

Open Source Software vs Close Source Software

2.3 Open Source Software

Open source software can be defined as software distributed under a licensing agreement which allows the source code (computer code) to be shared, viewed and modified by other users and organizations^[2]. The open source software contains certain criteria such as:

1. Free Redistribution
2. Source Code
3. Derived Works
4. Integrity of The Author's Source Code
5. No Discrimination Against Persons or Groups
6. No Discrimination Against Fields of Endeavor
7. Distribution of License
8. License Must Not Be Specific to a Product
9. License Must Not Restrict Other Software
10. License Must Be Technology-Neutral

2.4 Close Source Software

Close Source Software is machine programming authorized under the selective trustworthy advantages associated with the copyright holder. Close Source Software is created by an individual or firm who has privileges of utilizing pre-existing or flourishing brand new equipment to make new programming. The principle character of Close Source Software is that its source code is dependably kept a mystery from the outside world. Accordingly, the inside structure of exclusive programming will be not revealed. The Confinements on restrictive programming are by and large forced through a report called EULA (End-user license agreement) to which clients will be suspected to concur just before utilizing the programming. It meets expectations, simply like a written agreement of good use, ailments between the client and merchant.

2.5 Public Domain Software

The copyright holder gives it to people in general. Subsequently, it is no more copyrighted and limited and accordingly moves towards Open source.

2.6 Trial Ware

It is that kind of software which one is restricted as far as the duration of the time to utilization. The client receives the completed form of programs which may be restricted to the

use number of days or use a number of times. In the wake of paying and getting enrolled, the client will get limitless access to the utilization and redesigns of that system.

3. OPEN SOURCE SOFTWARE AND CLOSE SOURCE SOFTWARE

Plenty Of people all over the world choose to utilize Close Source Software instead of Open Source Software mainly because of the misguided judgment that Close Source Software is preferred underpinned over the open source software. After a few years of promoting both open source software and Close Source Software, it gets to be obviously recognizable that simply in light of the fact that you pay for Close Source Software does not imply that encouraging that product is any simpler; indeed, there are a lot of reasons why supporting Open Source Software is genuinely significantly less demanding. We will now focus on the OSS and Close Source software for the purpose of this paper evaluate OSS and Close Source software. There has been significant discussion and debate with respect towards pros and cons of open source Software and Close Source software. Understanding the benefits and pitfalls of both types of applications can support an individual to make a lot more well informed conclusion. The objective of this particular document is usually to provide a better perception of distinctions between open source and closed source (Close Source) software, and the advantages and disadvantages of each to enable a more informed decision making process when it comes to choosing between the two

Table 1- Open Source Software Versus Close Source Software

Factor	Open Source Software	Close Source Software
Cost	Open source software is free. This is a vast draw card, and if your in-house capacities are such that you can actualize, prepare and help at least expense to your association it might be an alluring alternative. Open source software requires a certain level of specialized skill to oversee content. It is along these lines vital to take into attention the expenses acquired once the product is gotten may be significant unless the assets are now set up to oversee it [4].	The expense of Close Source software will differ from a couple of thousand to a couple hundred thousand Rupees (INR), contingent upon the multifaceted nature of the framework needed. This expense is comprised of a base charge for programming, combination and administrations and yearly permitting/help charges. This expense may be restrictive for some; however, what the client is paying for being a more tweaked item from a trusted brand that incorporates larger amounts of security and usefulness, nonstop advancement, a more prominent capacity to scale, progressing preparing and packing and a lower prerequisite for specialized abilities. With a specific end goal to backing these components and keep up high accessibility sites, an instrument must be set up to recover the expenses. As specified formerly, progressively open source programming suppliers are moreover charging for additional items, extra administrations, and combination, lessening the crevice in cost between the two choices
Service and Support	Administration is one of the key issues with respect to open source programming. Open source software depends on its online group system to convey help by means of discussions forums and blogs. While there are huge, steadfast and captivated online groups that clients can turn to, time-poor customers of today are acquainted with the prompt administration and help that empowers issues to be determined in an auspicious way, and these groups can't promise the abnormal state of responsive.	In the event that the web is a vital channel for an association, Software is regularly an optional concern, with administration level and help structure prerequisites taking a point of reference for augmenting up-time and minimizing downtime. The administration is likely the best focal point of utilizing exclusive programming. Close Source software suppliers offer progressing backing to clients, a key offering point for clients without specialized mastery.
Innovation	Open source programming empowers innovation	Close Source software suppliers don't permit clients to

	by furnishing users with the opportunity also adaptability to adjust the product to suit, without confinement. Then again, innovation could conceivably be passed on to all clients or users of the software. It is a client's wish whether they wish to impart their innovation with any online groups, community and users must be eagerly participating in these groups to become conscious of such innovation. It has been deliberated about whether redid progressions to the original source code limit the future support and development of the product, as these can possibly bring about a constrained capacity to apply future redesigns, fixes or modules went for enhancing the product, leaving the user with a form that may have irresolvable issues. It is important to note that open source software suppliers generally fight to attract in large scale R&D.	view or adjust the source code. While this may be seen as a hindrance to some, it guarantees the security and unwavering quality of the product. Moreover, numerous exclusive programming suppliers customize software for particular clients to give more adaptability while putting resources into Research & Development to frequently offer new items and redesigns. Besides, Close Source software suppliers have online client groups that make esteem by imparting thoughts, procedures and best practices through critical systems, for example, gatherings and reviews, which likewise encourage advancement and permit the item to adjust with evolving needs. This advancement comes completely tried, and is accessible to all clients on the product. It doesn't oblige interest in Research & development or the specialized understanding of source code, and help with execution is by and large piece of the bundle. Since merchants must guarantee their product does not get to be excess, clients likewise profit from the sort of focused on advancement attempted persistent interest in R&D instead of "development for advancement's purpose", business concentrated instead of innovation centered
Usability	Open source software has been exceptionally reprimanded for its absence of ease of use. Generally, the innovation is not inspected by ease of use specialists and does not gratify to the larger part of the computer user. Open source software is generally designer driven, and without framework organization experience or the learning needed to control programming language, utilization of the software and capability to alter bugs as they emerge is regularly restricted to those with specialized ability. Moreover, open source software does not lawfully oblige documentation, Such as, user manuals or aides, upsetting the formation of such devices. At the point when documentation is given, it is frequently general, containing understood language, inhibiting learning [4]. Without adequate documentation users must depend on option means such as, online groups, community, discussion forums, accepting they are able to do, discovering them and the issue is one others have experienced or are ready to help resolve. These hindrances are impediments, yet are not inconceivable	Close Source software generally makes use of expert functionality, evaluating, and also as the software program is frequently geared towards a much more focused viewers, and as a consequence additional individualized, usability is usually positioned extremely high. In extension, comprehensive user instructions as well as manuals are supplied. This permits more rapidly working out and supplies an instantaneous referral, enabling users to maneuver alongside discovering curve much more rapidly Boosting solutions consist of training seminars, focused courses as well as extended assistance to aid maximize utilization of the software. It is additionally very important to keep in mind that even though many individuals witness Close Source software as —closed, today's Close Source software comes with a massive variety of components for the advancement through alternative techniques as well as developers

Security	Open source software is not necessarily developed in a controlled environment. While big players often have a concentrated development team, Often times the software is being developed by individuals all over the world who may not work on the software for the duration of its developing lifetime. This lack of continuity and common direction can lead to barriers to effective Communication is surrounding the software. It is a well-known fact among security experts that obscurity leads to security vulnerabilities . In contrast, open source code is visible for anyone to	Close Source software is considered much more secured since it is developed in a governed conditions by their intensive employees having a frequent direction. Additionally, the source code may perhaps be considered as well as modified through this professionals by itself, additionally is intensively audited, getting rid of the possibility of back-door Trojans as well as lowering the threat of any other bugs or obstacles with the software.
	see, leading to more eyes looking for potential vulnerabilities to be fixed .It is important to evaluate the ability of an open source community and or its service providers to respond quickly to find vulnerabilities. It is worth noting that open source communities fixed security vulnerabilities twice as quickly as commercial software vendors did, according to a recent study by Vera code	
Software Licenses	All software applications require a software license in order to run. A software license is a legal instrument that governs how the software can be used and distributed. The permit for most open-source programming uses what some have called copy left. This is a play on the expression "copyright." To comprehend what this implies, it is worth considering how open-source programming could be abused by somebody. Consider that anyone can download, adjust and convey open- source programming. What would prevent an organization from making it form of the product and afterward begin offering it? That is the place copy left comes in. The permit for open-source programming particularly expresses that a client is not permitted to put limitations on its utilization or conveyance. Along these lines, by consenting to the permit, you can't begin offering the product	Many Close Source software applications are also commercial, meaning that you have to pay for a license. However, many other Close Source software applications are free. The fact that software is free does not mean it is not Close Source . In many cases, software companies have invested many years of software development into a product. By making the software Close Source , they are protecting their investment and make it possible to commercialize their software. The revenue from software sales can then be used to continue developing the software.
Reliability	Since OSS are accessible on a substantial number of unconfirmed sites and even most of these conveyances might be changed by any technically sound user, all the distributions are dependable as far as security, heartiness, execution. The reason is that is a user includes/changes some segment; it might meets expectations great exclusively, however, may conflict with different segments and at last degrade the software. With a extensive work on its reliability feature ,now Open source software are considered as a reliable software. For Example Linux.	Close Source Software is developed by certain organizations or perhaps professional teams. Exclusively accomplished products will likely be given at the channels. Since there will be no un-validated alteration, the result is consistently trustworthy

Availability	Open Source Software is freely available over the net. They are available with 24*7 support from online community, discussion forums as well.	These will be accessible through their regarded companies that own the rights to the bundles. Now and again, the trial version is accessible free of charge download and testing
Flexibility	As organizations continually look to do more with less, adopting open source software can deliver real business value—in addition to greater flexibility[12], lower IT costs and increased opportunities for innovation. Open Source software Increase Flexibility at a Faster Pace and Lower Cost	When your business uses Close Source software such as Microsoft Windows and Office, you are on a treadmill that requires you to keep upgrading both software and hardware .Updates must be installed for the proper working ,but not all the updates are flexible with all versions of the Software's

4. WHY TO USE OPEN SOURCE SOFTWARE

Above Comparison is evidence that for a every successful Close Source software we have equivalent or sometime better Open Source Software .Following Table having some examples of Software comparison for some categories

Table -2 Alternatives to Close Source Software

Software Category	Close Source Software	Equivalent Open Source Software
Operating System	Microsoft Windows	Linux Ubuntu
Browser	Internet Explorer	Mozilla Firefox
Office automation	Microsoft Office	Open Office
Mathworks	MATLAB	Sci Lab
Graphics Tool	Adobe Photoshop	GIMP(GNU Image Manipulation Program
Drafting tool	Auto CAD	Archimedes
Web Editors	Adobe Dreamweaver	NVU
Desktop Publishing	Adobe Acrobat	PDF Creator
Blogs	Blogger	WOEDPRESS
Mobile	IOS	Android
Media Player	Windows Media Player	VLC Player
Database	Oracle ,Microsoft SQL Server	My SQL ,Mongo DB,HADOOP
Server	Microsoft Window Server	Red Hat Server,Ubuntu Server
Web Server	IIS	Apache

As We have so many Open Source Software Alternatives Available for the Existing Close Source Software which are equivalent or more reliable ,secure & flexible which leads to gaining the popularity for Open Source Software

5. POINTS FOR THE EVOLUTION OF OPEN SOURCE SOFTWARE^[14]

Many people in the business world today would like to use Close Source software instead of open source software due to the misconception that Close Source software is better recognized than open source software.^[7] Why android operating system based smart phones much more popular, why are Joomla, Drupal, Moodle equally famous to Close Source software just like Sharepoint, Blackboard etc We now go back to the points the institution has to speculate while making a decision about procurement of a total solution. Some critical criteria are.

- i. Free of Charge Distribution Legal Rights
- ii. Gain Access To Source Code
- iii. Authorization to customize
- iv. Distribute Modifications^[8]
- v. Flexibility
- vi. Scalability
- vii. Knowledge base and Self reliance
- viii. Cost
- ix. Maintenance/Longevity
- x. Performance
- xi. Usability

6. WHY TO PREFER OPEN SOURCE SOFTWARE SERVICES^[11]

The economic climates and characteristics of open source are persuasive, and the IT industry as a whole will carry on to feature its advantages involving proprietary professional circumstances. —More than 70 Percent of IT Professionals in the U.S. Prefer Open Source to Close Source Software for Continuity, Control^[9]. An possibility in order to save upon valuable currency exchange. It's estimated that Close Source software purchases costs our country of the order of a billion dollars per annum, many of which goes toward purchase primary software package such as MS Windows and MS Office. This specific could easily be conserved simply by switching over to Linux and a FOSS office suite, such as LibreOffice. In general, FOSS oriented software development keeps the cost low for the developer and the end user and this is most recommended in regards to our IT entrepreneurs, whose primary investment is the idea and the time spent. India has a cost advantage in terms of labour arbitrage. Close Source software disturbs this cost advantage by impacting massive overheads

7. CONCLUSION

It is currently observed that lots of organizations as well as service providers such as IBM, Hewlett Packard are already doing business with all the open source software at the present and it is likely to distribute to many other organizations as well. For These Reasons, the day is not far enough when the software industry will likely experience a great transformation in its application of free software in major as contrasted to that of the Close Source software. In our viewpoint the reality that there exists an option of source codes being exposed to the users has the potential to conquer their guidelines set by the Close Source software and creates its very own specialized niche within the software ecosystem. Thus, concluding by expressing there has been a switch over through the Close Source Software model to the open source software model and it is right here to stay but one major question which arises now is –whether the Close Source software manufacturers should be able to cope up with the obstacles positioned in opposition to them by the OSS and will they have the ability to contend with the OSS?

Table 3- Summary of Open Source Software Versus Close Source Software

	Closed Source				Open Source	
	Proprietary	Shareware	Freeware	Private	Commercial Support	Community Support
Source Code Available	No or only with restrictions	No	No	No or only with restrictions	Yes	Yes
Binary Executables Available	Yes	Yes	Yes	Yes	Yes	Usually
No Restrictions on	No	No	No	No	Yes	Yes
Unlimited Number of Users	No	No	No	No or only with restrictions	Yes	Yes
Right to Study How	No	No	No	No or only with	Yes	Yes
Right to Redistribute	No	No	Yes	No	Yes	Yes
Right to Modify	No	No	No	No or only with restrictions	Yes	Yes
Right to Fork	No	No	No	No	Usually	Usually

Why to Use OSS

- Reduced Costs
- Total Cost of Ownership (TCO)
- Development costs
- Support costs
- No per user/per instance costs
- Interoperability using open standards and APIs
- Can evaluate for suitability, security, etc.
- Can modify to address user’s needs
- Avoid vendor or integrator lock-in
- Ability to do rapid prototyping/proof of concept
- Can share and redistribute with other organizations

Why Not to Use OSS

- Software does not provide required functionality
- License compliance considerations
- Insufficient external support available
- Incompatibilities with existing systems or software
- Lack of required skills in the organization
- Missing features (e.g. support for FIPS approved encryption)

What are the benefits of using FOSS?

Besides the low cost of FOSS, there are many other reasons why public/private organizations are aggressively adopting FOSS. These include:

- Security
- Reliability/Stability
- Open standards and vendor independence
- Reduced reliance on imports
- Developing local software capacity
- Piracy, IPR, and the WTO
- Localization

Security

While there is no perfectly secure operating system or platform, factors such as development method, program architecture and target market can greatly affect the security of a system and consequently make it easier or more difficult to breach. There are some indications that FOSS systems are superior to proprietary systems in this respect:

1. The Gartner Group recommends that businesses switch from Microsoft Internet Information Server (IIS) to Apache or another web server, due to IIS's poor security track record. The Gartner Group noted that by July 2001 US enterprises had spent US\$1.2 billion simply fixing Code Red (IIS-related) vulnerabilities.
2. "Hacker Insurance" issued by J.S. Wurzler Underwriting Managers costs five to 15 percent more if Windows is used instead of GNU/Linux or Unix systems. Walter Kopf, senior vice president of underwriting at J.S. Wurzler Underwriting Managers, says, "We have found out that the possibility for loss is greater using the NT system." The security aspect has already encouraged many public organizations to switch or to consider switching to FOSS solutions. The French Customs and Indirect Taxation authority migrated to Red Hat Linux 6.2 largely because of security concerns.

Three reasons are often cited for FOSS's better security record:

- Availability of source code: The availability of the source code has made it easier for developers and users to discover and fix vulnerabilities, often before a flaw can be exploited. Many of the vulnerabilities of FOSS listed in Bugtraq were errors discovered during periodic audits and fixed without any known exploits. FOSS systems normally employ proactive rather than reactive .
- Security focus, instead of user-friendliness: FOSS can be said to run a large part of the internet and is therefore more focused on robustness and functionality, rather than ease of use. Before features are added to any major FOSS application, its security considerations are considered and the feature is added only if it is determined not to compromise system security.
- Roots: FOSS systems are mostly based on the multi-user, network-ready Unix model. Because of this, they come with a strong security and permission structure.

Such models were critical when multiple users shared a single powerful server—that is, if security was weak, a single user could crash the server, steal private data from other users or deprive other users of computing resources. Consequently, vulnerabilities in most applications result in only a limited security breach.

Reliability/Stability

FOSS systems are well known for their stability and reliability. There are many anecdotal stories of FOSS servers functioning for years without requiring maintenance. However, quantitative studies are more difficult to come by. Here are two of the studies conducted to date:

- In 1999 Zdnet run reliabilities test for different type of systems (Red Hat Linux, OpenLinux and Microsoft's Windows NT Server 4.0 with Service Pack 3). All systems ran on identical hardware systems and performed printing, web serving and file serving functions. The result was that NT crashed once every six weeks but none of the FOSS systems crashed at all during the entire 10 months.
- A stress test using random testing stressed seven commercial systems and the GNU/Linux system in 1995. Random characters were fed to these systems, to simulate garbage from bad data or users. The result was that the commercial systems had an average failure rate of 23 percent while Linux as a whole failed nine percent of the time. GNU utilities (software produced by the FSF under the GNU project) failed only six percent of the time. A follow-up study years later found that the flaws identified by the study were all fixed in the FOSS system, but were generally untouched in proprietary software.

Open standards and vendor independence

Open standards give users, whether individuals or governments, flexibility and freedom to change between different software packages, platforms and vendors. Proprietary, secret standards lock users into using software only from one vendor and leave them at the mercy of the vendor at a later stage, when all their data is in the vendor's proprietary format and the costs of converting them to an open standard is prohibitive.

Another advantage of FOSS is that they almost always use open standards. This is due to two primary reasons:

- Availability of the source code: With the source code, it is always possible to reverse-engineer and document the standard used by an application. All possible variations are plainly visible in the source code, making hiding a proprietary standard in FOSS systems impossible. Proprietary software, however, are much harder to reverse-engineer and in some cases are deliberately obfuscated.
- Active standards compliance: When established standards exist, such as HyperText Markup Language (HTML), which controls how web pages are displayed, FOSS projects actively work to follow the standards faithfully. The Mozilla web browser, a FOSS effort, is fully compliant with many standards from the World Wide Web Consortium (W3C). Webstandards.org notes that Mozilla is one of the most compliant browsers available today²⁸. Compliance with standards

is due to the FOSS development culture, where sharing and working together with other applications are the norm. It is also much easier to work with a globally dispersed group of developers when there is a published standard to adhere to. Using FOSS systems as a means of gaining vendor independence has been raised in several areas. A report to the UK Government concludes that “the existence of an OSS reference implementation of a data standard has often accelerated the adoption of such standards, and recommends that the Government consider selective sponsorship of OSS reference implementations.”

Reduced reliance on imports

A major incentive for developing countries to adopt FOSS systems is the enormous cost of proprietary software licenses. Because virtually all proprietary software in developing countries is imported, their purchase consumes precious hard currency and foreign reserves. These reserves could be better spent on other development goals. The European study, “Free/Libre and Open Source Software: Survey and Study”, also notes that, “The costs of this more service-oriented model of open source are then also normally spent within the economy of the governmental organization, and not necessary to large multinational companies. This has a positive feedback regarding employment, local investment base, tax revenue, etc.”³⁰

Developing local software capacity

It has been noted that there is a positive correlation between the growth of a FOSS developer base and the innovative capacities (software) of an economy. A report from the International Institute of Infonomics lists three reasons for this:

- Low barriers to entry: FOSS, which encourages free modification and redistribution, is easy to obtain, use and learn from. Proprietary software tends to be much more restrictive, not just in the limited availability of source code, but due to licensing, patent and copyright limitations. FOSS allows developers to build on existing knowledge and pre-built components, much like basic research.
- FOSS as an excellent training system: The open and collaborative nature of FOSS allows a student to examine and experiment with software concepts at virtually no direct cost to society. Likewise, a student can tap into the global collaborative FOSS development network that includes massive archives of technical information and interactive discussion tools.
- FOSS as a source of standards: FOSS often becomes a de facto standard by virtue of its dominance in a particular sector of an industry. By being involved in setting the standards in a particular FOSS application, a region can ensure that the standard produced takes into account regional needs and cultural considerations.

The FOSS developmental approach greatly facilitates not only innovation but also its dissemination. A Microsoft internal memo noted, “Research/teaching projects on top of Linux are easily ‘disseminated’ due to the wide availability of Linux source.

In particular, this often means that new research ideas are first implemented and available on Linux before they are available / incorporated into other platforms.”

Piracy, IPR, and the WTO

Software piracy is a problem in almost every country around the world. The Business Software Alliance estimates that software piracy in 2002 alone cost US\$13.08 billion. Even in developed nations where software is affordable in theory, piracy rates were as high as 24 percent in the United States and 35 percent in Europe. Piracy rates in developing countries, where lower incomes make software far more expensive, are upwards of 90 percent. Software piracy and lax laws against it can and does hurt a country in many ways.

A country with poor protection for Intellectual Property Rights (IPR) is not as attractive to foreign investors. Membership in the World Trade Organization (WTO) and access to its benefits are strongly affected by the level of protection given to IPR in a country. Finally, a culture of software piracy hurts local software development, as there is less incentive for local software developers to create a local product.

Localization

“Localization involves taking a product and making it linguistically and culturally appropriate to the target locale (country/region and language) where it will be used and sold.” . Localization is one of the areas where FOSS shines because of its open nature. Users are able to modify FOSS to suit the unique requirements of a particular cultural region, regardless of economic size. All that is necessary is the technical capability within a small number of individuals to create a minimally localized version of any FOSS. While the construction of a completely localized software platform is no small feat, it is at least possible. Microsoft’s decision in 1998 against producing an Icelandic version of Windows 98/95 would have had serious implications if it were not for the emergence of FOSS alternatives.

Most initial FOSS initiatives in the Asia-Pacific region have dealt with localizing FOSS. More details on localization can be found in the “Localization and Internationalization” section of this primer.

What are the shortcomings of FOSS?

For all the benefits FOSS brings, it is not suitable for every situation. There are areas where FOSS needs improvement.

Lack of business applications While there are many FOSS projects out there today, there are still many areas that lack a full-featured product, especially in the business world. The recent porting of Enterprise Resource Planning platforms such as SAP and Peoplesoft have helped cover the high-end application market, but the Small and Medium Enterprise (SME) market is still poorly served. Basic, polished accounting applications such as Quickbooks, Peachtree or Great Plains do not have FOSS equivalents at this time. This problem has come about in part due to the shortage of people competent in both technical and business subjects. Technical developers who encountered problems and wrote software to “scratch an itch” started most of the existing FOSS projects today. These projects are usually fairly technical in nature, such as the creation of web servers, programming languages/environments and networking tools. It is rare for a software developer to encounter accounting problems, for example, and have the business knowledge to create a technical solution.

Interoperability with proprietary systems FOSS systems, especially on the desktop, are not completely compatible with proprietary systems. For organizations that have already invested massive amounts of capital into proprietary applications and data storage formats, attempting to integrate FOSS

solutions can prove to be prohibitively expensive. Changing proprietary standards, which is often aimed at preventing the integration of alternate solutions, exacerbates this problem. In time, as organizations shift from proprietary to open standards, this problem should be reduced. 18

Documentation and “polish” Established FOSS lacks the extensive documentation and user-friendliness found in commercial software³⁷. The primary focus of early FOSS developers was functionality. Creating a program that worked well was far more important than ease of use. Besides the dearth of high-quality documentation, there are also user interface issues with FOSS Graphical User Interfaces (GUI). Because the GUI element in most FOSS systems is not a single element but a collection of different projects glued together, the behaviour of the GUI elements differ greatly. Command-to-save data differ from one program to another, quite unlike proprietary desktop operating systems such as the Mac OS X or Microsoft Windows. Cutting and pasting between different programs can be wildly inconsistent or even impossible. While there is significant ongoing work to unify the desktop, the desktop is likely to remain inconsistent for some time to come

There are two types of open software:

- Project/Community open source is developed and managed by a distributed community of developers who cooperatively improve and support the source code without remuneration. These projects may be copyrighted by the contributors directly but larger projects are typically run by non-profit foundations. Well-known examples of community open source projects are Linux and Apache Web Server.

- Commercial Open Source Software, or COSS, is distinguished by open source software of which the full copyright, patents and trademarks are controlled by a single entity. The owner only accepts code contributions if the contributor transfers copyright of the code to this entity. They may distribute their software for free or a fee. Their business model typically includes revenue from providing technical support and consulting services. In terms of revenue from licensing, Red Hat is still the largest COSS company, but Facebook is the largest COSS code contributor.

What are governments doing with FOSS?

Various governments around the world have begun to take notice of FOSS and launch initiatives to reap the benefits it poses. Many of these initiatives are still in the early stages, but there is a significant trend towards incorporating FOSS into procurement and development policies. Besides the large numbers of reports and white papers recommending FOSS solutions, there are reportedly about 70 proposed laws mandating or encouraging FOSS around the world³⁸. A few are at the national level while most are at much lower (state or city) levels. The following are highlights of some of the more noteworthy efforts from around the world.

Europe

Besides being home to a significant number of FOSS developers, Europe is also

an area with strong government interests in FOSS. In particular, the European Union, Germany, France and the United Kingdom are leading the way in FOSS development.

European Union

The European Union has produced a working paper that stresses Open Standards and encourages Free/Open Source Software where appropriate. The paper, titled “Linking Up Europe: the Importance of Interoperability for E-government Services”, focuses on connecting the different national e- government systems together. It is also critical of past developments that “resulted in closed, vertical, un-scalable and frequently proprietary information systems”³⁹. This paper was produced as part of the eEurope initiative. The European Union is also creating FOSS competency centers and funding the development of certain health-related applications⁴⁰.

Germany

Germany has many different initiatives underway. The German Bundestag uses Linux on its 150 servers⁴¹, while the city of Munich plans to switch over 14,000 desktops to Linux, despite Microsoft’s last minute price cuts⁴². The police force is also transitioning 11,000 clients to Linux. It is interesting to note that price is not always the factor cited for the large number of migrations to Linux. Germany’s Interior Minister, Otto Schilly, said, “We are raising computer security by avoiding a monoculture, and we are lowering dependence on a single supplier”⁴³. The German Parliament decided in 2001 that FOSS products should be used wherever costs could be decreased by their usage⁴⁴. The Ministry of Finance has an Apache/ Linux-based intranet system that supports 15,000 users⁴⁵.

France

The officially sanctioned Agency for Technologies of Information and Communication in Administration (ATICA) counts as part of its mission, “to encourage administrations to use free software and open standards.”⁴⁶ The Authority for Customs and Indirect Taxation has also migrated to Linux, citing security reasons⁴⁷. The French agency for e-government has made open standards mandatory for all public administrations to guarantee full interoperability⁴⁸.

United Kingdom

The United Kingdom (UK) has only recently started formulating policy regarding FOSS procurement but the policies that have been produced to date have been favorable towards FOSS. The UK is primarily interested in avoiding the proprietary lock-in problem and has produced a policy to “only use products for interoperability that support open standards and specifications in all future IT developments”⁴⁹. One of the most active proponents of FOSS is the National Health Service, in part due to the insolvency of a proprietary software vendor that forced hospitals to migrate to Linux⁵⁰.

Finland

It is only fitting that the homeland of Linux’s creator is also active in the FOSS arena. One of the more public initiatives is the gradual migration of the city of Turku to Linux and Open Office. All desktop systems will be migrated, with the first pilot project of 200 computers in progress.

The Finnish State Administration is also reportedly considering replacing all of its desktops with Linux, affecting as many as 147,000 computers⁵¹.

Americas

United States

Although there is no official FOSS policy in the US federal government, there have been a number of attempts to pass pro-FOSS legislation at the state level. The states include California⁵², Texas⁵³ and Oregon⁵⁴. To date none of the bills have been passed but the momentum is not expected to slow down anytime soon.

Finding detailed information about FOSS usage in the US government is difficult, but a survey from MITRE Corporation shows that the US Department of Defense used a total of 115 different FOSS applications, with 251 examples of their use⁵⁵. Additionally, multiple reports recommending the use of FOSS in the US Federal government have appeared, including one by the (US) President's Information Technology Advisory Committee (PITAC) which recommended that the US "Federal government should encourage the development of open source software as an alternate path for software development for high end computing"⁵⁶.

A few smaller public institutions have shifted over to FOSS platforms. The most well known is the City of Largo, Florida. They have transitioned 900 city employees over to GNU/Linux, saving over \$1 million in both hardware and software costs⁵⁷. The City of Largo did more than just use Linux as an operating system; they changed their entire computing model to a thin- client system (something which Microsoft Windows currently cannot do) and as a result saved a huge amount in hardware costs. The city of Houston, Texas has also shifted systems over to a FOSS platform after Microsoft demanded that the city change to a \$12 million dollar, multi-year licensing plan⁵⁸.

Peru

Peru is well known within the FOSS community for being one of the first countries in the world to have introduced legislation favoring FOSS in government procurement. The ensuing publicity, Microsoft's response and Dr. Edgar David Villanueva Nuñez's (legislation sponsor) powerful reply would occupy IT news media for quite a while. Among the choice quotes from Dr. Nunez's response are:

To guarantee the free access of citizens to public information, it is indispensable (sic) that the encoding of data is not tied to a single provider. The use of standard and open formats gives a guarantee of this free access, if necessary through the creation of compatible free software.

Although the bill remains stalled (after a US\$550,000 donation by Microsoft and pressure from the US Embassy), the reasoning behind the Peruvian bill is something all governments concerned with public data should consider.

Brazil

The Brazilian government plans to migrate 80 percent of all computers in state and state-owned institutions to Linux over the next three years. Pilot programs are already underway and a slow, gradual migration is planned. A "Chamber for the Implementation of Software Libre" has been set up by the government to smooth this transition. Among the reasons cited for this move are lower costs, increased production of local software and "democratiz(ing) access to knowledge"⁶⁰.

The Asia-Pacific

The Asia-Pacific region, with its mix of developed and developing nations, is a very active region in FOSS usage and development. Three of the major nations in the area—Japan, South Korea and China—have recently announced an initiative to create a FOSS operating system adapted to their regional needs⁶¹.

China

China is set to be a major stronghold for FOSS over the next few years. FOSS usage in the country is growing rapidly, with Linux growth alone expected to be 175 percent in 2003⁶².

A primary driver of this massive growth is the Chinese government itself. One of its goals is to create both a hardware and software industry that “will not fall into the foreign intellectual property rights trap”⁶³. Rather than becoming dependent on foreign hardware and software vendors, China is trying to develop its local technology industry, and FOSS fits well into its software needs. Recently, the Chinese government announced that government departments would be barred from purchasing foreign produced software, effectively eliminating most proprietary software vendors such as Microsoft and Oracle⁶⁴.

Beyond sponsoring the creation of localized versions of GNU/Linux (RedFlag Linux, Blue Point Linux, etc.), China is also implementing FOSS solutions at the government level. The city of Beijing has had a project to convert 2,000 desktops to Red Flag Linux since 2001⁶⁵. China Post Office signed a deal with IBM to run GNU/Linux at 1,200 branch offices⁶⁶. While these projects cover only a small fraction of the Chinese government, they also serve as capacity-building projects for future transitions.

The Yangfan and Qihang projects launched in January 2002 are part of the Beijing Municipal Government’s computerization project. The goal of these projects is to produce a localized GNU/Linux with functionality, consistency and ease of use matching that of Microsoft Windows 98. Over 150 engineers have completed their initial target of a basic operating system, office suite, web browser and email client. The latest iteration of the project includes font development and experimental transition of government applications to GNU/Linux⁶⁷.

China is also one of three countries (the other two are Japan and South Korea) that are forming a joint FOSS project that would cover the entire spectrum of software, from operating systems to middle ware and desktop applications⁶⁸.

India

While the Indian federal government currently has no official position on the FOSS/proprietary software issue⁶⁹, India represents a hotbed of FOSS development. There are many department level initiatives:

The Central Excise Department has moved 1,000 desktops to Linux.

The government supercomputer arm, the C-DAC, has moved over entirely to Linux⁷⁰.

The Supreme Court has several pilot projects under way.

At the state level, there have been several FOSS initiatives. The most prominent is the Madhya Pradesh state government's plan to use Linux in its e-governance and Headstart programs, according to Chief Minister Digvijay Singh⁷¹. Red Hat has installed its version of Linux on over 6,000 desktops in schools⁷², with more likely to come. The state of Kerala has also several initiatives underway, including e-government and educational initiatives.

Other state level initiatives have been announced, but little has been heard about these initiatives since Microsoft's much-publicized investment in 2002⁷³

Taiwan

In 2003, Taiwan launched its "National Open Source Plan", a two-year plan to build a software industry that could replace all of the proprietary software on government and educational systems. The primary drivers for this plan are the existing dependence on a monopoly supplier and the expected cost savings. The National Computer Center is drafting the basic plan while the national education system will be switched to FOSS "to provide a diverse IT education environment and ensure the people's rights to the freedom of information."⁷⁴ Expected savings from the plan are about NT\$2 billion for the government and NT\$10 billion for the society as a whole.

Thailand

An article in the Bangkok Post on 23rd June 2003 reported that the Thailand ICT Ministry had set a target of five percent Linux installations on government systems by the end of 2003. A 10 million baht budget has been allocated. The ultimate goal is to have 50 percent of all government systems on Linux. No time frame has been set for the more ambitious target but pilot projects are already underway.

Thailand's low-cost PC program is also credited with forcing Microsoft to offer both the Windows XP operating system and Microsoft Office for a mere US\$40, the lowest price available in the world at present (3rd quarter 2003)⁷⁵.

Malaysia

The government has expressed support for FOSS solutions since November 2001. In April 2002, the Association of Computer and Multimedia Industry of Malaysia (PIKOM) produced a paper suggesting that Malaysia “officially embrace OSS” in April 2002⁷⁶. Initial deployment will start on servers and then gradually shift to desktops to minimize disruptions in operations.

Malaysia also launched in July 2002 Komnas, a low-cost computer based on FOSS⁷⁷. Komnas carries a localized version of Linux, including office suite, web browser and various utilities.

Japan

Japan is considering moving its e-government projects over to FOSS platforms due to security problems in Microsoft Windows software⁷⁸. Authorities are reportedly putting together a panel of experts to study FOSS deployment. In the meantime, the Japanese government is moving its entire payroll system over to a GNU/Linux platform. The switch is expected to cut operating costs by half, especially maintenance costs from hardware⁷⁹.

The South African government has a policy preferring FOSS systems unless there are compelling reasons otherwise⁸⁰. Among the reasons cited for this preference is that with the traditional proprietary software model, South Africa ends up primarily being an importer of software, with little influence over how software is developed. It is hoped that using FOSS systems will change this.

Tanzania is also implementing FOSS systems in its government for cost reasons, while Uganda, Ghana and Zambia are also reportedly moving towards FOSS⁸¹.

What are some successful FOSS projects?

While FOSS may seem a relatively new concept, it has actually been around since long before the Internet came into existence. FOSS has more than proven that it is ready for prime time, mission-critical usage. In some cases, it is the critical linchpin that makes the Internet possible. The following is a small sample of successful FOSS projects.

BIND (DNS Server)

Internet addresses such as yahoo.com or microsoft.com would not function if not for Domain Name Servers (DNS). These servers take these human-friendly names and convert them into the computer-friendly numeric addresses and vice-versa. Without these servers, users would have to memorize numbers such as 202.187.94.12 in order to use a website.

The Berkeley Internet Name Domain (BIND) server runs 95 percent of all DNS servers⁸², including most of the DNS root servers. These servers hold the master record of all domain names on the Internet. BIND is a FOSS program licensed under a BSD-style license by the Internet Software Consortium.

Apache (Web Server)

Responsible for receiving and fulfilling requests from web browsers, the Apache web server is one of the foundations of the World Wide Web (WWW) as we know it today. Apache has been the number one web server since April 1996 and currently commands 62.53 percent of the total web server market⁸³ “. That is more than double the market share (27.17 percent) of its closest competitor, Microsoft’s IIS server.

These figures fluctuate monthly of course. The latest figures can be found at Netcraft’s Web Server Survey site, at: http://news.netcraft.com/archives/web_server_survey.html.

Sendmail (Email Server)

The Internet as we know it would not exist without email and once again, FOSS is one of the primary drivers. An email server’s (sometimes called a Mail Transport Agent or MTA) function is to deliver user email to its destination. Complex functionality, such as email forwarding and redirection, junk email rejection and routing, makes email servers rather complex systems. The problem of junk email (sometimes referred to as spam) makes security a critical feature, as spammers sending their unsolicited email to unsuspecting users would otherwise hijack an email server and render it useless to legitimate users.

A 2001 survey by D.J. Bernstein found that Unix Sendmail had the largest market share, at 42 percent of all email servers. This was larger than the share of its next two competitors combined, Microsoft Exchange and Unix qmail, with 18 percent and 17 percent, respectively⁸⁴. Note that qmail is a Unix-based email server but is not FOSS as its licensing terms are too restrictive.

OpenSSH (Secure Network Administration Tool)

Because Internet traffic can pass through multiple networks when a user connects into a remote server, security is a major concern. The Secure Shell (SSH) protocol allows system administrators to control their servers from a distance, safe in the knowledge that it is almost impossible to intercept and decipher the information that they may be transmitting.

OpenSSH, a FOSS implementation of the SSH protocol, has grown from a mere five percent of the market in 2000 to 66.8 percent of the market in April 2002. OpenSSH came into existence as a result of a restrictive licensing change in the standard SSH implementation at that time.

Open Office (Office Productivity Suite)

While FOSS products have been strong on the server side, FOSS desktop applications are relatively new. Open Office, which is based on the source code of the formerly proprietary StarOffice, is a FOSS equivalent of Microsoft Office, with most of its features. It includes a full-featured word processor, spreadsheet and presentation software. One of the advantages for many considering the shift from a Windows desktop environment to Open Office is that it reads most Microsoft Office documents without problems. This makes the transition relatively painless and Open Office has been used in recent high profile switches from Windows to Linux. While it does not have a very large market share as yet, its usage is expected to grow dramatically over time as more organizations use this full-featured, low- cost application.

Linux

What is Linux?

Linux is the most frequently heard FOSS buzzword in the mass media today. However, because of its common usage, the term Linux has been used to refer to broader and broader definitions. It is important to understand the different definitions of Linux to be able to follow the discussions on FOSS.

Linux as the kernel

Linux was originally the name of the kernel created by Linus Torvalds. A kernel is the critical center point of an operating system that controls CPU usage, memory management and hardware devices. It also mediates communication between the different programs running within the operating system. There are other FOSS kernels, including the Mach kernel that is the core of some of the BSD distributions.

Kernels are to a certain extent interchangeable. Most FOSS applications will run on a Mach kernel, Linux kernel or even the experimental GNU Hurd kernel, without too much difficulty. However, the kernel type greatly influences performance and the hardware platforms that the FOSS system can run on. For instance, the less mature GNU Hurd kernel can run only on the x86 (PC) architecture. The Linux kernel has been ported to run on almost any computing architecture, including the Playstation 285, mainframes and embedded devices.

Linux as a distribution

The more common usage of Linux today refers to a Linux distribution, which includes far more than the kernel. The Linux distribution (sometimes referred to as the GNU/Linux distribution in recognition of the GNU Project's significant contribution) contains the Linux kernel at its heart and all the FOSS components required to produce full operating system functionality. This includes the system libraries, GUI, various databases, web servers, email utilities, and others. These same components are also often found on other FOSS and even on proprietary operating systems. For instance, XFree86 is the default GUI foundation in Linux and BSD. XFree86 is also used on proprietary Unix systems such as Solaris, HP-UX and IBM's AIX system.

Reports that say "Munich May Opt for Linux After All"⁸⁶ refer to the Linux distribution, including word processing, printing and email software. Even though the Linux kernel forms less than 0.25 percent (binary file size) of a Linux distribution, its functionality is critical enough to allow the entire distribution to be called Linux.

There is no single Linux distribution. While all distributions contain the Linux kernel at its heart, the FOSS applications included and the configurations supported vary. There are multiple commercial distributions, several freely available, and numerous customized distributions that are targeted to the unique needs of different users. While the FOSS contents of different Linux distributions are mostly identical,

they are optimized for different uses such as for high-end servers, user-friendly desktops or even embedded systems. Localized distributions at a minimum include the fonts, input methods and menu translations necessary to use the software in the regional language.

Is Linux FOSS?

The Linux kernel is FOSS, licensed under the GNU General Public License. However, different distributions of Linux contain different components, some of which are not FOSS. For example, the German SuSE Linux distribution includes the non-FOSS YaST (Yet another Setup Tool) installation program.

The Debian GNU/Linux87 distribution is one of the few distributions that are committed to including only FOSS components (as defined by the Open Source Initiative) in its core distribution.

Where can one obtain Linux?

FOSS, in binary and source code format, is free and downloadable from the Internet. The Linux kernel itself can be downloaded from <http://www.kernel.org> and other applications from their respective websites. However, most users tend to obtain distributions of Linux. The following is a table of some of the most popular distributors of Linux:

Popular Linux Distributors

Debian	www.debian.org
Redhat	www.redhat.com
SuSe	www.suse.com
Mandrake	www.mandrakelinux.com
SlackWare	www.slackware.com
TurboLinux	www.turbolinux.com

The advantages of going with distributions of Linux are many. The single most important advantage of vendor Linux over “stock” Linux is that it saves users time:

- **Download time:** The Linux operating system and complementary software involve large files and long download times. A standard 56kbps modem would take at least 45 days to download a standard 3 CD distribution. Vendors also provide bundled software—browsers, server applications, office suite, etc.—that saves the users from the tedious work of hunting and downloading individual software packages.

Installation and compiling time: Many FOSS packages are downloadable only as source code. Users are required to compile and install the software themselves, assuming they are competent enough to do so. On a slow computer, compilation of source code may take days or even weeks. Vendor distributions of Linux often come precompiled and packaged with an easy installation system that takes less than an hour to install on most modern systems.

- **Quality assurance:** Vendors typically perform extensive testing to ensure that all of the components work well together. Since FOSS projects are developed independently, there is always the chance that changes in one package have outdated another package. Vendors resolve these dependencies for the user, supplying an integrated package that works “out of the box”.

Learning time: Vendors provide manuals and publish reference material (for sale) for their products, making Linux much easier for the average user to learn.