Open Source: a r/evolution in the software industry?

Original Publication: June 2004
v. 1.1.1 (April 4, 2005)

Salvino A. Salvaggio, Ph.D.
salvino@salvaggio.net
http://www.salvaggio.net/

Copyright (c) 2004-2005  Salvino A. Salvaggio
TABLE OF CONTENT

- Legal Notice ................................................................. 2
- Special thanks ............................................................... 3
- About the author ............................................................ 4
- Definitions ................................................................. 5
- Executive summary - Key messages ................................. 6
- Open Source: a r/evolution in the software industry ?
  1. Open Source: definition and history ............................ 7
  2. PROs and CONs of Open Source ................................. 47
  3. Is Open Source really changing the software industry ? .... 74
  4. Open Source in the company ....................................... 110
  5. Migration path to Open Source .................................... 126
  6. Review of cases ........................................................ 159
  7. Case study, NewCo in Electricity Distribution ............... 191

- Revision history ......................................................... 220

For better presentation of content, display this document as a slide show
LEGAL NOTICE

- Salvino A. Salvaggio (referred as SaS in this document) is the original author of this document that is protected by the Belgian Copyright (droit d'auteur) and he owns all the intellectual property rights on this document

- The author accepts any reproduction (total or partial, digital or not) of his document provided that it is only made for non-profit purposes. This document may be circulated, quoted, and reproduced for non-commercial and free-of-any-charge distribution on any support (paper, electronic files, CD-Rom, DVD, etc.)

- Regardless of the support, any quotation, reproduction and redistribution of this document (partial or total) must mention all the following information:
  - full name of its original author (Salvino A. Salvaggio)
  - his email address (salvino@salvaggio.net)
  - the URL address of his website (http://www.salvaggio.net/)

- The author warmly encourages readers to contribute to and improve this document. However,
  - in order to avoid chaotic multiplication of versions, any modification, improvement proposal must be submitted to the original author, discussed with him and approved by him (submission, discussion and approval can be done via email). The original author will then take care of including the approved modification into the document and to publish online the new version, with no guarantee on the timeframe
  - this document includes an invariant section, that can not be modified except by the original author himself, being the “Legal Notice”

- Redistributions are not allowed to add any additional restrictions when they redistribute this document

- This document must be considered as a scientific research document, not a commercial brochure. Brands and trademarks are mentioned only for scientific purposes (comparisons, analysis, examples, quotations, etc)

- This document was developed for the purpose of scientific knowledge improvement and class discussion rather than to illustrate either effective or ineffective handling of an administrative or managerial situation

- About 27 pages out of this document are based on slides coming from a McKinsey & Company’s internal report. Quotation and re-elaboration were explicitly allowed (by means of written document) by the McKinsey partner who authored the report

- A few pages out of this document are based on slides coming from an All2All's presentation. Quotation and re-elaboration were explicitly allowed (by means of written document) by the CEO of All2All who authored the presentation

- Salvino A. Salvaggio did this research and report as a personal private activity not linked at all with any of his current or previous employers. The views expressed in this document are only of Salvino A. Salvaggio

- In case of any dispute, the courts of Belgium will be the sole competent courts
SPECIAL THANKS

• Several people—experts, users, researchers, professionals, friends—provided me with highly valuable input while I was working on this research project. I want to thank them all, especially Eric Monnoyer, Jens-Ingo Brodesser, Mustafa Ylmaz Ali, Anton Kaska, Michael Hendrickx, Jacques Folon, Etienne Wery. However, they are not responsible for any mistake I might have left in this report.

• This research document also benefited from the comments and in-depth critical review of Stefan Champailler. He deserves special thanks.

• On top of a complete spelling and grammar review of the document, Sheldon Lee-Wen made several comments that helped me to improve the overall quality of the final version. Many thanks to him!

• Last but not least, I want also to thank all the professionals who devoted some of their precious working time to answering my interviews and questionnaires.
ABOUT THE AUTHOR

Salvino A. Salvaggio brings over 12 years of international senior management and consulting experience (innovation, strategy, organization, operations, marketing) as well as of entrepreneurial mindset and change management expertise

- Salvino is currently Head of Internet Portal in Qatar Telecom. He leads the project of creating a customer-focused new Internet portal offering an extended set of value added services for broadband and mobile users
- Before joining his current position, Salvino was an Expert consultant in McKinsey & Co. He was involved in management consulting projects primarily oriented to strategy definition, business planning, and business building of new and innovative ventures, especially in the ebusiness domain
- Before, Salvino worked as an Experienced Manager for Accenture. He led the development of the Brussels Innovation Center supporting new business projects and creative programmes
- During the 1990’s, he also launched and managed three businesses while working as a freelance consultant in the new media industry
- From 1999 to 2003, Salvino was external anchorman at the Belgian radio and TV public network (RTBF), expert in new digital media & communication
- In the 1980’s and 90’s, he was an academic in several European and North American universities
- Salvino has a Doctorate (Ph.D. - Communication Systems Theory). He also holds a corporate mMBA and several post-graduate programs in Anthropology, Political Sciences, Sociology, Philosophy. He has published more than 80 articles in professional and scientific international journals, as well as 8 books. Most of his publications are downloadable free of charge from his personal web site http://www.salvaggio.net/
DEFINITIONS (1/2)

- **Apache**: a open source web server. An application used to put content on the internet.
- **Back door**: In the context of software security, a hole left in the security of an application that allows its author to access its core information. Contrary to a bug, a back door is usually intentional.
- **EULA**: End User License Agreement. See Software License.
- **Free software**: Software that is free in the sense that it can't be made proprietary.
- **GPL**: General Public License. See Software License.
- **Graphical User Interface (GUI)**: In the context of an operating system, a functionality that is added on top of the operating system to allow the user to work with a desktop metaphor. For example, on Linux, the most famous interfaces are KDE and Gnome. On Windows, the interface is blended into the OS, so it doesn't exist on its own.
- **Hacker**: A slang term for a computer enthusiast. Among professional programmers, the term hacker implies an amateur or a programmer who lacks formal training. Depending on how it used, the term can be either complimentary or derogatory, although it is developing an increasingly derogatory connotation. The pejorative sense of hacker is becoming more prominent largely because the popular press has co-opted the term to refer to individuals who gain unauthorised access to computer systems for the purpose of stealing and corrupting data. Hackers, themselves, maintain that the proper term for such individuals is cracker. In this document, the term Hacker refers to its original complimentary definition.
- **IIS**: A closed source web server provided by Microsoft.
- **Lock in**: A situation which arises when it is difficult to replace a software application by another one (be it because functionalities differ, the application license or the contract with the vendor forbids it).
- **Migration**: In the software applications context, the action of moving from one application (or operating system) to another. For example, migrating form Microsoft Office to Open Office.
- **MySQL**: An open source relational database system.
- **NPV (Net Present Value)**: The future stream of benefits and costs converted into equivalent values today. This is done by assigning monetary values to benefits and costs, discounting future benefits and costs using an appropriate discount rate, and subtracting the sum total of discounted costs from the sum total of discounted benefits.
- **Open source software**: software for which one is allowed to read the source code. Optionally, one can also modify or redistribute it.

Source: Google labs definitions; online dictionaries and encyclopedias
DEFINITIONS (2/2)

• **Open source software**: software for which one is allowed to read the source code. Optionally, one can also modify or redistribute it

• **Oracle**: A powerful closed source database system

• **OS**: Operating System, for example Mac OS X, Windows or Linux

• **Patch**: A piece of program-code (open source or not), usually small, that is used to improve or fix an existing program

• **ROI (Return On Investment)**: Used to measure the efficiency with which financial resources available to a company are employed by management. Calculated: \[ \text{ROI} = \frac{\text{Annual profit}}{\text{Average amount Invested}} \]

• **Software License**: A legal text that defines how and/or under which circumstances people are allowed to use the software it covers

• **TCO**: Total cost of ownership

• **Viral**: In the context of a software license, denotes that if the software is used to build another software, the latter must be under the same license too

Source: Google labs definitions; online dictionaries and encyclopaedias
EXECUTIVE SUMMARY - KEY MESSAGES

• Any software whose code is available for users to look at, modify, reuse and redistribute freely can be called "Open Source Software" (OSS). There are however some differences between OSS and free software regarding mostly users’ duties and intellectual property rules.

• The phenomenon of OSS is not new: at the beginning, programmers, not software, were core; later on, rights of usage became a way to earn money; however, some programmers called for a return to the public sharing of software; then Linux came...

• Production of OSS happens according to a self-regulated, decentralized organization that gave OSS development unique opportunities while a new approach of collaborative work improved the sustainability of the project and made it able to impact key economic levers.

• Linux hype cycle has been guided, among others, by perception of PROs and CONs. Recently, perceived benefits of OSS seem to be overtaking barriers and increasing its appeal to businesses. As a matter of fact, global comparison highlights a real advantage to OSS. It is therefore not a surprise that finding evidences of OSS as a technologically and financially viable solution is becoming quite easy.

• Compared to vendors’ closed source software OSS secures huge IT investment savings, and total Return On Investment is highly attractive. Additionally, OSS users are empowered, for they do not depend anymore on vendors’ decisions.

• OSS users’ satisfaction is likely to foster a viral effect of adoption as evolution of OSS market share also shows.

• With OSS, the concept of vendors is disappearing. Shifts in vendors’ revenue streams trigger deep changes in the culture of the software business as well as in the industry at large.

• Available OSS solutions already meet a wide range of companies’ needs for any IT segment (core data center, server, office, image manipulation, content management system, etc.)

• Migrating to OSS should follow a sharp process: analysis of the pros and cons, analysis of economic efficiency/impact, implementation of the migration itself according to a planned path.

• Several real-life cases show migrating to OSS is a winning move from both IT strategy point of view and, especially, a business strategy point of view.
1. Open Source: definition and history
1. Open Source: definition and history

- What is Open Source?
- A little Open Source history
- An organized bazaar
OPEN SOURCE SOFTWARE: A SIMPLIFIED DEFINITION

Any software whose code is available for users to look at, modify, reuse and redistribute freely.
OPEN SOURCE vs FREE SOFTWARE?

How does open source software differ from free software?

In practice, nearly all software meeting one definition also meets the other: both licenses permit users the freedom to run the program for any purpose, to study and modify the program, and to freely redistribute copies of the original or modified program.

The term open source software is often used by people who wish to stress aspects such as high reliability and flexibility of the resulting program as the primary motivation for developing such software.

The term free software stresses freedom from control by another. To understand the concept, you should think of free as in “free speech,” not as in “free meal.”

Source: David A. Wheeler's personal home page; gnu.org; SaS analysis
Free / Open software refers to four kinds of freedom, for the users of the software:

- **Freedom 1**: The freedom to run the program, for any purpose
- **Freedom 2**: The freedom to study how the program works, and adapt it to your needs
- **Freedom 3**: The freedom to redistribute copies so you can help your neighbour
- **Freedom 4**: The freedom to improve the program, and release your improvements to the public, so that the whole community benefits

Access to the source code is a precondition for this.
A MORE SYSTEMATIC DEFINITION OF OPEN SOURCE SOFTWARE...

<table>
<thead>
<tr>
<th>For free:</th>
<th>Private software</th>
<th>Freeware (free private software)</th>
<th>Open source software</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Everybody can execute it</td>
<td>✗</td>
<td>✗</td>
<td>✓*</td>
</tr>
<tr>
<td>• Everybody can modify the source code</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>• Everybody can distribute it</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>• Everybody can distribute a modified version of the source code</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
</tbody>
</table>

Open source software is characterized by the ability to acquire a software freely, get access to its source code, modify it and distribute a modified version of it

*Collaborative Source Initiative promotes a different form of open source: access to code is free, freedom of adaptation is granted but execution requires payment of royalties*

Source: Press Clippings; Open Source-focused web sites, McKinsey
...AND A DETAILED COMPARISON OF TYPICAL OPEN SOURCE AND FREE SOFTWARE LICENSES

<table>
<thead>
<tr>
<th>Usage of original software</th>
<th>Open Source Software**</th>
<th>Free Software***</th>
</tr>
</thead>
<tbody>
<tr>
<td>• User can use the software at will, without restriction and without guarantee</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Redistribution of a modified version</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>• User can redistribute modifications as source or compiled code</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>• User can market the compiled code</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>• User can market the source code</td>
<td>✓</td>
<td>×</td>
</tr>
<tr>
<td>• User must make the source code of the modifications available to anyone who ask them, free of charge or for a minimum fee</td>
<td>×</td>
<td>✓</td>
</tr>
<tr>
<td>• User must publish modifications under the same license terms as the original software*</td>
<td>×</td>
<td>✓</td>
</tr>
</tbody>
</table>

* This is called a "viral" license
** Apache
*** GPL licensed software
Source:  S. Champailler
FROM FREEDOM TO DUTIES

While free source software users enjoy substantial freedom regarding 4 crucial aspects of the software industry, they also have clear duties:

• The users have to share any improvement with the community and the author(s)

• When distributing the software, the source code also has to be distributed too and the copyright (GPL license*) must be attached

Compared to open source users, users of free software must meet one further obligation:

• All linked libraries must also be under the GPL

This difference makes a free software license somehow more “pure” than the open source one

* GNU GPL stands for GNU General Public License, while GNU is a reflexive acronym for GNU ’s Not Unix. Copyright under GPL license is also called Copyleft. For detailed information on license, surf www.gnu.org/licenses/licenses.html

Source: debian.org
SPECIFICITIES OF OPEN SOURCE SOFTWARE

A source code available to everybody
- Everybody can get it mostly for free
- Everybody can modify it
- Everybody can distribute it

No clearly registered developers community
- Unformal (sometimes, non hierarchical) relationships among developers
- Charismatic leader ensuring consistency of developments
- Quite often times, large projects have a release schedule. Small projects do not
- Large projects have a centralized project planning and resources allocation. Small do not but project contributions are centralized
- Massive debugging approach

Organization & structure
Incentives/type of benefits
- Non-lucrative approach for most developers
- Merits of contributors made public ("meritocracy")

Specific intellectual property rules
- "Viral" licenses
- Non viral licenses

What are the open source model specificities?

A method of development*

Private software editors can adopt similar practices (e.g. Microsoft has similar development practices)

However, large projects such as Debian, KDE, GNOME, do have registered business entities or foundations as well as corporate sponsorship, an internet presence, clear organization or established ways of doing things.

Source: McKinsey analysis
INTELLECTUAL PROPERTY RULES

Open source software

- Around 85% of open source software is protected by the **General Public License (GPL)**, which aims at preventing any part of a cooperatively developed software from being turned into private software
  - **Users** are permitted to run the program, copy and/or modify it through its source code, and distribute it to others
  - **GPL is « viral »,** thus extends to any development integrating a piece of software protected by GPL. An application developed around a GPL-protected software must become itself GPL-protected
- **The GPL license is a must-have** to get the full support of the open source community, since other licenses could, under certain conditions, perceived as a bias to the open source ideology

Private software

- Software can be used and reproduced only with a license from the author allowing this usage
- Software cannot be distributed and modified without specific agreement from the author
- These activities are usually subject to financial payment

Non-viral licenses

- **A famous derivative of the GPL is the LGPL (Library Lesser General Public License)** which limits the « viral » aspect of GPL and ensures that developments made around a LGPL software are not necessarily GPL’ed
  - **Many companies** develop their own derivatives of the GPL to ensure an efficient protection of their software (e.g. IBM, Open Cascade, Dresdner with Open Adaptor)
  - **The Apache, BSD and the Mozilla Public licenses** are "non-viral" licenses: the code initially released may be extended with proprietary code to create proprietary products

* Apache is not GPL but is still strongly supported by the community

Source: McKinsey analysis; SaS analysis
THE COMMON DEFINITION* OF OPEN SOURCE SOFTWARE (1/11)

There are actually several definitions of Open Source. The one we focus on here belongs to the Open Source Initiative (OSI), a non-profit corporation dedicated to managing and promoting the Open Source Definition.

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

- Free Redistribution
- Source Code
- Modifications and Derived Works
- Integrity of The Author's Source Code
- No Discrimination Against Persons or Groups
- No Discrimination Against Fields of Endeavor
- Distribution of License
- License Must Not Be Specific to a Product
- License Must Not Restrict Other Software
- License Must Be Technology-Neutral

Implications

“The basic idea behind open source is very simple: When programmers can read, redistribute, and modify the source code for a piece of software, the software evolves. People improve it, people adapt it, people fix bugs. And this can happen at a speed that, if one is used to the slow pace of conventional software development, seems astonishing.

“We in the open source community have learned that this rapid evolutionary process produces better software than the traditional closed model, in which only a very few programmers can see the source and everybody else must blindly use an opaque block of bits.”

* There are actually several definitions of Open Source. The one we focus on here belongs to the Open Source Initiative (OSI), a non-profit corporation dedicated to managing and promoting the Open Source Definition.

Source: opensource.org
THE COMMON DEFINITION* OF OPEN SOURCE SOFTWARE (2/11)

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

- Free Redistribution
  - Source Code
  - Modifications and Derived Works
  - Integrity of The Author's Source Code
  - No Discrimination Against Persons or Groups
  - No Discrimination Against Fields of Endeavor
  - Distribution of License
  - License Must Not Be Specific to a Product
  - License Must Not Restrict Other Software
  - License Must Be Technology-Neutral

- 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*

---

* Actually, GPL allows programmers to ask fee for distribution of their software, even if Open Source Source: opensource.org
**THE COMMON DEFINITION OF OPEN SOURCE SOFTWARE (3/11)**

<table>
<thead>
<tr>
<th>Rationale</th>
<th>Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with several structured criteria:</th>
</tr>
</thead>
</table>
|           | • Free Redistribution  
|           | • Source Code  
|           | • Modifications and Derived Works  
|           | • Integrity of The Author's Source Code  
|           | • No Discrimination Against Persons or Groups  
|           | • No Discrimination Against Fields of Endeavor  
|           | • Distribution of License  
|           | • License Must Not Be Specific to a Product  
|           | • License Must Not Restrict Other Software  
|           | • License Must Be Technology-Neutral |

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

- Free Redistribution
- Source Code
- Modifications and Derived Works
- Integrity of The Author's Source Code
- No Discrimination Against Persons or Groups
- No Discrimination Against Fields of Endeavor
- Distribution of License
- License Must Not Be Specific to a Product
- License Must Not Restrict Other Software
- License Must Be Technology-Neutral

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

Source: opensource.org
### Rationale
Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with several structured criteria:

- Free Redistribution
- Source Code
- Modifications and Derived Works
- Integrity of The Author's Source Code
- No Discrimination Against Persons or Groups
- No Discrimination Against Fields of Endeavor
- Distribution of License
- License Must Not Be Specific to a Product
- License Must Not Restrict Other Software
- License Must Be Technology-Neutral

### The Common Definition of Open Source Software (5/11)

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

- Free Redistribution
- Source Code
- Modifications and Derived Works
- Integrity of The Author's Source Code
- No Discrimination Against Persons or Groups
- No Discrimination Against Fields of Endeavor
- Distribution of License
- License Must Not Be Specific to a Product
- License Must Not Restrict Other Software
- License Must Be Technology-Neutral

The license must not discriminate against any person or group of persons.
Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with several structured criteria:

- Free Redistribution
- Source Code
- Modifications and Derived Works
- Integrity of The Author's Source Code
- No Discrimination Against Persons or Groups
- No Discrimination Against Fields of Endeavor
- Distribution of License
- License Must Not Be Specific to a Product
- License Must Not Restrict Other Software
- License Must Be Technology-Neutral

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.
THE COMMON DEFINITION OF OPEN SOURCE SOFTWARE (8/11)

<table>
<thead>
<tr>
<th>Rationale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with several structured criteria:</td>
</tr>
</tbody>
</table>

- Free Redistribution
- Source Code
- Modifications and Derived Works
- Integrity of The Author's Source Code
- No Discrimination Against Persons or Groups
- No Discrimination Against Fields of Endeavor
- Distribution of License
- License Must Not Be Specific to a Product
- License Must Not Restrict Other Software
- License Must Be Technology-Neutral

• 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
THE COMMON DEFINITION* OF OPEN SOURCE SOFTWARE (9/11)

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

- Free Redistribution
- Source Code
- Modifications and Derived Works
- Integrity of The Author's Source Code
- No Discrimination Against Persons or Groups
- No Discrimination Against Fields of Endeavor
- Distribution of License
- License Must Not Be Specific to a Product
- License Must Not Restrict Other Software
- License Must Be Technology-Neutral

- 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*

* This point basically belongs to free software viral license, not to Open Source *senso strictu*

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

- Free Redistribution
- Source Code
- Modifications and Derived Works
- Integrity of The Author's Source Code
- No Discrimination Against Persons or Groups
- No Discrimination Against Fields of Endeavor
- Distribution of License
- License Must Not Be Specific to a Product
- License Must Not Restrict Other Software
- License Must Be Technology-Neutral

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

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

- Free Redistribution
- Source Code
- Modifications and Derived Works
- Integrity of The Author's Source Code
- No Discrimination Against Persons or Groups
- No Discrimination Against Fields of Endeavor
- Distribution of License
- License Must Not Be Specific to a Product
- License Must Not Restrict Other Software
- License Must Be Technology-Neutral

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

- No provision of the license may be predicated on any individual technology or style of interface
OPEN SOURCE SOFTWARE PHENOMENON: A FEW FACTS

WHAT IT IS

• Linux is the most famous open source solution but there are many other open software languages and solutions (e.g. Apache, PHP, OpenOffice, MySQL, etc.)

• Companies can make money with open source despite its being free, by providing related services and products (i.e. Red Hat, Alcove, All2All, etc.)

• Open source software versions are developed in order to be compatible with the existing previous versions, as is the case for most proprietary software

• An open source software is a software delivered with unrestricted access to its source code*

• Its license generally grants the right to run the program, own a copy of the program’s source code, modify and distribute a modified version under the same terms as the original**

WHAT IT IS NOT

• Open source is not only the work of high school and college students

• Open source providers are not only fly-by-night companies that won’t last: ex. IBM provides Linux

• An open source software is not necessarily less safe than a proprietary one

• Open source software are not necessarily less reliable than private software since each developer who contributes to its development periodically reviews and tests others’ job at the source code level

* A source code is a list of instructions in a programming language of a software

** Unless explicitly specified in the software license

Source: McKinsey analysis; Press; Interviews; Web sites; SaS analysis
1. Open Source: definition and history

- What is Open Source?
- A little Open Source history
- An organized bazaar
THE PHENOMENON OF OPEN SOURCE SOFTWARE IS NOT NEW

Even though the development of Open Source software only becomes well known over the last decade especially because of the success of Linux OS, it started long before, in the seventies.

1. At the beginning, programmers were paid for the act of programming, not for the programs themselves.

2. Only when computers reached the business world, did programmers begin to support themselves by restricting the rights to their software and charging fees for each copy.

3. As a reaction, Free Software idea has been popularized since the 80s, when the Free Software Foundation and its GNU Project were founded: people should have more freedom, and should appreciate their freedom to execute, modify and distribute software. A set of rules was designed, and codified in the GNU General Public License or GPL.

4. A growing community of programmers was attracted by Open Source philosophy and contributed by developing an impressive set of open software, from OS (Linux), to network tools (Apache, Mozilla), to desktop productivity applications (OpenOffice, etc.)

Source: EURESCOM; O'Reilly; http://ragib.hypermart.net/; NetworkWorldFusion
AT THE BEGINNING, PROGRAMMERS, NOT SOFTWARE, WERE CORE

The U.S. Patent and Trademark Office historically has been reluctant to grant patents on inventions relating to computer software

- In the 1970s, the P.T.O. avoided granting any patent if the invention utilized a calculation made by a computer. Their rationale was that patents
  - could only be granted to processes, machines, articles of manufacture, and compositions of matter
  - could not be granted to scientific truths or mathematical expressions of it

As such, software related inventions were considered non-statutory

When computers first reached universities, they were research tools. Software was passed around, and programmers were paid for the act of programming, not for the programs themselves.

Source: EURESCOM ; O'Reilly ; http://ragib.hypermart.net/ ; NetworkWorldFusion ; Bitlaw
LATER, RIGHTS OF USAGE BECAME A WAY TO EARN MONEY

A Brief History of the DOS

- **1980 April**, Tim Patterson begins writing an operating system for use with Seattle Computer Products' 8086-based computer
- **August**, QDOS 0.10 (Quick and Dirty Operating System) is shipped by Seattle Computer Products. Even though it had been created in only two man-months, the DOS worked surprisingly well
- **September**, Tim Patterson shows Microsoft his 86-DOS, written for the 8086 chip.
- **October**, Microsoft's Paul Allen contacts Seattle Computer Products' Tim Patterson, asking for the rights to sell SCP's DOS to an unnamed client (IBM). Microsoft pays less than US$100,000 for the right
- **December**, Seattle Computer Products renames QDOS to 86-DOS, releasing it as version 0.3. Microsoft then bought non-exclusive rights to market 86-DOS
- **1981 February**, MS-DOS runs for the first time on IBM's prototype microcomputer
- **July**, Microsoft buys all rights to DOS from Seattle Computer Products, and the name MS-DOS is adopted.

Source: EURESCOM; O'Reilly; http://ragib.hypemart.net/; NetworkWorldFusion; PC Museum
HOWEVER, SOME PROGRAMMERS CALLED FOR A RETURN TO THE PUBLIC SHARING OF SOFTWARE

As a political idea and a professional philosophy, several programmers promoted a new set of rules to grant people (both programmers and users) with more freedom as they should be assured of crucial rights: implement, copy, modify and redistribute code.

- **In the 70s**, many cooperative development efforts focused on building an operating system that could run on multiple platforms. The Unix OS emerged as the most successful of these efforts. Even though Unix was the result of a collaborative effort, AT&T claimed intellectual property rights.

- **1983**, as a reaction, Richard Stallman wrote the GNU Manifesto. As the author stated: “I consider that the golden rule requires that if I like a program I must share it with other people who like it.”

- **1984**, the GNU project is launched to develop the free operating system GNU (an acronym for “GNU's Not Unix”), and thereby give computer users the freedom that most of them have lost.

- **1985**, the MIT based X Consortium distributes the X Window System as Open Source software.

- The Free Software Foundation (FSF) is founded, dedicated to promoting computer users' right to use, study, copy, modify, and redistribute computer programs. The FSF also helps to spread awareness of the ethical and political issues of freedom in the use of software.

- **1989**, Cygnus, the first commercial support service provider for Open Source software is founded.

Source: EURESCOM; O'Reilly; [http://ragib.hypermart.net/](http://ragib.hypermart.net/); NetworkWorldFusion; GNU project and FSF; MITRE
Then Linux came...

- **1986**, developers attempted to build a free version of the Unix OS (the GNU project)

- A solution seemed to appear in form of MINIX. It was written from scratch by Andrew Tanenbaum, a Dutch professor who wanted to teach his students the inner workings of a real operating system. MINIX had the advantage that the source code was available

- In **1991**, Linus Torvalds created the Linux OS and gave hackers his code so they could contribute to the development

- Linux grew dramatically, new features, new GUIs were added

- This created a precedent and a demand for new applications based on the same freedom principles. Web servers, application servers, coding languages, CMS, DBs, etc. are now available as Open Source softwares. For instance, Sourceforge currently hosts more than 90,000 projects

Hundreds, thousands, hundreds of thousands of programmers joined the GNU project, then the Linux community, the Open Source community at large, developing tons of new applications for both servers and desktops.

Source: EURESCOM; O'Reilly; [http://ragib.hypermart.net/](http://ragib.hypermart.net/); NetworkWorldFusion; Ragib Hasan
AFTER SEVERAL YEARS IT WAS BEING CONSIDERED AS A MINOR CHALLENGER, LINUX ENTERS THE MAINSTREAM

- Researchers
- Scientists
- Service providers
- Small businesses
- “Hidden” use in medium-sized and large operations
- Enterprise early adopters: retail, finance, insurance
- Enterprise deployment
- Additional industry adoption
- Major enterprise adoption

Source: IDC
ALTHOUGH LINUX IS ORIGINALLY A EUROPEAN PHENOMENON, IT IS SPREADING WORLDWIDE

Geographical repartition of the developers involved in the Linux debian project

October 2004

Source: debian.org/devel/developers.loc
1. Open Source: definition and history

• What is Open Source?

• A little Open Source history

• An organized bazaar
CATHEDRAL OR BAZAAR?

Who would have thought in 1991 that a world-class operating system could coalesce as if by magic out of part-time hacking by several thousand developers scattered all over the planet, connected only by the tenuous strands of the Internet?

A research into software engineering during the last 20 years shows software development can adopt only 2 fundamentally different organizational models:

The Cathedral
- A centralized, highly organized team is responsible for producing software
- Team leader plans the project, assigns tasks and allocates resources

The Bazaar
- Community-based software development
- Project coordinator integrates contributions coming from a multitude of spontaneous contributors

Source: Eric S. Raymond, The Cathedral and the Bazaar; SaS analysis
THE CATHEDRAL: CENTRALIZED ORGANIZATION

• Hierarchical authority
• Limited number of selected developers
• Limited number of testers and reviewers

The Cathedral
Centralized organization

• Communication under confidentiality measures
• Limited number of scheduled releases
• Traditional project planning approach

The Project Management team takes decisions, allocates resources, sets deadlines, splits work streams, etc.

Source: Eric S. Raymond, *The Cathedral and the Bazaar*; SaS analysis
THE BAZAAR: COORDINATED ORGANIZATION

- Babbling chaos of differing agendas and approaches
- Unlimited number of testers and reviewers
- Decentralized implementation
- Self-regulated system
- Open communication over public forums on the Internet
- Unlimited number of self-selected developers
- Frequent releases (“Release early, release often”)

There is a working system (established on rules defined by the originators) on which contributors spontaneously plug their efforts

Source: Eric S. Raymond, *The Cathedral and the Bazaar*; SaS analysis
## Self-regulated, decentralized system

### CONs
- Key developers are not contractually “locked in” to the development of software
- Many web sites, none being authoritatively the home of Open Source
- Duplication of effort in documentation maintenance
- Many marketing efforts coming from different directions and organizations
- High number of versions of the same applications (including Linux) may make potential business clients confused, and high number of graphical libraries for GUIs make dialog and support more difficult
- No unique support framework provided to companies using Open Source

### PROs
- Independent organizations can act in their own best interest
- Having independent organizations reduces the incidence of bottleneck
- When ownership of software is distributed to many people around the world, this discourages attempts to privatize it as any one person or organization's asset
- Independent organizations can agree to disagree, when necessary
- Having multiple implementations of things
  - reduces the risks of experimentation
  - helps coping with conflict

*Source: Christopher Browne, *Linux and Decentralized Development*
...WHILE A NEW APPROACH OF COLLABORATIVE WORK IMPROVED THE SUSTAINABILITY OF THE WHOLE PROJECT

In a team work environment, each contributor can behave in very different ways even though s/he considers meeting personal interest as a key performance indicator

To maximize personal interest:

- **On the demand side**, the contributor can fight to overuse the common product of the team (congested public good problem)

- **On the supply side**, the contributor can reduce his participation and count on the team (free-rider behavior)

<table>
<thead>
<tr>
<th>Usage</th>
<th>Contribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>High</td>
<td>High</td>
</tr>
</tbody>
</table>

Closed source competitive model

Open Source collaborative model

To maximize personal interest:

- **On the demand side**, the contributor can fight to widespread the usage in order to increase the value of the whole project

- **On the supply side**, the contributor can actively contribute to increase the value of final outcome

Source: Christopher Browne, *Linux and Decentralized Development*; SaS analysis
THE MORE OPEN SOURCE USERS, THE MORE VALUABLE OPEN SOURCE BECOMES

Demand side

Software cannot be degraded by overuse

Widespread use of open source software tends to increase its value, as users fold in their own fixes and features (code patches)

Source: Christopher Browne, *Linux and Decentralized Development*; SaS analysis
THE MORE OPEN SOURCE CONTRIBUTORS, THE MORE VALUABLE OPEN SOURCE BECOMES

Supply side: why open source doesn't suffer from underprovisionning?

People don't merely need solutions, they need solutions on time

- If the payoff from fixing a bug or adding a feature is sufficient to any potential contributor, that person will dive in and do it

Putative market value of small patches to a common source base is hard to capture and to assign. Therefore the contributor gains when

- passing maintenance overhead of the software to the source-code owners and the rest of the project group
- others improve on his work in the future
- not having to personally maintain the software, and being able to afford more time on other projects

Source: Christopher Browne, *Linux and Decentralized Development*; SaS analysis
THE BAZAAR OVERTOOK THE CATHEDRAL...

In the mid-90s, “the fact that this bazaar style seemed to work, and work well, came as a distinct shock. (...) the Linux world not only didn't fly apart in confusion but seemed to go from strength to strength at a speed barely imaginable to cathedral-builders”

Source: Eric S. Raymond, *The Cathedral and the Bazaar*
Open source enables players to...

Reduce cost
- Reduce cost of development thanks to the access to cheaper, highly qualified and flexible developers around the world
- Reduce cost (if any) of acquisition & license
- Reduce costs of development of new products (i.e. Linux) or new functionalities
- Reduce costs of extending the lifetime of existing products

Generate revenue
- Generate direct revenue
- Generate indirect revenue (brand image, marketing argument)
- Sell more core products
- Sell other offerings (e.g. documentation, services)

Diversify portfolio
- Diversify supplier portfolio
- Diversify customer portfolio
- Increase bargaining power
- Decrease risks and dependence on highly concentrated players
- Increase the reach with a larger offering
- Develop niche/derived activities

...AND MADE OPEN SOURCE ABLE TO REALLY IMPACT KEY ECONOMIC LEVERS

Source: McKinsey analysis
2. PROs and CONs of Open Source
2. PROs and CONs of Open Source

• The Linux hype cycle

• CONs

• PROs

• Conclusion
LINUX HYPE CYCLE HAS BEEN GUIDED, AMONG OTHERS, BY PERCEPTION OF PROs AND CONs

Source: Gartner; Novell; Press clippings; SaS analysis
A FEW YEARS AGO, OPEN SOURCE APPEAL WAS NOT VERY CLEAR TO THE COMMUNITY OF ANALYSTS...

**Open Source in the press**

<table>
<thead>
<tr>
<th>Title</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Future looks bright for Linux</td>
<td>Computer Reseller, 01/31/2001</td>
</tr>
<tr>
<td>Linux for core business</td>
<td>Informatiques Magazine, 03/02/2001</td>
</tr>
<tr>
<td>Government and Industry unite to push open source</td>
<td>Computer Weekly, 12/12/2001</td>
</tr>
<tr>
<td>Linux, the penguin marches on</td>
<td>Newswire, 11/12/2001</td>
</tr>
<tr>
<td>Love for Linux</td>
<td>The Guardian, 10/11/2001</td>
</tr>
<tr>
<td>IT managers become more “open”-minded</td>
<td>InformationWeek 04/16/2001</td>
</tr>
<tr>
<td>Slowdown forces Linux firms to restructure</td>
<td>PC Dealer, 09/24/2001</td>
</tr>
<tr>
<td>Linux, some progress, but no tidal wave</td>
<td>01 Informatique, 11/02/2001</td>
</tr>
<tr>
<td>Linux companies continue to struggle</td>
<td>Network Magazine 09/17/2001</td>
</tr>
<tr>
<td>Red Hat remains in the red</td>
<td>Information Weekk, 12/24/2001</td>
</tr>
<tr>
<td>Why Caldera is wavering</td>
<td>Journal Informatique, 09/10/2001</td>
</tr>
<tr>
<td>Linux faces crucial year</td>
<td>IDG Data, 30/10/2001</td>
</tr>
</tbody>
</table>

Source: McKinsey analysis
WHILE RECENTLY PERCEIVED BENEFITS OF OPEN SOURCE SEEM OVERTAKING BARRIERS

Perceived benefits

Technical benefits

Independence from software vendors

Improvement of image

Financial benefits

Perceived barriers

Provider reliability

Doubts on support

Fear and uncertainty

Open Source in experts’ reports

Source: Forrester; IDC; Novell; KBSt; Internet.com; Statkontoret; GIGA; Robert Frances Group; SaS analysis
2. PROs and CONs of Open Source

• The Linux hype cycle

• CONs

• PROs

• Conclusion
INSUFFICIENT LEVEL OF CONSULTING SKILLS NURTURES COMPANIES’ DOUBTS ON ABILITY TO BE SUPPORTED

Availability of open source solutions is increasing but systems integrators still have limited serious offerings and skills

A crucial question for enterprises is how to fill the gap between existing open solutions and lack of skilled large IT consulting companies really able to help them.

Source: McK analysis; press clippings; SaS analysis
COMMUNITY-BASED SOFTWARE DEVELOPMENT MODEL ALSO SCARES NON-USER ENTERPRISES

While professional users of Open source feel comfortable with community responsiveness, potential enterprise customers doubt open source communities’ ability to adapt solutions to changing market requirements for business applications (e.g. HR regulatory requirements, B2B interactions and standards) even if commercial open source service providers are growing on the market.

Do you think open source communities are responsive enough in adapting software solutions to changing market?

<table>
<thead>
<tr>
<th></th>
<th>Non-Users</th>
<th>Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>25%</td>
<td>70%</td>
</tr>
<tr>
<td>No</td>
<td>75%</td>
<td>30%</td>
</tr>
</tbody>
</table>
LACK OF AWARENESS MAKES TECHNICAL SUPPORT A NON- USERS’ PRIMARY CONCERN

**Typical reactions of non users**

- "Who do I call when things go wrong with open source? There is no Hot Line in case of emergency"
- "You can't wring a vendor's neck when there's no vendor"

<table>
<thead>
<tr>
<th>Support Solution</th>
<th>&quot;Yes&quot; among non users</th>
<th>&quot;Yes&quot; among users</th>
</tr>
</thead>
<tbody>
<tr>
<td>Community of developers</td>
<td>45%</td>
<td>85%</td>
</tr>
<tr>
<td>Mailing lists and forums</td>
<td>30%</td>
<td>75%</td>
</tr>
<tr>
<td>Support databases</td>
<td>7%</td>
<td>50%</td>
</tr>
<tr>
<td>Specialized consulting firms</td>
<td>45%</td>
<td>90%</td>
</tr>
<tr>
<td>&quot;After-sales&quot; assistance</td>
<td>20%</td>
<td>65%</td>
</tr>
</tbody>
</table>

Support solutions exist but potential users seem not to know them very well compared to actual users*

---

* However, according to analysts, support is a bit of a fallacy because organizations pay a lot for support they rarely use. If they do use the support, sometimes support effectiveness at resolving the issue is not optimal. Unless they cannot get software patches without a support agreement, and/or the system requires 24x7 support and is critical to business functions, some analysts recommend organizations use per-incident support.

Source: Interviews; CIO.com; SaS analysis; Sheldon Lee-Wen
SOME OPEN SOURCE CONSULTANTS ARE STILL LESS KNOWLEDGEABLE ABOUT VENDORS’ ARCHITECTURES

<table>
<thead>
<tr>
<th>Average time to answer 14 questions on Windows NT architectures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minutes:Seconds</td>
</tr>
<tr>
<td>By IT generalist consultants</td>
</tr>
<tr>
<td>By Open Source consultants</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Average number of wrong answers</th>
</tr>
</thead>
<tbody>
<tr>
<td>By IT generalist consultants</td>
</tr>
<tr>
<td>By Open Source consultants</td>
</tr>
</tbody>
</table>

Open Source consultants seem being less knowledgeable about usual Windows NT architectures than IT generalist consultants

Source: Interviews; questionnaires; SaS analysis
ESTABLISHED SOFTWARE COMPANIES COMMUNICATE HEAVILY ON THE RISKS OF USING OPEN SOURCE

- Lower Windows Staffing Costs Provide a TCO Advantage Over Linux, IDC report, 2002

Microsoft-sponsored studies!


Reports prepared under contract from Microsoft!

On Jan. 2004, Microsoft launched an advertising campaign, referred to as "Get the Facts," which is designed to give customers information about the advantages of using its Windows operating system versus Linux.

Source: McKinsey analysis; eWeeks.com; microsoft-watch.com; press clippings; SaS analysis
2. PROs and CONs of Open Source

- The Linux hype cycle
- CONs
  - PROs
  - Conclusion
IT COMMUNITY NOW CONSIDERS OPEN SOURCE AS A STRONG AND RELIABLE TECHNOLOGY

**Stability:** Linux patches don’t generally require application reconfiguration or testing. Windows service packs, in contrast, often change everything from licensing terms to API internals.

**Continuity:** As the last 2 decades ups and downs show, most closed-source vendors don’t seem to be able to maintain continuity over time, forcing customers to change platform if they go bankrupt. While accessing to the source code gives customers the opportunity to develop its own software even if external vendor disappears.

**Security:** Only little security issues with Open Source. They were quickly remedied. The Microsoft toolset, on the other hand, has dozens of known outstanding security issues. Remediation is usually slow in coming. A large Open Source community keeps solutions under continuous control.

**Interoperability:** Enabling different systems to work together and exchange data. Interoperability between different systems is achieved by using common standards and specifications. This is easy when systems rely on open standards but becomes hard and expensive if proprietary norms.

**Flexibility:** Using Open Source software for any solution gives complete flexibility to meet users’ specific needs. Large proprietary vendors are not going to change their software and hardware for 1 customer, but using Open Source software gives users the freedom to customize.

**Technical benefits of Open Source**

* Actually, Linux is far less exposed to threats than Windows, also because it has quite a higher entry barrier for system administrators

** However, we should not underestimate the power of Visual Basic for application; it can get the work done quickly even if several developers consider its output as “dirty” from a programmer point of view

Source: Paul Murphy, *The pros and cons of business-app implementation via open-source software*; CETIS; builder.com; pcxperience.com
HIGHER STABILITY ALSO MEANS LOWER COSTS OF DOWNTIME

Assumptions
• 240 working days/year
• 8h/day
• Only downtime during working time considered
• Single server configuration (as opposed to cluster)
• Cost estimates only cover IT admin cost of downtime, not loss of revenue
• Fully burdened cost of labor (USD/hour): 28$
• Company size: 10 employees

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Average Downtime (Hours/year)</th>
<th>Downtime Cost (USD/year)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux Debian</td>
<td>4</td>
<td>1,075</td>
</tr>
<tr>
<td>Windows NT</td>
<td>29</td>
<td>8,060</td>
</tr>
</tbody>
</table>

Source: Robert Frances Group, 2002 ; heise.de ; dwheeler.com ; SaS analysis
DRAMATIC REDUCTION OF LOCK-IN EFFECTS EMPOWERS SOFTWARE USERS

Main disadvantages of being linked to a vendor

- Upgrades and new versions are not always initiated by users’ actual needs
- New versions sometimes require buying new hardware
- Because of no access to code, if vendors disappear, users can't improve/fix the software
- Users have little choice in case of tariff modification
- Being locked in lead to additional problems and expenses in case of migration

By relying on widely available source code, open source software users are not locked into one vendor for their solutions

Source: Statskontoret; SaS analysis
USING OPEN SOURCE SOLUTIONS GIVES BETTER CONTROL OVER IT ARCHITECTURE AND DEVELOPMENTS

How do you control your IT architecture since you migrated to Open Source?

<table>
<thead>
<tr>
<th></th>
<th>Better</th>
<th>Worst</th>
</tr>
</thead>
<tbody>
<tr>
<td>Large companies</td>
<td>97% 29</td>
<td>3% 3</td>
</tr>
<tr>
<td>SMEs</td>
<td>92%</td>
<td>8% 37</td>
</tr>
</tbody>
</table>

Source: interviews; SaS analysis
EARLY ADOPTERS OF INNOVATIVE SOLUTIONS USUALLY BENEFIT FROM INCREASED MEDIA VISIBILITY…

Press visibility index*

<table>
<thead>
<tr>
<th>Group</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pioneers</td>
<td>0.41</td>
</tr>
<tr>
<td>Early adopters</td>
<td>1.00</td>
</tr>
<tr>
<td>Followers</td>
<td>0.64</td>
</tr>
</tbody>
</table>

Online press has been devoting much more articles to early adopters than to followers

* Calculated as an index of press clippings related to the adoption of the innovative technology in the semi-specialized online press. Timing of “pioneers”, “early adopters”, “followers” depends on industry

Source: interviews; press clippings; SaS analysis
...WHILE MORE EASILY RETAINING THEIR IT STAFF

Churn rate index among IT staff* in 3 different categories of medium and large companies

<table>
<thead>
<tr>
<th>Category</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Pioneers&quot;</td>
<td>0.89</td>
</tr>
<tr>
<td>&quot;Early adopters&quot;</td>
<td>1.00</td>
</tr>
<tr>
<td>&quot;Followers&quot;</td>
<td>1.21</td>
</tr>
</tbody>
</table>

* Assumption: 1998-1999 (Pioneers), 2002 (Early adopters) and in 2004 (Followers)

Source: interviews; press clippings; SaS analysis
IN THE OS AND WEB SERVER SEGMENTS, OPEN SOURCE WINS THE HARD-COST COMPARISON

USD

Assumptions

- Simulation over 3 years
- Windows and Linux on Intel x-86 architecture
- MS Internet Information Server on Windows NT
- Apache on Debian Linux
- Sun SPARC with Apache
- Calculation per “Processing Unit”, i.e. number of servers required to process 100,000 hits per day
  - Windows: 7.6 servers/PU
  - Linux: 7.4 servers/PU
  - Solaris: 2.2 servers/PU

Total Cost of Ownership*

<table>
<thead>
<tr>
<th>Software purchase</th>
<th>Hardware purchase</th>
<th>System support &amp; administration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solaris</td>
<td>27,500</td>
<td>146,454</td>
</tr>
<tr>
<td>Windows</td>
<td>39,042</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td>38,015</td>
<td></td>
</tr>
<tr>
<td></td>
<td>36,060</td>
<td>74,475</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>143,640</td>
<td>190,662</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>387,566</td>
<td>561,520</td>
</tr>
</tbody>
</table>

* Does not take into consideration cost of training, cost of lost productivity during learning curve, savings in security, costs/savings of system availability

Source: Robert Frances Group, 2002
AND IS ALSO HIGHLY COMPETITIVE ON THE HR COSTS

Despite higher salary cost of Linux system administrators, administration cost of a Linux server is 4 times lower than the same running Windows.

Source: Robert Frances Group, 2002; SaS analysis
TCO OF OPEN SOURCE SOLUTIONS SECURES A VERY APPEALING ROI REGARDLESS THE SCALE

Assumptions

• Costs considered:
  – Downtime
  – Software license, maintenance and support
  – Training
  – Hardware
  – Discounts if purchase of high volume
  – System administrators
• Depreciation of hardware: 3 years
• Length of analysis: 3 years
• Cost of capital: 12%

* ROI = savings / cost to deploy
Source: Robert Frances Group, 2002; linuxworld.sys-con.com; SaS analysis
POTENTIAL IMPACT OF OPEN SOURCE ON A DESKTOP PC'S TCO – DEPRECIATED COST OVER THREE YEARS
USD, per year, per user

Assumptions for a 5,000-users organization with a three-year lifecycle

- Lower hardware requirements
- Minimal license cost of upgrades
- Improved reliability
- Adaptation to new applications

Procurement administration
Purchase
Hardware installation
Hardware upgrades and repair parts
Software upgrades and repair parts
Labor hardware upgrades and repair parts
Labor software upgrades and repair parts
OS
Office suite**
Total savings
Software installation
Training (IT and end users)
Lost end user productivity
Total impact

10
50
35
10
15
890
200
20
25
355
490
50
370

* Windows XP Professional, with Microsoft's 2004 prices
** Office 2003 with Microsoft’s 2004 prices

Source: Meta Group 2002; Gartner; IDC; dwheeler.com; McKinsey analysis; SaS analysis
POTENTIAL IMPACT OF OPEN SOURCE SOLUTIONS ON E-MAIL / EXCHANGE SERVERS* TCO**
USD, per year, per user

Assuming a three-year lifecycle for servers

* Including helpdesk support, assuming 7,000 users in one site, plus 10 sites of 50 people using WAN links to access mails servers at headquarters. Assuming only basic functionalities and no advanced groupware services are used, otherwise economics change significantly.

** Often times, the key feature of why companies choose exchange servers over normal mail servers is the advanced groupware services. A complete and consistent comparison should address this functionality, examining some open source solutions that include groupware, such as kolab, suse's solution, by nari, etc.

Source: Meta Group 2002; McKinsey analysis; SaS analysis
LOSS OF OPEN SOURCE DESKTOP USERS’ PRODUCTIVITY DRAMATICALLY DECREASES OVER TIME

Rationale

- Sophistication and refinement of graphical user interfaces (GUIs) of Open Source applications is making the learning curve easier
- Open Source GUIs are perceived as more familiar by end users
- End users spend less time and effort adapting their know-how to new environment
- Viral training amongst colleagues and friends is more and more common

Loss of productivity of desktop users due to migration to Open Source*
USD, per year, per user

* Assumption: 5,000-user organization with a three-year lifecycle; depreciated over 3 years
Source: Meta Group 2002; interviews; SaS analysis
2. PROs and CONs of Open Source

• The Linux hype cycle

• CONs

• PROs

• Conclusion
GLOBAL COMPARISON HIGHLIGHTS REAL ADVANTAGE TO OPEN SOURCE

According to several experts, Open Source software can be considered as a strong and viable alternative to traditional IT offering.

Source: interviews; SaS analysis
“The arguments for and against open-source software get trivialized. It's not a technology issue; it's a business issue.”

Andy Mulholland (CTO), Cap Gemini Ernst & Young
3. Is Open Source really changing the software industry?
3. Is Open Source really changing the software industry?

• Reminder: brief definition

• The Open Source market

• A new business model

• Is there a future for Open Source?
OPEN SOURCE SOFTWARE PHENOMENON

WHAT IT IS

• Linux is the most famous open source solution but there are many other open software languages and solutions (e.g. Apache, PHP, OpenOffice, MySQL, etc.)

• Companies can make money with open source despite its being free, by providing related services and products (i.e. Red Hat, Alcove, All2All, etc.)

• Open source software versions are developed in order to be compatible with the existing previous versions, as is the case for most proprietary software

• An open source software is a software delivered with unrestricted access to its source code*
• Its license generally grants the right to run the program, own a copy of the program’s source code, modify and distribute a modified version under the same terms as the original**

WHAT IT IS NOT

• Open source is not only the work of high school and college students

• Open source providers are not only fly-by-night companies that won't last: ex. SUN and IBM provide Linux

• An open source software is not necessarily less safe than a proprietary one

• Open source software are not necessarily less reliable than private software since each developer who contributes to its development periodically reviews and tests others' job at the source code level

* A source code is a list of instructions in a programming language of a software
** Unless explicitly specified in the software license

Source: McKinsey analysis; Press; Interviews; Web sites; SaS analysis
A DEFINITION OF OPEN SOURCE SOFTWARE

For free:
- Everybody can execute it
- Everybody can modify the source code
- Everybody can distribute it
- Everybody can distribute a modified version of the source code

<table>
<thead>
<tr>
<th></th>
<th>Private software</th>
<th>Freeware (free private software)</th>
<th>Open source software</th>
</tr>
</thead>
<tbody>
<tr>
<td>Everybody can execute</td>
<td>✗</td>
<td>✅</td>
<td>✅</td>
</tr>
<tr>
<td>Everybody can modify</td>
<td>✗</td>
<td>✗</td>
<td>✅</td>
</tr>
<tr>
<td>Everybody can distribute</td>
<td>✗</td>
<td>✅</td>
<td>✅</td>
</tr>
<tr>
<td>Everybody can distribute a modified version of the source code</td>
<td>✗</td>
<td>✗</td>
<td>✅</td>
</tr>
</tbody>
</table>

Open source software is characterized by the ability to acquire a software freely, get access to its source code, modify it and distribute a modified version of it.

Source: McKinsey analysis; Press Clippings; Open Source-focused web sites
OPEN SOURCE TECHNOLOGIES MAKE INTERNET & INTRANET PORTALS MORE EFFECTIVE THAN OTHERS

Why have several companies recently preferred to build their Internet and/or Intranet portal on Open Source technologies?

<table>
<thead>
<tr>
<th>Branded software</th>
<th>Open Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>• More expensive</td>
<td>• Cost effective</td>
</tr>
<tr>
<td>• Black box (lack of transparency)</td>
<td>• Transparent (not a “black box”)</td>
</tr>
<tr>
<td>• Only compliant with their own technological landscape</td>
<td>• Strongly supported by several Governments</td>
</tr>
<tr>
<td>• Fully dependent of vendors</td>
<td>• Compliant with main technology standards</td>
</tr>
<tr>
<td>• Not always modular</td>
<td>• Independent from vendors</td>
</tr>
<tr>
<td>• Hard to say if secure</td>
<td>• Modular</td>
</tr>
<tr>
<td></td>
<td>• Reliable</td>
</tr>
<tr>
<td></td>
<td>• Secure</td>
</tr>
<tr>
<td></td>
<td>• Usually appreciated by IT professionals</td>
</tr>
</tbody>
</table>

Source: online press clippings; SaS analysis
3. Is Open Source really changing the software industry?

• Reminder: brief definition

• The Open Source market

• A new business model

• Is there a future for Open Source?
SITUATION OF SUPPLY AND DEMAND

DEMAND

Demand exists and is growing quite fast, even if it is still somehow restricted to a number of IT domains

• Companies are increasingly interested but remain cautious when selecting solutions

• IT managers pay more and more attention to new open source solutions

• Some strong and reliable solutions have clearly emerged (e.g. Linux, Apache, MySQL), but users are still reluctant to use open source software in other domains (e.g. office suites)

• Companies only start now to really support programmers’ communities and projects

SUPPLY

Supply is potentially rich but fragile since many key IT players and stakeholders perceive it as a threat

• Many traditional software vendors are opposed to open source software, which is likely to compromise their current business model

• Some major IT players have developed specific offerings around open source software

• New entrants have emerged, but open source market is not yet consolidated

• Pro-open source users able to influence the expansion of supply and demand (e.g. Governments) have just started taking concrete decisions

Will there be an open source software market?

Source: McKinsey analysis; SaS analysis
SITUATION OF SUPPLY AND DEMAND

DEMAND

Demand exists and is growing quite fast, even if it is still somehow restricted to a number of IT domains

• Companies are increasingly interested but remain cautious when selecting solutions

• IT managers pay more and more attention to new open source solutions

• Some strong and reliable solutions have clearly emerged (e.g. Linux, Apache, MySQL), but users are still reluctant to use open source software in other domains (e.g. office suites)

• Companies only start now to really support programmers’ communities and projects

SUPPLY

Supply is potentially rich but fragile since many key IT players and stakeholders perceive it as a threat

• Many traditional software vendors are opposed to open source software, which is likely to compromise their current business model

• Some major IT players have developed specific offerings around open source software

• New entrants have emerged, but open source market is not yet consolidated

• Pro-open source users able to influence the expansion of supply and demand (e.g. Governments) have just started taking concrete decisions

Will there be an open source software market?

Source: McKinsey analysis; SaS analysis
PERCEIVED BENEFITS OF OPEN SOURCE SOLUTIONS

Description of perceived benefits

**Technical benefits**

- **Quality**: stability-performance
- **Security**:  
  - Reduced number of bugs  
  - Reduced number of "back doors"
- **Continuity**: source code always available for updates/ adaptations
- **Interoperability/flexibility**: compatibility and customization facilitated

**Independence from IT suppliers**

- **Impact on suppliers**
- **Control over IT architecture and developments**

**Improvement of image**

- **Externally**:  
  - Opportunity for a new offering positioning  
  - Marketing upside as open source is associated to innovation and programmers' community  
  - HR communication advantage in the war for talent
- **Internally**:  
  - Motivation of IT employees through new innovative projects

**Financial benefits**

- **Global cost analysis**: (user's point of view, and depends on context)  
  - Absence of acquisition and licensing cost  
  - Lower hardware requirements to run applications*  
  - No more accounting and management of licenses  
  - Reduction of external corrective maintenance interventions  
  - Increased services needs  
    - Integration  
    - Development  
    - Training

* This is mostly true for GNU/Linux

Source: McKinsey analysis ; interviews ; Benchmark Group ; Forrester
Several large companies were confident enough to migrate from their usual closed source software to open source software.

<table>
<thead>
<tr>
<th>Company</th>
<th>Description</th>
<th>Trigger for opensource usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hollywood graphic and special effects companies</td>
<td>• Graphics studios</td>
<td>• TCO, Reliability</td>
</tr>
<tr>
<td>BMW</td>
<td>• Corporate website, Dealers’ mini-sites and second-hand car site</td>
<td>• Reliability, Security</td>
</tr>
<tr>
<td>GlaxoSmithKline</td>
<td>• Corporate portal</td>
<td>• Security, TCO</td>
</tr>
<tr>
<td>International Press</td>
<td>• Belgian Prime Minister online press center (press releases, personalisation, multi-channel distribution)</td>
<td>• Security, reliability</td>
</tr>
<tr>
<td>Eurochamber</td>
<td>• Official European portal for the promotion of Japan/Europe trading. Both English and Japanese (double byte) languages</td>
<td>• Security, Reliability, TCO</td>
</tr>
<tr>
<td>Wireless Developer Network</td>
<td>• Large information portal devoted to news &amp; technical publications for wireless application developers and architects of the mobile Internet</td>
<td>• TCO, Reliability</td>
</tr>
<tr>
<td>McDonald's Germany</td>
<td>• Fast Food business deploys Linux for DNS, FTP, and proxy services on the Internet</td>
<td>• Performance, stability, flexibility</td>
</tr>
<tr>
<td>NYSE</td>
<td>• Stock exchange</td>
<td>• Scalability, Security, Reliability</td>
</tr>
</tbody>
</table>
## PROOF OF OPEN SOURCE AS A TECHNICALLY Viable SOLUTION (2/2)

<table>
<thead>
<tr>
<th>Company</th>
<th>Description</th>
<th>Trigger for opensource usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google</td>
<td>Largest search engine</td>
<td>TCO, Parallel processing</td>
</tr>
<tr>
<td>Ameritrade</td>
<td>Online broker</td>
<td>Security, Scalability</td>
</tr>
<tr>
<td>Apple / Macintosh</td>
<td>OS X (freeBSD)</td>
<td>Reliability</td>
</tr>
<tr>
<td>Orbitz</td>
<td>Online travel agency</td>
<td>TCO</td>
</tr>
<tr>
<td>Weather.com</td>
<td>Largest weather forecast website</td>
<td>TCO, Processing capacity</td>
</tr>
<tr>
<td>The Ogilvy Group</td>
<td>One of the worldwide largest advertising agency</td>
<td>Control over its own IT</td>
</tr>
</tbody>
</table>

Source: corporate web sites; Press clippings
As an online travel agency, Orbitz uses its IT to download fares, service search requests, run the company booking engine, and make all the services available online.

2003

- A total of 168 Sparc processors were used to support web applications
- Migration of web services to Linux
- Under Linux, 100 Intel processors now deliver twice the performance for one-tenth the cost

Thanks to Open Source, business users report paying less and getting more.

Source: Press clippings
**USERS’ SATISFACTION IS LIKELY TO FOSTER A VIRAL EFFECT OF ADOPTION**

<table>
<thead>
<tr>
<th>Are you satisfied with the Open Source solution you adopted in your company?*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Satisfied</td>
</tr>
<tr>
<td>95%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Have you already recommended your IT friends to adopt similar Open Source solutions in their own companies?*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
</tr>
<tr>
<td>98%</td>
</tr>
</tbody>
</table>

- Most of the IT managers are satisfied or very satisfied with the Open Source solutions they have deployed in their companies.

- Satisfaction clearly makes IT managers viral “evangelists” of Open Source solutions.

* Non scientific survey. N=42 IT managers among which 6 from large companies and 36 from SMEs.

Source: interviews; SaS analysis
OPEN SOURCE SOFTWARE MARKET SHARE

Percentage

Market share

- Web servers
- Operating systems
- Database servers
- CMS/Portal Servers
- Client operating systems

Infrastructure applications like Linux, Apache, MySQL and PostgreSQL are the main growth levers of open source software

Source: McKinsey analysis; Forrester; IDC; Ovum; interviews; netcraft; SaS analysis
## PERCEIVED BARRIERS TO ENTERPRISE ADOPTION

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>&quot;Provider&quot; reliability</strong></td>
</tr>
<tr>
<td>• Availability of services around open source is increasing but traditional systems integrators still have limited serious offerings and skills</td>
</tr>
<tr>
<td>• Enterprise customers doubt open source communities’ ability to adapt solutions to changing market requirements for business applications (e.g. HR regulatory requirements, B2B interactions and standards) even if open source service providers are growing on the market</td>
</tr>
<tr>
<td><strong>Migration Support</strong></td>
</tr>
<tr>
<td>• Some open source players do not have yet an extended experience with customers’ current IT environment</td>
</tr>
<tr>
<td>• Only a limited number of open source service providers also position themselves as migration consultants committed to reducing the risk of leaving their clients to perform a migration unaided</td>
</tr>
<tr>
<td><strong>Fear, Uncertainty and Doubt</strong></td>
</tr>
<tr>
<td>• Established software companies communicate heavily on the perceived risks of using open source solutions</td>
</tr>
<tr>
<td>• Companies are still unsure about the level of provider maturity and their experience with enterprise issues</td>
</tr>
</tbody>
</table>

**Source:** McKinsey analysis; SaS analysis

Barriers will mainly be removed by enterprises organizing demand and clarifying expectations to ensure quality services are available.
RISK ANALYSIS OF OPEN SOURCE

Risks of using open source

- **Lack of expertise and experience** may compromise successful implementations and maintenance of open source applications
  - Internally, it may be difficult to gather the required level of expertise
  - Externally, services providers may not have the required industry knowledge and/or critical mass to manage large-scale implementations of open source software

- **Long-term viability of solutions** is not yet guaranteed, except for infrastructure. Several applications are still under development or testing

- Implementing open source solutions may generate unexpected needs for training, thus increase the solution’s TCO

Risks of not using open source

- Get higher IT costs than competitors, which have potentially
  - Reduced their license costs through open source solutions
  - Reduced the cost of internal applications by collaborating with other industry players

- Be dependent on some IT vendors’ technologies and pricing policies thus
  - Lose control of IT purchasing (e.g. new pricing policies of Oracle and Microsoft)
  - Lose control of company’s technical architecture
  - Be “locked up” by proprietary standards (e.g. governments)

- Loose top performing IT people who wish to work with open source solutions

Excluding open source solutions is highly risky for companies, even though they should be careful in planning the IT migration

Source: McKinsey analysis
SITUATION OF SUPPLY AND DEMAND

DEMAND

Demand exists and is growing quite fast, even if it is still somehow restricted to a number of IT domains

• Companies are increasingly interested but remain cautious when selecting solutions

• IT managers pay more and more attention to new open source solutions

• Some strong and reliable solutions have clearly emerged (e.g. Linux, Apache, MySQL), but users are still reluctant to use open source software in other domains (e.g. office suites)

• Companies only start now to really support programmers’ communities and projects

SUPPLY

Supply is potentially rich but fragile since many key IT players and stakeholders perceive it as a threat

• Many traditional software vendors are opposed to open source software, which is likely to compromise their current business model

• Some major IT players have developed specific offerings around open source software

• New entrants have emerged, but open source market is not yet consolidated

• Pro-open source users able to influence the expansion of supply and demand (e.g. Governments) have just started taking concrete decisions

Will there be an open source software market?

Source: McKinsey analysis; SaS analysis
Incumbents perceive open source as a major threat to their business model, since it would:

- Drastically reduce their main revenue stream (license fees)
- Require a shift to a services-centric activity

**Applications providers**

- SAP: 30% License revenue, 70% Services revenue (100% = 7,800 m)
- Siebel: 35% License revenue, 65% Services revenue (100% = 1,600 m)

**Infrastructure providers**

- Oracle: 45% License revenue, 55% Services revenue (100% = 9,500 m)
- BEA: 43% License revenue, 57% Services revenue (100% = 934 m)
- CA: 12% License revenue, 88% Services revenue (100% = 3,120 m)
- Microsoft: 7% License revenue, 93% Services revenue (100% = 32,187 m)
- TIBCO: 47% License revenue, 53% Services revenue (100% = 264 m)

Source: McKinsey analysis; Yahoo finance; annual reports; corporate websites; SaS analysis
Editors have no interest in switching to a services business model since

• The operational margin is significantly lower

LICENCES AND SERVICES OPERATING MARGIN COMPARISON

Operating margins reachable – by activity

<table>
<thead>
<tr>
<th>Operating margin, Percent</th>
<th>Costs</th>
<th>Operating margins</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td></td>
<td>12%</td>
</tr>
<tr>
<td>65%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>35%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Operating margins of market leaders* on their software activities

Operating margins of market leaders** on their services activities

Source: McKinsey analysis; IDC; annual reports; Cherry Tree
### DESCRIPTION OF NEW ENTRANTS' BUSINESS MODELS

<table>
<thead>
<tr>
<th>Role</th>
<th>Typical revenue split</th>
<th>Products</th>
<th>Services</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integrator</td>
<td>• 100% services</td>
<td>• None: no release of branded products but use of others' certified versions</td>
<td>• Customized services: integration and specific complex developments, carried out in a project mode with consulting</td>
<td>• Alcôve, Ideal X</td>
</tr>
<tr>
<td>Editor and services</td>
<td>• 20% products</td>
<td>• Certified branded versions of Linux and other open source applications (i.e: PostgreSQL)</td>
<td>• Industrialized services: training classes, call center for support, implementation assistance, and consulting</td>
<td>• Red Hat, SUSE</td>
</tr>
<tr>
<td>Editor</td>
<td>• 85% products</td>
<td>• Certified branded versions of Linux and other open source applications (i.e: drivers)</td>
<td>• Light support and training</td>
<td>• Mandrake</td>
</tr>
</tbody>
</table>

Source: McKinsey analysis; interviews

NOT EXHAUSTIVE
## STAKEHOLDERS ABLE TO FOSTER THE DEVELOPMENT OF OPEN SOURCE (1/2)

<table>
<thead>
<tr>
<th>Type of player</th>
<th>Center of interest</th>
<th>Rationale</th>
</tr>
</thead>
<tbody>
<tr>
<td>States &amp; Administrations</td>
<td>• Low/no cost of license</td>
<td>• Open source software is a mean to gain political independence from any foreign and private software economy and foster a local software industry</td>
</tr>
<tr>
<td></td>
<td>• Independence vis-à-vis editors</td>
<td></td>
</tr>
<tr>
<td>Corporations</td>
<td>• Low/no cost of license</td>
<td>• Corporations increasingly seek to limit dependence on suppliers with a dominant market position and improve negotiation positions</td>
</tr>
<tr>
<td></td>
<td>• Freedom of supplier</td>
<td>• Open source enables the co-development of applications, thus the sharing of investments required</td>
</tr>
<tr>
<td></td>
<td>• Collaborative development of B2B / professional applications</td>
<td>• This is particularly relevant for B2B applications in very connected industries like banks or telecoms</td>
</tr>
</tbody>
</table>

- Think tanks are created to consider the feasibility of broadly moving to open source
- Many pilot projects are already launched (French Ministry of Culture, Dexia, Dresdner Bank, Bavaria Landers)
- Several projects are already in an advanced stage

Source: McKinsey analysis; SaS analysis
## STAKEHOLDERS ABLE TO FOSTER THE DEVELOPMENT OF OPEN SOURCE (2/2)

<table>
<thead>
<tr>
<th>Type of player</th>
<th>Center of interest</th>
<th>Rationale</th>
</tr>
</thead>
</table>
| ASPs & Linux services                  | • Leverage of open source to cancel license fees and lower cost of delivering services | • ASPs' margins are drastically eroded by the license fees paid to software editors  
• Open source is a means to develop vertical solutions and add-ons with limited investments in order to improve the value proposition of services |
| Electronics / Hardware manufacturers   | • Development of embedded software components                                         | • Software is not a core activity but is a major cost element to electronics manufacturers which could decrease development and per unit costs  
• Use of Linux  
• Creation of open source industry standards |                                                                                                                                                                  |
| Software laggards                      | • Shift to services activities                                                       | • Some hardware manufacturers like Dell leveraged Linux to penetrate the server entry market segment; IBM launched a new server strategy based on Linux operating system  
• As software takes an increasing role in some industries, open source is viewed as a means to avoid market dominance by a software vendor (e.g. mobile telecom, mobile gaming)  
• Vendors with an unsustainably low market share could leverage open source to shift to a services activity. Not being leaders means that margins of a services activity may be higher than their current software margins |

- Some players have already taken action (e.g. IBM, Nokia, Open cascade)  
- In spite of numerous announcements, concrete results start emerging only now

Source: McKinsey analysis; SaS analysis
**KEY FACTORS OF SUCCESS FOR A SOFTWARE LAGGARD SWITCHING TO OPEN SOURCE**

<table>
<thead>
<tr>
<th>Factor of success</th>
<th>Rationale</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Build value proposition on deep industry expertise</strong></td>
<td>• Successfully switching to a services model requires the provider to deliver added value beyond source code. Consequently, software companies willing to do so should target niche services markets requiring specific technical or industry knowledge</td>
</tr>
<tr>
<td><strong>“Outsource” R&amp;D and sales costs to community</strong></td>
<td>• By releasing the source code, the company relies on the contributions of community members to develop new features and upgrades. This opportunity to reduce the R&amp;D costs by sharing the costs of development is key to success and requires the company to set up and organize a programmers’ community • Companies must also &quot;evangelize&quot; the community so that it serves as a sales force and contributes to cost reductions</td>
</tr>
<tr>
<td><strong>Ensure continuous upgrade of expertise</strong></td>
<td>• Companies build their services offering on the expertise they have of the application. To preserve this competitive advantage, companies should ensure that all consultants are trained on new features developed by community members. It is also crucial to avoid “forking” of versions, which would generate a loss of expertise and thus decrease the value proposition of the services offering</td>
</tr>
</tbody>
</table>

*Unfortunately, niche markets have niche communities that may not be big enough to provide the usual benefits of Open Source software development model*

Source: McKinsey analysis; SaS analysis
The impact of open source on the value chain will mainly depend on
- The ability of end users to structure demand and participate in open source development projects
- The resistance to change of software vendors
- Support from pro-open source IT players not directly involved in the software segment

* All electronic systems included in consumer goods

Source: McKinsey analysis; SaS analysis
3. Is Open Source really changing the software industry?

- Reminder: brief definition
- The Open Source market
- A new business model
- Is there a future for Open Source?
WITH OPEN SOURCE, THE CONCEPT OF SOFTWARE VENDORS TENDS TO DISAPPEAR

Business sustainability is no longer based on license fees or royalties

Brains, skills, competencies become more relevant than product packaging

Software is oftentimes provided by a community not a vendor

Vendors are replaced by service and support providers

Source: All2All; SaS analysis
ROBUST BUT UNEXPENSIVE SOLUTIONS SECURE A REAL COMPETITIVE ADVANTAGE

- Robust
- Secure
- Standard
- Scalable
- Etc.

Source: SaS analysis
Shift in revenue streams triggers deep changes in the “culture” of business

Revenue streams of providers shift from licenses to services, trainings, customization, consulting, etc. This triggers an additional series of changes in the way of conceiving and doing the business.

<table>
<thead>
<tr>
<th>Cultural changes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Providers</strong></td>
</tr>
<tr>
<td>• Transformation in legal framework of the business</td>
</tr>
<tr>
<td>• New approach of product development</td>
</tr>
<tr>
<td>• Involvement in community-building relationships for software development and support</td>
</tr>
<tr>
<td>• Change in competition</td>
</tr>
<tr>
<td><strong>Users</strong></td>
</tr>
<tr>
<td>• Redistribution of cost items and apparition of un-usual follow-up costs</td>
</tr>
<tr>
<td>• More collaborative workflow with service providers</td>
</tr>
<tr>
<td>• Redefinition of KSFs and KPIs to assess both solutions and “vendors”</td>
</tr>
<tr>
<td>• Direct participation in software development</td>
</tr>
</tbody>
</table>

Source: All2All; SaS analysis
FOR PROVIDERS, OPEN SOURCE IS AN UNKNOWN MARKET WHERE THEY STILL HAVE TO FIND THEIR WAY…

The processes and procedures through which things get done from day-to-day

Capabilities possessed by the organization as a whole as distinct from the individuals

Service offering, package, place, pricing, positioning

From copyrights to copyleft

Capabilities possessed by the organization as a whole as distinct from the individuals

Staff

Processes

Skills

R&D

Legal

Products

Competitors

Marketing

Products

R&D

Processes

Staff

The way managers collectively behave with respect to use of time, attention and symbolic actions

Decentralized community of developers

The people in the organization, considered in terms of corporate demographics, not individual personalities

Source: All2All; SaS analysis
WHILE FOR USERS OPEN SOURCE MEANS A DEEP TRANSFORMATION OF IT STRATEGY

All components of IT strategy are impacted by introduction of Open Source in the organization.
THIS NEW MARKET GOES ALONG WITH NEW RELATIONSHIP BETWEEN USER AND PROVIDER

<table>
<thead>
<tr>
<th>How do users usually behave?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Closed Source paradigm</strong></td>
</tr>
<tr>
<td>• Users’ role is usually passive (buy, install, configure, use)</td>
</tr>
<tr>
<td>• Software support requires payment</td>
</tr>
<tr>
<td>• Bug reports don’t trigger immediate response; often times they even don’t trigger any response at all</td>
</tr>
<tr>
<td>• User can become part of the development effort</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

Source: All2All; SaS analysis
SEVERAL DIFFERENT BUSINESS MODELS ARE THEREFORE POSSIBLE

- **Software distribution**: Sales of packaged products based on Open Source software (ex. SuSe)
- **Development and sales**: Sales of in house developed software
- **Added value sales**: Open Source software is used to support sales of other hardware and software products
- **Service provider**: Sales of services, trainings, consulting, maintenance, support, etc.
- **Accessories**: Publication, edition, sales of specialized literature

*Source: Statskontoret*
3. Is Open Source really changing the software industry?

- Reminder: brief definition
- The Open Source market
- A new business model
- Is there a future for Open Source?
POTENTIAL SCENARIOS OF EVOLUTION FOR OPEN SOURCE SOFTWARE

Will open source succeed in penetrating the software market?

**YES**

- Major impact on the IT value chain

  - Key IT players support some open source projects and ensure that expansion of open source is limited to the segments of the value chain related to these projects
  - Stable open source solutions are available for niche, vertical or commodity sub-segments
  - Services players develop expertise and experience on these solutions

**NO**

- Major impact on sub-segments of the value chain

  - IT players manage to block the expansion of open source software outside the infrastructure solutions segment

---

Source: McKinsey analysis; SaS analysis
## KEY REQUIREMENTS FOR EACH SCENARIO

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Key requirements</th>
<th>Status</th>
</tr>
</thead>
</table>
| Major impact on the value chain        | • Demand-side succeeds in structuring their requirements, forcing suppliers to adopt open source solutions in their offerings. The key means to initiate this change is through participation of corporate resources in open source communities and solution development  
• Corporations and governments come together to promote open source solutions and share success stories and results  
• Complete offerings based on open source emerge, covering the entire value chain and specifically major service providers  
• Open source solutions achieve over 20% market share in several business segments with strong public success stories | • This scenario has been emerging since a couple of years                                                                   |
| Impact limited to a few sub-segments of the value chain | • Supply-side tolerates the development of open source applications in limited market sub-segments that are non-core to their offerings.  
• Open source initiatives are focused on niche applications, vertical/highly specialized applications or commodity functionality that has limited impact on major software providers business models  
• End customers use open source as any other kind of software, just cheaper, without participating in open source communities  
• Some IT services companies emerge to address these new needs, but major players remain passive since the critical mass justifying a major reaction is not reached | • This scenario is very well established in the industry                                                                       |
| No penetration on the software market   | • Traditional software providers and open source players move into direct market conflict  
• Major software providers actively prevent open source initiatives from succeeding (e.g. Acquisition, guerrilla marketing, etc)  
• Few success stories are published  
• IT players not directly in the software business and corporations do not allocate resources to open source initiatives and only use solutions passively as a marketing tool | • This scenario appears as totally outdated                                                                              |

Source: McKinsey analysis; SaaS analysis
### POSSIBLE ACTIONS LIKELY TO FOSTER THE USAGE OF OPEN SOURCE SOFTWARE

<table>
<thead>
<tr>
<th>Drivers</th>
<th>Actions to take</th>
<th>Consequences</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>States as users</strong></td>
<td>• European states massively migrate to open source solutions, even for their core applications</td>
<td>• Services companies develop/consolidate their open source skills to serve this demand, accelerating market maturity and availability of information</td>
</tr>
<tr>
<td></td>
<td>• European states push open source forward in companies where they are a significant shareholder</td>
<td>• «Para-public» companies start using open source solutions, but first only for their non-core applications</td>
</tr>
<tr>
<td></td>
<td>• China, Korea, India are creating large common open source projects</td>
<td>• Many companies now have enough elements and benchmarks to implement open source solutions</td>
</tr>
<tr>
<td><strong>States as regulators</strong></td>
<td>• European states set up new laws with incentives for the use of open source solutions</td>
<td>• Local software markets are boosted</td>
</tr>
<tr>
<td></td>
<td>– Tax decrease</td>
<td>• Companies in a «wait and see» attitudes decide to use open source solutions for non-critical applications first</td>
</tr>
<tr>
<td></td>
<td>– Subsidized research projects</td>
<td>• Consequently, demand is increasing, followed by the development of a broader services offering and a more mature market</td>
</tr>
<tr>
<td></td>
<td>– Privileges given to open source solutions for the attribution of public markets</td>
<td>• Younger generation is growing up as open source users in schools and universities</td>
</tr>
<tr>
<td></td>
<td>• China banned Microsoft from administration</td>
<td><strong>End users communities</strong></td>
</tr>
<tr>
<td><strong>End users communities</strong></td>
<td>• Coordinated actions against dominant IT vendors to avoid dependence</td>
<td>• Large corporations produce vertical business applications that are open source, forcing IT players to align with open source solutions</td>
</tr>
<tr>
<td></td>
<td>• Large corporations collaborate to develop business applications which become category killers and/or a standard in the industry</td>
<td>• Services activities develop, especially those combining open source skills and industry know-how</td>
</tr>
</tbody>
</table>

Source: McKinsey analysis; SaS analysis
WHAT STRATEGY FOR CORPORATIONS?

- Use open source software as easy and inexpensive solutions for minor/pilot projects
- Identify expertise to build and relationships to develop – invest in HR
- Integrate open source software in the IT strategy
- Evaluate the business case of migrating some elements of the technical infrastructure (e.g. eMail services, LAN servers)
- Collaborate with other industry players to co-develop open B2B applications and common platform

“Whether you like it or not, open source is here to stay”
Michel Bauwens, CEO

Source: McKinsey analysis; SaS
4. Open Source in the company
4. Open Source in the company

- Possible scenarios
  - Datacenter and IT department
  - “Light” server, office and content side
The type of IT segment determines which open source technologies to be used.

Where is Open Source used in the company?

- Open Source in the core datacenter segment
  - Server side
    - File and print servers
    - Fax servers
  - Client side
    - Office suite
    - Image manipulation
    - Accounting systems
  - Content side
    - Content management systems

Examples
- Software infrastructure
- IT department back-office applications

Not exhaustive.

Source: SaS analysis
4. Open Source in the company

• Possible scenarios

• Datacenter and IT department

• “Light” server, office and content side
RELYING ON OPEN SOURCE IN DATACENTER IMPACTS HARDWARE STRATEGY

Personalisation and control
Assembling the hardware in-house allows detailed personalisation and gives IT managers a chance to control every single detail of the IT infrastructure.

Cost effectiveness
Deployment of more cost effective IT infrastructure helps making the difference for competitiveness of products & services offering to the customer.

Independence and responsiveness
Building hardware in-house also allows to be both independent from hardware vendors and more responsive for replacement of damaged parts.

Linux runs on almost any server hardware, from mainframes to cheap self made servers assembled with standard PC spare parts (Sparc, PowerPC, Alpha, ix86, etc.)
IN ADDITION TO HUGE SOFTWARE COST SAVINGS, OPEN SOURCE ALSO HELPS REDUCING HARDWARE COSTS
2000-2003 ; USD thousands

Average annual expenditure of a datacenter*

<table>
<thead>
<tr>
<th></th>
<th>With vendors’ solutions (2000-01)</th>
<th>When shifted to Open Source (2002-03)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software license costs</td>
<td>240</td>
<td>30</td>
</tr>
<tr>
<td>Hardware costs</td>
<td>291</td>
<td>244</td>
</tr>
<tr>
<td>Running costs**</td>
<td>454</td>
<td>473</td>
</tr>
<tr>
<td></td>
<td>985</td>
<td></td>
</tr>
</tbody>
</table>

-87.5% -16.1% +4%

- Dramatic reduction of costs of software licensing because most open source applications are free of charge
- Substantial reduction of costs of hardware due to lower requirements and slightly longer lifecycle
- Running costs quite unchanged***

Source: Interviews; Financial statements; SaS analysis

* Does not include start-up and installation costs. Inflation discounted
** Includes costs of staff
*** Structure of costs of personnel changed over the period: total number of system administrator reduced by one; average salary of system and network administrators increased by ~10%; one new account manager hired
OPEN SOURCE SOFTWARE SATISFIES ALL THE MOST IMPORTANT REQUIREMENTS OF A DATACENTER

Main software requirements of datacenter

- Robust, stable and scalable software environments
- High level of security
- Multi-user systems
- Challenging Service Level Agreements
  - high availability of systems
  - no service interruption when system is updated
- Ease server clustering
- Possibility of expanding networks at low additional costs
- Limited number of points of software failure*
- Reliable complete web ISP management solutions
- Etc.

Possible Open Source software solutions

LAMP platform

- Linux, Apache, MySQL, PHP (sometimes, also Perl, Python)

Apache Jakarta Project

- Tomcat, Apache, Apache Group XML Project, Java

Relational Database Management System

- MySQL, Postgresql (+ admin tools like PhpMyAdmin, Cocoa MySQL)

* For instance, Linux does not require any Graphical User Interface (GUI) when deployed as a server

Source: All2All; Interviews; SaS analysis
**LAMP IS A POPULAR PLATFORM FOR PUBLICATION OF STRUCTURED INFORMATION OVER THE INTERNET**

**Definition**

LAMP is the acronym for a set of software working together as a platform for publication of structured/organised information over the Internet. It stands for:

- Linux: operating system
- Apache: HTTP server
- MySQL: relational database management system
- PHP: scripting language

**Main features**

- Highly popular web development platform mostly used on shared hosting servers able to provide mass client web hosting
- Robust support of virtual servers configuration
- High performance (fast, limited downtime, secure)
- PHP scripting language specially created to develop web applications
- Labor cost effective because web scripting language PHP usually reduces time of development up to 20% compared to similar vendors’ solutions
- Able to support several Open Source content management systems

Source: All2All; SaS analysis
APACHE JAKARTA PROJECT AIMS AT CREATING A LARGE
DEVELOPMENT PLATFORM INTERNET ORIENTED

LAMP platform

Apache Jakarta Project

Relational Database Management System

Definition

Apache Jakarta Project* creates and maintains open source solutions on the Java platform for distribution to the public at no charge. It aims at developing an extended set of software tools working as a web application development platform based on Java. Basically, the Apache Jakarta Project can be considered as a huge development platform including many different software components (portlet managers, page servers, servlet containers, tag libraries, classes, filters, etc).

Main features

• Based on Java coding language and Java classes but this is also a disadvantage as Java is not Open Source
• Allows shared hosting even though at a higher cost because each hosted web site requires its own Java Virtual Machine (JVM) which is quite resource consuming and dramatically limits the number of hosted web sites per server

* Jakarta is a project of The Apache Software Foundation (ASF)
Source: http://jakarta.apache.org/ ; All2All ; SaS analysis
RECENT RELEASES OF OPEN SOURCE RDBMSs STAND OUT THANK TO THEIR VERY HIGH QUALITY

**Definition**

Relational DataBase Management System (RDBMS) is a type of database management system (DBMS) that stores data in the form of related tables: data are organized in tabular files that can be related to each other by a common field (item). Relational databases are powerful because they require few assumptions about how data is related or how it will be extracted from the database.

**Main features**

- MySQL is used in LAMP web application platform
- PostgreSQL is a complete relational database (stored procedures, transactions, foreign keys)
- Both MySQL and PostgreSQL are completely Open Source, easy to deploy and can be configured through web interfaces (such as phpmyadmin or Cocoa MySQL)
- Very good performance, speed, security, flexibility, design
- Reliable even on “overwhelmed” hosts
- Replication facilitates hot backups
- Low TCO

Source: http://www.mysql.com/; http://www.postgresql.org/; All2All; SaS analysis
4. Open Source in the company

• Possible scenarios

• Datacenter and IT department

• “Light” server, office and content side
SAMBA is an Open Source/Free Software suite that provides seamless file and print services to client computers
• Complete Windows CIFS protocol to access Windows shares
• Unix home directories can also be accessed as if they were Windows shares
• Can also be accessed by Linux workstations via a Linux kernel function
• Complete user management and centralized backup possible
• Replace expensive vendors’ server software

Netatalk is a freely-available, kernel level implementation of the AppleTalk Protocol Suite, originally for BSD-derived systems
• Appleshare server allows Macintosh clients to access Unix accounts as if they were Appleshare volumes
• Can be combined with SAMBA on the same machine, the Linnux file server is then available to all platforms

MySQL, PostgreSQL are all Open Source reliable and robust database solutions
• Xindice is a promising XML database, still under beta testing

HylaFAX is an Open Source enterprise system for sending and receiving facsimiles as well as for sending alpha-numeric pages. The software is designed around a client-server architecture

Source: http://www.samba.org/ ; http://netatalk.sourceforge.org/ ; http://www.hylafax.org/ ; All2All ; SaS analysis
There are several Open Source office suites, word processors and spreadsheet software for Linux: OpenOffice, Applixware, Koffice into KDE graphical user interface, Abiword, Gnumeric, etc. OpenOffice.org suite currently is the most mature and it is free of charge

- Spreadsheet, drawing, slide show, word processor
- Support of multilingual interface
- ODBC or JDBC database connector to use a central MySQL or PostgreSQL database as a backend
- Runs also on MS Windows and Mac OS X
- Uses open standards XML as file format

The GIMP is the GNU Image Manipulation Program. It is a freely distributed piece of software for such tasks as photo retouching, image composition and image authoring. It works on many operating systems, in many languages

- GnuCash allows to track bank accounts, stocks, income and expenses. It is based on professional accounting principles to ensure balanced books and accurate reports
- SQL-Ledger is a double entry accounting system. Accounting data is stored in a SQL Server, for the display any text or GUI browser can be used
BUILDING THE WHOLE COMPANY UPON OPEN SOURCE LEADS TO HUGE COSTS SAVINGS
2001, USD millions

IT investment comparison for new player in the liberalized utilities sector, as per initial business plan*

<table>
<thead>
<tr>
<th>Year 1</th>
<th>Year 2</th>
<th>Year 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>3</td>
<td>2.5</td>
</tr>
<tr>
<td>5.7</td>
<td>4.4</td>
<td>4</td>
</tr>
</tbody>
</table>

Up to 4.4 m USD over 3 years could be saved by building the new company upon Open Source technologies

* Includes IT CapEx and OpEx (also cost of staff). Discounted over 3 years
Source: SaS analysis
DOZENS OF HIGHLY PERFORMANT OPEN SOURCE CMSs MEET THE FULL RANGE OF USERS’ EXPECTATIONS

Content Management Systems
are tools people can use to manage the content of their website with no need to write or edit HTML for each and every page

Main features
• Graphical layouts and content are independent each another
  – Graphical layouts are based on templates (HTML, Java, etc.)
  – Content lies in the database system
  – CMS creates webpages by « pasting » a graphical skin onto specific content and generates the final HTML source code
• Management of look & feel is therefore kept apart from management of content

• Market is rich in light integrated Open Source CMSs able to support simple or medium-complexity websites
• They mostly target individual users as well as small and medium enterprises with limited needs in content management
• Better known examples : PhpNuke, Spip, Typo3, Cofax, ezPublish, Midguard, OpenCms, etc.

• Top end Open Source CMS solutions have recently emerged and gained full respect in the community of business users
• They target medium and large companies that need to
  – Manage huge multi-format content
  – Customize in details the management processes of the content themselves
  – Publish onto a diverse set of media
• Main examples are RedHat E-CMS, Jahia*, DAD CMS*

* Jahia and DAD CMS are Open Source packages but users are required to pay a one-shot fee for the use of the tool
Source:  http:// www. smile.fr/ ; http:// www. collaborativesource. org/ ; mailing lists ; SaS analysis
USING OPEN SOURCE CMS FREES FINANCIAL RESOURCES THAT CAN BE INVESTED IN RICHER CONTENT
2003, USD thousands

Services

• Search engine
• Corporate information
• Services
• Real time updated external news
• Topic based files
• Personalization*
• Discussion forums*
• Chat*
• Newsletter
• CMS
• News entry and validation
• Ads and banners management tool
• Reporting and Statistics
• Content entry interfaces for external contributors
• Automated secure channels for press agency

Source: SaS analysis

* Basic functionality
** Including 3 month content for testing purposes

REAL EXAMPLE

With Open Source
With vendors’ solutions

Software license 256
Implementation** 264
Total software 337
Hardware 220
Other implementation 41
Total 573

Software license 73
Implementation** 297
Total software 553
Hardware 195
Other implementation 47
Total 820

+247
BUILDING A CORPORATE PORTAL ON FULL OPEN SOURCE SOFTWARE FREES EVEN MORE FINANCIAL RESOURCES
2005, USD thousands

**Services**

- Search engine
- Corporate information
- Real time updated external news
- Topic based files
- Personalization
- eShop + payment gateway
- Backoffice integration
- Discussion forums
- Chat
- Newsletter
- CMS
- News entry and validation
- Workflow management
- Ads and banners management tool
- Reporting and Statistics
- Content entry interfaces for external contributors
- Automated secure channels for press agencies

**REAL EXAMPLE**

<table>
<thead>
<tr>
<th></th>
<th>With Open Source</th>
<th>With vendors’ solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software license</td>
<td>274</td>
<td>687</td>
</tr>
<tr>
<td>Implementation</td>
<td>2127</td>
<td>2265</td>
</tr>
<tr>
<td>Total software</td>
<td>2401</td>
<td>2952</td>
</tr>
</tbody>
</table>

Source: SaaS analysis
5. Migration path to Open Source
5. Migration path to Open Source

- Guiding framework
- Analyzing specific PROs and CONs
- Analyzing economic efficiency
- Following the migration path
MAKING A DECISION ABOUT WHETHER TO MIGRATE TO OPEN SOURCE IS A 3 STEP PROCESS

<table>
<thead>
<tr>
<th>Analyze Pros and Cons</th>
<th>Analyze economic efficiency</th>
<th>Follow the migration path</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Conduct specific analysis of Pros and Cons of migration</td>
<td>• Compare medium and long term costs (direct, indirect)</td>
<td>• If go-decision taken, launch migration process against highly structure plan</td>
</tr>
<tr>
<td></td>
<td>• Estimate potential savings</td>
<td></td>
</tr>
</tbody>
</table>
5. Migration path to Open Source

• Guiding framework

• Analyzing specific PROs and CONs

• Analyzing economic efficiency

• Following the migration path
USERS DISSATISFIED WITH PROPRIETARY SOFTWARE ARE MORE LIKELY TO MIGRATE TO OPEN SOURCE

Satisfaction is an aggregate of several building blocks:
- Total cost of ownership
- Reliability
- Maintenance requirements
- Flexibility
- Ease of customization
- Continuity
- After sales support
- Perceived vendors way of behaving
- Etc.

<table>
<thead>
<tr>
<th>Level of satisfaction</th>
<th>Declared desire to migrate*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very satisfied</td>
<td>1.1</td>
</tr>
<tr>
<td>Satisfied</td>
<td>1.4</td>
</tr>
<tr>
<td>Neutral</td>
<td>2.4</td>
</tr>
<tr>
<td>Unsatisfied</td>
<td>3.7</td>
</tr>
<tr>
<td>Very unsatisfied</td>
<td>4.1</td>
</tr>
</tbody>
</table>

* Scale: 1 to 5. 1="I do not consider to migrate"; 5="I am considering to migrate as soon as possible"

Source: Interviews; SaS analysis
DECISION TO CONSIDER OPEN SOURCE USUALLY RESULTS FROM A TWO-DIMENSION ANALYSIS

Caveat…

Obstacles and benefits are a mix of
• rigorous fact-based analysis (measured performance, direct & indirect costs, ROI, PV/NPV, …)
• Intangible criteria (potential users’ perceptions & opinions)

Source: Interviews; SaS analysis
Doubts around support, products and available skills are barriers to migration

What are the main reasons preventing you to migrate to Open Source so far? *

- Lack of organized, “institutionalized” support: 40%
- Immaturity of products and split of versions: 25%
- Lack of internal skills: 20%
- Other: 15%

IT managers not supporting migration to Open Source perceive obstacles as rooted both in the OSS market, in the products and in their own companies.

* Figures must be considered as indicative, for they result from a series of no more than 40 informal interviews to IT managers

Source: Interviews; SaS analysis
# MIGRATION IS TRIGGERED BY A LARGER SET OF FACTORS

<table>
<thead>
<tr>
<th>Reasons for migrating to Open Source</th>
<th>Comments</th>
<th>Relevance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Choice</td>
<td>• OS breaks vendors monopoly by introducing new opportunities for choice</td>
<td>Very relevant</td>
</tr>
<tr>
<td>2. Dynamic</td>
<td>• OS projects and communities are often times considered dynamic, exciting, and appealing to IT staff</td>
<td>Very relevant</td>
</tr>
<tr>
<td>3. Reliability</td>
<td>• As OS solutions are increasingly robust, they become more reliable even to support mission critical processes</td>
<td>Marginally relevant</td>
</tr>
<tr>
<td>4. Security</td>
<td>• OS considered as a more secure alternative to proprietary software</td>
<td>Marginally relevant</td>
</tr>
<tr>
<td>5. Cost</td>
<td>• Both OS hard costs and total cost of ownership are more favorable than competing closed source solutions</td>
<td>Marginally relevant</td>
</tr>
<tr>
<td>6. Control</td>
<td>• OS gives users control over sources and future developments</td>
<td>Marginally relevant</td>
</tr>
<tr>
<td>7. Support</td>
<td>• Despite little “institutionalized” support, OS community is usually helpful to solve problems</td>
<td>Marginally relevant</td>
</tr>
</tbody>
</table>

Source: Novell 2003; SaS analysis
SUPPORT PLAYS A KEY BUT PARADOXICAL ROLE

Both migrators and non-migrators quoted support as a key factor contributing to the go-nogo decision to migrate.

On the one side, lack of organized support is dissuasive.

On the other side, possibility to count on community for support is somehow relevant in deciding to migrate.

Perhaps, these people don't know they can rely on the OS community. Perhaps, they don't know how to get support from the OS community.

Perhaps they feel comfortable with support from OS community because they know how to get it.

Making access to OS community support easier could foster OS adoption?
OPEN SOURCE BREAKS VENDORS’ MONOPOLY BY GIVING CHOICE TO USERS

PROs
- Vendor selection (OS reduces locking-in)
- Customization
- Upgrade timing
- Bug reporting

CONs
- Excess of choice even for basic functionalities (GNOME vs KDE vs…)

Source: Interviews; Novell 2003; SaS analysis
OPEN SOURCE PROJECTS ARE USUALLY CONSIDERED AS DYNAMIC AND APPEALING

<table>
<thead>
<tr>
<th>PROs</th>
<th>CONs</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Rapid development</td>
<td>• Too frequent releases</td>
</tr>
<tr>
<td>• Frequent releases</td>
<td>• Quick extinction</td>
</tr>
<tr>
<td>• Quick evolution</td>
<td>• Too fast and unpredictable an evolution</td>
</tr>
<tr>
<td>• Strong community</td>
<td></td>
</tr>
<tr>
<td>• Intellectually exciting</td>
<td></td>
</tr>
</tbody>
</table>
INCREASING ROBUSTNESS MAKES OPEN SOURCE SOLUTIONS MORE RELIABLE

<table>
<thead>
<tr>
<th>PROs</th>
<th>CONs</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Many eyeballs, fewer bugs both in development</td>
<td>• Too frequent releases</td>
</tr>
<tr>
<td>• Updates when ready, not to generate quarterly reports</td>
<td>• Quick extinction</td>
</tr>
<tr>
<td></td>
<td>• Too fast and unpredictable an evolution</td>
</tr>
</tbody>
</table>

Source: Interviews; Novell 2003; SaS analysis
EXPERTS USUALLY CONSIDER OPEN SOURCE A MORE SECURE ALTERNATIVE TO PROPRIETARY SOFTWARE

**PROs**
- Security through full-disclosure of source code reduce risk of having back doors
- Large community of testers
- Fast reaction for patches

**CONs**
- Difficulty to trace code parentage
- Is the community properly reviewing the code?

Source: Interviews; Novell 2003; SaS analysis
COSTS OF OPEN SOURCE ARE MORE FAVORABLE THAN COMPETING CLOSED SOURCE SOLUTIONS

PROs
- Low acquisition costs
- Lower total costs of ownership
- Repurposed hardware
- Leverage unpaid community for support

CONs
- New costs: training, external support, etc.
- Lost of productivity during learning curve

Source: Interviews; Novell 2003; SaS analysis
OPEN SOURCE GIVES USERS CONTROL OVER SOURCES AND FUTURE DEVELOPMENTS

**PROs**
- Access to source
- Access to the community of creators of the code
- High configurability
- Freedom of launching any future development

**CONs**
- Money does not equal influence on vendor
- No neck to choke in case of issue

Source: Interviews; Novell 2003; SaS analysis
SUPPORT IS PROVIDED BY BOTH COMMUNITY AND EXPERT / NICHE CONSULTING COMPANIES

**PROs**

- Choice and portability: not locked into a particular vendor
- Free access to community of developers and users

**CONs**

- No neck to choke in case of issue
- Bug fixes not necessarily incorporated in code base

Source: Interviews; Novell 2003; SaS analysis
5. Migration path to Open Source

- Guiding framework
- Analyzing specific PROs and CONs
- Analyzing economic efficiency
- Following the migration path
EVEN BEFORE IT COSTS ARE ANALYZED, A SUBSTANTIAL INCREASE IN IT ECONOMIC EFFICIENCY CAN BE ACHIEVED

Refine/renew IT strategy and define clear standards for the company

Non-IT measures impacting IT economic efficiency

Streamline business processes and align/modify IT accordingly

Review match between business and IT processes

NOT EXHAUSTIVE

Source: KBSt, Migration guide, 2003; SaS analysis
SEVERAL FACTORS IMPACT ECONOMIC EFFICIENCY OF IT SYSTEMS

**Impacting factors**

- Degree to which low-cost standard software cover the required functions
- Quality, flexibility of modification, development capability of the standards, technologies and software used
- Efficiency of migration process
- Consistent integration of components into the value chain
- Internal and/or external good service organization and high quality expertise
- Economic lifecycle of software solutions
- Costs and efficiency of the purchasing and sourcing process
- Competition in the field of software solutions and services

Overall economic efficiency of IT systems

Source: KBSt, *Migration guide*, 2003; SaS analysis
### EVALUATION OF ECONOMIC EFFICIENCY OF MIGRATION COVERS BOTH MONETARY AND BENEFIT ANALYSIS

#### Methodological principles of evaluation

<table>
<thead>
<tr>
<th>Approach method</th>
<th>Definition</th>
<th>Comments</th>
</tr>
</thead>
</table>
| **Monetary analysis** | • NPV is used to determine the monetary effects of the project  
  – Realistically describe money flows (costs AND revenues) with a focus on a common reference time (ex 3 or 5 years)  
  – Discount using a discounting value based on interest rate, WACC for the industry, etc. | • Priority must be given to monetary analysis (NPV): costs and savings expressed in a ROI |
| **Benefit analysis** | • Considers effects which cannot be measured in monetary terms  
  • Evaluates individually and independently weighted target criteria  
  • Translates evaluations into scales in order to quantify soft factors  
  • Also leads to urgency and strategy ratios | • Benefit analysis helps when monetary analysis is not sufficient or does not provide clear profitability assessment |

Source: KBSt, *Migration guide*, 2003; SaS analysis
EVALUATION OF ECONOMIC EFFICIENCY TAKES INTO ACCOUNT COST CATEGORIES AND APPLICATIONS...

<table>
<thead>
<tr>
<th>Cost categories</th>
<th>Migration Cost Matrix (example)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware</strong></td>
<td></td>
</tr>
<tr>
<td>• Comparison of requirements</td>
<td></td>
</tr>
<tr>
<td>• Maintenance costs; Costs for directory systems; System management &amp; security</td>
<td></td>
</tr>
<tr>
<td>• Administration, purchasing, license management; Support, consulting; Software maintenance; Training</td>
<td></td>
</tr>
<tr>
<td><strong>Software</strong></td>
<td></td>
</tr>
<tr>
<td>• File servers: $xyz</td>
<td>• Open Office: $0</td>
</tr>
<tr>
<td>• Print server: $xyz</td>
<td>• Gimp: $0</td>
</tr>
<tr>
<td>• MySQL: $0</td>
<td>• DB servers: $xyz</td>
</tr>
<tr>
<td>• Installation and configuration by consulting company ABC: $xyz</td>
<td></td>
</tr>
<tr>
<td><strong>Man Power</strong></td>
<td></td>
</tr>
<tr>
<td>• Training for 250 staff: $xyz</td>
<td></td>
</tr>
<tr>
<td>• Installation and configuration by consulting company ABC: $xyz</td>
<td></td>
</tr>
<tr>
<td><strong>Server infrastructure</strong></td>
<td><strong>Desktop infrastructure</strong></td>
</tr>
<tr>
<td>• DB servers: $xyz</td>
<td>• Web application server: $xyz</td>
</tr>
<tr>
<td><strong>Common applications</strong></td>
<td><strong>DB &amp; Web applications</strong></td>
</tr>
</tbody>
</table>

Source: KBSt, Migration guide, 2003; SaS analysis
Potential savings from migration considered as a means of reducing / eliminating “vendor dependence”

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reducing “vendor dependence” facilitates direct savings</td>
</tr>
<tr>
<td>• Freedom to select cheaper products</td>
</tr>
<tr>
<td>• Freedom to select better tailored products (no need to pay for useless functionalities)</td>
</tr>
<tr>
<td>• Possibility to outsource some support to the OS community for free</td>
</tr>
</tbody>
</table>

Macro-economic savings coming mostly as effects of reducing the “vendor dependence”
| • Better product quality |
| • Lower product price |
| • Higher innovation rate |

Source: KBSt, *Migration guide*, 2003; SaS analysis
5. Migration path to Open Source

- Guiding framework
- Analyzing specific PROs and CONs
- Analyzing economic efficiency
- Following the migration path
MIGRATION TO OPEN SOURCE NEEDS TO FOLLOW A SHARP PROCESS TO SECURE SUCCESS

<table>
<thead>
<tr>
<th>Secure pre-requisite</th>
<th>Definition of targets</th>
<th>Analysis of work process</th>
<th>Selection of standard software</th>
<th>First migration step</th>
<th>Client migration</th>
<th>Users training</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Management decision</td>
<td>• Analysis of current situation and of objectives</td>
<td>• Analysis of IT work process: – Features – Software – External partners</td>
<td>• Review of available software</td>
<td>• Decision on in-house or external support</td>
<td>• Final buy-in from end users</td>
<td></td>
</tr>
<tr>
<td>• Buy-in from stakeholders</td>
<td>• Refinement of project plan</td>
<td>• Optimization</td>
<td>• Assessment (including customization needs)</td>
<td>• Migration of servers</td>
<td>• Set helpdesk</td>
<td></td>
</tr>
<tr>
<td>• Assessment of migration policies</td>
<td>• Project planning</td>
<td>• Assessment of GUI for client software</td>
<td>• Optimization, fine tuning</td>
<td>• Optimization, testing, migration</td>
<td>• Nomination of beta testing dept</td>
<td></td>
</tr>
<tr>
<td>• Project planning</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Source: Interviews; All2All; SaS analysis
SECURING PRE-REQUISITE IS KEY

- Secure pre-requisite
- Analysis of targets
- Analysis of work process
- Selection of standard software
- First migration step
- Client migration
- Users training

- Management has to take sharp clear decision
  - Pros vs Cons
  - Feasibility
  - Go-nogo

- Endorse decision
- Communicate company wide; do road show to get buy-in
- Do not leave choice to employee to use new IT or continue using usual software

- Select appropriate migration policy
  - Partial or complete migration?
  - One-step or gentle migration path?

- Plan accurately the migration project in order to optimize
  - Reorganization and extension of company IT
  - Porting of complex work flows and mission critical processes onto new software

Source: Interviews; All2All; SaS analysis
BASICALLY, THE TRANSITION POLICY FRAMEWORK OFFERS FOUR DIFFERENT OPTIONS

<table>
<thead>
<tr>
<th>Methodology</th>
<th>One-Step</th>
<th>Gentle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• Interesting option for pilot projects before launching the total one-step migration</td>
<td>• If all required resources not secured yet and when only limited buy-in, or • For pilots</td>
</tr>
<tr>
<td></td>
<td>• When decision is endorsed by top management and all required resources made available</td>
<td>• In wealthy but conservative organizations, or • When large staff requires in depth training</td>
</tr>
</tbody>
</table>

Partial | Complete

Scope of migration

Selection of more appropriate option depends on many factors:
• Top management endorsement
• Available resources (time, money, people)
• Qualifications of staff
• Size of organization
• Role of IT in core business
• Etc.

Source: All2All ; KBSt, Migration guide, 2003 ; SaS analysis
MIGRATION CAN BE EITHER COMPLETE OR PARTIAL

**Partial/selective migration:**
Migrate only servers and back-office*, leaving clients untouched

**Complete migration:**
Migrate all IT to Open Source, back-office and clients

**PROs**
- Easy migration
- Transparent to users of the network (no apparent changes for end users)

**CONs**
- Does not take advantage of all strengths of Open Source:
  - Costs savings
  - Standardization
  - Open formats
  - Etc.

- Takes advantage of all strengths of Open Source
- Protects investment in the long term as open standards become ubiquitous through out the company

- Complex and demanding migration process
- Usually requires training to end-users
- Affects end-users’ productivity during learning curve

* Proxy, fileserver, printserver, mailserver

Source: Interviews; All2All; SaS analysis
AND IT CAN BE REALISED AS A ONE-STEP CHANGE OR A GENTLE TRANSITION

Typical reasons for one-step migration

- Support for legacy is phased out
- Admin and users are faced with far-reaching change
- Admin do not want to handle heterogeneous IT over time

Typical reasons for gentle migration

- Limited budget forces to distribute costs over time
- Need for extensive training of staff (IT and end users)
- Skepticism or lack of buy-in

Source: KBSt, *Migration guide*, 2003; SaS analysis
A CLEAR VISION OF THE TARGET MUST DRIVE THE MIGRATION PROCESS

- What do we want to achieve? Where do we want to be one/two year/s from now?
  - Collect data and information
    - Management wish-list
    - IT dept wish-list
    - End users wish-list
    - Run internal survey to all employees
  - Analyze data and information
  - Produce blue book describing expected future situation
- Are there quick wins?
- How can we go there? How can we make the jump?
- Refine initial project plan

Source: Interviews; All2All; SaS analysis
IN DEPTH ANALYZIS OF WORK PROCESSES ALSO GIVES AN OPPORTUNITY TO OPTIMIZE THEM

- Analysis of current situation:
  - What feature supporting what process?
  - What software supporting what feature?
  - How do various processes interact each other?
  - How do software communicate?
    - File formats?
    - Protocols?
    - etc
  - Who are the external partners involved (suppliers, financial institutions, etc)?

- Detailed documentation of all aspects
  - Structure and report information
  - Share documentation

- Optimization of processes
  - Streamline processes
  - Reduce duplication of functionalities
  - Benchmark used software against market offering
  - Review alignment of IT processes with business processes
  - Consider users' wish list

- Detailed documentation of all aspects
  - Structure and report information
  - Share documentation
  - Format documentation according to existing international standards

As a side effect of this accurate process of analysis, optimization and documentation, migration could also bring the company to an ISO-9000 certification

Source: Interviews; All2All; SaS analysis
SOFTWARE ACCORDING NEEDS, NOT THE OPPOSITE

1. Secure prerequisite
2. Definition of targets
3. Analysis of work process
4. Selection of standard software
5. First migration step
6. Client migration
7. Users training

Analysis of work processes is used to define needs for future IT environment

Standard software final selection

- Select more appropriate open server software, and office suite
- Assess, plan and implement software personalization / customization needs
- Figure out how to replace all proprietary file formats and protocols by open ones

Source: Interviews; All2All; SaS analysis
MIGRATING ALSO MEANS TESTING AND DOCUMENTING

Secure pre-requisite → Definition of targets → Analysis of work process → Selection of standard software → First migration step → Client migration

Users training

Pending on level of in house skills, IT management must decide*

Hire external support (consulting, training, etc.)

Launch migration on in house skills

Start migration by system servers and IT department (infrastructure services)

- Transparent for user basis that will not notice its printing over a Linux printer server or accessing files over a SAMBA file server

Continue migration by back office servers (middleware and services)

- database servers
- web servers supporting Intranet and/or Internet
- fax server
- groupware server
- messaging
- etc.

Run intensive and extensive tests & Document

* Actually, this decision is usually taken earlier in the process but come into effect at this stage

Source: Interviews; All2All; SaS analysis
END USERS MIGRATION REQUIRES TESTS, SUPPORT AND TRAININGS

<table>
<thead>
<tr>
<th>Secure pre-requisite</th>
<th>Definition of targets</th>
<th>Analysis of work process</th>
<th>Selection of standard software</th>
<th>First migration step</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Client migration**
- **Users training**

**Definition of targets**
- Secure prerequisite
- Analysis of work process
- Selection of standard software
- First migration step

**First migration step**
- Create beta test department
  - Volunteers (20-30 people) will have less resistance to innovation
  - Do not select a dept where work processes converge
  - Take care of psychological dimension in order avoid scaring user basis
  - Communicate in order to avoid negative rumors
  - Build documentation library
- Launch help desk to support migration amongst end users
- Pay attention to provide end users with adequate training

**Training**
- Beta test
- Help desk

**Selected GUI**
- GNOME
- K Desktop Environment (KDE)
- GnuStep
- Common Desktop Environment (CDE)

**Selected GUI**

Source: Interviews; All2All; SaS analysis
6. Review of cases
6. Review of cases

• Automotive
  – Sales and Marketing: Toyota US
  – Design: Formula 1 Williams BMW

• eBusiness: Amazon.com

• Education: HCPS College

• Energy: Enercom

• Media: Le Figaro

• Limited adoption
IN 2000-2001, TOYOTA MOTOR SALES USA FACED A TOUGH IT CHALLENGE

Initial challenge
Achieve the same level of reliability in the dealer IT network as in the vehicles

Situation
- 1,300 dealers (Toyota and Lexus) in 49 States + headquarters
- 90,000 employees
- Huge flood of information to disseminate to this field force
- Huge flood of information to gather from this field force

Problem faced
How to achieve multiple and sometimes conflicting goals of speed, flexibility, simplicity for an IT system to deliver and gather information across the nation?

Source: IDC
**Requirements**

- Provide information-rich content to the dealers over a national wide area network using limited bandwidth
- Store and forward huge cached content
- Do not change any Toyota production code
- Keep the cost per site (dealer) as low as possible

<table>
<thead>
<tr>
<th>Proposed solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red Hat Linux and Dell Computers partnered to propose and develop a common solution able to meet the requirements relying on surprisingly low systems specifications:</td>
</tr>
<tr>
<td>• 700 Mhz Pentium</td>
</tr>
<tr>
<td>• 128 MB RAM</td>
</tr>
<tr>
<td>• 20-30 Gb HD</td>
</tr>
<tr>
<td>• Cost per site : US$ 1,500</td>
</tr>
</tbody>
</table>
AND GOT AN EFFICIENT BRAND NEW IT SYSTEM FOR AN AMAZINGLY LOW COST...

2001, USD

**Toyota sales worldwide**

- Rest of the World: 3.4 million cars
- USA: 1.6 million cars

**Sales channel in the USA**

- Number of dealers: 1,300
- Cost of new IT system: US$ 1,500 per dealer
- Soft dev. OS: 250
- Hardware: 1,250

**Total cost of new IT system**:

- US$ 1,950,000
- or
- US$ ~1.18 per car sold in the USA

Source: IDC; SaS analysis
...WHILE REALIZING SUBSTANTIAL IMMEDIATE COST SAVINGS
USD, thousands

Contextual data

- Cost of closed source OS: US$ 500 per box
- Extra cost of hardware if closed source OS: US$ 300 per box

OS savings by not requiring server OS
Hardware savings by using Linux OS
Total immediate savings

Source: IDC; SaS analysis
**RELIABILITY AND MANAGEABILITY OF IT SYSTEM ALSO POSITIVELY IMPACTED BUSINESS EFFECTIVENESS**

<table>
<thead>
<tr>
<th>Main improvements</th>
<th>Business impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Decrease in number of system crashes</td>
<td>• Lower cost of downtime</td>
</tr>
<tr>
<td>• Launch of a remote server administration functionality</td>
<td>• Lower cost of maintenance (number of IT maintenance team travels decreases dramatically)</td>
</tr>
<tr>
<td>• Better dissemination of information across the dealer network</td>
<td>• Improvement in sales effectiveness</td>
</tr>
</tbody>
</table>

Source: IDC; SaS analysis
6. Review of cases

- Automotive
  - Sales and Marketing: Toyota US
  - Design: Formula 1 Williams BMW
- eBusiness: Amazon.com
- Education: HCPS College
- Energy: Enercom
- Media: Le Figaro
- Limited adoption
AERODYNAMICS IS KEY IN FORMULA 1 BUT REQUIRES HEAVY COMPUTE CAPABILITY

Initial challenge
Improve aerodynamic simulation to enhance F1 car performance

Situation
• BMW William Formula 1 is a leading F1 team
• Competitive advantage can be achieved through tires, engine power, chassis and aerodynamics, with the latter the most difficult to control
• Improvements of compute capability must comply with approved budget

Problem faced
How to reduce time taken to perform a complete simulation and how to improve quality of mathematical analysis supporting simulation?
LINUX CLUSTER SOLUTION WAS ADOPTED BECAUSE OF MATHEMATICAL EFFICIENCY AND COST EFFECTIVENESS

Benefits

- Biggest reduction in total analysis time within budget
- Threefold enhancement of simulation capabilities through more detailed computational fluid dynamics simulations
- Scalability of cluster that can therefore be expanded in the future
- More capacity to experiment with new car design concepts
- Linux able to exploit off-the-shelf applications that require heavy compute capability

BMW William Formula 1 installed a cluster of several hundreds HP ProLiant Intel-based servers:
- multi-rack Intel-Xeon processor-based system controlled through a head node HP Integrity (Itanium 64-bit) server

Source: vnunet.com; cfdreview.com; SaS analysis
6. Review of cases

- Automotive
  - Sales and Marketing: Toyota US
  - Design: Formula 1 Williams BMW
- eBusiness: Amazon.com
- Education: HCPS College
- Energy: Enercom
- Media: Le Figaro
- Limited adoption
AMAZON WAS LOOKING TO CUT ITS IT COST AND STILL PROVIDE AN EXCELLENT CUSTOMER EXPERIENCE

Initial challenge
Deploy new IT infrastructure and complete the migration before the peak season

Situation
• Worldwide largest e-retailer
• Seasonal spikes in traffic
• In 2000, net sales were around USD 1b per Quarter
• RISC/Unix-based IT system
• Technology and content expenditures were as high as USD 70m per Quarter (~7.2% of net sales)

Problem faced
How to reduce IT costs and still provide excellent customer experience without any service interruption?
Because Linux runs on commodity servers, Amazon expected to benefit from lower-cost computers

Requirements

- Complete migration before start of peak sales season
- Cut total cost of ownership of IT infrastructure
- Continue providing excellent customer service
- Scalability, stability, reliability across IT infrastructure

Proposed solution

- Replace RISC/Unix servers with commodity Intel hardware
- Migrate all applicable servers from RISC/Unix to Linux (based on availability of applications)
- Leverage IT staff Unix skills to facilitate Linux learning curve

Source: IDC; Amazon financial statements; SaS analysis
AMAZON WAS BOTH ABLE TO CUT ITS TECHNOLOGY AND CONTENT EXPENDITURE BY 24% OR USD 17.4m...

Amazon Technology and Content quarterly expenditure
USD million

3Q 2000 3Q 2001 3Q 2002 3Q 2003
71.2 53.8 52.9 53.8

Cost savings were mainly achieved by replacing more than one hundred USD 60,000 RISC/Unix boxes with USD 10,000 Intel servers.
...AND SUBSTANTIALLY REDUCE ITS CASH OUT FOR INVESTMENT IN SOFTWARE AND WEBSITE

Quarterly cash out for purchase of fixed assets, incl. internal-use software and Website development USD million

3Q 2000 | 3Q 2001 | 3Q 2002 | 3Q 2003
--- | --- | --- | ---
41.9 | 12.9 | 11.3 | 15.2

Cash out cliff partially due to lower IT expenditure

Source: IDC; Amazon Cash Flow Statements; SaS analysis
6. Review of cases

• Automotive
  – Sales and Marketing : Toyota US
  – Design : Formula 1 Williams BMW

• eBusiness : Amazon.com

• Education : HCPS College

• Energy : Enercom

• Media : Le Figaro

• Limited adoption
PUBLIC EDUCATION INSTITUTIONS MUST MEET BOTH NARROW BUDGET AND GROWING IT TRAINING DEMAND

Initial challenge
Deploy new software solutions in server rooms of the College

Situation
• Harrisonburg City Public Schools describes itself as a “budget-strapped public education institution”
• Extended IT capabilities are key to allow increasing number of students to practice

Problem faced
How to be cost effective in IT deployment considering that common proprietary software is quite expensive

Source: staff.harrisonburg.k12.va.us; SaS analysis
OPEN SOURCE SATISFIES SPECIFIC REQUIREMENTS OF NICHE EDUCATION MARKET

Requirements

• Meet strapped budget typical of public education institutions
• Provide enough computing capabilities to educate growing number of students interested in IT
• Minimize support
• Reduce risks of software piracy

Proposed solution

• Use only open source software
• Use low-requirement software in order to extend hardware usage beyond usual lifecycle
• Recycle old, retired PCs as basic servers and desktops for practice

Source: staff.harrisonburg.k12.va.us; SaS analysis
OPEN SOURCE SUBSTANTIALLY REDUCED OVERALL IT COSTS...
2002, USD thousand

Savings realized thank to adoption of open source across the College server rooms

Source: staff.harrisonburg.k12.va.us; SaS analysis
...AND MADE SOME BUDGET AVAILABLE TO INVEST IN ADDITIONAL COMPUTING CAPABILITIES

Possible IT strategies

Proprietary software

Open source software

Savings

Utilization of available resources

- USD 78,000
  - 2 web servers
  - 3 other servers

- 6 web servers
  - 8 other servers

Source: staff.harrisonburg.k12.va.us; benchmark; interviews; SaS analysis
LONG TERM COST SAVINGS WERE ALSO ACHIEVED

- Better security means lower risk of losing data or productive staff time needed to clean up after a security breach

- New hardware would imply purchase of new proprietary software licenses, while students copying proprietary software for their own usage is also a risk for the school

- A virus infection on a network server can be devastating in terms of data loss, down time, and staff time required for reconstruction

- The usable lifetime of a commercial software product is rarely longer than 2-to-4 years. At this point one must purchase a newer product or an upgrade to the existing one

Source: staff.harrisonburg.k12.va.us; SaS analysis
6. Review of cases

- Automotive
  - Sales and Marketing: Toyota US
  - Design: Formula 1 Williams BMW

- eBusiness: Amazon.com

- Education: HCPS College

- Energy: Enercom

- Media: Le Figaro

- Limited adoption
IN 2002, ENERCOM NEEDED TO SCALE ITS IT INFRASTRUCTURE IN A COST EFFECTIVE WAY

Initial challenge
Expand SAP user base from ~300 users to more than 700 while improving system performance

Situation
- Enercom is leader in wind power market
- EUR 1.2 b revenue, 40% market share
- Over 6,500 energy plants in 29 countries
- Fast growing company from a few hundreds employees in the 90s to ~5,500 in 2003
- Win NT & Oracle infrastructure

Problem faced
How to be cost effective in IT expansion considering that any extension of Win NT & Oracle architectures is very costly

Source: IDC
**Requirements**

- Upgrade the SAP platform in order to support more than twice the original client base
- Warranty greater SAP performance despite the higher number of clients supported
- Keep the migration and ongoing costs as low as possible

**Proposed solution**

Get rid of Windows NT and Oracle and have the SAP environment running on a new open source platform built upon

- MaxDB-MySQL
- Linux

MaxDB (SAP compliant) would reduce cost for additional users compared to proprietary DB

Source: IDC; SaS analysis
ENERCOM REALIZED IMMEDIATE SAVINGS OF EUR 146,000

2003, EUR thousand

ESTIMATES

Proprietary database licenses: 126
Proprietary OS: 94
Total savings: 220
Purchase of servers for Linux + MaxDB: 45
Cost of migration services: 25
Training MaxDB: 2
Linux OS package: 2
Immediate savings realized: 146

Source: IDC; SaS analysis
AND SECURED LONGER TERM SAVINGS BY CUTTING IT MANAGEMENT AND OPERATING COSTS

**No downtime**: Previously Enercom required 2h of DB downtime per week. Now, backups are done without discontinuing the service. Also, MaxDB has been running steadily without restart.

**Ease of admin**: DB admin is now easy through an intuitive GUI. Logs and status are available in a single view. Creating test environment is much easier than before.

**Stability**: The combination of Linux and MaxDB delivers Enercom a solid and reliable infrastructure that has proven to be easy to handle, stable and without previous limitations (memory and process) that Windows had.

**Ease of use & learning**: Learning curve of MaxBD for SAP is easy and fast. Training of a new admin takes now 1 week instead of more than 3 weeks previously.
6. Review of cases

• Automotive
  – Sales and Marketing: Toyota US
  – Design: Formula 1 Williams BMW

• eBusiness: Amazon.com

• Education: HCPS College

• Energy: Enercom

• Media: Le Figaro

• Limited adoption
What IT infrastructure to ensure no service discontinuity in supporting both complexity of business process and constant crushing 24 X 7 deadlines?

Problem faced

What IT infrastructure to ensure no service discontinuity in supporting both complexity of business process and constant crushing 24 X 7 deadlines?

Initial challenge
Enhance communication and email system meeting specific needs of a well established media

Situation
- Newspaper daily circulation approaching 1.4 m readers
- Le Figaro also publishes a host of weekly magazines
- Le Figaro coordinates its editorial content and subscriber bases with hundreds of reporters and staff across a complex Intranet

Source: IDC
**Requirements**

- The IT infrastructure has to mirror the newspaper business: it can never stop
- IT system has to support
  - 700 onsite mailboxes
  - Remote access
  - Anti-virus protection
  - Exchange of multimedia files
  - Real time Internet website and Intranet tools

**Proposed solution**

- Exploit Red Hat Linux 7.0 + PostgreSQL DB to develop customized applications such as new email system, easy to parameter
- Deploy new solution as fast as possible in order to reduce uncertainties due to IT transition phase
BENEFITS OF OPEN SOURCE EVEN EXCEEDED EXPECTATIONS

Red hat Linux covers almost all aspects of *Le Figaro* IT system:
- Redundant email servers
- Redundant FTP and web servers
- Remote access servers
- DNS servers

Benefits
- Stability/reliability (e.g., not a single crash during more than 300 days)
- Both global and granular control of email parameters
- Scalability
- Ease of use and implementation
- No cost of OS and DB licenses
- Low cost of implementation
- Low operating costs: support, maintenance, upgrades

Source: IDC; SaS analysis
6. Review of cases

• Automotive
  – Sales and Marketing : Toyota US
  – Design : Formula 1 Williams BMW

• eBusiness : Amazon.com

• Education : HCPS College

• Energy : Enercom

• Media : Le Figaro

• Limited adoption
SEVERAL COMPANIES HAVE BEEN USING OPEN SOURCE FOR YEARS IN SPECIFIC FIELDS OF THEIR BUSINESS

<table>
<thead>
<tr>
<th>Company</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Banca Commerciale Italiana</td>
<td>Web based application supporting online banking is written in Java running on Linux</td>
</tr>
<tr>
<td>TSB Bank New Zealand</td>
<td>Specialized image capture system speeding up cheque processing running on Linux</td>
</tr>
<tr>
<td>Deutsche Post Direkt</td>
<td>Postal mail routing system</td>
</tr>
<tr>
<td>British Petroleum</td>
<td>Oil and gas exploration through seismic imaging is based onto a cluster of more than 1,000 Intel Itanium and Linux boxes</td>
</tr>
<tr>
<td>MarienHospital (Germany)</td>
<td>SuSE Linux based servers supporting vendors’ databases</td>
</tr>
<tr>
<td>Afilias Ltd</td>
<td>Internet Service Provider using PostgreSQL database to support .org domain information</td>
</tr>
<tr>
<td>US Postal Services</td>
<td>Since 1997, Linux-based systems in mail distribution centers to recognize destination addresses on envelopes</td>
</tr>
<tr>
<td>Autonomous Community of Extremadura (Spain)</td>
<td>In 2002, the poorest region of Spain has adopted Linux as the official operating system of public schools and offices</td>
</tr>
<tr>
<td>Police of Niedersachsen (Germany)</td>
<td>In 2002, 11,000 workstations were migrated to Linux and OpenOffice</td>
</tr>
</tbody>
</table>
AN INCREASING NUMBER OF COMPANIES RECENTLY ANNOUNCED PLANS TO ADOPT OPEN SOURCE

<table>
<thead>
<tr>
<th>Date</th>
<th>Company</th>
<th>Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td>June 2004</td>
<td>State of France</td>
<td>Migrate all servers to Open Source</td>
</tr>
<tr>
<td>July 2004</td>
<td>City of Barcelona, Spain</td>
<td>Phase out its current Windows NT infrastructure in favor of open source software</td>
</tr>
<tr>
<td>July 2004</td>
<td>European Centre for Medium-Range Weather Forecasts</td>
<td>Installation of Linux cluster as a pilot project</td>
</tr>
<tr>
<td>July 2004</td>
<td>Government of Malaysia</td>
<td>All Government technology procurement will have a preference for open source software</td>
</tr>
</tbody>
</table>
7. Case study: NewCo in Electricity Distribution
7. Case study: NewCo in Electricity Distribution

- Case description
- Additional data
- “Resolution”
While looking out of the window of his small office at the snow that was quietly covering the city center on that magic Tuesday January 20th 2004, John Doutremont felt really happy and calm for the first time after 9 months spent working nights and days to put together the guidelines of the most ambitious, exciting and perhaps foolish project he ever initiated.

He was still smiling, and relaxing the muscles of his neck when he picked up the phone and called his business partners to share the great news with them. He had in fact just received an official fax from the National Regulation Authority for Electricity (NRAE). Their project of creating and launching a company of electricity distribution was approved, the first one in the recently deregulated national market. The new venture had just made a crucial step towards a more tangible future.

The task would be tough but according to their business plan it was worth the effort. Several studies and researches they conducted during the previous 6 months made John Doutremont and his partners confident it was possible to capture up to 12% of market share within 3 years after the launch. Small and Medium Enterprises (SMEs) as well as large companies would be their only target as they were not interested neither in the SOHOs (Small Office Home Office) nor in the residential market. Actually, the offering of products and services they had in mind was tailored for professional users.

Also, in order to avoid too large a sales force to deal with all customers, they had deeply explored opportunities offered by remote contact channels and came to the conclusion that 90% of clients would have subscribed the service through an online interface, keeping the sales force available for the 10% of the clients they thought would have special requirements or request special assistance.
Sources of revenue

The project consisted in fact in both delivering cheaper electricity to business end users and keeping their electricity consumption under strict control in order to help them rationalize the usage of energy. The clients would then benefit at least two quick wins enforcing each other: a lower consumption of electricity available at a lower rate. Providing the clients with elaborate hard facts on their consumption as well as cheaper electricity would allow the client to dramatically cut its costs of energy.

In other words, the delivery of cheaper energy was conceived as a mere consequence of the consumption streamline service while the price they would market their own products and services was calculated as

- a fixed fee for the service (covering the cost of the monitoring tool, the cost of installation and margin on the monitoring tool and installation) and
- a margin on the electricity provided.

How would the new company (NewCo) keep the clients’ consumption under control?

The NewCo would install* at each client monitoring point (a client might have more than one interested monitoring point) a specific data collector** that would record the consumption and send over the Internet every 5 minutes to NewCo IT back-office an encrypted file containing the data. This would allow NewCo to establish a clear and extremely detailed profile of the client’s consumption. Based on this function, together with other information on the productive activities provided by the client, NewCo would be able to “nominate” for energy on the next day/week and get it at the best rate on the international electricity market (instead of always buying it on the national market as it was the case under the monopolistic situation).

* Actually, installation will be outsourced to prevent hiring hundreds of agents but total number of Technical services people in NewCo will grow proportionally to the number of monitoring points
** This is basically just a handheld PDA in a closed box requiring a US $70 adaptation, connected to the Internet.
How would the new company deliver the electricity?

Once “nominated” at an electricity producer, this one injects the electricity on the national public network when needed. Thank to the data collector installed at the clients’ sites, NewCo tracks the actual consumption and is able to bill it. Also, because its whole system is interfaced with a web server, the clients would be able to access their consumption data (quantity, profile, etc.) online in real time (with a 5 minutes delay). Whenever there is a discrepancy between the quantity of electricity “nominated” and the quantity actually used by the client, this creates a “disbalanced” situation on the public network that implies NewCo to pay a penalty to the regulator of the public network. That is why NewCo needs to be extremely accurate in drawing the consumption profile of its clients.

* * *

The whole business system of NewCo was very clearly described in the business plan and resisted the challenge of the venture capitalists and the industry experts as well. However, despite the initiators of the project could already secure US $ 11.7 m from a large corporate VC well known in the industry, they had not yet received the green light from another VC fund to get the US$ 6.5 m they had requested.
On the same day, John Doutremont also called Richard Hochlaga, the partner in charge of his NewCo project at Westmount Fund Inc., one of the most dynamic and active Venture Capital fund in the country. He told him the good news from the National Regulation Authority for Electricity and then also asked him the current status of his US $ 6.5 m funding request.

“Well, Richard said, as you know we at Westmount are very supportive towards your project. We think the team you have gathered has the right set of skills, you all have a long business experience, you know the market better than everyone in this country and you can count on the deep technical expertise of some of your collaborators. Your business plan is crystal clear, robust and built upon a set of impressive market researches and international benchmarks. Needless to say then that you have very good chance to get us onboard.”

John Doutremont was pleased to ear this long list of appraisals from one of the toughest VC partner he had ever met, but he was also expecting the flip side of this nice assessment… when Richard, after a brief pause, also added :

“However John, in order to take the best decision, our board of partners would like some more details on how you plan to spend our money to build up a solid IT hardware infrastructure. In your business plan, you just give the high level rationale, followed by the figures of the total IT expenditures. But one of our new partners who comes from a technology consulting firm would feel more comfortable to invest in NewCo if you provide us with a more detailed and granular IT plan. No need to go through the details of investment for the software integration you will need, for this is already clear in your business plan. Same for the peripherals (printers, scanners, etc.), you explained quite well the rationale behind you needs. Same also for what you’re going to outsource: we have all understood that you won’t take care of implementing your website, your email servers, etc. Please just focus on some basics that are somewhat underdeveloped in your current document: hardware costs and license fees.”
“Yes, no problem”, John said, knowing he already had all the data in his files “we will adjust our business plan to present you more accurate data on the IT costs items we will rely on for our scenario.”

“Mmm, you see John, Richard added, our new IT partner is not convinced that exploring only one IT strategy is the best way to proceed, and we all agreed with him it would make a lot of business sense if you can also include in this addendum to the BP a cost comparison between 2 scenarios: branded software vs opensource software. John, don’t write a whole blue book; a few pages, sharp content, reasonable and fact-based estimates will be sufficient.

“As our next partners meeting is scheduled day after tomorrow, if you can send us the addendum by this evening, it would give us one day to read and evaluate it accurately. I am sure this is not big deal for you. Brutal cash-out figures will be fine for now, at this stage don’t spend time in discounting your figures or calculating depreciation.”

Right after this conversation, John called his business partners again to tell them to cancel any social activity they had plan to celebrate the NRAE good news… Despite the snow and cold weather outside, the new IT plan was about the heat their brain up, keeping them busy during the whole afternoon and evening.
7. Case study: *NewCo* in Electricity Distribution

- Case description
- Additional data
- “Resolution”
DEMOGRAPHICS
2002

- Total population: 60.1 m
  - Active: 27 m
- Total number of enterprises: 2,768,000

Source: CIA factbook
DISTRIBUTION OF ENTERPRISES BY NUMBER OF EMPLOYEES

Thousands; 2002

<table>
<thead>
<tr>
<th>Number of employees</th>
<th>0</th>
<th>1 - 9</th>
<th>10 - 499</th>
<th>500 - &gt;</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sources</td>
<td>1,339</td>
<td>1,195</td>
<td>232</td>
<td>2</td>
<td>2,768</td>
</tr>
</tbody>
</table>

Source: Insee
## AVERAGE NUMBER OF MONITORING POINTS PER ENTERPRISE

2002

<table>
<thead>
<tr>
<th>Size of the enterprise (number of employees)</th>
<th>Average number of monitoring points</th>
</tr>
</thead>
<tbody>
<tr>
<td>• 1 - 9</td>
<td>• 1.14</td>
</tr>
<tr>
<td>• 10 - 499</td>
<td>• 1.96</td>
</tr>
<tr>
<td>• 500 - more</td>
<td>• 3.35</td>
</tr>
</tbody>
</table>

Source: SaS analysis
### NEEDED HARDWARE INFRASTRUCTURE

<table>
<thead>
<tr>
<th><strong>Rationale</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Back hardware office infrastructure* is not related to the quantity of electricity provided but to</td>
</tr>
<tr>
<td>– Total number of monitoring points deployed at the clients’ sites</td>
</tr>
<tr>
<td>– Total amount of data collected</td>
</tr>
<tr>
<td>• All servers are the same model but database server</td>
</tr>
<tr>
<td>• Front office: all employees have the same model of strong laptop</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Minimum hardware needs</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• 1 data processing server supports 1,000 monitoring points (+ 1 hot back-up server every 10 servers)</td>
</tr>
<tr>
<td>• 1 load balancer for each cluster of 8 active data processing servers (+ 1 hot back-up every 10 load balancers)</td>
</tr>
<tr>
<td>• Required storage: each monitoring point sends a file of 6 Kb every 5 minutes</td>
</tr>
<tr>
<td>• 1 database server (+ hot back-up)</td>
</tr>
<tr>
<td>• 1 application server for online applications (+ hot back-up)</td>
</tr>
<tr>
<td>• 1 web server (+ hot back-up)</td>
</tr>
<tr>
<td>• 1 application server for back office operations (+ hot back-up)</td>
</tr>
<tr>
<td>• 1 staging environment server</td>
</tr>
<tr>
<td>• 2 file servers for company usage (+ hot back-up)</td>
</tr>
<tr>
<td>• Laptop provided to all employees (+ always 5 back-up laptops)</td>
</tr>
</tbody>
</table>

* Networking (router, hub) and security (firewalls) IT not considered
In order to reduce costs, management has decided to purchase one single model of server for all functions but database.

**Server requirements:**
- 2 CPUs at 2 Ghz
- 4 Gb RAM
- 2 X 36 Gb HD at 15,000 rpm
- 2 controllers RAID

**DB server requirements:**
- 4 CPUs at 2 Ghz
- 12 Gb RAM
- 2 X 73 Gb HD at 15,000 rpm
- 2 controllers RAID

Source: SaS analysis
## SYSTEM ADMINISTRATORS

### USD

<table>
<thead>
<tr>
<th>Platform</th>
<th>Annual salary cost</th>
<th>Servers per SysAdmin*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sun Solaris</td>
<td>85,000</td>
<td>6.5</td>
</tr>
<tr>
<td>Windows</td>
<td>68,000</td>
<td>10</td>
</tr>
<tr>
<td>Linux</td>
<td>71,000</td>
<td>44</td>
</tr>
</tbody>
</table>

* This is the number of servers one single System Administrator is able to simultaneously take care of during the year

Source: Robert Frances Group, 2002
The new company is planning to undergo 3 main development phases

- Business building
- Launch
- Growth*

* Each salesperson is able to treat 200 clients deal per year
ORGANISATION OF NEW COMPANY

At launch

CEO

2*

IT
3

Technical services
3

Sales & Marketing
5

Legal & Lobby
2

Support (HR, Finance…)
3

Product development
2

Growth phase
(1 year after launch)

CEO

3

IT
x

Technical services
5

Sales & Marketing
y

Legal & Lobby
2

Support (HR, Finance…)
5

Product development
2

* Includes secretary

n Full time equivalent
7. Case study: NewCo in Electricity Distribution

• Case description

• Additional data

• “Resolution”
NewCo’S CUSTOMER BASIS IS AMBITIOUS…

<table>
<thead>
<tr>
<th>Number of employees</th>
<th>Number of enterprises</th>
<th>Potential market</th>
<th>Market share</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>y1 %</td>
</tr>
<tr>
<td>0</td>
<td>1,339,000</td>
<td>0%</td>
<td>0</td>
</tr>
<tr>
<td>1 to 9</td>
<td>1,195,000</td>
<td>33%</td>
<td>394,350</td>
</tr>
<tr>
<td>10 to 499</td>
<td>232,000</td>
<td>100%</td>
<td>232,000</td>
</tr>
<tr>
<td>500 or more</td>
<td>2,000</td>
<td>100%</td>
<td>2,000</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td><strong>25,134</strong></td>
<td><strong>50,268</strong></td>
<td><strong>75,402</strong></td>
</tr>
</tbody>
</table>

Source: SaS analysis
...NUMBER OF MONITORING POINTS IS THEREFORE HIGH

<table>
<thead>
<tr>
<th>Size of Company</th>
<th>Monitoring Points per Company</th>
<th>Y1</th>
<th>Y2</th>
<th>Y3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 to 9</td>
<td>1.14</td>
<td>17,982</td>
<td>35,965</td>
<td>53,947</td>
</tr>
<tr>
<td>10 to 499</td>
<td>1.96</td>
<td>18,189</td>
<td>36,378</td>
<td>54,566</td>
</tr>
<tr>
<td>500 or more</td>
<td>3.35</td>
<td>268</td>
<td>536</td>
<td>804</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>36,439</strong></td>
<td><strong>72,878</strong></td>
<td><strong>109,317</strong></td>
<td></td>
</tr>
</tbody>
</table>

Source: SaS analysis
TOTAL NUMBER OF NEEDED SERVERS AS WELL AS OF SYSTEM ADMIN GROWS FAST

<table>
<thead>
<tr>
<th>Number of servers at end of year</th>
<th>1,000 monitoring points</th>
<th>37</th>
<th>73</th>
<th>110</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active data processing servers</td>
<td>1 every</td>
<td>10</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>Back-ups</td>
<td>1 each</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td>41</td>
<td>81</td>
<td>121</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Load balancer</th>
<th>8 active servers</th>
<th>5</th>
<th>10</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>Back-ups</td>
<td>1 every</td>
<td>10</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Application server</td>
<td></td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Web server</td>
<td></td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Staging environment</td>
<td></td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>File server</td>
<td></td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Database server</td>
<td></td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td>58</td>
<td>103</td>
<td>148</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Number of sys admin</th>
<th>6</th>
<th>11</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td>44</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Sun Solaris</td>
<td>6.5</td>
<td>10</td>
<td>17</td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td>24</td>
</tr>
</tbody>
</table>
REQUIRED STORAGE CAPACITY IS IMPRESSIVE

<table>
<thead>
<tr>
<th>Required storage capacity at end of year</th>
<th>1 file every 5 minutes</th>
<th>Number of files sent /year</th>
<th>1,915,242,250</th>
<th>5,745,726,749</th>
<th>9,576,211,248</th>
</tr>
</thead>
<tbody>
<tr>
<td>File dimension</td>
<td>6 Kb</td>
<td>TOTAL needed capacity</td>
<td>12 Tera b</td>
<td>35 Tera b</td>
<td>58 Tera b</td>
</tr>
</tbody>
</table>

\[ \text{Required storage capacity at end of year} = (365 \text{ days} \times 24 \text{ hours} \times 60 \text{ minutes} / 5 \text{ minutes}) \times \text{average number of monitoring points during year} \]
## Required sales people

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>Online contracts</th>
<th>Manual contracts</th>
<th>Total contracts</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>22,621</td>
<td>45,241</td>
<td>67,862</td>
</tr>
<tr>
<td>Required sales people</td>
<td>90%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2,513</td>
<td>5,027</td>
<td>7,540</td>
</tr>
<tr>
<td>Sales needed</td>
<td>1 for 200 deals</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>13</strong></td>
<td><strong>26</strong></td>
<td><strong>38</strong></td>
</tr>
</tbody>
</table>

Source: SaS analysis
FULL TIME EQUIVALENTS ALSO DETERMINES TOTAL NUMBER OF REQUIRED LAPTOPS

<table>
<thead>
<tr>
<th>Total number of people at end of year</th>
<th>IT</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Windows</td>
<td>Linux</td>
<td>Sun Solaris</td>
</tr>
<tr>
<td>CEO and secretaries</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Technical services</td>
<td>5</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>Sales &amp; Marketing</td>
<td>13</td>
<td>26</td>
<td>38</td>
</tr>
<tr>
<td>Legal</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Support</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Product development</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td><strong>35</strong></td>
<td><strong>59</strong></td>
<td><strong>80</strong></td>
</tr>
<tr>
<td><strong>Laptops</strong></td>
<td><strong>31</strong></td>
<td><strong>51</strong></td>
<td><strong>69</strong></td>
</tr>
<tr>
<td><strong>Sun Solaris</strong></td>
<td><strong>39</strong></td>
<td><strong>65</strong></td>
<td><strong>89</strong></td>
</tr>
</tbody>
</table>

3 possible scenarios pending to the technology adopted by NewCo

Source: SaS analysis
## COST OF IT: SCENARIO “MS WINDOWS”

<table>
<thead>
<tr>
<th>COST OF IT (in US $)</th>
<th>SCENARIO 1 : MS WINDOWS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Monitoring PDA</strong></td>
<td></td>
</tr>
<tr>
<td>Cost of average Windows PDA</td>
<td>295.0^1</td>
</tr>
<tr>
<td>Cost of PDAs</td>
<td>10,749,552 10,749,552 10,749,552</td>
</tr>
<tr>
<td><strong>Servers</strong></td>
<td></td>
</tr>
<tr>
<td>&quot;Basic server&quot;</td>
<td>8,000.0^2</td>
</tr>
<tr>
<td>Database server</td>
<td>16,000.0^3</td>
</tr>
<tr>
<td>Windows server 2003</td>
<td>4,000.0^4</td>
</tr>
<tr>
<td>(enterprise edition)</td>
<td></td>
</tr>
<tr>
<td>Additional licenses</td>
<td>1,600.0</td>
</tr>
<tr>
<td>Windows client access license (20)</td>
<td>799.0</td>
</tr>
<tr>
<td>Oracle database</td>
<td>960,000.0</td>
</tr>
<tr>
<td>Basic servers</td>
<td>56 45 45</td>
</tr>
<tr>
<td>Cost of basic servers</td>
<td>448,000 360,000 360,000</td>
</tr>
<tr>
<td>License</td>
<td>92,799 72,000 72,000</td>
</tr>
<tr>
<td>Database servers</td>
<td>2 0 0</td>
</tr>
<tr>
<td>Cost of database servers</td>
<td>32,000 0 0</td>
</tr>
<tr>
<td>License</td>
<td>960,000.0</td>
</tr>
<tr>
<td>Storage array</td>
<td>13 6 60,000.0^7</td>
</tr>
<tr>
<td>Cost (including license)</td>
<td>1 2 1</td>
</tr>
<tr>
<td>Number of sys admin</td>
<td>6 11 15</td>
</tr>
<tr>
<td>Cost of sys admin / year</td>
<td>68,000.0 408,000 748,000 1,020,000</td>
</tr>
<tr>
<td><strong>Office</strong></td>
<td></td>
</tr>
<tr>
<td>Number of Laptops</td>
<td>3,100.0^8</td>
</tr>
<tr>
<td></td>
<td>40 24 21</td>
</tr>
<tr>
<td></td>
<td>124,000 74,400 65,100</td>
</tr>
<tr>
<td><strong>TOTAL COST OF IT</strong></td>
<td></td>
</tr>
<tr>
<td>CapEx</td>
<td>12,470,351 11,379,952 11,310,652</td>
</tr>
<tr>
<td>OpEx</td>
<td>408,000 748,000 1,020,000</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td>12,878,351 12,127,952 12,330,652</td>
</tr>
</tbody>
</table>

Source: SaS analysis
1. HP iPAQ 4155 Pocket PC = 450 $ for one, including cost of Windows CE license. Assumption: Ordering huge quantities can reduce price by as much as 50% --> 225 $, + 70 US $ adaptation = 295 $

2. 2 CPUs at 2 Ghz, 4 Gb RAM, 2 X 36 Gb HD at 15,000 rpm, 2 controllers RAID. Dell: about 12,000 US $ each. Assumption: In case of high volume buying, up to 30% discount --> 8,000$

3. 4 CPUs at 2 Ghz, 12 Gb RAM, 2 X 73 Gb HD at 15,000 rpm, 2 controllers RAID. Dell: 25,000 US $. Assumption: In case of high volume buying, discount up to 35% --->16,000 US $


5. How to calculate the license cost of Oracle 10g database Enterprise edition? "Follow the instructions below to calculate the minimum number of named user plus licenses required for your intended hardware configuration.
   • Determine the number of processors on each server where the programs are installed and/or running.
   • Add together the processors on each server.
   • Multiply the total number of processors by n (the number of users).
   • The resultant number represents the minimum number of named user plus licenses required for this hardware configuration."
   Click "Add to cart" and get the result
   4 processors on each server, 2 servers = 8
   4 users --> 4 * 8 = 32
   input 32 in the form ==> 960,000 US $

6. Tera byte

7. Dell EMC CX300, 13 Tb

8. Dell Latitude 600 1.7 Ghz, 1 Gb RAM, 40 Gb HD, Microsoft Windows, Microsoft Office Professional
# COST OF IT: SCENARIO “LINUX”

<table>
<thead>
<tr>
<th>COST OF IT (in US $)</th>
<th>SCENARIO 2 : OPEN SOURCE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Y1</td>
</tr>
<tr>
<td>Monitoring PDA</td>
<td></td>
</tr>
<tr>
<td>Cost of average Linux PDA</td>
<td>223.5</td>
</tr>
<tr>
<td>Cost of PDAs</td>
<td>8,144,152</td>
</tr>
<tr>
<td><strong>Servers</strong></td>
<td></td>
</tr>
<tr>
<td>&quot;Basic server&quot;</td>
<td>8,000.0</td>
</tr>
<tr>
<td>Database server</td>
<td>16,000.0</td>
</tr>
<tr>
<td>Linux Debian distribution</td>
<td>0.0</td>
</tr>
<tr>
<td>MySQL</td>
<td>0.0</td>
</tr>
<tr>
<td>Basic servers</td>
<td></td>
</tr>
<tr>
<td>Cost of basic servers</td>
<td>448,000</td>
</tr>
<tr>
<td>License</td>
<td>0.0</td>
</tr>
<tr>
<td>Database servers</td>
<td></td>
</tr>
<tr>
<td>Cost of database servers</td>
<td>32,000</td>
</tr>
<tr>
<td>License</td>
<td>0.0</td>
</tr>
<tr>
<td>Storage array</td>
<td>60,000.0</td>
</tr>
<tr>
<td>Cost (including license)</td>
<td>60,000</td>
</tr>
<tr>
<td>Number of sys admin</td>
<td>2</td>
</tr>
<tr>
<td>Cost of sys admin / year</td>
<td>71,000.0</td>
</tr>
<tr>
<td><strong>Office</strong></td>
<td></td>
</tr>
<tr>
<td>Number of Laptops</td>
<td>2,320.0</td>
</tr>
<tr>
<td>Training of staff (new GUI, OpenOffice, etc.)</td>
<td></td>
</tr>
<tr>
<td>Loss of productivity</td>
<td>100.0</td>
</tr>
<tr>
<td>Loss of productivity</td>
<td>600.0</td>
</tr>
<tr>
<td><strong>TOTAL COST OF IT</strong></td>
<td></td>
</tr>
<tr>
<td>CapEx</td>
<td>8,795,752</td>
</tr>
<tr>
<td>OpEx</td>
<td>161,200</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td>8,956,952</td>
</tr>
</tbody>
</table>

Source: SaS analysis
1. Royal Linux PDA: 219 $. Assumption: Ordering huge quantities can reduce price by as much as 30% --> 153.5 $. Adaptation: + 70 $ = 223.5 $

2. 2 CPUs at 2 Ghz, 4 Gb RAM, 2 X 36 Gb HD at 15,000 rpm, 2 controllers RAID. Dell: about 12,000 US $ each. Assumption: In case of high volume buying, up to 30% discount --> 8,000$

3. 4 CPUs at 2 Ghz, 12 Gb RAM, 2 X 73 Gb HD at 15,000 rpm, 2 controllers RAID. Dell: 25,000 US $. Assumption: In case of high volume buying, discount up to 35% -->16,000 US $

4. Linux Debian is free

5. MySQL is free

6. Tera byte

7. Dell EMC CX300, 13 Tb

8. Dell Latitude 600 1.6 Ghz, 768 Mb RAM, 40 Gb HD = 2420 $ if basic Windows installed (no other software). Assumption: - 100 $ because no Windows OS needed = 2320 $
**COMPARISON OF SCENARIOS**

**USD**

<table>
<thead>
<tr>
<th></th>
<th>Y1</th>
<th>Y2</th>
<th>Y3</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MS scenario</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cash-out</td>
<td>12,878,351</td>
<td>12,127,952</td>
<td>12,330,652</td>
</tr>
<tr>
<td><strong>Cumulated Cash-out</strong></td>
<td>37,336,956</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discount rate</td>
<td>11.5%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PV</td>
<td>11,550,091</td>
<td>9,755,235</td>
<td>8,895,317</td>
</tr>
<tr>
<td><strong>Cumulated PV</strong></td>
<td>30,200,643</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Open Source Scenario</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cash-out</td>
<td>8,956,952</td>
<td>8,904,152</td>
<td>8,904,552</td>
</tr>
<tr>
<td><strong>Cumulated Cash-out</strong></td>
<td>26,765,657</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discount rate</td>
<td>11.5%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PV</td>
<td>8,033,141</td>
<td>7,162,141</td>
<td>6,423,733</td>
</tr>
<tr>
<td><strong>Cumulated PV</strong></td>
<td>21,619,015</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| delta cash-out      | 10,571,299|
| delta discounted    | 8,581,629 |

Thank to Open Source software this project could save as much as US $ 8.6 m in accumulated IT expenditures over 3 years.

Source: SaS analysis
## REVISION HISTORY

<table>
<thead>
<tr>
<th>Release</th>
<th>Date</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>v. 1.0</td>
<td>May 30, 2004</td>
<td>• First private release, only for a limited number of journalists. Legal notice not optimized yet.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Document not for distribution/redistribution</td>
</tr>
<tr>
<td>v. 1.0</td>
<td>June 1, 2004</td>
<td>• Document announced on SPIP mailing list on June 1, 2004</td>
</tr>
<tr>
<td>v. 1.0.1</td>
<td>June 2, 2004</td>
<td>• Slight ammendement to the Legal Notice</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Slide « Open Source software market share » (p. 86) corrected</td>
</tr>
<tr>
<td>v.1.0.2</td>
<td>June 7, 2004</td>
<td>• Status of Xindice (XML database) corrected (p. 121)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Update of the Linux debian developers location (p. 36)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Update of partial migration cases (p. 190)</td>
</tr>
<tr>
<td>v.1.0.3</td>
<td>June 9, 2004</td>
<td>• Update of list of companies using Open Source Software (p. 83)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Modification of « Legal Notice » (p. 2)</td>
</tr>
<tr>
<td>v.1.0.4</td>
<td>July 24, 2004</td>
<td>• Update of list of companies using Open Source Software (p. 83)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Correction of misspellings</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Format corrections</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Creation of page 191</td>
</tr>
<tr>
<td>v.1.0.5</td>
<td>October 10, 2004</td>
<td>• Update of the Linux debian developers location (p. 36)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Update of multi-criteria comparison (p. 72)</td>
</tr>
<tr>
<td>v.1.0.6</td>
<td>November 16, 2004</td>
<td>• Correction of misspellings</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Update of the number of projects hosted by SourceForge (p. 34)</td>
</tr>
<tr>
<td>v.1.0.7</td>
<td>December 3, 2004</td>
<td>• Add definition of Hacker in the Definitions page (p. 5)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Duplication of the Definitions page (now definitions on pages 5 and 6)</td>
</tr>
<tr>
<td>v.1.0.8</td>
<td>February 4, 2005</td>
<td>• Correction of misspellings</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• New example (p. 127)</td>
</tr>
<tr>
<td>v.1.1</td>
<td>March 15, 2005</td>
<td>• Complete spelling and grammar review</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Inclusion of comments from Sheldon Lee-Wen all over the document</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Update of certain figures</td>
</tr>
<tr>
<td>v.1.1.1</td>
<td>April 4, 2005</td>
<td>• Update of list of companies using Open Source Software (pp. 84-85)</td>
</tr>
</tbody>
</table>