



ISSN 1677-8464

Uso de XML para Representação de Interfaces com o Usuário

Sergio Aparecido Braga da Cruz¹
Maria Fernanda Moura²

Novas técnicas de desenvolvimento de sistemas têm sido adotadas para prover uma maior modularidade e desacoplamento entre seus componentes. Nestas técnicas, os componentes são agrupados em camadas com funcionalidades relacionadas. O modelo MVC é uma técnica de modelagem, adotada em sistemas interativos, que agrupa os seus componentes sob três perspectivas diferentes (Buschmann et al., 1996):

- *Model* – agrupa componentes relacionados ao processamento e geração de dados do sistema.
- *View* - agrupa componentes relacionados a apresentação de dados de saída do sistema.
- *Controller* – agrupa componentes relacionados à lógica de manipulação dos dados de entrada, adequando-os para utilização pela camada de processamento.

De acordo com Bluschmann et al. (1996) algumas vantagens na utilização do modelo MVC em sistemas interativos são:

- podem ser criados vários componentes definindo vários *Views* para apresentação diferenciada de um mesmo dado. Estes *Views* podem ser usa-

dos simultaneamente permitindo que um mesmo dado seja apresentado de várias maneiras ao mesmo tempo. Os *Views* que serão utilizados podem ser selecionados em tempo de execução;

- a separação em camadas do MVC permite a troca dos componentes na camada *View* e *Controller* sem afetar a camada *Model*. As camadas *View* e *Controller* definem a interface do sistema, que poderá ser então substituída completamente em tempo de execução sem afetar a camada de processamento de dados principal;
- o baixo acoplamento entre a interface (*View* e *Controller*) e a camada de processamento principal (*Model*) aumenta a portabilidade do sistema interativo uma vez que a interface pode ser alterada para se adequar aos elementos de interface disponíveis para uma nova plataforma.

A maior portabilidade dos elementos de interface para diferentes plataformas motivou que este modelo fosse utilizado no desenvolvimento da ferramenta ServCLIPS. Esta ferramenta foi desenvolvida no subprojeto Diagnóstico Remoto, vinculado ao projeto SVTTA (Serviços Virtuais para Transferência de

¹ Mestre em Engenharia Elétrica, Pesquisador da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo – 13083-970 – Campinas, SP. (e-mail: sergio@cnptia.embrapa.br)

² Mestre em Engenharia Elétrica, Pesquisadora da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo – 13083-970 – Campinas, SP. (e-mail: fernanda@cnptia.embrapa.br)

Tecnologia Agropecuária), descrita por Massruhá et al., (1999), e permite a execução de sistemas especialistas que podem ser consultados através da WWW (*World Wide Web*) (Krol, 1993). No contexto do subprojeto Diagnóstico Remoto esta ferramenta foi desenvolvida para permitir a construção de sistemas especialistas para diagnóstico remoto de doenças.

O uso do MVC no desenvolvimento da ferramenta ServCLIPS possibilita que, além da versão disponível na WWW, uma versão *stand-alone* da mesma ferramenta seja mais facilmente criada. Viabilizar esta possibilidade foi um dos requisitos iniciais no desenvolvimento da ferramenta ServCLIPS. Por isso, foi necessário descrever a interface com o usuário de maneira independente da plataforma (WWW ou aplicação *stand-alone*) por meio da criação de uma representação abstrata desta interface expressa na linguagem XML (Harold, 1999).

Estrutura da ferramenta ServCLIPS

A ferramenta ServCLIPS foi desenvolvida através da integração da aplicação CLIPS (*C Language Integrated Production System*) (Giarratano, 1998), responsável pela execução de regras de produção do

sistema especialista (máquina de inferência), com a solução Java Servlets (Hunter & Crawford, 1998) para construção de aplicações para a WWW. As linguagens de programação utilizadas em seu desenvolvimento foram Java e C, que foram integradas por meio do mecanismo JNI (*Java Native Interface*) (Sun Microsystems, 2002a).

A estrutura da ferramenta ServCLIPS (Fig. 1) pode ser dividida em 3 camadas de acordo com o modelo MVC:

- o processamento dos dados corresponde à execução da máquina de inferência do CLIPS (camada *Model*). As classes Java presentes nesta camada encapsulam a comunicação com a máquina de inferência;
- os dados, antes de serem processados pelo CLIPS, precisam ser identificados e transformados numa estrutura de dados adequada (camada *Controller*). A classe ServCLIPS extrai e transforma os dados recebidos para o formato adequado ao processamento pela máquina de inferência;
- o usuário recebe informações do sistema por meio de seu navegador WWW (*View*). O usuário submete novos dados para processamento interagindo com o navegador.

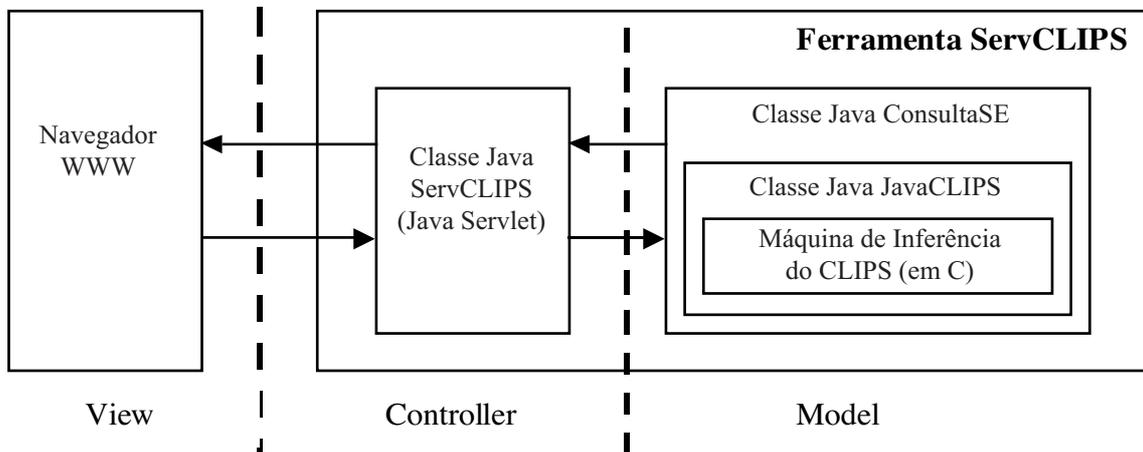


Fig. 1. Estrutura da ferramenta ServCLIPS.

Representação de Interfaces com o Usuário em XML

Para viabilizar a possibilidade de troca da interface (camadas *View* e *Controller*) da ferramenta ServCLIPS é necessário que seja adotado um formato padrão e independente para representação da apresentação dos dados. Esta representação deve-

rá ser transformada de maneira adequada para apresentação de acordo com a plataforma. Os dados a serem considerados aqui correspondem às questões e resultados de diagnóstico gerados pelo CLIPS durante a execução das regras dos sistemas especialistas.

Para representação das questões foram observadas algumas características gerais de elementos de interfaces disponíveis na maioria das plataformas

com interface gráfica, suficientes para descrever as questões geradas pelo CLIPS. Este elemento genérico é formado por um enunciado e por opções. O elemento pode possibilitar que a escolha das opções seja exclusiva ou não (múltiplas opções selecionáveis). Cada opção pode ser representada por um texto ou por uma figura. Algumas opções do menu podem ser pré-selecionadas antes da apresentação ao usuário. Ele pode ser apresentado na forma rolante, lista de opções ou tabular caso a plataforma suporte.

Um elemento genérico na forma de lista de opções com escolhas mutuamente exclusivas pode ser uma abstração para o elemento de interface *radio button* (Cooper, 1995) no ambiente Windows, e elemento de entrada de formulário HTML (HyperText Markup Language) (Ladd et al., 1998) do tipo *RADIO*. Alternativamente pode ser uma abstração para um *listbox* do Windows ou o elemento de formulário HTML do tipo *SELECT* sem múltipla escolha. Os elementos *radiobox* e *listbox* do ambiente Windows e os elementos *RADIO* e *SELECT* de formulários HTML

correspondem à apresentação de um mesmo elemento genérico especificando uma lista de opções com escolhas mutuamente exclusivas.

O resultado do diagnóstico corresponde à geração de um relatório estruturado. Neste relatório são apresentados o nome da doença identificada, o agente causal, sintomas, condições favoráveis ao seu surgimento, o seu controle e disseminação.

A linguagem XML é usada para expressar estes dados gerados pelo CLIPS. Esta linguagem tem sido bastante utilizada na descrição da semântica e da estrutura de dados devido ao seu formato simples e portátil, sua versatilidade e por ser uma solução não-proprietária bem especificada e normatizada pelo W3C (*World Wide Web Consortium*) (Jacobs, 2002). Estas características têm promovido o desenvolvimento de ferramentas que facilitam a utilização do XML nas mais diversas aplicações. Os dados formatados em XML seguem a estrutura descrita pela gramática em DTD (*Document Type Definition*):

```
<!DOCTYPE PAGINA [
  <!ELEMENT PAGINA (TITULOSE,(MENU+ | DOENCA))>
  <!ELEMENT TITULOSE ( #PCDATA)>
  <!ELEMENT MENU (FORMATO?,TIPO_ESCOLHA?,ENUNCIADO,OPCAO,OPCAO+)>
  <!ATTLIST MENU FORMATO (RADIO | ROLANTE | TABULAR) "RADIO">
  <!ATTLIST MENU TIPO_ESCOLHA (SIMPLES | MULTIPLA) "SIMPLES">
  <!ELEMENT ENUNCIADO (#PCDATA)>
  <!ELEMENT OPCA0 (SELECIONADA?,(TEXTOIFIGURA),VALOR)>
  <!ELEMENT SELECIONADA EMPTY>
  <!ELEMENT TEXTO (#PCDATA)>
  <!ELEMENT FIGURA (#PCDATA)>
  <!ELEMENT VALOR (#PCDATA)>
  <!ELEMENT
DOENCA(NOME,AGENTE,FIGURA,SINTOMAS?,CONDICOES?,FATORES?,CONTROLE?,
DISSEMINACAO?,CONTATO?)>
  <!ELEMENT NOME (#PCDATA)>
  <!ELEMENT AGENTE (#PCDATA)>
  <!ELEMENT SINTOMAS (#PCDATA)>
  <!ELEMENT CONDICOES (#PCDATA)>
  <!ELEMENT FATORES (#PCDATA)>
  <!ELEMENT CONTROLE (#PCDATA)>
  <!--ELEMENT DISSEMINACAO (#PCDATA)-->
  <!ELEMENT CONTATO EMPTY>
]>
```

A DTD especifica as regras estruturais de um documento em XML (Harold, 1999). A gramática citada indica que os dados gerados pelo CLIPS são estruturados num documento principal de nome *PAGINA*. Uma *PAGINA* apresenta o título do sistema especialista da doença (*TITULOSE*), mais um menu de opções (*MENU*) ou uma descrição de doença (*DOENCA*) indicando o diagnóstico encontrado. Cada menu de opções tem um tipo (*TIPO_ESCOLHA*), formato (*FORMATO*) e é formado por duas ou mais opções. Cada opção (*OPCAO*) pode ser apresentada

como um texto (*TEXTO*) ou figura (*FIGURA*), pode estar pré-selecionada e ter um valor associado (*VALOR*). Um diagnóstico de doença encontrada (*DOENCA*) deve conter o nome da doença (*NOME*), o agente causal (*AGENTE*) e uma figura (*FIGURA*). O diagnóstico pode conter descrição dos sintomas (*SINTOMAS*), condições propícias de aparecimento (*CONDICOES*), fatores (*FATORES*), controle (*CONTROLE*), forma de disseminação (*DISSEMINACAO*) e contato para maiores esclarecimentos (*CONTATO*).

Um exemplo de documento em XML indicando uma questão gerada pelo CLIPS seria:

```

<PAGINA>
  <TITULOSE>Diagnóstico de Doenças do Milho via WEB</TITULOSE>
  <MENU FORMATO="TABULAR" TIPO_ESCOLHA="SIMPLES">
    <ENUNCIADO>Escolha a figura que mais se assemelha aos sintomas observados:</ENUNCIADO>
    <OPCAO>
      <FIGURA>image41.jpg</FIGURA>
      <VALOR>(figura 41)</VALOR>
    </OPCAO>
    <OPCAO>
      <FIGURA>image42.jpg</FIGURA>
      <VALOR>(figura 42)</VALOR>
    </OPCAO>
    <OPCAO>
      <TEXTO>Nenhuma das opções anteriores</TEXTO>
      <VALOR>(figura 0)</VALOR>
    </OPCAO>
  </MENU>
</PAGINA>

```

Um exemplo de documento em XML indicando o resultado da execução do CLIPS onde uma doença pôde ser identificada seria:

```

<PAGINA>
  <TITULOSE>Diagnóstico de Doenças do Milho via WEB</TITULOSE>
  <DOENCA>
    <NOME>Podridão do Cartucho por Erwinia chrysanthemi</NOME>
    <FIGURA>image01.jpg</FIGURA>
    <AGENTE>Erwinia chrysanthemi</AGENTE>
    <SINTOMAS>murcha e seca de todas as folhas do cartucho decorrente da podridão da base do cartucho. É uma podridão aquosa, de odor tipicamente desagradável. As folhas se desprendem facilmente. Lesões aquosas podem ser observadas nas bainhas das outras folhas.</SINTOMAS>
    <CONDICOES>excesso de água no cartucho por longos períodos sob temperatura entre 30 e 35°C.</CONDICOES>
    <CONTROLE>em plantios irrigados, evitar aplicação de água em excesso.</CONTROLE>
  </DOENCA>
</PAGINA>

```

A camada *Model* da ferramenta ServCLIPS gera dados formatados em XML seguindo a gramática apresentada, conforme ilustrado na Fig. 2. Para que sejam apre-

sentados no formato final, estes dados são pós-processados utilizando ferramentas XML. Este processamento está indicado na Fig. 2 pela função *transf*.

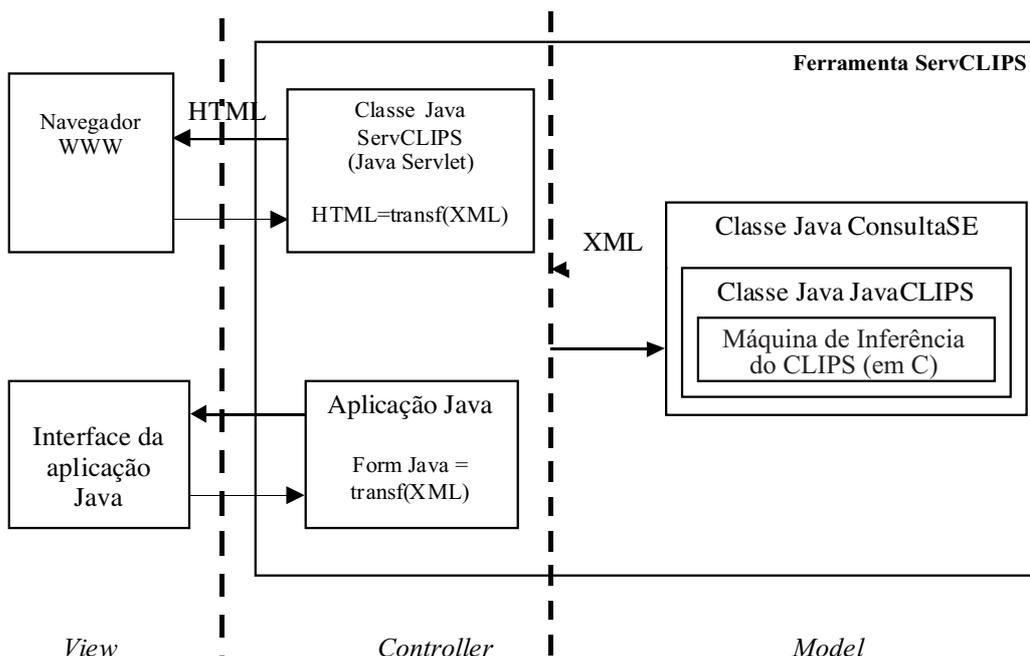


Fig. 2. Estrutura da ferramenta ServCLIPS.

O trecho em itálico indica a parte obtida pela tradução direta do documento XML gerado pelo CLIPS. O restante do documento HTML é gerado pela camada *Controller* do ServCLIPS, e contém informações adicionais que permitem o controle da execução do sistema. A folha de estilo utilizada na tradução do documento XML recebe alguns parâmetros adicionais que auxiliam na montagem do documento HTML fi-

nal. Estes parâmetros são lidos de arquivos de configuração específicos para cada doença e são passados como argumentos para o processador de XSL. Um destes parâmetros, por exemplo, indica a localização de imagens a serem apresentadas nas páginas HTML.

A apresentação da questão ao usuário no navegador WWW é mostrada na Fig. 3.



Fig. 3. Apresentação da questão CLIPS para o usuário.

Para o desenvolvimento de uma versão *stand-alone* (aplicação Java na Fig. 2), a transformação do XML deve gerar um formulário com elementos de interface disponíveis na plataforma Java. Esta transformação poderá ser realizada por meio de um interpretador de documentos XML. Este interpretador pode ser construído utilizando algumas das diversas ferramentas disponíveis para análise e processamento de documentos XML tais como:

- JAXP (*Java API for XML Processing*) - API (*Application Program Interface*) Java para processamento de documentos XML desenvolvida pela Sun MicroSystem (2002b).

- Xerces - *Parser XML* disponível para linguagens Java, C++ desenvolvido dentro do projeto Apache (The Apache Software Foundation, 2002).

Considerações Finais

A transformação da interface descrita em XML para o formulário HTML correspondente à interface gerada pela ferramenta ServCLIPS foi implementada em poucas linhas de código Java com o apoio da ferramenta XALAN (The Apache Software Foundation,

2002) para tratamento de arquivos XML. Esta ferramenta permite a transformação de documentos XML em HTML utilizando uma folha de estilo em XSL. A utilização da folha de estilo em XSL permite que a página HTML gerada possa ter sua apresentação completamente modificada (desde que a abstração MENU se mantenha válida) sem que nenhuma linha de código Java seja alterada. Com isto a interface de um sistema especialista utilizando o ServCLIPS pode gradualmente ser construída e aperfeiçoada. Uma questão gerada pelo sistema e representada por um elemento SELECT de formulários HTML pode ser substituída por uma imagem sensível, sem necessidade de alterar nenhuma linha do sistema especialista ou do código Java.

Uma possível extensão desta abordagem seria a definição de elementos XML para especificação de validação de campos. Estas especificações poderiam ser mapeadas em *scripts* na linguagem JavaScript (Ladd et al., 1998) no desenvolvimento de aplicações para WWW. Para aplicações em Java, estes elementos de validação definiriam testes que deveriam ser realizados sobre campos de um formulário Java antes da submissão dos dados para a camada de processamento.

A abordagem utilizada se restringiu às necessidades de interface do ServCLIPS, porém uma abordagem genérica conhecida como XForms (Michel, 2002) está sendo desenvolvida pelo W3C. Utilizando o XForms podem ser descritos desde o tratamento dos eventos no formulário, cálculo automático de campos até a validação de cada campo. Como resultado da submissão de um formulário XForms é gerada uma instância de dados formatado em XML. A especificação do XForms é relativamente recente e não existem muitas ferramentas permitindo o processamento de documentos neste formato. Algumas ferramentas comerciais, como por exemplo a WebAccess 2.0 (Mozquito Technologies, 2002), implementam o processamento de formulários de acordo com a especificação 1.0 do XForms demonstrando a viabilidade desta abordagem no tratamento de formulários. Algumas ferramentas em versão *beta*, tais como a X-Smiles (X-Smiles Org., 2002) e Chiba (Open Source Development Network, 2002) implementam parte da especificação 1.0 do XForms. Apesar desta tecnologia ainda não ser muito utilizada, é importante acompanhar a sua evolução uma vez que seu uso permitirá uma maior automação no processo de tratamento de formulários independente de plataforma. Uma abordagem bem menos complexa foi necessária no ServCLIPS e o uso de ferramentas gratuitas facilitou a implementação da solução. No ServCLIPS

foram necessárias descrições simples de interface, onde questões enviadas ao usuário são descritas basicamente por um elemento de interface do tipo questão de múltipla escolha. A descrição deste elemento de interface usando XML nesta situação foi mais simples e atendeu as necessidades da ferramenta. A descrição de formulários mais elaborados requer uma evolução da gramática DTD utilizada no ServCLIPS.

Uma abordagem mais simples que o XForm poderia ser utilizada para criação de uma *framework* para geração e tratamento de formulários, que geralmente é uma atividade trabalhosa e repetitiva.

Referências Bibliográficas

THE APACHE SOFTWARE FOUNDATION. *The Apache XML Project*. Disponível em: <<http://xml.apache.org>>. Acesso em: 21 jan. 2002.

BUSCHMANN, F.; MEUNIER, R.; ROHNERT, H.; SOMMERLAD, P.; STAL, M. *Pattern-oriented - software architecture*. Reading: Wiley, 1996. v. 1.

COOPER, A. *About face: the essentials of user interface design*. New York: Hungry Minds, 1995. 580 p.

GIARRATANO, J. C. *CLIPS: user's guide, version 6.10*. Boston: International Thompson Publishing, 1998. 154 p.

HAROLD, E. R. *XML bible*. Foster City: IDG Books Worldwide, 1999. 1015 p.

HUNTER, J.; CRAWFORD, W. *Java servlet programming*. Beijing: O' Reilly, 1998. 510 p. (The Java Series).

JACOBS, I. *About the World Wide Web Consortium (W3C)*. Disponível em: <<http://www.w3.org/Consortium/>>. Acesso em: 21 jan. 2002.

KROL, E. *The whole internet: user's guide & catalog*. Sebastopol: O'Reilly, 1993. 376 p.

LADD, E.; O'DONNELL, J.; ABLAN, J.; BANICK, S.; CASSADY-DORION, L.;

CHANDAK, R.; DOHERTY, D.; ELLSWORTH, M.; SANTA MARIA, P.; MORGAN,

M.; MORRISON, M.; NILES, M.; SLOAN, B.; SUTTER, R. *Platinum edition using HTML 4.0, Java 1.1 and JavaScript 1.2*. 2.ed. Indianapolis: QUE, 1998. 1395 p.

MASSRUHÁ, S. M. F. S.; SOUZA, E. de; ROMANI, L. A. S.; CRUZ, S. A. B. Virtual services for agricultural technology transfer. In: EUROPEAN CONFERENCE OF THE EUROPEAN FEDERATION FOR INFORMATION TECHNOLOGY IN AGRICULTURE, FOOD AND THE ENVIRONMENT – EFITA 99, 2., 1999, Bonn. *Role and potential of IT, Intranet and Internet for advisory services*. Bonn: Universität Bonn-ILB, 1999. p. 53-62.

MICHEL, T. *XForms - the next generation of Web forms*. Disponível em: <<http://www.w3.org/MarkUp/Forms/>>. Acesso em: 21 jan. 2002.

MOZQUITO TECHNOLOGIES. *Mozquito Technologies: reforming the Web*. Disponível em: <<http://www.mozquito.com/>>. Acesso em: 04 fev. 2002.

OPEN SOURCE DEVELOPMENT NETWORK. *SourceForge: Project Info – Chiba*. Disponível em: <<http://sourceforge.net/projects/chiba/>>. Acesso em: 04 fev. 2002.

SUN MICROSYSTEMS. *Java native interface specification*. Disponível em: <<http://java.sun.com/j2se/1.3/docs/guide/jni/spec/jniTOC.doc.html>>. Acesso em: 18 jan. 2002a.

SUN MICROSYSTEMS. *Java Technology & XML documentation*. Disponível em: <<http://java.sun.com/xml/docs.html>>. Acesso em: 21 jan. 2002b.

X-SMILES ORG. *X-Smiles: an open xml-browser for exotic devices*. Disponível em: <<http://www.x-smiles.org/>>. Acesso em: 04 fev. 2002.

Comunicado Técnico, 11

MINISTÉRIO DA AGRICULTURA,
PECUÁRIA E ABASTECIMENTO



Embrapa Informática Agropecuária Área de Comunicação e Negócios

Av. Dr. André Tosello s/nº
Cidade Universitária - "Zeferino Vaz"
Barão Geraldo - Caixa Postal 6041
13083-970 - Campinas, SP
Telefone/Fax: (19) 3789-5743
E-mail: sac@cnptia.embrapa.br

1ª edição

© Embrapa 2001

Comitê de Publicações

Presidente: Francisco Xavier Hemerly

Membros efetivos: Amarindo Fausto Soares, Ivanilde Dispatto, Marcia Izabel Fugisawa Souza, José Ruy Porto de Carvalho, Suzilei Almeida Carneiro

Suplentes: Fábio Cesar da Silva, João Francisco Gonçalves Antunes, Luciana Alvim Santos Romani, Maria Angélica de Andrade Leite, Moacir Pedroso Júnior

Expediente

Supervisor editorial: Ivanilde Dispatto

Normalização bibliográfica: Marcia Izabel Fugisawa Souza

Capa: Intermídia Publicações Científicas

Editoração Eletrônica: Intermídia Publicações Científicas