

Sistema de Integração de Tecnologias de Agregação de Medição

J. C. B. Leite, O. G. Loques, E. Fontana, A. Corradi, P. R. Santos e N. Rizzi

Resumo - O suporte às atividades de uma empresa requer a colaboração entre os sistemas de computação que ela possui para compartilhar informações e processos. Diversas organizações têm unido esforços no sentido de estabelecer padrões, arquiteturas e formatos de representação de dados que permitam a interoperabilidade entre os sistemas de uma empresa e mesmo entre empresas. Alguns pontos em comum podem ser observados entre as diversas iniciativas, um deles é que a integração de sistemas heterogêneos está sendo tratada como uma questão de “entendimento” entre esses sistemas, outro é a incorporação dos conceitos de Orientação a Objetos, permitindo a construção de modelos de dispositivos a partir de blocos funcionais básicos. Esse documento apresenta os resultados do projeto desenvolvido no Instituto de Computação da Universidade Federal Fluminense, no âmbito do Programa de P&D da ANEEL, em conjunto com a Eletropaulo, para a integração de sistemas heterogêneos de Medidores Eletrônicos de Energia Elétrica.

Palavras-chave — Integração baseada em modelos, interoperabilidade, medidores de energia elétrica, sistemas heterogêneos.

I. INTRODUÇÃO

Muitas vezes, o suporte às atividades de uma organização requer a colaboração entre os sistemas de software que ela possui, para compartilhar informações e processos disponíveis nesses sistemas. Assim, diversas organizações têm unido esforços no sentido de estabelecer padrões, arquiteturas, formas de representação de dados que permitam a interoperabilidade entre os sistemas de uma empresa. No âmbito das Empresas de Energia Elétrica (EEEs) surgiram várias propostas nesse sentido. A UCA (*Utility Communications Architecture*) desenvolvida pelo EPRI (*Electric Power Research Institute*) é uma proposta de arquitetura que tem como objetivo estabelecer padrões para integração das diversas áreas funcionais dentro das empresas do setor [UCA 97]. O documento GOMSFE (*Generic Object Model Substations & Feeder Equipment*) [UCA 00], um dos documentos básicos definidos pela UCA, define blocos de construção básicos para a composição de modelos lógicos de dispositivos. Outra proposta é a DLMS/COSEM (*Device Language Message Specification / Companion Specification for Energy Metering*) [DLMS 02] que é uma iniciativa de um grupo de

empresas européias e visa especificar um medidor de energia interoperável, que troca informações baseadas em serviços.

Alguns pontos em comum podem ser observados entre as diversas iniciativas. Um deles é que cada vez mais a integração de sistemas heterogêneos está sendo tratada como uma questão de “entendimento” entre esses sistemas. Assim, as propostas conduzem a uma abordagem que privilegia o entendimento da informação sendo compartilhada, através da utilização de modelos lógicos de informação. Esses mecanismos permitem que os sistemas possam entender corretamente os significados das informações sendo compartilhadas. Essa abordagem é chamada de Integração Semântica ou Integração Baseada em Modelos [Reyn 03].

Um outro ponto tem sido a incorporação de conceitos do paradigma de *software* conhecido como Orientação a Objetos. Essa tecnologia permite a construção dos modelos de dispositivos a partir da composição de blocos funcionais básicos.

Um dos principais problemas com as iniciativas é que acarretam uma completa reestruturação das empresas para a implantação de suas propostas.

No que diz respeito aos medidores de energia, o que acontece é que diferentes modelos com diferentes concepções do domínio do problema precisam de softwares distintos e proprietários para ler e alterar a totalidade dos dados em cada modelo.

O objetivo deste trabalho é apresentar um sistema que permite integrar diversos modelos de medidores de energia já existentes e que é extensível aos futuros medidores. A estratégia para atingir este objetivo foi a definição de um medidor genérico que representa, através de alguma especialização, os medidores já existentes. Deste modo, neste trabalho foi elaborado um modelo (lógico) genérico de medidor configurável e expansível, capaz de integrar a definição de diversos medidores específicos.

Uma tecnologia que vem se estabelecendo em muitas áreas da computação é a XML (*eXtensible Markup Language*) [W3C 98]. A XML surgiu como promessa e de fato vem se estabelecendo como uma ferramenta importante e amplamente usada na integração de sistemas. A XML é um padrão de representação de dados de forma estruturada e em formato texto que permite entre outras coisas especificar a semântica dos dados. Toda a modelagem desse trabalho foi criada utilizando tecnologias relacionadas a XML.

Com a proposta de uma arquitetura baseada na Internet na construção do sistema de integração, a questão da segurança dos dados foi uma preocupação adicional no desen-

J. C. B. Leite (e-mail julius@ic.uff.br), O. G. Loques (e-mail loques@ic.uff.br), E. Fontana (e-mail efontana@ic.uff.br) e A. Corradi trabalham no Instituto de Computação da Universidade Federal Fluminense.

N. Rizzi (e-mail: nelson.rizzi@aes.com) e P. R. Santos (e-mail: paulo.roberto@aes.com) trabalham na AES Eletropaulo.

volvimento deste trabalho. Diversos níveis de segurança foram incorporados ao sistema.

II. FUNDAMENTOS DA MODELAGEM EMPREGADA

Na modelagem foram empregados os conceitos de Orientação a Objetos. O uso desta abordagem permite a especificação de classes e objetos que definem de forma abstrata a semântica das funções de um medidor genérico.

Nesse sentido, o objetivo principal é definir blocos básicos de construção (componentes) reutilizáveis a serem usados na especificação de modelos de medidores específicos de diferentes fabricantes. Um conceito fundamental da Orientação a Objetos é o de “herança”, que permite expressar semelhanças entre classes na estrutura do domínio do problema. Isto permite a definição de novas classes baseadas em classes já definidas. A aplicação desse conceito permite que as funções e atributos comuns de um medidor genérico sejam definidos uma única vez, em uma classe abstrata. Desta forma, medidores específicos de cada fabricante podem ser obtidos como especializações ou implementações dessa classe. Instâncias correspondentes aos dispositivos físicos podem então ser representadas por objetos da classe associada (ver Figura 1).

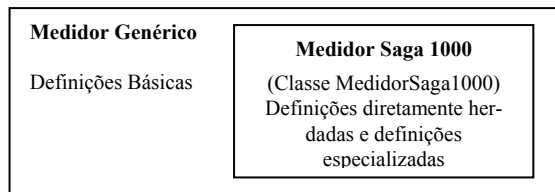


Figura 1: Um dos benefícios da modelagem de dispositivos é facilitar o reuso de definições comuns.

Uma iniciativa nessa área, e ainda em estágio de desenvolvimento, é a especificação COSEM [DLMS 02] produzida pela Associação DLMS. O modelo que é proposto nesse trabalho usa como base às classes de interface definidas nessa especificação, para a construção do modelo de medidor de energia genérico. COSEM foi escolhido, pois, no estágio atual dos esforços em direção à padronização de dispositivos de campo relacionados a empresas de energia elétrica, o COSEM é o único que já define uma modelagem de medidores de energia elétrica. Assim, o modelo foi baseado nessa especificação, procurando descrever através de suas classes as funções de medição definidas na Norma ABNT NBR 14522.

É importante ressaltar que não implementamos o padrão DLMS/COSEM, usamos apenas suas classes de interface como base para gerar os nossos blocos básicos de construção. Essas classes são mapeadas em tipos de dados compostos na estrutura de esquemas XSD (XML Schema Definition). A composição dessas classes mapeadas sobre o domínio do problema, forma um esquema XSD que representa um modelo lógico de um medidor de energia genérico, que podem ser especializados em XSDs de medidores específicos. Instâncias desses XSDs, são os arquivos XML que representam um estado (um objeto) em um medidor específico. A tecnologia XML foi escolhida como forma de repre-

sentação de dados devido as diversas vantagens que ela possui.

III. MODELAGEM DE CLASSES DE INTERFACE COSEM EM ESQUEMAS XSDS

O primeiro passo na modelagem foi a definição das classes de interface COSEM como blocos básicos de construção nos esquemas XSD. COSEM define 24 classes de interface básicas [DLMS 02]. Em nosso trabalho usamos apenas aquelas que seriam úteis para a modelagem dos medidores nacionais estudados. A notação utilizada para caracterizar uma classe de interface em COSEM, indicada na documentação referida, define o nome da classe, seus atributos e seus métodos. Tomemos como exemplo a classe de interface *Extended Register* (ver Figura 2):

Extended Register		0..n	Class id=4, Version=0		
Attribute(s)		Data Type	Min.	Max.	Def.
1 logical_name	(static)	octet-string			
2 value	(dyn)	instance specific			
3 scaler_unit	(static)	Scal_unit_type			
4 status	(dyn)	instance specific			
5 capture_time	(dyn)	Octet-string			

Figura 2: Classe Extended Register.

Nesse exemplo, os diversos campos indicados estão abaixo descritos (será mantida a nomenclatura em inglês, por compatibilidade com a documentação):

Class name	Extended Register	Descreve a classe
Cardinality	0..n	Especifica o número de instâncias dessa classe que pode existir em um dispositivo lógico.
Class_id	4	Identifica o código da classe (feito pela Associação DLMS)
Version	0	Código de versão da classe
Attributes	No exemplo, de logical_name a capture_time; podem ser estáticos ou dinâmicos	Especifica os atributos que pertencem à classe; logical_name é sempre o primeiro atributo de uma classe e identifica a instância
Data Type	e.g., octet-string	Define o tipo de dados do atributo
Min.		Especifica se o atributo tem um valor mínimo
Max.		Especifica se o atributo tem um valor máximo
Def.		Especifica se o atributo tem um valor default

Tabela 1 – Especificação dos campos da classe de interface *Extended Register*

Com o objetivo de facilitar a visualização da modelagem, as classes de interface e como consequência, o modelo lógico, serão apresentadas por um diagrama estrutural que representa os XSDs.

Por exemplo, a mesma classe *Extended Register* antes apresentada é representada na Figura 3 como o tipo complexo “ExtendedRegister”, com seus elementos constitutivos *class_id*, *version* e todos os seus atributos (*logical_name*, *value*, *scaler_unit*, *status*, e *capture_time*). Sendo essa notação recursiva, o atributo *scaler_unit* é representado pelo tipo complexo *scaler_unit_type*, que possui os elementos *scaler* e *unit*.

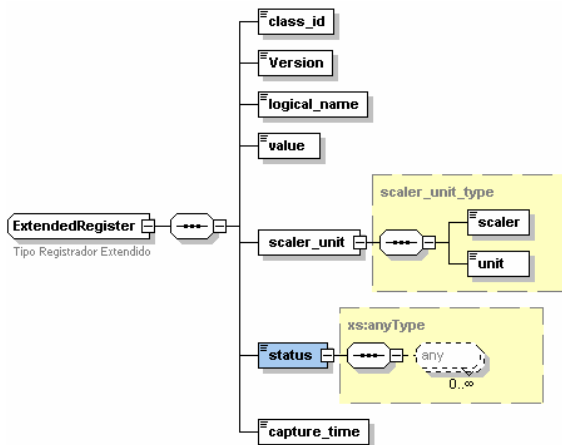


Figura 3: Diagrama estrutural da classe de interface Extended Register.

O formato texto do XSD representado no diagrama da Figura 3 pode ser visto no código 1. Como podemos ver, este XSD mostra as restrições que um objeto do tipo ExtendedRegister e seus atributos deve obedecer. Assim, uma grandeza mapeada como uma instância dessa classe deve ter, por exemplo, o atributo “value”, mapeado no XML como um elemento filho, deve ser do tipo inteiro longo (xs:long), e pode variar de 0 a 99999999.

```
<xs:complexType name="ExtendedRegister" class_id="4" version="0" >
  <xs:annotation>
    <xs:documentation>
      Tipo Registrador Estendido
    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="logical_name">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:whiteSpace value="preserve"/>
          <xs:minLength value="0"/>
          <xs:maxLength value="8"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="value">
      <xs:simpleType>
        <xs:restriction base="xs:long">
          <xs:minInclusive value="0"/>
          <xs:maxInclusive value="99999999"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="scaler_unit" type="scaler_unit_type"/>
    <xs:element name="status" type="xs:anyType"/>
    <xs:element name="capture_time" type="xs:dateTime"/>
  </xs:sequence>
</xs:complexType>
```

Código 1 – Código XSD da estrutura representada na Figura 3

Assim, uma grandeza de um medidor que possui todos os atributos de um tipo ExtendedRegister é modelada como uma instância desse tipo complexo de dados. Com isso, ganha-se em clareza na modelagem, pois se modela apenas uma vez a classe e, para cada grandeza do tipo ExtendedRegister, basta associá-la a esse tipo, como veremos nos itens posteriores.

Diversas outras classes COSEM foram mapeadas. Entre elas, classes que representam as Associações COSEM. O conceito foi incorporado ao modelo como uma forma de oferecer segurança, pois as associações definem as permissões de acesso de um usuário ao dispositivo.

IV. REPRESENTAÇÃO DE UM MEDIDOR DE ENERGIA GENÉRICO

CO

Com base na norma nacional e nos manuais dos medidores, identificamos as características gerais que um Medidor de Energia precisa possuir para atender às necessidades do mercado nacional. Visando sua validação, o modelo proposto, que atua como um elemento unificador, foi utilizado para representar as definições de dados de três diferentes modelos de medidores: o Saga 1000 da ESB, o Spectrum SX da Nansem e o Elo2180 da Elo. A representação unificada resultou em um medidor de energia genérico, que abrange o domínio de problema dos três modelos e a Norma Nacional ABNT 14522.

A estrutura do modelo é definida em sub-seções correspondentes às partes componentes de um medidor de energia genérico. Cada uma dessas partes é definida como um elemento composto de sub-elementos, esse refinamento prosseguindo até atingirmos um tipo básico como definido em COSEM. Um medidor, tipicamente, pode ser visto como indicado na Figura 4. A título de exemplo, ilustraremos a definição das grandezas de faturamento.

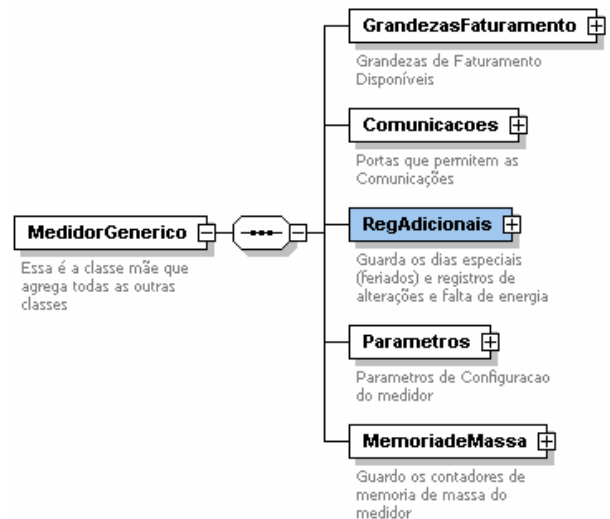


Figura 4: Medidor de energia genérico.

V. GRANDEZAS DE FATURAMENTO

O conjunto de grandezas escolhido para compor cada subgrupo de grandezas de faturamento de um medidor genérico é amplo o bastante para possibilitar a representação dos medidores mais comuns e está de acordo com a norma NBR 14522 da ABNT. Referindo-se à Figura 5, IG representa as Informações Gerais de um medidor, CN1 as grandezas relativas ao Canal 1 (que tratam de energia ativa) e assim por diante. Da especificação COSEM, ASN e ALN representam, respectivamente, as Associações de Short Name e de Logical Name, e LDN é o nome lógico do dispositivo. Podemos notar que no diagrama estão indicadas as duas possibilidades de associação de associações propostas pela especificação COSEM, o que não seria o caso para a especialização de um medidor específico. Contudo, como se trata do modelo genérico, de onde os modelos específicos serão obtidos, as duas possibilidades estão presentes.

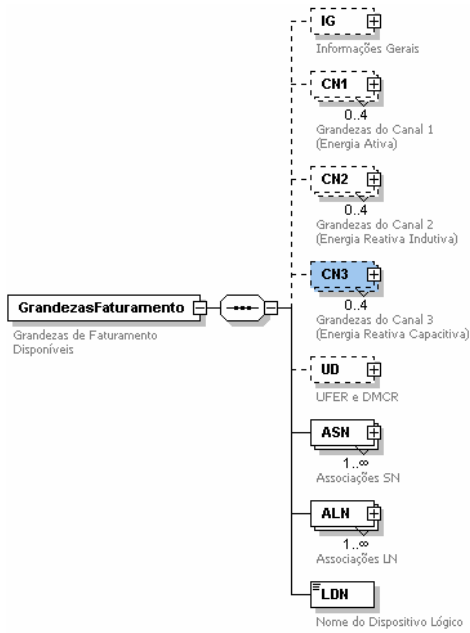


Figura 5: Representação das grandezas de faturamento de um medidor.

Cada um dos elementos que modelam as grandezas de faturamento é composto por elementos filhos, que representam as grandezas específicas. Um exemplo, como mostrado na Figura 6, apresenta uma representação hierárquica dos elementos que compõem o Canal 3 no modelo genérico. Em um medidor de energia o canal 3 contabiliza as informações necessárias ao faturamento relativas a energia reativa capacitiva. Essas informações ficam armazenadas nos registradores totalizadores geral, em horário de ponta, reservado e fora de ponta.

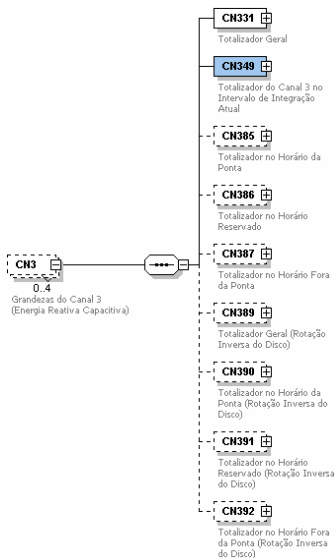


Figura 6: Representação das grandezas do Canal 3.

Adicionalmente, cada uma das grandezas terá um tipo de acordo com suas necessidades e seguindo a recomendação da especificação COSEM. Por exemplo, a grandeza CN349 (Totalizador do Canal 3 no Intervalo de Integração Atual), em destaque na Figura 6, é do tipo Extended Register. A Figura 7 mostra a associação da grandeza ao seu tipo. E assim para cada uma das outras grandezas associa-se uma das classes de interface COSEM que melhor mapeie seus

atributos. A nomenclatura que adotamos, no exemplo “CN349”, diz respeito ao Canal (CN3) e ao código de referência da grandeza nas normas ABNT (49).

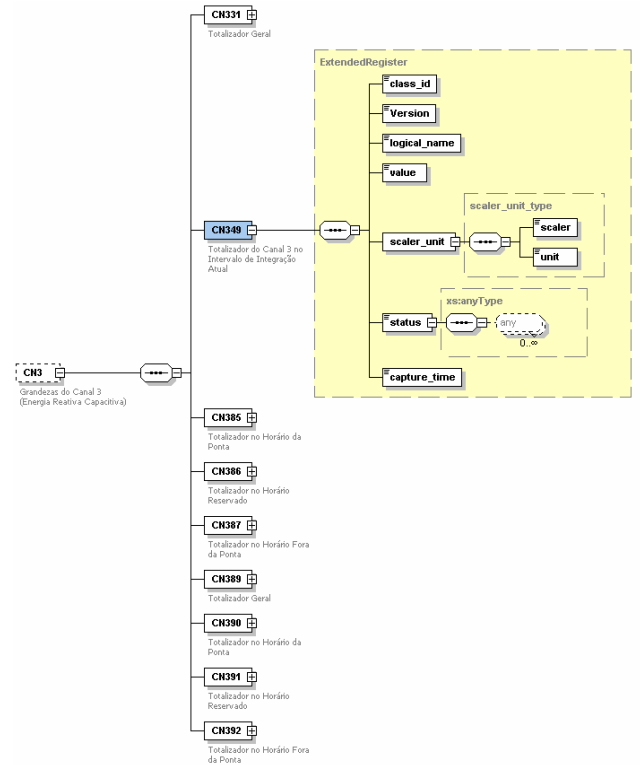


Figura 7: Declaração da grandeza CN349 como do tipo ExtendedRegister.

Finalmente, se a associação utilizada no dispositivo lógico é do tipo AssociationSN (Figura 8), haverá uma instância dessa classe de interface para cada associação que o dispositivo possa manter.

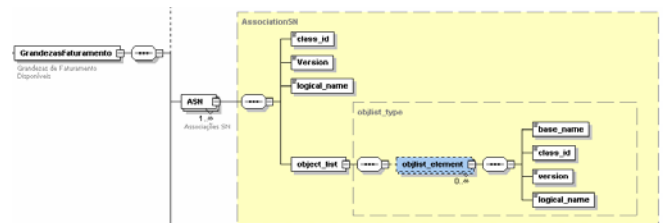


Figura 8: ASN como do tipo AssociationSN.

VI. MODELAGEM ADICIONAL

Ainda compondo um medidor genérico, quatro outros blocos de informação são especificados. O primeiro é relacionado às interfaces de comunicação disponíveis, ou seja, um bloco identificando as características básicas das portas de comunicação. O segundo é relacionado aos registros adicionais, como, por exemplo, os dias especiais (feriados), que são armazenados para o caso de tarifação diferenciada. O terceiro bloco armazena parâmetros de configuração do medidor. E o quarto bloco armazena dados relativos a memória de massa. As linhas gerais de modelagem seguem o que já foi demonstrado no bloco de grandezas de faturamento.

VII. CONSIDERAÇÕES FINAIS SOBRE A MODELAGEM

Foram apresentadas as linhas básicas da modelagem que foi empregada na definição de um medidor genérico. Um modelo genérico, em princípio, abrange toda a possível variedade de medidores, com os seus dados, grandezas e parâmetros. Um ponto importante foi a utilização da tecnologia XML para a modelagem, o que permite a fácil adaptação da representação genérica para qualquer medidor específico. Então, com esse modelo baseado em XSD, descrevemos as funcionalidades e também especificamos os formatos das grandezas e parâmetros dos medidores. A partir daí, para cada medidor específico, pode ser gerado um XSD que especializa o XSD do medidor genérico e que inclui apenas as peculiaridades do medidor específico.

Como exemplo, um XSD que modela o medidor Saga1000, possui no canal 3, apenas as grandezas que existem no dispositivo real, como mostra a Figura 9:

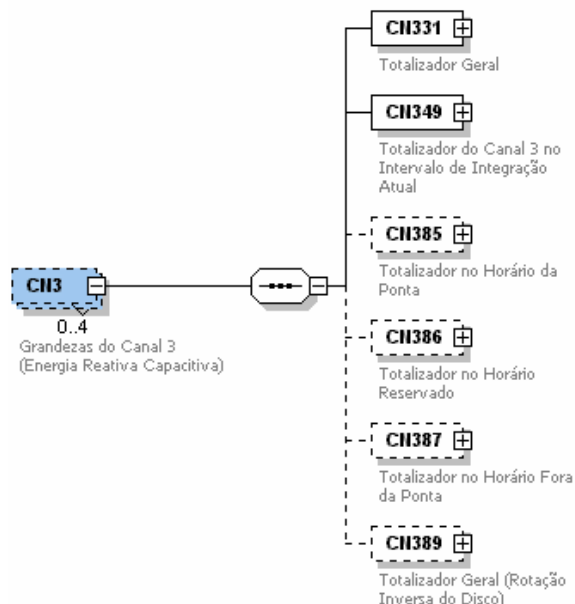


Figura 9: Representação do Canal 3 no medidor SAGA1000.

O XSD de um dispositivo específico define então o formato dos documentos XML que representam objetos (instâncias do XSD) que contém os dados do medidor em um determinado instante. Ou seja, um sistema que deseja realizar uma leitura dos dados de um medidor, solicita e espera receber um documento XML em conformidade com esse XSD, para que seja capaz de processar os dados corretamente. Em adição, devemos considerar o fato de que uma empresa possui diversos medidores de um determinado modelo, o XSD do modelo serve então para representar toda a gama de medidores desse modelo.

Para finalizar, ressaltamos que em toda a fase de modelagem, e como veremos, também na implementação do sistema, houve uma preocupação na adoção de padrões, e em soluções de *software* comerciais e gratuitas. Como vimos o COSEM foi usado para modelagem, sem, no entanto, desprezar a especificação das normas ABNT. E as tecnologias XML usadas como forma de modelagem e representação de dados.

VIII. O SISTEMA PADRONIZADO DE LEITURA E PARAMETRIZAÇÃO DE MEDIDORES

A solução adotada para o desenvolvimento do Sistema Padronizado de Leitura e Parametrização de Medidores de Energia (PLPM) foi a integração dos dados através de um modelo comum de informação que representa o domínio do problema e que pode ser particularizado para representar dispositivos similares de diferentes fabricantes.

O Sistema desenvolvido é capaz de manipular os dados de medidores e de entender o modelo, suas restrições e refinamentos de acordo com cada um dos XSDs específicos que representam cada medidor particular, suas restrições de acesso de acordo com cada usuário e cada grandeza e/ou parâmetro que se deseja ler / alterar.

A utilização de um cenário distribuído baseado na *web* para a construção da aplicação tornou-se conveniente por esta apresentar um ambiente suportado por diversas arquiteturas de hardware e software. Isso traz transparência e credibilidade, na medida em que permite a monitoração em tempo real dos dados, por todos os agentes envolvidos: gerador, vendedor, comprador, auditor, etc. A Figura 10 mostra a arquitetura da aplicação:

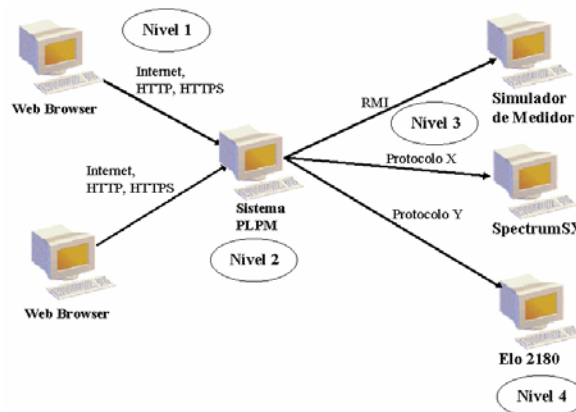


Figura 10 – Arquitetura geral do Sistema

As Interfaces de Usuários (IUs) do protótipo são descritas por páginas HTML, geradas pelo Sistema, contendo *applets* para a exibição dos dados das leituras realizadas no medidor e validação de parâmetros a serem carregados no medidor. As IUs são visualizadas em navegadores *web* e trocam dados codificados em HTML e XML com o Sistema através de protocolos padrões. O Sistema PLPM é um servidor *web*, com sub-funções específicas para ler, processar e armazenar os dados além de repassar parâmetros ao Medidor de Energia.

Como não possuíamos, a princípio, um medidor de energia para testar a aplicação, foi necessária a construção de um simulador de medidor. Basicamente o que esse simulador faz é retornar ao sistema PLPM, quando solicitado, uma leitura em formato XML. Esse arquivo de leitura é obtido pelo mapeamento de arquivos K7 (arquivos de leitura gerados por medidores de energia reais atuais) no formato XML, respeitando a restrição dos modelos específicos representados pelos seus respectivos XSDs.

É importante notar que o sistema PLPM é um sistema, a princípio, projetado para trabalhar com um “medidor ideal”. Para os nossos propósitos, caracterizamos como um medidor ideal, um medidor que quando solicitado pelo sistema,

devolva uma leitura de dados e / ou parâmetros no formato de um arquivo XML em conformidade com o modelo XSD que o descreve. Para adaptar o sistema PLPM aos medidores atuais é necessário o uso de linguagens de transformação como XSLT e *parsers* como JAXP. Essas linguagens permitem transformar os dados do formato de dados que os medidores fornecem para o XML que descreve seu estado e vice-versa.

O protocolo de comunicação entre o sistema PLPM e o simulador de medidor foi o RMI. A escolha desse protocolo foi devida ao fato de estarmos trabalhando com a linguagem Java, pois esta oferece facilidades para trabalhar com esse protocolo. Qualquer outro protocolo de comunicação poderia ser usado, desde que este seja capaz de transferir arquivos no formato XML.

Para ler dados, ou alterar os parâmetros de configuração de um medidor, um usuário precisa acessar a IU através de um navegador. No primeiro instante, é exigida a autenticação do usuário através de um formulário HTML. Se a autenticação for realizada com sucesso, a IU apresenta a interface de leitura e manipulação de dados. O usuário precisa então se identificar para estabelecer seu nível de permissão de acesso aos dados e indicar o medidor ao qual deseja ter acesso. A interface faz uma consulta ao sistema para identificar o usuário e o modelo do medidor e sua configuração padrão, de forma transparente para o usuário.

No contexto de leitura, a aplicação desenvolvida permite que arquivos com formato K7 sejam automaticamente mapeados, pelo simulador, em documentos XML que podem ser entendidos pelo sistema e então exibidos para leitura nas IUs. No contexto de parametrização, a aplicação permite que parâmetros de configuração de medidores de energia sejam inseridos pelos usuários e transformados para as mesmas representações XML, o que permite que sejam manipulados e armazenados de maneira padronizada.

O protótipo utiliza um modelo padronizado como mencionado anteriormente e, portanto, a inclusão de novos tipos de medidores que obedeçam a esse modelo torna-se bastante simplificada.

O Sistema então atua como o servidor da aplicação, ficando responsável por receber as requisições de consulta e atualização e repassá-las aos medidores. Ele mantém um banco de dados que armazena leituras e configurações dos medidores e possui módulos para tratamento dos dados manipulados.

No que se refere a questões de segurança incorporadas, na Figura 10 podemos observar alguns níveis em que dividimos o sistema:

Nível 1: Características de segurança convencionais de transporte de dados adotadas na internet, HTTPS e SSL;

Nível 2: *Login + password* com a tecnologia de *realms* e certificados digitais;

Nível 3: Associações – baseadas nos conceitos da especificação COSEM;

Nível 4: Características atualmente adotadas pelos medidores como o registro das interações entre um usuário e o medidor.

Deve ser acrescentado que no desenvolvimento da aplicação foi mantido um esquema de *log* no sistema PLPM e no simulador de medidor, que é uma maneira adicional de implementar características de segurança.

IX. PROCESSO DE LEITURA DO MEDIDOR

Em um processo de leitura em nosso sistema a interface é adaptada automaticamente, de forma transparente ao usuário, ao medidor específico. Isso é possível através da identificação das restrições de um medidor através de seu XSD. A interface também se adapta as restrições de acesso de cada classe de usuários. A Figura 11 ilustra os dados relativos ao canal 1 de um medidor Saga 1000, de uma leitura realizada por um usuário Consumidor

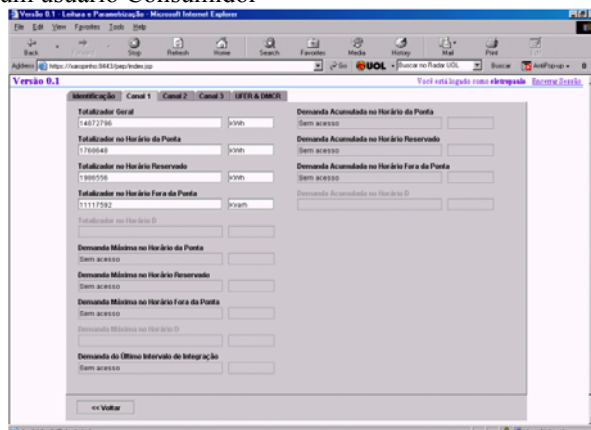


Figura 11 – Processo de leitura do medidor

Como podemos notar, a Figura 11 possui as grandezas listadas em três maneiras, essa foi a forma que achamos para ilustrar as restrições que um medidor possui em relação ao modelo genérico e as restrições de um usuário em relação às associações. As grandezas com o nome em negrito, e os campos em branco e com valores preenchidos como no exemplo as 4 primeiras, representam as grandezas que existem no medidor e que o usuário tem acesso. As grandezas com os nomes em cinza representam as grandezas que existem no modelo genérico, mas não são implementadas no modelo específico. As outras grandezas com o nome em negrito, porém com os campos preenchidos com o texto “Sem acesso”, representam as grandezas que o medidor possui, mas que o usuário atual não tem permissão de acesso. Lembrando que essas permissões são definidas nas associações.

X. CONCLUSÕES

O objetivo principal deste trabalho foi demonstrar, em termos conceituais e práticos, os benefícios e vantagens da abordagem de utilização de modelos para a integração de sistemas heterogêneos, principalmente no contexto de medidores de energia elétrica. Nessa abordagem, a interoperabilidade entre sistemas é tratada em um nível mais alto de abstração, no nível de representações de domínio.

O uso de modelos como estratégia de integração mostrou-se uma boa alternativa no processo de integração de sistemas. Essa abordagem permite que a solução esteja centrada nas funcionalidades e comportamento dos sistemas, separando-a dos detalhes das tecnologias utilizadas na implementação.

XI. BIBLIOGRAFIA

[DLMS 02] DLMS UA 1000-1, “COSEM Identification System and Interface Objects”, 5a. ed., DLMS User Association, 2002.

[Reyn 03] E.M.D. Reynoso, “Integração de Sistemas Baseada em Modelos: Uma Aplicação no Setor Elétrico”, Dissertação de Mestrado, PGCC-IC/UFF, 2003.

[UCA 00] EPRI, “*Generic Object Models for Substation & Feeder Equipment (GOMSFE)*”, Versão 0.91, Fevereiro 2000.

[UCA 97] EPRI, “*Introduction to UCA Version 2.0*”, Editorial Draft 1.0, 1997.

[W3C 98] W3C, “*Extensible Markup Language (XML) 1.0 (Second Edition)*”, W3C, 1998.