



XXII CBBB

CONGRESSO BRASILEIRO DE
BIBLIOTECONOMIA, DOCUMENTAÇÃO
E CIÊNCIA DA INFORMAÇÃO

08 A 11 DE JULHO DE 2007

IGUALDADE E DIVERSIDADE NO ACESSO À INFORMAÇÃO:
DA BIBLIOTECA TRADICIONAL À BIBLIOTECA DIGITAL

CENTRO DE CONVENÇÕES ULYSSES GUIMARÃES
BRASÍLIA - DF - BRASIL



Área Temática: Tecnologia da Informação

Sub-área: Ferramentas tecnológicas no tratamento da informação digital

**Título do Trabalho: SOFTWARE LIVRE: MODELOS DE SELEÇÃO COMO
SUBSÍDIO À GESTÃO BIBLIOTECÁRIA**

Autor: José Fernando Modesto da Silva

Instituição: Universidade de São Paulo

Escola de Comunicações e Artes

Departamento de Biblioteconomia e Documentação

Av. Prof. Lúcio Martins Rodrigues, 443

CEP: 05508-900 – São Paulo – SP

E-mail: fmodesto@usp.br

SOFTWARE LIVRE: MODELOS DE SELEÇÃO COMO SUBSÍDIO À GESTÃO BIBLIOTECÁRIA

José Fernando Modesto da Silva*

Resumo

O software torna-se uma alternativa confiável à informatização de bibliotecas. Neste sentido, o texto comenta alguns aspectos que caracterizam o software livre. Destaca o potencial tecnológico representado para a atividade bibliotecária. Desta forma, aborda a questão da seleção de programas por meio da utilização de modelos de avaliação de produtos livre de código aberto. Espera-se contribuir para o esclarecimento da comunidade sobre a alternativa tecnológica e os seus procedimentos de escolha de recursos.

Palavras – chave: Software livre; Código-fonte aberto; Automação de biblioteca; seleção de software; Biblioteca; Informática.

Introdução

Uma alternativa cada vez mais confiável, e destinada à informatização das bibliotecas é a adoção do software livre de código-fonte aberto, também conhecido como software livre (*free software*) ou fonte aberta (*open source*). Uma opção que tem conquistado significativo espaço no mercado das tecnologias de informação, merecendo atenção dos países em desenvolvimento.

No Brasil, o setor público, tem realizado ações para o uso desses programas, percebendo que é uma alternativa não só econômica ao pagamento de *royalties*, mas o de minorar a dependência tecnológica por meio da liberdade de acesso e uso do conhecimento digital pleno, sem códigos restritos.

Com relação ao setor das bibliotecas ou serviços de informação similares, este não ocupa fatia significativa no mercado de tecnologia da informação - TI, mas também não é área inexplorada. Embora sem dados conclusivos sobre o setor, a simples observação de produtos e serviços existente indica o interesse despertado. Apesar do mercado potencial, na área biblioteconômica é flagrante o desnível tecnológico. Algumas bibliotecas (especializadas, universitárias, públicas, e escolares), adotam complexos sistemas integrados de gestão. Outras tantas se encontram, ainda, na idade da “ficha lascada”, ou quando muito, utilizando recursos computacionais obsoletos (Modesto, 2003).

Comentários à parte, há necessidade de políticas públicas mais consistentes para área das bibliotecas (públicas e escolares), de forma a envolvê-la com o ambiente de informação compatível com o século 21. O século

* Professor do Departamento de Biblioteconomia e Documentação da Escola de Comunicações e Artes da Universidade de São Paulo
E-mail: fmodesto@usp.br

das redes digitais e da interoperabilidade de sistemas, das ações de inclusão digital da sociedade.

O propósito deste texto é comentar alguns dos aspectos que caracterizam o software livre. Destacar o potencial que representa para a atividade bibliotecária, bem como identificar algumas fontes de informação sobre ferramentas disponíveis para aplicação na biblioteca. Neste contexto, enfoca-se a questão da seleção de programas por meio da utilização de modelos de avaliação. Neste sentido, espera-se contribuir para melhor esclarecimento da comunidade sobre a alternativa tecnológica e os seus procedimentos de escolha de recursos. Certamente, não se esgota o tema e muito menos se aborda toda a problemática envolvendo o software livre de código aberto.

Compreendendo o Software

O uso do termo “software” é utilizado para explicar o componente imaterial que gerencia os aparelhos eletrônicos dos mais simples aos mais revolucionários. É uma lógica de comunicação entre homem e máquina, baseada em linguagem feita de 0 e 1. Os programadores conhecem essa lógica de fornecer instruções ao computador. O primeiro estágio é selecionar uma linguagem de programação para elaborar instruções, conhecida pelo nome de código fonte. É a estrutura de software legível e modificável apenas pelos programadores. Para ser compreendido pelo computador, o código fonte necessita ser convertido para o correspondente código executável ou binário.

Essa operação é efetuada por um programa denominado compilador. Nesta fase, o software deixa de ser compreensível para os programadores, que para isso precisam acessar o código fonte, se este estiver disponível. O advento do software proprietário, proposto pelas empresas de informática, impõe rígidas leis de copyright e patentes, obrigando a comercialização de programas apenas em formato binário sem o acesso ao respectivo código fonte (Ravaglia, 2004).

Conceito de Software Livre

Para compreender a dimensão do software livre é preciso rever a sua origem situada mais ou menos no início da década de 1970, período no qual computadores não eram de fácil acesso e nem o software detentor de licenças restritivas. É uma época em que os programas eram específicos para uma determinada arquitetura de máquina, e quando ocorria alguma mudança na arquitetura, os programas eram reescritos. Situação que propiciava um ambiente de cooperação e compartilhamento de informações e das fontes desenvolvidas, entre os usuários. Os membros deste ambiente denominavam-se *hackers* pela facilidade de estudar, modificar e desenvolver os códigos dos programas, compartilharem conhecimento e evitar a duplicação de esforços. Com a evolução da indústria de informática e o estabelecimento de um modelo de negócio dissociando software e hardware como partes integrantes geram, no princípio da década de 1980, o aparecimento do software com código fechado ou proprietário amparado em legislação da propriedade intelectual e em

contratos de licenciamento para uso (Porcel Iturralde e Rodríguez Mederos, 2005). Esses acontecimentos desencadeiam um movimento de oposição, na década de 1980, denominado software livre que vai se consolidando, posteriormente, ao longo dos anos em um movimento baseado em quatro conceitos básicos (Bacic, 2003):

- A liberdade de executar o programa, para qualquer propósito.
- A liberdade de estudar o funcionamento do programa, e adapta-lo para as necessidades. O acesso ao código-fonte é requisito para esta liberdade.
- A liberdade de distribuir cópias de modo a ampliar as possibilidades de acesso a tais programas.
- A liberdade de aperfeiçoar o programa e liberar os aperfeiçoamentos, de modo que a comunidade se beneficie, sem gastos adicionais.

O termo “software livre” não é sinônimo de gratuidade, mas de liberdade. Liberdade que o usuário tem de copiar, executar, estudar e modificar o programa numa espiral ascendente de inovações tecnológicas, baseada na cooperação e na livre circulação de conhecimento. O software livre é um movimento baseado no princípio do compartilhamento do conhecimento e na colaboração humana formando uma inteligência coletiva conectada pelas redes eletrônicas.

No âmbito bibliotecário, vem se cristalizando uma postura de promoção e difusão do livre acesso à informação, especialmente em unidades de caráter público, em cuja missão há agregado este princípio. Neste sentido, o Manifesto da Federação Internacional de Associações de Bibliotecas e Instituições – IFLA, sobre o uso da Internet, assinala que para atingir-se a igualdade, a liberdade ao entendimento e paz mundial é fundamental o livre acesso aos conteúdos das redes eletrônicas, o que justifica compreender que a aplicação do software livre no desenho e criação dos serviços de informação resulta na concretização dos objetivos colocados em execução sobre a idéia de uma sociedade da informação mais justas, baseada na liberdade de expressão, no livre acesso à informação e na cooperação social (Manifesto, 2002). Neste sentido, o movimento bibliotecário e o movimento do software livre são composições naturais. Ambos estimulam um processo de aprendizagem e uma cooperação baseada no compartilhamento de informações.

Conceito de Código-Fonte Aberto

O movimento do código-fonte aberto, ou simplesmente código aberto surge, em 1997, como dissidência ao movimento do software livre devido ao viés ideológico relativos a aspectos políticos, sociais e de liberdade empregado pela Fundação do Software Livre – FSF (*Free Software Foundation*), entidade criada para expansão do movimento. O entendimento é que o software deve ser aberto não por questões de liberdade, mas por ser o modelo de código aberto mais eficiente técnica e econômica. Se os programadores podem ler, alterar e distribuir o código, a evolução do programa surge com naturalidade (Saleh, 2004). Em 1997, é publicado a “Definição de Código-Fonte Aberto” indicando o

que caracteriza este tipo de software, e procurando distanciar-se dos preceitos ideológicos do movimento do software livre. Entretanto, acaba dando abertura para que o software de código aberto possa adotar licenças que o transforme em sistema proprietário, e comercializado como tal. Peres apud Seleh (2004) relaciona vários itens relativos à definição de código aberto, dos quais destacamos:

- 1) O software deve ser distribuído sem nenhuma exigência de pagamento; incluindo o código-fonte ou indicação de como obtê-lo.
- 2) A licença deve permitir modificações e trabalhos derivados no programa e sua distribuição nos mesmos termos da licença original. Ela não pode discriminar nenhum tipo de pessoa ou grupo, e nem qualquer tipo de uso do programa.
- 3) Os direitos sobre o programa devem ser definidos somente através de licença, não sendo permitido nenhum mecanismo adicional. Evita-se que o programa tenha uma licença livre obrigando o usuário, por exemplo, a concordar com os termos de não divulgação.

A adoção do termo “código-fonte aberto” ou “código aberto” (*open source*) passou a ser mais utilizado no universo de TI por ser mais assimilável a executivos e usuários comerciais; e para evitar a confusão existente entre “software livre” e “software grátis” uma vez que os dois termos no idioma inglês são designados pela mesma palavra: “*free*”. De qualquer forma, o software livre e/ou de código aberto é uma mudança radical no desenvolvimento de sistemas, ao contrário dos softwares comerciais cuja cópia, execução ou modificação é proibida sem a prévia autorização do fabricante que detém os direitos autorais sobre o produto. É importante para os gestores de bibliotecas, que tencionem optar pelo modelo livre de informatização, procurem se informar dos recursos existentes e conhecer um pouco mais sobre o assunto. Neste aspecto, repositórios para se buscar programas em geral ou visualizar o nível de desenvolvimento, são:

- SourceForge – listando mais de 100 mil projetos de software livre.
- Codehaus – repositório para armazenamento de projetos de código aberto mais focado em linguagem Java.
- ObjectWeb – comunidade virtual de código aberto comprometida em desenvolver e tornar disponíveis programas de qualidade.
- Tigris - comunidade de código aberto focada na construção de ferramentas para desenvolvimento de software de colaboração.
- Java.net: repositório de códigos para sistemas em linguagem Java.

Repositório para a área de biblioteconomia, pode ser consultado o *Open Source Systems for Libraries* (<http://www.oss4lib.org/>). Apresenta informações sobre diversos projetos dedicados ao ambiente bibliotecário: protocolo Z39.50; programas para manipulação de registros MARC (*Machine-Readable Cataloging*); e sistemas integrados para biblioteca. Outro aspecto interessante,

na medida em que o software livre ganha espaços no ambiente biblioteconômico, organizações profissionais constituem grupos de interesse no assunto, caso do *Library and Information Technology Association* – LITA, com o *Open Source Systems Interest Group*, criado em 2000, com o objetivo de encorajar as bibliotecas e bibliotecários a participarem ativamente da comunidade de código aberto, desenvolvendo, adaptando, melhorando, e estabelecendo soluções livre. Mesmo, grupos de profissionais individualmente agregam-se em fóruns de interesse sobre o tema. Estas situações, na área da Biblioteconomia, contribuem para o crescente surgimento de bibliotecários com habilidade em programação e desenvolvimento de sistemas comerciais ou trabalhando com software livre para bibliotecas. Tal fato aos poucos suavizará, certamente, a adesão ao novo cenário tecnológico.

A Seleção de Software Livre

Os gestores de biblioteca necessitam estar plenamente consciente da escolha do software livre. Um pacote comercial ocupa menos tempo de avaliação e teste, bastando ajustar à biblioteca. Um software livre, da mesma forma, deve ser avaliado e experimentado, mas não necessariamente no mesmo tempo e com a mesma característica de procedimentos. Independente de qualquer questionamento sobre software livre ou comercial é sempre recomendável à biblioteca não perder o foco do planejamento. Assim, é preciso avaliar os custos para manter em funcionamento um sistema computacional, desde a aquisição, instalação, atualização e a manutenção. Informações importantes para entender os dispêndios envolvidos na aquisição da tecnologia. O bibliotecário, enquanto gestor, não deve analisar a tecnologia como despesa, mas como investimento de retorno positivo ao longo do tempo. A escolha do software a ser adotado, também é um projeto desafiador, pois com os benefícios oferecidos pelos sistemas, acompanham riscos como: problemas de compatibilidade, usabilidade e mesmo legalidade. Corte e col. (2002) salienta que não há sistema ideal, e independente do acerto na escolha, o atendimento aos requisitos funcionais e de desempenho podem não ser completamente atendidos. Por esse motivo, é importante que a biblioteca defina seus requisitos complementares a certificação de que as funções básicas e necessárias estejam contempladas. As bibliotecas na escolha do software proprietário ou comercial defronta-se com restrições relativas ao:

- a) Alto custo do sistema;
- b) Código-fonte fechado, impedindo acesso às informações de segurança e qualidade do produto;
- c) Aprisionamento tecnológico, manifesta na dependência do fornecedor realizar melhorias, caso estejam nos seus planos para o produto.

Os benefícios estão centrados no suporte dos produtos garantindo infraestrutura de auxílio aos clientes (instalação, manutenção, operação, treinamento

etc.) na solução de problemas. Com relação ao software livre, as razões para a escolha geralmente incluem:

- a) Baixo custo;
- b) Código aberto, com o benefício do acesso e revisão da programação e na baixa incidência de erros (*bugs*);
- c) Arquitetura de desenvolvimento aberta, com projetos desenvolvidos colaborativamente;
- d) Qualidade decorrente da forma de desenvolvimento requerendo que os projetos sejam bem gerenciados para resultar em produtos bons e estáveis.

Entretanto, apesar dos benefícios citados, a adoção do software livre apresenta dificuldades na escolha. Atualmente, há um vasto número de projetos livres, oriundos desde “esforços individuais de baixa qualidade a soluções corporativas de alta qualidade” (OpenBRR, 2005). Alguns outros desafios enfrentados na escolha do software livre podem ser:

- A variedade de opções em algumas categorias de programas;
- Ausência de suporte profissional na maioria dos pacotes distribuídos;
- Baixa longevidade dos programas, em especial se não mantido por organizações;
- Volatilidade do produto, sempre em constante mudança. Aspecto que obriga o usuário a estar preparado para acompanhar e implementar as atualizações e alterações rápidas nos programas;
- Imaturidade dos projetos, geralmente desenvolvido pelo esforço individual dos programadores (*hobby* ou entusiasmo) interessados em criar algo, mas cujo interesse ou objetivo pode mudar deixando órfão o sistema.

Diante do cenário, o software livre apresenta uma variedade de níveis de qualidade. Programas que amplamente utilizados acabam evoluindo e tornando-se produtos compatíveis ou melhores que seus similares comerciais; até produtos deficientes que apesar de uso livre oferecem mais riscos que benefícios. Na questão da escolha de software comercial, a literatura da área, em português, apresenta trabalhos como o de: Corte e col. (1999 e 2002); Viana (2001) e Café e col.(2001); e Balby (2002), entre outros, que abordam critérios para avaliar programas. Na seleção de software livre, o bibliotecário necessita, também, estabelecer procedimentos avaliativos pertinentes às características deste produto, e que podem ser acrescidos de critérios relativos aos produtos comerciais, peculiares a qualquer tipo programa. Para determinar se um produto de código aberto é adequado para uma organização, há uma série de modelos desenvolvidos para análise de maturidade de um produto código aberto. Não só é uma boa maneira de encontrar e obter produtos interessantes, mas ainda, afastar-se de projetos imaturos e de pouca qualidade. Tais modelos se propõem servir como ferramenta útil para orientar a discussão sobre

aplicação de software livre no ambiente de trabalho. Neste sentido, podem ser citados modelos específicos de avaliação de software livre como:

Open Source Maturity Model (OSMM), projetado para auxiliar as organizações a implantarem software de código aberto. É um processo estruturado em três fases. Em cada fase são pontuados os principais elementos de um produto: software, suporte, documentação, treinamento, integração do sistema, e serviços profissionais. Em resumo as fases são: A) Fase 1, avaliação dos elementos do produto: a organização avalia cada elemento do produto em um processo de quatro passos: define requisitos da organização; recursos existentes; avalia maturidade do elemento; e nomeia valores ao elemento. O resultado desta fase é um conjunto de valores para cada elemento principal do produto. B Fase 2, aplicação de valores aos elementos do produto: nem todo elemento tem o mesmo grau de importância, mas o software é fundamental; suporte é crítico; e a documentação é necessária, mas menos importante que os dois elementos anteriores. Nesta fase são aplicadas pesos às contagens auferidas pelos elementos individuais, refletindo o nível geral de importância na maturidade do produto. Cada organização é livre para ajustar os valores padrões que reflitam suas necessidades particulares. C) Fase 3, cálculo das contagens gerais do produto para determinar se o produto é satisfatório para as necessidades da organização. A contagem também pode ser avaliada para determinar problemas com o produto (Navica, 2007). No quadro abaixo se visualiza uma planilha para o modelo:

| | fase 1, avaliação dos elementos do produto | | | | Phase 2 | Phase 3 |
|------------------------|--|---------------------|----------------------|-----------------------|-------------------|--------------------------------------|
| Elementos | definição requisitos | recursos existentes | maturidade elementos | valores dos elementos | valores dos pesos | pontuação para maturidade do produto |
| Software Produto | | | | | | |
| Suporte | | | | | | |
| Documentação | | | | | | |
| Treinamento | | | | | | |
| Integração do produto | | | | | | |
| Serviços profissionais | | | | | | |

Quadro 01: Modelo de Avaliação de Maturidade de Software Código Aberto

Fonte: Navica, 2007

The Capgemini Open Source Maturity Model é um modelo para determinar se o produto de código aberto atende satisfatoriamente às necessidades do usuário (Capgemini, 2003). Trabalha com indicadores de avaliação, alguns dos quais podem ser visualizados no Quadro 02.

| Indicadores | Comentários |
|---|---|
| Idade (Produto) | Maior longevidade, menor descontinuidade |
| Pontos de distribuição (Produto) | Pontos definidos maior a chance de mercado. |
| Comunidade de desenvolvimento (Produto) | Quanto maior o grupo de desenvolvedores ativos menores a chance de descontinuidade. Grupo grande requer organização para efetividade do trabalho coletivo. |
| Hierarquias humanas (Produto) | Projetos sob única liderança de controle tendem durar pouco tempo. Projetos com controle delegado a outros membros ativos, exploram novos caminhos mantendo a versão estável. |
| Licenciamento (Produto) | Códigos abertos apresentam variadas opções de licenças que dizem muito sobre o modo como o sistema foi planejado para uso. Alguns são muito restritivos, que se tornam assunto da discussão quanto a serem ainda considerados código aberto. |
| Modularidade (Integração) | Produto de código aberto ganha mercado compartilhando, desperta interesse em partes da sua funcionalidade. Desenvolvedor pode realizar esquema de licenciamento mais flexível (protegendo o núcleo, liberando outras partes), possibilitando desenvolvimentos comerciais com partes do sistema. Não sacrifica princípio de código aberto. |
| Padrões (Integração) | Software comercial fixa seu próprio padrão para proteger o investimento no produto. Produtos de código aberto aderem aos padrões para sua difusão e aceitação |
| Suporte (Usuário) | Alguns produtos o suporte é por e-mail ao desenvolvedor. Outros mantêm um ou mais grupos de discussão. Alguns mantêm discussão ativa agrupando um número grande de membros oferecendo auxílio. A maneira como o apoio é fornecido diz muito sobre seriedade para com os usuários. |
| Facilidade de uso (Use) | Produto torna-se popular surgem suportes independentes de cursos ou treinamento. Indica que o produto se tornou maduro. Geralmente, a indicação é notada quando os usuários começam a escrever documentos específicos sobre o uso do software. |
| Comunidade de usuário (Aceitação) | Há produtos que não geram barulho; outros têm vários grupos de discussão ocupados. Quando um código aberto é bem recebido surge uma explosão de pedidos de usuário, sugestões e relatórios de problema. No grupo de discussão circulam grande volume de mensagens. É a puberdade do código aberto. |
| Penetração de mercado (Aceitação) | A base instalada informa a importância do produto para os usuários. Produto com grande base instalada fornece um estímulo adicional para formar as |

| | |
|--|--|
| | comunidades discutindo o programa. A base instalada indica um produto mais maduro. |
|--|--|

Quadro 02: Modelo de Maturidade de Código Aberto do Capgemini

Fonte: Capgemini, 2003, Adaptação Fernando Modesto

Business Readiness Rating (BRR), é um modelo para avaliação de pacotes de software livre de acordo com as necessidades dos usuários, desenvolvido pelo SpikeSource (o Centro para Investigação de Software Livre na Carnegie Mellon West) e a Intel Corporation. Propõe padronizar diferentes tipos de avaliações, e neste sentido, o modelo é aberto visando estimular aos usuários compartilharem suas avaliações. O processo é dividido em quatro fases. A primeira, “avaliação rápida”, visa selecionar produtos para uma lista de candidatos à avaliação pelo modelo. Na segunda fase é quantificada a importância de categorias ou métricas do programa adotando-se pesos (1 – mais alta, 12 – mais baixa). As categorias avaliadas são: funcionalidade, usabilidade, qualidade, segurança, desempenho, escalabilidade, arquitetura, suporte, documentação, adoção (publicações sobre), comunidade, e profissionalismo. A terceira fase envolve a coleta de dados e seu processamento para cálculo dos valores coletados em cada categoria na fase anterior. Por fim, na fase 4, é feita a tradução dos dados com os índices das categorias e os fatores de peso com indicação do índice de avaliação do software. Em sendo o BRR um modelo aberto, pode ser customizado, de forma a ser aplicado a quaisquer situação do ambiente de código aberto. É interessante destacar, que a fase 1 do modelo é um filtro inicial, que apesar de simples permite aos usuários rapidamente acrescentarem ou excluírem pacotes com confiança (OpenBRR, 2005). Neste sentido, alguns indicadores usados nesta fase e que não são exaustivos, podendo a critérios de o usuários serem inseridos, são:

- Qual a situação legal ou de licenciamento do software?
- Adere padronizações e padrões pertinentes à atividade do usuário?
- Há algum usuário de referência para o sistema?
- Alguma organização conhecida é associada aos esforços de desenvolvimento?
- Qual é a linguagem de programação utilizada?
- Suporta internacionalização e localização para a linguagem de sua região?
- Existem avaliações de terceiros sobre o software?
- Há livros, artigos ou textos publicados sobre o software?
- Tem sido acompanhado por analistas da área ou institutos de avaliação?

Como observado, indagações básicas que possibilitam uma leitura mais atenta sobre o produto livre pesquisado.

Conclusão

Compreender o significado do software livre ou de código aberto é importante para apoiar de maneira consciente a opção de informatização da biblioteca. Uma escolha não apenas baseada na gratuidade, mas nos benefícios do aprendizado decorrente desde o processo de seleção e avaliação do produto até o conhecimento tecnológico envolvido. A escolha do melhor produto de código aberto deve estar argumentada em procedimento sistematizado por critérios. É importante, também, para aumentar o grau de confiabilidade na adoção desta alternativa livre. O software livre embora seja uma alternativa ao mercado comercial, não é sinônimo de facilidades, ou como já mencionado de gratuidade, mas de liberdade de uso. Certamente, há um custo financeiro envolvido que pode ser maior ou menor ao exigido por um produto comercial. Entretanto, a principal dificuldade enfrentada pelas bibliotecas deve ser a mudança cultural. Migrar de plataforma (Windows - gráficos e intuitivos, para Linux), ou de sistema fechado e suportado por fornecedor para sistemas abertos e coletivamente gerenciado pode ser um grande desafio. O software livre requer investimento de tempo e treinamento no domínio das operações, manutenção e aprimoramento. Uma salutar tendência, na área da Biblioteconomia, é o surgimento crescente de bibliotecários com habilidade em programação e desenvolvimento de sistemas. Alguns trabalhando com software livre, desde a criação e/ou aprimoramento de produtos específicos para bibliotecas, fato que aos poucos suavizam a adesão ao novo cenário tecnológico. Neste aspecto, o propósito de um modelo de avaliação de software livre, contribui para que profissionais e organizações enfrentem com tranquilidade o desafio de criar ou escolher um produto maduro. Localizar, avaliar e escolher um produto de código aberto em acordo com as necessidades da biblioteca, como de qualquer organização são mais seguras por meio de um processo formalizado de avaliação. Na web existem vários modelos, neste trabalho apresentamos a indicação de três que podem ser mais bem estudados e adaptados para as necessidades tecnológicas da área.

Referências

Bacic, Nicolas Michel. O software livre como alternativa ao aprisionamento tecnológico imposto pelo software proprietário. Campinas: UNICAMP, 2003.

Balby, Cláudia N. Roteiro de avaliação de sistemas informatizados para bibliotecas e centros de documentação. São Paulo, atualizado 2002. (Apostila disciplina documentação e informática – CBD/ECA/USP). Disponível em: <<http://www.eca.usp.br/prof/fmodesto/disc/DocInf/texto/rotei02.pdf>> Acesso em: 20/03/2007.

Café, Lígia; Santos, Christophe dos; Macedo, Flávia. Proposta de um método para escolha de software de automação de bibliotecas. Ciência da Informação, Brasília, vol.30, no. 2, p.70-79, ago. 2001. Disponível em <<http://dici.ibict.br/archive/00000208/>>

Capgemini. The Capgemini Open Source Maturity Model, Aug./2003. Disponível em

<
http://www.seriouslyopen.org/nuke/html/modules/Downloads/osmm/GB_Expert_Letter_Open_Source_Maturity_Model_1.5.3.pdf> Acesso em: 10/03/2007

Côrte, Adelaide Ramos e et al. Automação de bibliotecas e centros de documentação: o processo de avaliação e seleção de softwares. Ciência da Informação, Brasília, v. 28, n.3, p.241-256, set./dez. 1999. Disponível em <<http://www.ibict.br/cienciadainformacao/viewarticle.php?id=322&layout=abstract>>

Côrte, Adelaide Ramos e et al. Avaliação de softwares para bibliotecas e arquivos: uma visão do cenário nacional. 2. ed., rev. e ampl. São Paulo: Polis, 2002.

Library & Information Technology Association. Open Source Systems Interest Group. Disponível em: <
<http://www.ala.org/ala/lita/litamembership/litaigs/opensourcesystem/opensourcesystems.cfm>> Acesso em: 24/03/2007.

Manifesto sobre Internet da IFLA. Aprovado durante a reunião do Conselho da «68th IFLA General Conference and Council», em 23 de Agosto de 2002, em Glasgow, Escócia. Disponível em: <http://rcbp.iplb.pt/docs/IFLA.pdf> Acesso em: 05/12/2006.

Modesto, Fernando. Software livre: alternativas e desafios para a comunidade bibliotecária. OFAJ: Tecnologia da Informação, abril/2004. Disponível em: http://www.ofaj.com.br/colunas_conteudo.php?cod=192 . Acesso em: 10/02/2007.

Navica. The Open Source Maturity Model (OSMM). Disponível em: <
<http://www.navicasoft.com/pages/osmmoverview.htm>> Acesso em: 15/03/2007.

OpenBRR.org. Modelo de levantamento para Avaliação de preparo para Negócio (*Business Readiness Rating*): uma proposta de padrões abertos para facilitar a avaliação e adoção de soluções de software livre. Trad. Alexandre R. L. Marcondes. © 2005. Disponível em: <<http://www.openbrr.org>> Acesso em: 10.06.2006

Porcel Iturralde, Maria Laura; Rodrigues Mederos, Mabel. Software livre: uma alternativa para las bibliotecas. Acimed, v. 13, n. 6, 2005. Disponível em: http://bvs.sld.cu/revistas/aci/vol13_6_05/aci09605.htm. Acesso em: 05/09/2006.

Ravaglia, Valério. Código livre: a face obscura do software. ECOSPY, ano 1, n. 2, 2004.

Saleh, Amir Mostafa. Adoção de tecnologia: um estudo sobre o uso de software livre nas empresas. Dissertação (mestrado) – Faculdade de Economia, Administração e Contabilidade. São Paulo: Universidade de São Paulo, 2004.

Spike Source. Disponível em:<<http://www.spikesource.com/>> Acesso em: 25/03/2007

VIANA, Michelangelo Mazzardo Marques. Características desejáveis em um sistema de automação de bibliotecas, 30 de Julho de 2001. Disponível em <http://paginas.terra.com.br/educacao/mique/sistema_bibliotecas.html> Acesso em: 20/09/2005.

Repositórios de Software livre:

Codehau: <http://www.codehaus.org/>

Java.net: <http://java.net/>

ObjectWeb: <http://www.objectweb.org/>

Open Source Systems for Libraries: <http://www.oss4lib.org/>

SourceForge: <http://www.sourceforge.net/>

Tigris: <http://www.tigris.org/>

Sobre o autor

José **FERNANDO MODESTO** da Silva

Bacharel em Biblioteconomia e Documentação e Mestre em Biblioteconomia pela Pontifícia Universidade Católica de Campinas – PUCAMP. Especialista em Análise de Sistemas pela Fundação de Ensino de Osasco – FIEO. Doutor em Comunicação pela Escola de Comunicações e Artes da Universidade de São Paulo. Professor-doutor do Departamento de Biblioteconomia e Documentação da ECA/USP. Pesquisador vinculado ao Núcleo de Pesquisa da Produção Científica em “Ciência da Informação” da ECA/USP. No movimento classista, atuou como: Diretor do Sindicato dos Bibliotecários do Estado de São Paulo (1984 – 1985); Coordenador de Cursos da Associação Paulista de Bibliotecários - APB (1986 –1987); Vice-Presidente da APB (1987 – 1989); Presidente do Conselho Regional de Biblioteconomia de São Paulo (1991 –1993); Conselheiro do Conselho Federal de Biblioteconomia – CFB (1994 – 1997); e Presidente do CFB (2000 – 2002). Autor das monografias: “Microinformática em bibliotecas das universidades públicas do Estado de São Paulo: um estudo exploratório” (Campinas, PUCAMP, 1989); “A ambientação da microinformática nos serviços bibliotecários” (Osasco, FIEO, 1997); “Internet – Biblioteca – Comunidade Acadêmica: conhecimentos, usos e impactos; pesquisa com três universidades paulistas (UNESP – UNICAMP – USP)” (São Paulo, USP, 2002); Organizador: “Comunicação & Produção Científica: contexto, indicadores e avaliação” (com: Población, D A; Witter, G P. São Paulo: Angellara, 2006).