Software Livre e de Código Aberto

Fernando Castor

Centro de Informática – Universidade Federal de Pernambuco

O que essas organizações têm em comum?

1. O que é Software Livre?
2. O que é Software de Código Aberto?
3. Quais são as diferenças e semelhanças entre os dois?
4. Empresas estão interessadas em FLOSS?
5. Como você pode se beneficiar?

Software Livre

O que é Software Livre?

Ideologicamente

- Software cujos usuários têm liberdade para
  - executar
  - copiar
  - distribuir
  - estudar
  - modificar
  - melhorar
- Usuários controlam os programas e não o contrário
- **Premissa básica**: liberdade é uma coisa boa para indivíduos e sociedade
  - **Movimento Social**

O que é Software Livre?

Estritamente falando

Um programa é **livre** se seus usuários têm as 4 liberdades essenciais:

0. Liberdade de executar o programa, para qualquer fim.
1. Liberdade para estudar o funcionamento do programa e modificá-lo para fazer o que você desejar.
2. Liberdade para redistribuir cópias.
   - Assim, você ajuda seu vizinho.
3. Liberdade para distribuir cópias de suas versões modificadas.
   - Assim, a comunidade toda se beneficia com suas modificações.

Acesso ao código fonte é precondição para as liberdades 1 e 3.

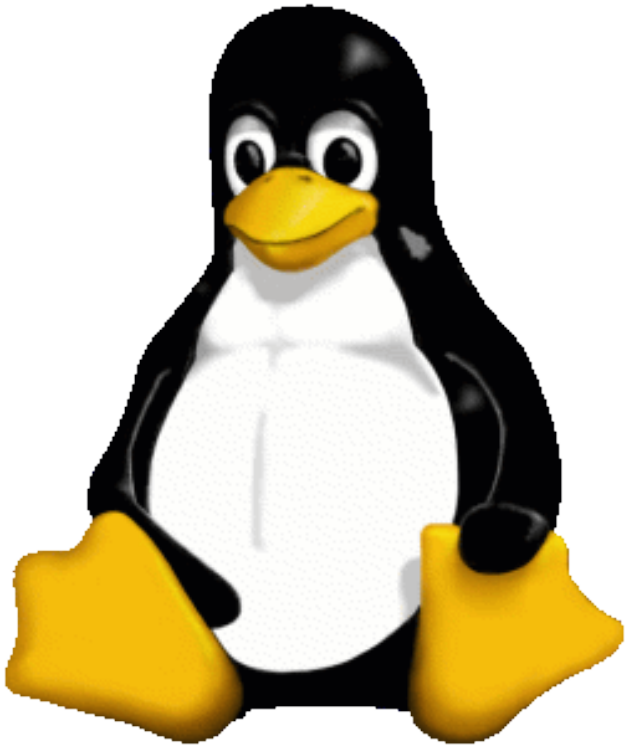
O que é Software Livre?

Legalmente

- Software **distribuído** de acordo com uma **licença** que respeita as 4 liberdades

Este programa é software Livre?

Este programa é software Livre?



Este programa é software Livre?

(considerando apenas a versão gratuita)

Software de Código Aberto

O que é Software de Código Aberto?

~~Ideologicamente~~

O foco do movimento Open Source **não é ideológico**.

Em poucas palavras...

- "Campanha de Marketing" em prol do Software Livre
  - Com foco em vantagens práticas
  - Sem viés ideológico
  - Para chamar a atenção de grandes corporações

Por que "Código Aberto"?

- A palavra "free" em *Free Software* é problemática
  - *"Free as in freedom, not free beer"*

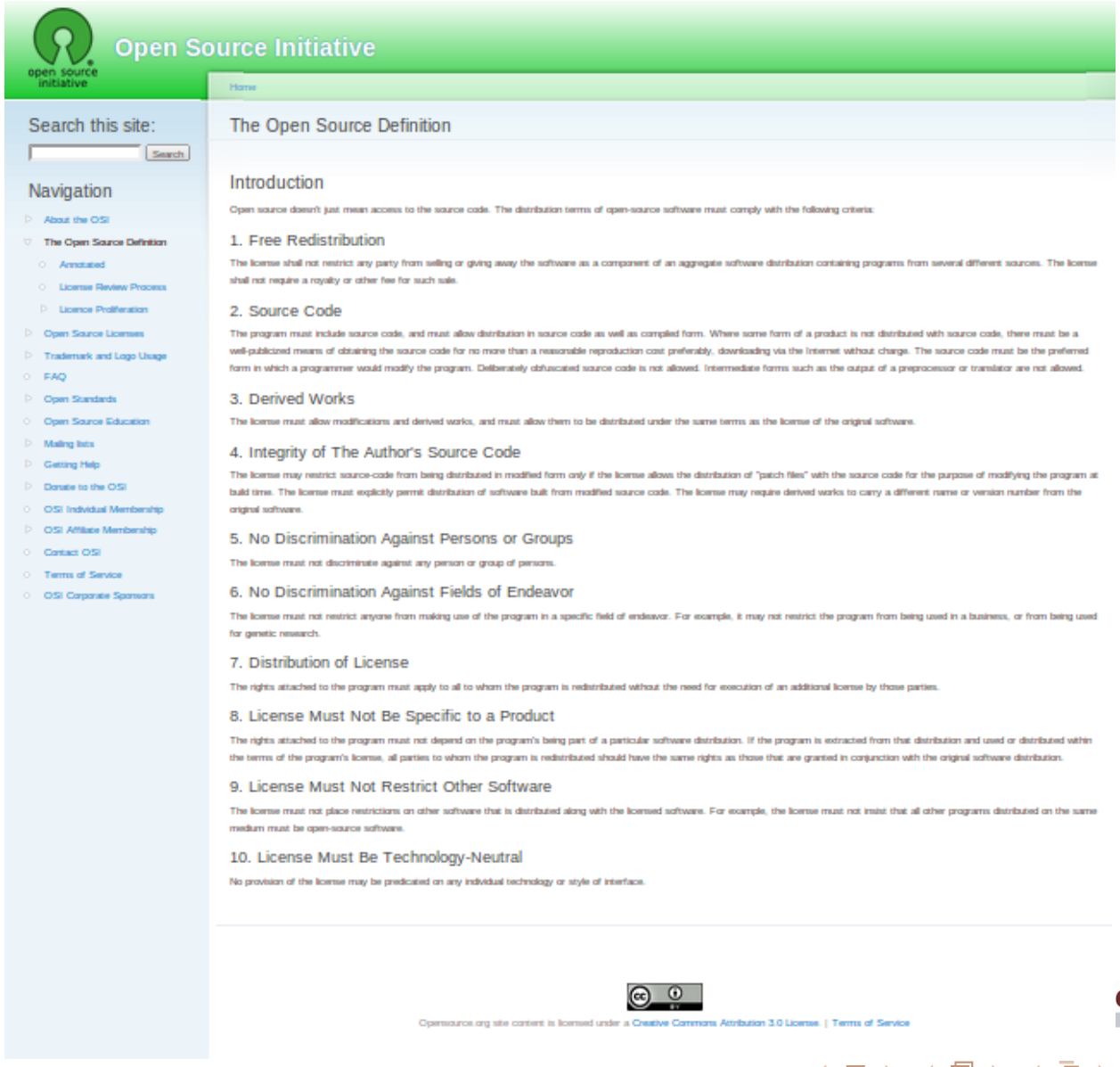
- Com "Código Aberto" a ênfase fica em algo mais palpável, o código

Open source doesn't just mean access to the source code

Práticas

1. Desenvolvimento distribuído
2. Depuração em massa, com usuários envolvidos no desenvolvimento
3. Software disponível cedo e frequentemente
4. Meritocracia
5. Sem uma hierarquia rígida e sem coerção
6. Abertura do código fonte é uma pré-condição

Software de Código Aberto também tem uma definição



Open Source Initiative

open source initiative

Home

Search this site:

Search

Navigation

- About the OSI
- The Open Source Definition
  - Annotated
  - License Review Process
  - Licence Proliferation
- Open Source Licenses
- Trademark and Logo Usage
- FAQ
- Open Standards
- Open Source Education
- Mailing lists
- Getting Help
- Donate to the OSI
- OSI Individual Membership
- OSI Affiliate Membership
- Contact OSI
- Terms of Service
- OSI Corporate Sponsors

The Open Source Definition

Introduction

Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with the following criteria:

1. Free Redistribution

The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

2. Source Code

The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost preferably, downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.

3. Derived Works

The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.

4. Integrity of The Author's Source Code

The license may restrict source-code from being distributed in modified form *only* if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.

5. No Discrimination Against Persons or Groups

The license must not discriminate against any person or group of persons.

6. No Discrimination Against Fields of Endeavor

The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.

7. Distribution of License

The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.

8. License Must Not Be Specific to a Product

The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.

9. License Must Not Restrict Other Software

The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.

10. License Must Be Technology-Neutral

No provision of the license may be predicated on any individual technology or style of interface.

Opensource.org site content is licensed under a Creative Commons Attribution 3.0 License. | Terms of Service

ca · E

ines

SPG SOFTWARE·PRODUCTIVITY·GROUP

“Open Source” é mais popular que “Free Software”

Diferenças e Semelhanças entre FS e OSS

Em pouquíssimas palavras...

A democracia é um direito moral

vs

A democracia é o sistema mais eficiente que se conhece

Pragmaticamente

- Práticas **indistinguíveis**
  - Principalmente após “The Cathedral and the Bazaar”
- Amplo espectro de pontos de vista
- Dois meios de atingir o mesmo fim

Pragmaticamente

- Práticas **indistinguíveis**
  - Principalmente após “The Cathedral and the Bazaar”
- Amplo espectro de pontos de vista
- Dois meios de atingir o mesmo fim

Aspectos Econômicos de Software Livre e de Código Aberto

Eu posso vender software livre? E de código aberto?

Resposta curta: **SIM!**

Na prática

- Não faz sentido
- E cobrar por instalações em máquinas, pode?
- Restringir uso comercial?

Economia de dom?

1. Boa explicação para os primórdios do OSS
   - Em particular, justifica a ação de voluntários

Economia de dom?

1. Boa explicação para os primórdios do OSS
   - Em particular, justifica a ação de voluntários
2. Não explica tão bem o interesse de corporações
   - Economia de escassez, teoricamente
   - Por que gastar dinheiro em algo disponível gratuitamente?
   - Por que alguém abriria o código do seu produto?

Um *potlach* dos Kwakwaka'wakw

Free as in free beer?

- A IBM gastou, entre 1995 e 2005, US$ 1bi no Linux
  - Para fins de desenvolvimento e promoção

Free as in free beer?

- A IBM gastou, entre 1995 e 2005, US$ 1bi no Linux
  - Para fins de desenvolvimento e promoção
- E, três anos atrás, anunciou o seguinte:

IBM Industries & solutions Services Products Support & downloads My IBM

News room
News releases
Press kits
Image gallery
Biographies
Background
News room feeds
Global news rooms
News room search
Media contacts

Related links
- IT Analyst support center
- Investor relations

News room > News releases >

IBM Commits $1 Billion to Fuel Linux and Open Source Innovation on Power Systems

Select a topic or year

News release | Contact(s) information
Related XML feeds | Related resources

NEW ORLEANS, LA - 17 Sep 2013: At LinuxCon 2013 today, IBM (NYSE: IBM) announced plans to invest $1 billion (USD) in new Linux and open source technologies for IBM's Power Systems servers. The investment aims to help clients capitalize on big data and cloud computing with modern systems built to handle the new wave of applications coming to the data center in the post-PC era.

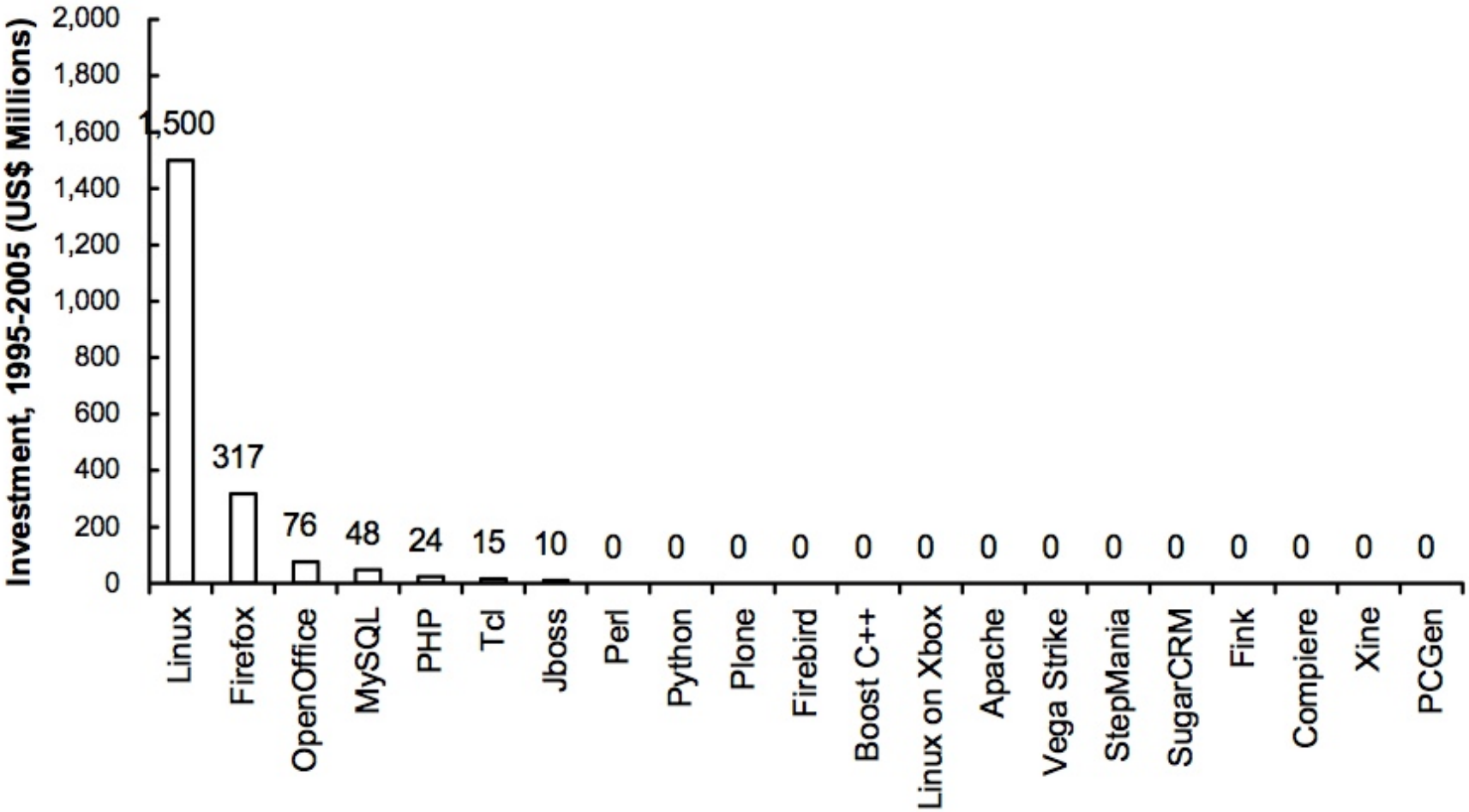
Free as in free beer?

- A Oracle (graças a **aquisições**) era dona, em 2010, de
  - MySQL
  - Java
  - Berkeley DB
  - OpenOffice
  - VirtualBox

Free as in free beer?

- A Oracle (graças a **aquisições**) era dona, em 2010, de
  - MySQL
  - Java
  - Berkeley DB
  - OpenOffice
  - VirtualBox
- O Google desenvolveu um dos melhores Web Browsers
  - E **abriu ele**, via projeto Chromium
  - Incluindo o **engenho V8** para JavaScript

Investimento em projetos FLOSS, por projeto

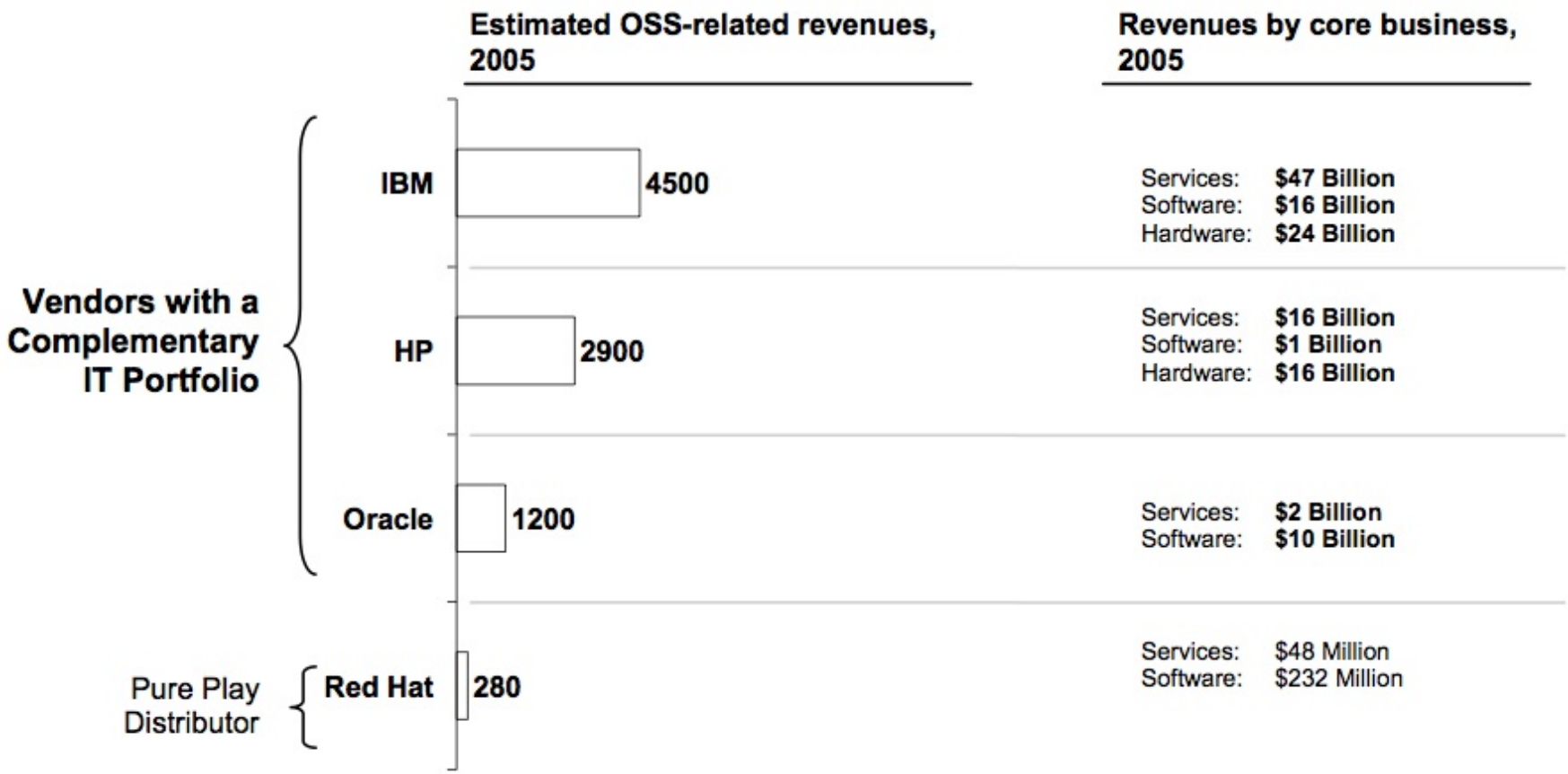


Receita relacionada a FLOSS vs. receita por área

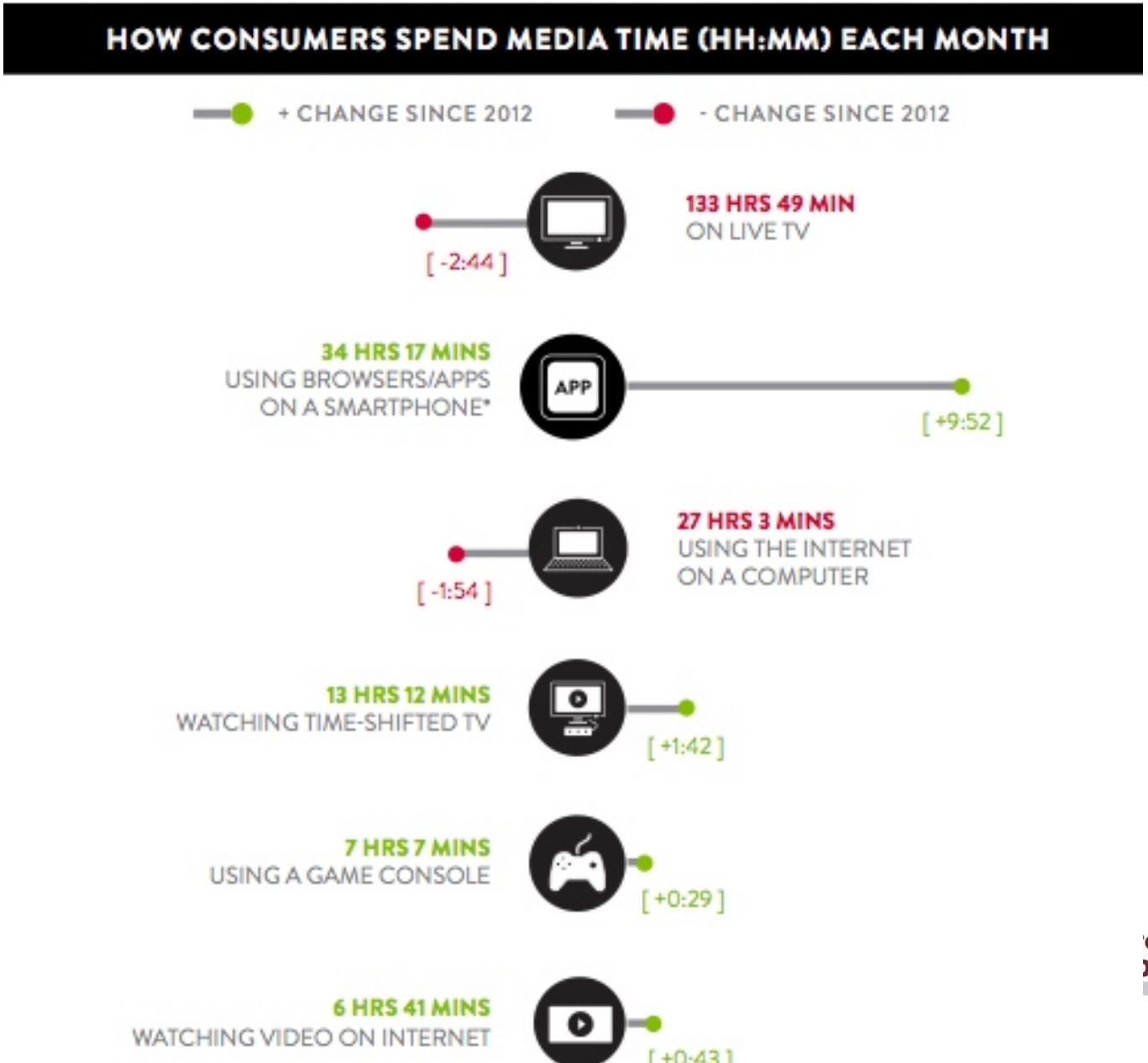
Corporações não tem fins humanitários...

Receita relacionada a FLOSS vs. receita por área

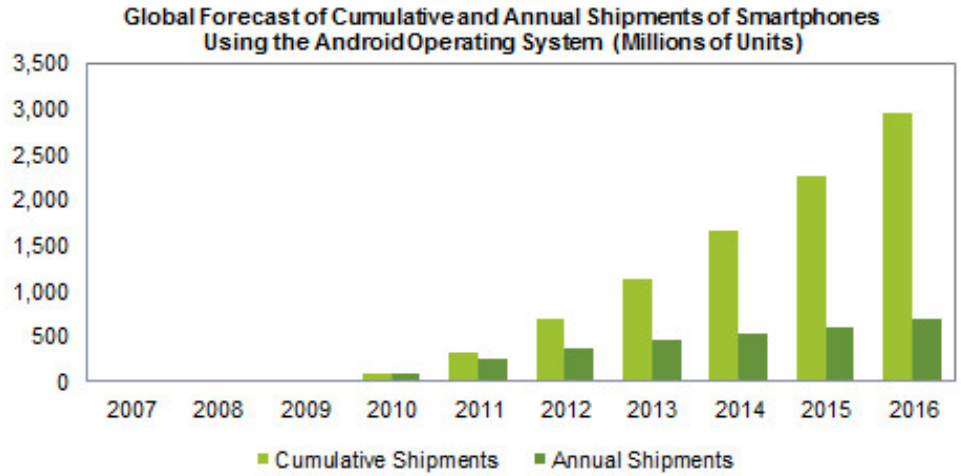
Corporações não tem fins humanitários...



Como consumidores acessam a Internet

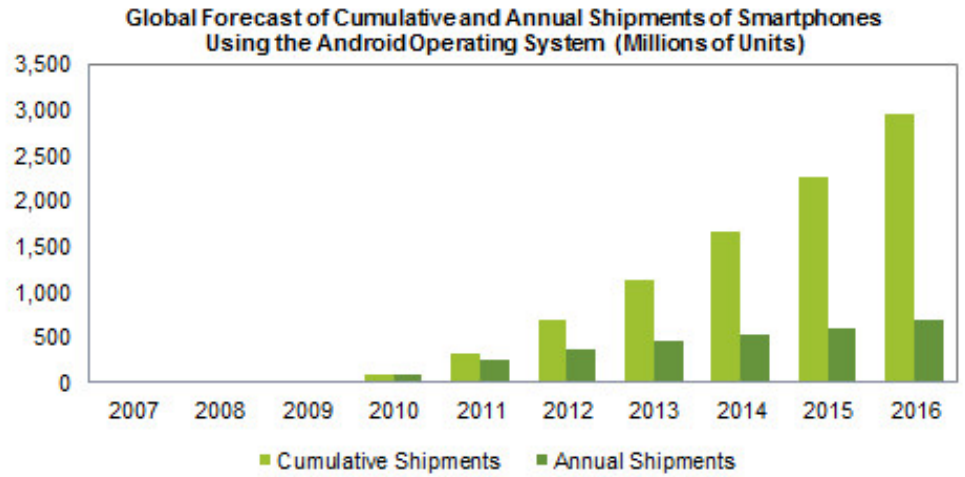


Smartphones com o sistema Android



Source: IHS iSuppli/Screen Digest Research September 2012

Smartphones com o sistema Android



Source: IHS iSuppli/Screen Digest Research September 2012

Cada um desses smartphones...

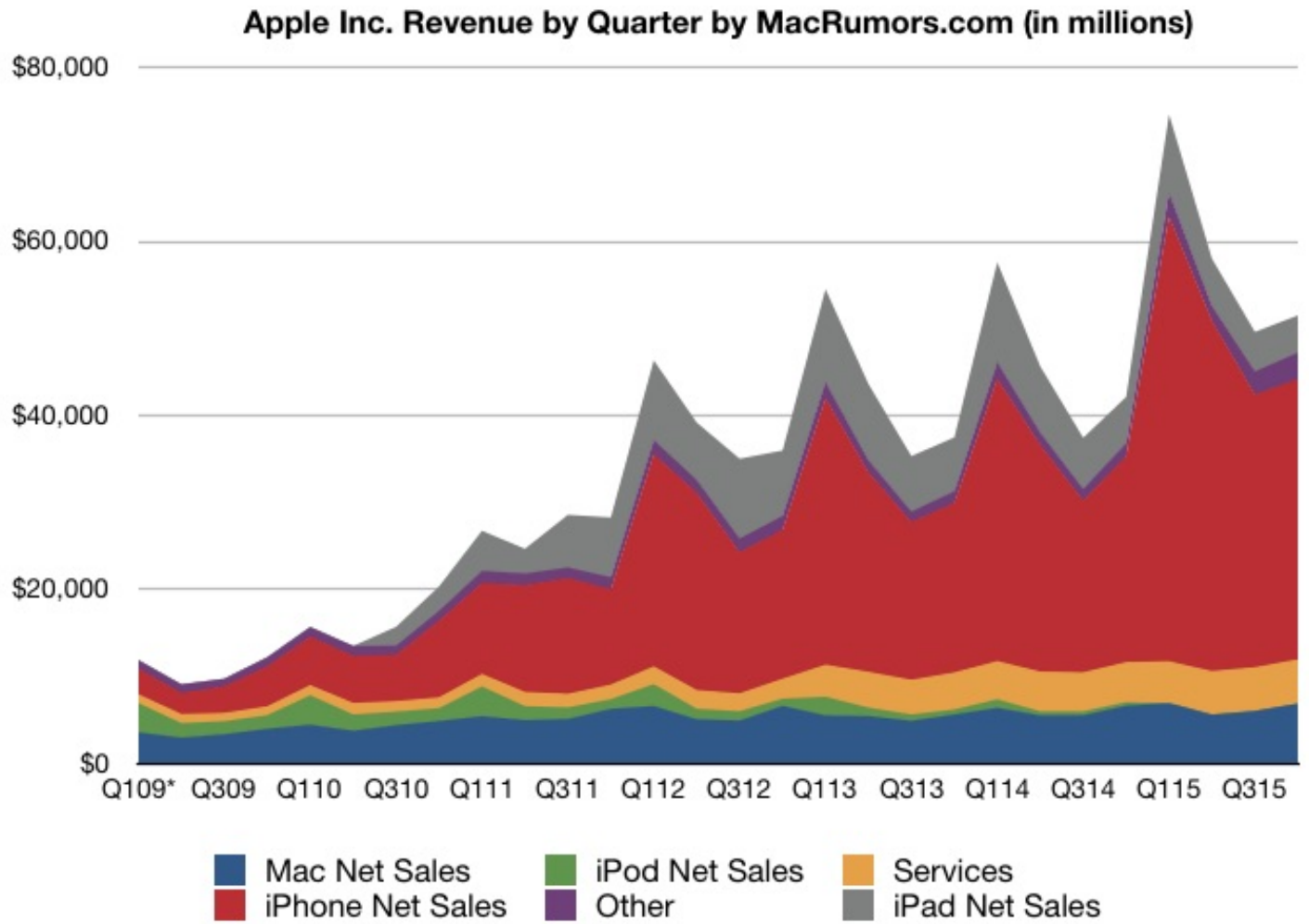
- é um ponto de entrada para pessoas conectarem-se à Internet;
- funciona bem com as ferramentas do Google;
- e coleta informações que o Google pode usar!

Missão do Google: *... to organize the world's information and make it universally accessible and useful.*

Mais recentemente...

- 5+ anos de desenvolvimento (primeiro *commit* em 2010!)
- *open source um ano e meio depois de seu lançamento*

Fontes de receita da Apple



Dois pontos de vista complementares

Software pode ou não ser um diferencial de negócios

1. Software livre não é a principal fonte de receita para IBM, HP, Google ou Apple
2. Mas **ajuda a dividir custos e riscos** para produtos que não são um diferencial de negócios

Dois pontos de vista complementares

Software pode ou não ser um diferencial de negócios

1. Software livre não é a principal fonte de receita para IBM, HP, Google ou Apple
2. Mas **ajuda a dividir custos e riscos** para produtos que não são um diferencial de negócios
3. Se software não é a diferenciação, é um centro de **custo**, não de lucros

Tecnologia meio vs. tecnologia fim

1. Software pode ser uma tecnologia "possibilitadora"
2. Desenvolvimento de software proprietário é uma minoria
   - Menos de 10% das vagas para desenvolvimento de software
   - Mais de 70% estão em empresas que são "usuárias de TI"

Se tiverem que aprender apenas uma coisa, aprendam isso

Vale a pena abrir o código para

1. software que não é um diferencial de negócios
2. software que é uma tecnologia possibilitadora para vários negócios

Se tiverem que aprender apenas uma coisa, aprendam isso

Vale a pena abrir o código para

1. software que não é um diferencial de negócios
2. software que é uma tecnologia possibilitadora para vários negócios

Porque os custos e o risco do desenvolvimento são divididos

Se tiverem que aprender apenas uma coisa, aprendam isso

Vale a pena abrir o código para

1. software que não é um diferencial de negócios
2. software que é uma tecnologia possibilitadora para vários negócios

Porque os custos e o risco do desenvolvimento são divididos

Software Livre como parte de um modelo de negócios "puro" não é comum

Como Você Pode se Beneficiar?

Software livre é grátis

Software livre é grátis?

Pode aumentar sua empregabilidade!

Education

Experience

- *September 2009 - December 2009.* Travelled from Japan to England without flying.
- *December 2008 - September 2009.* Mathematical Systems, Tokyo: statistical software consultancy. Member of Knowledge Engineering team doing custom development in Common Lisp for NTT.
  - Monitoring system for large numbers (100k) of routers, with Holt-Winters' seasonal exponential time series forecasting to detect aberrations.
  - Visualisation of large (cyclic) directed graphs by condensing nodes while attempting to preserve the graph structure.
  - Efficient storage and retrieval of routing tables.
  - Portable persistent memory-mapped object database integrating into Lisp object system with the Meta-Object Protocol (MOP), including a precise compacting garbage collector and atomic transactions (manardb).
- *January 2006 - November 2008.* Travelled in the Middle East, South East Asia, China, North Korea, India, and Japan.
- *October 2004 - December 2005.* Alphamosaic (bought by Broadcom), Cambridge: semiconductor company specialising in low-power video. Engineer on the software research team.
  - Embedded C software development: system-wide and video codec performance optimization, dynamic DMA resource management infrastructure, lightweight reliable mutual exclusion without malloc, H.264 video decode (dramatically increased performance of an already heavily optimized system, directly benefiting the video iPod project), virtual filesystem layer, device drivers.
  - Algorithm evaluation for hardware design: image sensor pipeline (image processing in Matlab), and simulated performance of different i-cache and d-cache sizes and associativities.
  - On-site customer support at Samsung on DMB (mobile TV) phone project. Starting with a dysfunctional project, rebuilt trust with Samsung.
  - High-level ASIC design and software emulation of components for next generation platform (JPEG acceleration).
- *Winter 2003.* Volunteer at NGO wifi ISP in Mozambique.
- *Easter 2003.* Hospital in Bethlehem. Volunteer IT system administrator. Continued regularly until 2007.
- *2002-2004.* Trinity College Students' Union.
- *2001-2002.* Cambridge University Computing Society. Chairman.
- *Summer 2000.* Linux Information Systems AG, Munich: productivity software start-up. Intern.
- *Spring 2000.* Linux kernel. Contributor. USB driver for a scanner; accepted and integrated into the Linux kernel (drivers/usb/microtek.c). Userspace program to assist reverse engineering (usb-robot).
- *1998.* Unreleased game framework in C++ with dynamic code generation.

Recent hobby projects

- teepeedee2, a high-performance web application platform in Common Lisp, directly calling Linux system calls. Benchmarked serving more than 10k dynamically generated pages per second on one processor core.
- dysfunkycom, ranked seventh out of about three hundred teams in the ICFP 2009 programming competition.

Pode aumentar sua empregabilidade!

Education

Experience

- *September 2009 - December 2009.* Travelled from Japan to England without flying.
- *December 2008 - September 2009.* Mathematical Systems, Tokyo: statistical software consultancy. Member of Knowledge Engineering team doing custom development in Common Lisp for NTT.
  - Monitoring system for large numbers (100k) of routers, with Holt-Winters' seasonal exponential time series forecasting to detect aberrations.
  - Visualisation of large (cyclic) directed graphs by condensing nodes while attempting to preserve the graph structure.
  - Efficient storage and retrieval of routing tables.
  - Portable persistent memory-mapped object database integrating into Lisp object system with the Meta-Object Protocol (MOP), including a precise compacting garbage collector and atomic transactions (manardb).
- *January 2006 - November 2008.* Travelled in the Middle East, South East Asia, China, North Korea, India, and Japan.
- *October 2004 - December 2005.* Alphamosaic (bought by Broadcom), Cambridge: semiconductor company specialising in low-power video. Engineer on the software research team.
  - Embedded C software development: system-wide and video codec performance optimization, dynamic DMA resource management infrastructure, lightweight reliable mutual exclusion without malloc, H.264 video decode (dramatically increased performance of an already heavily optimized system, directly benefiting the video iPod project), virtual filesystem layer, device drivers.
  - Algorithm evaluation for hardware design: image sensor pipeline (image processing in Matlab), and simulated performance of different i-cache and d-cache sizes and associativities.
  - On-site customer support at Samsung on DMB (mobile TV) phone project. Starting with a dysfunctional project, rebuilt trust with Samsung.
  - High-level ASIC design and software emulation of components for next generation platform (JPEG acceleration).
- *Winter 2003.* Volunteer at NGO wifi ISP in Mozambique.
- *Easter 2003.* Hospital in Bethlehem. Volunteer IT system administrator. Continued regularly until 2007.
- *2002-2004.* Trinity College Students' Union.
- *2001-2002.* Cambridge University Computing Society. Chairman.
- *Summer 2000.* Linux Information Systems AG, Munich: productivity software start-up. Intern.
- *Spring 2000.* Linux kernel. Contributor. USB driver for a scanner; accepted and integrated into the Linux kernel (drivers/usb/microtek.c). Userspace program to assist reverse engineering (usb-robot).
- *1998.* Unreleased game framework in C++ with dynamic code generation.

Recent hobby projects

- teepeedee2, a high-performance web application platform in Common Lisp, directly calling Linux system calls. Benchmarked serving more than 10k dynamically generated pages per second on one processor core.
- dysfunkycom, ranked seventh out of about three hundred teams in the ICFP 2009 programming competition.

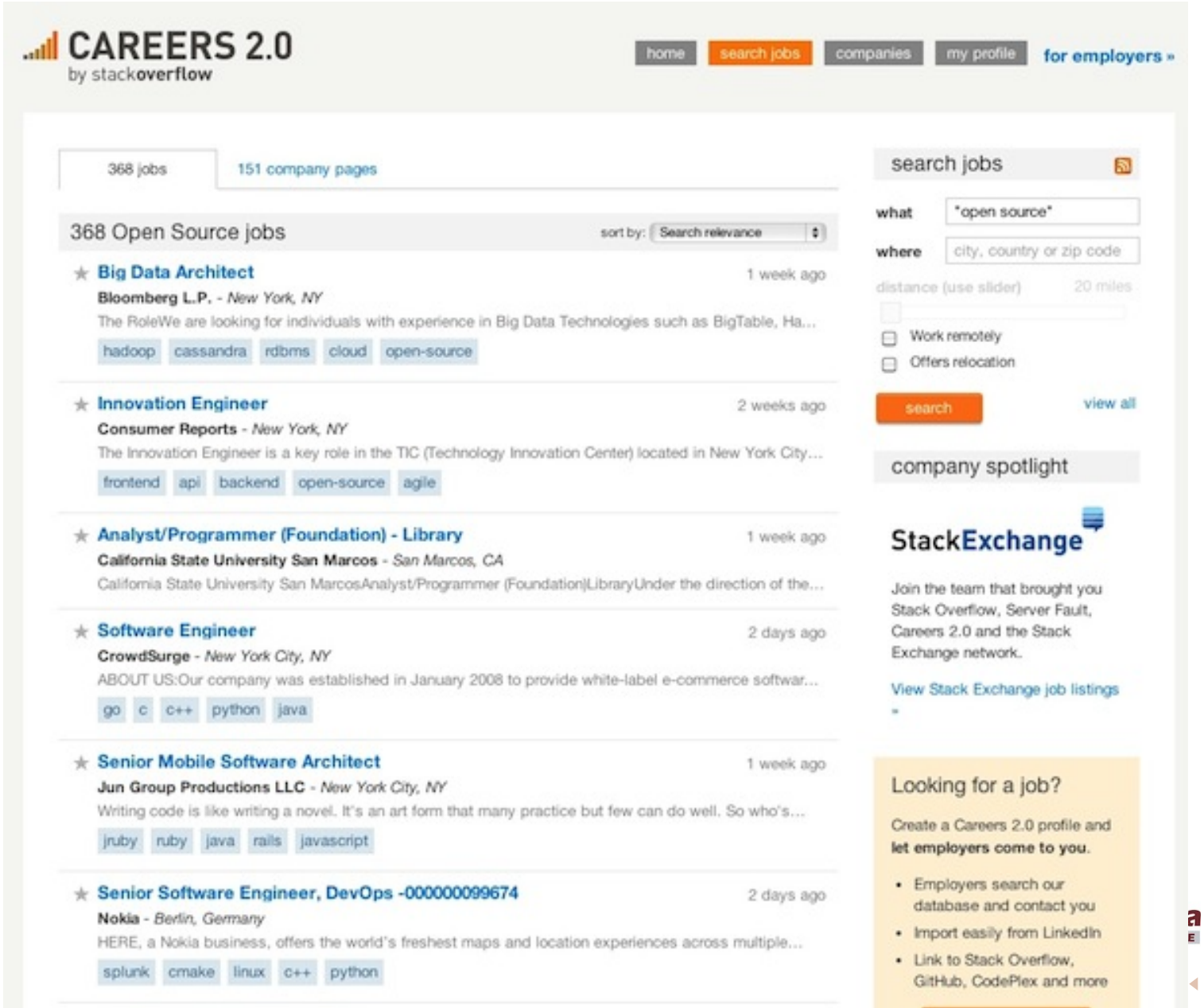
Centro de Informática U·F·P·E

ines

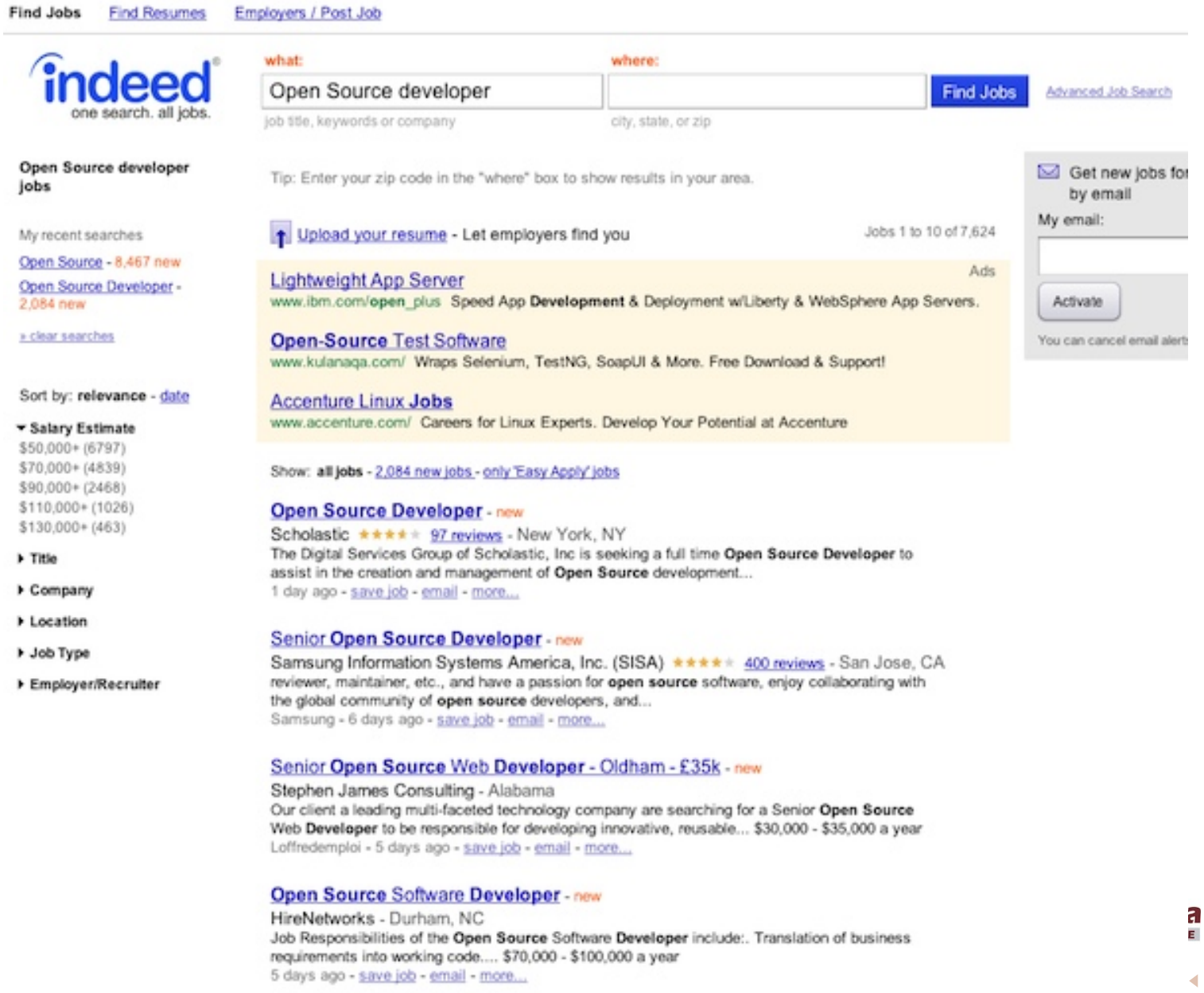
SPG SOFTWARE·PRODUCTIVITY·GROUP

- *Spring 2000.* Linux kernel. Contributor. USB driver for a scanner; accepted an
(drivers/usb/microtek.c). Userspace program to assist reverse engineering (usb-r

Há muitas oportunidades para trabalhar com FLOSS!



Há muitas oportunidades para trabalhar com FLOSS!



Find Jobs | Find Resumes | Employers / Post Job

indeed
one search. all jobs.

what: Open Source developer (job title, keywords or company)
where: (city, state, or zip)
Find Jobs | Advanced Job Search

Open Source developer jobs

My recent searches
Open Source - 8,467 new
Open Source Developer - 2,084 new
» clear searches

Sort by: **relevance** - date

Salary Estimate
$50,000+ (6797)
$70,000+ (4839)
$90,000+ (2468)
$110,000+ (1026)
$130,000+ (463)

Title
Company
Location
Job Type
Employer/Recruiter

Tip: Enter your zip code in the "where" box to show results in your area.

Upload your resume - Let employers find you

Jobs 1 to 10 of 7,624

Ads

Lightweight App Server
www.ibm.com/open_plus Speed App **Development** & Deployment w/Liberty & WebSphere App Servers.

Open-Source Test Software
www.kulanaqa.com/ Wraps Selenium, TestNG, SoapUI & More. Free Download & Support!

Accenture Linux **Jobs**
www.accenture.com/ Careers for Linux Experts. Develop Your Potential at Accenture

Get new jobs for by email
My email:
Activate
You can cancel email alert

Show: **all jobs** - 2,084 new jobs - only 'Easy Apply' jobs

Open Source Developer - new
Scholastic ★★★★☆ 97 reviews - New York, NY
The Digital Services Group of Scholastic, Inc is seeking a full time **Open Source Developer** to assist in the creation and management of **Open Source** development...
1 day ago - save job - email - more...

Senior **Open Source Developer** - new
Samsung Information Systems America, Inc. (SISA) ★★★★☆ 400 reviews - San Jose, CA
reviewer, maintainer, etc., and have a passion for **open source** software, enjoy collaborating with the global community of **open source** developers, and...
Samsung - 6 days ago - save job - email - more...

Senior **Open Source** Web **Developer** - Oldham - £35k - new
Stephen James Consulting - Alabama
Our client a leading multi-faceted technology company are searching for a Senior **Open Source** Web **Developer** to be responsible for developing innovative, reusable... $30,000 - $35,000 a year
Loffredemploi - 5 days ago - save job - email - more...

Open Source Software **Developer** - new
HireNetworks - Durham, NC
Job Responsibilities of the **Open Source** Software **Developer** include:. Translation of business requirements into working code.... $70,000 - $100,000 a year
5 days ago - save job - email - more...

Sua empresa pode economizar!

BLACK DUCK SOFTWARE ESTIMATES DEVELOPMENT COST OF OPEN SOURCE SOFTWARE AT $387 BILLION

Reuse of Open Source Software Could Save U.S. Companies $22 Billion a Year

WALTHAM, Mass., April 14, 2009 - Black Duck Software, a leading provider of products and services for accelerating software development through the managed use of open source software (OSS), estimates the total development cost of open source software at more than $387 Billion U.S.

To put this figure into perspective, the cost to develop open source software code is equivalent to nearly 50 percent of the U.S. government's recently passed stimulus bill, the $787 billion "American Recovery and Reinvestment Act."

Black Duck Software has the industry's most comprehensive database of open source software and related metadata. According to the company's research, there are over 200,000 open source projects representing over 4.9 billion lines of code. Using its detailed knowledge of open source projects and applying standard industry cost estimation techniques, Black Duck estimates that the total development cost of OSS exceeds $387 billion and represents a collective investment of more than two million developer years.

An additional analysis, which estimates that 10 percent of IT application development spending is redundant with existing open source projects, indicates U.S. companies could realize savings of more than $22 billion a year through the reuse of OSS in application development.

Sua empresa pode economizar!

BLACK DUCK SOFTWARE ESTIMATES DEVELOPMENT COST OF OPEN SOURCE SOFTWARE AT $387 BILLION

Reuse of Open Source Software Could Save U.S. Companies $22 Billion a Year

WALTHAM, Mass., April 14, 2009 - Black Duck Software, a leading provider of products and services for accelerating software development through the managed use of open source software (OSS), estimates the total development cost of open source software at more than $387 Billion U.S.

To put this figure into perspective, the cost to develop open source software code is equivalent to nearly 50 percent of the U.S. government's recently passed stimulus bill, the $787 billion "American Recovery and Reinvestment Act."

Black Duck Software has the industry's most comprehensive database of open source software and related metadata. According to the company's research, there are over 200,000 open source projects representing over 4.9 billion lines of code. Using its detailed knowledge of open source projects and applying standard industry cost estimation techniques, Black Duck estimates that the total development cost of OSS exceeds $387 billion and represents a collective investment of more than two million developer years.

An additional analysis, which estimates that 10 percent of IT application development spending is redundant with existing open source projects, indicates U.S. companies could realize savings of more than $22 billion a year through the reuse of OSS in application development.

Fonte de dados para pesquisa

MSR 2017

The 14th International Conference on Mining Software Repositories
May 20-21, 2017. Buenos Aires, Argentina.

Home | Program | Mining Challenge | Data Showcase | MSR Awards | Registration | Venue | Dates | Hall of Fame | Organization

Welcome to the official website of **MSR 2017**!

The Mining Software Repositories (MSR) field analyzes the rich data available in software repositories to uncover interesting and actionable information about software systems and projects. The goal of this two-day conference is to advance the science and practice of MSR. The 14th International Conference on Mining Software Repositories is sponsored by IEEE TCSE, ACM SIGSOFT, and SADIO.

The MSR conference is co-located with ICSE 2017, at **Buenos Aires, Argentina, 20-21 May, 2017.**

Call For Papers

Software repositories such as source control systems, archived communications between project personnel, and defect tracking systems are used to help manage the progress of software projects. Software practitioners and researchers are recognizing the benefits of mining this information to support the maintenance of software systems, improve software design/reuse, and empirically validate novel ideas and techniques. Research is now proceeding to uncover the ways in which mining these repositories can help to understand software development and software evolution, to support predictions about software development, and to exploit this knowledge in planning future development. The goal of this two-day international

Obrigado!

Resumo

- **(0)** executar, **(1)** estudar e modificar, **(2)** distribuir e **(3)** distribuir versões modificadas
- Software Livre e Software de Código Aberto são dois lados da mesma moeda
- Vale a pena abrir o código para
  1. software que **não é um diferencial** de negócios
  2. software que é uma **tecnologia possibilitadora** para vários negócios
- Há muitas formas de se beneficiar profissionalmente com FLOSS

Licença

Você tem o direito de:

Compartilhar — copiar e redistribuir o material em qualquer formato

Adaptar — remixar, transformar, e criar a partir do material

De acordo com os seguintes termos:

Attribution — Você deve atribuir o devido crédito, fornecer um link para a licença e indicar se foram feitas alterações. Você pode fazê-lo de qualquer forma razoável, mas não de uma forma que sugira que o licenciante o apoia ou aprova o seu uso.

NonCommercial — Você não pode usar o material para fins comerciais.

ShareAlike — Se você remixar, transformar ou criar a partir do material, tem de distribuir suas contribuições sob a mesma licença

Mais informações em http://creativecommons.org/licenses/by-nc-sa/3.0/br/