

Comunicado 51

Técnico

Dezembro, 2003
Campinas, SP

ISSN 1677-8464

Utilizando a API HTTPUnit no Sistema Agritempo

Adriano Frazoni Otavian¹
Luciana Alvim Romani²
Sílvio Roberto Medeiros Evangelista³
Marcos Lordello Chaim⁴
Hilton Silveira Pinto⁵

The image shows three screenshots of the Agritempo system's weather forecast interface. Each screenshot displays a table with columns for 'Tempo' (Weather), 'Temperatura' (Temperature), 'Chuva' (Rain), and 'Vento' (Wind). The first screenshot shows a forecast for 'Poucas Nuvens' (Few clouds) with temperatures ranging from 19°C to 23°C and wind directions like NNE and N. The second screenshot, labeled 'segunda) - Previsão Completa', shows a forecast for 'Nublado' (Cloudy) with temperatures from 19°C to 20°C and wind directions like ENE and S. The third screenshot, labeled 'terça) - Previsão Completa', shows a forecast for 'Céu Claro' (Clear sky) with temperatures from 16°C to 34°C and wind directions like ENE, E, and SE.

O "Sistema de Monitoramento Agrometeorológico — Agritempo" (Evangelista et al., 2003) desenvolvido pela Embrapa Informática Agropecuária em parceria com o Centro de Pesquisas Meteorológicas e Climáticas aplicadas à Agricultura (Cepagri) da Unicamp permite aos usuários o acesso às informações meteorológicas e agrometeorológicas de diversos municípios e estados brasileiros. Além de informar as situações climáticas atual e prevista, o sistema alimenta a Rede Nacional de Agrometeorologia (RNA) do Ministério da Agricultura, Pecuária e Abastecimento (MAPA) com informações básicas que orientam o zoneamento agrícola brasileiro.

O Agritempo foi desenvolvido tendo como infra-estrutura o serviço World Wide Web (World Wide Web Consortium, 2003a) (Tanenbaum, 1996) da Internet, as tecnologias Java™ (Sun Microsystems, 2003a) e Java™ Servlets (Sun Microsystems, 2003b) e a plataforma Oracle de banco de dados (Oracle Corporation, 2003).

O sistema utiliza dados recebidos diariamente de estações meteorológicas na geração de boletins agrometeorológicos e na geração de mapas para estiagem, estiagem agrícola

porcentagem de água disponível no solo, temperaturas máximas, mínimas e médias, evapotranspiração potencial, precipitação, precipitação mensal acumulada e número de dias com chuvas no mês. Os dados coletados diariamente pelas estações meteorológicas são recebidos de várias instituições em diferentes formatos e passam por um processo de migração, onde são convertidos para o formato de armazenamento do sistema Agritempo e validados antes de serem inseridos no banco de dados.

Devido à demanda por parte dos usuários de informações sobre a previsão do tempo, foram incorporados ao sistema Agritempo mapas com previsão de 24 horas de tratamento fitossanitário, de necessidade de irrigação, de condições de colheita e de condições de manejo do solo; mapas com previsão de 24 e 48 horas de precipitação, de temperatura mínima e temperatura máxima do ar e mapas com previsão para 24, 48, 72, 96 e 120 horas de ocorrência de geadas em determinadas localidades. Todos os mapas são em relação à data atual e são importantes por alertarem sobre condições climáticas extremas que possam prejudicar determinadas culturas, como geadas ou excesso ou falta de precipitação.

¹ Bacharel em Engenharia de Computação, Técnico de Nivel Superior da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo - 13083-970 - Campinas, SP. (e-mail: adriano@cnptia.embrapa.br).

² M.Sc. em Ciência da Computação, Pesquisadora da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo - 13083-970 - Campinas, SP. (e-mail: luciana@cnptia.embrapa.br).

³ Dr. em Engenharia Elétrica, Técnico de Nivel Superior da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo - 13083-970 - Campinas, SP. (e-mail: silvio@cnptia.embrapa.br).

⁴ Dr. em Engenharia Elétrica, Pesquisador da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo - 13083-970 - Campinas, SP. (e-mail: chaim@cnptia.embrapa.br).

⁵ Dr. em Agronomia Professor da Unicamp, Cepagri/Unicamp - Cidade Universitária "Zeferino Vaz", Av. André Tosello, 209, Barão Geraldo - 13083-886 - Campinas, SP. (e-mail: hilton@cpa.unicamp.br)

Para a automação do processo de atualização dos dados de previsão do tempo, tornou-se necessária a criação de uma aplicação que receberia como entrada um arquivo com os dados de previsão do tempo e teria como saída a atualização dessas informações na base de dados do sistema Agritempo. No entanto, a definição do formato do arquivo de dados e a criação da aplicação demandariam um tempo maior que a simples criação de uma aplicação capaz de recuperar esses mesmos dados através de uma interface já disponível, as páginas HTML.

Este trabalho apresenta o uso da API HTTPUnit para recuperar dados de previsão do tempo utilizados na geração dos mapas a partir do site www.somarmeteorologia.com.br (Somar, 2003) através de uma aplicação desenvolvida com a Interface de Programação de Aplicações (API) HTTPUnit (Gold, 2003). Esta API emula as partes relevantes do comportamento de um navegador web, incluindo comunicação HTTP (Tanenbaum, 1996; World Wide Web Consortium, 2003b), interpretação de páginas HTML (Tanenbaum, 1996; World Wide Web Consortium, 2003e), envio de formulários, execução de funções JavaScript (Sun Microsystems, 2003c) e suporte a cookies (Cookies, 2003) possibilitando que uma aplicação desenvolvida em Java examine as páginas retornadas como texto, como XML DOM (World Wide Web Consortium, 2003c; World Wide Web Consortium, 2003d) ou como objetos definidos na própria API que abstraem formulários, tabelas e links.

API HTTPUnit

HTTPUnit é uma API Java de domínio público e de código-fonte aberto desenvolvida para acessar *sites* na web sem o uso de um navegador. Esta API é um conjunto de classes Java para se testar aplicações web acessíveis através do protocolo HTTP e foi projetada principalmente para testes automatizados de acordo com a metodologia de desenvolvimento eXtreme Programming (XP). Mais informações sobre eXtreme Programming podem ser encontradas em (Jeffries, 2003).

Um aspecto importante desta API é que se pode testar completamente uma aplicação web e não apenas páginas únicas. Devido às informações adquiridas através de formulários ou de cookies, podem ser escritos testes que cobrem toda uma sessão. Por exemplo, se sua aplicação web inclui um carrinho de compras, é possível escrever um teste que realize a autenticação na aplicação (log in), selecione um item, coloque este item no carrinho de compras e finalize a compra. Como os testes são escritos em Java, não há limite para a sua profundidade (Scheinblum, 2003).

O centro da API HTTPUnit é a classe `WebConversation`. Ela é responsável por manipular requisições, por resolver URLs relativas e por manter o contexto da sessão que é feito através de um cookie retornado pelo servidor. Este cookie contém o código de identificação da sessão entre o cliente e o servidor web e deve ser enviado pelo cliente ao servidor em todas as requisições com a finalidade de sua identificação.

Exemplo:

```
WebConversation wc = new WebConversation ();
WebRequest req = new
    GetMethodWebRequest("http://www.agritempo.gov.br
        /index.php");
WebResponse resp = wc.getResponse (req );
```

Uma vez que uma requisição a um servidor web é feita com sucesso, é necessário se interpretar o resultado desta requisição, o que é feito através de ferramentas para manipular elementos HTML, como formulários, tabelas, *links*, botões e *frames* (Gold, 2003).

A API possibilita que aplicações encontrem links através do texto expresso neles e que os utilizem como requisições a novas páginas. *Links* com imagens também podem ser utilizados e são encontrados através do texto alternativo a cada imagem.

Exemplo:

```
WebConversation wc = new WebConversation ();
WebResponse resp =
wc.getResponse("http://www.agritempo.gov.br/index.php");
WebLink link = resp.getLinkWith( "login" );
link.click();
WebResponse jdoc = wc.getCurrentPage();
```

Muitos *sites* fazem uso de tabelas para controlar a formatação das páginas. Pode-se tirar proveito disso utilizando as tabelas como elementos discretos e recuperando qualquer uma de suas células. O conteúdo das células desejadas pode então ser tratado como texto, *links*, formulários, botões ou ainda outras tabelas. Assim pode-se recuperar diretamente células de uma tabela sem que seja necessário interpretar toda a tabela.

Exemplo: suponha-se que a Tabela 1 seja a segunda tabela em uma página.

Tabela 1. Exemplo de tabela HTML.

Estação	Temperatura Máxima
Campinas	18,8
São Paulo	22,4

O seguinte código fará a verificação.

```
String[][]tabela= resp.getTables()[1].as Text();
assertEquals("A String 'Estação' não foi encontrada",
"Estação", tabela[0][0]);
assertEquals("A String 'Temperatura Máxima' não foi
encontrada", "Temperatura Máxima", tabela[0][1]);
assertEquals("A String 'Campinas' não foi encontrada",
tabela[1][0]);
assertEquals("O Valor '18,8' não foi encontrado", "18,8",
tabela[1][1]);
assertEquals("A String 'São Paulo' não foi encontrada",
"São Paulo", tabela[2][0]);
assertEquals("O Valor '22,4' não foi encontrado", "22,4",
tabela[2][1]);
```

O Método `AssertEquals` assegura que dois valores são iguais. Caso não sejam, o teste falhará e uma exceção do tipo `AssertionFailedError` será lançada com a mensagem fornecida ao método.

Uma aplicação HTTPUnit pode verificar os controles e os valores padrões de uma aplicação web; se estiverem

corretos, preenche cada formulário HTML e submete-o para o servidor web, como um navegador real faria clicando um botão de submissão. O envio de um formulário pode ser feito simplesmente obtendo o formulário e chamando seu método "submit".

Exemplo: O código a seguir cria o formulário representado na Fig.1.

```
<form action="login" method="post" target="_blank">
<table width="100%" border="0" cellspacing="0"
Cellpadding="0">
  <tr>
    <td>Acesso Restrito</td>
  </tr>
  <tr>
    <td>Usuário</td>
    <td><input type="text" name="Login" size="10"
maxlength="50"></td>
  </tr>
  <tr>
    <td>senha</td>
    <td><input type="password" name="senha" size="10"
maxlength="50">
    <td align="center"><input type="submit"
value="OK"></td>
  </tr>
</table>
</Form>
```

Fig. 1. Formulário de acesso restrito.

A submissão deste formulário pode ser feita simplesmente obtendo o objeto formulário, fornecendo os parâmetros login e senha e chamando seu método "submit".

```
WebForm form = resp.getForms()[0];
form.setParameterValue( "Login", nome_usuario);
form.setParameterValue( "Senha", senha_usuario);
form.submit();
```

Se nenhum erro ocorrer, a execução do código acima não fornecerá qualquer mensagem. Caso algum erro ocorra, a execução será encerrada e uma exceção será lançada informando o erro.

Aplicação PrevisaoTempoSomar

Devido à necessidade de automação do processo de atualização dos dados de previsão do tempo, tornou-se necessária a criação de uma aplicação que recebesse como entrada um arquivo com os dados de previsão do tempo e tivesse como saída a atualização dessas informações na base de dados do sistema Agritempo. No entanto, a definição do formato do arquivo de dados e a criação da

aplicação demandariam um tempo maior que a simples criação de uma aplicação capaz de recuperar esses mesmos dados através de uma interface já disponível, as páginas HTML.

Utilizando a API HTTPUnit criou-se uma aplicação capaz de recuperar os dados de previsão de tempo disponibilizados na web pela empresa Southern Marine Weather Services S/C Ltda. (SOMAR Meteorologia), parceira do projeto.

A aplicação se conecta ao *site* da empresa e, navegando neste *site*, é capaz de recuperar os dados de previsão do tempo (temperatura, precipitação, intensidade e direção do vento, umidade e possibilidade de geada) para os municípios requisitados e de atualizar os dados na base de dados do sistema Agritempo. É utilizada a estrutura hierárquica do HTML e o conteúdo de células específicas é recuperado e interpretado para inclusão ou atualização no banco de dados do sistema Agritempo.

Algoritmo da Aplicação:

```
para cada estado, faça:
  carrega da base de dados todas as estações desse
estado para cada estação recuperada, faça:
  recupera o nome do município onde fica a
estação
  gera requisição para o site somar com o nome
do município e o estado
  recupera da web a página com a previsão do
tempo para o município
  recupera da página HTML a 1ª tabela
  parser1 para a 1ª tabela
  salva previsões para o dia atual, 24h e 48h
na base de dados
  recupera da página HTML a 2ª tabela
  parser2 para a 2ª tabela
  salva previsões para 72h, 96h e 120h na base
de dados
fim
fim
fim
```

Rotina Parser1: esta rotina faz a interpretação dos dados detalhados de previsão do tempo contidos nas células da primeira tabela da página HTML (Tabela 2) e converte os dados para o formato de representação do banco de dados do sistema Agritempo.

Tabela 2. Exemplo de Tabela HTML com dados detalhados de previsão do tempo.

10/11 (segunda-feira) - Previsão Completa						
Período	Tempo	Temperatura	Chuva	Vento	Umidade	Geadas
Madrugada	Nublado	19° C	0 mm	ENE - 10 km/h	79,00%	Nenhum
Manhã	Poucas Nuvens	19° C	0 mm	ENE - 3 km/h	64,00%	Nenhum
Tarde	Céu Claro	32°C	0 mm	S - 11 km/h	40,00%	Nenhum
Noite	Céu Claro	20°C	0 mm	ESE - 17 km/h	74,00%	Nenhum

11/11 (terça-feira) - Previsão Completa						
Período	Tempo	Temperatura	Chuva	Vento	Umidade	Geadas
Madrugada	Céu Claro	18° C	0 mm	ENE - 10 km/h	79,00%	Nenhum
Manhã	Céu Claro	18° C	0 mm	E - 6 km/h	65,00%	Nenhum
Tarde	Céu Claro	34°C	0 mm	SE - 8 km/h	41,00%	Nenhum
Noite	Poucas Nuvens	21°C	0 mm	E - 13 km/h	73,00%	Nenhum

12/11 (quarta-feira) - Previsão Completa						
Período	Tempo	Temperatura	Chuva	Vento	Umidade	Geadas
Madrugada	Poucas Nuvens	20° C	0 mm	NNE - 16 km/h	79,00%	Nenhum
Manhã	Poucas Nuvens	19° C	0 mm	N - 21 km/h	62,00%	Nenhum
Tarde	Poucas Nuvens	35°C	0 mm	NW - 15 km/h	40,00%	Nenhum
Noite	Poucas Nuvens	23°C	0 mm	N - 15 km/h	66,00%	Nenhum

Algoritmo de parser1:

```

WebTable tab1 = response.getTables()[0];
tempo = tempo_predominante(tab1.getTableCell(1,1),
    tab1.getTableCell(4,1));
temperatura_maxima=maximo(tab1.getTableCell(1,2),
    tab1.getTableCell(4,2));
temperatura_minima=minimo(tab1.getTableCell(1,2),
    tab1.getTableCell(4,2));
chuva=soma(tab1.getTableCell(1,3),
    tab1.getTableCell(4,3));
vento = vento_predominante(tab1.getTableCell(1,4),
    tab1.getTableCell(4,4));
umidade_maxima=maximo(tab1.getTableCell(1,5),
    tab1.getTableCell(4,5));
umidade_minima=minimo(tab1.getTableCell(1,5),
    tab1.getTableCell(4,5));
geada = condicao_predominante(tab1.getTableCell(1,6),
    tab1.getTableCell(4,6));
cria_previsao(tempo, temperatura_maxima,
    temperatura_minima, chuva, vento, umidade_maxima,
    umidade_minima, geada);
salva_previsao;

```

Rotina Parser2: Esta rotina faz a interpretação dos dados resumidos de previsão do tempo contidos nas células da segunda tabela da página HTML (Tabela 3) e converte os dados para o formato de representação do banco de dados do sistema Agritempo.

Algoritmo de Parser2:

```

WebTable tab2 = response.getTables()[1];
int linhas = tab2.getRowCount();
de i=0 ate i<linhas, faça:
    tempo=tab2.getTableCell(i,1);
    Temperatura_maxima=maximo
        (tab2.getTableCell(i,2));
    temperatura_minima=minimo
        (tab2.getTableCell(i,2));
    chuva=tab2.getTableCell(i,3);
    geada=tab2.getTableCell(i,6);
    cria_previsao(tempo, temperatura_maxima,
        temperatura_minima, chuva, geada);
    salva_previsao;

```

Os dados de previsão do tempo recuperados pela aplicação PrevisaoTempoSomar e disponíveis na base de dados do sistema Agritempo são então processados pelo sistema que atualizará os mapas de previsão do tempo disponíveis. Maiores informações sobre a geração dos mapas em (Romani et al., 2003). As Fig. 2 e 3 são exemplos de mapas gerados com dados recuperados pela aplicação previsaoTempoSomar.

Tabela 3. Exemplo de tabela HTML com dados resumidos de previsão do tempo.

<i>Dia</i>	<i>Tempo</i>	<i>Temperatura</i>	<i>Chuva</i>	<i>Geadas</i>
13/11/2003	Pancada de Chuva	20°C/30°C	8 mm	Nenhum
14/11/2003	Chuviscos	16°C/22°C	2 mm	Nenhum
15/11/2003	Nublado com Chuva	17°C/25°C	5 mm	Nenhum
16/11/2003	Nublado com Chuva	21°C/28°C	4 mm	Nenhum
17/11/2003	Chuvas	22°C/27°C	13 mm	Nenhum
18/11/2003	Nublado com Chuva	22°C/27°C	9 mm	Nenhum
19/11/2003	Nublado com Chuva	22°C/27°C	11 mm	Nenhum
20/11/2003	Chuvas	20°C/25°C	7 mm	Nenhum
21/11/2003	Nublado com Chuva	21°C/28°C	6 mm	Nenhum
22/11/2003	Chuvas	21°C/23°C	16 mm	Nenhum
23/11/2003	Chuvas	21°C/25°C	20 mm	Nenhum
24/11/2003	Chuvas	19°C/26°C	7 mm	Nenhum

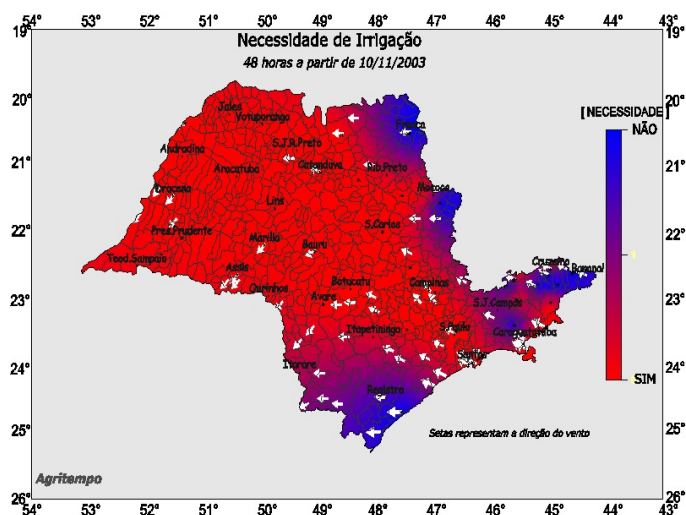


Fig. 2. Mapa de Necessidade de Irrigação para o estado de São Paulo.

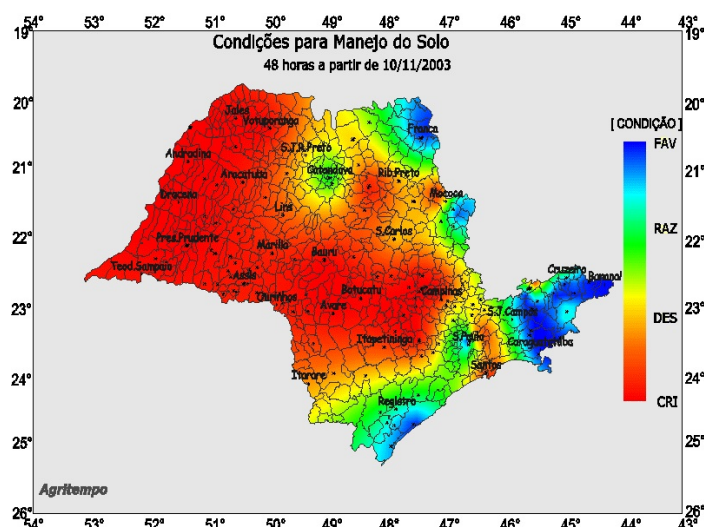


Fig. 3. Mapa de Condições para Manejo de Solo para o estado de São Paulo.

Conclusões

Utilizando recursos da linguagem Java como orientação a objetos e reaproveitamento de classes e recursos da API HTTPUnit como a facilidade de comunicação HTTP e de interpretação de páginas HTML, torna-se fácil criar aplicações capazes de interagir com *sites* na web, sem qualquer alteração nas páginas ou nos *sites* envolvidos.

HTTPUnit é uma ferramenta para XP (eXtreme Programming) desenvolvida principalmente como instrumento de teste, voltada para o teste de interfaces na web e que também pode ser utilizada como ferramenta para integração rápida de *sites* pois permite que uma aplicação Java recupere facilmente dados a partir de páginas HTML.

Devido à demanda por parte dos usuários de informações sobre a previsão do tempo, foram incorporados ao sistema Agritempo mapas com previsão. Todos os mapas são em relação à data atual e são importantes por alertarem sobre condições climáticas extremas que possam prejudicar determinadas culturas.

Para a automação do processo de atualização dos dados de previsão do tempo, tornou-se necessária a criação de uma aplicação que interpretasse um arquivo com os dados de previsão do tempo e atualizasse essas informações na base de dados do sistema Agritempo. No entanto, a definição do formato do arquivo de dados, que envolveria outras pessoas e outras aplicações de conversão de dados, e a criação da aplicação propriamente dita demandariam um tempo maior que a simples criação de uma aplicação capaz de recuperar esses mesmos dados através de uma interface já disponível, as páginas HTML.

A solução apresentada tem se mostrado bastante eficiente dada a necessidade de se obter da web os dados de previsão de tempo atualizados duas vezes ao dia. O desempenho da aplicação é dependente das condições dos servidores e do tráfego na rede, tendo em média 90 minutos como tempo de execução para recuperar e interpretar aproximadamente 600 páginas HTML e salvar os dados na base de dados do sistema Agritempo.

Pode-se concluir que é possível construir aplicações para realizar a integração automatizada entre sistemas distribuídos ou *sites* na web utilizando APIs com código fonte aberto e com as interfaces já disponíveis, como por exemplo as páginas HTML.

Referências Bibliográficas

COOKIES HTTP State Management Mechanism. Disponível em:

< <http://www.ietf.org/rfc/rfc2965.txt?number=2965> > . Acesso em: 10 nov. 2003.

EVANGELISTA, S. R. M.; TERNES, S.; SANTOS, E. H. dos; ASSAD, E. D.; ROMANI, L. A. S.; FRANZONI, A. Agroclima - sistema de monitoramento agroclimatológico. In: CONGRESSO BRASILEIRO DE AGROMETEOROLOGIA, 13., 2003, Santa Maria. Situação atual e perspectivas da agrometeorologia: anais... Santa Maria: UNIFRA: SBA: UFSM, 2003. v.1, p. 603-604.

GOLD, R. HttpUnit CookBook and HttpUnit tutorial. Disponível em: < <http://www.httunit.org> > . Acesso em: 27 jun. 2003.

JEFFRIES, R. E. XProgramming.com: an extreme programming resource. Disponível em: < <http://www.xprogramming.com> > . Acesso em: 10 nov. 2003.

ORACLE CORPORATION. Oracle 9i Databases. Disponível em: < <http://otn.oracle.com/docs/products/oracle9i/content.html> > . Acesso em: 10 nov. 2003.

ROMANI, L. A. S.; EVANGELISTA, S. R. M.; ZULLO JÚNIOR, J.; AGUIAR, D.; FONSECA, M. Geração de mapas agrometeorológicos em tempo real via Internet. In: CONGRESSO BRASILEIRO DA SOCIEDADE BRASILEIRA DE INFORMÁTICA APLICADA À AGROPECUÁRIA E AGROINDÚSTRIA, 4., 2003, Porto Seguro. Anais... Lavras: SBIAGRO, 2003. p. 1-4. 1 CD-ROM.

SCHEINBLUM, J. Test entire Web applications with HttpUnit. Disponível em: < <http://www.zdnet.com.au/builder/program/java/story/0,200034779,20264830,00.htm> > 16 Abril 2002, ZDNet Australia. Acesso em: 10 nov. 2003.

SOMAR METEOROLOGIA. Southern Marine Weather Services. Disponível em: < <http://www.somarmeteorologia.com.br> > . Acesso em: 10 nov. 2003.

SUN MICROSYSTEMS. The source for Java technology. Disponível em: < <http://java.sun.com> > . Acesso em: 10 nov. 2003a.

SUN MICROSYSTEMS. Java servlet technology. Disponível em: < <http://java.sun.com/products/servlet> > . Acesso em: 10 nov. 2003b.

SUN MICROSYSTEMS. Client-Side JavaScript reference. Disponível em < <http://docs.sun.com/source/816-6408-10/contents.htm> > . Acesso em: 10 nov. 2003c.

TANENBAUM, A.S. Computer networks. 3rd ed. New Jersey: Prentice Hall, 1996. 813 p.

WORLD WIDE WEB CONSORTIUM. About the World Wide Web. Disponível em: < <http://www.w3c.org/WWW/> > . Acesso em: 10 nov. 2003a.

WORLD WIDE WEB CONSORTIUM. HTTP - hypertext transfer protocol. Disponível em: < <http://www.w3c.org/Protocols/> > . Acesso em: 10 nov. 2003b.

WORLD WIDE WEB CONSORTIUM. Document object model (DOM). Disponível em: < <http://www.w3c.org/DOM/> > . Acesso em: 10 nov. 2003c.

WORLD WIDE WEB CONSORTIUM. Extensible markup language (XML). Disponível em: < <http://www.w3c.org/XML/> > . Acesso em: 10 nov. 2003d.

WORLD WIDE WEB CONSORTIUM. HyperText markup language - HTML [Home page]. Disponível em: <<http://www.w3c.org/MarkUp/>>. Acesso em: 10 nov. 2003e.

**Comunicado
Técnico, 51**

Ministério da Agricultura,
Pecuária e Abastecimento

Governo
Federal

Embrapa Informática Agropecuária
Área de Comunicação e Negócios (ACN)
Endereço: Caixa Postal 6041 - Barão Geraldo
13083-970 - Campinas, SP
Fone: (19) 3789-5743
Fax: (19) 3289-9594
e-mail: sac@cnptia.embrapa.com.br

1ª edição on-line - 2003

ÓTodos os direitos reservados.

**Comitê de
Publicações**

Presidente: *Luciana Alvim Santos Romani*
Membros Efetivos: *Carla Geovana Macário, José Ruy Porto de Carvalho, Marcia Izabel Fugisawa Souza, Marcos Lordello Chaim, Suzilei Almeida Carneiro.*
Suplentes: *Carlos Alberto Alves Meira, Eduardo Delgado Assad, Maria Angelica Andrade Leite, Maria Fernanda Moura, Maria Goretti Gurgel Praxedis.*

Expediente

Supervisor editorial: *Ivanilde Dispatto*
Normalização bibliográfica: *Maria Goretti Gurgel Praxedis*
Editoração eletrônica: *Área de Comunicação e Negócios*