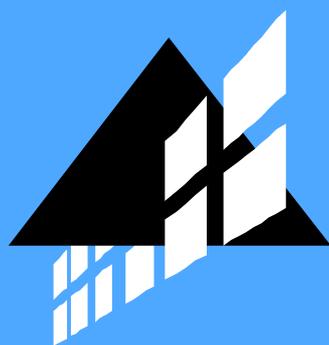


---

# Manual de Utilização WebView MT7000

Rev. D 04/2007  
Cód. Doc.: MU299043



altus

---

---

Nenhuma parte deste documento pode ser copiada ou reproduzida sem o consentimento prévio e por escrito da Altus Sistemas de Informática S.A., que se reserva o direito de efetuar alterações sem prévio comunicado.

Conforme o Código de Defesa do Consumidor vigente no Brasil, informamos a seguir, aos clientes que utilizam nossos produtos, aspectos relacionados com a segurança de pessoas e instalações.

Os equipamentos de automação industrial fabricados pela Altus são robustos e confiáveis devido ao rígido controle de qualidade a que são submetidos. No entanto, equipamentos eletrônicos de controle industrial (controladores programáveis, comandos numéricos, etc.) podem causar danos às máquinas ou processos por eles controlados em caso de defeito em suas partes e peças ou de erros de programação ou instalação, podendo inclusive colocar em risco vidas humanas.

O usuário deve analisar as possíveis conseqüências destes defeitos e providenciar instalações adicionais externas de segurança que, em caso de necessidade, sirvam para preservar a segurança do sistema, principalmente nos casos da instalação inicial e de testes.

É imprescindível a leitura completa dos manuais e/ou características técnicas do produto antes da instalação ou utilização do mesmo.

A Altus garante os seus equipamentos conforme descrito nas Condições Gerais de Fornecimento, anexada às propostas comerciais.

A Altus garante que seus equipamentos funcionam de acordo com as descrições contidas explicitamente em seus manuais e/ou características técnicas, não garantindo a satisfação de algum tipo particular de aplicação dos equipamentos.

A Altus desconsiderará qualquer outra garantia, direta ou implícita, principalmente quando se tratar de fornecimento de terceiros.

Pedidos de informações adicionais sobre o fornecimento e/ou características dos equipamentos e serviços Altus devem ser feitos por escrito. A Altus não se responsabiliza por informações fornecidas sobre seus equipamentos sem registro formal.

### DIREITOS AUTORAIS

Série Ponto, MasterTool, Quark, ALNET e WebPlc são marcas registradas da Altus Sistemas de Informática S.A.

IBM é marca registrada da International Business Machines Corporation.

# Sumário

<b>1. INTRODUÇÃO.....</b>	<b>5</b>
<b>O WebView .....</b>	<b>5</b>
<b>Os Applets .....</b>	<b>5</b>
<b>As Páginas HTML.....</b>	<b>6</b>
<b>Documentos Relacionados a este Manual.....</b>	<b>7</b>
<b>Inspeção Visual.....</b>	<b>8</b>
<b>Suporte Técnico.....</b>	<b>8</b>
<b>Mensagens de Advertência Utilizadas neste Manual .....</b>	<b>8</b>
<b>2. DESCRIÇÃO TÉCNICA.....</b>	<b>9</b>
<b>Características Técnicas.....</b>	<b>9</b>
Características Gerais.....	9
Interfaces Ethernet com Servidor Web.....	9
WebView.....	10
<b>Desempenho e Limites do Software .....</b>	<b>11</b>
Desempenho .....	12
<b>Dados para Compra.....</b>	<b>13</b>
Itens Integrantes .....	13
Código do Produto .....	13
Produtos Relacionados .....	13
<b>3. PROGRAMAÇÃO.....</b>	<b>14</b>
<b>Criando Páginas HTML .....</b>	<b>14</b>
<b>Criando Páginas com Applets WebView.....</b>	<b>15</b>
<b>Configurando os Parâmetros dos Applets .....</b>	<b>17</b>
<b>Enviando uma Página HTML para o WebGate.....</b>	<b>19</b>
<b>Apresentado a Página no Browser.....</b>	<b>20</b>
Verificando a Comunicação.....	20
Apresentando a Página .....	21
Informando Usuário e Senha .....	22
Tamanho do Texto .....	23
<b>4. UTILIZAÇÃO.....</b>	<b>24</b>
<b>Características Gerais.....</b>	<b>24</b>
Parâmetros dos Applets .....	24
Operandos do CP .....	24
Cor nos Applets.....	24
Tamanho dos Applets.....	25
Frequência de atualização dos Applets.....	26
Escala de Engenharia .....	26
<b>Banco de Applets.....</b>	<b>27</b>
Driver 28	
AnalogInput .....	30
DigitalInput.....	31
AnalogOutput .....	33
DigitalOutput .....	34

---

StringAnimation.....	35
ColorAnimation .....	37
ImageAnimation .....	39
PushButton.....	41
CheckBox .....	42
RadioButton.....	45
ComboBox.....	47
ListBox .....	50
TimeStamp.....	53
BarGraph .....	54
PenChart.....	56
EventSummary.....	58
AlarmSummary.....	63
Variável Local.....	68
Receita.....	69
<b>5. INSTALAÇÃO.....</b>	<b>73</b>
<b>Instalando o WebView .....</b>	<b>73</b>
O CDROM de Distribuição .....	73
Requisitos de Hardware e Software .....	74
Iniciando a Instalação.....	74
Modo Demonstração .....	75
O que é Instalado ?.....	75
Desinstalação .....	76
Instalando o Acrobat Reader.....	76
<b>Instalando o Plug-In Java .....</b>	<b>76</b>
Verificando o Plug-In Java .....	77
<b>Instalando o Programa de FTP.....</b>	<b>78</b>
<b>Instalando o Editor HTML.....</b>	<b>78</b>
<b>Configurando o WebGate.....</b>	<b>79</b>
Carregando a Biblioteca de Applets no WebGate.....	79
<b>6. MANUTENÇÃO .....</b>	<b>81</b>
<b>Parâmetros com Valores Inválidos.....</b>	<b>81</b>
<b>Falta de Parâmetros Obrigatórios.....</b>	<b>82</b>
<b>Problemas na Comunicação com o CP .....</b>	<b>83</b>
<b>Problemas na Comunicação com o WebGate.....</b>	<b>83</b>
<b>Congelamento do Browser.....</b>	<b>83</b>
<b>A Janela Java Console .....</b>	<b>83</b>
<b>Erros de Execução nos Applets.....</b>	<b>85</b>
<b>Problemas na Máquina Virtual Java.....</b>	<b>85</b>
<b>7. ANEXO I: EXEMPLOS DE APLICAÇÃO .....</b>	<b>86</b>
<b>Monitoração de Operandos do CP .....</b>	<b>86</b>
<b>Controle de Iluminação.....</b>	<b>86</b>
<b>Exaustão de Gases.....</b>	<b>88</b>
<b>Controle de Nível de Tanque .....</b>	<b>88</b>
<b>Automação Predial.....</b>	<b>90</b>
<b>8. ANEXO II: TABELA DE CORES .....</b>	<b>91</b>
<b>9. GLOSSÁRIO .....</b>	<b>92</b>

---



# 1. Introdução

O WebView é um sistema de supervisão e controle embutido na Interface Ethernet Industrial PO7091. O produto permite a elaboração de telas HTML incluindo recursos para acesso de leitura e escrita a operandos de um controlador programável (CP) Altus. As telas são carregadas no PO7091, formando um sistema de supervisão e controle em tempo real do processo. As telas são visualizadas em browsers Internet Explorer tradicionais em qualquer microcomputador ligado a internet ou intranet do PO7091.

O WebView permite criar telas HTML para representar o processo incluindo recursos tais como monitoração e escrita de valores digitais e analógicos do CP, gráficos de barra ou curvas de tendência com o valor instantâneo ou o histórico de uma variável do CP, animação da tela com cores, imagens GIF ou texto configuráveis, apresentação de alarmes e alertas, dentre outros.

Toda a tecnologia já existente na criação de páginas HTML para a Internet pode ser utilizada com o WebView. As telas podem ser criadas com Editores HTML disponíveis no mercado. As monitorações, gráficos, animações e demais recursos do WebView podem ser facilmente incluídos a partir destes editores, através de componentes parametrizáveis denominados *applets*. Não é necessário nenhum conhecimento prévio de HTML ou qualquer outra linguagem de programação.

## O WebView

O WebView é um software composto por um banco de applets utilizados no desenvolvimento de páginas HTML (Hypertext Markup Language). O WebView pode ser utilizado com um dos seguintes servidores Web da Altus:

- PO7091: Interface Ethernet Industrial
- PO9900: WebGate
- PO9901: WebGate Plus

O WebView destina-se ao desenvolvimento de aplicações de supervisão e controle, onde navegadores para páginas HTML são utilizados como IHM (Interface Homem Máquina) para equipamentos de automação industrial com suporte XML (Extensible Markup Language) sobre TCP/IP (Transmission Control Protocol/Internet Protocol), tipicamente para utilização numa rede local em ambiente de Intranet.

Atualmente qualquer controlador programável Altus das séries AL-2000, Piccolo, Quark ou Grano possui suporte XML sobre TCP/IP através do gateway PO9900 WebGate ou PO9901 WebGate Plus. Na Série Ponto também pode ser utilizado o módulo PO7091 Interface Ethernet Industrial.

## Os Applets

Applets são programas desenvolvidos em linguagem Java que são carregados em um Servidor Web e executados pelo browser Web na máquina local. Applets dependem de browsers que os suportem para serem executados.

No contexto WebView, os applets são componentes Java que podem se comunicar com controladores programáveis Altus. Através destes, pode-se facilmente criar páginas HTML para apresentação e/ou edição de valores de operandos dos CPs em browsers tradicionais. Isto torna possível que um browser seja utilizado no papel de uma IHM (Interface Homem Máquina) para CPs Altus.

Os applets do WebView possuem, tipicamente, uma grande quantidade de parâmetros que podem ser usados para configuração dos mesmos, mas, após sua inclusão em uma página HTML, basta a definição de poucos parâmetros obrigatórios para torná-lo funcional.

Cada applet do WebView corresponde a uma aplicação completa capaz de fazer a aquisição de informações dos CPs, apresentação destas informações em formato de texto ou gráfico, entrada de dados, envio de comandos e valores.

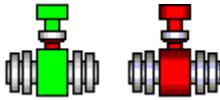
## As Páginas HTML

As páginas HTML do WebView podem ser desenvolvidas em editores HTML já existentes no mercado, tais como o FrontPage Express da Microsoft ou DreamWeaver da MacroMedia. Pode-se utilizar os recursos existentes nestes editores e adicionar os applets WebView desejados. As páginas são enviadas para o WebGate, juntamente com os applets WebView, e podem ser visualizadas no browser Internet Explorer.

O WebView permite construir páginas HTML com as seguintes funcionalidades :

- monitoração e apresentação no browser de valores analógicos e digitais de operandos do CP
- edição de valores analógicos e digitais de operandos do CP, com limites mínimo e máximo configuráveis
- conversão de escala de valores analógicos para unidades de engenharia
- envio de comandos digitais para o CP através de botões típicos configuráveis (exemplo: botão Ligar, botão Desligar)
- apresentação e edição de valores do CP em controles típicos do Windows, tais como ComboBox, ListBox, CheckBox e RadioButton
- animação de tela com cores, imagens e texto configuráveis conforme valores de operandos do CP
- apresentação de BarGraph e Gráficos X-Y a partir de valores de operandos do CP
- apresentação de alarmes e eventos a partir de valores de operandos do CP
- atualização dinâmica de todos os applets da tela com uma frequência configurável individualmente
- cor ou imagem de fundo, tipo e tamanho do fonte configuráveis individualmente
- concentração das comunicações em um único applet para melhorar o desempenho e o tempo de apresentação das telas
- agrupamento de parâmetros em arquivos de receita para apresentar uma mesma tela HTML para diferentes operandos do CP

Os applets WebView e seus parâmetros estão descritos detalhadamente neste manual. A tabela a seguir mostra alguns exemplos destes applets :

<b>AnalogInput</b>	
<b>ImageAnimation</b>	
<b>PushButton</b>	
<b>ColorAnimation</b>	
<b>StringAnimation</b>	

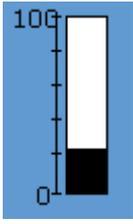
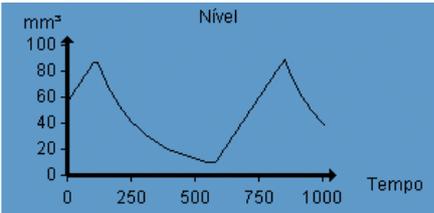
BarGraph										
CheckBox	<input checked="" type="checkbox"/> Simular Nível									
RadioButton	<input checked="" type="radio"/> Manual <input type="radio"/> Auto									
ComboBox	Clorato Clorato Tanque Soda Tanque HCl									
TimeStamp	06/02/98 01:14:01									
PenChart										
EventSum	<table border="1"> <thead> <tr> <th>Horas(hh:mm)</th> <th>Evento(s)</th> </tr> </thead> <tbody> <tr> <td>19:21</td> <td>Nível Tanque A Muito Alto</td> </tr> <tr> <td>19:23</td> <td>Nível Tanque A Muito Baixo</td> </tr> <tr> <td>20:05</td> <td>Nível Tanque B Muito Alto</td> </tr> </tbody> </table>	Horas(hh:mm)	Evento(s)	19:21	Nível Tanque A Muito Alto	19:23	Nível Tanque A Muito Baixo	20:05	Nível Tanque B Muito Alto	
Horas(hh:mm)	Evento(s)									
19:21	Nível Tanque A Muito Alto									
19:23	Nível Tanque A Muito Baixo									
20:05	Nível Tanque B Muito Alto									
AlarmSum	<table border="1"> <thead> <tr> <th>ACK</th> <th>Horas(hh:mm)</th> <th>Evento(s)</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/></td> <td>21:56</td> <td>Bomba A ligada</td> </tr> <tr> <td><input type="checkbox"/></td> <td>19:21</td> <td>Caldeira C1 nível HH</td> </tr> </tbody> </table>	ACK	Horas(hh:mm)	Evento(s)	<input checked="" type="checkbox"/>	21:56	Bomba A ligada	<input type="checkbox"/>	19:21	Caldeira C1 nível HH
ACK	Horas(hh:mm)	Evento(s)								
<input checked="" type="checkbox"/>	21:56	Bomba A ligada								
<input type="checkbox"/>	19:21	Caldeira C1 nível HH								

Tabela 1-1. Exemplos de Applets WebView

## Documentos Relacionados a este Manual

Para obter informações adicionais sobre o WebView e servidores Web, podem ser consultados outros documentos (manuais e características técnicas) além deste. Estes documentos encontram-se disponíveis em sua última revisão em [www.altus.com.br](http://www.altus.com.br).

Cada produto possui um documento denominado Característica Técnica (CT), onde encontram-se as características do produto em questão. Adicionalmente o produto pode possuir Manuais de Utilização (o código do manuais são citados na CT).

Aconselha-se os seguintes documentos como fonte de informação adicional:

- Características Técnicas do WebView
- Manual de Utilização do WebGate - PO9900
- Manual de Utilização do WebGate Plus - PO9901

- Manual de Utilização Interface Industrial Wthernet - PO7091
- MasterTool Programming Manual de Utilização - MT4100

### Inspeção Visual

Antes de proceder à instalação, é recomendável fazer uma inspeção visual cuidadosa dos equipamentos, verificando se não há danos causados pelo transporte. Verifique se todos os componentes de seu pedido estão em perfeito estado. Em caso de defeitos, informe a companhia transportadora e o representante ou distribuidor Altus mais próximo.

É importante registrar o número de série de cada equipamento recebido, bem como as revisões de software, caso existentes. Essas informações serão necessárias caso se necessite contatar o Suporte Técnico da Altus.

### Suporte Técnico

Para entrar em contato com o Suporte Técnico da Altus em São Leopoldo, RS, ligue para +55-51-589-9500. Para conhecer os centros de Suporte Técnico da Altus existentes em outras localidades, consulte nosso site ([www.altus.com.br](http://www.altus.com.br)) ou envie um email para [altus@altus.com.br](mailto:altus@altus.com.br).

Se o equipamento já estiver instalado, tenha em mãos as seguintes informações ao solicitar assistência:

- os modelos dos equipamentos utilizados e a configuração do sistema instalado.
- o número de série da UCP.
- a revisão do equipamento e a versão do software executivo, constantes na etiqueta afixada na lateral do produto.
- informações sobre o modo de operação da UCP, obtidas através do programador MasterTool.
- o conteúdo do programa aplicativo (módulos), obtido através do programador MasterTool.
- a versão do programador utilizado.

### Mensagens de Advertência Utilizadas neste Manual

Neste manual, as mensagens de advertência apresentarão os seguintes formatos e significados:

**PERIGO:**

Relata causas potenciais que se não observadas levam a danos à integridade física e saúde, patrimônio, meio ambiente e perda da produção.

**CUIDADO:**

Relata detalhes de configuração, aplicação e instalação que devem ser seguidos para evitar condições que possam levar a falha do sistema e suas consequências relacionadas.

**ATENÇÃO:**

Indicam detalhes importantes de configuração, aplicação ou instalação para obtenção da máxima performance operacional do sistema.

## 2. Descrição Técnica

Este capítulo apresenta as características técnicas do WebView, abordando as partes integrantes do sistema, sua arquitetura e características gerais.

### Características Técnicas

#### Características Gerais

MT7000 – WebView	
<b>Plataforma</b>	PC
<b>Sistema Operacional</b>	<ul style="list-style-type: none"> <li>Windows 98 (2nd edition)</li> <li>Windows 2000 (service pack 3)</li> <li>Windows XP Professional (service pack 1)</li> </ul>
<b>Interfaces Ethernet Compatíveis</b>	<ul style="list-style-type: none"> <li>PO7091: Interface Ethernet Industrial</li> <li>PO9901: Webgate Plus</li> <li>PO9900: Webgate</li> </ul>
<b>Browser Compatível (1)</b>	Internet Explorer 5.5 ou 6.0
<b>Plug-In Java para Browser</b>	Java 2 Run Time Environment SE versão 1.4.2
<b>Formatos suportados</b>	HTML, XML, JAVA, JAVA SCRIPT, VBScript, FLASH e outros
<b>Comandos XML disponíveis</b>	Leitura e escrita de operandos do CP
<b>Operandos do CP</b>	Operandos do tipo %M, %A, %D, %TM e %TD
<b>Monitoração Cíclica</b>	Sim
<b>Softwares Externos Necessários</b>	<ul style="list-style-type: none"> <li>Editor HTML</li> <li>FTP</li> <li>Browser Internet Explorer</li> </ul>
<b>Parametrização dos Applets</b>	Através de qualquer Editor HTML
<b>Comunicação com Servidor Web</b>	Através de qualquer programa de FTP
<b>Limites</b>	Ver seção mais adiante neste capítulo

(1) o uso de uma versão diferente do Internet Explorer ou de um outro browser pode acarretar em divergências de compatibilidade e prejudicar o funcionamento e a performance do produto, o qual foi projetado para operar com as versões indicadas do browser.

#### Interfaces Ethernet com Servidor Web

A Interface Ethernet Industrial (PO7091) e as interfaces de comunicação WebGate (PO9900) e WebGate Plus (PO9901) permitem que controladores programáveis Altus das séries Piccolo, Quark, AL-2000, Ponto e Grano sejam conectados a uma rede Ethernet. Simultaneamente, é possível acessar estes equipamentos pela interface Web, onde estão disponíveis serviços WWW e FTP, garantindo o suporte para acesso via browser. Esta comunicação se dá através do protocolo de aplicação HTTP com o padrão XML.

Estes equipamentos dispõem do serviço de Servidor Web no formato HTML, permitindo a criação de páginas dinâmicas com dados de processo. Para tanto, possuem um sistema de arquivos em memória não volátil, onde são armazenados os arquivos HTML, animações e imagens.

#### ATENÇÃO:

Os três produtos WebGate (PO9900), WebGate Plus (PO9901) e Interface Ethernet Industrial (PO7091) possuem estes recursos de serviços WWW e FTP. **Este manual utiliza apenas o termo WebGate para se referenciar a qualquer um destes produtos.**

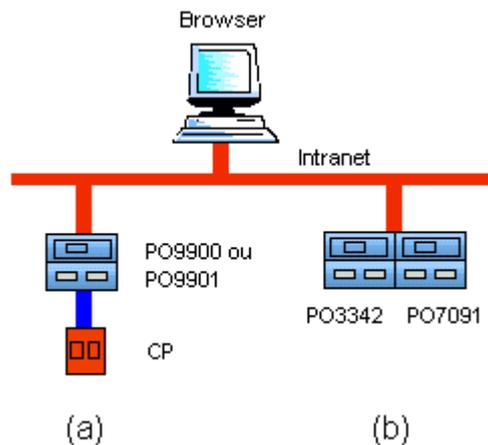
A construção de sistemas de supervisão para o WebGate não está limitada ao HTML. É possível também utilizar qualquer outra tecnologia que não dependa de processamento no Servidor Web, tais como XML, JAVA, JAVA SCRIPT, VBScript e FLASH, dentre outras .

A carga de novas páginas é feita via Intranet ou Internet, utilizando o protocolo FTP.

Maiores informações sobre os produtos WebGate (PO9900), WebGate Plus (PO9901) e Interface Ethernet Industrial (PO7091) podem ser obtidas nos manuais correspondentes.

### WebView

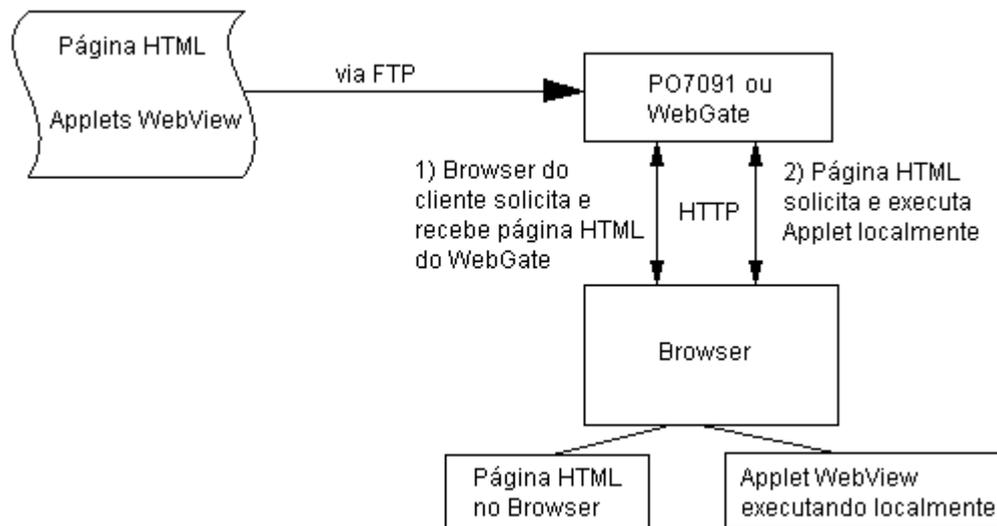
A arquitetura de utilização do WebView é constituída por uma rede Intranet ou Internet com um microcomputador executando um browser e um PO7091 em um barramento Ponto do CP PO3342 (Figura 2-1b) ou um WebGate conectado a um CP (Figura 2-1a).



**Figura 2-1: Arquitetura de Utilização do WebView**

Conforme já mencionado, o WebView é formado por um banco de applets Java que podem ser incluídos em quaisquer telas HTML.

Os applets WebView são pequenos programas que ficam armazenados no Servidor Web, dentro de um arquivo denominado “**webview.jar**”. Este arquivo é carregado automaticamente para as máquinas dos clientes quando estes acessam páginas HTML que fazem referências aos applets. A Figura 2-2 mostra como é feito o carregamento dos applets para o browser na máquina do cliente.



**Figura 2-2: Carregamento de Applets para a Máquina do Cliente**

As páginas HTML são carregadas no WebGate através da Intranet ou Internet utilizando o protocolo FTP disponível no WebGate.

Para a visualização dos applets, utiliza-se o navegador do microcomputador que, via o serviço HTTP, acessa a interface Web do WebGate e disponibiliza, através do mesmo, páginas HTML que podem monitorar e escrever valores em operandos do CP.

## Desempenho e Limites do Software

O WebView possui as seguintes limites de software:

	Limites
<b>Quantidade de applets por página HTML</b>	Sem Driver: 10 applets Com Driver: 100 applets O Applet Driver é um concentrador de comunicações que visa otimizar a utilização do sistema
<b>Quantidade e Tamanho de páginas HTML</b>	Limite de memória do PO7091, PO9901 ou PO9900. Tipicamente 150 kbytes. Nos PO7091 ou PO9901 pode-se acrescentar expansão com via m card PO8541 com capacidade de 128Mb
<b>Monitorações Cíclicas</b>	Cinco applets a cada 1 segundo Dez applets a cada 2 segundos Com mais applets, ver limites do Driver
<b>Limites do Driver: Capacidade</b>	Até 16 blocos de operando com 64 operandos cada
<b>Limites do Driver: Quantidade de Operandos Monitorados Simultaneamente</b>	O limite é de 64 operandos por segundo, com um ou mais blocos, por exemplo: - Um bloco de 64 operandos a cada 1 segundo - Dois blocos de 32 operandos a cada 1 segundo - Três blocos de 42 operandos a cada 2 segundos - Quatro blocos de 48 operandos a cada 3 segundos - Cinco blocos de 64 operandos a cada 5 segundos
<b>Menor período de monitoração</b>	1 segundo
<b>Maior período de monitoração</b>	5 minutos
<b>Quantidade de browsers abertos simultaneamente:</b>	- No mesmo microcomputador: 1 browser - Em microcomputador diferentes: até 3 browsers, considerando que o desempenho diminui a cada novo browser fazendo monitorações cíclicas.

## Desempenho

O desempenho do WebView em termos de tempo de resposta na apresentação de uma tela depende do 1) tempo de resposta do Servidor Web utilizado (PO7091, PO9901 ou PO9900) e também do 2) tempo da comunicação entre o CP e Servidor Web. Os principais fatores que influenciam neste desempenho são mostrados na tabela a seguir.

	Descrição
<b>Tráfego na rede ethernet do Servidor Web</b>	Em qualquer equipamento conectado a uma rede ethernet o desempenho poderá ser influenciado pelo tráfego na rede. Recomenda-se que o WebView seja utilizado em uma intranet ou internet com um tráfego em condições aceitáveis de utilização. Podem ser utilizados switches para isolar o tráfego em determinados segmentos para melhorar as condições gerais de utilização.
<b>Número de Browsers abertos e Conexões simultâneas com o servidor</b>	O desempenho diminui com a quantidade de conexões simultâneas abertas com o Servidor Web. Cada browser aberto representa no mínimo uma conexão. O applet Driver representa outra conexão. Cada applet de uma página HTML que não utiliza o Driver também representa mais uma conexão, sendo assim extremamente recomendável que se utilize o applet Driver em páginas com mais de 5 applets.
<b>Tempo de resposta do CP</b>	O tempo de resposta na comunicação entre o Servidor Web e o CP é um dos principais fatores que limitam o desempenho do WebView. Cada uma das conexões simultâneas abertas precisa esperar que o CP responda ao Servidor Web a solicitação feita pelo applet no browser.  No caso do PO7091 a comunicação com o CP é feita pelo barramento GBL, que é mais rápida do que no PO9901 ou PO9900, nos quais esta comunicação é feita pelo canal serial RS-232 do CP.
<b>Tempo de varredura do programa aplicativo do CP</b>	Cada solicitação feita pelo Servidor Web ao CP é respondida somente no final do ciclo de varredura do programa aplicativo no CP. Isto significa que programas aplicativos com um tempo de varredura maior levam mais tempo para responder e diminuem o desempenho geral do WebView.  O tempo de varredura do programa aplicativo do CP pode ser visualizado através do comando Informações no software de programação dos CPs, MasterTool.
<b>Tempo de carga do arquivo <i>webview.jar</i></b>	Quando um browser apresenta uma tela do WebView, o arquivo <i>webview.jar</i> é carregado do Servidor Web para o browser. Esta carga é relativamente lenta, tipicamente 3 ou 4 segundos, mas é feita uma única vez para cada browser aberto. Quando se troca de tela num mesmo browser, este arquivo não é recarregado.  Recomenda-se que um sistema desenvolvido com o WebView utilize uma tela de abertura na qual aconteça a carga do <i>webview.jar</i> , para que durante a navegação nas demais telas do sistema este tempo de carga não se faça presente.

## Dados para Compra

### Itens Integrantes

A embalagem do produto contém um CDROM com os seguintes itens:

- Software WebView
- Manual de Utilização do WebView (PDF)
- Contrato de Licença de Software Altus

### Código do Produto

O seguinte código deve ser usado para compra do produto:

Código	Denominação
MT7000	WebView: Software de Supervisão e Controle via Web

### Produtos Relacionados

Os seguintes produtos devem ser adquiridos separadamente quando necessário:

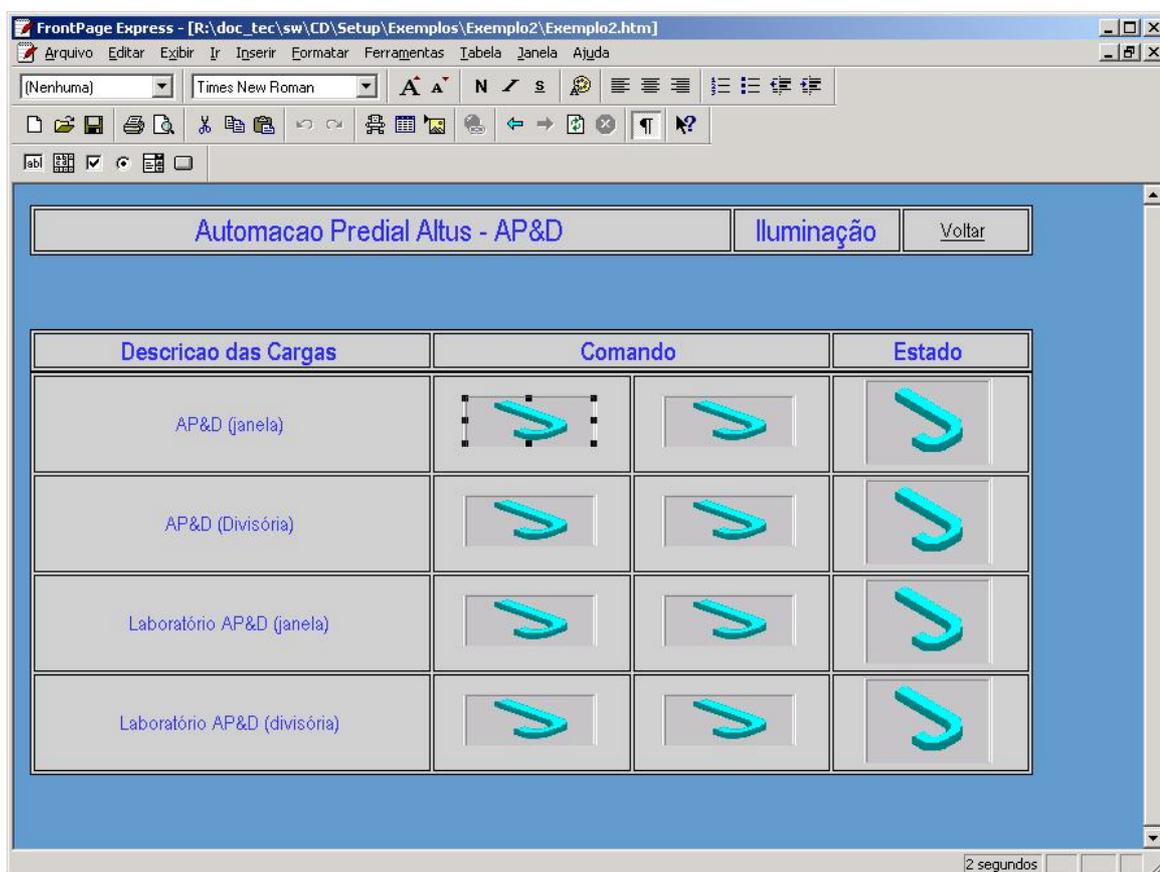
Código	Denominação
PO9900	WebGate
PO9901	WebGate Plus
PO7091	Interface Ethernet Industrial
PO8541	M Card 128 Mb
MT4100	MasterTool Programming

## 3. Programação

### Criando Páginas HTML

Para se criar páginas HTML, pode-se usar um editor de texto simples qualquer, como o NotePad, ou um editor específico para a criação de páginas HTML, como o FrontPage, da Microsoft ou DreamWeaver, da Macromedia.

Nos passos a seguir descrevem-se exemplos de criação de páginas para o WebView utilizando-se o Editor HTML MS FrontPage 97 Express. Sua tela principal é mostrada na figura a seguir, onde está sendo editada a página HTML do exemplo de Controle de Iluminação, disponível no CDROM do produto:



**Figura 3-1. Tela principal do MS FrontPage 97 Express**

Nesta tela estão incluídos 12 applets, cada um representado pelo símbolo , contendo seu próprio conjunto de parâmetros.

O nome dos arquivos HTML deve ter no máximo oito caracteres de comprimento e a extensão deve ser “HTM” e não de “HTML” (três caracteres em vez de quatro).

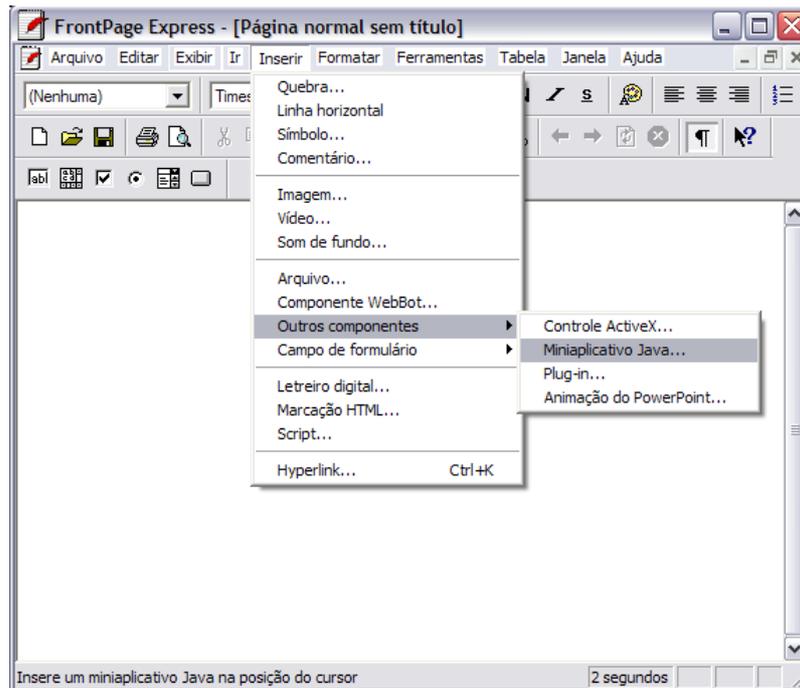
**ATENÇÃO:**

os editores HTML mais modernos possuem o recurso de **layers**, que são painéis que podem ser inseridos nas páginas. Estes painéis podem ser facilmente posicionados em qualquer posição da página, e dentro dele podem ser inseridos quaisquer componentes HTML, tais como texto, figuras ou applets, inclusive.

## Criando Páginas com Applets WebView

Para se criar páginas HTML com applets WebView no FrontPage, deve-se:

1. Selecionar a opção *Inserir* no menu, em seguida o sub-menu *Outros componentes* e dentro deste selecionar a opção *Miniaplicativo Java...*, conforme mostra a figura a seguir;



**Figura 3-2. Inserção de Applets Java**

2. Será exibida a janela *Propriedades do miniaplicativo Java*, mostrada na figura a seguir:



**Figura 3-3. Janela de propriedades do miniaplicativo Java**

3. No campo *Origem do miniaplicativo*, digitar o nome do applet a ser utilizado, seguido da extensão “.class”. Por exemplo, se o applet a ser utilizado for o AnalogInput, digitar “AnalogInput.class”;

**ATENÇÃO:**

**O nome dos applets é case sensitive**, e deve ser digitado com as letras maiúsculas e minúsculas exatamente da forma como definido neste manual. Por exemplo, o correto é “AnalogInput.class” e não “Analoginput.class”, porque a letra “i” está minúscula e deveria ser maiúscula

4. Para configurar o tamanho da área que o applet irá ocupar dentro da página HTML, podem ser utilizados os campos *Largura* e *Altura* que se encontram na seção *Tamanho* da janela atual;
5. Em seguida, clicar no botão *Atributos*: será exibida a janela *Atributos estendidos*, como mostrado na figura a seguir:



Figura 3-4. Janela de atributos estendidos

6. Nesta janela, clique no botão *Adicionar...*. É exibida a janela *Par nome/valor*, de acordo com a figura a seguir;

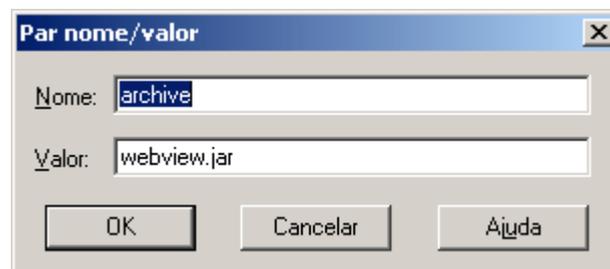


Figura 3-5. Janela de entrada de valores de atributos estendidos

7. No campo *Nome*, deve-se inserir o atributo *archive* e, em valor, o string “**webview.jar**”, e em seguida clicar no botão *OK* para inserir este atributo estendido;

**ATENÇÃO:**

O texto **webview.jar** deve ser escrito exatamente desta forma, com todas as letras minúsculo. Se for escrito de maneira diferente, a carga da página HTML pode ficar lenta porque o browser vai tentar ler novamente o mesmo arquivo "webview.jar" do WebGate (pois estando escrito de forma diferente, ele entende que é outro arquivo e carrega novamente do WebGate)

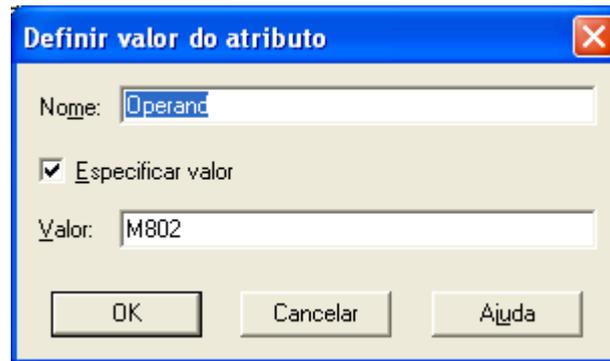
8. Na sequência, clica-se novamente em *OK* na janela *Atributos estendidos*, e mais uma vez em *OK* na janela *Propriedades do miniaplicativo Java*.

## Configurando os Parâmetros dos Applets

Para se configurar os parâmetros dos applets WebView através de páginas HTML criadas utilizando-se o FrontPage, deve-se:

1. Abrir a janela *Propriedades do miniaplicativo Java* do applet desejado, clicando-se com o botão direito do mouse sobre a área deste applet e selecionando a opção desejada no menu flutuante;
2. Em seguida, clica-se no botão *Adicionar...* na região *Parâmetros do miniaplicativo*;
3. Na caixa de diálogo *Definir valor do atributo* (mostrada na Figura 3-6), deve-se digitar no campo *Nome* o nome do parâmetro desejado e, com a caixa de checagem *Especificar valor* selecionada, digita-se no campo *Valor* o valor do respectivo parâmetro a ser cadastrado.

Por exemplo, para inserir o parâmetro “Operand” com valor igual a “M802”, deve-se digitar “Operand” no campo *Nome* e “M802” no campo *Valor*, conforme mostra a figura a seguir.

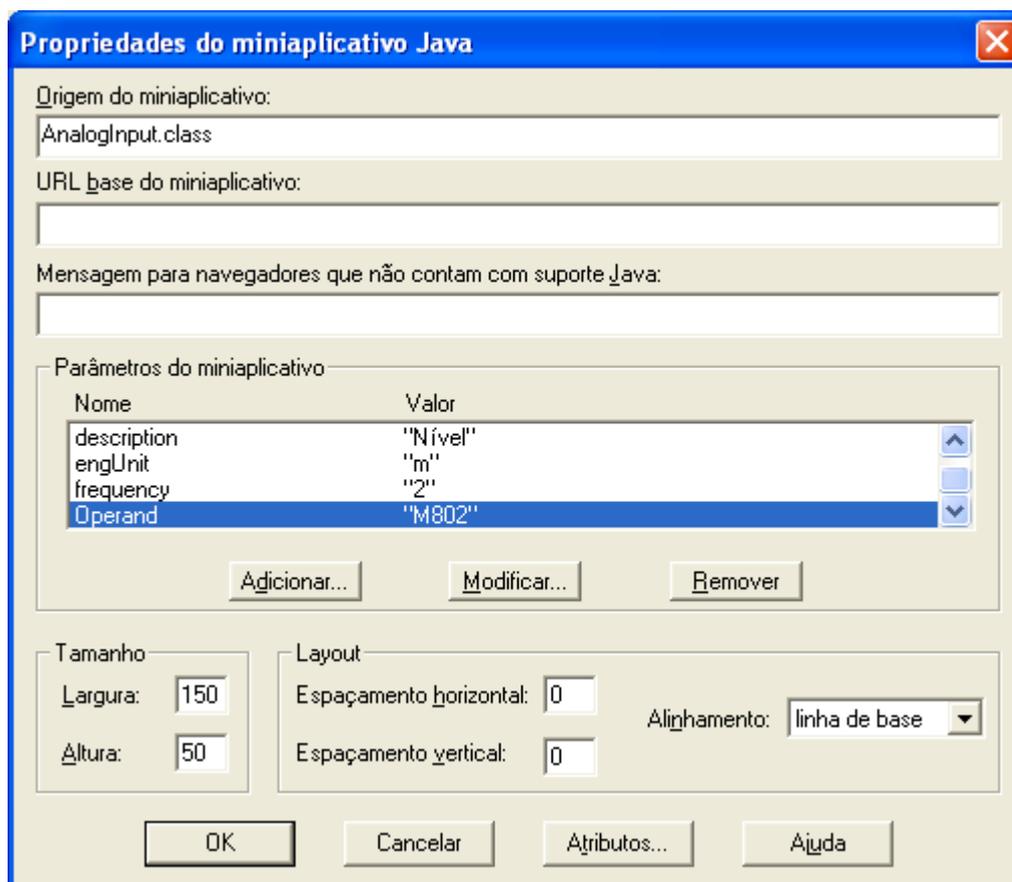


**Figura 3-6. Entrada de parâmetros e valores**

- Para concluir a inserção do novo parâmetro no applet, basta clicar no botão *OK*. Um exemplo de tela com todos os parâmetros configurados é mostrado na Figura 3-7.

**ATENÇÃO:**

Deve-se tomar cuidado para não escrever o nome do parâmetro de forma incorreta, como por exemplo “Description ” em vez de “Description”. O espaço em branco incluído a mais no fim do nome do parâmetro impossibilita que o applet execute da forma esperada.



### Figura 3-7. Parâmetros Configurados

Para facilitar a edição, pode-se utilizar os recursos copiar e colar dos Editores HTML para duplicar um applet já inserido e parametrizado. Na nova copia pode-se alterar apenas os parâmetros desejados.

## Enviando uma Página HTML para o WebGate

Para enviar uma página HTML para o WebGate, deve-se utilizar uma ferramenta de FTP qualquer. Neste manual foi utilizado o software WS-FTP LE.

No WS-FTP LE, deve-se seguir os seguintes passos:

1. Clique no botão *Connect* da tela principal do WS-FTP LE, para abrir a janela de conexão (*Session Properties*), mostrada na Figura 3-8.

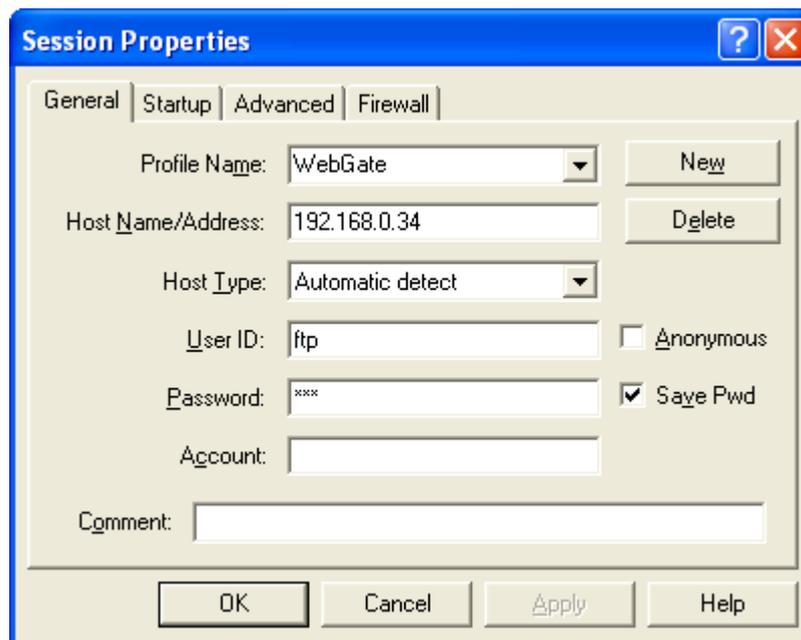


Figura 3-8. Janela de Conexão

2. Com a guia *General* selecionada, no campo *Profile Name*, digite um nome descritivo para a conexão com o WebGate (por exemplo, “WebGate”). Em *Host Name/Address*, digite o endereço IP do WebGate a ser utilizado. Nos campos *User ID* e *Password*, digite, respectivamente, o login e a senha de acesso ao WebGate (se desejar que o programa grave esta senha, selecione a opção *Save Pwd* ao lado do campo *Password*).
3. Clique em *Apply* e depois em *OK* para realizar a conexão com o WebGate.
4. Depois de realizada a conexão, deve se selecionar o diretório “WWW” do WebGate, que fica na região *Remote System* (na parte direita da tela principal), pois é neste diretório que serão armazenadas as páginas HTML do WebView.

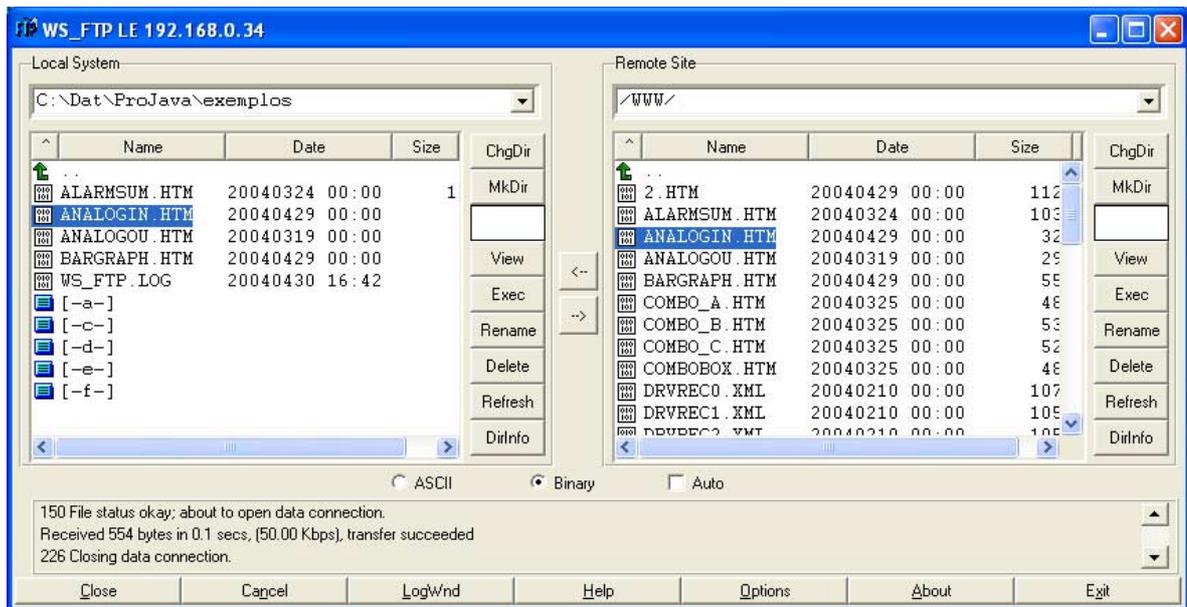


Figura 3-9. Tela principal do WS-FTP LE

- Para enviar uma página HTML (ou algum outro componente, como um gif) para o WebGate, busque e selecione a página desejada na região *Local System* (na parte esquerda da tela principal) e em seguida clique no botão com uma seta à direita (→) no centro da tela principal. A Figura 3-9 mostra a página “AnalogIn.HTM” selecionada para ser transmitida para o WebGate.

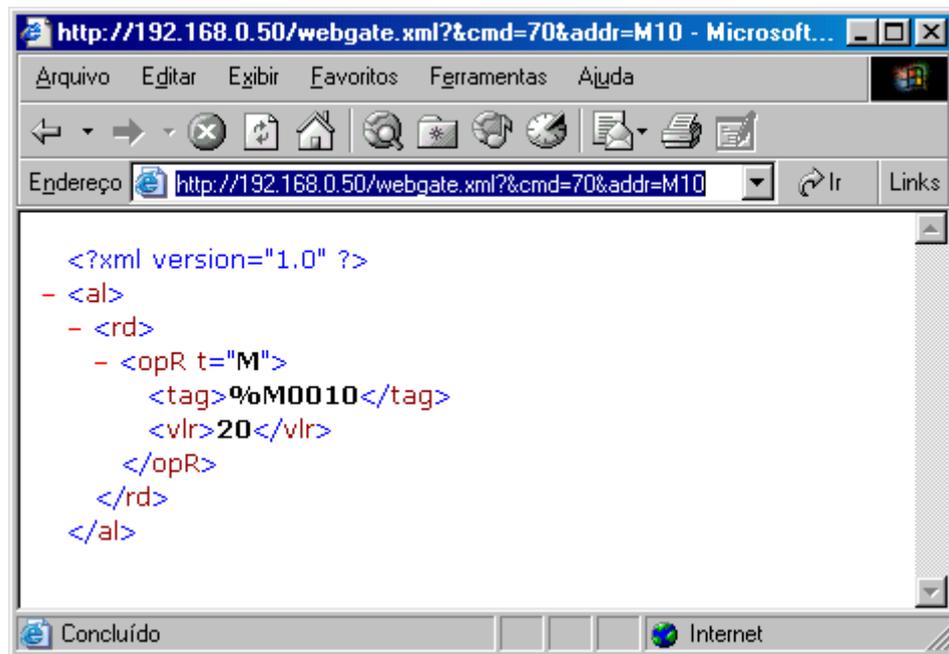
## Apresentado a Página no Browser

### Verificando a Comunicação

Depois de instalado o WebView, recomenda-se que seja feita uma verificação na comunicação com o WebGate e CP, antes de se acessar uma página HTML.

Esta verificação pode ser feita utilizando-se um comando que solicita diretamente ao WebGate o valor de um determinado operando do CP, mostrando o resultado no browser.

Para fazer esta verificação, deve-se digitar o seguinte texto na caixa de endereço do browser : <http://192.168.0.50/webgate.xml?&cmd=70&addr=M10> . Deve-se substituir o endereço IP informado (192.168.0.50) pelo endereço IP configurado no WebGate. O comando está solicitando o valor do operando M10. O WebGate recebe o comando, solicita ao CP o valor do operando M10, e responde para o browser no formato XML. O resultado é apresentado pelo browser no formato mostrado na figura a seguir:



**Figura 3-10. Verificando a Comunicação com CP e WebGate**

Se o valor do operando for apresentado no browser no formato mostrado na Figura 3-10, é sinal de que a comunicação entre o microcomputador, WebGate e CP está ocorrendo corretamente.

### Apresentando a Página

Para acessar uma página carregada no WebGate e visualizá-la através de um browser (navegador Web), basta digitar no campo de endereço o endereço IP do WebGate seguido de uma barra (“/”) e o nome do arquivo com a página HTML que foi carregada com a extensão “.HTM”.

Por exemplo, para acessar a página “Exemplo1” no WebGate de endereço IP 192.168.0.34, deve-se digitar “192.168.0.34/exemplo1.htm”. Esta página é um dos exemplos fornecidos juntos com o produto, está disponível no CDROM e pode ser vista neste manual no anexo Exemplos de Aplicação, na Figura 7-1.

Caso apenas o endereço IP seja digitado no browser, sem especificar a página HTML, como por exemplo “192.168.0.34”, o browser vai apresentar a página default “index.htm”, caso esta exista no WebGate.



**Figura 3-11. Ícone Java**

Ao apresentar as páginas com applets WebView, o Windows apresenta o ícone Java da Figura 3-11 na barra de tarefas, no canto inferior direito do monitor de vídeo. Se este ícone não for apresentado, é sinal de que o plug-in da máquina virtual Java pode não estar instalado ou apresentou algum problema de execução. A apresentação deste ícone pode ser desabilitada no Windows.

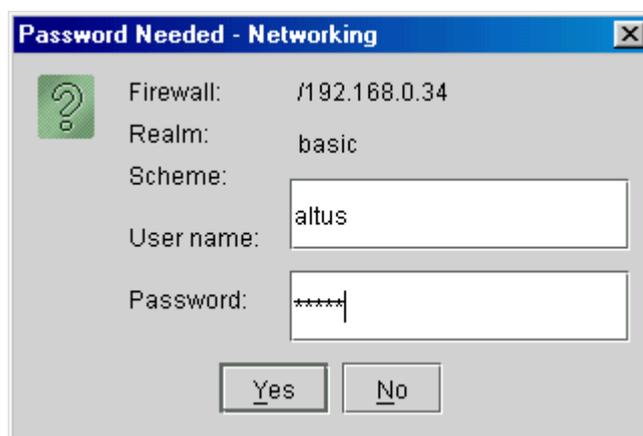
A primeira vez que uma janela de browser tenta apresentar uma tela HTML contendo um applet WebView, ele busca automaticamente o arquivo “webview.jar” do WebGate. Isto pode levar de 1 a 5 segundos, mas é feito uma única vez. Isto significa que logo que se abre uma nova janela de browser, a página HTML demora para ser apresentada, mas as páginas subsequentes não apresentarão esta mesma demora. Cada janela de browser aberta solicita sua própria cópia do “webview.jar” .

**ATENÇÃO:**

Recomenda-se que as aplicações desenvolvidas com o WebView possuam uma página inicial de abertura que utilize um applet Java, de forma que o tempo de carga do arquivo “webview.jar” aconteça apenas nesta tela.

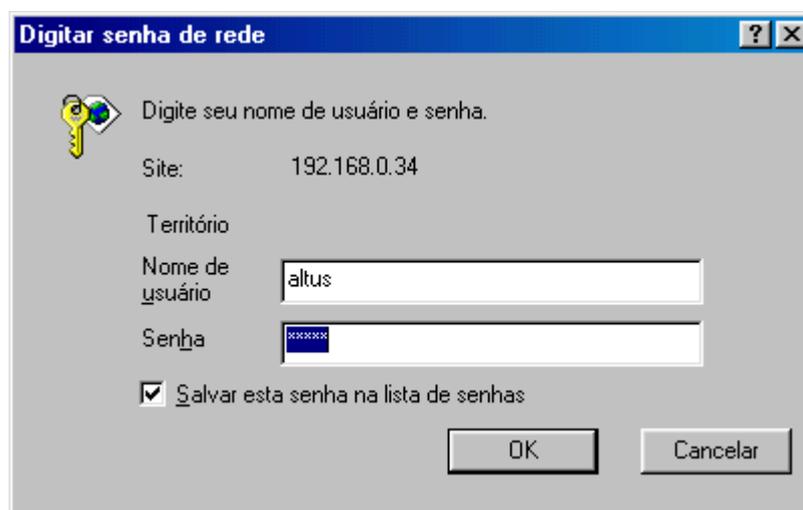
**Informando Usuário e Senha**

Ao carregar pela primeira vez, uma página HTML com applets WebView pode solicitar que sejam informados o nome e senha do usuário para acesso ao WebGate, conforme mostra a figura a seguir.



**Figura 3-12. Informando Usuário e Senha para Acesso ao WebGate**

Para cada nova janela de browser aberta será necessário realizar esta operação. Para evitar isto, pode-se salvar o nome e senha do usuário na lista de senhas do Internet Explorer, marcando-se o check box *Salvar esta senha na lista de senhas*, conforme mostra a figura a seguir. Esta tela é apresentada quando se solicita a página “login.htm” do WebGate.



**Figura 3-13. Salvando Nome e Senha do Usuário**

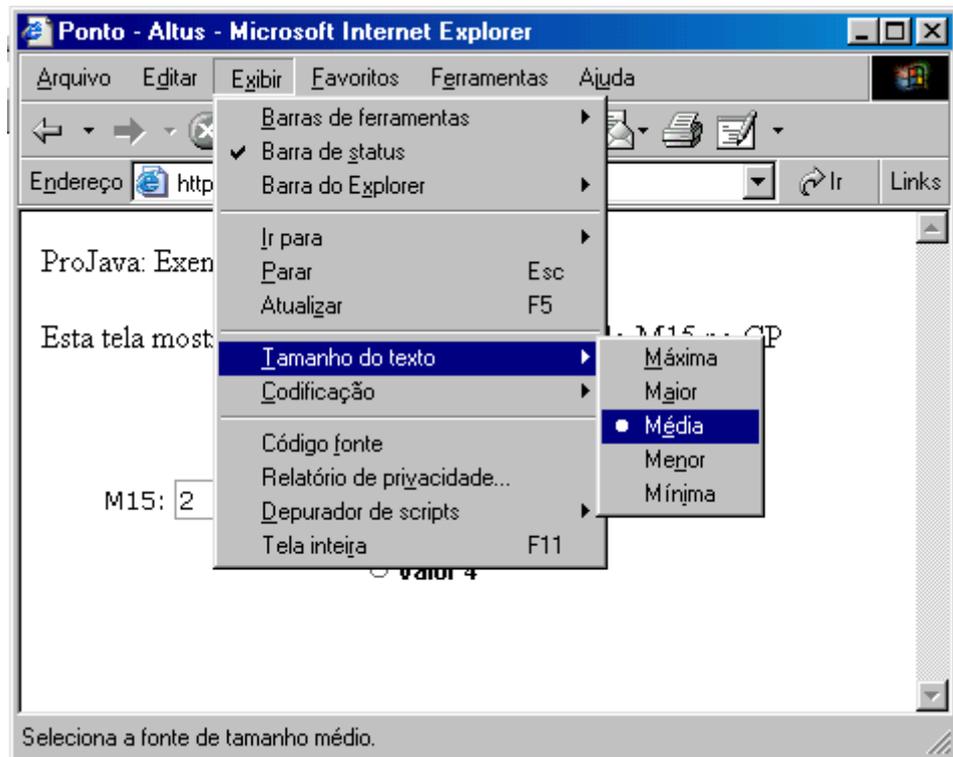
Para habilitar a janela de login mostrada na Figura 3-13, deve-se digitar no browser o endereço IP seguido de login.htm, como por exemplo “192.168.0.34/login.htm” . Após fornecer um usuário e senha válidos e marcar o check box *Salvar esta senha na lista de senhas*, as novas janelas de browser não solicitarão mais que o usuário e senha sejam informados, e a tela da Figura 3-12 não será mais apresentada.

**ATENÇÃO:**

Recomenda-se que as aplicações desenvolvidas com o WebView possuam uma página de login onde devem ser informados o nome e senha do usuário para acesso ao WebGate.

**Tamanho do Texto**

O tamanho do texto apresentado pelo Internet Explorer deve estar configurado para “Médio” para que as páginas possam ser corretamente apresentadas. Esta configuração pode ser feita a partir da opção de opção menu "Exibir \ Tamanho do Texto" , conforme mostra a Figura 3-14.



**Figura 3-14. Tamanho do Texto no Internet Explorer**

## 4. Utilização

### Características Gerais

#### Parâmetros dos Applets

Os applets WebView são configurados através de seus parâmetros, descritos detalhadamente neste capítulo. Alguns parâmetros são obrigatórios enquanto outros são opcionais. Parâmetros obrigatórios precisam sempre ser configurados para que o applet execute corretamente. Parâmetros opcionais podem ser ou não configurados; caso não sejam, assumem um valor default correspondente.

Os **parâmetros obrigatórios** aparecem em **negrito** na documentação de cada applet. Os parâmetros opcionais são mostrados com seus valores default correspondentes.

#### Operandos do CP

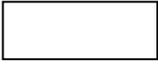
Todos os applets acessam um ou mais operandos do CP, para leitura e/ou escrita de valores. Alguns applets acessam ciclicamente o CP, outros apenas na carga do applet ou no instante da escrita de valores.

Os operandos do CP que podem ser acessados pelo WebView são:

- operandos memória ou bit de memória (exemplo: M10 ou M10.3)
- operandos auxiliar ou bit de auxiliar (exemplo: A11 ou A11.4)
- operandos octeto de E/S ou bit de octeto de E/S (exemplo: E12 ou E12.5, S13 ou S13.6)
- operandos decimal ou bit de decimal (exemplo: D14 ou D14.7)
- operandos tabela de memória ou posição de tabela de memória (exemplo: TM3 ou TM3[4])
- operandos tabela de decimal ou posição de tabela de decimal (exemplo: TD5 ou TD5[6])

#### Cor nos Applets

Todos os parâmetros de applets que se referem a cor podem ser informados com o nome da cor por extenso ou pelo seu código em hexadecimal. A tabela a seguir mostra as principais cores utilizadas no WebView.

String	Hexadecimal	Cor
Preto	000000	
Branco	FFFFFF	
Vermelho	FF0000	
Azul	0000FF	
Verde	008000	
Amarelo	FFFF00	
Laranja	FFA000	
Rosa	FF00C0	
Marrom	800000	
Bege	E0C0A0	
Cinza	A0A0A0	

O Anexo II mostra outros nomes de cor que podem ser utilizados.

Quando se criam páginas HTML que incluem figuras no formato JPEG ou GIF, o browser pode apresentar pequenas diferenças entre uma mesma cor definida na figura e na página, conforme mostra os diferentes tons de cinza na figura a seguir.



**Figura 4-1. Diferentes Tons da Cor Cinza**

Para que isto não aconteça, a resolução do adaptador de vídeo deve ser configurada para *True Color (32 Bits)* ou superior.

### Tamanho dos Applets

O tamanho que cada applet ocupa na tela HTML é uma área retangular cuja altura e largura devem ser configuradas pelo usuário. Isto pode ser feito através de um editor HTML, conforme mostra a

Figura 3-3 da seção **Criando Páginas com Applets WebView**, ou diretamente no código HTML, através dos parâmetros *width* e *height*, conforme mostra a figura a seguir.

```
<HTML>
  <applet code= "DigitalInput.class" archive=" webview.jar" width=150 height=50 >
    <param name = "Operand" value = "M802.0" >
  </applet>
</HTML>
```

**Figura 4-2. Configurando o Tamanho dos Applets**

**ATENÇÃO:**

Os parâmetros *width* e *height* devem ser configurados com um valor inteiro, sem aspas. Por outro lado, o nome e o valor dos parâmetros dos applets devem ser configurados com aspas duplas.

### Frequência de atualização dos Applets

O parâmetro *Frequency*, presente nos applets que fazem monitorações cíclicas do CP, define a frequência com que estas monitorações são realizadas. Após cada monitoração o applet é redesenhado para refletir o novo valor lido do CP.

Nos applets que utilizam o Driver para centralizar as monitorações, o parâmetro *Frequency* do applet deixa de ser considerado. Neste caso, a monitoração acontece de acordo com os parâmetros *Frequency* definidos no Driver, que se encarrega de solicitar que cada applet seja redesenhado após as monitorações correspondentes serem realizadas.

Um parâmetro *Frequency* configurado como zero significa que a monitoração vai ser realizada apenas uma vez, na carga da página HTML.

### Escala de Engenharia

Os parâmetros *MinCP*, *MaxCP*, *MinEng*, *MaxEng*, presentes em alguns applets, são utilizados para converter o valor de um operando do CP para unidades de engenharia, antes de apresentar na tela do browser. Estes parâmetros são opcionais. Se algum deles for omitido, nenhuma conversão será feita, apresentando-se o mesmo valor lido do CP.

Através destes parâmetros pode-se fazer qualquer conversão de escala. A conversão típica é um multiplicador, que pode ser maior ou menor que 1, mostrando na tela valores maiores ou menores que o armazenado no CP, respectivamente.

Por exemplo, para mostrar na tela do browser um valor dez vezes menor que o existente no CP, necessita-se de um multiplicador "x0.1" (divide por dez). Para mostrar um valor dez vezes maior, o multiplicador deve ser "x10". A tabela a seguir mostra exemplos de valores dos parâmetros *MinCP*, *MaxCP*, *MinEng*, *MaxEng* para obter estes multiplicadores:

Parâmetro	Multiplicador x0.1	Multiplicador x10
MinCP	0	0
MaxCp	1000	1000
MinEng	0	0
MaxEng	100	10000

**Tabela 4-1. Exemplos de Escala de Engenharia**

O valor dos parâmetros *MinCP* e *MaxCp* devem ser escolhidos em função dos valores mínimo e máximo que o operando correspondente pode assumir no CP.

## **Banco de Applets**

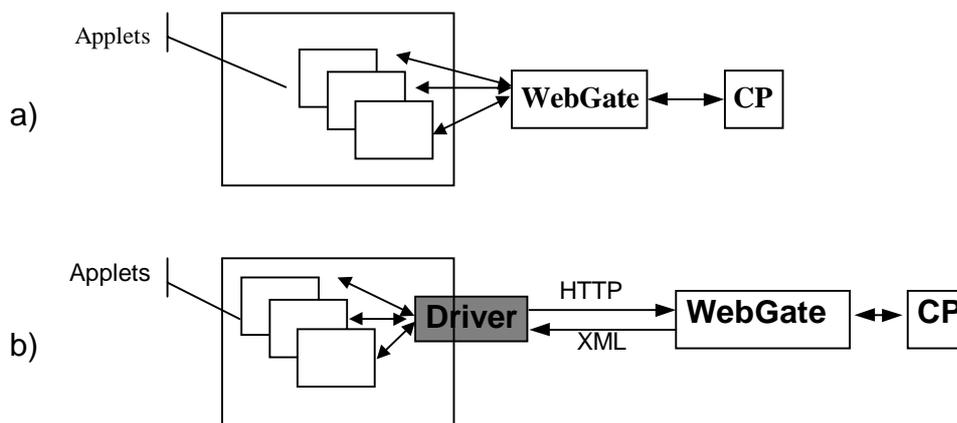
O WebView é formado por um banco de applets Java que podem ser incluídos em telas HTML desenvolvidas pelo usuário. O WebView é constituído pelos seguintes applets:

- Driver
- AnalogInput
- DigitalInput
- AnalogOutput
- DigitalOutput
- StringAnimation
- ColorAnimation
- ImageAnimation
- PushButton
- CheckBox
- RadioButton
- ComboBox
- ListBox
- TimeStamp
- BarGraph
- PenChart
- EventSummary
- AlarmSummary
- Receita
- Variável Local

Esta seção apresenta detalhadamente cada um destes applets que compõem o banco de applets. Para cada um é apresentada uma descrição geral do funcionamento, os parâmetros a serem configurados e um exemplo de utilização. Todos os exemplos apresentados podem ser encontrados no CDROM do produto.

## Driver

O Driver é um applet concentrador de comunicações entre os demais applets e o WebGate/CP. Quando uma página não utiliza o Driver (Figura 4-3a), cada applet realiza uma comunicação independente com o WebGate. Quando o Driver é utilizado (Figura 4-3b), os applets da página não falam diretamente com o WebGate, mas sim com o Driver, e este é quem faz todas as comunicações com o WebGate.



**Figura 4-3. Funcionamento do Applet Driver**

Com o Driver, os operandos a serem lidos podem ser agrupados em blocos para otimizar a comunicação. Podem ser definidas frequências de leitura diferentes para blocos diferentes, conforme a necessidade do processo a ser monitorado. Um bloco de operando pode ser uma tabela inteira, parte da tabela ou um conjunto de operandos memória, auxiliar ou decimal.

Para se criar uma página com Driver, deve-se inserir e parametrizar o applet Driver com os blocos de operandos do CP a serem monitoradas e a ciclicidade, conforme os parâmetros descritos adiante.

Para que os demais applets da página acessem o WebGate pelo Driver deve-se inserir o caracter “:” (dois pontos) antes do valor do parâmetro Operand destes applets. Por exemplo, para mostrar com o AnalogInput o valor do operando M30 através do Driver, o parâmetro “Operand” deve ser configurado como “:M30”. Outros exemplos: “: TM3”, “:TM3[15]”.

**ATENÇÃO:**

Recomenda-se que todas as páginas com mais de 5 applets que fazem monitoração cíclica utilizem o Driver para otimizar as comunicações.

### Parâmetros do Driver

A seguir encontram-se os parâmetros do componente Driver. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de seus possíveis valores e o valor default, usado no caso do parâmetro não ter um valor definido no HTML.

Parâmetro	Descrição	Valores	Valor Default
QtyPolling	Quantidade de blocos de operandos que serão monitorados pelo Driver.	De 1 a 16	Não existe
Operand[1..n]	Especifica o operando inicial do bloco a ser monitorado.	Ex: M7, TM3, TM3[15]	Não existe
BlockSize[1..n]	Tamanho do bloco de operandos que será monitorado	De 1 a 64	Não existe
Frequency[1..n]	Tempo de polling de cada operando em segundos.	De 0 a 600	1

Tabela 4-2 . Parâmetros do Driver

No exemplo a seguir, temos um Driver segundo os parâmetros da tabela abaixo:

Parâmetro	Valor
NumInterval	3
Operand1	M0
Operand2	S0
Operand3	TM4
BlockSize1	40
BlockSize 2	9
BlockSize 3	15
Frequency1	10
Frequency2	20
Frequency3	15

Para este exemplo, teremos um Driver que realiza uma leitura dos operandos:

- Do M0 ao M39 a cada 10 segundos,
- Do S0 ao S8 a cada 20 segundos e
- TM4 posições 0 a 14 a cada 15 segundos.

A escrita de um operando via Driver é feita diretamente no CP, ou seja, os valores locais serão atualizados apenas na próxima leitura realizada pelo Driver. A escrita é feita por operando de forma ad hoc e individualmente, ou seja, cada requisição de forçamento implicará numa comunicação do Driver com o WebGate.

O applet Driver não se desenha na tela, é “transparente” em tempo de execução.

#### Exemplo de HTML com Uso do Driver

Este exemplo mostra um applet Driver configurado para monitorar dois blocos de operando. O primeiro bloco corresponde aos operandos M100 a M107, monitorado a cada 5 segundos. O segundo bloco corresponde ao operando tabela TM3, 20 posições, monitorado a cada 10 segundos.

```
<HTML>
<applet code="Driver" width="0" height="0" archive=" webview.jar">
  <param name="QtyPolling" value="2">

  <param name="Operand1" value="M100">
  <param name="BlockSize1" value="8">
  <param name="Frequency1" value="5">
```

```

<param name="Operand2" value="TM3">
<param name="BlockSize2" value="20">
<param name="Frequency2" value="10">
</applet>
</HTML>

```

**ATENÇÃO:**

Recomenda-se que o applet Driver seja chamado no início das páginas HTML, antes dos demais applets.

**AnalogInput**

Applet responsável por ler valores analógicos de operandos do CP e apresentá-los na tela do browser.

*Parâmetros do AnalogInput*

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Exemplo de Valores	Valor Default
<b>Operand</b>	Especifica o operando a ser lido	Ex: M7, A2, TM3[4], D40	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
FieldColor	Cor de fundo do campo	Ex: Amarelo	O mesmo configurado para background
<b>Características relativas à Fonte</b>			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito 11 – negrito e itálico	00
<b>Características específicas do Applet</b>			
Description	Descrição associada ao operando, apenas para apresentação na tela		Em branco
DecimalDigits	Quantidade de casas decimais	De 0 a 5	0 (zero)
EngUnit	Unidade de Engenharia, apenas para apresentação na tela	Ex: s, mm, l	Em branco
<b>Escala de Engenharia</b>			
MinCP	Valor mínimo do operando no CP		Não existe
MaxCP	Valor máximo do operando no CP		Não existe
MinEng	Valor mínimo relativo do operando		Não existe
MaxEng	Valor máximo relativo do operando		Não existe

**Tabela 4-3. Parâmetros do applet AnalogInput**

Os parâmetros MinCP, MaxCP, MinEng, MaxEng são utilizados para converter o valor de um operando do CP para unidades de engenharia. São opcionais. Se algum deles for omitido no HTML ou for inválido, nenhuma conversão será feita, apresentando o mesmo valor lido no CP.

#### Exemplo de HTML com AnalogInput

Este exemplo mostra uma página HTML monitorando um operando (M802) do CP a cada 2 segundos, com a cor de fundo “Celeste”, o texto “Nível” e a unidade “m”, de metros.

```
<HTML>
  <applet code= "AnalogInput.class" archive="webview.jar" width=150 height=50>
    <param name = "Operand" value = "M802">
    <param name = "frequency" value="2">
    <param name = "description" value = "Nível">
    <param name = "engUnit" value = "m">
    <param name = "background" value = "Celeste">
  </applet>
</HTML>
```

#### Visualização do Exemplo do AnalogInput

**Figura 4-4. Visualização do Exemplo do AnalogInput**

#### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- DigitaInput
- AnalogOutput
- DigitaOutput

#### DigitalInput

Applet responsável por ler valores digitais do CP e apresentá-los na tela do browser.

#### Parâmetros do DigitalInput

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
Operand	Especifica o operando a ser lido	Ex: A1.3, M7.1, D40.7	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
FieldColor	Cor de fundo do campo	Ex: Amarelo	O mesmo configurado para background
Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito 11 – negrito e itálico	00
Características específicas do Applet			
Description	Descrição associada ao operando, apenas para apresentação na tela.		Em branco

**Tabela 4-4. Parâmetros do applet DigitalInput**

Os valores apresentados são apenas 0 ou 1. Se o parâmetro Operand corresponder a um operando sem subdivisão cujo valor no CP não é 0 ou 1, ainda assim o valor apresentado será 0 ou 1 (para valores diferentes de zero).

#### Exemplo de HTML com DigitalInput

Este exemplo mostra uma página HTML monitorando um bit de operando do CP (M802.0) a cada segundo, com a cor de fundo “Celeste” e o texto “Estado”.

```
<HTML>
  <applet code= "DigitalInput.class" archive="webview.jar" width=150 height=50>
    <param name = "Operand" value = "M802.0">
    <param name = "frequency" value = "1">
    <param name = "description" value = "Estado">
    <param name = "background" value = "Celeste">
  </applet>
</HTML>
```

#### Visualização do Exemplo do DigitalInput



**Figura 4-5. Visualização do Exemplo do DigitalInput**

#### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- AnalogInput
- DigitalOutput
- AnalogOutput

## AnalogOutput

Applet responsável por ler e escrever valores analógicos de operandos do CP. O valor é lido e apresentado uma única vez, na carga da tela HTML pelo browser. O valor digitado é escrito no CP quando se pressiona ENTER.

### Parâmetros do AnalogOutput

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
<b>Operand</b>	Especifica o operando a ser escrito	Ex: M7, A2, TM3[4], D40	Não existe
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
FieldColor	Cor de fundo do campo	Ex: Amarelo	O mesmo configurado para background
Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito 11 – negrito e itálico	00
Características específicas do Applet			
Description	Descrição associada ao operando, apenas para apresentação na tela		em branco
DecimalDigits	Quantidade de casas decimais	De 0 a 5	0 (zero)
EngUnit	Unidade de Engenharia, apenas para apresentação na tela	Ex: s, mm, l	em branco
MinLimit	Valor mínimo que pode ser escrito pelo usuário	Ex: 10, 80	0 (zero)
MaxLimit	Valor máximo que pode ser escrito pelo usuário	Ex: 70, 90	32767
Escala de Engenharia			
MinCP	Valor mínimo do operando no CP		Não existe
MaxCP	Valor máximo do operando no CP		Não existe
MinEng	Valor mínimo relativo do operando		Não existe
MaxEng	Valor máximo relativo do operando		Não existe

**Tabela 4-5. Parâmetros do applet AnalogOutput**

Os parâmetros MinLimit e MaxLimit são utilizados para consistir o valor digitado no applet. Valores fora deste intervalo não são aceitos.

Os parâmetros MinCP, MaxCP, MinEng, MaxEng são utilizados para converter o valor de um operando do CP para unidades de engenharia. São opcionais. Se algum deles for omitido no HTML ou for inválido, nenhuma conversão será feita, apresentando o mesmo valor lido no CP.

### Exemplo de HTML com AnalogOutput

Este exemplo mostra uma página HTML onde pode-se modificar o valor do operando M802. Após pressionar ENTER, o valor digitado será enviado para o CP. Foi configurado a cor de fundo "Celeste", o texto "Abertura" e a unidade "%". Nenhuma conversão de escala foi especificada.

```
<HTML>
  <applet code= "AnalogOutput.class" archive=" webview.jar " width=150 height=50>
    <param name = "Operand" value = "M802">
    <param name = "description" value = "Abertura">
    <param name = "engUnit" value = "%">
    <param name = "background" value = "Celeste">
  </applet>
</HTML>
```

### Visualização do Exemplo do AnalogOutput

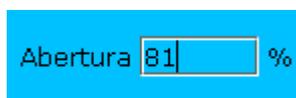


Figura 4-6. Visualização do Exemplo do AnalogOutput

### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- AnalogInput
- DigitaInput
- AnalogInput

### DigitalOutput

Applet responsável por ler e escrever valores digitais de operandos do CP. O valor é lido e apresentado uma única vez, na carga da tela HTML pelo browser. O valor digitado é escrito no CP quando se pressiona ENTER.

### Parâmetros do DigitalOutput

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
Operand	Especifica o operando a ser escrito	Ex: A1.3, A15.4	Não existe
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
FieldColor	Cor de fundo do campo	Ex: Amarelo	O mesmo configurado para background
Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito	00 – normal	00

	e/ou itálico.	01 – itálico 10 – negrito 11 – negrito e itálico	
Características específicas do Applet			
Description	Descrição associada ao operando, apenas para apresentação na tela.		Em branco

**Tabela 4-6. Parâmetros do applet DigitalOutput**

O parâmetro *Operand* só aceita bits de operando do tipo auxiliar (%A).

#### Exemplo de HTML com Uso do DigitalOutput

Este exemplo mostra uma página HTML onde pode-se modificar um bit de operando do CP (A13.7). Após pressionar ENTER, o valor digitado será enviado para o CP.

```
<HTML>
  <applet code= "DigitalOutput.class" archive=" webview.jar " width=150 height=50>
    <param name = "Operand" value = "A13.7">
    <param name = "description" value = "Comando">
    <param name = "background" value = "Celeste">
  </applet>
</HTML>
```

#### Visualização do Exemplo do Applet DigitalOutput



**Figura 4-7. Visualização do Exemplo do DigitalOutput**

#### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- AnalogInput
- DigitaInput
- AnalogOutput

#### StringAnimation

Applet responsável por ler um operando do CP e apresentar seu valor na forma de uma string configurável. Na criação da página HTML, define-se uma string para cada faixa de valores desejada. Na execução da página, o browser apresenta a string correspondente, conforme o valor do operando lido do CP.

#### Parâmetros do StringAnimation

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
<b>Operand</b>	Especifica o operando a ser lido do CP	Ex: M7, TM3[4], D40	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
Características relativas à Fonte			
FontType [1..n]	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize [1..n]	Tamanho da fonte	De 8 a 32	12
FontColor [1..n]	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle [1..n]	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito 11 – negrito e itálico	00
Características específicas do Applet			
<b>String[1..n]</b>	Descrição associada à faixa de valores do operando, apenas para apresentação na tela	Ex: aberto, fechado	Não existe
<b>Interval[1..n]</b>	Faixa de valores do operando	Ex: 0-50, 2-3	0-0
<b>NumInterval</b>	Quantidade de faixa de valores declarados	De 2 a 32	2

Tabela 4-7. Parâmetros do applet StringAnimation

**NumInterval**

Define a quantidade de strings e faixas de valores correspondentes declarados.

**String[1..n]**

Representa as strings que poderão ser exibidas pelo applet associadas a cada faixa de valores do parâmetro *Interval [1..n]*, conforme o valor lido do operando. Se o valor lido não pertencer a nenhuma das faixas de intervalo declaradas, será exibida a string “?”.

**Interval[1..n]**

Representa as faixas de valores possíveis para o operando monitorado, e estarão associadas a cada string no parâmetro *String[1..n]*. Desta forma, os valores declarados no parâmetro *Interval1* estarão associados à exibição da string declarada em *String1*, e assim sucessivamente.

*Exemplo de HTML com Uso do StringAnimation*

O exemplo a seguir mostra o StringAnimation configurado para apresentar as strings “Aberto”, “Fechado” e “Semi-aberto” para as faixas de valores “1 a 10”, “11 a 20” e “21 a 30”, respectivamente. O operando M20 está configurado para ser monitorado do CP a cada três segundos, e seu valor instantâneo é usado para selecionar qual das três strings será apresentada.

```
<HTML>
  <applet code = "StringAnimation.class" archive=" webview.jar " width = "150" height =
  "50">
    <param name = "Operand" value = "M20">
    <param name = "Frequency" value="3">
    <param name = "background" value = "Celeste">
    <param name = "String1" value = "Aberto">
    <param name = "String2" value = "Fechado">
    <param name = "String3" value = "Semi-aberto">
    <param name = "numinterval" value = "3">
    <param name = "interval1" value = "1-10">
    <param name = "interval2" value = "11-20">
```

```

        <param name = "interval3" value = "21-30">
    </applet>
</HTML>

```

### Visualização do Exemplo do Applet StringAnimation

A figura a seguir mostra o exemplo do StringAnimation para quatro valores diferentes do operando M20, lido do CP a cada 3 segundos.

a)		para M20 = 5 (faixa 1-10)
b)		para M20 = 15 (faixa 11-20)
c)		para M20 = 25 (faixa 21-30)
d)		para M20 = 35 (fora de todas as faixas ou sem comunicação com CP)

**Figura 4-8. Visualização do Exemplo do StringAnimation**

### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- ColorAnimation
- ImageAnimation

### ColorAnimation

Applet responsável por ler um operando do CP e apresentar seu valor na forma de um retângulo preenchido com uma cor configurável. Na criação da página HTML, define-se uma cor para cada faixa de valores desejada. Na execução da página, o browser apresenta um retângulo preenchido com a cor correspondente, conforme o valor do operando lido do CP.

### Parâmetros do ColorAnimation

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
Operand	Especifica o operando a ser lido do CP	Ex: M7, TM3[4], D40	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
Características específicas do Applet			
AnimColor[1..n]	Cor associada à faixa	Ex: amarelo, azul	Preto

	de valores do operando		
<b>Interval[1..n]</b>	Faixa de valores do operando	Ex: 0-50, 2-3	0-0
<b>NumInterval</b>	Indica o número de faixa de valores declarados	De 2 a 32	2

Tabela 4-8. Parâmetros do applet ColorAnimation

**NumInterval**

Define a quantidade de cores e faixas de valores correspondentes declarados.

**AnimColor[1..n]**

Representa as cores que poderão ser exibidas pelo applet associadas a cada faixa de valores do parâmetro *Interval [1..n]*, conforme o valor lido do operando. Se o valor lido não pertencer a nenhuma das faixas de intervalo declaradas, será exibido um retângulo de cor branca com um "X" vermelho em seu interior.

As cores podem ser definidas por meio do valor hexadecimal que representa a mesma ou através do nome da cor por extenso, da mesma forma que no parâmetro background. Ver tabela de cores no Apêndice I.

**Interval[1..n]**

Representa as faixas de valores possíveis para o operando monitorado, e estarão associadas a cada cor no parâmetro *AnimColor[1..n]*. Desta forma, os valores declarados no parâmetro *Interval1* estarão associados à exibição da cor declarada em *AnimColor1*, e assim sucessivamente.

*Exemplo de HTML com Uso do ColorAnimation*

O exemplo a seguir mostra o ColorAnimation configurado para apresentar as cores "azul" , "vermelho" e "verde" para as faixas de valores "1 a 10" , "11 a 20" e "21 a 30", respectivamente. O operando M20 está configurado para ser monitorado do CP a cada três segundos, e seu valor instantâneo é usado para selecionar qual das três cores será apresentada.

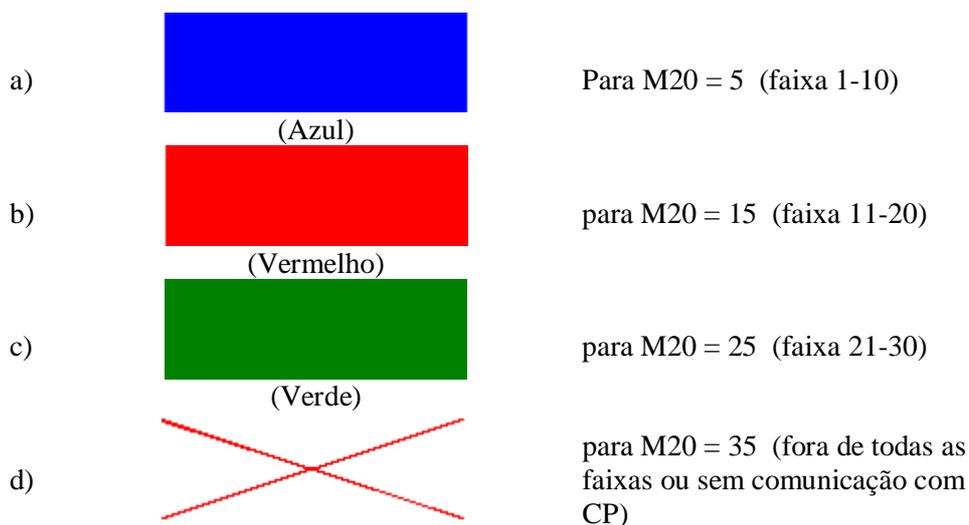
```

<HTML>
  <applet code= "ColorAnimation.class" archive=" webview.jar " width = "150" height = "50">
    <param name = "Operand" value = "M20">
    <param name = "AnimColor1" value = "azul">
    <param name = "AnimColor2" value = "vermelho">
    <param name = "AnimColor3" value = "verde">
    <param name = "frequency" value="1">
    <param name = "numinterval" value = "3">
    <param name = "interval1" value = "1-10">
    <param name = "interval2" value = "11-20">
    <param name = "interval3" value = "21-30">
  </applet>
</HTML>

```

*Visualização do Exemplo do Applet ColorAnimation*

A figura a seguir mostra o exemplo do ColorAnimation para quatro valores diferentes do operando M20, lido do CP a cada 3 segundos.



**Figura 4-9. Visualização do Exemplo do ColorAnimation**

### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- StringAnimation
- ImageAnimation

### ImageAnimation

Applet responsável por ler um operando do CP e apresentar seu valor em forma de uma imagem, um arquivo gif ou gif animado.

Na criação da página HTML, define-se um arquivo gif para cada faixa de valores desejada. Na execução da página, o browser apresenta a imagem do arquivo gif correspondente, conforme o valor do operando lido do CP.

### Parâmetros do ImageAnimation

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
<b>Operand</b>	Especifica o operando a ser lido do CP	Ex: M7, TM3[4], D40	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
Características específicas do Applet			
<b>Image[1..n]</b>	Imagem associada à faixa de valores do operando	Ex: bomba.gif, valv.gif	Não existe
<b>Interval[1..n]</b>	Faixa de valores do operando	Ex: 0-50, 2-3	0-0
<b>NumInterval</b>	Indica o número de faixa de valores declarados	De 2 a 32	2

Tabela 4-9. Parâmetros do applet ImageAnimation

**NumInterval**

Define a quantidade de imagens e faixa de valores correspondentes declarados.

**Image[1..n]**

Representa os arquivos GIF de imagens que poderão ser exibidas pelo applet associadas a cada faixa de valores do parâmetro *Interval* [1..n], conforme o valor lido do operando. Caso o valor lido do operando do CP não esteja em alguma das faixas declaradas, será exibido um retângulo de cor branca com um "X" vermelho em seu interior.

**Interval[1..n]**

Representa as faixas de valores possíveis para o operando monitorado, e estarão associadas a cada imagem no parâmetro *Image*[1..n]. Desta forma, os valores declarados no parâmetro *Interval* estarão associados à exibição da imagem declarada em *Image*1, e assim sucessivamente.

As imagens podem ser GIFs estáticos ou animados, desenvolvidos através de um editor convencional de GIFs. As imagens, para serem usadas pelo applet, devem ser carregadas por meio de FTP para o diretório onde se encontram os HTMLs no WebGate.

*Exemplo de HTML com Uso do ImageAnimation*

O exemplo a seguir mostra o ImageAnimation configurado para apresentar imagens de lâmpada apagada, parcialmente acesa e lâmpada acesa. Estas imagens estão armazenadas nos arquivos "off1.gif", "off2.gif" e "on1.gif" e correspondem às faixas de valores "1 a 10", "11 a 20" e "21 a 30", respectivamente. O operando M20 está configurado para ser monitorado do CP a cada três segundos, e seu valor instantâneo é usado para selecionar qual das três imagens será apresentada.

```
<HTML>
  <applet code ="ImageAnimation.class" archive=" webview.jar " width = "80" height = "71">
    <param name = "Operand" value = "M20">
    <param name = "frequency" value="1">
    <param name = "Image1" value = "Off1.gif">
    <param name = "Image2" value = "Off2.gif">
    <param name = "Image3" value = "On1.gif">
    <param name = "numinterval" value = "3">
    <param name = "interval1" value = "1-10">
    <param name = "interval2" value = "11-20">
    <param name = "interval3" value = "21-30">
  </applet>
</HTML>
```

**ATENÇÃO:**

É possível carregar as imagens GIF em diretórios do WebGate. Para informar os nomes, deve-se acrescentar o path do diretório, separado pelo caracter "/", antes do nome do GIF. Por exemplo, o parâmetro "FIGURAS/VALV.GIF" se refere a imagem VALV.GIF armazenada no diretório FIGURAS.

*Visualização do Exemplo do Applet ImageAnimation*

A figura a seguir mostra o exemplo do ImageAnimation para quatro valores diferentes do operando M20, lido do CP a cada 3 segundos.

a)		para M20 = 5 (faixa 1-10)
b)		para M20 = 15 (faixa 11-20)
c)		para M20 = 25 (faixa 21-30)
d)		para M20 = 35 (fora de todas as faixas ou sem comunicação com CP)

**Figura 4-10. Visualização do Exemplo do ImageAnimation**

### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- StringAnimation
- ColorAnimation

### PushButton

Applet responsável por apresentar um botão na tela do browser. Quando o botão é pressionado, um determinado valor é forçado em determinado operando do CP. Tanto o valor quanto o operando são configurados pelos parâmetros do applet.

### Parâmetros do PushButton

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
<b>Operand</b>	Especifica o operando a ser escrito no CP	Ex: M7, TM3[4], D40.7, A13.7	Não existe
Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito 11 – negrito e itálico	00
Características específicas do Applet			
BtDescription	String associada ao	Ex: abrir, fechar	Em branco

	rótulo do botão		
BtColor	Define a cor do botão	Ex: "FFFFFF", "gelo"	Cinza claro
Value	Valor do Forçamento	Ex: 1, 100	1 (um)

**Tabela 4-10. Parâmetros do applet PushButton**

O parâmetro *BtDescription* é utilizado como o nome do botão.

O parâmetro *Value* define o valor que vai ser forçado em *Operand* a cada vez que o botão for pressionado.

#### Exemplo de HTML com Uso do PushButton

O exemplo a seguir mostra um PushButton configurado para escrever o valor 200 no operando M20 do CP, quando o botão for pressionado.

```
<HTML>
  <applet code= "PushButton.class" archive=" webview.jar " width = "150" height = "50">
    <param name = "Operand" value = "M20">
    <param name = "BtDescription" value = "Salvar">
    <param name = "value" value = "300">
  </applet>
</HTML>
```

#### Visualização do Exemplo do Applet PushButton



**Figura 4-11. Visualização do Exemplo do PushButton**

#### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- CheckBox
- RadioButton
- ComboBox
- ListBox

#### CheckBox

Applet responsável por apresentar um controle do tipo “check box” na tela do browser, que realiza a leitura e/ou escrita de valores armazenados em operandos do CP.

#### Parâmetros do CheckBox

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
Operand	Especifica o operando	Ex: A3, M7, D40	Não existe

	a ser lido/escrito no CP		
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 - normal 01 - itálico 10 - negrito 11 - negrito e itálico	00
Características específicas do Applet			
<b>NumElem</b>	Quantidade de check boxes. Cada elemento está relacionado a um bit do operando.	De 1 a 16	Não existe
<b>Description [1..n]</b>	Descrição de cada check box		Não existe
BtDescription	Descrição do botão opcional para envio de todos os check box de uma única vez para o CP	Ex: OK, Ligar	Botão não é apresentado
ReadOnly	Define se é permitido ao usuário realizar forçamentos	True / False	False

Tabela 4-11 . Parâmetros do applet CheckBox

Os parâmetros em negrito são obrigatórios. Para os demais será utilizado o valor default quando os respectivos forem omitidos no HTML.

### Operand

Especifica o operando que vai ser lido e/ou escrito no CP, associado aos check boxes. **Cada bit** deste operando corresponde a um check box diferente. **Este applet é orientado a bit e não a valor.**

### NumElem

Define a quantidade de check boxes apresentados. Cada check box é apresentado com a sua descrição e está associado a um bit de *Operand*. O primeiro check box está associado ao bit 0 de *Operand*, o segundo ao bit 1, e assim por diante.

### Description[1..n]

Define a descrição apresentada para cada check box. O parâmetro descrição do primeiro check box é "Description1", do segundo "Description2", e assim por diante.

### ATENÇÃO:

Devem ser definidos uma quantidade de descrições igual a definida no parâmetro *NumElem*.

### BtDescription

Opcionalmente pode-se desejar enviar para o CP os valores do check boxes todos de uma única vez, e não instantaneamente quando se clica em cada check box. Para isto, utiliza-se o parâmetro *BtDescription*, que faz com que o applet apresente um botão que faz esta operação.

### ReadOnly

Este parâmetro define o comportamento somente leitura. Quando valer *true*, o applet apenas monitora do CP o valor dos check box, mas não permite que o usuário faça nenhuma alteração. Neste caso, os check boxes estarão desabilitados.

Este parâmetro só faz sentido quando o parâmetro *BtDescription* não é declarado.

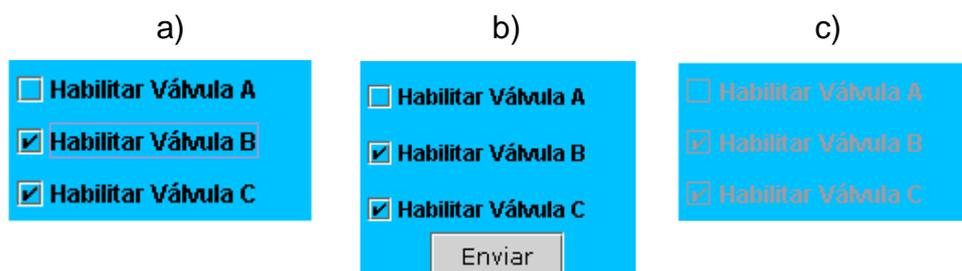
### Descrição do Funcionamento

Quando o applet é executado ele apresenta na tela do browser todos os check boxes e respectivas descrições declarados no HTML através dos parâmetros *NumElem* e *Description[1..n]*.

Em seguida o applet monitora (uma única vez) no CP o valor do operando definido em *Operand*. Cada bit com valor 1 será apresentado como **um check box** marcado e cada bit com valor 0 será apresentado como um check box desmarcado.

São possíveis três tipos de comportamento deste applet, conforme os parâmetros *BtDescription* e *ReadOnly*:

- Sem Botão (Figura 4-12a)
- Com Botão (Figura 4-12b)
- Sem Botão e Somente Leitura (Figura 4-12c)



**Figura 4-12. Tipos de Comportamento do Applet CheckBox**

O applet CheckBox no modo sem botão (sem *BtDescription*, sem *ReadOnly*, conforme mostra a Figura 4-12a) permite que o usuário possa clicar em qualquer check box, para marca-lo ou desmarca-lo. Ao clicar, o valor correspondente é imediatamente enviado ao CP.

O CheckBox no modo com botão (com *BtDescription*, sem *ReadOnly*, conforme mostra a Figura 4-12b) permite que o usuário possa clicar em qualquer check box, para marca-lo ou desmarca-lo, porem nenhum valor é enviado ao CP até que o botão especificado por *BtDescription* seja pressionado.

O CheckBox no modo sem botão e somente leitura (sem *BtDescription*, com *ReadOnly*, conforme mostra a Figura 4-12c) não permite ao usuário clicar em nenhum dos check box. Nenhum valor é enviado ao CP, o applet se comporta como somente leitura.

### Exemplo de HTML com Uso do CheckBox

Este exemplo mostra um applet CheckBox configurado para 3 check boxes, no modo sem botão, visualizado na Figura 4-12a.

```
<HTML>
  <applet code="CheckBox.class" archive = " webview.jar " width = 150 height = 80>
    <param name = "Operand" value = "M50">
    <param name = "NumElem" value = "3">
    <param name = "Description1" value = "Habilitar Válvula A">
    <param name = "Description2" value = "Habilitar Válvula B">
    <param name = "Description3" value = "Habilitar Válvula C">
    <param name="Background" value="celeste">
  </applet>
</HTML>
```

### Visualização do Exemplo do Applet CheckBox

O exemplo do applet CheckBox pode ser visualizado na Figura 4-12a.

### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- ComboBox
- RadioButton
- ListBox

### RadioButton

Applet responsável por apresentar um elemento do tipo RadioButton na tela do browser, que realiza a leitura/escrita de valores armazenados em operandos do CP.

### Parâmetros do RadioButton

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
<b>Operand</b>	Especifica o operando a ser lido/escrito no CP	Ex: A3, M7, D40, TM3[4]	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito 11 – negrito e itálico	00
Características específicas do Applet			
BtDescription	Descrição do evento/operação associado ao operando	Ex: OK, Ligar	Não existe
<b>NumElem</b>	Número de elementos apresentados	De 1 a 16	Não existe
<b>Description [1..n]</b>	Descrição associada a cada elemento		Não existe
ReadOnly	Indica se é permitido ao usuário realizar forçamentos (só faz sentido quando o parâmetro BtDescription não é declarado)	True / False	False

**Tabela 4-12. Parâmetros do applet RadioButton**

### Operand

Especifica o operando que vai ser lido e/ou escrito no CP, associado aos radio buttons. **Este applet é orientado a valor, e não a bit.** O valor 1 de *Operand* corresponde ao primeiro radio button, o valor 2 ao segundo, e assim por diante.

**NumElem**

Define a quantidade de radio buttons apresentados. Cada radio button é apresentado com a sua descrição e está associado ao valor de *Operand*.

**Description[1..n]**

Define a descrição apresentada para cada radio button. O parâmetro descrição do primeiro radio button é “Description1”, do segundo “Description2”, e assim por diante.

**ATENÇÃO:**

Devem ser definidos uma quantidade de descrições igual a definida no parâmetro *NumElem*.

**BtDescription**

Opcionalmente pode-se desejar enviar para o CP o valor do radio button selecionado uma única vez, quando se clica em um botão específico, e não instantaneamente quando se clica em cada radio button. Para isto, utiliza-se o parâmetro *BtDescription*, que faz com que o applet apresente um botão que faz esta operação.

**ReadOnly**

Este parâmetro define o comportamento somente leitura. Quando valer *true*, o applet apenas monitora do CP o valor dos radio buttons, mas não permite que o usuário faça nenhuma alteração. Neste caso, os radio buttons estarão desabilitados.

Este parâmetro só faz sentido quando o parâmetro *BtDescription* não é declarado.

**Descrição do Funcionamento**

Quando o applet é executado ele apresenta na tela do browser todos os radio buttons e respectivas descrições declarados no HTML através dos parâmetros *NumElem* e *Description[1..n]*.

Em seguida o applet monitora (uma única vez) no CP o valor do operando definido em *Operand*. Conforme o valor monitorado, o radio button correspondente será marcado. É importante ressaltar o fato de que somente um radio por vez poderá estar marcado, sendo, portanto, mutuamente exclusivo. Caso o valor monitorado não corresponda a nenhum radio button (valor negativo, zero ou maior que *NumElem*), então todos os radio buttons estarão desmarcados.

São possíveis três tipos de comportamento deste applet, conforme os parâmetros *BtDescription* e *ReadOnly* :

- Sem Botão (Figura 4-13a)
- Com Botão (Figura 4-13b)
- Sem Botão e Somente Leitura (Figura 4-13c)



### Figura 4-13. Tipos de Comportamento do Applet RadioButton

O applet RadioButton no modo sem botão (sem *BtDescription*, sem *ReadOnly*, conforme mostra a Figura 4-13a) permite que o usuário possa clicar em qualquer radio button, para marca-lo ou desmarca-lo. Ao clicar, o valor correspondente é imediatamente enviado ao CP.

O RadioButton no modo com botão (com *BtDescription*, sem *ReadOnly*, conforme mostra a Figura 4-13b) permite que o usuário possa clicar em qualquer radio button, para marca-lo ou desmarca-lo, porem nenhum valor é enviado ao CP até que o botão especificado por *BtDescription* seja pressionado.

O RadioButton no modo sem botão e somente leitura (sem *BtDescription*, com *ReadOnly*, conforme mostra a Figura 4-13c) não permite ao usuário clicar em nenhum dos radio button. Nenhum valor é enviado ao CP, o applet se comporta como somente leitura.

#### Exemplo de HTML com Uso do RadioButton

Este exemplo mostra um applet RadioButton configurado para 3 radio buttons, no modo sem botão, visualizado na Figura 4-13a.

```
<HTML>
  <applet code="RadioButton.class" archive = " webview.jar " width = 150 height = 80>
    <param name = "Operand" value = "TM3[4]">
    <param name = "NumElem" value = "3">
    <param name = "Description1" value = "Selecionar Motor 1">
    <param name = "Description2" value = "Selecionar Motor 2">
    <param name = "Description3" value = "Selecionar Motor 3">
    <param name="Background" value="celeste">
  </applet>
</HTML>
```

#### Visualização do Exemplo do Applet RadioButton

O exemplo do applet RadioButton pode ser visualizado na Figura 4-13a.

#### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- ComboBox
- CheckBox
- ListBox

#### ComboBox

Applet responsável por apresentar um controle do tipo “combo box” na tela do browser, que realiza a leitura e/ou escrita de valores armazenados em operandos do CP.

#### Parâmetros do ComboBox

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
-----------	-----------	---------	---------------

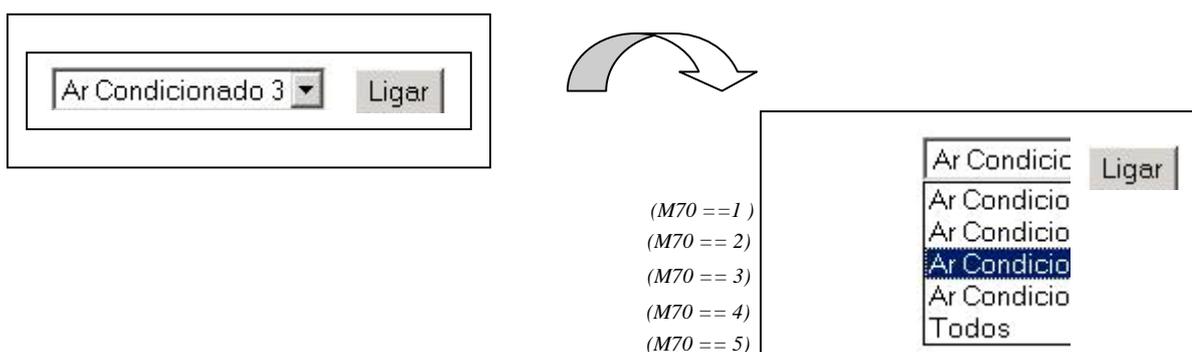
<b>Operand</b>	Especifica o operando a ser lido/escrito no CP	Ex: M7, D40	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito 11 – negrito e itálico	00
Características específicas do Applet			
<b>NumElem</b>	Número de elementos apresentados	De 1 a 256	Não existe
<b>Description [1..n]</b>	Descrição associada a cada elemento		Não existe
BitDescription	Descrição do evento/operação associado ao operando	Ex: OK, Ligar	Não existe
ZeroBase	Valores do CP estão na faixa 0..N-1, em vez de 1..N	Ex: 0/1, V/F, true/false	false

Tabela 4-13. Parâmetros do applet ComboBox

**Operand**

Especifica o operando que vai ser lido e/ou escrito no CP, associado ao combo box. **Este applet é orientado a valor, e não a bit.** O valor 1 de *Operand* corresponde ao primeiro item da combo box, o valor 2 ao segundo item, e assim por diante.

A figura a seguir mostra um esboço de combo box onde o parâmetro *Operand* foi especificado como "M70". Esta figura representa o valor instantâneo de M70 como sendo 3, pois o terceiro item da combo está selecionado.

Figura 4-14. Valores de *Operand* no Applet ComboBox**NumElem**

Define a quantidade de itens apresentados na combo box. Cada item é apresentado com a sua descrição e está associado ao valor de *Operand*.

**Description[1..n]**

Define a descrição apresentada para cada item da combo box. O parâmetro descrição do primeiro item é "Description1", do segundo "Description2", e assim por diante.

**ATENÇÃO:**

Devem ser definidos uma quantidade de descrições igual a definida no parâmetro *NumElem*.

**BtDescription**

Opcionalmente pode-se desejar enviar para o CP o valor selecionado na combo box uma única vez, quando se clica em um botão específico, e não instantaneamente quando se clica em cada item da combo. Para isto, utiliza-se o parâmetro *BtDescription*, que faz com que o applet apresente um botão para esta operação.

**ZeroBase**

Parâmetro booleano opcional que define que os valores do CP estão na faixa 0..N-1. Caso contrário é usado a faixa 1..N. Ao usar esta opção, o valor zero no CP irá corresponder a *Description1* e assim sucessivamente.

*Descrição do Funcionamento*

Quando o applet é executado ele apresenta na tela do browser os itens da combo box conforme suas respectivas descrições declarados no HTML através dos parâmetros *NumElem* e *Description[1..n]*.

Em seguida o applet monitora (uma única vez) no CP o valor do operando definido em *Operand*. Conforme o valor monitorado, o item correspondente da combo box será selecionado. Caso o valor monitorado não corresponda a nenhum item da combo (valor negativo, zero ou maior que *NumElem*), então nenhum item estará selecionado.

É importante ressaltar o fato de que somente um item por vez poderá estar marcado, sendo, portanto, mutuamente exclusivo.

São possíveis dois tipos de comportamento deste applet, conforme o parâmetro *BtDescription*:

- Sem Botão (Figura 4-15a)
- Com Botão (Figura 4-15b)



**Figura 4-15. Tipos de Comportamento do Applet ComboBox**

O applet ComboBox no modo sem botão (sem *BtDescription*, conforme mostra a Figura 4-15a) permite que o usuário possa clicar em qualquer item da combo box para selecioná-lo. Ao clicar, o valor correspondente é imediatamente enviado ao CP.

O ComboBox no modo com botão (com *BtDescription*, conforme mostra a Figura 4-15b) permite que o usuário possa clicar em qualquer item da combo box para selecioná-lo, porém nenhum valor é enviado ao CP até que o botão especificado por *BtDescription* seja pressionado.

Conforme mostrado nas Figuras 4-15a e 4-15b, o applet apresenta apenas o item selecionado. Quando o usuário pressiona o botão , a combo apresenta todos os itens cadastrados, conforme mostra a Figura 4-16.

*Exemplo de HTML com Uso do ComboBox*

Este exemplo mostra um applet ComboBox configurado para 4 itens, no modo sem botão.

```
<HTML>
  <applet code="ComboBox.class" archive = " webview.jar " width = 180 height = 50>
    <param name = "Operand" value = "M50">
```

```

<param name = "NumElem" value = "4">
<param name = "Description1" value = "Ar Condicionado 1">
<param name = "Description2" value = "Ar Condicionado 2">
<param name = "Description3" value = "Ar Condicionado 3">
<param name = "Description4" value = "Habilitar Todos">
<param name="Background" value="celeste">

</applet>
</HTML>

```

### Visualização do Exemplo do Applet ComboBox

O exemplo do applet ComboBox pode ser visualizado na figura a seguir.

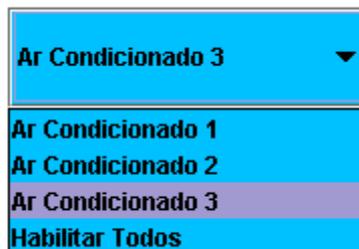


Figura 4-16. Visualização do Exemplo do ComboBox

### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- CheckBox
- RadioButton
- ListBox

### ListBox

Applet responsável por apresentar um controle do tipo “List box” na tela do browser, que realiza a leitura e/ou escrita de valores armazenados em operandos do CP.

### Parâmetros do ListBox

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
<b>Operand</b>	Especifica o operando a ser lido/escrito no CP	Ex: A3, M7, D40	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito	00

		11 – negrito e itálico	
Características específicas do Applet			
BtDescription	Descrição do evento/operação associado ao operando	Ex: Desligar	Não existe
NumElem	Número de elementos que serão apresentados. Cada elemento está relacionado a um bit do operando.	De 1 a 16	Não existe
Description [1..n]	Descrição associada a cada elemento		Não existe
MultiSelect	Indica se o usuário pode selecionar mais de um elemento da lista	0 – única escolha 1 – múltipla escolha	1

Tabela 4-14. Parâmetros do applet ListBox

**Operand**

Especifica o operando que vai ser lido e/ou escrito no CP, associado aos itens do list box. **Cada bit** deste operando corresponde a um item diferente. **Este applet é orientado a bit, e não a valor.**

**NumElem**

Define a quantidade de itens apresentados no list box. Cada item é apresentado com a sua descrição e está associado a um bit de *Operand*. O primeiro item está associado ao bit 0 de *Operand*, o segundo ao bit 1, e assim por diante.

**Description[1..n]**

Define a descrição apresentada para item do list box. O parâmetro descrição do primeiro item é “Description1”, do segundo “Description2”, e assim por diante.

**ATENÇÃO:**

Devem ser definidos uma quantidade de descrições igual a definida no parâmetro *NumElem*.

**BtDescription**

Opcionalmente pode-se desejar enviar para o CP os itens selecionados no list box todos de uma única vez, e não instantaneamente quando se clica em cada item. Para isto, utiliza-se o parâmetro *BtDescription*, que faz com que o applet apresente um botão para esta operação.

**MultiSelect**

Caso o *MultiSelect* esteja definido como false, só pode ser selecionado um elemento por vez, sendo então mutuamente exclusivo. Caso contrário, *MultiSelect* definido como true, pode-se fazer múltiplas seleções de elementos da lista.

**Descrição do Funcionamento**

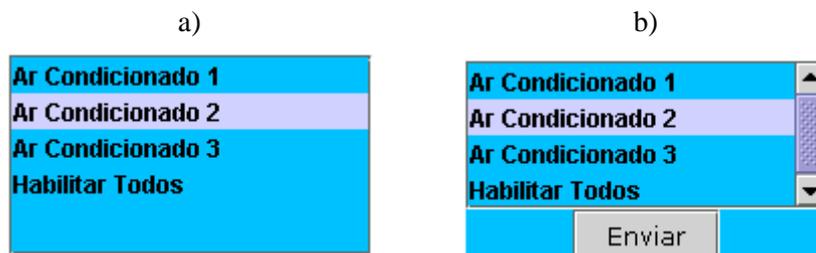
Quando o applet é executado ele apresenta na tela do browser todos os itens do list box conforme suas respectivas descrições declarados no HTML através dos parâmetros *NumElem* e *Description[1..n]*.

Em seguida o applet monitora (uma única vez) no CP o valor do operando definido em *Operand*. Cada bit com valor 1 será apresentado como um item selecionado no list box, e cada bit com valor 0 será apresentado como um item não selecionado.

Cada bit com valor 1 será apresentado no list box como um item selecionado e cada bit com valor 0 será apresentado como um não selecionado.

São possíveis dois tipos de comportamento deste applet, conforme o parâmetro *BtDescription*:

- Sem Botão (Figura 4-17a)
- Com Botão (Figura 4-17b)



**Figura 4-17. Tipos de Comportamento do Applet ListBox**

O applet ListBox no modo sem botão (sem *BtDescription*, conforme mostra a Figura 4-17a) permite que o usuário possa clicar em qualquer item do list box para selecioná-lo. Ao clicar, o valor correspondente é imediatamente enviado ao CP.

O ListBox no modo com botão (com *BtDescription*, conforme mostra a Figura 4-17b) permite que o usuário possa clicar em qualquer item do list box para selecioná-lo, porém nenhum valor é enviado ao CP até que o botão especificado por *BtDescription* seja pressionado.

#### Exemplo de HTML com Uso do ListBox

Este exemplo mostra um applet ListBox configurado para 4 itens, no modo sem botão.

```
<HTML>
  <applet code="ListBox.class" archive = " webview.jar " width = 180 height = 100>
    <param name = "Operand" value = "M50">
    <param name = "NumElem" value = "4">
    <param name = "Description1" value = "Ar Condicionado 1">
    <param name = "Description2" value = "Ar Condicionado 2">
    <param name = "Description3" value = "Ar Condicionado 3">
    <param name = "Description4" value = "Habilitar Todos">
    <param name="Background" value="celeste">
  </applet>
</HTML>
```

#### Visualização do Exemplo do Applet ListBox

O exemplo do applet ListBox pode ser visualizado na Figura 4-17a.

#### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- CheckBox
- RadioButton
- ComboBox

## TimeStamp

Applet responsável por apresentar a data e a hora do CP na tela do browser. A data e hora devem estar armazenadas no CP no formato definido aqui (o mesmo utilizado nas funções relógio F-RELG.048 dos CPs).

### Parâmetros do TimeStamp

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
<b>Operand</b>	Especifica o 1º de 6 operandos a serem lidos	Ex: M7, TM3[4], D40	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito 11 – negrito e itálico	00
Características específicas do Applet			
<b>MaskDate</b>	Máscara para a data (formato de apresentação)	Exs: dd, mm/aaaa	dd/mm/aaaa
<b>MaskHour</b>	Máscara para a hora (formato de apresentação)	Exs: hh:mm, mm:ss	hh:mm:ss

**Tabela 4-15. Parâmetros do applet TimeStamp**

Os seis operandos necessários para leitura da hora e data do controlador programável deverão estar na seguinte seqüência:

- Segundo
- Minuto
- Hora
- Dia
- Mês
- Ano

Esta seqüência é a mesma utilizada no módulo função para acesso ao relógio de tempo real do CP, F-RELG.048.

O parâmetro *Operand* informado no HTML corresponde ao operando dos segundos. Os demais 5 valores são lidos dos operandos consecutivos.

O dia, mês, ano, hora, minuto e segundo são apresentados com no mínimo dois dígitos. Exemplo certo: 26/09/03 08:03 . Exemplo errado: 26/9/3 8:03 .

O applet pode apresentar o ano com 4 dígitos, Para isto, basta que o operando correspondente no CP contenha o valor 2003, por exemplo. Então o applet irá mostrar o ano com 4 dígitos (por exemplo, 26/09/2003 08:03).

Caso algum dos operandos não possa ser lido do CP por qualquer razão, o string "?" será apresentado no interior do applet.

### MaskDate

Especifica o formato da máscara da data que será exibida, sendo o formato default com dois dígitos para o dia e o mês e quatro dígitos para o ano.

### MaskHour

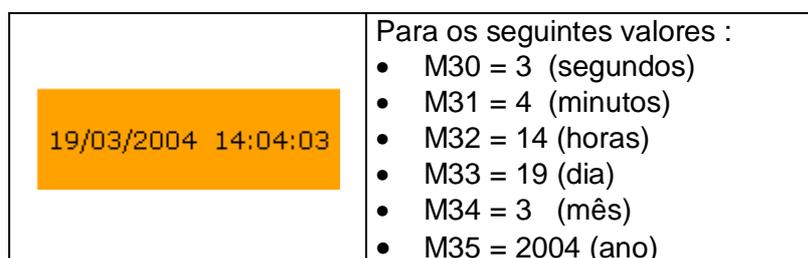
Especifica o formato da máscara da hora que será exibida, sendo o formato default com dois dígitos para hora, minuto e segundo.

### Exemplo de HTML com Uso do TimeStamp

```
<HTML>
  <applet code = "TimeStamp.class" archive = " webview.jar " width = "150" height = "50">
    <param name = "Operand" value = "M30">
    <param name = "frequency" value="3">
    <param name = "background" value = "laranja">
  </applet>
</HTML>
```

### Visualização do Exemplo do Applet TimeStamp

A figura a seguir mostra o exemplo do TimeStamp para os seis valores de operandos M lidos do CP. A cada 3 segundos estes mesmos operando são lidos novamente, e o applet é redesenhado.



**Figura 4-18. Visualização do Exemplo do TimeStamp**

### BarGraph

Applet responsável por apresentar uma barra de nível em função do valor de um operando lido do CP.

### Parâmetros do BarGraph

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
Operand	Especifica o operando a ser lido do CP	Ex: M7, TM3[4], D40	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco

Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito 11 – negrito e itálico	00
Características específicas do Applet			
Description	Título do BarGraph		Em branco
Div	Número de divisões na escala apresentada na tela (entre MinEng e MaxEng)	De 2 a 16	A barra de escala não é apresentada
BarForeColor	Cor da barra	Ex: Vermelho	Preto
BarBackColor	Cor de fundo da barra	Ex: Branco	Branco
FullSize	Indica que a barra deve ocupar toda a área do applet	Ex: true, false	False
Escala de Engenharia			
MinCP	Valor mínimo do operando no CP	Ex: 10	Não existe
MaxCP	Valor máximo do operando no CP	Ex: 30	Não existe
MinEng	Valor mínimo relativo do operando	Ex: 100	0
MaxEng	Valor máximo relativo do operando	Ex: 1800	100

Tabela 4-16. Parâmetros do applet BarGraph

**MinCP, MaxCP, MinEng, MaxEng**

Os parâmetros MinCP, MaxCP, MinEng, MaxEng são utilizados para converter o valor de um operando do CP para unidades de engenharia. São opcionais. Se algum deles for omitido no HTML ou for inválido, nenhuma conversão será feita, apresentando o mesmo valor lido no CP.

**Div**

A barra de nível apresenta um intervalo de valores a serem visualizados, sendo o valor inicial correspondente a *MinEng* e o valor final a *MaxEng*. A barra é apresentada dividida em *Div* partes.

**Background, BarForeColor, BarBackColor**

O parâmetro background refere-se à cor ou imagem de fundo do Applet inteiro. A cor da barra e sua cor de fundo podem ser configurada por *BarForeColor* e *BarBackColor*, respectivamente.

**Exemplo de HTML com Uso do BarGraph**

```

<HTML>
  <APPLET code= "BarGraph.class" archive=" webview.jar " width = "100" height = "150">
    <param name = "Operand" value = "M806">
    <param name = "frequency" value = "1">
    <param name = "fontColor" value = "verde">
    <param name = "barForeColor" value = "vermelho">
    <param name = "div" value = "4">
    <param name = "mincp" value = "0">
    <param name = "maxcp" value = "100">
    <param name = "mineng" value = "0">
    <param name = "maxeng" value = "100">
  </APPLET>
</HTML>

```

### Visualização do Exemplo do Applet BarGraph

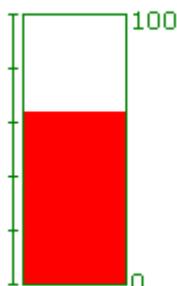


Figura 4-19. Visualização do Exemplo do BarGraph

### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- PenChart

### PenChart

Applet responsável por apresentar um gráfico na tela, a partir de um conjunto de valores lidos do CP. Cada valor lido corresponderá a um ponto plotado no gráfico. A quantidade de pontos é configurada no applet.

É possível configurar o applet para que o gráfico se atualize automaticamente, através do parâmetro “Frequency”. Existem dois modos de operação distintos: **modo XY** e **modo PenMode**. No modo XY, a cada atualização todos os valores do gráfico são lidos do CP de uma única vez, e o gráfico inteiro é redesenhado. No modo PenMode, a cada atualização apenas um valor é lido do CP. Neste instante todo o gráfico é deslocado de um ponto para a esquerda, o valor mais a esquerda é descartado e o novo valor é plotado na posição mais a direita.

### Parâmetros do PenChart

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
Operand	Especifica o 1º operando a ser lido	Ex: M7, TM3[4], D40	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito 11 – negrito e itálico	00
Características específicas do Applet			
Description	Título do gráfico	“Nível do Tanque”	Em branco
DescriptionX	String associada ao eixo x	Ex: tempo, seg	Em branco
DescriptionY	String associada ao	Ex: Vazão mm3	Em branco

	eixo y		
IntervalX	Valor inicial e final do eixo x	Ex: 0-1800	0-1000
GraphColor	Cor do Gráfico		Preto
AxisColor	Cor dos Eixos		Mesmo que GraphColor
DivX	Número de divisões na escala do eixo x	De 0 a 16	4
DivY	Número de divisões na escala do eixo y	De 0 a 16	5
QtdValores	Quantidade de pontos analógicos a serem lidos do CP e plotados no gráfico	De 2 a 64	50
PenMode	Modo de operação: XY ou PenMode.	True / False	False
Escala de Engenharia			
MinCP	Valor mínimo do operando no CP	Ex: 0	Não existe
MaxCP	Valor máximo do operando no CP	Ex: 4096	Não existe
MinEng	Valor mínimo relativo do operando	Ex: 0	Não existe
MaxEng	Valor máximo relativo do operando	Ex: 800	Não existe

Tabela 4-17. Parâmetros do applet PenChart

**Description, DescriptionX, DescriptionY**

Esses parâmetros são opcionais. O *Description* define um título para o gráfico. Os parâmetros *DescriptionX* e *DescriptionY* definem a unidade dos eixos X e Y, respectivamente.

Se algum deles for omitido no HTML, apenas deixará de aparecer a descrição correspondente no gráfico.

**MinCP, MaxCP, MinEng, MaxEng**

Os parâmetros MinCP, MaxCP, MinEng, MaxEng são utilizados para converter o valor de um operando do CP para unidades de engenharia. São opcionais. Se algum deles for omitido no HTML ou for inválido, nenhuma conversão será feita, apresentando o mesmo valor lido no CP.

**IntervalX, DivX, DivY**

O *IntervalX* fornece o intervalo a ser mostrado no eixo x da variável *DescriptionX*, com o parâmetro *DivX* dividindo esse intervalo na escala de seu eixo. No caso do eixo y, o intervalo é obtido utilizando o *MinEng* como o valor inicial do eixo y, o *MaxEng* como o valor final do eixo y e o intervalo criado pelos esses dois últimos parâmetros é dividido em *DivY* partes.

**QtdValores**

De acordo com o valor fornecido nesse parâmetro, serão lidos *QtdValores* do CP, a partir do valor inicial do *Operand*, para serem plotados no gráfico. Caso *QtdValores* seja definido como 40 e o *Operand* parametrizado como M100, então os pontos lidos no CP serão dos operandos M100 ao M139, totalizando 40 pontos plotados no gráfico.

**Exemplo de HTML com Uso do PenChart**

Este exemplo mostra um gráfico de 26 valores armazenados no CP, a partir do operando M902 (M902 a M927, portanto). Estes valores são lidos e plotados novamente a cada dois segundos.

```
<HTML>
  <APPLET code="PenChart.class" archive=" webview.jar " width=500 height=350>
  <PARAM name = "Operand" value = "M902">
```

```

<PARAM name = "frequency" value="2">
<PARAM name = "QtdValores" value = "26">
<PARAM name = "background" value = "Palha">
<PARAM name = "graphColor" value = "Azul">
<PARAM name = "axisColor" value = "Verde">
<PARAM name = "fontColor" value = "Vermelho">
<param name = "fontStyle" value = "10">
<PARAM name = "fontSize" value = "40">
<PARAM name = "description" value = "Vazão de Gás">
<PARAM name = "descriptionX" value = "Seg.">
<PARAM name = "descriptionY" value = "G/h">
<PARAM name = "DivX" value = "5">
<PARAM name = "DivY" value = "5">
<PARAM name = "IntervalX" value = "0-500">
</APPLET>
</HTML>

```

### Visualização do Exemplo do Applet PenChart

A figura a seguir mostra o exemplo do PenChart em execução. Os valores apresentados no gráfico dependem do valor instantâneo dos operando no CP.



Figura 4-20. Visualização do Exemplo do PenChart

### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- BarGraph

### EventSummary

Este applet tem por objetivo mostrar na tela do browser uma lista de eventos. Cada evento ocupa uma linha contendo sua descrição e o horário de registro. A lista de eventos ativos e o horário de cada um ficam armazenados em operandos no CP.

Hora (hh:mm)	Alarme(s)
20:35	Motor A Ligado
10:15	Válvula Y aberta
05:05	Motor A Desligado
02:10	Válvula Y fechada

Figura 4-21. Esboço do EventSummary

**ATENÇÃO:**

Este applet depende de programa aplicativo no CP para funcionar corretamente.

*Parâmetros do EventSummary*

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
<b>Operand</b>	Especifica o 1º operando M ou TM a ser lido	Operandos M ou TM Ex: M7, TM2, TM3[4]	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito 11 – negrito e itálico	00
Características específicas do Applet			
<b>Description[1..n]</b>	Descrição do evento associado ao bit correspondente do operando		Não existe
<b>NumEvents</b>	Número de eventos cadastrados	De 1 a 64	16
<b>OperTime</b>	Operando associado ao horário em que o evento aconteceu	Operandos M ou TM Ex: M8	Não existe

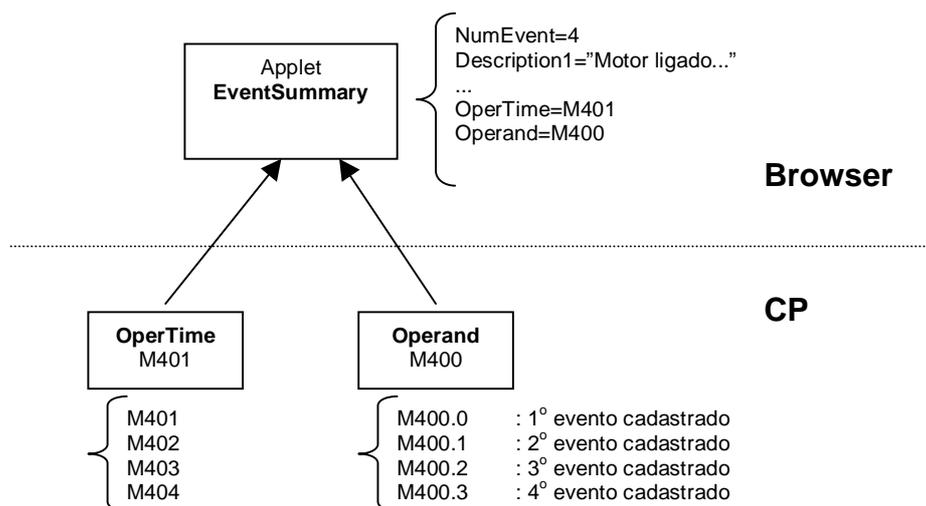
Tabela 4-18. Parâmetros do applet EventSummary

**Arquitetura de Funcionamento**

Para que este applet funcione corretamente, algumas configurações devem ser feitas na página HTML (nos parâmetros deste applet) e outras devem ser feitas no programa aplicativo do CP.

No HTML devem ser configurados a quantidade de eventos possíveis, o texto de cada um e os operandos do CP correspondentes, através dos parâmetros *NumEvents*, *Description[1..n]*, *Operand*, e *OperTime*.

No programa aplicativo devem ser alocados os operandos do CP correspondentes a *Operand* e *OperTime*, e deve ser programado as lógicas de ativação e armazenamento do horário do evento. O programa exemplo deste Applet, disponível no CDROM deste produto, apresenta uma tela HTML e um programa aplicativo correspondente, desenvolvido no MasterTool para UCPs AL-2002 (pode ser convertido para qualquer outra UCP).



**Figura 4-22: Arquitetura de Funcionamento do EventSummary**

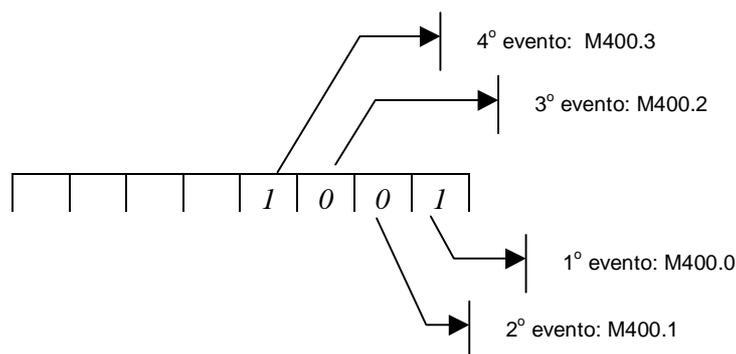
A Figura 4-22 representa um applet EventSummary declarado com 4 eventos e os respectivos operandos do CP. O applet, que executa no browser, lê duas faixas de operandos do CP: as faixas que começam pelos operandos declarados em *Operand* e *OperTime*. Este applet não escreve nenhum operando no CP.

O programa aplicativo do CP é quem deve gerar os bits de eventos (faixa *Operand*) e os respectivos horários (faixa *OperTime*). O applet apenas apresenta os valores presentes no CP.

### Descrição do Funcionamento

Ao ser executado na tela HTML, o applet monitora o CP e apresenta apenas os eventos ativos. Como o applet sabe se um evento está ativo ou não? Através do bit de status associado ao evento. Cada evento cadastrado possui um bit de status, alocados a partir do bit zero do parâmetro *Operand*. O bit em 1 indica que o evento está ativo.

Por exemplo, se *Operand* for configurado como "M400", o bit M400.0 corresponde ao status do primeiro evento cadastrado, o bits M400.1 ao segundo, etc. Se o valor instantâneo de M400 no CP for 9 (1001 em binário), teremos os seguintes bits de eventos ativos:



**Figura 4-23: Bits de Evento no EventSummary**

A Figura 4-23 mostra que o primeiro evento cadastrado está ativo, o segundo e terceiro não estão ativos e o 4º está ativo.

O limite máximo de eventos cadastrados está especificado na Tabela 4-18. Caso o tamanho (em bits) do operando informado pelo parâmetro *Operand* seja insuficiente para armazenar todos os bits de eventos, então um (ou mais) operandos consecutivos serão utilizados para este fim, conforme resume a tabela a seguir:

Quantidade de Eventos Cadastrados	Quantidade de Operandos necessários
	Tipo M ou posições de TM
De 1 a 8	1
De 9 a 16	1
De 17 a 24	2
De 25 a 32	2
De 33 a 40	3
De 41 a 48	3
De 49 a 56	4
De 57 a 63	4

### Horário dos Eventos

Para os eventos ativos, o applet apresenta também o horário do evento, armazenado no CP a partir do operando determinado pelo parâmetro *OperTime*. Caso este parâmetro não esteja definido no HTML, a coluna “Hora” será omitida.

O programa aplicativo do CP é responsável por armazenar o horário de ativação do evento nos operandos correspondentes. O horário deve ser armazenado no formato HH:MM, de dois bytes. Tipicamente utiliza-se um operando M para cada horário de alarme, ficando a hora no byte mais significativo e o minuto no menos significativo.

### NumEvents

Define a quantidade de eventos cadastrados no applet.

### Description[1..NumEvents]

Descrição de cada evento. Este texto é apresentado pelo applet quando o evento correspondente estiver ativo. Devem ser definidos *NumEvents* descrições.

### Operand

Operando inicial do CP que armazena os par de bits de status de cada alarme.

### OperTime

Operando inicial do CP que armazena o horário de cada evento. Cada evento tem um horário associado. Especifica-se o operando que se encontra o horário do primeiro evento e os outros serão lidos sequencialmente de acordo com o parâmetro *NumEvents*.

**ATENÇÃO:**

Recomenda-se utilizar operandos do tipo memória nos parâmetros *Operand* e *OperTime*.

### Exemplo de HTML com Uso do EventSummary

Este exemplo mostra um applet *EventSummary* configurado para apresentar 4 eventos. Utilizou-se aqui outro applet, *CheckBox*, para simular o acontecimento dos eventos no processo.

```
<HTML>
  <applet code="EventSummary.class" width="380" height="150" archive="webview.jar">
    <param name="Operand" value="M400">
    <param name="OperTime" value="M401">
    <param name="NumEvents" value="4">
    <param name="Description1" value="Evento 1: Motor A Ligado">
    <param name="Description2" value="Evento 2: Motor A Desligado">
    <param name="Description3" value="Evento 3: Válvula Y aberta">
    <param name="Description4" value="Evento 4: Válvula Y fechada">
    <param name="Background" value="celeste">
```

```

        <param name="Frequency" value="2">
    </applet>
    <p>Ativação dos Eventos (simulando o processo)</p>
    <applet code="CheckBox.class" width="150" height="100" archive=" webview.jar ">
        <param name="Operand" value="A51">
        <param name="NumElem" value="4">
        <param name="Description1" value="Alarme 1">
        <param name="Description2" value="Alarme 2">
        <param name="Description3" value="Alarme 3">
        <param name="Description4" value="Alarme 4">
    </applet>
</HTML>

```

Para que este exemplo funcione, é preciso carregar no CP o programa aplicativo correspondente, disponível no CDRom deste produto, no diretório “Exemplos\Applets\EventSummary\ladder”. Este programa aplicativo foi desenvolvido no MasterTool para UCPs AL-2002, mas pode ser convertido e utilizado em qualquer outra UCP Altus.

### Visualização do Exemplo do Applet EventSummary

A figura a seguir mostra o exemplo do EventSummary em execução, com 2 eventos ativos sendo apresentados.

Para gerar um dos 4 eventos, basta clicar no check box de simulação correspondente, e deve-se visualizar o evento gerado no applet AlarmSummary.

Horas(hh:mm)	Evento(s)
18:27	Evento 2: Motor A Desligado
18:28	Evento 4: Válvula Y fechada

Ativação dos Eventos (simulando o processo)

Alarme 1  
 Alarme 2  
 Alarme 3  
 Alarme 4

Figura 4-24. Visualização do Exemplo do EventSummary

### Applets Relacionados

O applet a seguir pode ter função ou parâmetros semelhantes, podendo facilitar o entendimento do applet descrito.

- AlarmSummary

## AlarmSummary

Este applet tem por objetivo mostrar na tela do browser uma lista de alarmes ativos. Cada alarme ocupa uma linha, contendo a descrição do alarme, o horário e um botão de reconhecimento, opcional. Os alarmes não reconhecidos aparecem em cor vermelha, e os já reconhecidos em verde. A lista de alarmes ativos, o status reconhecido/não reconhecido e o horário de cada um ficam armazenados em operandos no CP.

ACK	Hora (hh:mm)	Alarme(s)
<input type="checkbox"/>	20:35	Nível do Tanque A Muito Alto
<input checked="" type="checkbox"/>	10:15	Nível do Tanque B Muito Baixo
<input checked="" type="checkbox"/>	05:05	Temperatura Alta Motor
<input type="checkbox"/>	02:10	Temperatura Muito Alta Motor 2

Figura 4-25. Esboço do AlarmSummary

### ATENÇÃO:

Este applet depende de programa aplicativo no CP para funcionar corretamente.

## Parâmetros do AlarmSummary

A seguir encontram-se os parâmetros deste applet. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de valores possíveis e o valor default.

Parâmetro	Descrição	Valores	Valor Default
<b>Operand</b>	Especifica o 1º operando M ou TM a ser lido com o par de bits de status de cada alarme	Operandos M ou TM Ex: M7, TM2, TM3[4]	Não existe
Frequency	Tempo de polling em segundos	De 0 a 600	0 (zero)
Background	Define a cor ou imagem de background	Ex: Verde, back.gif	Branco
Características relativas à Fonte			
FontType	Tipo da fonte	Ex: Arial, Times	Verdana
FontSize	Tamanho da fonte	De 8 a 32	12
FontColor	Cor da fonte	Ex: FFFFFFFF, cinza	Preto
FontStyle	Estilo da fonte: negrito e/ou itálico.	00 – normal 01 – itálico 10 – negrito 11 – negrito e itálico	00
Características específicas do Applet			
<b>Description[1..n]</b>	Descrição dos alarmes associados ao operando		Não existe
<b>NumAlarms</b>	Número de alarmes cadastrados	De 1 a 64	8
<b>OperTime</b>	Operando contendo o horário do 1º alarme cadastrado. Os demais horários são lidos dos operandos subsequentes.	Operandos M ou TM Ex: M6	Não existe
<b>OperAck</b>	Operando Auxiliar com os bits de reconhecimento de	Operando do tipo Auxiliar. Ex: A50	Não existe

	alarmes, forçados para 1 quando check box de ack for pressionado		
--	--	--	--

Tabela 4-19. Parâmetros do applet AlarmSummary

### Arquitetura de Funcionamento

Para que este applet funcione corretamente, algumas configurações devem ser feitas na página HTML (nos parâmetros deste applet) e outras devem ser feitas no programa aplicativo do CP.

No HTML devem ser configurados a quantidade de alarmes possíveis, o texto de cada um e os operandos do CP correspondentes, através dos parâmetros *NumAlarms*, *Description[1..n]*, *Operand*, *OperTime* e *OperAck*.

No programa aplicativo devem ser alocados os operandos do CP correspondentes a *Operand*, *OperTime* e *OperAck*, e deve ser programado as lógicas de ativação, reconhecimento e armazenamento do horário do alarme. O programa exemplo deste applet, disponível no CDROM deste produto, apresenta uma tela HTML e um programa aplicativo correspondente, desenvolvido no MasterTool para UCPs AL-2002 (pode ser convertido para qualquer outra UCP).

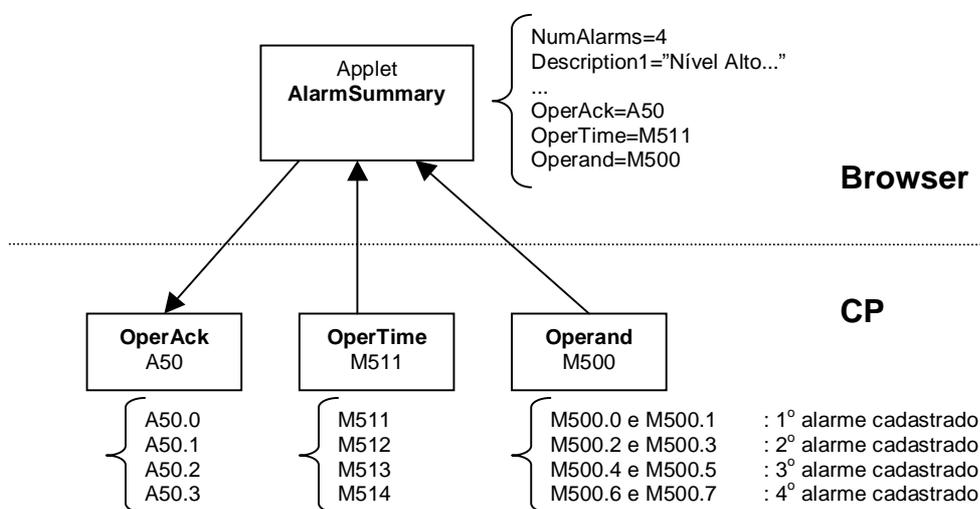


Figura 4-26: Arquitetura de Funcionamento do AlarmSummary

A Figura 4-26 representa um applet AlarmSummary declarado com 4 alarmes e os respectivos operandos do CP. O applet, que executa no browser, lê duas faixas de operandos do CP e escreve uma: as faixas que começam pelos operandos declarados em *Operand*, *OperTime* (lidas pelo applet) e *OperAck* (escrita pelo applet), respectivamente.

O programa aplicativo do CP é quem deve gerar os bits de status dos alarmes (faixa *Operand*) e os respectivos horários (faixa *OperTime*). O applet apenas apresenta os valores presentes no CP. O programa aplicativo deve também monitorar o valor dos operandos de reconhecimento de alarme (faixa *OperAck*), que são forçados para o valor 1 pelo applet, quando o usuário pressiona o check-box de reconhecimento.

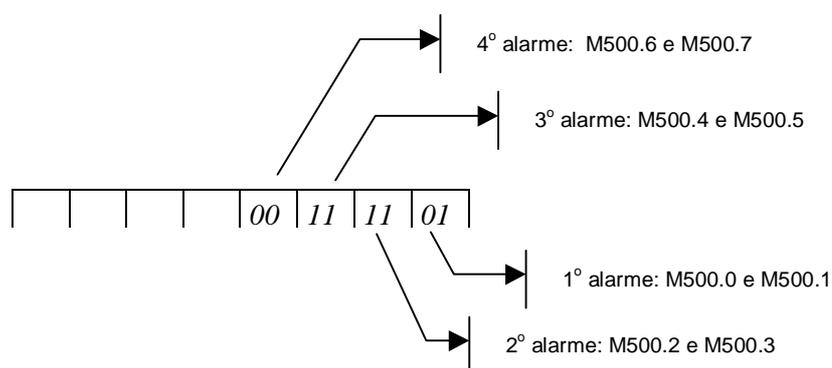
### Descrição do Funcionamento

Ao ser executado na tela HTML, o applet monitora o CP e apresenta apenas os alarmes ativos. Como o applet sabe se um alarme está ativo ou não? Através de dois bits de status associados ao alarme. Cada alarme cadastrado possui estes dois bits de status, alocados a partir do bit zero do parâmetro *Operand*. O primeiro bit indica se o alarme está ativo (bit em 1) ou não e o segundo indica se aquele alarme ativo já foi reconhecido (bit em 1) ou não, conforme mostra a tabela a seguir:

2º Bit	1º Bit	Status do Alarme	Apresentação na tela
0	0	Não ativo, não reconhecido	Não apresentado
0	1	Ativo, não reconhecido	Descrição do alarme em vermelho
1	0	Situação inválida	Não apresentado
1	1	Ativo, já reconhecido	Descrição do alarme em verde

**Tabela 4-20: Bits de cada Alarme em AlarmSummary**

Por exemplo, se *Operand* for configurado como “M500”, os bits M500.0 e M500.1 corresponderão ao status do primeiro alarme cadastrado, os bits M500.2 e M500.3 ao segundo, etc. Se o valor instantâneo de M500 no CP for 61 (00 11 11 01 em binário), teremos os seguintes bits de status por alarme:



**Figura 4-27: Bits de Status de cada Alarme no AlarmSummary**

A Figura 4-27 mostra que o primeiro alarme cadastrado está ativo e não reconhecido (01), o segundo e terceiro estão ativos e reconhecidos (11) e o 4º não está ativo (00). Neste caso, os alarmes ativos e reconhecidos (2º e 3º) são apresentados em cor verde e os ativos e não reconhecidos (1º) são apresentados em cor vermelha, conforme representado na Figura 4-25.

O limite máximo de alarmes cadastrados está definido na Tabela 4-19. Caso o tamanho (em bits) do operando informado pelo parâmetro *Operand* seja insuficiente para armazenar todos os bits de alarmes, então um (ou mais) operandos consecutivos serão utilizados para este fim, conforme resume a tabela a seguir:

Quantidade de Alarmes Cadastrados	Quantidade de Operandos necessários
	Tipo M ou posições de TM
De 1 a 4	1
De 5 a 8	1
De 9 a 12	2
De 13 a 16	2
De 17 a 20	3
De 21 a 24	3
De 25 a 28	4
...	...
De 61 a 64	8

### Horário dos Alarmes

Para os alarmes ativos, o applet apresenta também o horário do alarme, armazenado no CP a partir do operando determinado pelo parâmetro *OperTime*. Caso este parâmetro não esteja definido no HTML, a coluna “Hora” será omitida.

O programa aplicativo do CP é responsável por armazenar o horário de ativação do alarme nos operandos correspondentes. O horário deve ser armazenado no formato HH:MM, de dois bytes. Tipicamente utiliza-se um operando M para cada horário de alarme, ficando a hora no byte mais significativo e o minuto no menos significativo.

### Reconhecimento dos Alarmes

Ao carregar, o applet apresentará, para cada alarme ativo, o status de já reconhecido ou ainda não reconhecido, de acordo com o segundo bit de status de cada alarme. Este status pode ser visualizado com uma marca no check box de cada alarme na coluna “ACK” e também pela cor do texto do alarme na coluna “Alarme(s)”.

Para reconhecer um alarme ativo, deve-se clicar no check box correspondente. Ao reconhecer um alarme, o applet escreve o valor 1 em um bit do CP determinado pelo parâmetro *OperAck*.

#### ATENÇÃO:

Os dois bits de status dos alarmes são sempre atualizados apenas pelo programa aplicativo do CP (e nunca pelo applet). Quando o usuário reconhece o(s) alarme(s), faz com o que o CP realize uma varredura nos bits de status de cada alarme para atualizar os mesmos. Se a situação de alarme ainda existe o par de bits recebe o valor 11, caso contrário recebe o valor 00.

### NumAlarms

Define a quantidade de alarmes cadastrados no applet.

#### Description[1..NumAlarms]

Descrição de cada alarme. Este texto é apresentado pelo applet quando o alarme correspondente estiver ativo. É apresentado em cor verde se já foi reconhecido, ou em cor vermelha se ainda não. Devem ser definidos *NumAlarms* descrições.

### Operand

Operando inicial do CP que armazena os par de bits de status de cada alarme.

### OperTime

Operando inicial do CP que armazena o horário de cada alarme. Cada alarme tem um horário associado. Especifica-se o operando que se encontra o horário do primeiro alarme e os outros serão lidos seqüencialmente de acordo com o parâmetro *NumAlarms*.

### OperAck

Operando inicial do CP utilizado no reconhecimento de alarmes. Cada bit deste operando está associado a um alarme. Quando o usuário pressiona o check-box de reconhecimento de alarme, o applet força para zero o bit correspondente ao alarme sendo reconhecido.

#### ATENÇÃO:

Recomenda-se utilizar operandos do tipo memória nos parâmetros *Operand* e *OperTime* e do tipo auxiliar no parâmetro *OperAck*.

### Exemplo de HTML com Uso do AlarmSummary

Este exemplo mostra um applet AlarmSummary configurado para apresentar 4 alarmes. Utilizou-se aqui outro applet, CheckBox, para simular o acontecimento dos alarmes no processo.

```
<HTML>
  <applet code="AlarmSummary.class" width="400" height="150" archive=" webview.jar ">
    <param name="Operand" value="M500">
    <param name="OperTime" value="M511">
```

```

<param name="OperAck" value="A50">
<param name="NumAlarms" value="4">
<param name="Description1" value="Alarme 1: ALM Nível muito alto">
<param name="Description2" value="Alarme 2: ALM Nível Alto">
<param name="Description3" value="Alarme 3: ALM Nível baixo">
<param name="Description4" value="Alarme 4: ALM Nível muito Baixo">
<param name="Background" value="celeste">
<param name="Frequency" value="2">
</applet>
<p>Ativação dos Alarmes (simulando o processo)</p>
<applet code="CheckBox.class" width="150" height="100" archive=" webview.jar ">
  <param name="Operand" value="A51">
  <param name="NumElem" value="4">
  <param name="Description1" value="Alarme 1">
  <param name="Description2" value="Alarme 2">
  <param name="Description3" value="Alarme 3">
  <param name="Description4" value="Alarme 4">
</applet>
</HTML>

```

Para que este exemplo funcione, é preciso carregar no CP o programa aplicativo correspondente, disponível no CDROM deste produto, no diretório “Exemplos\Applets\AlarmSummary\ladder”. Este programa aplicativo foi desenvolvido no MasterTool para UCPs AL-2002, mas pode ser convertido e utilizado em qualquer outra UCP Altus.

#### Visualização do Exemplo do Applet AlarmSummary

A figura a seguir mostra o exemplo do AlarmSummary em execução, com 4 alarmes ativos sendo apresentados.

Para gerar um dos 4 alarmes, basta clicar no check box de simulação correspondente, e deve-se visualizar o alarme gerado no applet AlarmSummary. Pode-se também reconhecer os alarmes ativos, e depois desmarcar o check box de simulação para fazer com que o AlarmSummary retire o alarme da lista apresentada.

ACK	Horas(hh:mm)	Evento(s)
<input type="checkbox"/>	11:16	Alarme 1: ALM Nível muito alto
<input checked="" type="checkbox"/>	11:11	Alarme 2: ALM Nível Alto
<input checked="" type="checkbox"/>	11:10	Alarme 3: ALM Nível baixo
<input type="checkbox"/>	11:09	Alarme 4: ALM Nível muito Baixo

Ativação dos Alarmes (simulando o processo)

Alarme 1

Alarme 2

Alarme 3

Alarme 4

### Figura 4-28. Visualização do Exemplo do AlarmSummary

#### Applets Relacionados

Os applets a seguir possuem função ou parâmetros semelhantes a este:

- EventSummary

#### Variável Local

A Variável Local não é exatamente um applet, é um recurso que permite que os applets acessem variáveis armazenadas no próprio microcomputador, em vez de no CP. O comportamento dos applets WebView continua o mesmo, apenas o local de armazenamento do valor lido ou escrito é que passa a ser no microcomputador em vez de ser no CP.

Para todos os applets do sistema WebView é permitido o uso de Variáveis Locais. Estas podem ser usadas para armazenamento temporário, uma vez que não forçam valores na CP, e têm seus valores compartilhados por todos os applets presentes em uma mesma página HTML.

Para indicar que o applet deve acessar os operandos como uma variável local, deve-se acrescentar o caracter “#” antes do endereço do operando, no parâmetro Operand, da seguinte forma: [#][nome\_variável]. Neste caso, qualquer nome de variável pode ser utilizado, e não apenas os nomes válidos de operandos do CP. Exemplos: “#nome\_receita”, “#numero\_maquina” .

As variáveis locais podem ser usadas junto com o recurso de Receitas para permitir a parametrização de arquivos HTML.

#### Exemplo de HTML com Uso de Variável Local

```
<HTML>
  Digite o valor da variável local "#Aux" : <p>
  <applet code= "AnalogOutput.class" archive=" webview.jar " width=150 height=50>
    <param name = "Operand" value = "#Aux">
  </applet>
  <p>Ou selecine um destes valores :<p>
  <applet code="RadioButton.class" width="150" height="100" archive=" webview.jar ">
    <param name="Operand" value="#Aux">
    <param name="NumElem" value="4">
    <param name="Description1" value="Valor 1">
    <param name="Description2" value="Valor 2">
    <param name="Description3" value="Valor 3">
    <param name="Description4" value="Valor 4">
  </applet>
  <p>E verifique sua apresentação :<p>
  <applet code="AnalogInput" width="150" height="50" archive=" webview.jar ">
    <param name="Operand" value="#Aux">
    <param name="frequency" value="1">
  </applet>
</HTML>
```

Neste exemplo tem-se o uso da Variável Local “#Aux” nos applets AnalogOutput, RadioButton e AnalogInput.

## Receita

Uma Receita não é um applet, é um recurso que permite uma forma alternativa de parametrização dos demais applets. Na forma convencional, os applets e seus respectivos parâmetros ficam armazenados nos arquivos HTML. Com a receita, os parâmetros a serem informados para cada applet são os mesmos, porém ficam armazenados em outro arquivo, independente do HTML, no formato XML.

Para criar páginas com uso de Receita, ou seja, para indicar que o applet deve ler os valores dos parâmetros através de um arquivo XML de receita, externo ao HTML, basta utilizar o parâmetro “RecipeFile” no HTML. Neste caso, o único parâmetro informado no HTML é “RecipeFile”, que define de onde os valores dos demais parâmetros do applet devem ser lidos:

- RecipeFile não informado: o applet procura os parâmetros no mesmo arquivo HTML, junto com o applet.
- RecipeFile informado: o applet procura os parâmetros no arquivo XML indicado por RecipeFile.

Por exemplo, para indicar que os parâmetros de um applet AnalogInput devem ser lidos do arquivo de receita chamado “receita1.xml”, basta configurar o parâmetro “RecipeFile” deste applet como “receita1.xml”.

### Parâmetros do Receita

A seguir encontram-se os parâmetros do Receita. A tabela apresenta para cada parâmetro uma descrição sucinta, exemplos de seus possíveis valores e o valor default, usado no caso do parâmetro não ter um valor definido no HTML.

Parâmetro	Descrição	Valores	Valor Default
RecipeFile	Nome do arquivo XML com receita do respectivo applet. Este arquivo fica no WebGate juntamente com o webview.jar.	Ex: receita.xml	Não existe

**Tabela 4-21. Parâmetros do Receita**

Este recurso permite que se utilize uma mesma tela HTML para apresentar diferentes conjuntos de operandos do CP. Isto pode ser útil em telas de tamanho relativamente grande (uma tela com figuras, por exemplo), devido às limitações de memória do WebGate.

Ao projetar uma tela HTML baseada em receita, deve-se agrupar cada conjunto de parâmetros variável em um único arquivo de parametrização, chamado de receita. Podem existir um ou vários arquivos de receita, para uma mesma tela HTML. Na carga da tela, o(s) arquivo(s) de receita (com os parâmetros) especificado(s) no HTML (por RecipeFile) também é(são) carregado(s) do WebGate, e a parametrização dos applets é feita a partir deste.

O arquivo XML com a parametrização da receita é um arquivo texto padrão ASCII. Ele deve ser criado pelo usuário, usando qualquer editor de texto de mercado, tal como o Notepad. O arquivo deve usar o formato do seguinte exemplo:

```

<?xml version="1.0" ?>
<RecipeFile>
  <Applet name="AI001">
    <Operand>M808</Operand>
    <Frequency>1</Frequency>
    <Description>M808</Description>
  </Applet>
  <Applet name="AI002">
    <Operand>M818</Operand>
    <Frequency>1</Frequency>
    <Description>M818</Description>
  </Applet>
</RecipeFile>

```

**Figura 4-29. Formato do Arquivo XML de Receita**

Este arquivo XML deve conter todos os parâmetros do applet. Quando se usa receita, nenhum parâmetro exceto RecipeFile é lido do HTML. O “nome\_do\_applet” corresponde ao parâmetro “name” informado no HTML do applet.

Os passos executados pelo browser durante a carga de tela HTML até a leitura do arquivo XML de receita são:

- 1) Browser inicia processamento de um tag applet no HTML;
- 2) Browser verifica se existe no HTML o parâmetro ReceifFile para este applet:
  - 3) não existe: lê os parâmetros do próprio HTML;
  - 4) existe:
    - 5) o applet carrega o arquivo especificado em RecipeFile do WebGate (uma única vez);
    - 6) o applet lê o valor dos parâmetros a partir deste arquivo (uma vez p/ cada parâmetro).

O arquivo XML de receita é apenas lido pelo applet, nunca é modificado. Para agrupar parâmetros de vários applets em um mesmo arquivo HTML, deve-se informar parâmetros “name” diferentes em cada applet, no arquivo HTML, e referenciar-se estes mesmos “names” no arquivo XML.

### *Receita com Variável Local*

O arquivo XML de receita especificado no parâmetro RecipeFile pode ser fixo no HTML ou composto por partes fixas e partes variáveis, determinadas em tempo de execução, através de variáveis locais. Exemplos:

- RecipeFile = "Receita1.xml" : nome de arquivo fixo, “Receita.xml”.
- RecipeFile = “Receita#numero\_maquina#.xml” : nome composto pelo string fixo “Receita” mais a variável local “numero\_maquina”. Quando a página está sendo carregada, o applet consulta o valor da variável local e concatena com a parte fixa, formando o nome do arquivo de receita. Por exemplo, se “numero\_maquina” for uma variável local com o valor 17, então o arquivo de receita será “Receita17.xml”

Observar que é possível informar um ou mais nomes de variáveis locais para formar o nome do arquivo de receita.

O exemplo2 descrito mais adiante neste capítulo mostra um exemplo de utilização de receita com variável local.

*Exemplo1: Uma Tela HTML e Um Arquivo de Receita com Nome Fixo*

Este exemplo mostra o arquivo TELA1.HTM com dois applets, um AnalogInput e um BarGraph. O primeiro monitora o operando M30 a cada segundo e o segundo o M70 do CP, a cada dois segundos. Os parâmetros Operand, Frequency, Description e BarForeColor destes dois applets não estão no HTML, mas sim no arquivo RECEITA1.XML. Observar que a diferenciação é feita pelo parâmetro “name” do applet.

```
<HTML>
<APPLET code= "AnalogInput.class" name=Applet1 archive="webview.jar" width=175 height=45>
  <param name = "RecipeFile" value = "Receita1.xml">
</APPLET>
<APPLET code= "BarGraph.class" name= Applet2 archive="webview.jar" width=250 height=130>
  <param name = "RecipeFile" value = "Receita1.xml">
</APPLET>
</HTML>
```

**Figura 4-30. Arquivo TELA1.HTM**

```
<?xml version="1.0" ?>
<RecipeFile>
  <Applet name="Applet1">
    <Operand>M30</Operand>
    <Frequency>1</Frequency>
    <Description>Teste1</Description>
  </Applet>
  <Applet name="Applet2">
    <Operand>M50</Operand>
    <Frequency>2</Frequency>
    <BarForeColor>Vermelho</BarForeColor>
  </Applet>
</RecipeFile>
```

**Figura 4-31. Arquivo RECEITA1.XML**

*Exemplo2: Uma Tela HTML e Vários Arquivos de Receita*

Este exemplo mostra como utilizar uma mesma tela HTML para apresentar valores de diferentes conjuntos de operandos do CP. Os mesmos applets apresentam operandos diferentes, conforme o arquivo de receita informado.

Por exemplo, o TELA1.HTM utilizado no exemplo anterior possuía um AnalogInput do M30 e um BarGraph do M70. Como fazer para que outros conjuntos de operandos possam ser apresentados, sem que seja necessário duplicar o arquivo TELA1.HTM ? Como apresentar o M30 e M50 em uma situação, com o BarGraph de cor vermelha, e o M131 e M151 em outra, com o BarGraph de cor azul, com esta mesma tela HTML ?

Para isto pode-se utilizar as variáveis locais. Cria-se uma variável local no arquivo HTML, que possa ser modificada pelo usuário, e utiliza-se esta variável para especificar o RecipeFile. A figura a seguir mostra as alterações feitas a partir do TELA1.HTM utilizado na seção anterior, que foram:

- criou-se a variável local “#NumTela#” através de um applet AnalogOutput (Applet 3 inserido)
- configurou-se o RecipeFile especificando um nome de arquivo XML que depende do valor da variável local “#NumTela#”

```

<HTML>
<applet code= "AnalogOutput.class" name=Applet3 archive="webview.jar" width=400
height=100>
  <param name = "Description" value = "Selecione o número da tela desejada: ">
  <param name = "Operand" value = " #NumTela# ">
</applet>
<applet code= "AnalogInput.class" name=Applet1 archive="webview.jar" width=175 height=45>
  <param name = "RecipeFile" value = "Receita#NumTela#.xml">
</applet>
<applet code= "BarGraph.class" name= Applet2 archive="webview.jar" width=250 height=130>
  <param name = "RecipeFile" value = "Receita#NumTela#.xml">
</APPLET>
</HTML>

```

**Figura 4-32. Arquivo TELA2.HTM**

Conforme o valor da variável local #NumTela#, um arquivo de receita, com parâmetros diferentes será selecionado, conforme mostra a figura a seguir. Se NumTela valer 1, o arquivo Receita1.xml, com os operandos M30, M50 e cor vermelha será selecionado. Se valer 2, é selecionado o arquivo Receita2.xml, com os operandos M131, M151 e cor azul.

**a) Arquivo Receita1.xml**

```

<?xml version="1.0" ?>
<RecipeFile>
  <Applet name="Applet1">
    <Operand>M30</Operand>
    <Frequency>1</Frequency>
    <Description>Teste1</Description>
  </Applet>
  <Applet name="Applet2">
    <Operand>M50</Operand>
    <Frequency>2</Frequency>
  <BarForeColor>Vermelho</BarForeColor>
  </Applet>
</RecipeFile>

```

**b) Arquivo Receita2.xml**

```

<?xml version="1.0" ?>
<RecipeFile>
  <Applet name="Applet1">
    <Operand>M131</Operand>
    <Frequency>1</Frequency>
    <Description>Teste1</Description>
  </Applet>
  <Applet name="Applet2">
    <Operand>M151</Operand>
    <Frequency>2</Frequency>
    <BarForeColor>Azul</BarForeColor>
  </Applet>
</RecipeFile>

```

**Figura 4-33. Arquivos Receita1.xml e Receita2.xml**

**ATENÇÃO:**

Neste mesmo exemplo, os parâmetros *Frequency* dos dois *applets* e *Description* do primeiro applet são iguais nas duas receitas, logo poderiam ser definidos diretamente no HTML reduzindo o tamanho total da página.

## 5. Instalação

Além dos recursos do WebView distribuído no CDROM do produto, algumas ferramentas externas, de mercado, são necessárias para sua utilização. Algumas configurações também precisam ser feitas no microcomputador e no WebGate. Este capítulo apresenta tudo que é necessário instalar e configurar para que o WebView seja utilizado.

Para a criação de telas HTML utilizando o WebView é necessário

- a biblioteca de applets WebView
- o plug-in Java JRE 1.4.2
- um programa de FTP
- um programa Editor HTML
- um browser compatível

Os dois primeiros itens desta lista são fornecidos no CDROM do produto. Os demais são ferramentas de mercado que devem ser adquiridas separadamente.

Para iniciar a utilização do WebView, recomenda-se a seguinte seqüência de instalação:

1. instalar o WebView e fornecer a chave de software
2. instalar o plug-in Java JRE 1.4.2
3. instalar o programa de FTP
4. instalar o programa Editor HTML
5. configurar o WebGate (enviar webview.jar)
6. visualizar a primeira tela de exemplo

### ATENÇÃO:

Este roteiro de instalação é válido para os usuários que desenvolvem páginas HTML usando os recursos do WebView. O usuário ou cliente final que apenas visualiza páginas pode instalar apenas o plug-in JRE 1.4.2, conforme seção **Instalando o Plug In Java**.

## Instalando o WebView

### O CDROM de Distribuição

O WebView é distribuído em um CDROM contendo os arquivos e diretórios listados a seguir .

Arquivos:

- AJUDA.TXT: Instruções rápidas p/ novos usuário
- HELP.TXT: Versão do arquivo Ajuda.txt em Inglês
- LEIAME.TXT : Informações recentes, não incluídas no manual do produto
- README.TXT : Versão do arquivo Leiname.txt em Inglês
- CONTRATO.TXT : Contrato de Licença de Software Altus
- CONTRACT.TXT : Versão do arquivo Contrato.txt em Inglês
- SETUP.EXE : Instalador, deve ser executado para iniciar a instalação
- AUTORUN.INF: habilita o auto-run da instalação quando insere CDROM

Diretórios:

[MANUAL] : Manual e CT do WebView

[PLUG-IN] : Plug In Java  
 [ACROBAT] : Acrobat Reader

### Requisitos de Hardware e Software

Para a instalação e utilização do WebView são recomendados os seguintes requisitos mínimos de hardware e software:

<b>Plataforma</b>	PC
<b>Sistema Operacional</b>	Windows 98 (2nd edition) Windows 2000 (service pack 3) Windows XP Professional (service pack 1)
<b>Processador</b>	Pentium 3 com 700 MHz (mínimo)
<b>Espaço em Disco</b>	500 MB livre (mínimo)
<b>Memória RAM</b>	256 MB (mínimo)
<b>Vídeo</b>	Super VGA
<b>Resolução</b>	800x600 (mínima)

### Iniciando a Instalação

Para instalar o WebView, deve-se executar o programa SETUP.EXE que é encontrado no diretório raiz do CDROM.

O programa de instalação solicita que o usuário preencha o contrato de licença de software: nome da empresa, número de série, chave de software **e também o endereço IP do WebGate no qual vai ser utilizado o WebView**, conforme mostra a Figura 5-1.

Para registrar o software instalado :

- 1) Informe o nome da empresa e o número de série do produto corretamente
- 2) Informe o endereço IP do PD7091 ou WebGate e a chave de software
- 3) Clique OK para concluir o registro do software, ou
- 4) Clique DEMO para utilizar em modo demonstração.

Nome da Empresa :

Número de Série :

Endereço IP:

Chave de Software :

Figura 5-1: Instalação do WebView

Após o preenchimento das informações de licença de software, a instalação prossegue com o plug-in JRE 1.4.2 .

Ao final da instalação, é apresentada uma tela confirmando ou não o sucesso da operação.

**ATENÇÃO:**

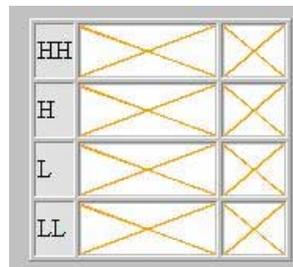
Para instalar ou desinstalar o WebView no Windows NT/2000/XP é necessário utilizar um login com privilégios de administrador.

### Modo Demonstração

Pode-se executar o WebView em modo demonstração. Neste modo de operação, não é necessário preencher os dados do contrato de licença de software, mas a funcionalidade do produto fica reduzida, pois apenas 5 applets poderão se comunicar com o CP.

**ATENÇÃO:**

Em modo demonstração, o WebView permite que apenas 5 applets se comuniquem normalmente com o CP. Neste caso, os demais applets serão apresentados como um retângulo branco com um “X” laranja por toda a sua extensão, conforme mostra a Figura 5-2.



**Figura 5-2: Applets não apresentados em Modo Demonstração**

Para se instalar a versão de demonstração do WebView, deve-se pressionar o botão “Demo” na tela de preenchimento de licença de software (ver Figura 5-1).

Para verificar se o WebView está executando em modo demonstração, deve-se utilizar a janela *Java Console*, aberta a partir do botão direito do mouse quando este está posicionado sobre o ícone do Java na barra de tarefas. Mais detalhes desta janela podem ser obtidos na seção **A Janela Java Console**, no capítulo **Manutenção**.

### O que é Instalado ?

Durante a instalação do WebView, os seguintes arquivos e diretórios são criados no microcomputador:

- Arquivo **webview.jar**
- Arquivo **webview.lic**
- Diretório [manual], com o manual do WebView
- Diretório [exemplos], com exemplos de utilização

Os arquivos **webview.jar** e **webview.lic** contêm todos os applets que formam o WebView, no formato compactado. Estes arquivos devem ser enviados para o WebGate, conforme descrito na seção **Carregando a Biblioteca de Applets no WebGate**.

O diretório [manual] contém o manual do WebView. O diretório [exemplos] possui exemplos de utilização dos applets.

## Desinstalação

Para se desinstalar o WebView, deve-se seguir os seguintes passos:

- Fechar todos os programas
- Clicar no botão Iniciar do Windows, apontar para Configurações e, em seguida, clicar em Painel de Controle
- Clicar duas vezes no ícone Adicionar/Remover programas
- Clicar em “WebView” na guia Instalar/desinstalar e, em seguida, em Adicionar/Remover
- Seguir as instruções na tela

### ATENÇÃO:

Para instalar ou desinstalar o WebView no Windows NT/2000/XP é necessário utilizar um login com direitos de administrador.

## Instalando o Acrobat Reader

Os arquivos de CTs e manuais neste CDROM encontram-se no formato PDF e necessitam do visualizador "Acrobat Reader" ou compatível para serem visualizados.

O visualizador "Acrobat Reader" pode ser instalado executando-se o arquivo que encontra-se no diretório "[acrobat]" na raiz do CDROM.

## Instalando o Plug-In Java

É necessário a instalação do plug-in "Java 2 Run Time Environment SE" versão 1.4.2 , distribuído no CDROM deste produto, no diretório “plug-in”.

Para isto, a versão do Internet Explorer deve ser igual a 5.5 ou 6.0. Para verificar a versão, deve-se executar a opção de menu “Ajuda / Sobre o Internet Explorer” , conforme mostra a figura a seguir.



Versão do Internet Explorer deve ser 5.5 ou 6.0

Figura 5-3: Versão do Internet Explorer

Para a instalação do plug-in , deve-se executar o arquivo existente no diretório “[plug-in]” do CDROM deste produto. A tela inicial é mostrada na figura a seguir.

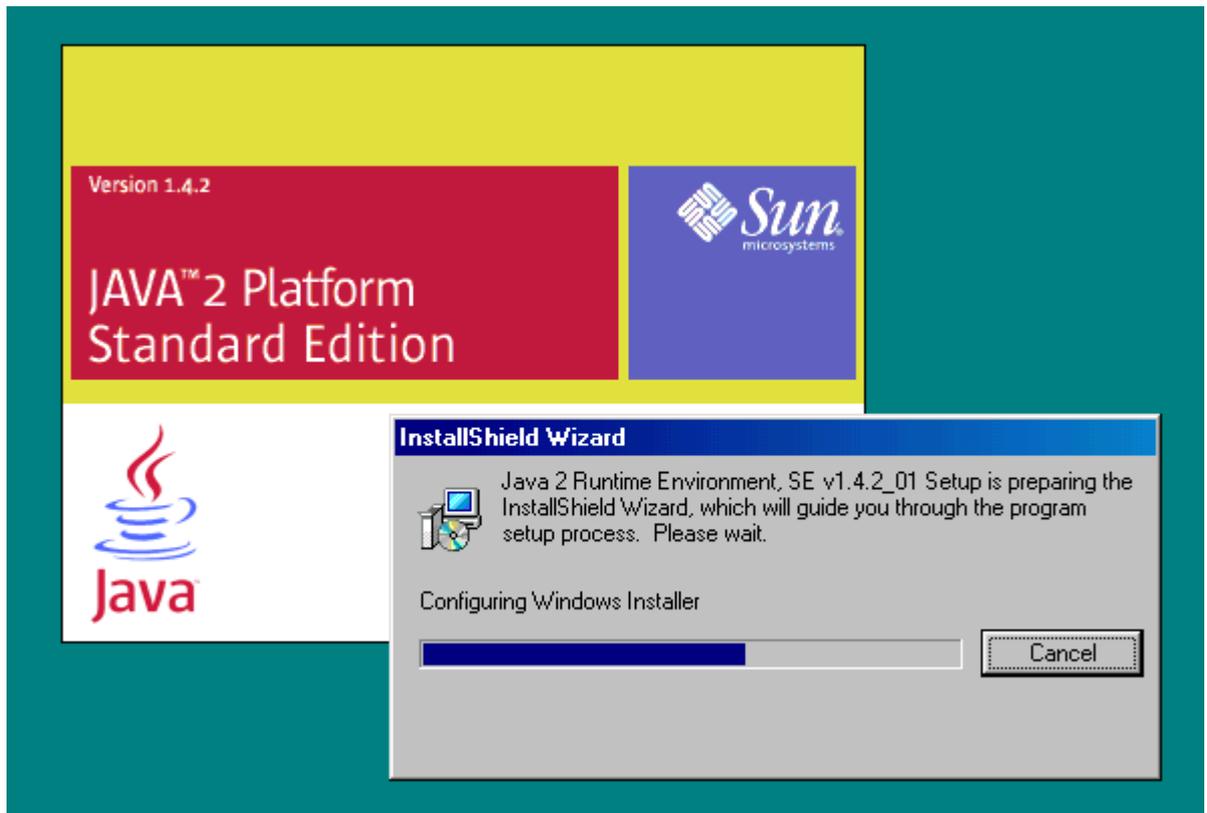


Figura 5-4: Instalação do Plug-in Java

Após esta tela inicial, a instalação prossegue com a seguinte seqüência de telas:

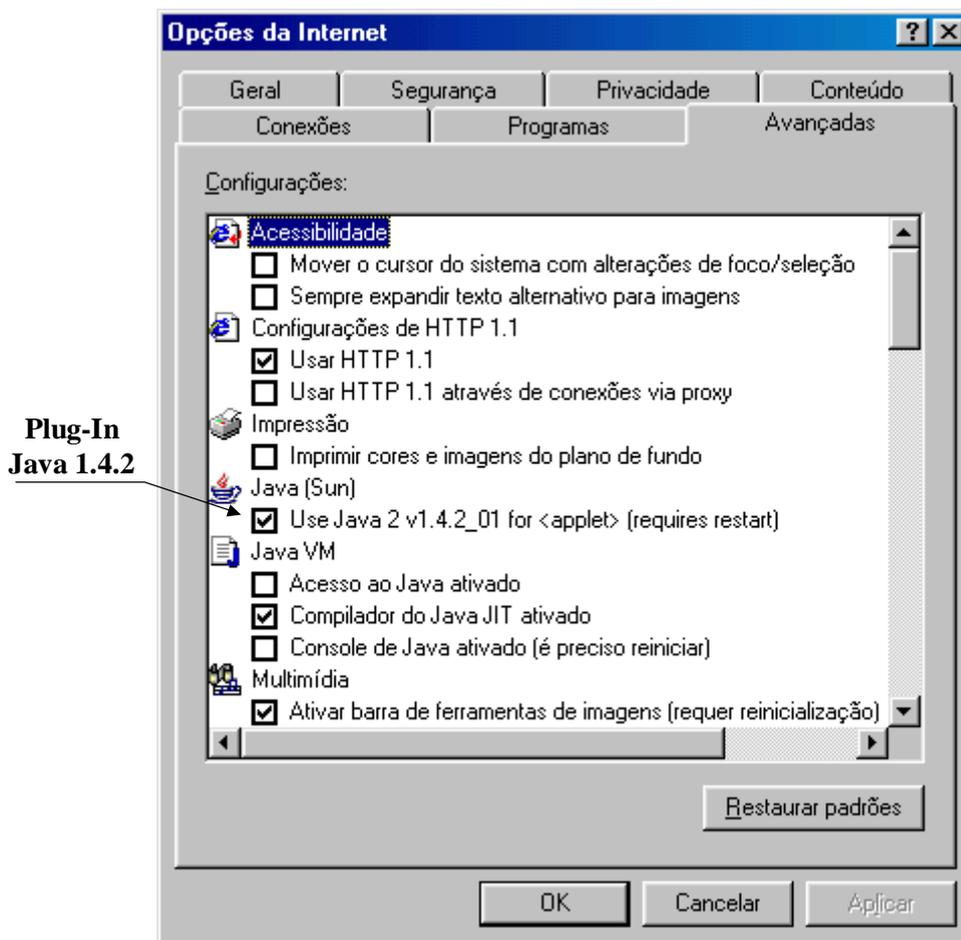
- *Welcome*: apresenta a janela de boas vindas;
- *Licence Agreement*: apresenta o Contrato de Licença de Software da Sun Microsystems. Estando de acordo com os termos do contrato, deve-se aceitá-lo para que a instalação prossiga;
- *Setup Type*: deve-se manter a opção “Typical” ;

**ATENÇÃO:**

Para instalar ou desinstalar o plug-in no Windows NT/2000/XP é necessário utilizar um login com direitos de administrador. Após a instalação, deve-se reiniciar o microcomputador para que as alterações tenham efeito.

### Verificando o Plug-In Java

Após a instalação do plug-in, deve-se verificar se o Internet Explorer está corretamente configurado para utilizá-lo, através do menu "Ferramentas / Opções da Internet", guia "Avançadas". A opção "Use Java 2 v1.4.2\_01 for <applet>" deve estar habilitada, conforme mostra a figura a seguir:



**Figura 5-5: Plug In Java 2 V1.4.2**

Caso a opção não esteja assinalada, é preciso habilitá-la.

## Instalando o Programa de FTP

O FTP (File Transfer Protocol) é um protocolo para transferência de arquivos muito utilizado na Internet. Assim como o http é um protocolo utilizado para acessar sites de páginas da Web, o FTP é utilizado para fazer a transferência de arquivos de uma máquina para outra na rede.

O FTP permite a conexão do microcomputador com o WebGate para fazer upload e download de arquivos entre eles.

Qualquer programa de FTP compatível com a RFC 793 (Transmission Control Protocol) pode ser utilizado.

## Instalando o Editor HTML

A utilização de um editor HTML é necessária para criar ou realizar modificações desejáveis nas páginas HTML com os applets WebView.

HTML é uma linguagem que serve para confeccionar documentos de hipertexto, sendo uma forma de criar documentos estruturados com links. Diversas páginas existentes nos vários sites da internet são construídas com HTML. No caso do WebView, é possível criar páginas HTML que se comunicam com os controladores programáveis Altus, inserindo nestas páginas os applets WebView.

Existem diversos editores de HTML no mercado, cada um com características próprias, mas o objetivo de todos é permitir a criação e modificação de páginas em HTML. Recomenda-se a utilização de um dos seguintes editores HTML:

- Microsoft FrontPage Express
- Macromedia DreamWeaver

## Configurando o WebGate

A principal configuração a ser feita no WebGate é o seu endereço IP. Para isto, consultar o Manual de Utilização do WebGate correspondente.

Para a utilização do WebView, o WebGate e o CP devem estar ligados, conectados e configurados. No caso dos PO9900 e PO9901, devem estar conectados ao CP via rede ALNET I, disponível em qualquer controlador programável Altus. No caso do PO7091, deve estar conectado em um barramento Ponto da CPU PO3342. Devem também estar conectados à rede local Ethernet, através da Interface Ethernet 10BaseT. Recomenda-se realizar as seguintes verificações:

- o CP deve estar energizado e conectado ao WebGate com o cabo especificado no Manual do WebGate
- o WebGate deve estar energizado e conectado a rede ethernet, com um endereço IP válido
- o LED de atividade do WebGate deve estar piscando
- executar o comando “ping” para verificar se o WebGate está corretamente conectado a rede.

Para informações sobre instalação e configuração do WebGate e CP e conexão entre eles, deve-se consultar o manual dos respectivos produtos.

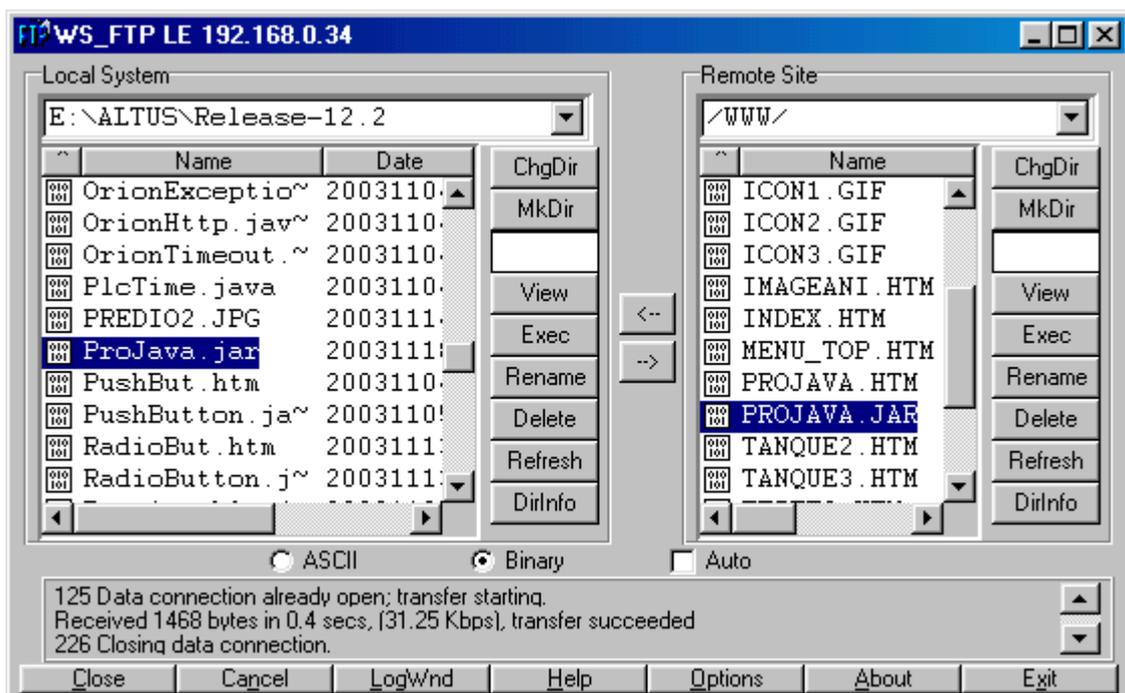
### ATENÇÃO

Para a utilização do WebView, recomenda-se que o WebGate não esteja configurado no modo mestre, apenas no modo escravo, por questões de desempenho nas comunicações.

Para maiores detalhes deve-se consultar o Manual de Utilização do WebGate correspondente.

## Carregando a Biblioteca de Applets no WebGate

Os arquivos **webview.jar** e **webview.lic** contêm a biblioteca de applets que formam o WebView, no formato compactado. Estes dois arquivos devem ser carregados no WebGate, no diretório “[WWW]”, através do programa de FTP.



**Figura 5-6: Carregando o webview.jar**

A Figura 5-6 mostra o arquivo webview.jar selecionado para ser enviado para o WebGate, no programa WS-FTP LE. A lista de arquivos da esquerda correspondem aos arquivos locais no microcomputador, enquanto que a lista da direita corresponde aos arquivos remotos armazenados no WebGate.

Pode-se obter mais detalhes da utilização do programa de FTP da Figura 5-6 na seção **Enviando uma Página HTML para o WebGate**, no capítulo **Programação**.

**ATENÇÃO**

Deve-se utilizar o modo "Binary" para fazer as transmissões para o WebGate

## 6. Manutenção

Este capítulo fornece informações sobre os problemas mais comuns que podem ser encontrados durante a configuração e utilização do WebView.

Checklist de Problemas de Comunicação:

- verificar se o comando ping para endereço IP do WebGate está funcionando
- verificar se o led NET do WebGate (atividade na rede) está piscando
- verificar se o WebGate está corretamente conectado ao CP
- verificar se o CP está em modo execução
- verificar se o MasterTool consegue acessar normalmente o CP através do WebGate
- não usar o WebGate em modo mestre

Checklist de Problemas de Execução dos applets

- verificar se o plug-in Java JRE 1.4.2 foi corretamente instalado
- verificar se o WebView não se encontra em modo demonstração
- verificar o tamanho do arquivo enviado para o WebGate após o FTP
- verificar se o FTP está sendo feito no modo “Binary”
- não enviar arquivos para o diretório raiz do webGate, sempre para o diretório WWW ou sub-diretório deste
- verificar se uma página não possui muitos applets monitorando o CP sem que o applet Driver tenha sido utilizado
- verificar se o programa aplicativo do CP não está continuamente escrevendo o valor de um operando que deseja-se alterar através de um applet
- verificar se todos os parâmetros obrigatórios estão configurados
- verificar se o texto “webview.jar” foi digitado exatamente desta forma (é case sensitive)
- verificar se os nomes das classes foram digitados de forma correta (é case sensitive)
- não inserir por engano um espaço em branco no final do nome do parâmetro
- não inserir duas vezes o mesmo parâmetro (o segundo é desconsiderado)
- utilizar o console Java para obter detalhes de erros de execução
- utilizar o parâmetro “ConsoleEnable” para obter detalhes de erros de execução

### Parâmetros com Valores Inválidos

Alguns dos parâmetros dos applets são consistidos em tempo de execução para verificar se possuem valores válidos. Caso o parâmetro tenha um valor inválido, uma mensagem de erro é mostrada, tal como mostra a Figura 6-1. Os parâmetros consistidos e os valores válidos são mostrados na tabela a seguir:

Parâmetro	Valor Válido
Operand OperTime OperAck	precisa ser um operando válido do CP
Background	precisa ser um nome de cor ou um número hexadecimal ou um nome de arquivo válido
FontColor BarForeColor BarBackColor GraphColor AxisColor	precisa ser um nome de cor ou um número hexadecimal válido

Frequency	
FontSize	
DecimalDigits	
MinLimit	
MaxLimite	
NumInterval	
Value	
NumElem	precisa ser um número inteiro válido
Div	
DivX	
DivY	
QtdValores	
NumEvents	
NumAlarms	
QtyPolling	

**Tabela 6-1. Valores válidos dos Parâmetros**

A Figura 6-1 mostra um exemplo de mensagem de erro para o parâmetro “frequency” do applet “AnalogInput”. Este parâmetro deveria ser preenchido com um valor inteiro, mas em vez disto foi preenchido com o string “acc”, conforme mostra a figura.



**Figura 6-1. Mensagem de Erro para Parâmetro Inválido**

**ATENÇÃO:**

O nome do applet que gerou a mensagem de erro também é mostrado no título da janela de erro.

## Falta de Parâmetros Obrigatórios

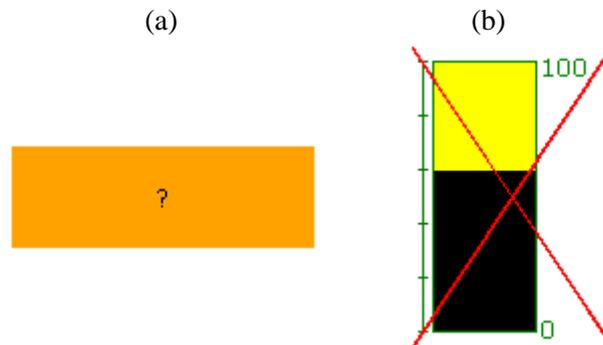
Quando um parâmetro obrigatório de qualquer applet não é informado, ele não executa corretamente. A mensagem da Figura 6-2 mostra um exemplo de tela de erro em um applet AnalogInput onde o parâmetro *Operand* não foi informado.



**Figura 6-2. Falta do Parâmetro *Operand***

## Problemas na Comunicação com o CP

Em caso de uma falha de comunicação entre o WebGate e o CP, pode acontecer que as páginas HTML sejam carregadas normalmente no browser, mas não os valores apresentados pelos applets WebView. Isto acontece porque as páginas e os applets ficam armazenados no WebGate, enquanto que os valores dos operandos são lidos do CP pelo WebGate. Nestes casos, os applets podem apresentar o carácter “?” no lugar do valor ou um “x” vermelho sobre ele mesmo, conforme mostrado nas Figura 6-3a e Figura 6-3b, respectivamente.



**Figura 6-3. Problemas na Comunicação com o CP**

Um problema semelhante a este pode acontecer quando um applet solicita a leitura de um operando que não esteja configurado no CP. Por exemplo, um applet AnalogInput solicita a leitura do operando M2500, mas o CP correspondente está configurado para uma quantidade menor de operandos memória. O efeito visual na apresentação dos applets em caso de erros deste tipo é o mesmo da Figura 6-3.

## Problemas na Comunicação com o WebGate

Em caso de uma falha de comunicação entre o microcomputador e o WebGate, nenhuma página HTML poderá ser apresentada. É o mesmo efeito que acontece quando se tenta acessar um determinado endereço IP na internet e o servidor correspondente está fora do ar ou não existe. Neste caso, deve-se verificar as instalações do WebGate. No mínimo o comando ping deve estar funcionando para que as páginas possam ser visualizadas. Maiores detalhes podem ser obtidos no Manual de Utilização do WebGate.

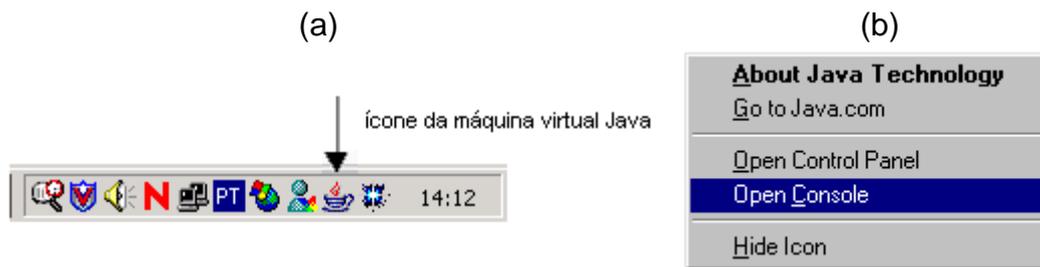
## Congelamento do Browser

Quando o browser não responder mais aos comandos do mouse ou teclado, numa primeira análise pode parecer que ele está congelado, mas pode ser que não. Pode acontecer que uma janela de erro disparada por um applet esteja com o foco da aplicação, esperando por em ENTER do usuário. É comum que tais janelas fiquem escondidas por trás das janelas do mouse, ficando difícil visualizá-las.

Para contornar este problema, pode-se listar as tarefas ativas do Windows, através das teclas ALT+TAB. Caso exista uma janela de erro fazendo com que o browser fique congelado, esta janela aparecerá nesta lista com o ícone do Java (a xícara de café). Deve-se selecionar esta tarefa e pressionar ENTER na mensagem correspondente.

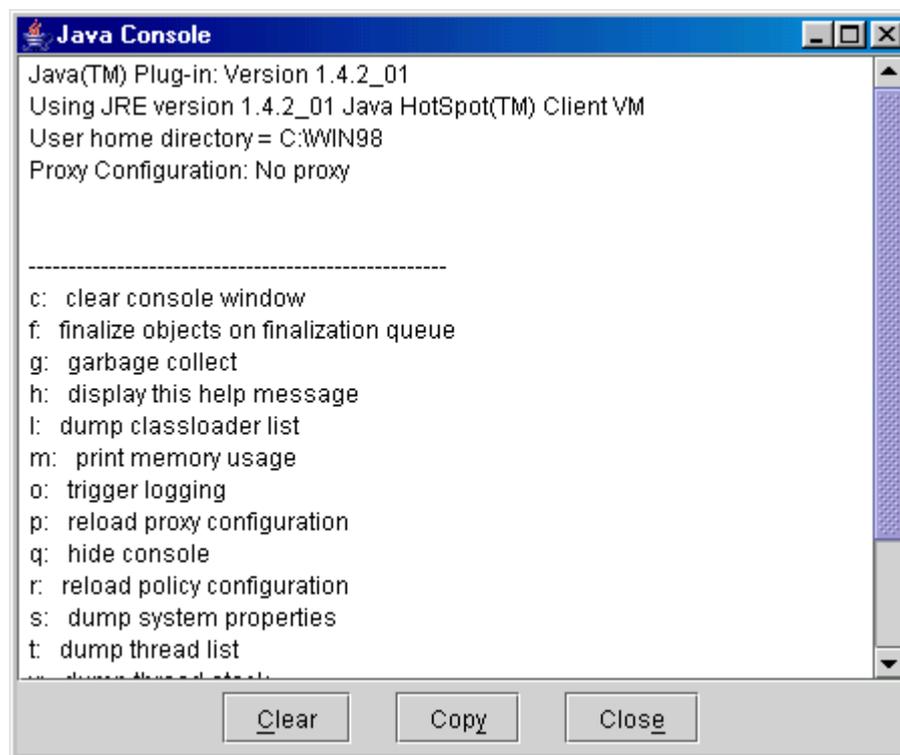
## A Janela Java Console

Maiores informações sobre erros em tempo de execução nas classes Java dos applets WebView podem ser visualizadas através da janela “Java Console”. Esta janela pode ser aberta a partir do ícone de execução da máquina virtual Java. Este ícone é apresentado na barra de tarefas quando uma página com um applet Java é visualizada no browser, conforme mostra a Figura 6-4a.



**Figura 6-4. Abrindo a Janela “Java Console”**

Para abrir a janela “Java Console”, deve-se clicar com o botão da direita do mouse sobre o ícone da máquina virtual Java e selecionar a opção “Open Console”, conforme mostra a Figura 6-4b. A janela é mostrada na Figura 6-5.



**Figura 6-5. Janela “Java Console”**

É possível configurar os applets WebView para que eles apresentem informações de depuração na janela “Java Console”. Para isto, deve-se configurar o parâmetro “**ConsoleEnable**” com o valor “true” em cada applet que se deseja depurar.

**ATENÇÃO:**

Recomenda-se que o parâmetro “ConsoleEnable” não seja mantido habilitado na versão final da página por questões de performance.

## Erros de Execução nos Applets

Quando a execução de um applet não pode ser efetuada ou é interrompida por qualquer razão, este applet não chega a ter sua apresentação concluída na tela. Nestes casos, o browser mostra um retângulo cinza com as dimensões configuradas para o applet e um “x” vermelho no canto superior esquerdo, conforme mostra a figura a seguir.



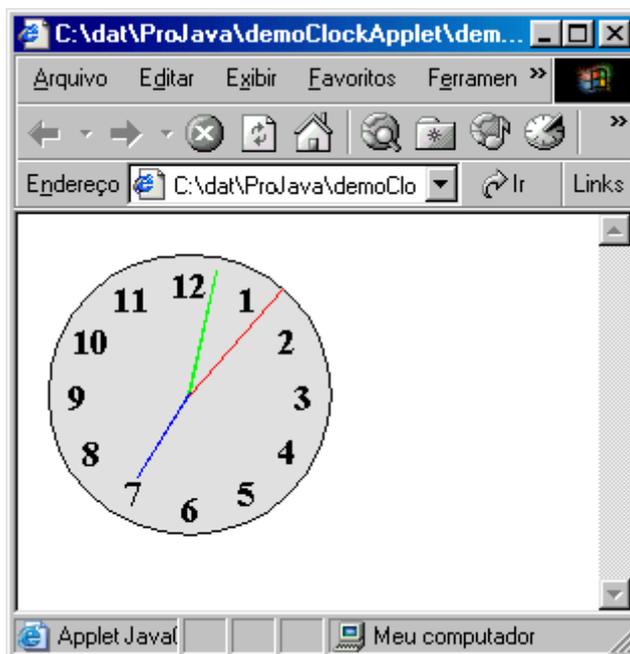
**Figura 6-6. Erro de Execução no Applet**

Nestas situações, muitas vezes aparece a mensagem “Applet xxx not inited” na linha de status do Internet Explorer, onde “xxx” é o nome do applet que gerou o erro. Deve-se posicionar o cursor sobre o applet para observar esta mensagem. Também nestes casos é possível abrir a janela “Console Java”, descrita anteriormente, para obter mais detalhes sobre o erro ocorrido.

## Problemas na Máquina Virtual Java

Para que o browser possa mostrar corretamente as páginas WebView, o plug-in Java deve ter sido corretamente instalado, conforme descrito anteriormente neste manual. Caso por algum problema o browser não consiga executar applets, então ele também não conseguirá apresentar nenhum applet WebView.

Pode-se verificar se o browser está conseguindo executar applets visualizando um exemplo que não faz parte do WebView. Este exemplo apresenta um relógio conforme mostrado na Figura 6-7, e pode ser visualizado a partir do arquivo “Clock.HTM” disponível no diretório “Manutencao/Clock” do CDROM deste produto.



**Figura 6-7. Verificando a Execução de um Applet Qualquer**

## 7. Anexo I: Exemplos de Aplicação

Este capítulo apresenta alguns exemplos de aplicação do WebView. As páginas mostradas aqui estão disponíveis no CDROM do produto.

### Monitoração de Operandos do CP

Este exemplo apresenta uma página HTML bastante simples, na qual o valor do operando M15 do CP é apresentado na tela, com seu valor atualizado a cada dois segundos. Pode-se alterar o valor deste operando clicando-se em um dos botões de radio. Após esta operação o novo valor é atualizado no CP e o valor apresentado é atualizado após alguns instantes (menos de dois segundos).

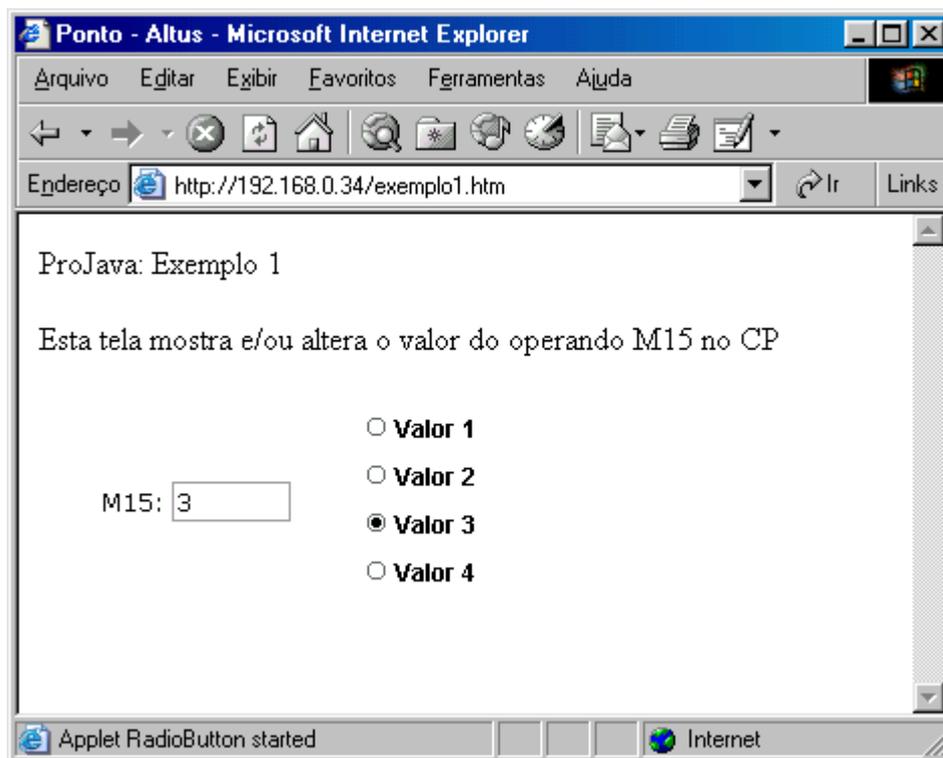


Figura 7-1. Monitoração de Operandos do CP

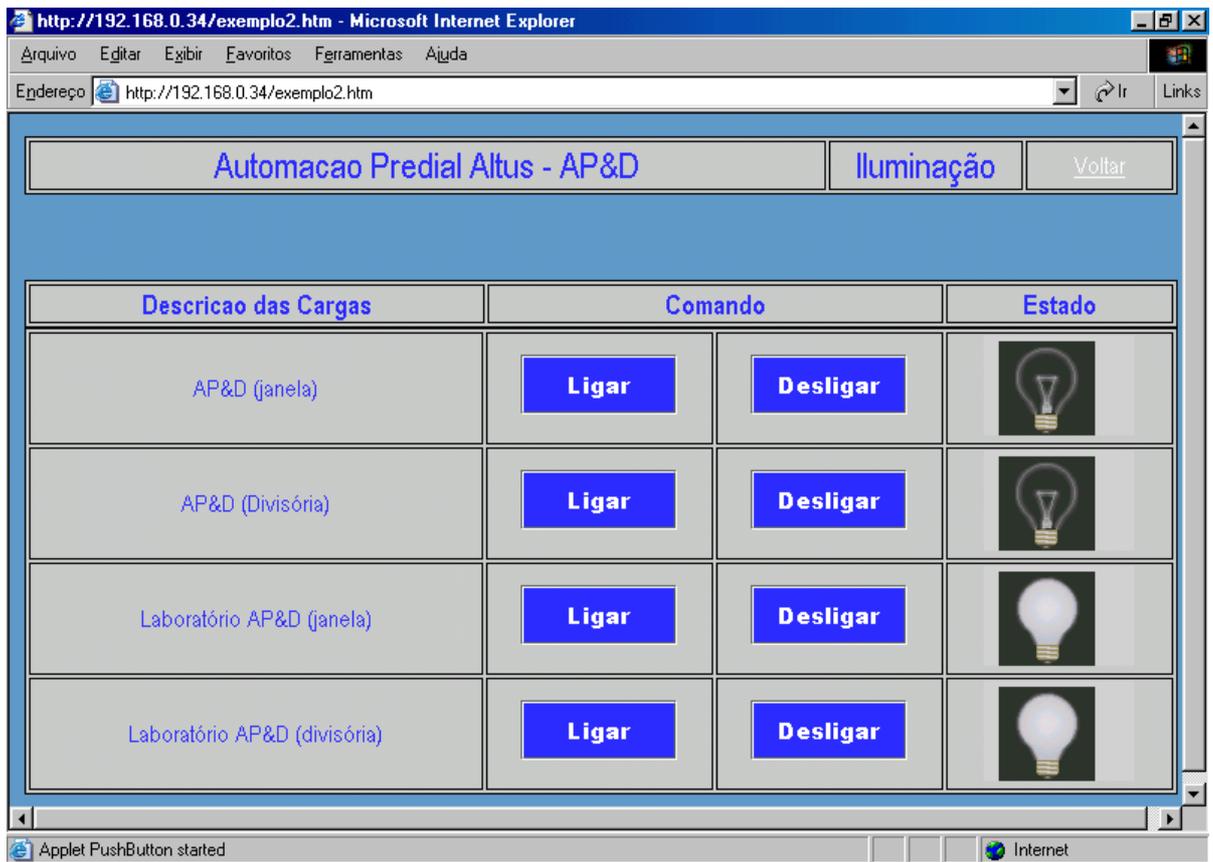
A Figura 7-1 mostra o exemplo 1 sendo apresentado no browser Internet Explorer. Esta página foi construída utilizando-se dois applets WebView, AnalogInput e RadioButton.

Este arquivo HTML pode ser encontrado no diretório "Exemplos\Exemplo1" do CDROM deste produto.

### Controle de Iluminação

Este exemplo apresenta uma página HTML de controle de iluminação de um sistema de automação predial.

Nesta tela pode-se ligar ou desligar a iluminação de alguns setores da empresa através dos applets PushButton correspondentes, e o estado ligado/desligado é apresentado através dos applets ImageAnimation com as imagens de lâmpada acesa e lâmpada apagada, conforme mostra a Figura 7-2.



**Figura 7-2. Controle de Iluminação**

Os arquivos que compõe este exemplo podem ser encontrado no diretório “Exemplos\Exemplo2” do CDROM deste produto.

## Exaustão de Gases

Este exemplo apresenta uma página HTML do sistema de exaustão de gases de uma aciaria, conforme mostra a Figura 7-3. Este é um exemplo de tela de controle de processo industrial onde vários applets WebView são utilizados e a comunicação com o CP feita através do applet Driver.

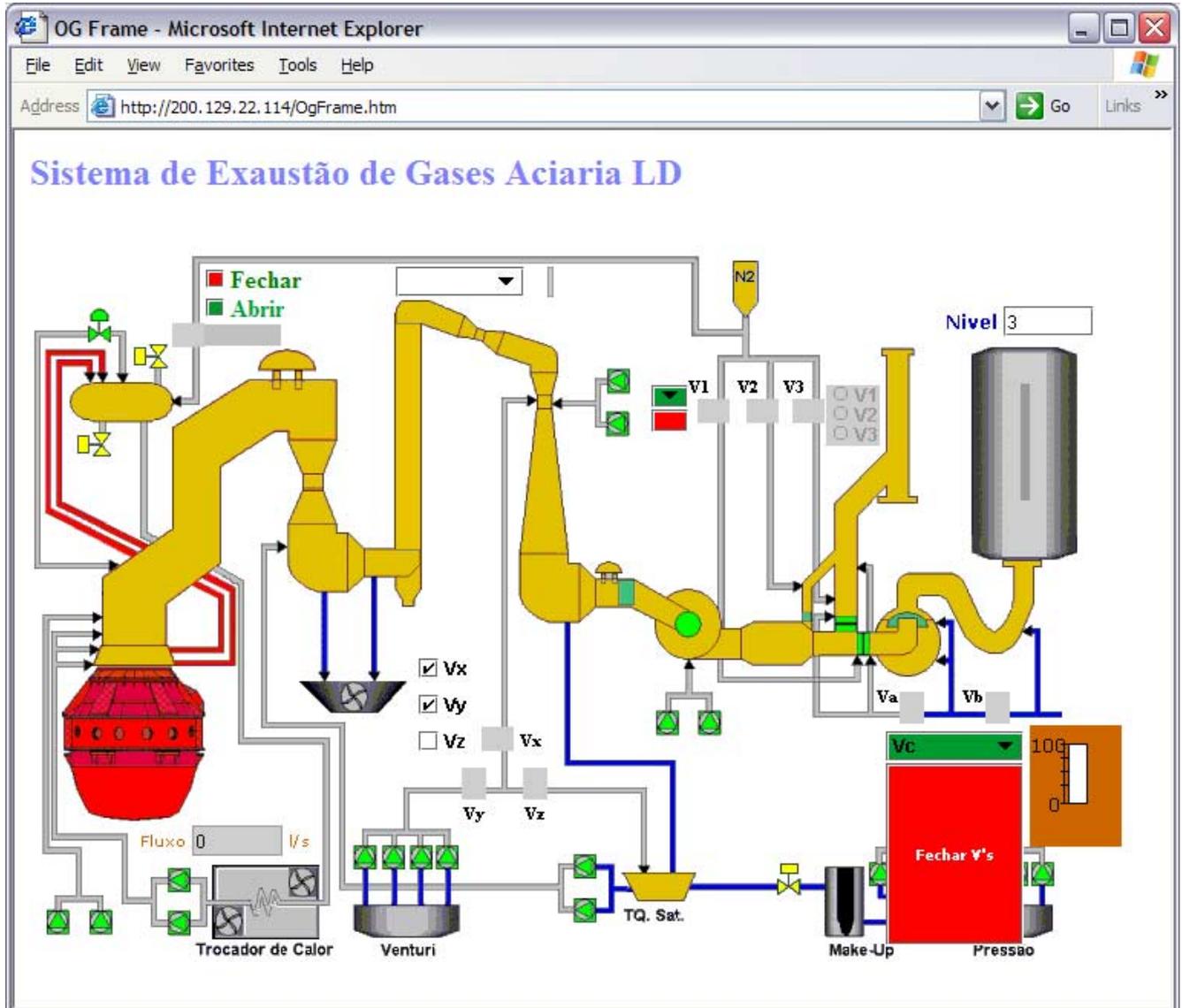


Figura 7-3. Sistema de Exaustão de Gases de Aciaria LD

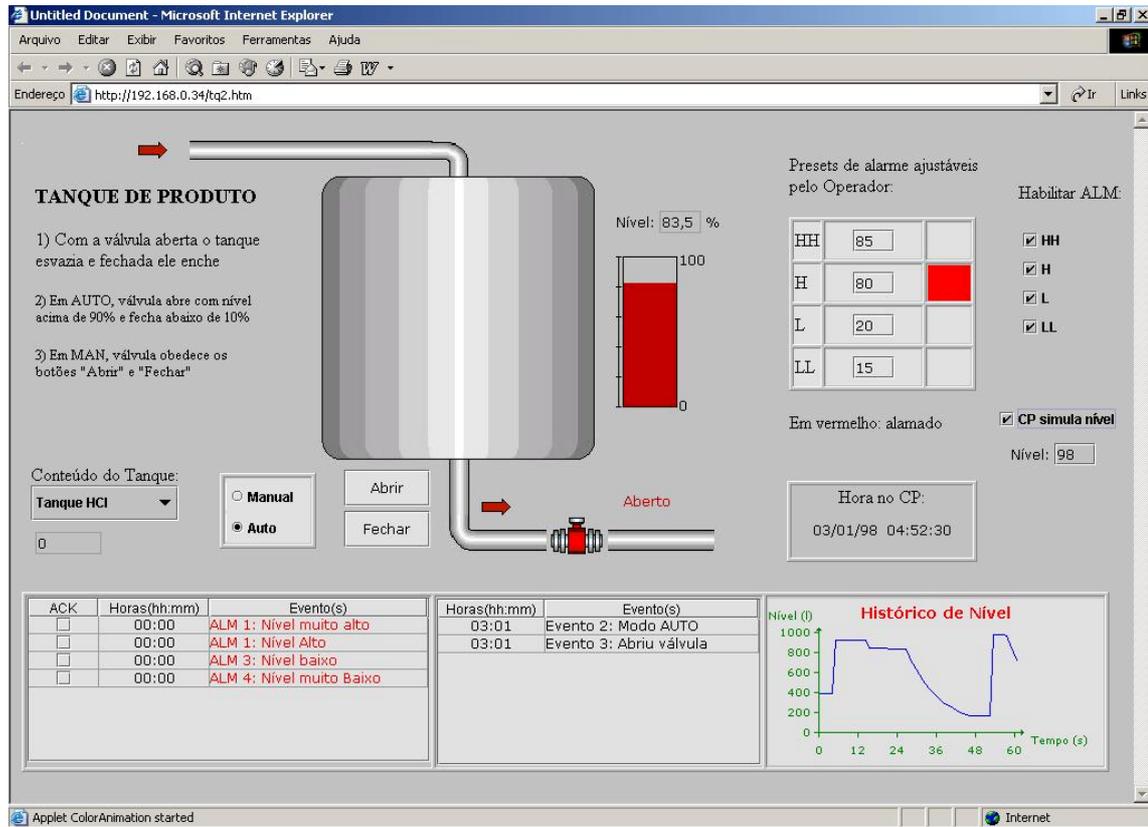
Os arquivos que compõe este exemplo podem ser encontrado no diretório “Exemplos\Exemplo3” do CDRom deste produto.

## Controle de Nível de Tanque

Este exemplo uma simulação de controle de nível de um tanque com a entrada contínua de um determinado produto líquido, uma medição de nível e uma válvula de saída no fundo do tanque.

O programa aplicativo de simulação possui dois modos de operação para o tanque e a válvula: em modo automático, a válvula abre com nível acima de 90% e fecha com nível abaixo de 10%. Quando a válvula está aberta, o tanque está esvaziando e quando fechada o tanque está enchendo.

O exemplo permite a configuração e apresentação de 4 limites de alarmes: HH, H, L e LL. O valor de cada um destes presets pode ser habilitado e informado individualmente. O exemplo mostra ainda uma lista de eventos que registra que os comandos *Abrir*, *Fechar*, *Auto* e *Man* e um gráfico com o histórico do nível do tanque nos últimos 60 segundos.



**Figura 7-4. Controle de Nível de Tanque**

Esta tela foi construída utilizando-se applets WebView que permitem selecionar o modo manual ou automático de operação, abrir ou fechar a válvula, visualizar o estado aberto ou fechado da válvula através de uma imagem GIF e de um texto associado, visualizar o nível do tanque através de um valor analógico e de um bargraph, visualizar o histórico de nível do tanque nos últimos 60 segundos, habilitar e configurar individualmente os presets HH, H, L e LL de alarme de nível e visualizar o horário atual no CP.

A tela possui ainda uma lista de alarmes, mostrando em vermelho os alarmes ativos e em verde os já reconhecidos, e também uma lista de eventos que registra os comandos *Abrir*, *Fechar*, *Auto* e *Man*.

Esta tela foi construída utilizando-se um editor de texto convencional de mercado. Foi selecionado o fundo cinza e inserido o bitmap com a imagem do tanque. Em seguida, os textos e applets WebView foram inseridos e posicionados na tela utilizando-se o recurso de *layers*. Os parâmetros de cada applet foram configurados conforme o Manual de Utilização do WebView. Os operandos do CP utilizados em cada um foram selecionados conforme o programa aplicativo de simulação existente no CP. O arquivo HTM gerado, as figuras e o bitmap utilizado precisam ser carregados no PO7091 ou WebGate para serem visualizados no browser.

Todos os arquivos que formam esta tela e o programa aplicativo do CP que compõe este exemplo podem ser encontrado no diretório "Exemplos\Exemplo4" do CDROM deste produto.

## Automação Predial

Este exemplo mostra um Sistema de Automação Predial Completo, com uma tela de abertura contendo recursos shockwave flash e que faz a carga inicial do “webview.jar” da maneira recomendada, podendo ser utilizado como referência para novas aplicações.

O exemplo possui um menu para as demais telas do sistema, que por sua vez utilizam a maioria dos Applets do WebView. Todas as telas utilizam o Applet Driver para concentrar as comunicações, e a tela do sistema de Ar Condicionado utiliza o recurso de receita.

A Figura 7-5 mostra a Tela de Ar Condicionado deste sistema, na qual pode ser observada o mesmo menu presente na tela de abertura, Applets AnalogInput mostrando a temperatura dos apartamentos e links para telas com gráficos baseados no Applet PenChart para mostrar um histórico destas temperaturas.

The screenshot shows a web browser window titled "Untitled Document - Microsoft Internet Explorer" with the address bar showing "http://192.168.0.34/". The main content area is titled "Sistema de Ar Condicionado Central" and contains a table with the following data:

Componente	Temperatura Ar Insuflado	Temperatura Ambiente	Ação
Chiller	0,0 C	0,0 C	<a href="#">ExibirGráfico</a>
Apartamento 303	0,0 C	0,6 C	<a href="#">ExibirGráfico</a>
Apartamento 302	9,8 C	0,1 C	<a href="#">ExibirGráfico</a>
Apartamento 301	0,9 C	0,6 C	<a href="#">ExibirGráfico</a>
Apartamento 203	4,3 C	2,0 C	<a href="#">ExibirGráfico</a>
Apartamento 202	810,6 C	0,0 C	<a href="#">ExibirGráfico</a>
Apartamento 201	0,0 C	0,0 C	<a href="#">ExibirGráfico</a>
Apartamento 103	0,1 C	0,8 C	<a href="#">ExibirGráfico</a>
Apartamento 102	1,0 C	777,4 C	<a href="#">ExibirGráfico</a>
Apartamento 101	3,6 C	7,4 C	<a href="#">ExibirGráfico</a>

At the bottom of the interface, there are several status fields: "ACK", "Horas(hh:mm)", "Alarme(s)", and "CLP Data/Hora" with the value "00/00/00 00:00:00". The browser's taskbar at the bottom shows the system tray with the time "16:09".

Figura 7-5. Sistema de Automação Predial

Todos os arquivos que compõe este exemplo podem ser encontrado no diretório “Exemplos\Exemplo5” do CDROM deste produto.

## 8. Anexo II: Tabela de Cores

A tabela abaixo relaciona os 16 nomes padrão, suportados por todos os browsers que exibem cores, e seus respectivos códigos RGB em hexadecimal.

NOME	CÓDIGO	NOME	CÓDIGO
 red	FF0000	 maroon	800000
 lime	00FF00	 green	008000
 blue	0000FF	 navy	000080
 yellow	FFFF00	 olive	808000
 aqua	00FFFF	 teal	008080
 fuchsia	FF00FF	 purple	800080
 white	FFFFFF	 silver	C0C0C0
 black	000000	 gray	808080

Tabela 8-1. Nomes de Cores Padrão

O WebView também aceita os seguintes nomes de cor em português:

	<b>Preto</b> 000000	<b>Grana</b> 00A040	<b>Laranja</b> FFA000	<b>Caramelo</b> C06000	<b>Bege</b> E0C0A0
<b>Azul</b> 0000FF	<b>Anil</b> 0000C0	<b>Marinho</b> 000080	<b>Lilas</b> C000FF	<b>Uva</b> 8080FF	<b>Ameixa</b> C0C0FF
<b>Roxo</b> 8000FF	<b>Calypso</b> 0080FF	<b>Celeste</b> 00C0FF	<b>Ciano</b> 00FFFF	<b>Petroleo</b> 00C0C0	<b>Prussia</b> 008080
<b>Lima</b> 00FF00	<b>Abacate</b> 00C000	<b>Verde</b> 008000	<b>Esmeralda</b> 00FFC0	<b>Limao</b> 80FF80	<b>Bambu</b> C0FFC0
<b>Primavera</b> 40FF80	<b>Palha</b> FFFA00	<b>Carambola</b> C0FF00	<b>Amarelo</b> FFFF00	<b>Palmeira</b> C0C000	<b>Oliva</b> 808000
<b>Vermelho</b> FF0000	<b>Terracota</b> C00000	<b>Marrom</b> 800000	<b>Rosa</b> FF00C0	<b>Fuchsia</b> FF00FF	<b>Vinho</b> C000C0
<b>Carmim</b> FF0080	<b>Telha</b> FF8000	<b>Ocre</b> FFC000	<b>Zarcao</b> FF8080	<b>Flamingo</b> FFC0C0	<b>Purpura</b> 800080
<b>Branco</b> FFFFFF	<b>Gelo</b> E0E0E0	<b>Prata</b> C0C0C0	<b>Cinza</b> A0A0A0	<b>Grafite</b> 808080	<b>Chumbo</b> 606060

Tabela 8-2. Nomes de Cor em Português

## 9. Glossário

<b>Bit</b>	Unidade básica de informação, podendo estar no estado 0 ou 1.
<b>Byte</b>	Unidade de informação composta por oito bits.
<b>Canal serial</b>	Interface de um equipamento que transfere dados no modo serial.
<b>Ciclo de varredura</b>	Uma execução completa do programa aplicativo de um controlador programável.
<b>Circuito de cão de guarda</b>	Circuito eletrônico destinado a verificar a integridade do funcionamento de um equipamento.
<b>Código comercial</b>	Código do produto, formado pelas letras PO, seguidas por quatro números.
<b>Controlador programável</b>	Também chamado de CP. Equipamento que realiza controle sob o comando de um programa aplicativo. É composto de uma UCP, uma fonte de alimentação e uma estrutura de E/S.
<b>CP</b>	Veja controlador programável.
<b>Database</b>	Banco de dados.
<b>Default</b>	Valor predefinido para uma variável, utilizado em caso de não haver definição.
<b>Diagnóstico</b>	Procedimento utilizado para detectar e isolar falhas. É também o conjunto de dados usados para tal determinação, que serve para a análise e correção de problemas.
<b>Download</b>	Carga de programa ou configuração no CP.
<b>E/S</b>	Veja entrada/saída.
<b>Entrada/saída</b>	Também chamado de E/S. Dispositivos de E/S de dados de um sistema. No caso de CPs, correspondem tipicamente a módulos digitais ou analógicos de entrada ou saída que monitoram ou acionam o dispositivo controlado.
<b>Estação de supervisão</b>	Equipamento ligado a uma rede de CPs ou instrumentação com a finalidade de monitorar ou controlar variáveis de um processo.
<b>Hardware</b>	Equipamentos físicos usados em processamento de dados onde normalmente são executados programas (software).
<b>Kbytes</b>	Unidade representativa de quantidade de memória. Representa 1024 bytes.
<b>Linguagem de programação</b>	Um conjunto de regras e convenções utilizado para a elaboração de um programa.
<b>Linguagem de relés e blocos Altus</b>	Conjunto de instruções e operandos que permitem a edição de um programa aplicativo para ser utilizado em um CP.
<b>Lógica</b>	Matriz gráfica onde são inseridas as instruções de linguagem de um diagrama de relés que compõe um programa aplicativo. Um conjunto de lógicas ordenadas seqüencialmente constitui um módulo de programa.
<b>MasterTool</b>	Identifica o programa Altus para microcomputador, executável em ambiente WINDOWS®, que permite o desenvolvimento de aplicativos para os CPs das séries Ponto, Piccolo, AL-2000, AL-3000 e Quark. Ao longo do manual, este programa é referido pela própria sigla ou como programador MasterTool.
<b>Menu</b>	Conjunto de opções disponíveis e exibidas por um programa no vídeo e que podem ser selecionadas pelo usuário a fim de ativar ou executar uma determinada tarefa.
<b>Mestre</b>	Equipamento ligado a uma rede de comunicação de onde se originam solicitações de comandos para outros equipamentos da rede.
<b>Módulo (referindo-se a hardware)</b>	Elemento básico de um sistema completo que possui funções bem definidas. Normalmente é ligado ao sistema por conectores, podendo ser facilmente substituído.
<b>Módulo (referindo-se a software)</b>	Parte de um programa aplicativo capaz de realizar uma função específica. Pode ser executado independentemente ou em conjunto com outros módulos, trocando informações através da passagem de parâmetros.
<b>Nibble</b>	Unidade de informação composta por quatro bits.
<b>Octeto</b>	Conjunto de oito bits numerados de 0 a 7.
<b>Operandos</b>	Elementos sobre os quais as instruções atuam. Podem representar constantes, variáveis ou um conjunto de variáveis.
<b>PC</b>	Sigla para programmable controller. É a abreviatura de controlador programável em inglês.
<b>Programa aplicativo</b>	É o programa carregado em um CP, que determina o funcionamento de uma máquina ou processo.
<b>Protocolo</b>	Regras de procedimentos e formatos convencionais que, mediante sinais de controle, permitem o estabelecimento de uma transmissão de dados e a recuperação de erros entre equipamentos.
<b>Rede de comunicação</b>	Conjunto de equipamentos (nós) interconectados por canais de comunicação.
<b>Rede de comunicação mestre-escravo</b>	Rede de comunicação onde as transferências de informações são iniciadas somente a partir de um único nó (mestre da rede) ligado ao barramento de dados. Os demais nós da rede (escravos) apenas respondem quando solicitados.
<b>Rede de comunicação multimestre</b>	Rede de comunicação onde as transferências de informações são iniciadas por qualquer nó ligado ao barramento de dados.
<b>RX</b>	Sigla usada para indicar recepção serial.

<b>Software</b>	Programas de computador, procedimentos e regras relacionadas à operação de um sistema de processamento de dados.
<b>Subsistema de E/S</b>	Conjunto de módulos de E/S digitais ou analógicos e interfaces de um controlador programável.
<b>Time-out</b>	Tempo preestabelecido máximo para que uma comunicação seja completada. Se for excedido procedimentos de retentiva ou diagnóstico serão ativados.
<b>Toggle</b>	Elemento que possui dois estados estáveis, trocados alternadamente a cada ativação.
<b>TX</b>	Sigla usada para indicar transmissão serial.
<b>UCP</b>	Sigla para unidade central de processamento. Controla o fluxo de informações, interpreta e executa as instruções do programa e monitora os dispositivos do sistema.
<b>Upload</b>	Leitura do programa ou configuração do CP.
<b>Word</b>	Unidade de informação composta por 16 bits.