

# Software Livre Suportando Ambientes Científicos de Alta Tecnologia

Celso Providelo<sup>1</sup>, Francisco J. Monaco<sup>2</sup>, Ralf Keding<sup>3</sup>

<sup>1</sup>Departamento de Engenharia Elétrica  
Escola de Engenharia de São Carlos - Universidade de São Paulo  
Av. Trabalhador Sancarlense, 400 – 13566-530 São Carlos, SP

<sup>2</sup>Instituto de Ciências Matemáticas e de Computação  
Universidade de São Paulo  
Av. Trabalhador Sancarlense, 400 – 13560-970 São Carlos, SP

<sup>3</sup>Otto-Schott Institut für Glaschemie  
Friedrich-Schiller Universität  
Fraunhoferstr, 6 – 07743 Jena, Deutschland

cprov@sel.eesc.usp.br, monaco@icmc.usp.br, Ralf.Keding@uni-jena.de

**Abstract.** *This article deals with the advantages of using Free Software and its methodologies in the development of high-end scientific instrumentation systems. Highlighted advantages are presented not only in the academic context, but also in the industrial environment. The system OSI-VGA in development in Otto-Schott Institut für Glaschemie in Germany is used as a case study.*

**Resumo.** *Este artigo trata das vantagens alcançadas ao se adotar software livre e suas metodologias no desenvolvimento e implantação de sistemas de instrumentação científica de alta tecnologia. Vantagens destacadas são apresentadas não apenas no contexto acadêmico, mas também em ambientes industriais. O sistema OSI-VGA em desenvolvimento no Otto-Schott Institut für Glaschemie na Alemanha. é discutido como estudo de caso.*

## 1. Introdução

Ao contrário da equivocada primeira impressão que ocorre àqueles que se surpreendem perante o “paradoxal sucesso” do Software Livre [Stallman, 1984] a despeito de seus princípios e métodos não ortodoxos (do ponto de vista das metodologias tradicionais vigentes na indústria proprietária), seus proponentes não creditam seus bons resultados à inteligência peculiar dos membros de sua comunidade, mas ao seu modelo de desenvolvimento aberto e cooperativo, baseado no acesso não privilegiado à informação. Na vanguarda do seu movimento, já se cogita, em diversas instâncias, que a experiência é válida para outras áreas do conhecimento tais como a indústria da música, da imprensa e da eletrônica<sup>1</sup>. Este artigo comenta a influência da disseminação do Software Livre no desenvolvimento de sistemas de Instrumentação Científica e discute seus impactos através de um estudo de caso.

### 1.1. Sistemas de Instrumentação Científica

Os sistemas de instrumentação científica considerados nesse trabalho compreendem a associação de técnicas, ferramentas e profissionais multidisciplinares com o intuito de elaborar novos conceitos em uma área específica do conhecimento. Mais que a utilização de SL na implantação deste

---

<sup>1</sup>Free Music, Free Print e Free Hardware, dentre outras propostas, já são largamente discutidas na Internet.

tipo de sistema, este artigo discute mudanças mais profundas no que diz respeito à concepção e constante adaptação destas aplicações.

## 2. Estudo de caso

Um exemplo de sistemas de instrumentação científica é o *LabScape* [Arnstein L., 2002] desenvolvido na *Universidade de Washington*. Sistemas deste gênero associam ferramentas disponíveis como: base de dados, sistemas automáticos de aquisição de dados, ferramentas computacionais de cálculo, sistemas de diagnóstico assistido por computador, entre outros, para auxiliar o indivíduo em tarefas que requerem alta confiabilidade, tais como: pesquisa científica, manutenção preditiva, controle de processos industriais e agricultura de precisão.

O sistema que apresentamos, o OSI-VGA [Providelo and Panosso, 2001], em desenvolvimento no *Otto-Schott Institut für Glasschemie* na cidade de *Jena*, *Alemanha* é utilizado tanto na análise laboratorial e pesquisa de novos materiais (meio acadêmico) quanto no controle de precisão dos níveis de pureza na linha de produção de vidros planos (meio industrial).

O diagrama da Figura 1 mostra o funcionamento do sistema: requisição de medida (1), gerenciamento de sessão(2), aquisição de dados (3), armazenamento dos dados (4), aviso de encerramento (5)<sup>2</sup>. A arquitetura geral da aplicação é ilustrada na Figura 2.

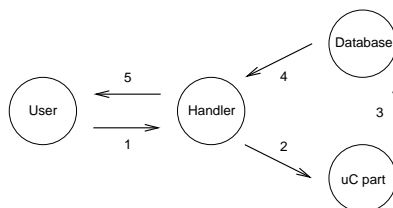


Figura 1: Diagrama do Sistema OSI-VGA

O componente de software aparente ao usuário foi escrito em C++ utilizando a biblioteca gráfica QT2. O desenvolvimento se deu com a ajuda do *Kdevelop* [kde, 2000] um IDE (Integrated Development Environment), o qual facilitou o trabalho de compilação, *debug* e teste. O “handler”, ou manipulador de tarefas, foi escrito em *C-ANSI*, visando melhor desempenho e implementa todas as funções destinadas ao compartilhamento dos recursos utilizados, gerencia sessões e mantém a integridade e segurança do sistema.

A parte microcontrolada foi implementada a fim de realizar as tarefas que necessitam de coerência temporal (*Hard Real-Time* [Tanenbaum, 2001]), aliviando a carga de processamento nos componentes superiores e evitando *paches* no Kernel Linux atual (que não dispõe de capacidades adequadas para a aplicação). A interface neste ponto ocorre por meio de um duto de mensagens via porta serial utilizando a técnica *HR<sup>2</sup>* [Providelo, 2001].

A base de dados utilizada foi PostgreSQL [pgs, 2000] pela funcionalidade e confiabilidade oferecida, e o *front-end* do sistema é mostrado na Figura 3, onde vê-se a GUI que auxilia a visualização dos dados e a elaboração do processo de medida, criando uma interface rápida para o trabalho de pesquisa.

<sup>2</sup>Utiliza-se o protocolo *IDNP* para redes de dispositivos interativos [Monaco, 2002]

<sup>3</sup>*Human-readable Interface*.

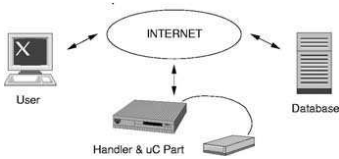


Figura 2: Topologia do Sistema OSI-VGA

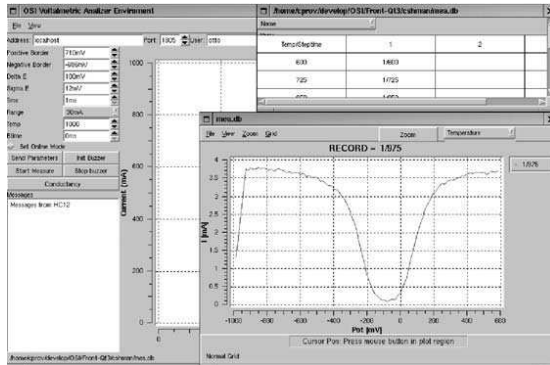


Figura 3: Screenshot do Sistema OSI-VGA

### 3. Sobre a metodologia utilizada

Como comentado na seção 1, o projeto foi concebido segundo os princípios e técnicas consagrados na Comunidade Software Livre, utilizando a infra-estrutura de uma incubadora de projetos (a popular *Sourceforge*), a qual disponibiliza os serviços de forma gratuita e confiável na Internet. Foi criado um repositório, um *bug-tracker*, uma lista de discussão via e-mail, um *site* para a divulgação de notícias e pedidos de ajuda.

O número de participantes não é nada comparado a projetos nitidamente mais populares como *Apache* ou *Mozilla* — assim como *Eric S. Raymond* sugere em *The Cathedral and the Bazaar* [Raymond, 1998] — no entanto o nível de sofisticação real do software gerado é muito alto se comparado outros softwares proprietários disponíveis no mercado. Credita-se tal fato à possibilidade de os usuários, mesmo em número reduzido, terem participação direta no desenvolvimento do sistema. Mesmo aqueles que não são programadores têm acesso à toda informação de que necessitam para entender e opinar no projeto, o que, além de propiciar seu aprendizado mesmo em tópicos alheios ao seu conhecimento específico, constitui orientação valiosa para a equipe de desenvolvedores. É clara a importância de se seguir consistentemente as recomendações que facilitem essa interação, tais como: simplicidade, clareza, boa documentação (no que concerne à programação, em particular, um bom guia é o *GNU Code Standards*, escrito por Richard M. Stallman e disponibilizado pela *Free-Software Foundation* [Stallman, 1984]).

### 4. Lições e Conclusões

Em comum com outros projetos de mesma natureza, este trabalho ressalta o modo como a adoção de ferramentas utilizadas no processo de desenvolvimento de software livre, no âmbito que se refere ao acesso a informação e respaldo ao trabalho colaborativo, propicia o intercâmbio de

informação entre os envolvidos, acelerando o processo de aprendizado e mitigando deficiências entre componentes do grupo. Outro aspecto importante que deve ser levado em consideração é a satisfação de “ensinar aprendendo”, ou seja, o profissional envolvido em um projeto concebido sob os princípios do *Software Livre* associa seus conhecimentos ao conhecimento comum do grupo e o assimila, fazendo com que ele se beneficie do conhecimento dos outros integrantes também.

O que torna o exemplo do sistema OSI-VGA particularmente interessante, todavia, é a forma como sua experiência evidencia a disseminação do SL como desafio à indústria do conhecimento proprietário. Sua influência no projeto em exame não se restringe ao software produzido, mas foi capaz de motivar toda uma equipe de trabalho, incluindo programadores, projetistas de hardware e usuários cientistas a assumirem uma atitude coerente em relação ao sistema completo que, desde a documentação de operação até especificação dos circuitos eletrônicos, está sendo disponibilizado sob a licença GNU GPL.

Tem-se observado vantagens de sistemas deste gênero no ambiente acadêmico-industrial, pois trazem consigo algumas qualidades muito desejáveis não só no software, em seu aspecto técnico, mas no sistema como um todo, qualidades estas provenientes desta nova metodologia de desenvolvimento, não apenas em termos modularidade, facilidades no compartilhamento de recursos e escalabilidade, mas também na flexibilidade e autonomia permitidas pela liberdade de colaborar propiciada pelo modelo SL. Por outro lado, evidenciam-se os conflitos pertinentes ao trabalho científico: sob o ideal de produzir conhecimento e disseminá-lo para o bem comum, o benefício de desfrutar do software livre constitui, em si, um desafio ético à sua utilização na produção de conhecimento proprietário, o que demonstra como o impacto do SL no ambiente científico tende a ir além das meras aplicações computacionais.

## Referências

- (2000). Kdevelop. <http://www.kdevelop.org>.
- (2000). Postgresql. <http://www.postgresql.org>.
- Arnstein L., Borriello G., C. S. H. C. S. J. (2002). Labscape: A smart environment for the cell biology laboratory. *IEEE Pervasive Systems*, vol. 1(no. 3).
- Monaco, F. J. (2002). *Proposta de uma rede para dispositivos telemáticos*. PhD thesis, EESC-USP.
- Providelo, C. (2001). Concepts of human readable interface (hri).
- Providelo, C. and Panosso, A. (2001). Microcontrolled implementation of a voltametric analyzer in glasses. Technical report, Otto-Schott Institut für Glaschemie and SEL-EESC-USP.
- Raymond, E. S. (1998). *The Cathedral and the Bazaar*. O'Reilly.
- Stallman, R. M. (1984). [www.fs.org](http://www.fs.org).
- Tanenbaum, A. S. (2001). *Modern Operating System*. Prentice Hall, second edition edition.