

Introduction to open source software

Sebastian Rahtz

OSS Watch

May 2005, Timor Leste



I am Sebastian Rahtz:

- Information Manager for *Oxford University Computing Services*
- Manager of *OSS Watch*, the UK national Open Source Advisory Service

`sebastian.rahtz@oucs.ox.ac.uk`

OSS Watch provides unbiased advice and guidance about free and open source software for UK further and higher education.

Technically, what does ‘Open Source’ mean?

Software for which:

- the source code is **available** to the end-user;
- the source code can be **modified** by the end-user;
- there are no restrictions on **redistribution** or use;
- the **licensing conditions** are usually intended to *facilitate* continued re-use and wide availability of the software, in both commercial and non-commercial contexts;
- the **cost of acquisition** to the end-user is often minimal.

Technically, what does 'Open Source' mean?

Software for which:

- the source code is **available** to the end-user;
- the source code can be **modified** by the end-user;
- there are no restrictions on **redistribution** or use;
- the **licensing conditions** are usually intended to *facilitate* continued re-use and wide availability of the software, in both commercial and non-commercial contexts;
- the **cost of acquisition** to the end-user is often minimal.

Technically, what does ‘Open Source’ mean?

Software for which:

- the source code is **available** to the end-user;
- the source code can be **modified** by the end-user;
- there are no restrictions on **redistribution** or use;
- the **licensing conditions** are usually intended to *facilitate* continued re-use and wide availability of the software, in both commercial and non-commercial contexts;
- the **cost of acquisition** to the end-user is often minimal.

Technically, what does ‘Open Source’ mean?

Software for which:

- the source code is **available** to the end-user;
- the source code can be **modified** by the end-user;
- there are no restrictions on **redistribution** or use;
- the **licensing conditions** are usually intended to *facilitate* continued re-use and wide availability of the software, in both commercial and non-commercial contexts;
- the **cost of acquisition** to the end-user is often minimal.

Technically, what does ‘Open Source’ mean?

Software for which:

- the source code is **available** to the end-user;
- the source code can be **modified** by the end-user;
- there are no restrictions on **redistribution** or use;
- the **licensing conditions** are usually intended to *facilitate* continued re-use and wide availability of the software, in both commercial and non-commercial contexts;
- the **cost of acquisition** to the end-user is often minimal.

Virtues of free and open source software

- It has no secrets: the innards are available for anyone to inspect
- It is not privately controlled: so likely to promote open rather than proprietary formats
- It is *typically* maintained by communities rather than single corporations: so bug fixes and enhancement are often frequent and free
- It is usually distributed free of charge (developers make their money from support, training, customisation and specialist add-ons; not marketing)



Virtues of free and open source software

- It has no secrets: the innards are available for anyone to inspect
- It is not privately controlled: so likely to promote open rather than proprietary formats
- It is *typically* maintained by communities rather than single corporations: so bug fixes and enhancement are often frequent and free
- It is usually distributed free of charge (developers make their money from support, training, customisation and specialist add-ons; not marketing)



Virtues of free and open source software

- It has no secrets: the innards are available for anyone to inspect
- It is not privately controlled: so likely to promote open rather than proprietary formats
- It is *typically* maintained by communities rather than single corporations: so bug fixes and enhancement are often frequent and free
- It is usually distributed free of charge (developers make their money from support, training, customisation and specialist add-ons; not marketing)



Virtues of free and open source software

- It has no secrets: the innards are available for anyone to inspect
- It is not privately controlled: so likely to promote open rather than proprietary formats
- It is *typically* maintained by communities rather than single corporations: so bug fixes and enhancement are often frequent and free
- It is usually distributed free of charge (developers make their money from support, training, customisation and specialist add-ons; not marketing)



Clearing up misunderstandings

- **Free** software is not the same as **free beer**. Open source software may or may not cost money
- The cost of **ownership** often bears little relation to the cost of acquiring a piece of software
- Open source software is strictly in the legal framework of copyright and licensing. It is not about pirating software, or giving things away
- Open source software does not mandate **exclusivity**. You can use open source programs under Windows (eg The OpenCD)
- People do not choose software solely on the basis of open source. **Interoperability and open standards** for data are equally important



Clearing up misunderstandings

- **Free** software is not the same as **free beer**. Open source software may or may not cost money
- The cost of **ownership** often bears little relation to the cost of acquiring a piece of software
- Open source software is strictly in the legal framework of copyright and licensing. It is not about pirating software, or giving things away
- Open source software does not mandate **exclusivity**. You can use open source programs under Windows (eg The OpenCD)
- People do not choose software solely on the basis of open source. **Interoperability and open standards** for data are equally important



Clearing up misunderstandings

- **Free** software is not the same as **free beer**. Open source software may or may not cost money
- The cost of **ownership** often bears little relation to the cost of acquiring a piece of software
- Open source software is strictly in the legal framework of copyright and licensing. It is not about pirating software, or giving things away
- Open source software does not mandate **exclusivity**. You can use open source programs under Windows (eg The OpenCD)
- People do not choose software solely on the basis of open source. **Interoperability and open standards** for data are equally important



Clearing up misunderstandings

- **Free** software is not the same as **free beer**. Open source software may or may not cost money
- The cost of **ownership** often bears little relation to the cost of acquiring a piece of software
- Open source software is strictly in the legal framework of copyright and licensing. It is not about pirating software, or giving things away
- Open source software does not mandate **exclusivity**. You can use open source programs under Windows (eg The OpenCD)
- People do not choose software solely on the basis of open source. **Interoperability and open standards** for data are equally important



Clearing up misunderstandings

- **Free** software is not the same as **free beer**. Open source software may or may not cost money
- The cost of **ownership** often bears little relation to the cost of acquiring a piece of software
- Open source software is strictly in the legal framework of copyright and licensing. It is not about pirating software, or giving things away
- Open source software does not mandate **exclusivity**. You can use open source programs under Windows (eg The OpenCD)
- People do not choose software solely on the basis of open source. **Interoperability and open standards** for data are equally important



The well-known examples

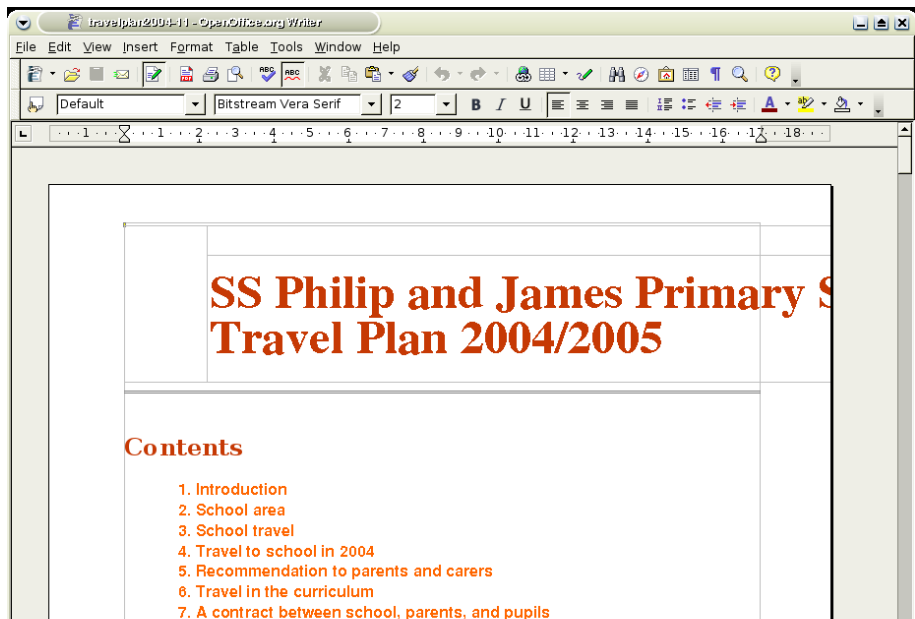
- Linux operating system
- Apache web server
- Mozilla web browser and email client
- Perl, Python and PHP scripting languages
- MySQL database
- OpenOffice office suite
- uPortal portal framework
- Gimp picture editor
- Moodle Virtual Learning Environment
- Ubuntu desktop distribution

in the education sector, open source offerings exist in most areas of networked services and end-user applications. MIS applications are less mature.

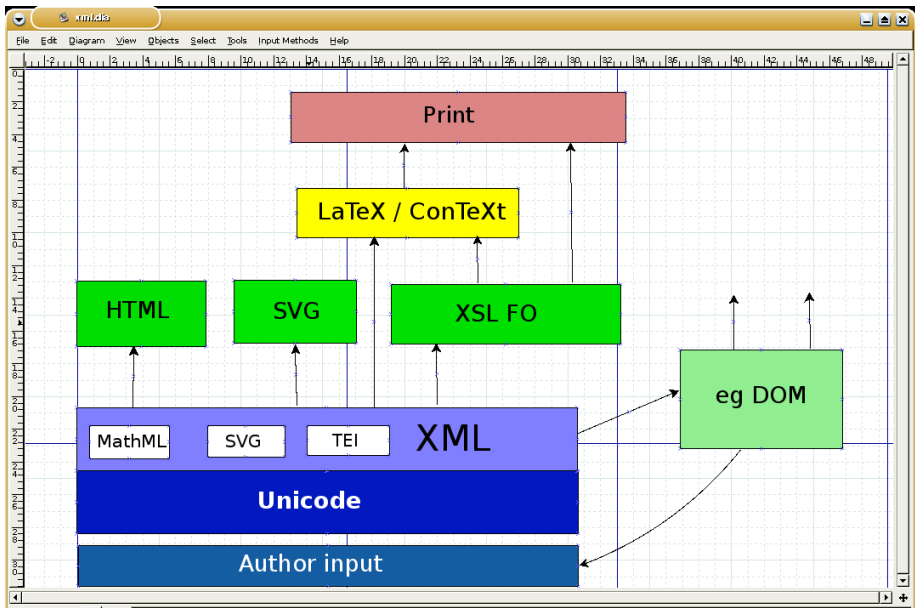
Desktops?

The image displays a collage of various open-source software interfaces:

- Spreadsheet:** A window showing a table with columns A-J and rows 1-13. The data includes years and values, with a line graph overlaid showing a trend from 1800 to 2000.
- Paintbrush:** A window titled "Paintbrush" showing a list of files and folders, including "5-03-16-rscu", "no.pdf", "uide.to.Open", "dance-Draft-", "ges/", "neryary.txt", "cpolicy.pdf", "cpolicy.txt", "cpolicy.xml", "cpolicy.xml", "n Source in", "policy_vers", "gmail.doc", "gmail.txt", "des/", "des.css", "des.js", "desP5.xsl", "desP5.xsl", "des.xsl", "ks/", "qy-lba"]", and "Rus ding file://".
- Music Player:** A window titled "X MULTIMEDIA SYSTEM" showing a digital display of "00:24", "1 CARTER/ IN NEW YORK - SILENCE", and "100 kbps 44 kHz". It includes playback controls and a "SEE DISPLAY" button.
- Web Browser:** A window showing the TEI Consortium website, including a navigation menu with links like "Software", "History", "FAQs", "PS", "Consortium", "Activities", "SIGs", "Join in/Contact", and "Members area". The main content area is titled "TEI home page" and "TEI tools".



Diagrams?



Games?



Email?

The screenshot shows the Debian Thunderbird email client. The top menu bar includes File, Edit, View, Go, Message, Enigmail, Tools, and Help. Below the menu is a toolbar with icons for Get Mail, Write, Address Book, Decrypt, Reply, Reply All, Forward, Delete, Junk, Print, and Stop. The left sidebar shows the Folders pane with a tree view: Herald (Inbox), Local Folders (Inbox (3), Unsent Messages (5), Drafts, Templates, Sent, Trash (8)), and a list of other folders like ctan, Inbox2, Inbox3, New, Test, Timor4, toFeb04, Triedtosend (162), Unsent-Fri...6 BST 2005, Unsent-Th...2005 (20), Unsent-Tue...2 BST 2005, and Unsent-We...005 (18). The main pane displays a list of emails in the inbox, sorted by Date. The selected email is "Re: Approaching notes in TEI" by Syd Bauman, dated 01:01. The email content is visible in the bottom pane, showing a discussion about TEI Guidelines and markup.

Subject: Re: Approaching notes in TEI

From: [Syd Bauman <Syd_Bauman@BROWN.EDU>](mailto:Syd_Bauman@BROWN.EDU)

Reply-To: Syd_Bauman@BROWN.EDU

Date: 01:01

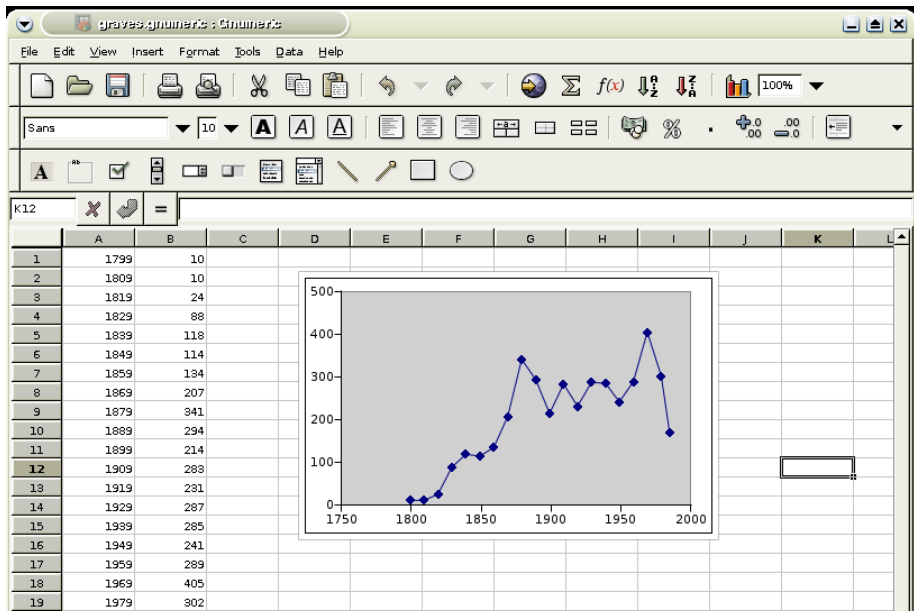
To: TEI-L@LISTSERV.BROWN.EDU

I think there is nothing wrong with your suggested <noteSubj>. On the other hand, if you want any kind of interoperability with generic TEI tools (of which, I know, there are very few at the moment), you would do much better to stick with either in-line <note> elements or out of line <note> elements that point to markup. Furthermore using <note>s makes your encoding clear to any reader even mildly familiar with the TEI Guidelines.

First, the descriptions of <note> in P4 pretty clearly identify it as something to be used to mark existing notes

As Lou has pointed out, this isn't really the intent, and I think you've thoroughly demonstrated that a new example of a modern editor's <note> is called for in P5.

Spreadsheet?



Virtual Learning Environment?

The screenshot shows the Moodle website interface within a Mozilla Firefox browser window. The browser's address bar displays 'http://moodle.org/'. The Moodle logo, featuring a graduation cap and the word 'moodle' in a stylized orange font, is prominently displayed at the top. To the right of the logo, a user login status indicates 'You are logged in as: Guest User (Login)' and a language dropdown menu is set to 'English (en)'. The main content area is divided into several sections: 'Main Menu' on the left with links like 'Free Support', 'Download Moodle', and 'Languages'; 'My courses' in the center with links to 'Community Discussion (in English)' and 'Community Discussion (in other languages)'; and a right-hand column with descriptive text about Moodle as a course management system (CMS) and its open-source nature. A 'Latest News' section at the bottom left mentions a milestone of reaching 100 countries.

Moodle - A Free, Open Source Course Management System for Online Learning - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://moodle.org/

You are logged in as: Guest User (Login)

English (en)

Main Menu

- Free Support
- Download Moodle
- Languages
- Modules and add-ons
- Frequently Asked Questions
- Documentation
- Development
- Donations
- Ratings
- Moodle Buzz
- Moodle Sites

My courses

- Community Discussion (in English)**
 - Using Moodle
 - Moodle Conferences
 - Moodle Documentation Project
 - Moodle Exchange
 - Moodle for Business Uses
 - Moodle for Language Teaching
- Community Discussion (in other languages)**
 - مoodle العربي
 - ???? Moodle
 - Deutschsprachiges Moodle
 - Finnish Moodle

Moodle is a course management system (CMS) - a software package designed to help educators create quality online courses. Such e-learning systems are sometimes also called Learning Management Systems (LMS) or Virtual Learning Environments (VLE). One of the main advantages of Moodle over other systems is a strong grounding in social constructionist pedagogy.

Moodle is Open Source software, which means you are free to download it, use it, modify it and even distribute it (under the terms of the GNU General Public License). Moodle runs without modification on Unix, Linux, Windows, Mac OS X, Netware and any other system that supports

Latest News

7 Dec, 15:04 - Martin Dougiamas
Moodle reaches 100 countries - 92 to

The current practical picture

Browser	Good
Desktop OS	Acceptable
Content Management Systems	Good
Digital library services	Good
Email	Good
Integrated groupware	Acceptable
Library catalogues	Weak
Network services	Good
Office suite	Acceptable
Payroll	Weak
Scientific workstation	Good
Student administration	Weak
VLE & portal	Good

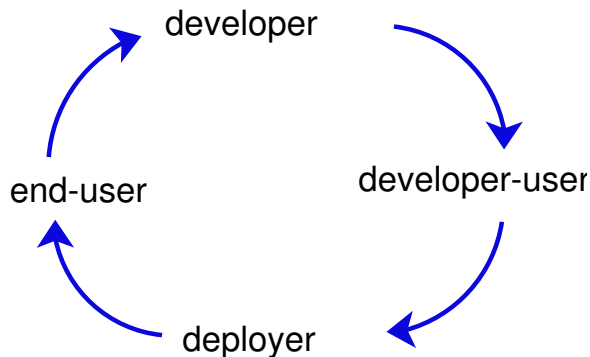
Being *possible* does not make it the *right choice*



Open source is a development methodology

- **Programmer commitment**, because the programmers work on the software they **need**
- **Rapid change**, because programmers want to see results
- **Unconstrained specifications**, because there is no external client
- **Collective responsibility** for the code
- **Response to change**, dictated by (perhaps unexpected) use

Open source is about community



Those who ‘merely’ deploy open source software are also part of the open source community

Why do people keep working on open source?

The desire to learn technical skills by joining an open project is strong. Typical reasons for staying in open source are:

- improving skills: 32%
- ideology: 31%
- seeking recognition: 12%
- improving software: 24%

Why do companies work in open source?

- implementation of open standards might as well be done in a shared way to save costs
- pyramidal consulting works: using open software means that your support team are spared the 80% of questions which are easy, leaving you the remaining 20%
- making 'needed improvements' only to open source is economically efficient. Work on the things you care about
- the revenue margin on licences is 85%, on support 54%; eg IBM and Novell are now depending more on services than licensing

Open source is about conflict and change

Our industry has an almost totally monopolistic provider:

	Good?	Bad?
.NET	Single framework	No room for choice
Office	It does everything	It has insufficient security barriers
Windows	A smooth upgrade path	No room for innovation

It is hard to decide whether this is good or bad

Why should Timor care about open source?

- ➊ *Saving money.* You can build good systems without paying a lot of money. Everyone can have a copy.
- ➋ *Training.* Timor needs good software engineers. Open source lowers the barrier to getting involved in real development
- ➌ *Localisation.* Timor has its own languages and culture, which should be reflected in the software used in schools and businesses. Microsoft will not write a spell checker for Fataluco.
- ➍ *Control.* Use software you can influence the direction of, which does not leave you at the mercy of big companies.
- ➎ *Community.* Timorese people can work as equals on projects around the world, and influence the future.



Why should Timor care about open source?

- ➊ *Saving money.* You can build good systems without paying a lot of money. Everyone can have a copy.
- ➋ *Training.* Timor needs good software engineers. Open source lowers the barrier to getting involved in real development
- ➌ *Localisation.* Timor has its own languages and culture, which should be reflected in the software used in schools and businesses. Microsoft will not write a spell checker for Fataluco.
- ➍ *Control.* Use software you can influence the direction of, which does not leave you at the mercy of big companies.
- ➎ *Community.* Timorese people can work as equals on projects around the world, and influence the future.



Why should Timor care about open source?

- ➊ *Saving money.* You can build good systems without paying a lot of money. Everyone can have a copy.
- ➋ *Training.* Timor needs good software engineers. Open source lowers the barrier to getting involved in real development
- ➌ *Localisation.* Timor has its own languages and culture, which should be reflected in the software used in schools and businesses. Microsoft will not write a spell checker for Fataluco.
- ➍ *Control.* Use software you can influence the direction of, which does not leave you at the mercy of big companies.
- ➎ *Community.* Timorese people can work as equals on projects around the world, and influence the future.



Why should Timor care about open source?

- ➊ *Saving money.* You can build good systems without paying a lot of money. Everyone can have a copy.
- ➋ *Training.* Timor needs good software engineers. Open source lowers the barrier to getting involved in real development
- ➌ *Localisation.* Timor has its own languages and culture, which should be reflected in the software used in schools and businesses. Microsoft will not write a spell checker for Fataluco.
- ➍ *Control.* Use software you can influence the direction of, which does not leave you at the mercy of big companies.
- ➎ *Community.* Timorese people can work as equals on projects around the world, and influence the future.



Why should Timor care about open source?

- ① *Saving money.* You can build good systems without paying a lot of money. Everyone can have a copy.
- ② *Training.* Timor needs good software engineers. Open source lowers the barrier to getting involved in real development
- ③ *Localisation.* Timor has its own languages and culture, which should be reflected in the software used in schools and businesses. Microsoft will not write a spell checker for Fataluco.
- ④ *Control.* Use software you can influence the direction of, which does not leave you at the mercy of big companies.
- ⑤ *Community.* Timorese people can work as equals on projects around the world, and influence the future.

