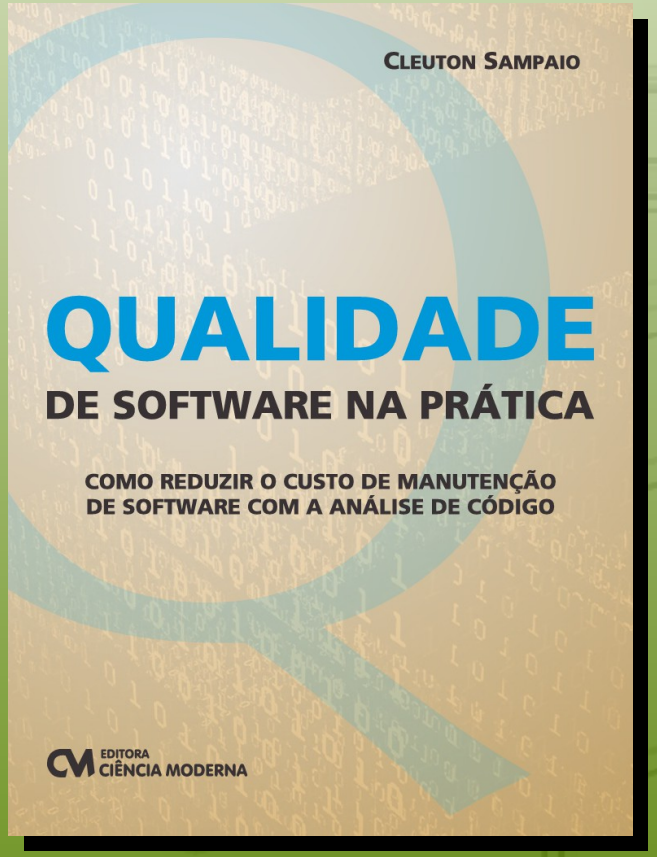
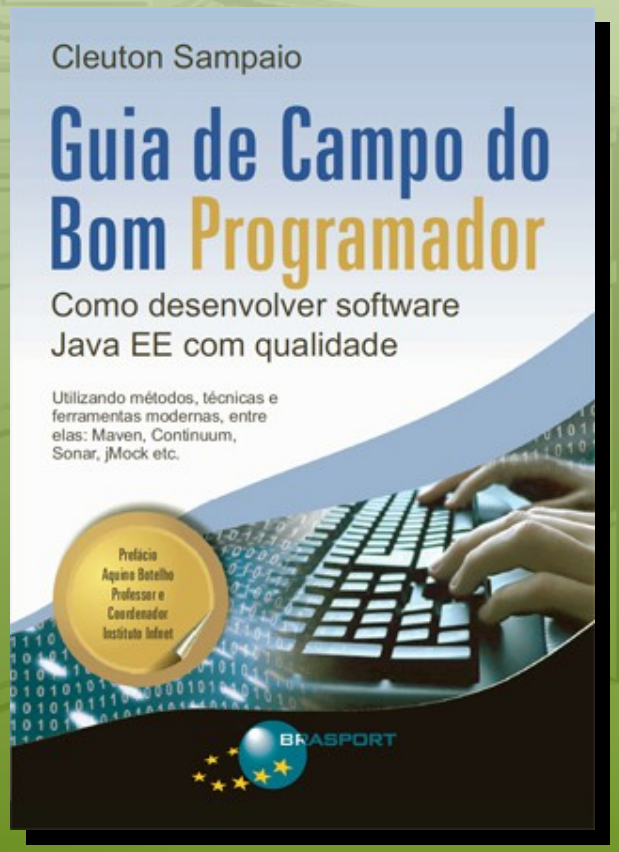
Software livre em ambiente corporativo

Cleuton Sampaio

$ whoami

- Cleuton Sampaio, M.Sc., PMP, SCEA
- Cleuton.sampaio@gmail.com





$ whoami

- Participante e desenvolvedor de soluções Software Livre:

jQana Quality tool
jqana.com
GitHub

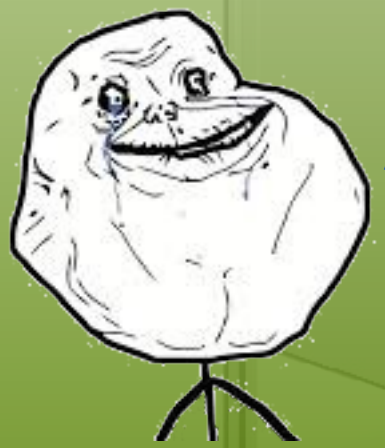
Application Engine
Aamoframework.org

WebXpose Server
Google Code

Hercules
Password Protector
Google Code

Peraí...

Pelo seu envolvimento com software livre, sua opinião deve ser tendenciosa, certo?



Pragmatismo!

- Você deve usar o melhor dos dois mundos: Proprietário e Software Livre;
- Muitos dos meus projetos Software Livre foram criados com computador Mac (Apple) ou com Microsoft Windows;
- Uso Android, assim como uso iPad (Apple);
- Uso: Inkscape, LibreOffice e Gimp, assim como uso: Microsoft Office, AutoDesk Sketchbook e outros;
- Programo em Java™ assim como programo em .NET (Microsoft).

Decisão de usar Software Livre

- Deve ser desapaixonada, baseada em critérios, que podem ser:
  - Políticos;
  - Econômicos (TCO);
  - De Transparência;
  - De Interoperabilidade;
  - De Funcionalidade.

Políticos

- Usar soluções e plataformas Software Livre possibilita atingir um público maior, sem exigências de renda ou de posse de licenças;
- Exemplo: Até poucos anos, o IRPF só podia ser declarado em computadores que rodavam Microsoft Windows. Hoje, pode ser declarado em quase todos os tipos de equipamento.

Econômicos

- Quando a decisão implica em um menor TCO, ou Custo Total de Propriedade. O custo da Licença, geralmente, tem o menor impacto;
- **Exemplo**: Em uma empresa onde todos estão acostumados com MS Windows, a mudança para Linux pode acarretar um TCO inicial maior, que deve ser considerado, além da mudança cultural, mas, e no longo prazo?

Transparência

- Você precisa saber COMO a função está sendo executada, ou como o algoritmo foi implementado, seja por segurança ou por política;
- Exemplo: O sistema operacional da máquina que vai executar o seu "Firewall" ou mesmo o próprio "Firewall";

Interoperabilidade

- Ou Portabilidade, é a capacidade de trocar informações ou executar o software entre plataformas diferentes. Software Livre é, geralmente, baseado em padrões de mercado, o que facilita tudo;
- **Exemplo**: Criar documentos públicos usando um software que usa formato **proprietário** ou usar um que gera formato aberto (ODF)?

Funcionalidade

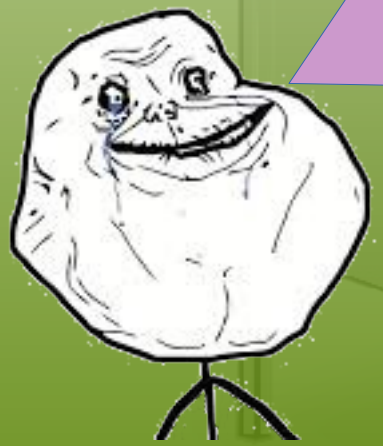
- O software realmente faz o que você necessita? Não adianta tentar usar uma solução Software Livre "meia-bomba", caso ela não lhe atenda;
- **Exemplo**: Para atender às suas necessidades de design, você tem que usar 3 ou 4 Softwares Livres, enquanto existe um único software proprietário que faz tudo o que você precisa.

Não use software Software Livre se...

- É apenas para economizar a grana do custo das licenças dos softwares proprietários;
- É para agradar àquele Diretor que gosta de Software Livre;
- É para ser simpático com o público;
- Você é um usuário "Chupa sangue" ou "Somente venha a nós" (*);
  - (*) Só usa, não contribui nem retribui.

Procure saber!

Pô, legal! Mas software livre é a mesma coisa que "Open Source" ou "Freeware"? Existe diferença?



Free Software Foundation

- Não podemos continuar sem falar no projeto GNU e na Free Software Foundation;
- O projeto GNU criou um sistema operacional totalmente livre e é patrocinado pela FSF;
- A FSF e o projeto GNU foram fundados por Richard Stallman;
- http://www.fsf.org/

Software Livre ou Free Software

- Um programa é software livre se os usuários possuem as quatro liberdades essenciais:
  - A liberdade de executar o programa, para qualquer propósito (liberdade 0).
  - A liberdade de estudar como o programa funciona, e adaptá-lo às suas necessidades (liberdade 1). Para tanto, acesso ao código-fonte é um pré-requisito.
  - A liberdade de redistribuir cópias de modo que você possa ajudar ao próximo (liberdade 2).
  - A liberdade de distribuir cópias de suas versões modificadas a outros (liberdade 3). Desta forma, você pode dar a toda comunidade a chance de beneficiar de suas mudanças. Para tanto, acesso ao código-fonte é um pré-requisito.

Open Source

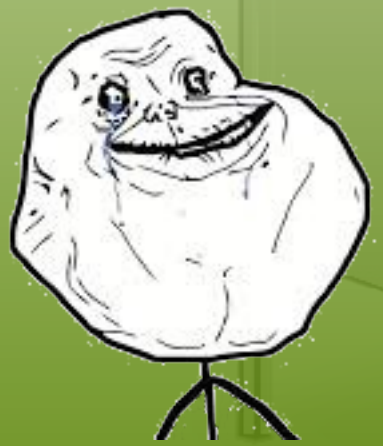
- São programas cujo código fonte é disponibilizado pelo Desenvolvedor. Mas isto não quer dizer que sejam considerados como software livre;
- Existe o termo FOSS: Free and Open Source Software;
- Um programa Open Source pode não ser FOSS;
- Um programa FOSS pode não ser Software Livre;
- Para ser Software Livre, as quatro liberdades básicas devem ser possíveis.

Freeware

- É software disponibilizado para uso, sem a exigência de pagamento de licença de uso, ou com licença opcional;
- Não significa que o código fonte estará disponível, e, se estiver, não significa que você tenha as liberdades de alterá-lo e redistribuí-lo.

E agora?

Pô, me confundiu mais ainda...



Então, lá vai mais complicação...

- Segundo a FSF:
  - Com essas liberdades, os usuários (tanto individualmente quanto coletivamente) controlam o programa e o que ele faz por eles;
  - Quando os usuários não controlam o programa, o programa controla os usuários. O desenvolvedor controla o programa e, por meio dele, controla os usuários. Esse programa não-livre e "proprietário" é, portanto, um instrumento de poder injusto.
  - Assim sendo, "software livre" é uma questão de liberdade, não de preço. Para entender o conceito, pense em "liberdade de expressão", não em "cerveja grátis".

Tudo depende da licença de uso...

- A licença de uso é quem diz se é um Software Livre ou não;
- Software Livre, geralmente, usa uma licença do tipo "CopyLeft":
  - *Copyleft é uma forma de usar a legislação de proteção dos direitos autorais com o objetivo de retirar barreiras à utilização, difusão e modificação de uma obra criativa devido à aplicação clássica das normas de propriedade intelectual, exigindo que as mesmas liberdades sejam preservadas em versões modificadas.*

Domínio Público x CopyLeft

- **Domínio público:** Qualquer um pode fazer o que quiser com o código fonte, inclusive criar uma versão proprietária e não compartilhar com os outros;
- **CopyLeft:** Você é obrigado a compartilhar o código fonte das suas alterações, usando algum tipo de licença CopyLeft;
- A inteção do CopyLeft é evitar que seja criado um software proprietário, a partir de um software de domínio público;
- General Public Licence – **GPL V3** é a licença padrão para softwares livres com **Copyleft**, criada pela **FSF**;

E se eu usar um software GPL?

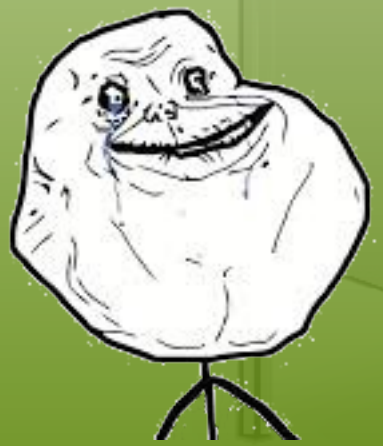
- Se você criar um trabalho derivado, este trabalho também tem que ser distribuído como GPL V3;
- Compilar ou usar componentes licenciados como GPL V3 pode ser considerado como trabalho derivado:
  - "Combinar dois módulos significa conecta-los de forma que eles agora formam um único programa maior. Se qualquer das partes for coberta pela GPL, toda a combinação também precisa ser disponibilizada sob a GPL --se você não puder ou não quiser fazer isso, então você não pode combina-las."
  - http://www.gnu.org/licenses/gpl-faq.html#MereAggregation

Como é a Jurisprudência?

- Existem casos de decisões baseadas no princípio CopyLeft, todas com vitória do CopyLeft, ou acordos caríssimos para os violadores:
  - Jacobsen v. Katzer http://en.wikipedia.org/wiki/Jacobsen_v._Katzer ;
  - BusyBox x Monsoon Multimedia;
  - BusyBox x Verizon Communications;
- Segundo a mídia, a balança está pendendo para o lado CopyLeft, sempre que uma violação é levada à justiça;

Atenção gestores!

Qual é o impacto do licenciamento para minha empresa? Eu posso ser processado? Como? O que devo fazer?

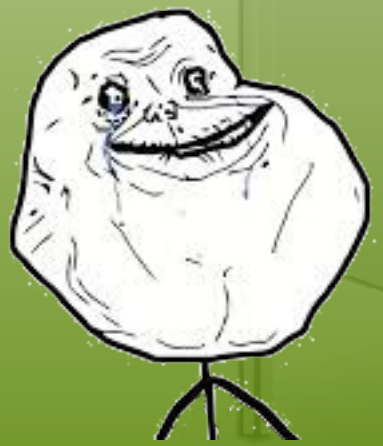


A resposta é: Depende...

- O princípio geral é: Se você usar Software Livre, licenciado com CopyLeft, em parte ou todo, para criar outro software, e pretender distribuí-lo, mesmo gratuitamente, terá que fazer sob algum tipo de licença CopyLeft, disponibilizando o código fonte e os avisos das licenças;
- Se você vai usar Software Livre com CopyLeft, em parte ou todo, para criar software de uso em âmbito interno, ou restrito, não é necessário disponibilizar o código fonte;
- Veja o site: "http://gpl-violations.org/"

Situações cinzentas...

E se eu criar um software para um cliente? Como fica isso?



Se você criou software para um Cliente

- Como regra geral, disponibilize as licenças dos Softwares Livres utilizados e o código fonte de tudo;
- Se o código fonte é propriedade do Cliente, então, o problema é dele, mas é sua obrigação avisá-lo disso;
- Se você retém o código fonte e cobra por manutenções, é uma área cinza;
- Se você distribui o software, mesmo em nome do seu Cliente, é uma área cinza;
- Áreas cinzas podem causar problemas com violações do direito de CopyLeft!

Tipos de licenças mais comuns

- GPL (GNU General Public License): Ela reforça as quatro liberdades, e impede que o código-fonte (ou suas derivações) sejam "apoderados" por terceiros, virando software proprietário. É a que garante realmente as quatro liberdades, porém, restringe o software derivado através do "Copyleft";
- LGPL (GNU Lesser General Public License): Apesar de ser baseada na GPL, permite o uso do software, neste caso bibliotecas de código, com software proprietário, desde que seja mantido o aviso de "copyright". Weak CopyLeft;
- BSD (Berkeley Software Distribution): É utilizada pelo Unix derivado da universidade de Berlkeley, e é muito permissiva, não impondo o princípio de "Copyleft";
- Apache (Apache Software Foundation): É utilizada pelos produtos da ASF, e é permissiva, não impondo o princípio de "Copyleft";
- Mozilla (Licença pública Mozilla): É utilizada por vários produtos Mozilla, como o navegador Firefox. Permite exceções ao "Copyleft" – weak CopyLeft;

Falhas comuns

- Usar texto padrão de Copyright para sua Empresa. Isto não é verdade se você usou Software Livre licenciado;
- Esquecer de incluir a oferta do código fonte;
- Incluir a versão incorreta do código fonte;
- Esquecer de incluir os scripts de compilação, instalação e execução;
- Incluir apenas um link para a licença, ao invés de uma cópia exata da licença;
- Incluir apenas um link para baixar o código, ao invés de oferecer a entrega em meio físico;
- No caso de software oferecido como Download, usar links para licença ou para código fonte em servidores que não são seus.

Prospecção de software livre

- Assim como software proprietário, você deve prospectar com cuidado o software livre:
  - Crie uma lista de critérios e atribua pesos;
  - Avalie os candidatos de acordo com sua lista, e não sua preferência pessoal;
  - Verifique também a solidez do Desenvolvedor: Está ativo? A comunidade está ativa? Qual é a frequência de lançamentos de versões?
  - Verifique a qualidade do software: Tem relatórios de build? Tem métricas disponíveis? Quem mais usa?

Atenção aos desenvolvedores

- Atividade dos desenvolvedores: Verificar os "mailing lists", e as informações do site. Há "roadmap", qual é a frequência de "releases"?
- Como está a atividade de "bug fix"? Verifique se existe um site de gestão de "issues", e verifique a frequência de correção de bugs;

Atenção à Comunidade

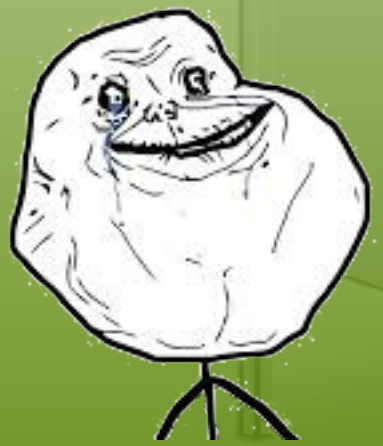
- Todo Software Livre é suportado por uma Comunidade de usuários;
- Verifique como está a atividade dessa comunidade, veja a "mailing list" e os fóruns;
- Verifique quais são as principais queixas, e se são respondidas;

Sustentação e suporte de software livre

- Assim como faz com softwares proprietários, uma empresa deve assumir um compromisso de sustentar o uso e prestar suporte a um software livre;
- Sustentação: Verificar patches, atualizações, e funcionamento;
- Suporte: Tirar dúvidas, resolver problemas.

Não é bem assim...

Então, não tem diferença entre suportar um software livre e um software proprietário?



Suporte de Software Livre

- Software Livre é suportado por sua Comunidade;
- Exige participação e envolvimento;
- Espera-se contribuições e retribuições: "Give back to the community";
- Software proprietário tem contrato de suporte, com atualizações constantes;
- Poucos Softwares Livres oferecem suporte pago;

O que se espera de um usuário de S.L.

- "Give back to the Community" (retribua à Comunidade)?
  - É se engajar nas comunidades;
  - Tirando dúvidas de outros usuários;
  - Fornecendo documentação ou tradução;
  - Criando bug fixes ou ajudando a testar novas versões;
  - Contribuindo com exemplos e ferramentas auxiliares;
  - Contribuindo com software derivado.

Se você pretende ser assim...

- Usuários "venha a nós" (ao "vosso reino": nada!)
  - Só usam o software e em nada contribuem. Só procuram a Comunidade quando dá problema;
- Usuários "Chupa sangue" (além de não contribuir, exigem)
  - Visitam a comunidade, mas só para se beneficiar;
  - Enchem a paciência da comunidade com suas dúvidas e reclamações;
- Nesses casos: NÃO USE SOFTWARE LIVRE!

Distribuição híbrida

- Algumas comunidades e empresas oferecem produtos em duas modalidades:
  - Codebase diferenciado: Uma versão gratuita e aberta, tipo "Community";
  - Suporte premium: Um contrato de suporte pago, no qual você é atendido como em um software proprietário;

Exemplos

- Jboss EAP;
- Alfresco;
- Sonar (SonarQube);
- MySQL;

Na minha opinião (e experiência), se você representa uma Empresa, dificilmente conseguirá usar as versões gratuidas, ou "community" de alguns destes softwares. São muitas restrições...

FOSS x NSOSS: Cuidado com armadilhas

- FOSS: Free and Open Source Software;
- NSOSS: Not So Open Source Software;
- Algumas empresas criam produtos ditos "Open Source", que, na verdade, são apenas chamarizes para vender versões "enterprise" ou mesmo contratos de suporte premium;
- Elas direcionam o usuário de tal forma, que não haverá opção de usar a versão gratuita;

Transformação em NSOSS

- Algumas vezes, o software Open Source é transformado em NSOSS. Os sintomas são:
  - Criação de codebase separada;
  - Oferta de versão ou contrato de suporte "premium";
  - Mudança no suporte ao uso;
  - Substituição de componentes GPL por outros;
  - Mudanças nas regras de uso ou na licença.

E qual é o problema disso?

- Algumas respostas que já vi em Comunidades de suporte:
  - "Essa questão só pode ser respondida pelo nosso suporte premium";
  - "Estamos substituindo o componente XPTO por uma implementação interna, logo, essa função agora está como "deprecated"";
  - "A versão community não inclui mais essa funcionalidade";
  - "O acesso ao código fonte é reservado aos clientes do contrato premium";

Software proprietário é melhor que NSOSS

- Você tem um contrato de suporte, com prazo de atendimento acordado;
- Você tem um "roadmap", com evolução das versões;
- Tem atualizações periódicas.

Resumindo

- A decisão de usar Software Livre, em ambiente corporativo, depende de:
  - Estudo de impacto e de TCO;
  - Seleção cuidadosa de softwares, com análise de utilização e de licenças disponíveis;
  - Engajamento nas comunidades de suporte;
- Se sua empresa não pretende mudar a postura, enquanto usuária de software, então é melhor contratar software proprietário. Sai mais barato;

Obrigado

- Contatos: cleuton.sampaio@gmail.com
- Site: http://www.obomprogramador.com
- Site: http://www.thecodebakers.org
- Site: http://jqana.org
- Veja outras palestras:
  http://www.softwarelivre.gov.br/search?SearchableText=cleuton