

# Desenvolvimento de Sistemas no Âmbito do Governo Padrões e Multiplataforma

Eng. Pablo J. Madril  
pmadril@uol.com.br

# Quem somos?

- **GU-CORBA** : Capítulo Sul-Americano do Grupo Vertical de Saúde da OMG
- **SouJava** : Sociedade de Usuários Java, 5.500 desenvolvedores e usuários Java
- **SBIS** : Sociedade Brasileira de Informática em Saúde
- **GU-Software Livre** : Grupo de usuários de Software Livre

Manifesto

[www.soujava.org.br/manifesto](http://www.soujava.org.br/manifesto)



# Motivação : o Governo desenvolvendo sistemas

- Necessidade de padronização das informações que trafegam entre governo e a sociedade
- Diversidade de plataformas de Hardware e Software atualmente usadas na área governamental
- Falta de critérios que orientem o desenvolvimento de software pelo e para o governo

# Necessidades Básicas

- Independência de fornecedores e tecnologias
- Longevidade das informações armazenadas
- Segurança
- Igualdade de oportunidades para desenvolvedores nacionais e estrangeiros

# Quais são as Soluções?

- Uso e definição de Padrões Abertos
- Desenvolvimento Multiplataforma

# Porque só Código Livre não resolve ?

- O governo já é proprietário do código que desenvolve (ele mesmo ou através de terceiros por licitações)
- Sem orientação, este desenvolvimento impõe uma determinada plataforma :
  - Software: Sistema Operacional, Banco de Dados, etc
  - Hardware: Máquinas ou tecnologias específicas
- Código Livre resolve apenas parte dos problemas...

# Porque Padrões Abertos ?

- O que são padrões abertos?
  - Representam conhecimento e o melhor consenso a respeito de um determinado assunto.
  - Precisam estar disponíveis publicamente para não discriminar ninguém.
    - Exemplos : RTFs do W3C, padrões da OMG (CORBA)
    - Servem, na falta de outros (são pagos): ABNT, ISO, ANSI
- Porque usar?
  - Regulamentam as interações entre sistemas
  - Separam definição de implementação
- Vantagens
  - Permitem a independência de fornecedores
- Desvantagens
  - Não aproveitam características específicas de alguns produtos

# Como definir Padrões Abertos

- Precisam ter uma organização e um mecanismo de definição que permita aos interessados participar da definição do padrão
- O resultado do consenso (padrões) precisa estar disponível publicamente
- Precisam ter pelo menos uma implementação concreta (produto ou código)
- Precisam ter implementações em várias plataformas
- Exemplos :
  - [www.w3c.org](http://www.w3c.org) (HTML, XML, SOAP, etc)
  - [www.omg.org](http://www.omg.org) (UML, CORBA, etc)
  - [www.jcp.org](http://www.jcp.org) (Java, J2EE, J2SE, J2ME, etc)
  - [www.ecma.ch](http://www.ecma.ch) (ECMAScript, CLI, etc)

# Bons exemplos

- CNS: Projeto do Cartão Nacional de Saúde
  - Padrão XML para troca de informações entre o governo e as prestadoras de serviços
  - Qualquer um pode implementar da forma que quiser e trocar informações sem problemas com o governo
- Padrão XML: Público – Disponível em <http://www.w3.org/XML/>
- Padrão de troca de informações: Público – Disponível em <http://www.datasus.gov.br/dtd/>

# Maus Exemplos ou Lições da reserva de mercado

- HosPub - desenvolvido durante a reserva de mercado
  - Roda em Linux
  - Linguagem proprietária que pre-compila para C
  - Utiliza um Banco de Dados nacional
    - Proprietário e pago
    - Não padronizado: não permite exportação de dados sem perda de informações
      - Não é SQL puro
      - Tem características próprias que não podem ser exportadas
  - Criou um **Monopólio Nacional** tão ruim ou pior que os estrangeiros

# Porque Multiplataforma?

- O governo já possui várias plataformas de hardware e software diferentes
- Independência de fornecedores
- As tecnologias avançam muito rápido e o hardware se torna obsoleto: independência no tempo
- Não pode limitar as escolhas de empresas ou do cidadão que irá utilizar o software

# Como definir Código Multiplataforma

- Não adianta rodar só em duas plataformas
  - Levar em conta: diferentes SOs e hardware
- Tem que seguir padrões abertos
  - Ex. : Posix, Apache
- Pode ser desenvolvido em qualquer linguagem ou ferramenta
  - obrigatoriedade de suportar várias plataformas
- Software multiplataforma é uma realidade atual, com milhões de exemplos e muitas tecnologias

# Bons exemplos

- TSE : Tribunal Superior Eleitoral – Eleições 2002 – Programa Divinet
  - Permitia acompanhar a apuração em tempo real
  - A versão 2000 só funcionava em uma plataforma
  - A versão 2002 tem também uma versão Java para operar em multiplataforma (MS Windows, Linux, MacOS X, etc) e em varios idiomas (port,ing, esp)
    - <http://www.tse.gov.br/divnet2002/index.htm>

# Maus exemplos

- O CNPq exige que a documentação seja enviada em um formato proprietário
  - [http://www.cnpq.br/plataformalattes/formproposta\\_utilizacao.htm](http://www.cnpq.br/plataformalattes/formproposta_utilizacao.htm)
- Programas da Receita Federal só funcionam em um único Sistema Operacional
  - <http://www.receita.fazenda.gov.br/Pessoafisica/receitanet/Informacoes.htm>
  - <http://www.receita.fazenda.gov.br/Pagamentos/darf/sicalcOrienta.htm>
  - <http://www.receita.fazenda.gov.br/Pagamentos/darf/sicalcInsComp.htm>
- E quem não quer usar, como fica? Excluído?
  - Em alguns casos, o uso do software é obrigatório...

# pseudo-Padrão e pseudo-Multiplataforma

- :-) A Microsoft padronizou a base da tecnologia .NET (**Common Language Infrastructure CLI**) por pressão do mercado

<http://www.ecma.ch/ecma1/STAND/ecma-335.htm>

- **Infelizmente**, só um sub-conjunto (<10%) foi padronizado
- :-) A Microsoft anunciou que o **Common Language Infrastructure (CLI)** roda em varias plataformas
  - **Infelizmente é só um subset** e funciona apenas em Windows, FreeBSD e MacOSX (que é FreeBSD)

# Prós e Contras

- Ter código que só roda em um Sistema Operacional ou em uma plataforma de hardware é ruim, qualquer que seja o sistema ou plataforma
- Nem código livre, nem o código proprietário podem ser excluídos de participar do processo comum da sociedade
- Hoje já existem fabricantes que fornecem Linux pré-instalado nas máquinas
  - Não existem mais padrões de fato
  - Os usuários destas máquinas não podem ser obrigados a comprar outro sistema operacional para interagir com o governo

# Conclusões

- O governo tem que parar de definir tecnologias e passar a definir, estabelecer e exigir padrões abertos
- Deve ser preservada a liberdade de escolha do cidadão como forma de elevar e proteger a soberania nacional

# Perguntas e Respostas

- Muito Obrigado pela atenção
  - [www.soujava.org.br](http://www.soujava.org.br)
  - [www.corba.org.br](http://www.corba.org.br)
  - [www.sbis.org.br](http://www.sbis.org.br)
  - [www.freesoftware.org.br](http://www.freesoftware.org.br)