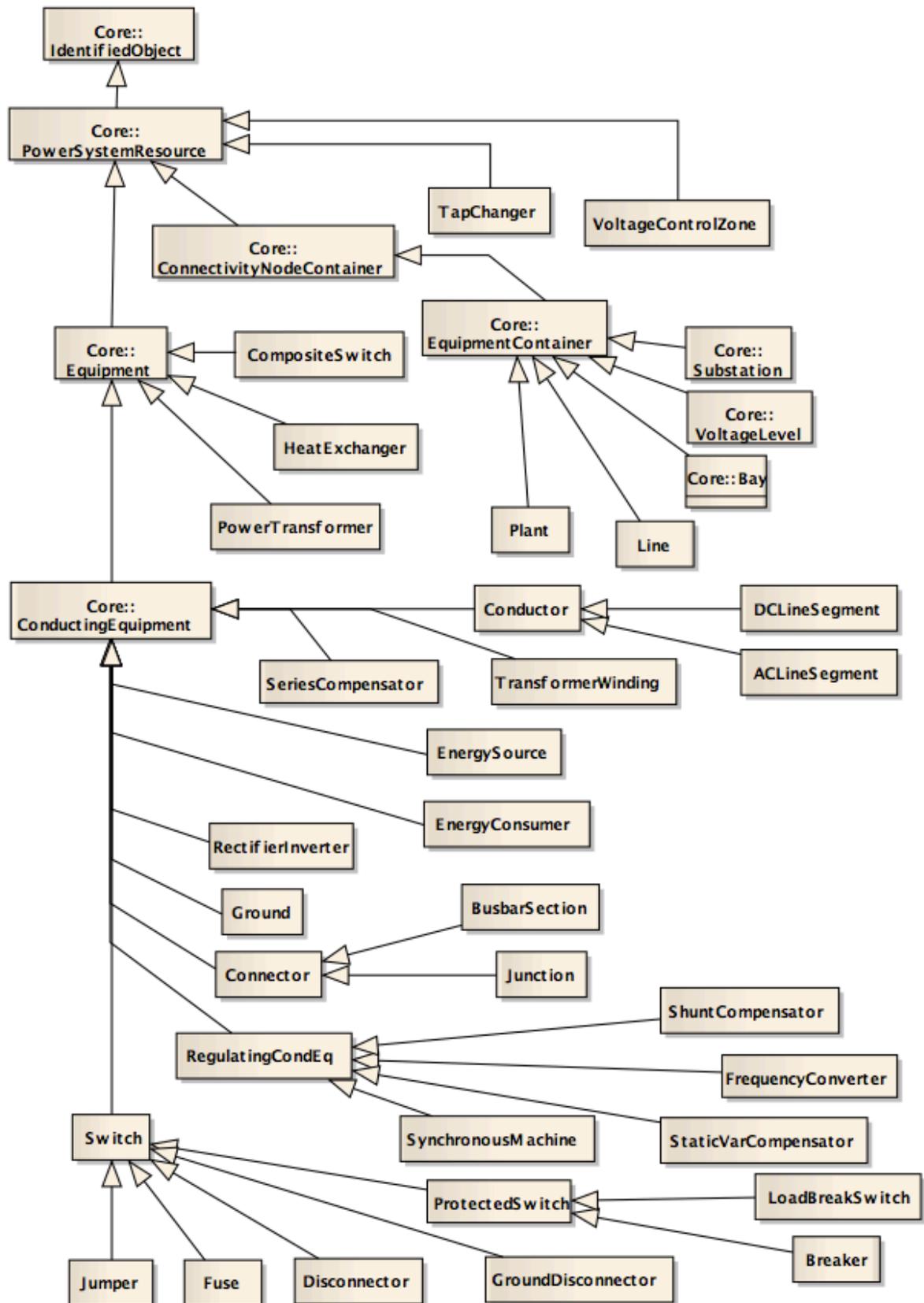
## b) Contêineres de equipamentos



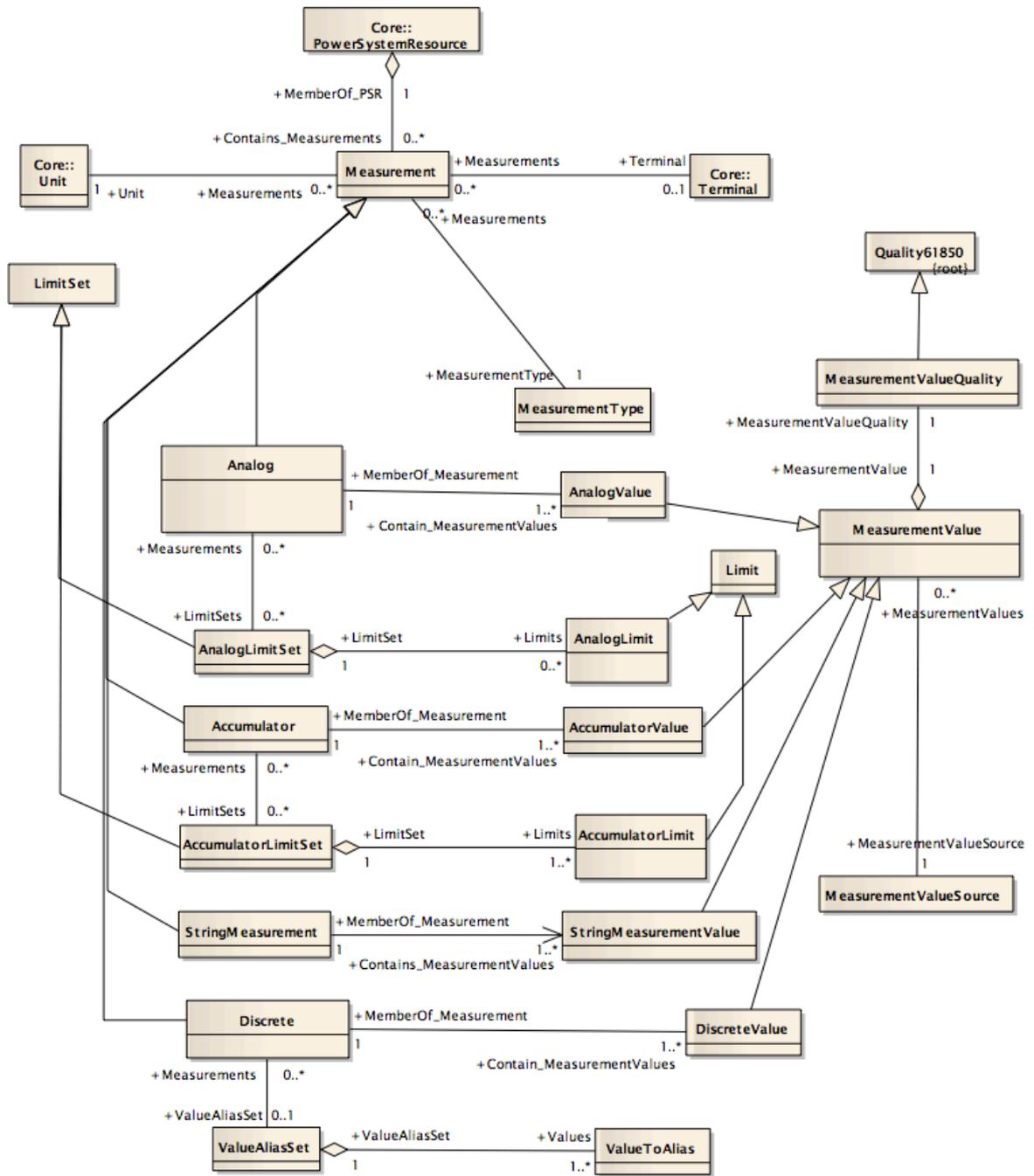
c) Classes de conectividade



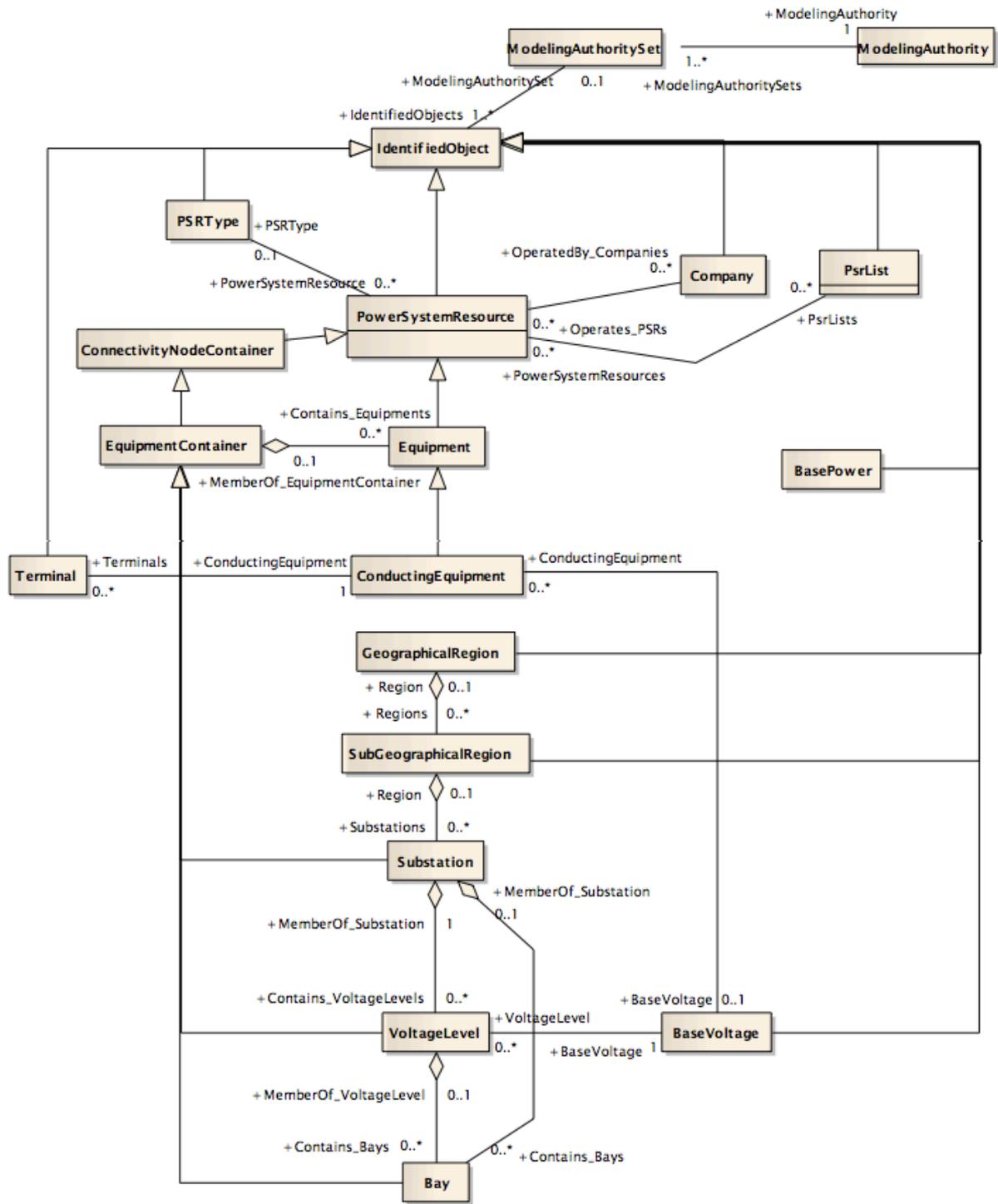
## d) Generalizações de equipamentos



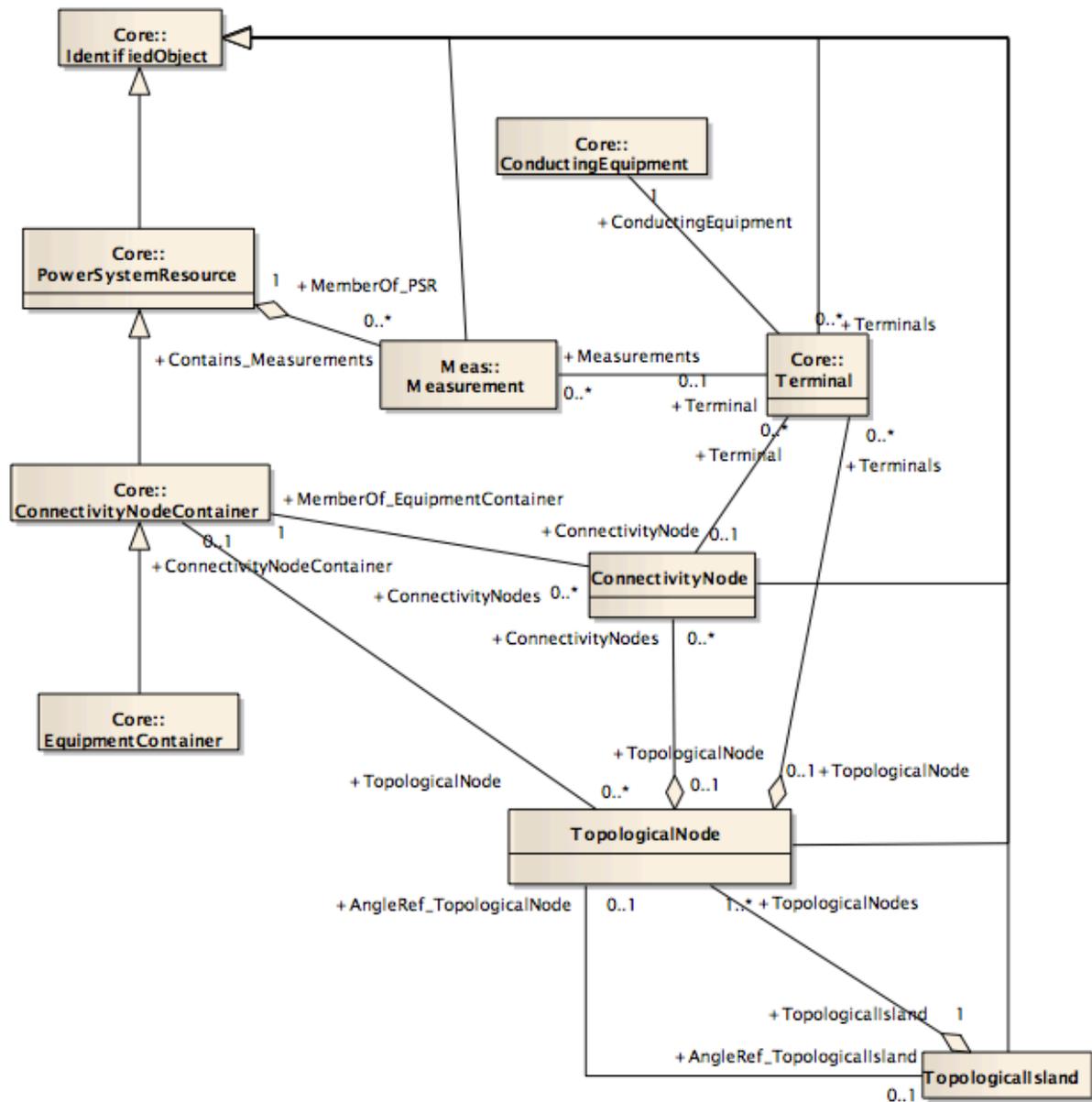
e) Classes de medidas



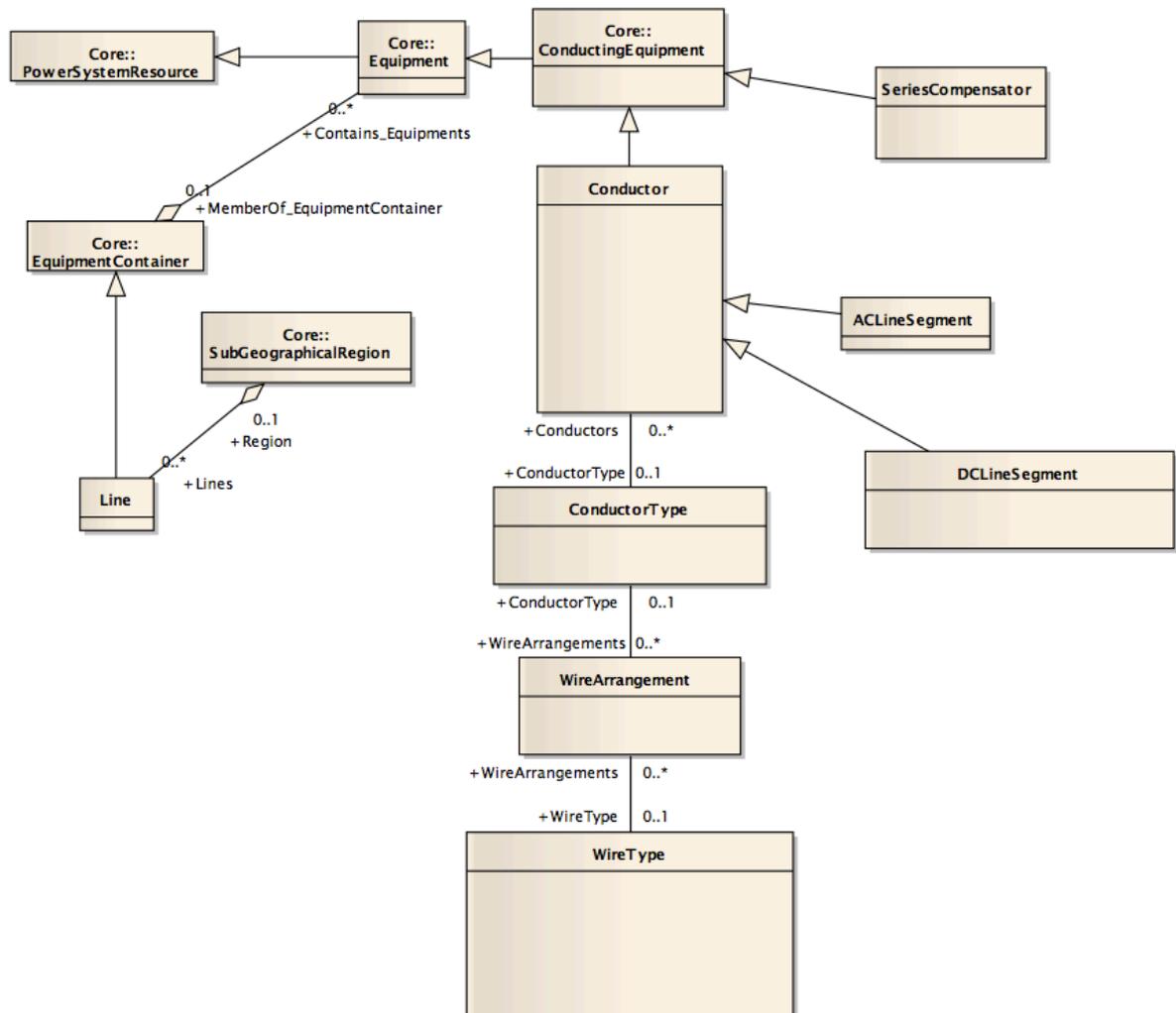
f) Tensão de base



## g) Nó topológico



## h) Cabos e Condutores



## a) Tela de acesso aos dados das chaves e dos trechos do Centro

The screenshot shows a web interface for the 'Centro' substation. At the top left, there is a green arrow-shaped button labeled 'Centro'. Below it, the title 'Interface de Monitoramento de Dados' is displayed in a large, bold, black font. Underneath the title, the word 'Menu' is written in a smaller font. The main content area consists of two stacked rectangular boxes. The top box contains two buttons: 'Disjuntores e Religadores' and 'Trechos'. The bottom box contains a single button: 'Retornar ao Menu Inicial'.

## b) Tela de acesso aos dados da SE Aquiraz

The screenshot shows a web interface for the 'SE Aquiraz' substation. At the top left, there is a green arrow-shaped button labeled 'SE AQZ'. Below it, the title 'SE Aquiraz' is displayed in a large, bold, black font. Underneath the title, the word 'Menu' is written in a smaller font. The main content area consists of two stacked rectangular boxes. The top box contains two buttons: 'Inserir Dados da Subestação Aquiraz' and 'Dados Atuais da Subestação Aquiraz'. The bottom box contains a single button: 'Retornar ao Menu de Subestações'. At the bottom of the interface, there is a decorative horizontal bar with a gradient from light gray to white.

**c) Tela de preenchimento dos dados de duas chaves do Centro**

Chave RS5	
<input type="radio"/> Aberto <input type="radio"/> Fechado	
Terminal 33	
Corrente	<input type="text"/>
Tensão	<input type="text"/>
Terminal 34	
Corrente	<input type="text"/>
Tensão	<input type="text"/>
Chave RL4	
<input type="radio"/> Aberto <input type="radio"/> Fechado	
Terminal 35	
Corrente	<input type="text"/>
Tensão	<input type="text"/>
Terminal 36	
Corrente	<input type="text"/>
Tensão	<input type="text"/>
<input type="button" value="Inserir Dados"/> <input type="button" value="Limpar Formulário"/>	

#### d) Tela de acesso aos dados das subestações

The screenshot shows a web interface for monitoring data. At the top left, there is a green arrow-shaped button labeled 'Subestações'. Below it, the main title 'Interface de Monitoramento de Dados' is displayed in a large, bold, black font. Underneath the title, the word 'Menu' is written in a smaller font. The interface is divided into two main sections, each enclosed in a rectangular box. The first box contains four buttons stacked vertically: 'Substação Aquiraz', 'Substação Messejana', 'Substação Jabuti', and 'Substação Água Fria'. The second box contains three buttons stacked vertically: 'Gerar XML das Subestações', 'Leitura de Dados de Subestações', and 'Retornar ao Menu Inicial'. At the bottom of the interface, there is a large, light gray, rounded rectangular bar. On the far left and right sides of this bar, there are large curly braces '{' and '}' respectively.

#### e) Tela seleção de arquivo para leitura de dados das subestações

The screenshot shows a web interface for selecting a file for data reading. At the top left, there is a green arrow-shaped button labeled 'Subestações'. Below it, the main title 'Subestações' is displayed in a large, bold, black font. Underneath the title, the text 'Selecione o arquivo para leitura' is written in a smaller font. Below this text, there is a button labeled 'Selecionar arquivo...' followed by the text 'Nenhum arquivo selecionado.'. Below this, there is a button labeled 'Ler'.