

MOBILIDADE NO MONITORAMENTO DE SUBESTAÇÕES ELÉTRICAS POR MEIO DE SERVIÇOS WEB

MOBILITY IN THE MONITORING OF ELECTRICAL SUBSTATIONS BY MEANS OF WEB SERVICES

Tiago A. Idalencio¹, Edson Luiz Padoin¹, Fabiano Salvadori¹, Maurício de Campos¹, Paulo S. Sausen¹ e Sérgio Dill¹

RESUMO

Apresenta-se, neste trabalho, um sistema para automação de subestações de energia elétrica que fazem uso de unidades remotas de aquisição de dados, pelas quais a comunicação com o controlador supervisor pode realizar-se via rádio (Wi-Fi), RS232, RS485 ou via redes do tipo *ethernet*. Com o objetivo de expandir o sistema e possibilitar o acesso por meio de equipamentos móveis, desenvolveu-se um *web service*. Assim, as funções e a operação da subestação podem ser monitoradas e controladas de qualquer ponto remoto, por equipamentos móveis que tenham JVM instaladas e acesso à *internet*. A partir dessa nova estrutura, o projeto apresenta características que, entre outras, mostram-se interessantes, como, por exemplo, mobilidade, robustez, versatilidade e confiabilidade.

Palavras-chave: monitoramento, subestações elétricas, serviços *web*.

ABSTRACT

This paper shows a system for automating electrical substations that makes use of Remote Unit Data Acquisition where the communication with the supervisor controller can be done by radio (Wi-Fi), RS232, RS485 or through ethernet networks. In order to expand the solution and allow it to be accessed by mobile devices we also developed an integrated web service. Thus, substa-

¹Grupo de Automação Industrial e Controle – GAIC.

Departamento de Tecnologia – DETEC.

Universidade Regional do Noroeste do Estado do Rio Grande do Sul - UNIJUI,

Rua do Comércio, 3000 – Bairro Universitário – CEP 98.7000-000 – Ijuí – RS - Brasil.

{tiagoal, padoin, f.salvadori, campos, sausen, dill}@unijui.edu.br

tions can be monitored and controlled remotely through any mobile device with JVM installed and Internet access. This new project structure adds new interesting functionalities and facilities like mobility, robustness, versatility and, reliability.

Keywords: *monitoring, electrical substations, web services*

INTRODUÇÃO

A automação de processos sempre esteve intimamente ligada à instrumentação e ao controle. Usualmente, a estratégia convencional utilizada para a aquisição de grandezas no controle de um processo consiste em alocar sensores próximos à fonte da grandeza de interesse. Os dados coletados pelos sensores são transmitidos para o local de processamento através de uma estrutura cabeada de comunicação de dados.

Com o objetivo de obter um funcionamento ininterrupto, ou o mais próximo disso, as empresas, cada vez mais, investem em equipamentos e recursos que facilitam e qualificam o controle das máquinas que gerenciam seu funcionamento. Por meio de recursos móveis, acessam-se essas funções de qualquer local e a qualquer momento, bastando o usuário dispor de sistemas compatíveis. Desse modo, o desenvolvimento de ferramentas de gerenciamento supervisão torna-se uma área importante a ser explorada, pois permite, principalmente, a mobilidade das pessoas nos mais diversos setores.

Os celulares tornaram-se populares e, com isso, surgiu a necessidade de desenvolverem-se aplicativos para esse equipamento. Os modelos mais recentes de celulares e também de *personal digital assistant* (PDAs) já possuem Java Virtual Machine (JVM) instalada compatível, com a tecnologia Java Mobile Edition (J2ME), possibilitando a execução de aplicações java e o acesso às informações de outros módulos processadores. Assim, podem-se desenvolver aplicativos que possibilitam a exploração e a análise dos dados armazenados no servidor que indicam o estado atual do equipamento monitorado. Os aplicativos podem ser classificados em dois grupos: no primeiro, o técnico tem o controle através da sua *interface*, celular, por exemplo, e acessa uma base de dados por meio de serviços disponíveis na *web*; no segundo, o processo pode automatizar-se com o uso de recursos de programação em base de dados que disparam ações, quando determinado evento acontece.

Neste trabalho, estende-se a solução apresentada em Salvadori et al. (2003), incorporando ao sistema o módulo de controle por meio de dispositivos móveis. Primeiramente, descreve-se o sistema de monitoramento de subestações elétricas, constituído dos subsistemas de aquisição de dados, servidor remoto e controlador super-

visório. Na sequência, apresentam-se os recusos de *Web services* e sua aplicação no contexto desta pesquisa e finaliza-se com a aplicação, objetivando o monitoramento, a distância e o gerenciamento em tempo real, a partir de equipamentos móveis.

DESCRIÇÃO DO SISTEMA

O sistema de monitoramento de subestações elétricas apresenta-se como resultado de projetos de P&D desenvolvidos em parceria com a CEEE-D/RS no âmbito dos fundos setoriais, os quais definem as tecnologias para a aquisição dos sinais de interesse, processamento e transmissão, de maneira que implementam um sistema de aquisição de dados e controle supervisão (MOREIRA, 2001; LI, 2002; GRACEK, 2001). Desses projetos, resultou o sistema apresentado em Salvadori et al. (2003).

Dentre as características do sistema, existem algumas que o diferenciam dos demais sistemas de automação de subestações: (a) facilidade de implementação de sistemas microprocessados (microcontroladores e/ou DSPs); (b) utilização de um sistema operacional, plataforma de desenvolvimento não proprietária e livre de licença; (c) baixo custo de implementação (*hardware e software*); (d) portabilidade e mobilidade; (e) facilidade de implementação de múltiplas rotas de comunicação (i.e., segurança) e (f) possibilidade de utilização de estruturas de rede já existentes.

O sistema foi dividido em quatro subsistemas, conforme apresenta a figura 1 e as seções seguintes:

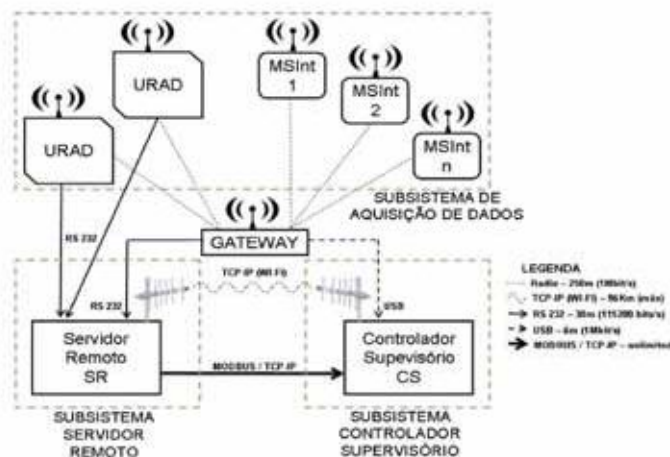


Figura 1 – Arquitetura do Sistema de Monitoramento de Subestações.

SUBSISTEMA DE AQUISIÇÃO DE DADOS

Esse subsistema forma-se a partir dos dispositivos responsáveis pela aquisição e pré-processamento dos dados coletados. Utiliza-se uma unidade de aquisição estática, projetada, inicialmente, para aquisição de três tensões e três correntes da subestação de energia, denominada Unidade Remota de Aquisição de Dados (URAD).

Apresentada na figura 2, a URAD pode comunicar-se com o servidor remoto (SR), via RS 232, RS 485 ou via rádio. No caso da utilização da RS 485, possibilita-se a conexão de múltiplas URADs para cada SR. Controla-se a URAD por um processador digital de sinal (DSP), modelo TMS320F2812 com 64 KB de memória para programas e 64 KB para dados. Opera na frequência de 150 MHz e se programa com as linguagens C e C++. O conversor analógico-digital (A/D) é de 12 bits e opera na frequência de 25 Mhz com tempo de conversão de 80 ns.

A leitura do conversor A/D pelo microcontrolador é feita a uma taxa de 10 amostras por ciclo, o que representa 600 leituras por segundo para as entradas de tensão e corrente das fases S e T. Para medir os valores de corrente e tensão em cada uma das fases e o ângulo da fase, necessita-se de 14 bytes (2 bytes cada). Assim, realizando-se 10 amostras por ciclo e considerando-se 60 ciclos por segundo, observa-se que a taxa de dados gerada é de 8,4 kB por segundo.

Os dados obtidos junto aos conversores A/D são transmitidos ao computador utilizando o protocolo de comunicação MODBUS®. Vale-se desse protocolo pela solicitação da agência financiadora do projeto, a qual define uma estrutura de mensagens compostas por *bytes*, que os dispositivos conectados devem ser capazes de reconhecer, independentemente do tipo de rede utilizada.



Figura 2 – Unidade Remota de Aquisição de Dados (URAD): sensores de tensão (LV20) e corrente (LA20); conector barramento de dados; conector da alimentação; saída RS 232 e 485; DSP TMS320F2812; transceiver TRW 2.4G®.

SUBSISTEMA SERVIDOR REMOTO – SR

Para desempenhar as funções de SR, utiliza-se uma Soekris, modelo Net4521, apresentada na figura 3. Adequada para trabalhar em lugares inóspitos e sob severas condições de funcionamento, tem como principais características: processador AMD 133 MHz, 64 MB SDRAM, *socket CompactFlash* tipo I/II, duas interfaces *FastEthernet*, uma porta serial DB9, dois *slots* PC-Card/Cardbus, e alimentação via fonte externa de 12V DC.

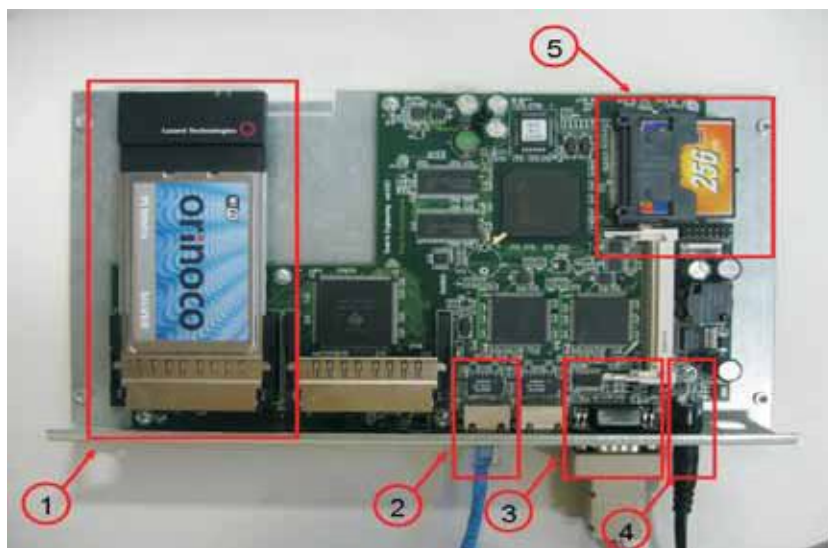


Figura 3 – Servidor remoto (SR): (1) – Placa adaptadora PCMCIA Orinoco; (2) – Conexão SR com a rede Ethernet; (3) – Conexão SR com a URAD ou com o MSInt Gateway; (4) – Alimentação do SR e (5) – Cartão de memória

Utilizou-se o sistema operacional *voyage linux*, uma distribuição *Linux* baseada em Debian e customizada para trabalhar em plataformas voltadas para sistemas embarcados. A escolha por essa distribuição deve-se ao fato de sua fácil instalação e de seu bom desempenho no *hardware* utilizado pelo projeto.

A comunicação entre o subsistema de aquisição de dados e o subsistema CS é função do SR. Apresenta-se, na figura 4, o fluxograma de transmissão do *URADServer* e do *URADClient*, *softwares* desenvolvidos em linguagem C.

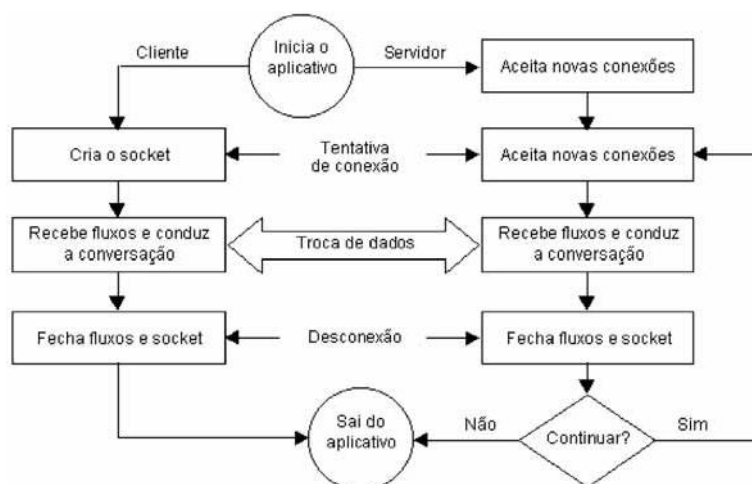


Figura 4 – Fluxograma da transmissão de dados

O módulo *URADServer* permanece residente nos SRs, recebendo dados originários da(s) URAD(s). A partir dos dados recebidos, o *URADServer* realiza ações de pré-processamento nos dados antes de transmiti-los para o CS. O módulo *URADClient* permanece residente no CS e é responsável pela recepção dos dados enviados pelo módulo(s) *URADServer*(s), descompactação e disponibilização para o módulo *eletroplot*.

No que se refere à rede, o SR desempenha o papel de *gateway*. Para desempenhar essa função foram adicionadas as seguintes opções de comunicação: porta serial (RS232/RS485), RF ou USB com possibilidades de entrada, convertendo para TCP/IP (sem fio e cabeada), utilizando ou não MODBUS (i.e., protocolo de comunicação utilizado na automação industrial). Esse conjunto de alternativas para transmissão dos dados, disponibilizada pelo SR, permite agregar segurança ao sistema de automação. Qualquer falha que exista numa forma de comunicação, imediatamente outro módulo de comunicação pode ser ativado em tempo de execução.

SUBSISTEMA CONTROLADOR SUPERVISÓRIO - CS

Gerencia-se o sistema por qualquer computador que possuir o módulo supervisor (IHM) instalado e tiver acesso à *internet*, (ZIMMER, 2006). A IHM foi delimitada, inicialmente, para apresentar os dados coletados em tempo real. O mó-

dulo supervisorio denominado *eletroplot* foi desenvolvido em linguagem C++ sobre a plataforma *linux* utilizando objetos da biblioteca gráfica QWT v. 5.0 e banco de dados MySQL.

A figura 5 apresenta o módulo EletroPlot, mas salienta-se que existem duas possibilidades de visualização dos dados: em gráficos ou em tabelas.

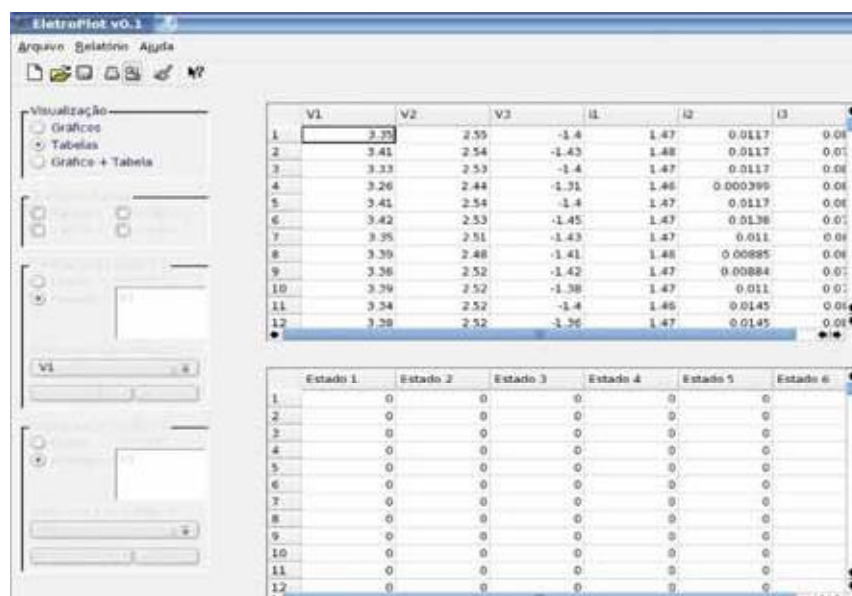


Figura 5 – Módulo EletroPlot

O WEB SERVICE ACRESCENTADO AO PROJETO

Inicialmente, precisava-se desenvolver uma aplicação servidora, responsável por buscar informações no banco de dados e disponibilizá-las aos dispositivos móveis, por meio de um *web service*. Para construir, porém, uma aplicação baseada em *web services*, faz-se primeiro necessário definir os padrões responsáveis e especificar os mecanismos para troca de mensagens, bem como as operações de pesquisar, publicar e vincular.

Foi construído um *web services*, com base no modelo cliente-servidor, em que a aplicação servidora registra o serviço dentro de um catálogo central de endereços, o UDDI. Esse registro é feito pelo provedor de serviços que publica o *web services* (HENDRICKS, 2002). Ao implementar a aplicação cliente, o consumidor pesquisa a existência de um determinado serviço e, caso seja encontrado,

realiza-se o vínculo com o provedor de serviços por mensagens SOAP sobre o protocolo de transporte, que pode ser o HTTP.

Para suportar os *web services*, é necessária a existência de um container, como o que está disponível em ferramentas como Sun Java System Application Server - SJSAS (2007), entre outras. No momento, desenvolvem-se testes com os servidores Tomcat e o SJSAS, ressaltando que o segundo, devido à linguagem de programação utilizada ser JAVA (DEITEL, 2005) e o *web service* ser projetado para ser abrigado em um container J2EE, apresenta algumas vantagens.

Para o desenvolvimento do *web service* e sua publicação também precisa-se definir uma ferramenta. Para tanto, analisaram-se duas alternativas, o *Java Web services Developer Pack* (2007) e o *Apache-Axis* (2007).

O JWSDP é um pacote da Sun Microsystems específico para a criação de *Web services* com Java. Todas as tecnologias utilizadas nesse pacote baseiam-se no modelo de comunicação RPC, o qual fornece o suporte de comunicação entre as aplicações cliente e servidor. O pacote constitui-se de um conjunto de APIs que possuem recursos e funcionalidades que permitem construir e distribuir aplicações *web*, das quais destacam-se o *wscompile* e *wsdeploy* (2007).

Já o *Apache-axis* é um *framework* de código aberto, baseado na plataforma Java e no padrão XML, também utilizada para construção e registro de *web services*, no padrão SOAP (Apache). Optou-se por utilizar o JWSDP para o desenvolvimento da aplicação distribuída, pelo fato dele permitir a execução do *web service* em diversas plataformas, por ser *open source* e, principalmente, pela ferramenta API JAX-RPC, que é utilizada para comunicação, tanto pelo servidor quanto pelo cliente de forma muito transparente, não havendo a necessidade de o programador se preocupar em implementar classes para a comunicação.

Deve-se destacar que, para acessar um *web service*, uma aplicação cliente pode desenvolver-se de três formas: *stubs*, *dynamic proxies* e *DII – dynamic invocation interface*. Essas formas determinam o grau de acoplamento e a dificuldade de programação. A utilização do JAX-RPC, possibilita as três formas de acesso. Com *stubs*, utilizam-se classes geradas por ferramentas durante o desenvolvimento do *web service*. Com *dynamic proxies*, as classes criam-se dinamicamente com base no *WSDL - Web service Definition Language*. E, por último, com *DII*, descobrem-se e invocam-se os métodos em tempo de execução.

Outro ponto a ser definido reside na transformação do *software* Java para *WSDL*, visto que pode realizar-se de duas formas, *Java-2-WSDL* ou *WSDL-2-Java*. A primeira é mais fácil, pois geralmente há as classes previamente definidas. Para realizar a transformação existem várias ferramentas disponíveis, das quais

foram analisadas a *Axis* e a *XStream*, ambas também disponíveis para os sistemas operacionais *windows* e *linux*. A primeira trabalha com *Tomcat(htdocs)* e, quando utilizada na forma de *WSDL-2-Java*, gera *skeleton* para o servidor e um *stub* para o cliente. A segunda, *XStream*, apresenta facilidade de utilização para realizar o mapeamento de objetos Java para documentos XML e vice-versa.

O *web service* criado disponibilizará informações aos clientes, rodando em *palms* ou celular e buscará informações persistidas no banco de dados. Os clientes acessam o *web service*, rodando em um *browser* do tipo mini-opera, no aparelho utilizado como meio de acesso.

Dentre as tecnologias de celular existentes, destacam-se AMPS, TDMA, CDMA, GSM, porém somente as duas últimas podem utilizar conexão de dados e executar aplicativos J2ME (MUCHOW, 2006). Nesse sentido, vale-se de um aparelho de celular modelo Nokia 6265, do qual se efetivou o desbloqueio, para que seja possível rodar tais aplicações. A figura 6 apresenta o celular utilizado no projeto com o *browser* mini-opera e também com uma conexão *ssh* com o servidor remoto(SR).

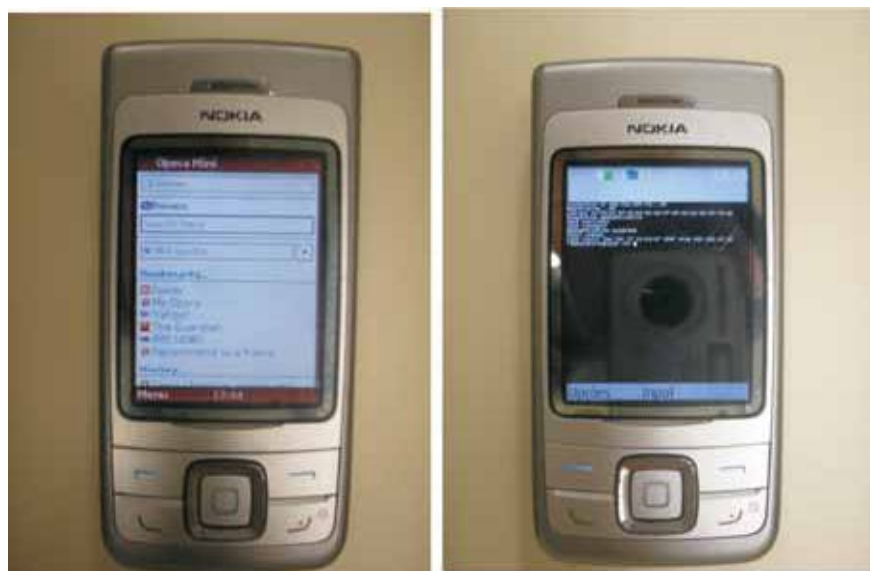


Figura 6 – Dispositivo móvel com browser Mini-Opera e com conexão SSH.

Para o desenvolvimento da aplicação cliente, empregaram-se as ferramentas *Sun Java Wireless Toolkit* que consistem de um *framework J2ME* e *easy eclipse*. Para acessar o *web service* através do celular, usa-se o *browser* mini-opera, também desenvolvido em J2ME.

CONCLUSÃO

A grande maioria dos sistemas de automação de subestações (NETO, 2000) utiliza comunicações baseadas em cabos tipo par trançado, fibra ótica ou comunicação serial, padrões RS232 e RS485 para tráfego de dados e comandos, outros até apresentam baseadas em controladores lógicos programáveis, ambos com um alto custo de implementação, principalmente quando se utilizam entradas analógicas, as quais, normalmente, são multiplexadas.

Tais soluções são, geralmente, baseadas em sistemas dedicados e soluções proprietárias que apresentam alto custo de instalação e manutenção. Isso torna as alterações e adequações do projeto/sistema uma tarefa trabalhosa e de alto custo financeiro.

Neste trabalho apresentou-se os resultados obtidos no desenvolvimento de um *web service*, aplicado à automação de subestações. Com uma nova abordagem, ele permite o acesso aos dados em sistemas de automação através da *web* por meio de dispositivos móveis. Sua principal contribuição consiste no aprimoramento do sistema desenvolvido, possibilitando um gerenciamento em tempo real do sistema, a partir de um *web service*, e no desenvolvimento das características de mobilidade, robustez, versatilidade e confiabilidade.

AGRADECIMENTOS

Os autores agradecem o apoio financeiro fornecido da Companhia Estadual de Energia Elétrica do Rio Grande do Sul (CEEE-RS).

REFERÊNCIAS

AXIS Apache Documentation. Disponível em: <http://ws.apache.org/axis/java/index.html>

DEITEL, H. M. **Java Como Programar**, 6. ed. Pearson Education. São Paulo, 2005.

GRACEK, D; GEYNISMAN, O; PROUDFOOT, D; MINNIK, K. **Migrating From Scada To Automation, Transmission And Distribution Conference And Exposition**. IEEE/PES, v. 1, p. 343-348, 2001.

HENDRICKS, Mack. **Java Web services**. Rio de Janeiro: Alta Books, 2002.

JbossWS. Disponível em <http://www.jboss.org/wiki/Wiki.jsp?page=JBossWS>

Java Web services Developer Pack. Disponível em <http://java.sun.com/webservices/jwsdp/index.jsp>

LI, D.; SERIZAWA Y.; Kiuchi M. **Concept Design For A Web-Based Supervisory Control And Data-Acquisition (SCADA) System**, Transmission And Distribution Conference And Exhibition: Asia Pacific. IEEE/PES, v. 1, p. 32-36, 2002.

MOREIRA, J.A.C.; CARVALHO, A.R.C.D; ALMEIDA, P.C; PEREIRA, A. C. **Bauru Security Control Scheme**, IEEE Porto Power Tech Conference, Porto, Portugal, 2001.

MUCHOW, John W. **Core J2ME: Tecnologia e MIDP**. The Sun Microsystems Press. Pearson Education. São Paulo, 2006.

NETO, G. L. de Castro; KANASHIRO, A.G.; MALAGODI, C.V.S. **Desenvolvimento e Implantação de Sistema de Monitoramento de Subestações**, Eletricidade Moderna, p. 286-297, 2000.

SALVADORI, F.; SAUSEN, P.S.; HARTMANN, L.V.; CAMPOS, de M.; PADDOIN, E.L. and Leandro, G.V., **Acquisition And Transmission Data Monitoring System Applied To Energy Substation**, Industrial Informatics. INDIN 2003. Proceedings. IEEE International Conference on 21-24 Aug. 2003 p. 60 – 64, Banff, Alberta, Canada.

Sun Technology. Disponível em: <http://java.sun.com/>

Tomcat, Jacarta- Tomcat. Disponível em: <http://tomcat.apache.org/>

WebSphere Application Server Zone. Disponível em: <http://www-128.ibm.com/developerworks/websphere/zones/was/bigpicture.html>

ZIMMER, D.; RHODES, D. **Human Machine Interfaces, Industry Applications Magazine**, IEEE v. 12, Issue 2, March-April, p. 29 – 35, 2006.