bridges.org
spanning the international digital divide

Comparison Study of Free/Open Source and Proprietary Software in an African Context

Implementation and Policy-Making to Optimise Public-Access to ICT

bridges.org, in collaboration with SchoolNet Africa, the International Development Research Center and the Open Society Institute

24 May 2005

For more information, contact:
info@bridges.org

http://www.bridges.org

Collaborating partners:

IDRC CRDI

SchoolNetAfrica

OPEN SOCIETY INSTITUTE

ACKNOWLEDGEMENTS

This report was made possible by the generous support of the International Development Research Center (IDRC) and the Open Society Institute (OSI). The research was conducted in collaboration with SchoolNet Africa and AITEC Uganda.

We acknowledge the contributions of the researchers who participated in this study: Dr. Hillar Addo, Vincent W Bagiire, Jennifer Hüsler, James Wire Lunghabo, Philipp Schmidt, Liam Smit, Paul Ssengooba.

We would like to thank the 121 computer labs and 40 policy-makers that provided the information and data that this report is based upon; their willingness to engage and share experiences enriched this research immeasurably. Their names are not listed here, because all contributions were made anonymously. We are also grateful to the many organisations and individuals that introduced us to the computer labs and helped with logistical support.

Finally, we would like to recognise the contributions of the advisory group members who gave us their time and attention throughout the study, including by reviewing and commenting on milestone documents and drafts. Their insight and advice helped ensure that the work remained balanced and objective. The group was comprised of experts in the field with diverse views on the topic:

- **Joseph Alhadeff**, Vice President Global Public Policy/ Chief Privacy Officer, Oracle
- **Chose Choeu**, Director Legal and Corporate Affairs, Microsoft South Africa
- **Johan Eksteen**, ICT Coordination Manager, Council for Scientific and Industrial Research
- **Laurent Evrard**, Director Computer Services, Polytechnic Namibia
- **Richard Fuchs**, Director ICT4D, International Development Research Center
- **Riff Fullan**, Program Coordinator, bellanet
- **Johan Helsingius**, Board Member and Technical Advisor, bridges.org
- **Shafika Isaacs**, Executive Director, SchoolNet Africa
- **Joris Komen**, Executive Director, SchoolNet Namibia
- **Allen Luyima**, Technical Coordinator, SchoolNet Uganda
- **Simbo Ntiro**, Founder eThinkTank Tanzania
- **Jonathan Peizer**, Chief Technology Officer, Open Society Institute
- **Stormy Peters**, Manager HP Open Source Program, Hewlett Packard
- **Theo Schoemans**, Managing Director, Schoemans IT
- **Guido Sohne**, sohne.net
- **Steve Song**, Managing Director, Connectivity Africa
- **Fred Tipson**, Director International Affairs, Microsoft
- **Bill Vass**, CIO, SUN Microsystems
- **Arno Webb**, CIO Department of Arts and Culture, Government Information Technology Officers Council, South Africa

TABLE OF CONTENTS

ACKNOWLEDGEMENTS 2
EXECUTIVE SUMMARY 5
1 INTRODUCTION 13
1.1 Public ICT access and software choices in Africa 13
1.2 Issues around public access to ICT 14
1.3 Choosing software for public-access computer labs 14
1.4 Software choices in the African context 17
1.5 The need for this study 18
1.6 Roadmap to the report 19
2 OVERVIEW OF METHODOLOGY AND APPROACH 20
2.1 Framing the study with a Real Access/Real Impact approach 20
2.2 Data and information collection 21
2.3 The computer labs studied 22
3 OVERVIEW OF THE COUNTRIES STUDIED 24
3.1 Namibia 25
3.2 South Africa 27
3.3 Uganda 29
4 FINDINGS ON KEY ISSUES INFLUENCING SOFTWARE CHOICES 31
4.1 Appropriateness of software to local needs and conditions 31
4.2 Software cost and affordability 44
4.3 Locally relevant and useful applications, content and services 53
4.4 Capacity-building for end users 62
4.5 Technical capacity of computer lab staff 64
4.6 Availability and quality of technical support 66
4.7 The impact of awareness on software choices 76
4.8 The effects of policy and political will on software choices 78
4.9 Self-sustainability: a critical factor for computer labs 81
5 OBSERVATIONS AND RECOMMENDATIONS 86
5.1 Key observations of the study 86
5.2 Key challenges and lessons learned in public-access computer labs in Africa 88
5.3 Recommendations to public-access computer labs in Africa 92
5.4 Recommendations to decision-makers setting policies that affect public-access computer labs in Africa 95
6 CONCLUDING REMARKS 102

ANNEXES (attached separately)

Annex 1. Terminology debate: free, open source, proprietary, commercial software
Annex 2. Research methodology
Annex 3. Thin client configuration
Annex 4. Definitions and glossary
Annex 5. Background and further reading
Annex 6. Additional aspects of software use in public-access computer labs
Annex 7. Field-study data
Annex 8. List of key local stakeholders and initiatives

Index of Tables

Table 1: Field study -- Overview of study sample (country breakdown) 22
Table 2: Field study -- Lab types studied (schools, Internet café, ...) 23
Table 3: Field study -- Desktop operating systems used in the computer labs 35
Table 4: Field study -- Working and non-working computers 42
Table 5: Field study -- Combined analysis, reliability of FOSS thin-client systems 42
Table 6: Field study -- Frequency of software "crashes" 42
Table 7: Field study -- Perception about software reliability 43
Table 8: Field study -- Ranking of cost factors for set-up costs 45
Table 9: Field study -- Computer lab network configurations 53
Table 10: Field study -- Internet access in computer labs 58
Table 11: Field study -- Staff experience with different software applications 65
Table 12: Field study -- Support types used in the labs 68
Table 13: Field study -- Ability of internal staff to resolve technical problems 69
Table 14: Field study -- Computer experience of technical support staff 70
Table 15: Field study -- Types of support used in the lab 71
Table 16: Field study -- Access to outside technical support 72
Table 17: Field study -- Quality of outside support 72
Table 18: Field study -- Initial set-up costs 83
Table 19: Field study -- Ongoing costs 83

EXECUTIVE SUMMARY

Public ICT access and software choices in Africa

Information and communications technology (ICT) can reward those who use it well with increased economic opportunities and income, better quality of life, and cultural and political advantages. Those who do not use it are left behind, and ICT disparities exacerbate existing inequities. Many governments, development agencies, and community organisations have responded to this problem with public-access projects aimed at bringing technology to disadvantaged countries and communities. Frequently these projects set up computer labs with Internet connectivity in public spaces like schools and community centres, targeting people who may never own a computer or use one in their workplace.

Given the practical realities faced by these public-access projects, the type of technology used can be a make-or-break factor in their success. It is crucial that the decision-makers behind these efforts carefully weigh the pros and cons of different technology solutions, make informed decisions about the design of public-access computer labs, and make smart choices about how to balance spending limited funds on things like computer hardware, software, and Internet connectivity. In this context, the choice of software implemented in public computer labs is a core issue, one which has been the subject of considerable debate in Africa recently.

Choosing software for public-access computer labs

The decision about whether free/open source software or proprietary software is implemented in a computer lab is one of many important decisions faced by public-access projects. What kind of software is best for public-access computer labs in Africa? This seemingly straightforward issue is a point of contention for many.

- **Free/open source software (FOSS)**. Some argue that free/open source software is the best choice for public-access projects because of its low cost and the associated benefits it brings to society and the economy. This kind of software is distributed together with its underlying source code, under a certain kind of copyright. FOSS copyright licenses allow everyone to read, modify, and redistribute the source code, so programmers can improve and adapt the software, and fix bugs. And the software can be shared with others, so users can give it to their neighbours, colleagues and friends. Some FOSS licenses prevent software developers from distributing their modifications and additions under a non-FOSS license. However, these restrictions have no impact on end users, who are the focus of this study.

- **Proprietary software (PS)**. Others maintain that proprietary software is a better option, arguing that it offers comparable benefits and that there is more to the cost issue than meets the eye. Proprietary software is privately owned and controlled, usually by a company. The owners of proprietary software hold a copyright that awards them the exclusive rights to publish, copy, modify, and distribute the software and they usually keep the source code hidden. Most proprietary software companies sell an "end-user license" that defines how people are allowed to use the software on their computers -- for example, only allowing non-commercial uses, or restricting the user's ability to share the software.

The choice of software raises a number of deep (and often divisive) issues. A polarised debate draws on complex cost-benefit analyses and philosophical underpinnings. Well-intentioned public-access projects -- some ignorant of what is at stake, and others overwhelmed by the magnitude of the discussion -- get caught in the middle. And decision-makers are urged to choose.

The software comparison study

The Comparison Study of Free/Open Source and Proprietary Software in an African Context was undertaken in 2003 by bridges.org in collaboration with SchoolNet Africa, the International Development Research Center (IDRC), and Open Society Institute (OSI). The study looked at the software environments in public-access computer labs in Namibia, South Africa and Uganda, and examined the factors that affect software choices, the realities of the current situation in Africa, and the long-term implications of software choices for Africa, considering both ground- and policy-level issues. It is the first study of its kind in Africa.

The software comparison study had two main objectives:

1. To examine ground-level realities and direct implications of the choice between free/open source and proprietary software in the African context, and
2. To inform decision-making so as to optimise public-access to ICT in Africa.

The research methodology for the study was developed from a combination of literature review, impressions collected during an initial scoping study, and application of bridges.org's *Real Access/Real Impact* framework. The report's findings are based on quantitative and qualitative data collected in 2003-2004. Researchers visited 121 computer labs, collected information from lab staff through a comprehensive questionnaire, and interviewed lab managers and policy-level decision-makers. The collected data was analysed by an outside statistical consultant and combined with background research where appropriate.

This report presents the findings of the study, providing objective information intended to inform decision-makers in Africa and beyond. This report is targeted to both policy-level decision-makers and managers of computer labs, but it may also be relevant to the interested layperson.

Overview of the countries studied

While the study was designed to produce results that would be as broadly applicable as possible to a wide array of countries and settings, practical limitations required a focus on a few representative countries: Namibia, South Africa, and Uganda were selected. These three countries offer a diverse environment in terms of awareness levels, the interest and involvement of the private sector and civil society, and the positions governments are taking.

Namibia has more FOSS computer labs in schools than any other African country, yet the Government is involved in a strategic public-private partnership with a multinational proprietary software vendor. The South Africa Government has developed one of the few official government policies regarding software choices and many Government officials have expressed their support for FOSS. At the same time, the President has accepted one of the most comprehensive proprietary software donations to schools in the world. In Uganda, the Government has not developed a clear position on the issue of software choices, but is conducting a teacher training programme that is based on proprietary software applications; there are some strong supporters of FOSS in academia and among smaller ICT businesses, but the industry as a whole is dominated by proprietary software companies.

The majority of the computer labs visited in this study have only the minimum infrastructure and equipment necessary to make the operation of computers possible. The size of labs varies widely depending on their economic situation and the source of donations; some of the more affluent labs have 30 or 40 new computers, while poor labs may only have two or three PCs, which are often second-hand or refurbished. Most computer labs have at least a minimum level of security and furniture, because many donors require these kinds of basic provisions to be met before they will donate computers to a school. It is rare to find air conditioning or even fans being used to cool the labs, and sometimes a heavy price is paid for not sealing the room against heat and dust. In addition, poor electric and telecommunications infrastructure is a problem, especially outside of the main towns where frequent electricity failures and unreliable telephone lines are common.

Findings of the study

The findings of the study look at the key issues that affect software choices in the public-access lab environment, and illustrate the context of these choices in African labs. The following criteria that affect software choices and ultimately determine whether people have *Real Access* to ICT in Africa's public-access computer labs were considered:

- Appropriateness of software to local needs and conditions;
- Software costs and affordability;
- Locally relevant and useful applications, content and services;
- Capacity-building for end users;
- Technical capacity of computer lab staff;
- Availability and quality of technical support;
- The impact of awareness on software choices;
- The effects of policy and political will on software choices; and
- Self-sustainability: a critical factor for computer labs.

Key observations from the study

- Both free/open source software and proprietary software can be used to offer technology solutions appropriate for African public-access computer labs.
- The thin-client model provides a reliable, cost-effective and popular solution for public-access computer labs in Africa.
- Software license costs for proprietary software are significant in principle, but in practice they are not borne by many of the public-access computer labs in Africa.
- At ground level in Africa, the potential for cost savings gained from the use of FOSS depends on many factors.
- General ICT skills levels -- especially for installation and maintenance of software -- are low overall, and experience with proprietary software is more pervasive.
- Training courses for PS are more widely available than for FOSS.
- The fact that FOSS makes source code available and encourages modifications is not exploited by the vast majority of public-access lab staff or users in Africa because they lack the necessary skills. However, it does offers an opportunity for local service providers to create customised applications.
- The availability and quality of local technical support in Africa (for both free/open source and proprietary software) is reasonably high overall, although FOSS support tends to rely on free services.
- While there are a number of projects underway to translate software into African languages, these localised versions are not yet widely used in public-access labs and there is some disagreement about the value of local language software.
- Most FOSS labs in Africa are set up and supported by a small group of enthusiastic implementing organisations, so the success of FOSS labs relies heavily on their efforts.
- General trends with regard to economic situation, age of the lab, and staff experience can be identified among labs that use the same type of software.

Key challenges and lessons learned in public-access computer labs in Africa

- Financial constraints are the key obstacle for African public-access labs, regardless of whether they are using free/open source or proprietary software.
- With only rare exceptions, the current models for public-access computer labs in Africa are not self-sustainable, regardless of whether they are using free/open source or proprietary software. Linking sustainability with the effective delivery of social services could make public-access labs worth subsidising over the long term.
- Overall, local lab staff members lack the skills and expertise to help users and communities realise the potential benefits of computers for their lives.

- Most African public-access computer labs lack a full understanding of their users' needs. This also reduces the labs' opportunities to charge for services.
- While the limited number of locally-relevant applications is an issue in some African public-access labs, the overall lack of awareness among lab staff about software alternatives is a more significant factor limiting the availability of locally-relevant applications.
- Confidence in and enthusiasm for certain kinds of software are important success factors for public-access labs in the African context, but many are hindered by the effects of low levels of technical expertise among staff and users.
- In addition to cost and lack of awareness, the absence of effective local channels for obtaining software make procurement difficult for many public-access labs in Africa.
- The availability of effective technical support (to ensure that the computers remain operational) has a significant impact on the success of public-access labs, but support for proprietary software (usually commercial technical support) is more readily available in Africa than support for free/open source software (usually non-commercial technical support).
- Local staff and users are rarely involved in the software choices made in African public-access labs.

Recommendations to public-access computer labs in Africa

Computer labs in Africa are faced with very significant obstacles to providing public access to ICT in the communities they serve. The software choice is one important decision, which is intrinsically linked to cost, technical support and many other factors that determine a lab's success in the short and long term. Pragmatic choices are necessary to overcome these obstacles and ensure that people are able to use ICT effectively to improve their lives, especially those users whose only opportunity to access ICT is through public labs.

The following recommendations are directed to the staff and managers of public-access computer labs in Africa, with the intention of focusing their attention on the most critical areas related to the effective choice and implementation of software in their labs.

- Analyse user needs by listening to your stakeholders and educating them about the possibilities that computers offer.
- To the greatest extent possible, provide software applications that are the most relevant in the local context.
- Ensure that the lab has access to appropriate, affordable, and effective technical support to keep the computers operational.
- Do not let persuasive sales pitches or strong advocacy make the choice of software seem like it has to be an either-or scenario: consider a mix of FOSS and PS if that best suits lab needs.
- Keep it simple by using standard software applications, and make the most of staff experience.
- Be smart about proprietary software donations, and consider both short- and long-term costs and benefits.

- Analyse all costs associated with owning and using software over the short and long term, and within the context of the particular lab and the local economic environment.
- When making software choices, choose reliability and stability over functionality.
- Balance the cost and effort required to upgrade software with the expected benefits to the user. If the lab chooses free/open source software, the incentive to use the latest versions of applications might be higher, due to frequent improvements in usability, functionality and reliability. Regardless of the type of software chosen, always apply the latest patches and updates.
- If the lab chooses the Microsoft Windows operating system, adequately protect computers against viruses and other malicious code to minimise support costs.[1]
- Allow only minimal access to the system configuration to avoid users making changes that could cause computers to stop working.

Recommendations to decision-makers setting policies that affect African public-access computer labs

There are various decision-makers who lead or influence policies affecting public-access computer labs in Africa, including government officials, private sector actors, influential non-governmental organisations (NGOs), bureaucrats at international institutions, and development aid organisations and donors. They carry the responsibility for long-term vision and planning to ensure effective and sustainable public access to ICT that brings concrete benefits to individuals, as well as society and the economy. Those who make determinations about the software that is rolled out in public-access computer labs are often in the position to drive change at a large scale; however, they are frequently also faced with complicated constraints that affect their choices. Usually their decisions impact more than just a few users in one local computer lab, but often tens or hundreds of computer labs and thousands of users. And their support for certain types of software can also have an impact on the broader social, economic, and political environment.

The following recommendations are directed to those decision-makers whoare setting policies that affect public-access computer labs in Africa. These can be used as guidelines during the decision-making process and provide background to the key issues.

- Address the sustainability problem for public-access computer labs by better equipping them to deliver a social return on investment (improved education, healthcare, economic opportunities, etc.) that is worth subsidising over the long term.
- Increase local capacity, and actively involve lab staff and managers in decision-making processes about software and other aspects of public-access labs.

1 While this report does not focus on specific proprietary software products, the general term "proprietary software" is too broad when used in combination with the issue of protection against computer viruses. This particular point is relevant for the Microsoft Windows operating system. See the section "Recommendations to public-access computer labs in Africa" for more information.

- When making software decisions that involve significant change (such as upgrading operating systems or migrating to FOSS applications), implement effective change management that addresses issues of training, process redesign and reluctance to change in public-access labs.
- If you choose to promote FOSS solutions across society and the economy, focus on raising awareness about software use and benefits at all levels: decision-makers, community stakeholders, lab staff, and end users.
- Focus on generic computer literacy skills to increase e-literacy broadly.
- Conduct high-level surveys to determine user needs and encourage public-access labs to add local information collected from the communities they serve.
- Encourage and support the development of locally-relevant applications that run on multiple operating systems.
- Use public funds to create software that can fill specific local needs and be considered a public good.
- Assess the need and availability of local language software and consider all available strategies for promoting its development.
- Make use of web-based content (that can be accessed from a variety of different software platforms), if possible, to reduce the impact of the software choice.
- Make smart choices about accepting software (and related) donations and keep public-private partnerships transparent.
- Take all steps possible to ensure that sufficient technical support is available for large projects to be rolled out.
- Analyse the Total Cost of Ownership in the specific local environment, rather than relying on general statements of the cost advantages of one solution.
- Consider the pros and cons of diversity in the computing environment, to make balanced choices between increased costs on the one hand and increased value and opportunity on the other.
- Increase Internet access to address the lack of applications and enable the use of online technical support.
- If you choose to promote FOSS solutions across society and the economy, ensure that alternative software procurement and delivery models are available where cheap and fast Internet connectivity is not an option.
- Fully understand the implications of software choices so you can leverage bargaining power to bring donations and other benefits to public-access labs.
- Partner with the local computer labs to evaluate existing policies with regard to software choices and ICT use for public access.
- Support open standards (such as open document formats) to minimise the impact of a rapidly changing technology environment.

Concluding remarks

It is difficult to resist the appeal of concepts such as information sharing, collaboration, and freedom of knowledge, which are foundations of the FOSS movement. But in Africa, it is important to remember that FOSS is just one software option in the larger ICT toolbox. And ICT is merely a means to an end that is most valuable when it supports broad social and economic goals, such as facilitating healthcare delivery, making small businesses more competitive, or improving education and government services. In this context, the discussion of software choices necessarily moves from philosophical underpinnings to pragmatic concerns.

Free/open source software is now enjoying widespread interest among government officials in Africa, and large international companies are contributing to its development and pushing for its adoption across the continent. Some proprietary software companies are also working to address the key problems in public-access computer labs. But meanwhile, specific software applications (whether FOSS or proprietary) that could make computers more useful to local communities are still missing.

A number of key obstacles characterise the software choice for public-access labs in Africa. But perhaps more importantly, many ICT projects struggle with fundamental difficulties that go beyond the choice of software. In particular, labs need sustainable business models, regardless of whether they are using FOSS or PS. In the absence of profits, sustainability plans that are linked with the effective delivery of social services could make public-access labs worth subsidising over the long term.

Should PS vendors pay closer attention to the practical problems facing public-access labs, and build on the commitment to deliver on social and development goals, their value proposition for Africa remains high. However, the momentum in Africa is currently in favour of FOSS, whose supporters are riding on a growing wave of enthusiasm based on successes in other developing countries. FOSS supporters in Africa have an opportunity to capitalise on this enthusiasm, but need to overcome serious hurdles to translate the hype surrounding FOSS into tangible benefits for larger parts of society.

1 INTRODUCTION

1.1 Public ICT access and software choices in Africa

A desire to promote equal access to information and communications technology (ICT) has led to a variety of projects that bring computers to disadvantaged groups in Africa through computer labs in public places. Given the practical realities facing these public-access projects, the type of technology used can be a make-or-break factor in their success. This makes it crucial that the decision-makers behind these efforts carefully weigh the pros and cons of different technology solutions. In this context, the choice of software implemented in public computer labs is a core issue, one which has been the subject of considerable debate in Africa recently.[2]

What kind of software is best for public-access computer labs in Africa? This seemingly straightforward issue is a point of contention for many. Some argue that free/open source software (FOSS) is the best choice for public-access projects because of its low cost and the associated benefits it brings to society and the economy. Others maintain that proprietary software (PS) is a better option, arguing that it offers comparable benefits and that there is more to the cost issue than meets the eye.[3] And because both sides in the debate take such strong positions, the fact that this is not an either-or scenario is often overlooked: a mix of PS and FOSS can be used on the same computer, or different computers in a single lab.

The debate has heated up recently in Africa, as FOSS begins to be considered a viable alternative to PS on a larger scale, and governments are being encouraged to define their positions. At the same time, approaches to the broader issue of public access to ICT in Africa are being re-examined, in part because so many public computer labs installed in the past have failed to become sustainable.

The Comparison Study of Free/Open Source and Proprietary Software in an African Context was undertaken in 2003 by bridges.org in collaboration with SchoolNet Africa, the International Development Research Center (IDRC), and Open Society Institute (OSI). The study looked at the software environments in public-access computer labs in Namibia, South Africa and Uganda, and examined the factors that affect software choices, the realities of the current situation at ground level in Africa, and the long-term implications of software choices for Africa, considering both ground- and policy-level issues. It is the first study of its kind in Africa. This report presents the findings of the study, providing objective information intended to inform decision-makers in Africa and beyond.

---

2 For the purpose of this report, the term "computer lab" is used to describe a site (such as a school, library, or community centre) where a number of computers are housed together and managed by an individual or organisation that is responsible for them.

3 More detailed descriptions of "free", "open source", and "proprietary" software and how the terms are used in this report can be found in the section "Choosing software for public-access computer labs" and in Annex 1.

1.2 Issues around public access to ICT

ICT can reward those who use it well with increased economic opportunities and income, better quality of life, and cultural and political advantages. Those who do not use it are left behind, and ICT disparities exacerbate existing inequities. The overall trend is that privileged countries and groups acquire and use ICT more effectively, and because the technology benefits them in an exponential way, they become even more privileged. The full range of ICT is part of the scenario -- from telephones to television, from voice-over-Internet-Protocol to personal digital assistants -- but computers and connections form the foundation. This so-called "digital divide" is a complex problem that manifests itself in different ways across countries and communities. These issues are especially critical in Africa, where the benefits of ICT are limited because so few people have access to computers and Internet connections.

Many governments, development agencies, and community organisations have responded to this problem with public-access projects aimed at bringing technology to disadvantaged countries and communities. Frequently these projects set up computer labs with Internet connectivity in public spaces, targeting people who may never own a computer or use one in their workplace. Schools are becoming the main focus of public-access projects, but public computer labs are also located in community centres, government offices, and other places that the general public visits.

Government agencies, donors, and community organisations that support public-access initiatives -- as well as the individuals that actually run the projects -- need to make informed decisions about the design of public-access computer labs. They must have clear objectives for ICT-access projects and understand what it will take to achieve their social (or commercial) goals. And they must make smart choices about spending limited funds on things like computer hardware, software, and Internet connectivity, to make the most of the lab in the short and long term.

1.3 Choosing software for public-access computer labs

The decision about whether free/open source software or proprietary software is implemented in a computer lab is one of many important decisions faced by public-access projects. On the surface, this seems like it would be a simple matter, but the choice of software raises a number of deep (and often divisive) issues. A polarised debate draws on complex cost-benefit analyses and philosophical underpinnings. Proponents on both sides argue that their preferred kind of software is better for individual and organisational users, software developers, domestic software markets, society, and the economy. Well-intentioned public-access projects -- some ignorant of what is at stake, and others overwhelmed by the magnitude of the discussion -- get caught in the middle. And decision-makers are urged to choose.

Free/open source software (FOSS)

During the last few years free/open source software has been recommended by many for use in public-access projects. This kind of software is distributed together with its underlying source code under a certain kind of copyright. FOSS copyright licenses allow everyone to read, modify, and redistribute the source code, so programmers can improve and adapt the software, and fix bugs. And the software can be shared with others, so users can give it to their neighbours, colleagues and friends.

FOSS is developed and promulgated by a loosely-connected, global community of programmers, software development and ICT services firms and users, who report bugs and make suggestions for improvements. Software developers launch and contribute to FOSS projects for a variety of reasons and (unlike PS developers) many do so without direct financial compensation or commercial interests, although many companies contribute to FOSS projects related to their business models. The FOSS community recognises the importance of creators' rights while at the same time appreciating the value to be gained from sharing knowledge. FOSS developers have established an environment where programmers share information in order to build on one another's work, but creators get certain rights because of their original contribution. The authors of a free/open source software application hold the copyright for the work, but distribute the software under a license that grants a number of substantial rights to the users, including the rights to use the software freely, access the source code, modify the source code, and share the original or modified version with others.

Some FOSS licenses allow modified "open source" software to be turned into proprietary software, while others prevent software developers from distributing their modifications and additions under a non-FOSS license.[4] More specifically, "free software" licenses require all software that is based on free software to remain free software (and be distributed under a free software license), making it impossible to turn modified free software (or a combination of free software with other software) into proprietary software. These restrictions have no impact on end users, who are the focus of this study.

FOSS developers and supporters have a variety of practical, social and economic (and even political) motivations. They claim that this software development approach results in more reliable and secure software, because contributors continually inspect and improve each other's code. And they highlight the fact that because FOSS usually comes without license fees, the software is available at little or no cost. Proponents further point out that since FOSS source code can be modified, it promotes "home grown" software development and makes it easier for local programmers to acquire skills. And it is implied that this in turn has a positive effect on the local economy. Many supporters also call for the use

4 Within the community of software developers that share source code, those who support "free software" and those who support "open source software" differ in their philosophical beliefs about freedom and the rights and restrictions their licenses place on users and developers of the software. These differences have implications on the way software is developed and the related business models. However, the report focuses on the perspective of the *user*, and the fundamental rights granted to *users* by free or open source software licenses are similar. For this reason -- and to avoid unnecessary complexity -- the term "free/open source software" (or "FOSS") was chosen for use in this report to cover both "free" and "open source" software. Where differences between free and open source software are relevant to the discussion at hand it is marked explicitely. For a more detailed description of terminology and the various software types, see Annex 1.

of FOSS on moral and ethical grounds: they consider software a building block of the information society that must be free, or at least readily available, for all to use and share. Strong antipathy toward Microsoft (the world's largest software company, which holds a monopoly position in the desktop software market) is also an emotional driver for many who promote FOSS.

Proprietary software (PS)

Despite the recent groundswell of attention on FOSS, proprietary software is still considered by many to be the best choice for public-access computer labs in Africa.[5] Proprietary software is privately owned and controlled, usually by a company. The owners of proprietary software hold a copyright that awards them the exclusive rights to publish, copy, modify, and distribute the software and they usually keep the source code hidden. Most proprietary software companies sell an "end-user license" to people who use the software programme on their computers.[6] The end-user license agreement limits the way the software can be used -- for example, only allowing non-commercial uses -- and it often restricts sharing.[7]

Proprietary software companies have invested in research and development to write new software applications, and they use licensing fees to recover these costs. Proprietary software companies usually keep the source code secret, and build security mechanisms into the software that prevent circumvention of the end-user agreement and inhibit access to the inner workings of the software.

However, PS companies sometimes choose to make the source code available, such as when they need to address the preferences of an important customer. For example, some government clients have voiced concerns that the software they use to handle confidential government information is controlled by a company in a different country, which raises obvious national security issues. Microsoft has addressed this concern with its Shared Source Initiative, where certain clients are given "read-only" access to the source code of a number of software applications and software development tools, including the Windows XP operating system. Microsoft argues that through this type of initiative it is able to offer those aspects of the FOSS software development model that are most important to its clients. FOSS supporters answer that being able to see the code is insufficient because it is the right to modify code that is important. They also point out that inspecting the millions of lines of code that make up the Windows XP operating system is a significant effort that few organisations can afford.

Many proprietary software vendors point out that software cost is not a factor in the public-access context, because they offer special low rates or make donations of software to non-commercial computer labs. Many feel strongly that proprietary software should be used to bring computers to disadvantaged

5 Occasionally the term "commercial software" is used to describe "proprietary software", but the implication that FOSS is non-commercial is incorrect. The term "proprietary software" is used in this report.

6 Not all proprietary software licenses require payment. For example, freeware (software that can be used without paying for a license), abandonware (software that the copyright holder no longer supports or distributes), or shareware (software that can be used in certain environments or for a trial-period free of charge) are examples of proprietary software that might be free of cost to the user.

7 Both FOSS and PS licenses contain certain restrictions. As mentioned above, the FOSS restrictions apply to the development of software, which are largely not relevant to the user of the software. PS licenses place further restrictions on the end user, governing where and how the software can be used and shared with others.

communities. They maintain that the more user-friendly nature of most proprietary software programmes offers the fastest route across the digital divide for non-technical users. They further claim that proprietary software is more reliable because it is developed by professionals and built around customer demands. And because PS is already so widespread, supporters highlight the likelihood that new users will prefer to learn on software that is used by most others. PS supporters also criticise FOSS business models as "unsustainable", and focus on the proven success of existing economic models underlying the proprietary software industry; they argue that widespread economic development can be achieved by growing a local industry based on proprietary software models.

1.4 Software choices in the African context

A number of recent developments have brought these issues to the forefront in Africa. In 2002, the announcement of a Microsoft donation of software to 32,000 schools in South Africa sparked a public outcry from the FOSS community. The South African Government was not in a position to turn down this offer, yet many argued that putting Microsoft products in all the schools would create a generation of learners locked-in to expensive proprietary software. Around the same time, SchoolNet Namibia was rolling out free/open source software, exploring opportunities and drawing attention to the issues. But critics questioned whether there was sufficient technical capacity in African countries to support FOSS on a wide scale. Others wondered if there was capacity to support any kind of software. Similar incidents in other countries have helped place these issues on the public agenda.

Decision-makers choose software solutions for many reasons, including cost, security, and compatibility with existing infrastructure, or sometimes simply because they trust that the most popular products will be the most appropriate for their needs. Those making decisions that influence software use in Africa face increasing pressure from many directions to make the "right" choices, but there are very different views on exactly what that means. The debate about whether FOSS or proprietary software should be used for public access to ICT in Africa makes the matter more urgent, but also more complex: the stakes are higher in Africa because the vast majority of people have little or no access to ICT, so the continent as a whole risks exclusion from the information society.

FOSS groups have emerged in many African countries, and most are actively encouraging migration away from proprietary software. They argue that FOSS presents the best opportunity for achieving universal access to ICT in Africa, and that it reduces reliance on applications from North America or Europe by making it possible for Africans to develop software locally. These FOSS supporters typically include academics, software developers and small local companies; and increasingly larger local and international companies -- including HP, IBM, Novell, and SUN Microsystems -- are also adding their support. On the other hand, multinational proprietary software companies are lobbying African governments with offers of donations and other social investments. When these companies make generous offers of computer equipment and training, which could help to increase access to ICT for citizens, many African governments feel they are not in a position to decline, even if some are concerned that such donations will lead to vendor lock-in and reduce competition in the long term.

1.5 The need for this study

While this debate plays out in online discussion lists, conferences, academic reports, and the offices of public and private sector actors, the impact of the decision-making will be felt first at the ground level: in the computer labs across Africa that are tasked with providing public access to ICT in a effective and sustainable way. Yet there is almost no reliable data on the realities of software use in public-access projects in Africa. It is clear that software choices should be weighed against many factors and embedded in the local context. But too often some aspects are ignored in favour of a narrow focus on technical or economic issues.

Moreover, many well-intentioned recommendations are based on experiences in other parts of the world, but they do not take local African constraints into account. For example, the tremendous lack of capacity and poor infrastructure have a considerable impact on the feasibility of certain software solutions, and the existing skills and support base are fundamental to the rollout of ICT projects. In addition, cost analysis models adapted from developed countries are not directly applicable in this environment, where cost of labour is significantly less relevant compared to equipment expenditure.

Governments, international donors, non-governmental organisations (NGOs), small and medium businesses and many others are looking for advice about how to navigate these issues and determine which software is "better" over the short and long term. As initiatives aimed at tackling the digital divide in Africa move forward, they are seeking objective information about whether and how they should support one software environment over another. They need facts, based on ground-level experience in Africa, and analysis that places these facts into the policy-level context. At the ground level, computer lab operators also need help understanding the pros and cons of the different types of software. Often the local computer lab managers have little influence on the choice of software, because they lack resources and depend largely on donations. But in order to develop a sense of ownership and empowerment, local staff members and users need to be aware of the different options that exist and how they compare.

Intended audience

This report is targeted to both policy-level decision-makers and managers of computer labs, but it may also be relevant to the interested layperson. It offers a detailed description of the real issues faced by public-access labs in Namibia, South Africa and Uganda; describes the impact of policy-level decisions on computer labs, their staff, and end users; and gives an overview of how the software debate is playing out in an Africa context more broadly. The report does not specifically cater to the interests of the private sector. However the results are relevant in particular for (1) companies involved or interested in public-private partnerships (PPP) activities that need to understand the implications of their technology choices or mandates, and (2) developers of FOSS and PS applications, who will gain a better understanding of the end-user needs and requirements in the public-access environment. Hopefully this will lead to further improvements of the software to address the challenges of the local context.

1.6 Roadmap to the report

This report on the software comparison study is presented in four sections:

Introduction and Background: Sections 1-3 of the report include this introduction to the issues and the report, together with a high-level description of the methodology used in the field study, and an overview of the current situation in the countries and labs studied.

Findings: The presentation of findings in Section 4 looks through the lens of bridges.org's *Real Access/Real Impact* framework to consider the issues affecting and affected by software choices in the African context. Each issue starts with an overview of the criterion and how it is relevant to the public-access environment in Africa. A description of what was seen in the computer labs gives a closer look at the trends that emerged from the field study, and specific examples highlight additional points worth noting. The impact of policy-level issues is also explained, and a short overview of the issues around financial sustainability is provided. The findings are based on information collected through the questionnaire, lab visits, and interviews. They combine local observations with background from existing reports and studies.[8]

Observations, recommendations, and conclusions: Sections 5-6 present broad observations, recommendations and conclusions based on the results of the field study. The first part recalls key findings of the study; highlights lessons learned by public-access computer labs; and considers their obstacles to software use. This section raises both short- and long-term implications of the software choice and provides a set of recommendations. It draws on best practice in the field to identify the potential pitfalls and opportunities that public-access decision-makers should be aware of. Short concluding remarks wrap up the report.

Annexes: This report includes a number of component parts which are themselves separate outputs resulting from this study, and they are included here as Annexes. For example, the innovative methodology used -- which brings together a research concept with a *Real Access/Real Impact* approach -- has been tested and adapted through this study, and it is itself a project output. It could be used as a foundation for similar studies in other countries. Other outputs include an annotated bibliography of research materials and an annotated list of stakeholders and initiatives. The statistical analysis of the questionnaire data is provided as foundation for other research studies. (In addition, the online version of this report will provide links to a range of resources relevant to public-access labs in Africa.)

8 Prior to this assessment no comprehensive review had been done to analyse these diverse materials together in the African context. A range of information was compiled as part of the background research, including from local and international research bodies. An annotated bibliography of these sources is included as Annex 5 and a list of key local actors is provided as Annex 8.

2 OVERVIEW OF METHODOLOGY AND APPROACH

The software comparison study had two main objectives:

1. To examine ground-level realities and direct implications of the choice between free/open source and proprietary software in the African context, and
2. To inform decision-making so as to optimise public-access to ICT in Africa.

The study linked ground-level and policy-level issues related to software choices. The research methodology was developed from a combination of literature review, impressions collected during an initial scoping study, and application of the *Real Access/Real Impact* framework (described below). The report's findings are based on quantitative and qualitative data collected in 2003-2004. Researchers visited 121 computer labs, and collected information from lab staff,lab managers, and policy-level decision-makers in Namibia, South Africa and Uganda. The collected data was analysed by an outside statistical consultant and combined with background research where appropriate.

2.1 Framing the study with a Real Access/Real Impact approach

Especially in the public-access environment, a comparison of FOSS and PS cannot only focus on economic and technical criteria. Software choices are not really about the technology, they are about people -– the lab managers that implement software solutions and the computer users, their level of awareness and expertise, the perceptions they have of certain solutions, and the circumstances of their daily lives that influence the value ICT can have. And at the same time, there are any number of variables in a computer lab that can affect or be affected by the choice of software that is used. In most cases it is these factors -- the so-called "enabling environment" that surrounds software use -- that will determine which software option is the better choice in any given circumstance. To get at this range of issues, this study used bridges.org's *Real Access/Real Impact* approach to frame the research methodology; identify gaps; provide a logical framework for data collection and interview questions; structure a broad analysis of the enabling environment; and support a comprehensive understanding of all aspects of the choice between FOSS and PS.

The idea behind the *Real Access/Real Impact* framework is that, despite the potential benefits offered by ICT, computers and connections will mean nothing to people in developing countries if they do not use them effectively. People may have physical access to very useful technology, but they will not use it if it is not appropriate to their needs, if they cannot afford to use it, if technical support is unavailable, if it adds too much burden to their already busy day (or even if they just perceive that it will), or if there are laws that limit its use. So in order for ICT to have a *Real Impact* on ground level development, people in developing countries need to have more than just physical access to technology, they need to have *Real Access*. This study tackled the two objectives above by considering the following criteria that affect software choices and ultimately determine whether people have *Real Access* to ICT in Africa's public-access computer labs:

- Appropriateness of software to local needs and conditions;
- Software costs and affordability;
- Locally relevant and useful applications, content and services;
- Capacity-building for end users;
- Technical capacity of computer lab staff;
- Availability and quality of technical support;
- The impact of awareness on software choices;
- The effects of policy and political will on software choices; and
- Self-sustainability: a critical factor for computer labs.

2.2 Data and information collection

The study combined quantitative and qualitative data collection. Information was collected from background research, site visits to computer labs, unstructured interviews, and questionnaires.[9]

- **Background research.** Bridges.org conducted extensive background research on software choices and public access to inform the methodology. Based on this research, and together with input from SchoolNet Africa on the role of ICT in education, an initial list of issues were defined for detailed analysis.

- **Methodology / scoping study.** During an initial scoping study, 14 computer labs in Namibia and South Africa were visited and individuals from 27 organisations interviewed to gain a better understanding of and further refine the list of issues.

- **Main field-study computer lab visits.** The data that informs the findings of this report was collected by five researchers during site visits to 28 labs in Namibia, 52 in South Africa and 41 in Uganda in 2003-2004. The visits helped to put the study into the context of the practical conditions that the computer labs face, and many of the findings and data analysis are set to the background of researchers' impressions from these visits. During the visits, computer lab staff filled out a questionnaire and participated in a structured interview.

  - Questionnaire. A comprehensive questionnaire that asked about the computer labs' experiences with computers and software was used to collect information for analysis of the issues mentioned above. The questionnaire was developed using the *Real Access/Real Impact* framework and combined questions on real-life experience, awareness, and perceptions held by lab staff members.

  - Structured interviews. In-depth interviews were conducted with the computer lab staff using a structured approach. Interviews were carried out with staff members in all 121 participating labs, after the staff member had filled out the questionnaire. Most of the interviews were conducted one-on-

9 For more information on the data collection and methodology, see Annex 2.

one and used a basic list of questions, but left sufficient room for the conversation to move to topics that mattered most to the respondents. Some of the guiding questions included:[10]

- Please describe how the software used in the lab was chosen.
- Do you have any plans for the lab to change its choice of software in the future? If so, please explain the rationale for these considerations.
- Is there any software that you or the users would like to have, but do not? If not, why don't you have it?
- Are there any laws or government policies that affect the use of computers in this lab or the choice of software?

2.3 The computer labs studied

The sample of computer labs studied is not representative: the breakdown of FOSS, PS and multi-platform labs in the sample is not proportionate to the overall breakdown of labs in each respective country as a whole. The number of each kind of lab studied in each country also varies (for example, the South African sample contains more PS labs and the Namibian more multi-platform labs). Many factors influenced sample selection, but the general focus weighed toward equal treatment of FOSS and PS labs rather than demographic representation.[11]

Type of lab:	**Total: 121 labs**		**Namibia: 28 labs**		**South Africa: 52 labs**		**Uganda: 41 labs**	
Free/Open Source Software (FOSS)[12]	35	29%	9	32%	15	29%	11	27%
Proprietary Software (PS)	39	32%	4	14%	29	56%	6	15%
Multi-platform	47	39%	15	54%	8	15%	24	58%

Table 1: Field study -- Overview of study sample (country breakdown)

In Namibia, the sample selection was designed to include computer labs from the country's most populous areas: the capital city Windhoek and the northern area around Oshakati and Ongwediwa (which is partly very rural). The intention was to visit about equal number of FOSS labs and PS labs, but researchers often found once they arrived on site that labs had different software than they had expected.

In South Africa, the small number of FOSS labs at the time of the study made it feasible to try to identify all the FOSS labs that could be found and then match them with a roughly comparable proprietary software lab in a nearby area. In

10 For the full list of questions, see Annex 2.
11 For more information on the sample selection and lab breakdown, see Annex 2.
12 Awareness of the differences between "free" and "open source" software was limited among the sample group of the research. The term "open source" is more commonly known and was used in the questionnaires when respondents were asked to describe the type of software they use. However, in tables that present the results of data analysis -- and for reasons outlined above -- the term "FOSS" is used.

Uganda, labs were chosen in the capital city as well as in the rural areas towards the east and west of the country, where the highest incidence of Internet access is found. During the field study many computer labs initially labelled as "FOSS" or "PS" labs were discovered to actually be running a mix of software, referred to here as "multi-platform" labs. This accounts for the large number of multi-platform labs in the study.

For a number of reasons the majority of computer labs visited were located in schools, including because schools currently offer the most favourable environment for public-access computer labs. School computer labs tend to attract funding from a variety of sources and already have (at least some of) the infrastructure in place to support a computer lab. Schools are also a natural focal point for community activity, especially in rural communities. And because e-literacy campaigns run in all the countries covered, schools are often the first place in a community to receive computers. Of the computer labs in this study, 74% are in schools. For the FOSS labs the percentage is even higher: 89% of FOSS labs studied are in schools (11% are commercial Internet cafés). The PS labs in this study are more diverse: some are in schools, others operate as commercial Internet cafés, and some are located in community centres.

	Total (N=121)		**FOSS (N=35)**		**PS (N=39)**		**Multi-platform (N=47)**	
School -lab	74%	90	89%	31	67%	26	70%	33
Commercial Internet café	7%	8	11%	4	5%	2	4%	2
Community lab	10%	12	0%		15%	6	13%	6
Other	6%	7	0%		13%	5	4%	2
Missing	3%	4	0%		0%		9%	4

Table 2: Field study -- Lab types studied (schools, Internet café, ...)

3 OVERVIEW OF THE COUNTRIES STUDIED

While the study was designed to produce results that would be as broadly applicable as possible to a wide array of countries and settings, practical limitations required a focus on a few representative countries: the research team chose Namibia, South Africa, and Uganda. These three countries offer a diverse environment in terms of awareness levels, the interest and involvement of the private sector and civil society, and the positions governments are taking.

At the time of the study, Namibia had more FOSS computer labs in schools than any other African country, yet the Government entered into a strategic public-private partnership with a multinational proprietary software vendor.[13] The South Africa Government has developed one of the few official government policies regarding software choices and many Government officials have expressed their support for FOSS.[14] At the same time, the South African Government has accepted one of the most comprehensive proprietary software donations to schools worldwide. In Uganda, the Government has not developed a clear position on the issue of software choices, but is conducting a teacher training programme that is based on proprietary software; there are some strong supporters of FOSS in academia and among smaller ICT businesses, but the industry as a whole is dominated by proprietary software companies.

Public-access realities in Namibia, South Africa and Uganda

A variety of government-run telecentre programs exist in the countries covered by this study. They differ in size and focus. For example, multi-purpose community centres (MPCCs) in South Africa are frequently based around a whole suite of e-government services, which include a computer lab and are often co-located with a community hall. In Namibia, most of the public-access labs that are established by the Government are affiliated with Namibia's distance learning programme and housed in libraries. While in each of the countries relatively small groups of organisations are involved in setting up the majority of public-access labs, the approaches they take differ significantly. However, in all three countries, many of the computer labs struggle with similar fundamental problems.

The majority of the computer labs visited in this study have only the minimum infrastructure and equipment necessary to make the operation of computers possible. The size of labs varies widely depending on their economic situation and the source of donations; some of the more affluent labs have 30 or 40 new computers, while poor labs may only have two or three PCs, often second-hand or refurbished. Most computer labs have at least a minimum level of security and furniture, because many donors require these kinds of basic provisions to be met before they will donate computers to a school. But even the most rudimentary conditions are not always in place. Especially in schools, the computer labs are often incorporated into the library, and in some cases existing classrooms have been converted into designated computer rooms. However,

---

13 Since February 2002, SchoolNet Namibia has provided hardware, training and connectivity to close to 500 schools. http://www.schoolnet.na, last accessed October 2004.

14 See http://www.newsforge.com/article.pl?sid=03/06/17/1655215&mode=thread&tid=19, last accessed November 2004. South African Minister of Public Service and Administration, Geraldine Frasier-Moleketi mentioned FOSS on a number of occasions. See the following link for an example: http://www.itweb.co.za/sections/software/2002/0210291210.asp?A=AFN&S=All, last accessed November 2004.

some schools do not have libraries or other appropriate rooms. For example, one small public elementary school in Northern Namibia had placed its three computers in a storage room that had no windows.

It is rare to find air conditioning or even fans being used to cool the labs, and sometimes a heavy price is paid for not sealing the room against heat and dust. For example, the pervasive red dust of KwaZulu Natal in South Africa seemed to get into everything, including the computers. Another lab had a hole in the roof that leaked during the winter, but there were no funds available to repair it.

In addition poor electric and telecommunications infrastructure is a problem, especially outside of the main towns where frequent electricity failures and unreliable telephone lines are common. One computer lab in Namibia was cut off from electricity for six weeks because of public maintenance work, until a volunteer who worked at the school strung a cable from the school's circuit to the main power grid.

3.1 Namibia

Namibia is a sparsely populated country and almost the entire population of approximately 1.9 million is concentrated in two regions: the capital of Windhoek and the northern area around Ongwediwa, Ondangwa and Oshakati. While Windhoek offers good infrastructure and services, computer labs in the rural and poor northern region frequently experience power failures and lack access to many of the services and support options available in the capital.

Namibia has only one fixed-line telecommunications operator (Telecom Namibia) with 120,000 users, and also has only one mobile telecommunications operator (MTC) with 220,000 users. Namibia has a teledensity of 6.2 telephone lines per 100 inhabitants. The national telecommunications backbone transmission system is fully digital and contains 6,000 km of fibre optic cable and 60,000 km of copper wire. While all major towns and routes are provisioned by Telecom Namibia, MTC is in the process of rolling out coverage to areas outside of the major towns. Namibia had 65,000 Internet users (3.3% of the population) as of December 2003, the majority of whom are connected via dial-up.[15]

The Namibian Government has drafted an ICT Bill, which mentions public-access as an important aspect of its overall strategy to bring ICT to the country. The drafting process is finished, but the Bill lacks a political champion for implementation at present. Neither the Bill nor Government procurement guidelines make reference to a particular type of software, but the Government is a large consumer of proprietary software. It signed a comprehensive licensing agreement with Microsoft in 2002 to control the internal use of unlicensed software. Within the administration, the Office of the Prime Minster centrally manages the Government’s ICT strategy and has in the past implemented proprietary software solutions exclusively. The Namibian Government has also entered a comprehensive public-private partnership with Microsoft (described in more detail below). At the same time, the Ministry of Local and Regional Government and Housing, and the Ministry of Higher Education are piloting FOSS applications and are beginning to roll them out internally.

15 For a high-level overview of the country, including key economic and telecommunications figures, see http://www.cia.gov/cia/publications/factbook/geos/wa.html, last accessed December 2004.

Most public-access efforts in Namibia focus on schools and libraries. Namibia has about 1520 primary and secondary schools, but only 30% of these have access to electricity, telephones and other necessary facilities needed for computer labs.[16] The key organisations involved in rolling out ICT to schools are SchoolNet Namibia and Microsoft, together with various partners. These efforts benefit from a considerable number of international volunteers that are often assigned as teachers to primary or secondary schools through organisations such as Volunteer Service Overseas, Peace Corps, and WorldTeach.

SchoolNet Namibia, a non-profit organisation, operates a refurbishment and support centre in Windhoek, and promotes a FOSS-based solution for computer labs in schools across the country. SchoolNet Namibia not only provides computers, but -- through an agreement with the Namibian telecommunications provider -- also offers flat-rate, low-cost Internet connectivity. SchoolNet Namibia set up 80 computer labs during 2003 and 107 in 2004[17] and reports that it has provided hardware, training and connectivity to a total of 500 schools since February 2000.[18] The organisation initially operated only from its Windhoek headquarters, but in 2003 it opened a liaison office in Northern Namibia to decrease support and service times. SchoolNet Namibia has rolled out more FOSS labs to schools than any other organisation in Africa and it is often used as an example for the use of FOSS in education. In addition, the organisation gained a reputation as an aggressive FOSS supporter when its open letter to Microsoft Namibia was picked up by the international press.[19] SchoolNet currently deploys OpenLearn computer labs, an education-focused software package that is developed and supported by the South African company DireqLearn.

Namibia is also one of the focus countries of Microsoft's global Partners in Learning initiative and host to the Namibian Pathfinder project, a public-private partnership established between Microsoft and the Parliament of Namibia in 2003. The project partners set out to establish a number of test sites to develop and implement a new software solution especially suitable for the education sector. Microsoft's Learning Suite (based on Windows Server 2003, Windows XP and the Encarta encyclopaedia) is currently being tested in 13 pilot schools that were selected by the Ministry of Basic Education, Sports and Culture. A number of other organisations are involved in the project, including the Ministry of Education's National Institute for Educational Development (NIED), which is providing teacher training, as well as WorldTeach and the American Federation of Teachers (AFT), whose volunteer teachers are helping with equipment operation in schools. As an additional component of Microsoft's partnership with the Namibian Parliament, a National Computer Refurbishment Centre was established in the second half of 2004.[20]

16 http://www.schoolnet.na/about/history.html, last accessed November 2004.
17 Email from Ebben Hatuikulipi, SchoolNet Namibia, November 2004.
18 http://www.schoolnet.na, last accessed October 2004.
19 For the full letter, see http://www.schoolnet.na/news/stories/msft20021111.html, last accessed February 2005. For press coverage, see http://www.itweb.co.za/sections/software/2002/0211041201.asp?S=Computing&A=COM&O=FRGN, last accessed November 2004.
20 *Namibian Pathfinder Update,* Microsoft, June 2004.

3.2 South Africa

On the whole, South Africa has better telecommunications infrastructure and ICT access than the rest of the continent, but the quality and affordability of access is not on the same level yet as in many developed nations. The proportion of the country's population with access to ICT is growing, but like most other countries, South Africa also has a large internal digital divide, which limits access to ICT and its impact for certain groups of society. South Africa has 11.35 fixed-line telephones per 100 people (that is, 4.84 million fixed lines -- the number has been falling since 2001 mostly due to the growth of mobile telephone networks). South Africa has 18.7 million mobile phone users (of whom 80% are active users) as of June 2004, out of a population of 46.6 million. South Africa currently has only one licensed fixed-line telecommunications operator (Telkom), three licensed mobile telecommunications operators (Vodacom, MTN, and Cell-C), two licensed wireless telecommunications providers (Sentech and Wireless Business Solutions), and numerous value-added network service providers (which are in the process of applying for licences that will allow them to provide Voice-over-Internet-Protocol, or VoIP, services from February 2005). There are currently 3.5 million Internet users (7.4% of the population), the majority of whom are connected via dial-up, although broadband and wireless access solutions are growing in market share.[21]

As in many other countries, the public sector is the largest user of ICT and traditionally a large user of proprietary software. The South African Government began to openly debate open standards and free/open source software as early as 2001.[22] The Government Information Technology Officers Council (GITOC) was tasked to prepare briefings on the issue for the Cabinet, and commissioned background research from the National Advisory Council on Innovation (NACI). In the first version of its discussion document *Free/Libre and Open Source Software and Open Standards in South Africa: A Critical Issue for Addressing the Digital Divide*, NACI argued that it was "time for South Africa to promote open software and open standards".[23] The interest in FOSS is supported by the increasing costs the South African Government is paying for software licenses for its hundreds of thousands of employees. The State Information Technology Agency (SITA), the Government agency that procures software for the public sector, specified its total 2002 budget for software licenses, upgrades and support to be ZAR 9.4 billion, and expects a move to FOSS would bring down costs. SITA's Chief Information Officer estimated the potential license-cost savings that could be gained through the use of free/open source software to be up to ZAR 3 billion.[24]

---

21 For a high-level overview of the country, including key economic and telecommunications figures, see http://www.cia.gov/cia/publications/factbook/geos/sf.html, last accessed December 2004.

22 While the NACI report mentions both open source software and open standards, it is important to draw a distinction between the two terms and concepts. For a more detailed description of free/open source software, see section "Choosing software for public-access computer labs" and Annex 1. For a more detailed description of open standards, see section "Recommendations to decision-makers setting policies that affect African public-access computer labs".

23 To download the report, see http://www.naci.org.za/floss/, last accessed October 2004.

24 According to Mojalefa Moseki, CIO of the State Information Technology Agency (SITA), in an interview with Lesley Stone, *Business Day 1st Edition*. "State to save billions on software." 20 Jan 2003. An exact breakdown of the budget was not available to the researchers, and it is not clear how the expected savings were calculated. In addition, it is worth nothing that according to Microsoft, the total revenue from all Microsoft software purchased by the South African Government for the same time period is less than 111 Million ZAR.

In January 2003, GITOC presented a draft FOSS policy to the Government. The document recommended the use of open standards and procurement of free/open source software in Government in cases where a merit-based comparison shows that either solution is suitable; this position was endorsed by the Cabinet in June 2003.[25] While the policy describes a balanced approach, statements by the Minister of Public Service and Administration as well as officials from SITA indicate an underlying position in favour of FOSS. This is confirmed by a number of Government activities that support the local development of FOSS, such as the creation of Meraka (see below).

While the South African Government has gone further in its support for FOSS than other African governments, it has also accepted one of the largest proprietary software donations on the continent. In 2002 President Thabo Mbeki accepted Microsoft's offer to make software available to all of South African public schools free of charge; according to Microsoft around 6000 schools have received Microsoft software under the programme so far.[26]

In the private sector, a substantial part of the local ICT industry (combining South African companies and local subsidiaries of multi-national corporations) develops, sells or provides services based on proprietary software. A recent Microsoft-funded study by BMI-Tech estimates the economic footprint of Microsoft-related business (direct and indirect contribution to the GDP) in South Africa to be more than ZAR 9 billion.[27] The study included not only revenue from direct sales, but also the activities of channel partners, such as resellers and training providers. The dominant market position of Microsoft is confirmed by the pervasiveness of Microsoft Windows and Office across all sectors of society. However, South Africa's private sector also features one of the continent's most active FOSS communities. A number of small and medium-sized South African companies offer FOSS applications and solutions, implementation and support. And large multi-nationals in South Africa are starting to add FOSS solutions to their portfolios. For example, Hewlett Packard, one of the world's largest computer manufacturers (and also a reseller of Microsoft software), worked with local FOSS experts to develop the "441", a Linux-based computer that allows four work places (monitor, keyboard, mouse) to be connected to one computer.

And not only the public and private sectors show an interest in FOSS. A number of non-governmental and non-profit organisations also raise awareness, develop, and promote FOSS. For example, the *Go Open Source* campaign -- a partnership between Meraka[28] (a government-funded FOSS centre) and The Shuttleworth Foundation[29] (a well-resourced foundation funded by a high-profile South African entrepreneur) -- is using traditional marketing channels, including television, to spread awareness, and engages in a number of ground-level projects to increase FOSS uptake. Translate.org is leading the translation of free/open source applications into African languages; in the second-half of 2004, the Pretoria-

25 http://www.infoworld.com/article/03/07/08/HNafrolinux_1.html, last accessed November 2004.
26 Email from Albie Bester, Microsoft South Africa, December 2004. See also http://www.microsoft.com/southafrica/corp/government.mspx, last accessed November 2004.
27 *WHITE PAPER Microsoft Market Impact Study: A South African Perspective,* IDC/BMI-TechKnowledge Sponsored by Microsoft, June 2004. Press release available at http://www.microsoft.com/southafrica/Press/press-764.mspx
28 Meraka -- http://www.meraka.org.za/postnukeII/
29 The Shuttleworth Foundation -- http://www.tsf.org

based organisation released versions of OpenOffice in Zulu, Northern Soto, Tswana and Afrikaans. And "impi Linux" is a South African version of Linux developed by the Linux User Groups in Gauteng.

In South Africa, a wide variety of initiatives are promoting or implementing public-access computer labs. Over the past few years projects driven by businesses, civil society and all sectors of government -- at city, provincial, and national levels -- have set up telecentres, multi-purpose community centres, and computer labs in schools and libraries.

3.3 Uganda

Overall, ICT access is low in Uganda, however the Government is working to improve the situation. The current population of Uganda is approximately 24.7 million. On average there is one fixed-line telephone per 58 people (1.72 fixed-line telephones per 100 people). As of 1999, there were only 57,200 fixed-line telephones in the country, with a waiting list of more than 9,000 per year and an average waiting period of more than a year. Uganda has 393,000 mobile phone users (one mobile phone per 70 people or 1.43 mobile phones per 100 people) with 83.5% of the total telephone subscribers in 2001 being mobile users. It had on average one computer per 323 people (0.31 computers per 100 people) and 125,000 Internet users (0.5% of population) as of December 2003. According to recent NUA surveys, around 90% of communication facilities in Uganda are concentrated in urban areas. Inter-city telecommunications traffic is carried by wire, microwave radio relay, and radio-telephone communication stations, with fixed and mobile cellular systems used for short-range traffic. International connections include two satellite earth stations (one Inmarsat and one Intelsat satellite) plus analogue links to Kenya and Tanzania. Uganda was the first African country where the number of mobile users outnumbered fixed-line users.[30]

Uganda has no specific policy on software choices to date, and the Government has not expressed a clear position. The National ICT policy focuses on the deployment of e-government, and Linux and Unix are mentioned as operating system alternatives to Microsoft Windows. The Ugandan Government is a large user of proprietary software and has, over the last few years, entered into agreements with Microsoft to address the issue of unlicensed software within the Government. The Government has also recently formed a partnership with Microsoft that provides training to teachers to support the integration of ICT in teaching and learning.[31] In August 2004, the United States Trade and Development Agency awarded the Ugandan Government a grant to facilitate its e-government strategy, and an announcement that it had hired Microsoft led to a heated email discussion among local FOSS supporters.[32] It later turned out the announcement was incorrect with regard to the involvement of Microsoft, but it had sparked a lot of debate and prompted the local FOSS community to collaborate more closely and engage in awareness-raising activities.

30 For a high-level overview of the country, including key economic and telecommunications figures, see http://www.cia.gov/cia/publications/factbook/geos/ug.htm, last accessed December 2004.
31 http://www.newvision.co.ug/D/9/35/424383, last accessed March 2005.
32 http://www.tda.gov/trade/press/2004/August%202004/Aug11_04Uganda.html, last accessed October 2004

FOSS support in Uganda is largely driven by academia, civil society and local small and medium-sized ICT companies. In 2003, Martyrs University in Nkosi started to migrate all of its desktop and back-end systems to FOSS. A partnership between Martyrs and Kampala-based Linux Solutions (a private company that provides FOSS solutions and services) set up the East African Centre for Open Source Software (EACOSS) in 2004, the first FOSS training and certification centre of its kind in Africa.[33] The Women of Uganda Network (WOUGNET) has created a list of FOSS initiatives active in Uganda.[34] According to a Ugandan expert, the local Internet service providers are among the first businesses to use FOSS on a larger scale for back-end servers. Despite a number of active and vocal FOSS supporters, the majority of local ICT businesses are providing solutions based on or including proprietary software, and in the area of desktop software the use of FOSS is still marginal.

With regard to public-access labs, the majority of school labs are created with the support SchoolNet Uganda or the UConnect project. One forum for discussion of software choices is the i-Network, an IICD-initiated interest group of practitioners from the field of ICT and ICT-for-development in Uganda. It has featured debates on the merits of different software solutions for different users.

33 EACOSS -- http://www.eacoss.org/, last accessed November 2004.
34 http://www.wougnet.org/ICTpolicy/ug/fosiug.html, last accessed November 2004.

4 FINDINGS ON KEY ISSUES INFLUENCING SOFTWARE CHOICES

The findings are structured according to the key issues that affect software choices in the public-access lab environment. Each issue is examined in some level of detail, grounded in the original data that was collected during the field study and elaborated with contextual information gathered from background research, interviews with policy-level experts, and feedback from workshops and conferences. In this way the larger issues are introduced, but complemented with real examples from the study to highlight specific points.[35] Additional sections describe the impact of policy on software choices and the factors that determine the sustainability of public-access labs.

4.1 Appropriateness of software to local needs and conditions

The software used in public-access computer labs must be appropriate to the local needs and conditions in Africa. In practice, appropriateness must be gauged in terms of the local environment for each particular lab, and the technical specifications and usability of the software must be suitable to how people and organisations need and want to put computers to use. A wide variety of software is now available, and it is important to think broadly about options for appropriate technology.

Basic functional requirements

Appropriate software solutions for public-access computer labs can be realised using FOSS, PS, or a combination of both. Different FOSS and PS software packages have individual strengths and weaknesses, but good options for both types of software exist to meet basic functional requirements. The standard software set-up in most of the public-access labs in this study includes: operating system, word processor, spreadsheet -- and in labs with Internet access -- email client and web browser.[36]

Ability to run on older hardware

The age of the computers used has a direct impact on the software choices available to a public-access lab. Not only is older hardware more likely to develop technical problems, it may not be able to run the most recent versions of many software programmes.

This study asked the labs about the actual age of their computers, and to gauge how old the hardware is it also looked at processor types and whether computers are second-hand. Many labs of the labs visited in this study use old hardware, including about half that use computers with Pentium I and Pentium II processors, and over half using second-hand computers.[37]

35 The detailed results of the field study are provided in Annex 7; however, their usefulness is limited without the deeper understanding of the context and environment the data was collected in, which has been presented here to help guide interpretation.
36 For more information on the applications used by the labs in this study, see Annex 7.
37 Many PS labs did not provide any information about the processors in their computers, most likely due to a lack of detailed knowledge about the kind of hardware they are using.

New software has increasingly large requirements for processing speed, memory and harddrive capacity, calling for more and more powerful hardware to run it. For example, a current version of a word processor will run very slowly on a computer that is only five or six years old. Since it is very common for public-access labs in Africa to use old computers, they are often prevented from upgrading to the latest version of operating systems and applications. A common argument in favour of FOSS claims that it is more appropriate for Africa because it can run on old computers while PS solutions require new computers. This is true to a certain extent: some FOSS solutions run well on older hardware, as do certain PS solutions. But the latest versions of all the most sophisticated desktop applications -- whether FOSS or PS -- have similar (notably, high level) hardware requirements.

At the same time, older versions or less powerful (and often less complicated) software applications can be well-suited to public-access computer labs because they run faster on older hardware and are easier to use. The point is illustrated by the latest versions of the leading word processor applications, which are full-blown content-management, collaboration and desktop publishing suites, with heavy requirements for processing speed and memory, and complex functionality. But Microsoft Wordpad, for example, is a smaller word processor that lacks many of the advanced features of Microsoft Word or OpenOffice.org, and it is simple to use and runs well on old hardware. 31% of respondents from PS labs and 45% of respondents from multi-platform labs in this study reported that they use Wordpad; staff members in these labs did not indicate that they are concerned about the lack of advanced features or considering switching to more powerful alternatives.

Usability

The usability, or user-friendliness, of applications and operating systems is a crucial factor in determining whether a certain type of software is appropriate for public-access labs where staff members and users have little experience with computers. There are two aspects of software usability that must be considered:

- **Use of basic desktop applications**, such as typing a letter in a word processor or sending an email using an email client.
- **Installation and configuration of software**, such as setting up the operating system, installing new hardware (like scanners or printers), and configuring the network.

User-friendliness of basic desktop applications

Until recently, the greater user-friendliness of end-user applications was widely lauded as one of the greatest advantages of PS. Proprietary software companies traditionally place a great deal of attention on user interface design and guiding the user's interaction with the computer. However, during the last few years great strides have been made to improve the user-friendliness of FOSS applications. In particular, this has been accomplished through the development of interfaces similar to those of popular PS applications, which also make it easier for PS users to switch to FOSS.

> "Old open source is horrible, new (2002-2003) is nice. Proprietary was nice even in 1998 or 1999".
>
> -- Lab manager in Namibia, on developments in FOSS user-friendliness over the past few years --

Recent reviews of the latest versions of KDE and GNOME (two of the most popular FOSS desktop user interfaces), together with the research team's impressions from site visits, suggest that usability of the software applications most common in public-access labs is now very similar for FOSS and proprietary alternatives.[38] But this was not reflected in the field-study responses collected from the labs.

Most lab staff members that participated in this study consider PS applications easier to use than FOSS. The majority of staff members in FOSS and multi-platform labs described both types of software as "easy to use", but it is worth mentioning that 21% of staff members in FOSS labs and 35% of staff members in multi-platform labs rated PS higher than FOSS in terms of ease-of-use. That said, the PS lab staff members are less likely to have experience with FOSS and would not be able to comment, while staff members in FOSS or multi-platform labs usually have experience with PS.

The lack of recognition of FOSS's improved usability among the participating labs could be due to a number of factors. Some labs use older versions of FOSS applications, which are considerably less user-friendly. Also, most staff members in FOSS labs have experience with Microsoft applications, and any alternatives (FOSS or otherwise) that require different ways of doing things are likely to be considered more difficult to use. Perhaps because overall the expertise of interviewees was limited, they tended to want to stick to what little they already know, and some appeared fearful of change. On the other hand, those without any prior experience using computers generally perceived usability of FOSS as adequate.

> "We were not used to the new OSS software. We had only a day's training and it was not a popular software within the community [so there was no one local from whom we could] seek help. We had to fiddle around with it. The kids had not used computers before, however, and did not see anything different."
>
> -- Computer lab manager, South Africa --

38 For reviews of KDE, see http://www.eweek.com/article2/0,1759,1650299,00.asp, and for reviews of GNOME see http://www.eweek.com/article2/0,1759,1650295,00.asp. The relevantine AG has conducted two comprehensive usability studies of Linux. The results of the first study are available in English at http://www.linux-usability.de/download/linux_usability_report_en.pdf, last accessed November 2004. The second study that measured recent improvements in the KDE desktop is only available in German. See http://www.relevantine.de/ for more information.

Ease-of-use in software installation and configuration

While FOSS usability is similar to PS in the application space, the situation is different for software configuration and administration. In most computer labs these tasks are usually performed by a specially-trained staff member or external technical support contractor (or volunteer, as is more frequently the case in the FOSS labs visited in this study).

Since the first graphical user interfaces appeared on the market in the mid-1980s, PS vendors have placed increasing emphasis on automating most configuration tasks so they can be performed by less-skilled (and less-costly) technical staff, or even by end users with only basic technical skills. FOSS solutions have been much slower to follow; they were not originally designed for the non-skilled user, and much of the configuration still assumes at least a basic familiarity with the inner workings of the operating system.

The importance of configuration and administration means that the appropriateness of a particular software solution for a public-access computer lab is closely linked with the availability of local technical skills. In particular, the use of FOSS solutions depends on a technically-skilled staff member or a reliable external partner that can provide configuration and support services.

> "If a Windows computer breaks you can just reboot and it often fixes the problem. On a Linux system you need to edit a file -- most of my users do not know what a file is or what 'editing it' means."
>
> -- Computer trainer, Namibia --

Across all lab types in this study, respondents rated PS higher than FOSS when asked which type of "software applications are easy to set up (configure) for non-technical users as well". In particular, multi-platform labs -- which offer a good testbed because they install and configure both kinds of software -- clearly favour PS (48%); only 2% reported that FOSS is easy for the non-technical user to set up and configure.

Multi-user operating systems

Mis-configuration is a major support problem in many of the computer labs visited as part of this study. In some cases, less-skilled users make changes to configurations without understanding the implications. In other cases, deliberate changes are made by more advanced users who are unauthorised to do so (sometimes with malicious intent). Some changes are easily fixed by local staff, but frequently they lead to serious problems that can only be solved with outside help, and sometimes even require complete re-installation of the software. Some labs described the extreme measures they were forced to take prevent these kinds of problems, including banning low-level users from the lab and closely watching the activities of advanced users.

> "On the first computers that were received, learners deleted MSFT Word completely, and even local support could not fix it."
>
> -- Lab manager, Namibia --

> "We had to close the lab to only learners that had some training. Every time the lab was open to everyone, they were changing desktop options, other options -- out of boredom they would change the configuration. With Windows [95 and 98] computers they can actually damage the machines. With Linux thin clients you can shut them down -- and there is no way you could damage them physically."
>
> -- Lab manager, Namibia --

Operating systems that support multiple user accounts and assign different levels of rights to these accounts can help avoid modifications to the system configuration by end users. In practice, this means end users only have access to their personal files, but they are not able to change configuration options like the network settings. A special administrator account (sometimes called a "root account" on Linux systems) is the only account from which these more fundamental modifications can be made.

All standard FOSS operating systems (including GNU/Linux, FreeBSD, OpenBSD, and NetBSD) and the more recent versions of Microsoft Windows (Windows 2000 and Windows XP) support different levels of user accounts and separate administrator privileges. However, Windows 95 and 98 -- which are still used on desktops in 64% of PS labs and 77% of multi-platform labs in this study -- allow all users to make system changes, rendering labs vulnerable to the problems described above.

Q25: Which Operating System is installed on the desktop computers in the lab? (Tick all that apply)	**FOSS N=35**	**PS N=39**	**Multi-platform N=47**
Microsoft Windows 95 / 98 or ME	2.86%	64.10%	76.60%
Microsoft Windows 2000 or XP	0.00%	58.97%	57.45%
Linux	97.14%	5.13%	63.83%
Mac OS / Mac OSX	0.00%	0.00%	4.26%
Do not know	2.86%	2.56%	0.00%
Other (please specify):	5.71%	2.56%	6.38%

Table 3: Field study -- Desktop operating systems used in the computer labs

Compatibility and exchange of data and documents

Background on the issue:

The ability to exchange electronic data and documents with others is a basic requirement in today's computing environment. However, when files are shared between different software applications -- and even between different versions of the same applications -- problems are often encountered. Incompatibility of file formats means that some software cannot read files created in other applications. For example, a ".sxw" text document created in OpenOffice cannot be opened in Microsoft Word. Some FOSS applications now offer conversion tools that make it possible to open and work in files created in the most common formats of other software programmes, including a few applications that can save files in Microsoft Office file formats. So OpenOffice can read ".doc" files created in Microsoft Word, and save them back to the Word format. Software updates also come installed with conversion tools, which allow new applications to open files created in previous versions; so, for example, Microsoft Word 2003 users can read and save files in a Word 95 format. But conversion tools rarely work flawlessly. Formatting problems are common, so documents look different when they are opened in another programme or version. And complex files like spreadsheets that use formulas or macros (a single user-defined command that executes a series of commands) are especially problematic.

The issue of conversion between different file formats is related to a broader discussion on technical and open standards.[39] Where certain applications enjoy widespread use, "de facto standards" for file formats emerge. A de facto standard is established because so many people use the file format that it becomes a prevailing norm, as distinct from more formal standards that are developed by standard-making bodies. For example, in the case of office productivity applications (the applications most used in public-access labs in Africa), Microsoft Office and the file formats it uses (such as ".doc" files for Word documents and ".xls" for Excel spreadsheets) have become a de facto standard because they are used on more than 94% of all desktop computers around the world.[40]

However, when the specifications of a dominant file format are kept secret and not shared with others, it becomes more difficult to develop conversion tools, and problems are likely to arise. For example, when the "owner" of the de facto standard introduces new functionalities as part of the file format, existing conversion tools cease to function properly. So if Microsoft introduces new functionalities in Word, the OpenOffice conversion tools no longer work (at least until an update of the conversion tool is created.) This can be used to limit competition and tie customers to particular software.

Open standards ensure that all software developers (proprietary or FOSS) have access to the exact specifications they require to design applications that can exchange data.[41] Recent developments in the European Commission indicate that the OpenOffice XML format (also known as OO.o XML) could be adopted by the International Organization for Standardization (ISO) as an ISO standard.[42] This is likely to encourage more widespread adoption of this open standard, which could help put an end to compatibility issues. Microsoft is also trying to encourage broader support for the latest version of its Office file formats, by offering a royalty-free licensing programme for the Microsoft Office 2003 XML Reference Schemas.[43] However, these developments alone are not sufficient to make Office XML schemas an open standard. For example, there is criticism that these schemas mix text and binary formats that can only be read by a computer, which limits the benefits of using XML and the ability of others to implement the schema.

---

39 For more information, see the text box "Open standards" in the section "Recommendations to decision-makers setting policies that affect African public-access computer labs"

40 *Open Source: Open for Business,* Computer Science Corporation, 2004. Available at http://www.csc.com/features/2004/uploads/LEF_OPENSOURCE.pdf.

41 For more information, see the discussion of open standards in the section "Observations and conclusions".

42 "Open Office XML may satisfy ISO", Michael Singer, *internetnews.com,* 28 September 2004, http://www.internetnews.com/dev-news/article.php/3414101, last accessed November 2004.

43 For more information, see http://www.microsoft.com/office/xml/overview.mspx.

African realities:

African public-access labs face a choice between the most popular applications (which define the de facto standard file formats), or alternatives that are likely to have some degree of conversion problems when they exchange data and documents with the standard applications. Conversion problems are aggravated by lack of user skills and training. Users that are aware of the difference in file formats and who understand how to use conversion tools effectively are less likely to experience compatibility problems. Also, there are some simple measures that can help to minimise conversion problems especially in the use of office productivity applications (such as limiting the use of certain formatting), but these require some level of experience and familiarity with the differences between applications.[44] At the same time, compatibility improvements are a key development focus area for the Microsoft Office alternatives (including OpenOffice and StarOffice), so that continuing improvements can be expected.[45]

Incompatibility between files from different software applications and conversion issues are a reality for a significant number of labs that participated in this study: 26% of PS and cross-platform labs, and 37% of FOSS labs reported problems. These problems were seen not only between different types of software (FOSS and PS), but also between different versions of the same applications, such as Microsoft Publisher 97 and Publisher 2000. Sharing files between applications from different proprietary vendors was also sometimes problematic. For example, computer labs using Corel software reported frequent conversion difficulties when sharing files between Corel Wordperfect and Microsoft Word, and between Corel Presentations and Microsoft Powerpoint. Sometimes conversion problems are clearly related to lack of experience, as illustrated by the small number of lab managers who confused the term "incompatibility" with the inability to open certain file types. For example, some reported that they could not open files in the portable document format (.pdf), which they had downloaded from the Internet and erroneously attempted to open in their word processor.

Ability to modify source code and applications

One of the features of FOSS is the user's right to modify the software (by accessing and changing the source code). This provides an opportunity for the development of advanced technical skills and the design of derivative applications that target the needs of specific user groups. FOSS proponents point out that this is particularly relevant to developing countries, where the expected market size might not provide sufficient incentive for proprietary software companies to address all local needs.

To determine if computer lab staff members are aware of the ability to modify FOSS and whether they believe that FOSS provides valuable flexibility, the labs in this study were asked to choose which type of software they think "offers more flexibility and can be modified to suit [their] local needs". Many of the lab staff chose FOSS -- 46% of FOSS labs, 49% of multi-platform labs and 31% of

---

44 "When Word-XML Conversions Get Nasty", Michael Gross, *CMS-Watch*, 20 January 2004. Available at http://www.cmswatch.com/Features/TopicWatch/FeaturedTopic/?feature_id=98, last accessed November 2004.
45 Email from Larry Ciraulo, Sun Microsystems, 8 November 2004.

PS labs. However, overall there is widespread lack of awareness and expertise on this issue, including among the 40% of FOSS labs that said they do not know which type of software offers the ability to be "modified to suit their needs".

The right to modify source code can be relevant to public-access labs in two ways. First, there are some expectations that local users will start reviewing source code, learn about the way the software works, build high-level skills, and modify the software they are using to better fit their specific needs. However, the reality is that this requires considerable technical skills, which by and large are not available in the public-access computer labs that participated in this study. For example, virtually none of the lab managers, let alone users, have the skills to analyse or modify source code. And some seemed to confuse the concept of "modifying source code" with changing configuration options provided by a software application (for example changing the paper size, default font, etc.). Therefore, whether source code can be viewed or edited, is unlikely to have a direct impact on the way the African lab managers (or the majority of users in these labs) use the software.

Second, the right to modify source code may enable the development of new software applications that address a local need and are offered to the labs by service providers or vendors. This could be very relevant to community labs in Africa. There are examples where innovative solutions to local problems were developed using FOSS. And the availability of so-called "building blocks" (pieces of software that can be combined to create new applications) significantly aids the development process for applications that might otherwise not be economically feasible in these markets. However, this software development is unlikely to be done by the staff or users in public-access labs. Support and service providers, such as the local ICT industry, government departments focusing on public-access, or SchoolNet organisations, are best positioned to investigate these opportunities.

For example, a number of school labs in South Africa are using a custom application created by Wizzy Digital Courier, which enables Internet use at lower costs.[46] The software was written using only FOSS tools. It connects to the Internet during the night and takes advantage of Telkom South Africa's special offer of ZAR 7 for a call of any length (this is approximately US$1.10, which is relatively cheap compared to daytime rates for telephone calls in South Africa). The Wizzy software then uploads email that was written by teachers and learners during the day, downloads new email that has arrived, and "scoops" a list of specified web pages for viewing on the next day.[47] This application shows the power of a custom solution that is developed to address a specific local problem: the lack of affordable Internet access to many schools. No proprietary alternative exists.

Software activation and registration

Background on the issue:

Proprietary software vendors use a range of technical mechanisms to curtail the unauthorised use of their applications. One of these mechanisms is "registration". Registration usually requires the user to enter a license key to

46 Wizzy Digital Courier, http://www.wizzy.com.
47 The term "web" is used synonymous with World Wide Web throughout the report.

"unlock" the software. Some programmes do not work at all without license keys, others provide only limited functionality or cease to function after a trial period. These applications are only intended for use on one computer, but usually no measures are taken to prevent installation on another computer. Most of these kinds of registrations happen on the user's computer only and do not store any information on the company's server; after the license key is entered into the local application, restrictions are simply switched off.

Microsoft introduced a new mechanism called "activation" with its Windows XP operating system. This is slightly different from traditional registration, in that it assigns the software application to a specific computer. To match software with the hardware it is installed on, a unique identifier is created on the user's computer (the identifier contains information from different parts of the hardware -- the exact details are secret), and is stored in a central server. Activation requires either an Internet connection or telephone call to transmit the identifier.

African realities:

Activation is a fairly standard operation in North America and Europe, and even in many African countries, but it can present a significant obstacle to public-access labs in some African countries. One way this causes problems is when hardware parts are replaced. The older hardware used in many of the labs in this study is more prone to failure and requires replacement of parts more frequently. As each software installation is matched with a specific computer (based on a secret combination of different parts of the computers hardware) these hardware changes can sometimes cause the installed software to stop functioning, because the software interprets the changed hardware as a different computer than it was originally installed on. Another way activation is a problem is when calls and connectivity are expensive or unavailable. In some African countries there is no local activation number so the process requires either a (costly) long-distance call, or else use of the Internet which can also be problematic (for example, if Internet connectivity is unavailable or prohibitively expensive).

During the field study, some of the labs in Namibia either reported problems with software activation, or indicated that they would have problems with activation if they were using software that required it. Overall -- combining data for all three countries -- few labs reported this problem, because in 64% of PS labs and 77% of multi-platform labs older versions of software are still in use. In addition, when Ugandan experts verified why the need for activation was not an obstacle in their country, the overwhelming response was that mostly unlicensed software is used, which is sold or distributed with software tools to circumvent the activation mechanism. In South Africa, at least in the case of schools, software can now be obtained under a type of Microsoft educational license that does not require activation.

Computer viruses and other malicious code

Susceptibility to computer viruses is another key factor informing the choice of software. Computer viruses and other malicious code can cause a variety of problems, such as loss of data and damaged system configurations, which require support and lead to downtime when computers cannot be used.[48] This can be a heavy burden for public-access in Africa, where many computer labs do not have skilled technical support staff in-house, and "disinfecting" the computers might require the help of external support.

Background on the issue:

Today the majority of computer viruses and worms (malevolent programmes that "reproduce" and often exploit an infected computer as a base to attack others) target vulnerabilities in the integration of the Microsoft Outlook email client, Microsoft Internet Explorer web browser and Microsoft Windows operating system. FOSS users are, for the most part, unmolested by viruses. There is a debate around the question of whether PS is in general more vulnerable to attack than FOSS. Some argue that because the source code is open for review by a wide variety of software developers, FOSS weaknesses are better identified and removed. Others counter that it is not about technical vulnerability, but rather that Microsoft in particular is the target of attack because it is more widely used and therefore offers a more effective vehicle to those who seek to do the widest possible damage. However, some commentators point out that if popularity is the main reason for security problems, then Apache (a FOSS application and the market leader in web server software) would also show more security issues and incidents of exploits than proprietary alternatives, yet that is not the case. There is a wealth of research on whether one kind of software is more secure than the other (most of which is partisan), but more detail would be beyond the scope of this study.[49]

The most common distribution channel for malicious code is email and the World Wide Web. For Microsoft Windows-based computers, the effective use of anti-virus software is necessary to reduce the number of virus infections, especially when the computers are connected to the Internet. Another measure taken by some vendors is the integration of personal firewalls into the latest operating systems, which can help prevent worms, trojan horses and spyware.

African realities:

Keeping anti-virus software current is itself a burden on many public-access labs in Africa, again, because it requires costly Internet connectivity. Technical arguments aside, the fact remains that 36% of the PS labs and 53% of the multi-platform labs interviewed in this study suffer from computer viruses at least once a month. The FOSS labs were significantly less likely to suffer from problems related to computer viruses: 94% of the FOSS labs reported that they "never" get infected by computer viruses. In the extreme example of one school in South Africa, virus infections of their Windows 98 computers required the lab manager to periodically remove the harddrives and send them off site for "disinfection" -- an expensive and time consuming process. The computers were old and the harddrives did not have sufficient room to install anti-virus software,

48 The term "malicious code" is used here to combine a number of different categories of applications that are installed on a user's computer without their knowledge or consent, including trojan horses, spyware, distributed denial of service clients, and others.

49 These are two examples of security reports that give opposing views, which together can provide a better understanding of the situation:
(1) "Is Linux More Secure Than Windows?", Forrester Research, 19 March 2004. Available at http://www.microsoft.com/windowsserversystem/facts/analyses/vulnerable.mspx.
(2) "Security: Linux versus Windows", Nicholas Petreley, 22 October 2004. Available at http://www.theregister.co.uk/security/security_report_windows_vs_linux/, last accessed October 2004.

nor could they run it from CD-ROM as the computers did not have CD-ROM drives. The school also used Linux on some computers in the lab and these never got infected by viruses.

> How often does the lab get infected by software viruses? "Whenever we don't have an up-to-date anti-virus software [installed] while using the Internet."
>
> -- Lab Manager, South Africa --

Reliability and stability

Many factors affect the reliability (also termed "stability") of computers, and software is only one of them. Other important factors are the quality of initial software configuration, effective and regular technical support, and the skills of on-site staff (because lack of capacity is often at the root of reliability problems). Impressions and data from the field study suggest that a reliable public-access lab can be run with either FOSS or PS. At a practical level, perhaps the most basic indicator of the reliability of computers in the lab is how many are working (or not) at any given point.

In the labs that participated in this study, all of the computers were working during the field visits at 60% of FOSS labs, compared to 44% of PS labs and 45% of multi-platform labs. However, these figures are related to the total number of computers in the various labs (more computers increase the likelihood that some will not be working). A different way to look at overall reliability is to consider the number of non-working computers within the total number of computers in all labs, so respondents were also asked how many computers were in their computer lab and how many of these were not working. The results are similar regardless of the type of software used: 15% of computers in FOSS, PS and multi-platform labs were reported to be not working. It is worth noting that -- although the sample size was small -- labs using a FOSS-based thin-client configuration were among the most reliable seen in the study: in 10 of the 16 labs (63%) using this configuration all of the computers were working during the field visits. Perhaps this is not so surprising though because thin-client configurations have fewer points of failure (the server computer being the main one), and once they are configured well, non-technical users are prevented from making changes to the configuration settings.[50]

50 For more information, see section on "thin client" and Annex 3.

Q17: How many computers are (generally) fully operational? If some computers are not working, please provide a number.[51]	**FOSS N=38**	**PS N=25**	**Multi-platform N=46**
% of labs with all working computers	60.00%	43.59%	44.68%
% of labs with some not working computers	34.29%	53.85%	53.19%
% of non working computers	14.89%	14.65%	14.54%
Missing	5.71%	2.56%	2.13%

Table 4: Field study -- Working and non-working computers

	All computers working	**Some computers not working**
FOSS (total 35 labs)		
Thin-client / terminal server network	10	6
Individual workstations connected to a server (such as a file server, print server, or Internet gateway)	8	4

Table 5: Field study -- Combined analysis, reliability of FOSS thin-client systems

But reliability cannot only be measured by categorising equipment as "working" or "not working", because computers might not be stable or reliable, but can still be considered to be working. For example, how often computers "crash" is another indicator of a computer's reliability in public-access labs. A crash happens when a malfunction halts the system and requires the user to turn the computer off and restart it. Crashes are not always the cause of software errors, but hardware and also the quality of configuration have an impact as well.

Among the participating FOSS labs, 15% reported daily software crashes while almost 50% never experience a crash. In PS labs, only 3% of labs report daily crashes, and the proportion of labs that never experience a crash is also only 3%. As the comparison between lab types shows, a larger number of FOSS labs show more frequent crashes than PS labs, while at the same time the majority of FOSS labs never experiences crashes; this seems to indicate that other factors have a significant impact on this aspect of reliability. Based on researcher impressions from site visits, the ability of local staff members to maintain and support the computers plays a decisive role in equipment reliability.

Q36: How often does the operating system or applications stop working or "crash"?	**FOSS N=35**	**PS N=39**	**Multi-platform N=47**
Daily	14.29%	2.56%	4.26%
Once a week	20.00%	17.95%	14.89%
Once a month	11.43%	25.64%	34.04%
Never	48.57%	48.72%	36.17%
Other	0.00%	2.56%	8.51%
Missing	5.71%	2.56%	2.13%

Table 6: Field study -- Frequency of software "crashes"

51 The results provided in this table are based on the answers to question 17, combined with total numbers of computers in all labs.

Perceptions about reliability

The numbers of working computers and crash frequencies aside, the overall perception of the computer lab staff that participated in the study is that FOSS is more reliable. When asked which type of software they consider to be "reliable (does not crash)", 33% of all labs chose free/open source software, 11% chose PS and 17% said both types (27% said they do not know). FOSS and PS labs are more likely to weigh their perceptions in favour of the respective types of software they use. Yet multi-platform labs (where running both types of software gives them a good foundation for comparison) answered most strongly in favour of FOSS reliability: 47% said FOSS is reliable, compared to 37% of FOSS labs and 13% of PS labs.

Q66: Please choose the most appropriate answer for the following statement: The software is reliable (does not crash). The statement is most appropriate for:	**All N=121**	**FOSS N=35**	**PS N=39**	**Multi-platform N=47**
Open source	33.06%	37.14%	12.82%	46.81%
Proprietary	10.74%	8.57%	15.38%	8.51%
Both	16.53%	14.29%	12.82%	21.28%
None	5.79%	2.86%	7.69%	6.38%
Do not know	27.27%	37.14%	38.46%	10.64%
Missing	6.61%	0.00%	12.82%	6.38%

Table 7: Field study -- Perception about software reliability

4.2 Software cost and affordability

Once it is determined that both FOSS and PS can provide appropriate solutions for public-access computer labs, the next question is whether these labs can afford to obtain the software. Combined with other elements of the enabling environment, the cost of licensing and using software is a key issue that can determine whether a software choice is the right one for a particular computer lab. This is especially relevant in Africa, where funding is a critical issue for most public-access labs.

Within this context, software licenses make up one part of the total cost of using computers for public access, but there are a number of other important cost factors. At a macro level, significant infrastructure investment is often needed to bring technology to communities that lack electricity, access to telephone networks, or computer equipment. At the micro level, pricey hardware, expensive support, and high costs of telecommunications and Internet connectivity -- especially in remote areas -- are also significant factors. In addition, because of widespread use of unlicensed software and the availability of PS donations, software license costs are often not borne by the public-access labs themselves.

And affordability is only the immediate problem, which shifts to a more fundamental question of sustainability in the long term. Many organisations are able to cover the substantial acquisition costs of computer equipment and other infrastructure for their labs only through donations. After the initial installation is complete however, donors expect the labs to cover ongoing expenses for training, technical support, replacement of old hardware and software upgrades themselves.

Total cost of ownership

The "Total Cost of Ownership" (TCO) framework is used to combine the many factors of cost that apply to the operation of computer equipment -- including hardware, training, and support -- and measure them over the equipment's expected lifetime (usually 3-5 years in the business environment in developed countries). According to several TCO studies, initial capital expenditure makes up only a fraction of product lifetime costs as the costs of support and maintenance often outweigh set-up costs for hardware and software. However most of these studies have been conducted in developed-world computing environments and focus on business settings, and their findings are not directly applicable to the African context of public-access computer labs.

One reason is that the cost ratios of labour, services, and software/hardware acquisition in Africa differ significantly from those of developed countries. In Africa, hardware is more expensive and makes up a larger part of the TCO model; labour and services on the other hand are usually cheaper, and less significant by comparison. Moreover, the harsh environment of many public-access labs causes hardware to fail more frequently; for example, frequent power outages that do not leave time to properly turn the computers off increase the number of harddisk failures.

In addition, many of the TCO comparison studies for FOSS and PS that have been conducted in North America and Europe show contradictory findings. Some conclude that it is cheaper to use FOSS, while others conclude that it is cheaper to use PS. The studies differ in the weight they attribute to the various cost factors and the context in which the software is used. Applying similar generic cost comparisons to the public-access computer lab environment in Africa is difficult, if not impossible. This is due to the unique situation of each lab and the realities of collecting and interpreting comprehensive and credible data about software costs in African computer labs. For example, most of the computer labs in this study lack the accounting practices required to provide reliable cost information.[52] And many issues that relate to cost in the African context are unique to each lab -- external factors with social, cultural, economic and political dimensions -- skewing a generic comparison so that it would not be valid when applied more generally. A better approach is a broad ranking of cost factors and detailed analysis of these factors in the specific environment of each computer lab.

For example, the labs in this study were asked to rate several key set-up cost factors (software licenses, hardware, technical support, and training for staff) in terms of how much they contribute to the total cost of set-up, on a scale from 1 (least) to 5 (most). Hardware constitutes the most significant set-up cost overall: the labs ranked hardware costs at average values of 3.57 (FOSS), 3.58 (PS) and 3.62 (multi-platform). The PS labs ranked software licenses second (average value of 2.56), but most of the labs were set up before large-scale license donations came into effect in Namibia and South Africa. Software licenses were the least significant set-up cost for multi-platform labs (1.49) and FOSS labs (0.79). In FOSS and multi-platform labs technical support was ranked second (2.47 for FOSS labs and 2.22 for multi-platform labs).

Q74: In terms of initial set-up costs, please rate the following in terms of how much they contribute to total cost using numbers from 1 (least) to 5 (most) N=121	**FOSS labs**	**PS labs**	**Multi-platform labs**
Hardware	3.57 (40%)	3.58 (49%)	3.62 (79%)
Software Licenses	0.79 (40%)	2.56 (46%)	1.49 (79%)
Training for staff	1.88 (46%)	2.00 (46%)	1.51 (79%)
Technical Support	2.47 (43%)	2.29 (44%)	2.22 (79%)
Values in brackets indicate % of labs that answered the question			

Table 8: Field study -- Ranking of cost factors for set-up costs

52 During the field-study interviews, many respondents found it difficult to rank cost factors. Less than 60% of respondents answered questions related to cost. The research questionnaires also included a number of questions that asked for specific amounts spent on lab operations, such as software licenses, hardware, training, etc. Unfortunately too few labs were able to answer these questions. For example, only 14% of FOSS labs specified the cost of their server hardware, and only 15% of PS labs specified the initial cost of software licenses. Reasons include lack of cost information (in the case of donations); lack of accounting processes that would enable labs to keep track of their budgets; and caution to release financial data. As a result the data collected was considered unsubstantiated and therefore not included in this report. The results related to cost should therefore be treated as very broad indicators, and no conclusions should be based on these results alone.

Licensing fees

The use of free/open source software obviates the need for payment of the software license fees that are required for proprietary software, and this cost savings is often the first argument brought forward by FOSS supporters. License costs are generally a higher proportion of the total costs of running software in developing countries as compared to developed countries (again, because hardware is expensive and the cost of labour/services is lower). Also, the actual cost of software licenses when normalised against the national gross domestic product (GDP) figures can be significant.

A much-cited study on open source in developing countries translates the US street price for Microsoft Windows XP and Microsoft Office into local prices by comparing a country's GDP per capita to the GDP per capita of the United States.[53] This goes beyond a straight currency conversion of sales price and allows a comparison of prices within the context of the local economy. For example, a straight currency conversion compares the price of a copy of Windows XP and MS Office XP in the US (US$ 560) to the price of Windows XP Home Edition and Office 2003 in South Africa (ZAR 5347 or approximately US$ 952).[54] However, using the GDP methodology, the US purchase price of US$ 560 would equal US$ 7,541 in South Africa, US$ 11,420 in Namibia, and US$ 79,324 in Uganda. These amounts are out of reach for the majority of public-access projects, but two factors limit the actual expenditure for licenses borne by computer labs that use PS: the widespread use of unlicensed software and the availability of PS donations.

Widespread use of unlicensed software in Africa

> "People rarely discuss 'copyright', rather it is the 'right to copy'"
>
> -- Vincent Waiswa, AITEC Uganda --

The impressions of the researchers in this study are consistent with expert opinions and the findings of industry-funded studies: all agree that the unlawful use of unlicensed software (often referred to as "software piracy") is very common in many African countries.[55] The Business Software Alliance, an industry group that represents the interests of many large software and service companies (but which counts both PS and FOSS supporters among its members), keeps track of the occurrence of unlawful software use around the world.[56] Its 2004 *Global Software Piracy Study* asserts that the piracy level for

53 "Licence fees and GDP per capita: The case for open source in developing countries", Rishab Aiyer Ghosh, *First Monday*, volume 8, number 12 (December 2003). Available at http://firstmonday.org/issues/issue8_12/ghosh/index.html, last accessed July 2004.

54 For the US price see the above study. The SA price was taken from http://www.digitalplanet.co.za/dp/products/productdetails.asp?levelid=249&parentid=74&IsLevel=True&manu=Microsoft&mid=6. The currency exchange rate was obtained from Oanda.com's currency converter, and it is based on the average credit card rate; converted with the published exchange rate on 6 December 2004, of ZAR 1 = US$ 0.178.

55 Some people -- especially from the free software community -- disavow the use of the term "piracy" in this way, because they feel it creates a negative and misleading perception.

56 There is some criticism that the BSA is biased toward large multinational software vendors, but the comprehensive research on software piracy that it publishes is conducted by IDC, an independent market research firm, and no empirical data is available that would contradict the findings of the study.

the general Africa category (which is called "Other Africa" in the study and includes Uganda and Namibia) is 81%.[57] The implication is that 81% of proprietary software installed and used in these countries is not lawfully licensed. South Africa, on the other hand, is one of the few African countries that was individually assessed, and it ranks among the 20 countries worldwide with the lowest piracy rates, at 36%.

> "In Uganda only a few corporate companies and a handful of government institutions purchase software. The rest of the businesses you see on the streets of Kampala use pirated software. It's unfortunate but the reality. The weakness of the copyright laws or even there non-existence has made it hard for the likes of Microsoft to enforce license fees, and as a result people believe software is free and [therefore] cheap."
>
> -- ICT entrepreneur, Uganda --

The widespread use of unlicensed software offers one explanation for why a significant number of the Ugandan computer labs in this study answered "proprietary software" when asked which type of software they considered to be "cheap", despite its relatively high local prices (referring to both actual prices and the GDP comparison study mentioned above). Ugandan experts indicate that another reason for this perception could also be the availability of low-cost support and training services for PS, which are cheaper than that available for FOSS thanks to stiff competition in this market.

While it is still commonplace today, the long-term viability of using unlicensed software in Africa is questionable. Organisations like the Business Software Alliance and proprietary software vendors (notably Microsoft) are lobbying African governments to pass more restrictive intellectual property legislation and to effectively enforce it. And governments aiming to join international trade bodies or intellectual property agreements are pressured to crack down on copyright infringements in their countries. Some FOSS supporters argue that the license-based business model of proprietary software companies is outdated and rather than trying to curb the illegal use of unlicensed software, governments should devote their energy to helping develop and promote FOSS alternatives.

Weighing costs related to software donations

Donations of hardware, software and in-kind services can help make ICT more widely available to people in Africa and many public-access computer labs have come to rely on them. Over the past few years a number of high-profile software donations have eliminated (or at least reduced) software license costs for certain beneficiaries. For example, Microsoft and other proprietary vendors have given

57 *First Annual BSA and IDC Global Software Piracy Study*, Business Software Alliance (BSA), July 04. Available from http://www.bsa.org. The information in the report is less detailed for most African countries than for other parts of the world. No African country was part of an in-depth survey of 15 countries, very few African economies are observed permanently as part of IDC's PC Tracker and Software Tracker surveys, and data for Africa was taken at a regional level based on IDC's rest-of-region estimates. According to the sub-Saharan IDC partner, individual country data was aggregated to regional reports.

away software at no cost and offered substantial discounts to schools and non-profit organisations in many countries around the world, including Namibia and South Africa.

However, there is a concern that donations of specific brands of software can create a degree of vendor lock-in, where computer labs are effectively tied to the use of that software in the long term because once they start with it, it is harder to change later. And when a donation is only a once-off, users pay for upgrades in the future. It is also worth noting that software donations often only cover one aspect of software-related expenses, so labs have to pay for hardware, software installation, technical support, training and other costs just to put themselves in a position to benefit from the donation. When vendor lock-in happens at a large scale it can have a significant economic impact, including to the benefit of the software companies that made the initial donation. And as a result, software donors are frequently criticised for the commercial intentions behind their acts of goodwill. Yet, in Africa, where one priority is getting technology -- any brand or type of technology -- more widely used to improve things like healthcare, education and other government services, the threat of lock-in to proprietary software seems less important to many.[58] Furthermore, donations do help make computers available to communities that could not otherwise afford them, and many schools and other public-access computer labs are in no position to turn them down. For example, of the labs in this study, 54% cover initial set-up costs (not only software licenses) with the help of donations, government grants or subsidies.

Vendor lock-in and migration costs in the African context

As mentioned previously, Microsoft made a donation of software available to South African public schools in 2002. It now costs these schools less to obtain a copy of Microsoft Office (which is shipped to them when they sign an agreement) than it does to download a free copy of OpenOffice from the Internet and burn it onto a CD-ROM. This donation helps solve a problem for those schools that are in a position to take advantage of the offer (notably those schools that have electricity and computers), by making it simple and cheap for them to obtain software.[59] But critics argue that the South African Government made a grave mistake in accepting this donation, because it provides an opportunity for Microsoft to achieve vendor lock-in among the students and teachers of the 32,000 eligible schools.

However, some commentators point out that a discussion of vendor lock-in should distinguish between the lab management and the users. They maintain that donations alone do not create future customers and that vendor lock-in is a less important issue for users of the computer lab if they have received generic computer training. They argue that the widespread use of certain software in the business community (for example Microsoft Office) and aggressive marketing

58 This is one of the few places in this report where a distinction between "free software" and "open source" software is necessary. Supporters of "free software" would disagree that the threat of vendor lock-in is less important in developing countries and argue that the type of software is a crucial decision as it relates to the "freedom" of the users. Open source supporters are more likely to focus on the purpose of technology use, but believe that open source software offers the more effective solution that proprietary software.

59 According to Microsoft around 6000 schools have received Microsoft software under the programme to date. Email from Albie Bester, Microsoft South Africa, December 2004. See also http://www.microsoft.com/southafrica/corp/government.mspx, last accessed November 2004.

efforts that focus public attention also influence software decisions.[60] Users might investigate migration from PS to FOSS (defeating vendor lock-in) for a number of reasons -- including the expectation of reduced costs, an interest in learning how the software works, a desire to customise software for specific needs, or simply the wish to be part of the FOSS community -- and in most cases an initial familiarity with PS is an additional factor, but not a deterrent to these considerations. Or at least it has not been a deterrent for many of the FOSS users in North America and Europe who started out on proprietary platforms and later changed to FOSS.

In Africa, the switch might not be quite as easy; a number of barriers exist in the African context that may discourage computer labs from moving away from the software they start out with initially, regardless of whether it is PS or FOSS. Perhaps most important is the concern that a software migration will cause the initial investment in the computer lab to be lost. For example, where a lab has devoted scarce resources to provide staff with technical training for a particular type of software (especially advanced skills like software installation and configuration), then it may be unwilling to sink additional resources into training for a different type of software later on.

However, especially in the FOSS labs, many staff members and users have little or no previous experience with computers so little or no investment in training has been made that would be lost. For example, in 34% of FOSS labs the technical support staff member had less than one year experience (only 3% of PS labs had similarly inexperienced staff) and 57% of FOSS support staff members said that they have not received sufficient training to fill their support role. A second factor is the direct costs of migration, including installation, down-time (during which the computers can not be used), and reduced efficiency as users adapt to the new system. Yet in the community lab environment the monetary value attributed to down-time and lower efficiency is not as significant as it might be in a business environment.

Once-off donations and future upgrades

Some critics of proprietary software donations argue that once-off gifts are expensive for computer labs over the long-term because costly software upgrades are inevitable at some point. Problems can occur if software is not upgraded for a long time and becomes very out of date, such as the inability to run new applications (on an old operating system) or to open and save new file types to exchange information with others. However, the main obstacle for upgrades is often not software license costs, but the cost of hardware required to run newer versions of the software. In addition, the need for upgrading software is often over-estimated: new is not automatically better. The reality in many public-access labs in Africa is that most users only draw on a limited part of software functionality, so the benefits of upgrading to newer software versions are insignificant and might not justify the cost of upgrading.[61]

---

60 Unfortunately there is no available research that analyses how the use of a certain kind of software in Africa creates vendor lock-in.

61 For more information, see the sections "Recommendations to public-access computer labs" and "Ability to run on older hardware".

Commercial motivations for social action

Product donations are one way that software vendors can help foster socio-economic development, which costs them very little but can have a major impact on the lives of users in the public-access computer labs that receive the donated products. But this is not just about socially-minded intentions, social investment programmes are designed to support corporate goals, and software donations usually have some degree of commercial motivation behind them. The fact is that donations put software into the hands of potential future customers. And as companies hope that consumers will to stick with the brand they are familiar with when they make purchasing decisions in the future, donations can help them enter new markets or shore up their hold on existing markets. In particular, software donations to public-access computer labs are commonly criticised (especially gifts to schools) because vendors can use donations as a device to build relationships with governments in the hope that it might influence lucrative procurement decisions to their favour.

But many commentators ask, "so what?". All companies should take concrete social action to help improve the world, and giving what is easiest to give is an obvious first step. In many ways, software vendors (and many other corporate givers) are "damned if they do, and damned if they don't"; that is, they are criticised for commercial motivations if they make donations, and they are criticised for failing to help even in the most basic ways if they do not. The private sector is increasingly requested to pull some of the weight for social and economic development, and commercial motivations as part of these activities should be expected. Transparency about public-private partnerships, corporate donations, and procurement practices can help governments and businesses remain above criticism.

Other factors that affect cost

Three additional factors that affect cost in the public-access environment in Africa were mentioned in computer lab interviews:

- **Local procurement channels for FOSS are missing.** Many of the latest FOSS applications must be download from the Internet, which can result in considerable telecommunication charges and in some cases the available bandwidth does not even allow download of large software programmes. For example, a large university in Uganda was unable to download the latest version of the RedHat Linux distribution (which consists of hundreds of megabytes of data) due to slow Internet connectivity. To acquire the software, it asked an organisation in South Africa (with better Internet connectivity) to download the programme, copy it onto a CD-ROM, and send it by postal mail. To address this problem, some efforts are piloted that would lower the cost of obtaining FOSS and make it more available to more people. For example, the Shuttleworth Foundations Freedom-Toaster is a simple computer placed in a public environment (such as a university) that allows burning of CD-ROMs with open source software free of cost.[62]

62 For more information, see http://www.shuttleworthfoundation.org/index.php?option=content&task=view&id=287&Itemid=41; last accessed October 2004

- **Corruption and theft.** This study did not specifically investigate problems resulting from lack of procurement guidelines, corruption, embezzlement or theft. And very few of the participating labs have significant control over the software choice or the budget for set-up costs, so there is very little potential for that level of misuse in the public-access environment. However, some labs mentioned the effect that corruption can have on their ability to purchase computers and software; and given the limited financial means in most labs, the loss of even a small amount of money can put a stop to their efforts to operate a computer lab.

> "When the computers were taken out of the boxes for the first time, 6 of 15 computers were broken; five of these were then repaired by a hired technician. One is still not functional and will need to be replaced. However the computers were bought from a company that today cannot be found any more. [The principal had insisted to choose this supplier over a cheaper alternative.] They were bought without operating systems, which was bought separately. The software was stolen, before it could be installed and the recovered CD-ROMs were kept as evidence by the police for 6 months; it could not be installed on the computers until now."
>
> -- Teacher, Namibia --

- **Cost of keeping a software licenses inventory.** Another factor related to cost -- especially in larger computer labs, and only relevant for PS labs -- is the effort required to maintain an up-to-date software license inventory. At a minimum these inventories record the license number of the software application and the serial number of the computer the software is installed on. The cost in terms of staff time for creating and maintaining such an inventory in labs that use multiple PS applications on each computer can be considerable (especially where staff are inexperienced). And in light of tougher copyright restrictions and enforced intellectual property legislation, organisations must keep inventories to be certain they are not using more licenses than they own -- and to be able to demonstrate this in case of audits.

Thin-client configurations to reduce hardware costs

The cost of computer hardware is a significant hurdle for most of the public-access computer labs that participated in this study. Purchase prices put new equipment beyond the reach of most small businesses, schools, community initiatives, and households in Africa. And there is reluctance among donors to fund the purchase of new hardware for development programmes. Second-hand computers -- which can be refurbished and resold for a low price -- offer promise to address the upfront-cost issue. And plenty of used computers are available. The relatively short lifespan of computers (usually three years in the business environment in developed countries, where most of the donated hardware originates), means that organisations around the world are producing a near-constant supply of used hardware that has to be disposed of in some way. Moreover, the trend toward environmental laws that require computer owners to take responsibility for the disposal of computer hardware means that finding ways to extend the useful life of the computer may be cheaper than simply throwing it away. As a result, unprecedented numbers of second-hand computers are expected to reach Africa in the next few years. In addition, for many labs that already have computers, continuously upgrading equipment is not a possibility. Finding new and more effective ways of using the existing old hardware is the only alternative to increase performance and reliability of their set-ups.

The thin-client model is a software solution that reduces the need for new, powerful, expensive hardware, and allows less-powerful, older, and -- most importantly -- cheaper hardware to be used effectively.

Background on the issue:

A thin-client system uses a single computer as a server, where all of the software applications are stored and run, networked together with a number of "dumb terminals" (or thin clients), which provide little more than monitors and keyboards. The actual processing is done at the server level (where the speed is needed), so the client computers require little memory and harddrive capacity. This way of distributing the work can enhance the overall performance and reliability of old computers. A thin-client system can also simplify technical support, because things like software upgrades and trouble-shooting take place at the server level instead of at each desktop terminal. However, the initial installation of most current thin-client systems and the occasional maintenance procedure -- something that should not be required too frequently in a well-configured system -- demand advanced technical skills.

The slower processors of even the oldest computers are well suited to deployment as thin clients. While all terminals are fully functional, the clients do not have any moving parts, and the only harddrive required is installed in the server with the clients sharing access to it. Since harddrives are among the components most likely to fail in a lab, reducing the total number of harddrives should also make the lab cheaper to run. As a result, the TCO of refurbished computers when used as desktop workstations might be higher than for new computers, but thin clients present a solution that can significantly reduce initial set-up costs and can compete with new computers even in long-term cost analysis.[63] And since thin-clients require less memory and processor speed than a stand-alone desktop computer, even older computers can be made to run with a degree of performance they would otherwise not achieve.[64]

African realities:

Thin-client systems are the most common set-up of computer labs that exclusively use FOSS in Africa. In multi-platform labs, FOSS was mostly used on the server, together with Windows-based desktop computers. According to the collected data, thin-client solutions were seen in almost half (46%) of the FOSS-based labs that participated in this study. However, based on researchers' knowledge of the labs they visited, the real percentage is significantly higher. Participants were often not able to differentiate between desktop computers with their own harddrives that are connected to a server as part of a local area network, and a thin-client system where the terminals are not able to function

63 For more information on an upcoming study that will provide more information on the TCO of refurbished computers, see http://www.openresearch.co.za.

64 For additional background on the refurbishment industry and centres in Africa, see *How to set up and operate a successful computer refurbishment centre in Africa*, bridges.org, 1 November 2004. Available at http://www.bridges.org/refurb/.

without a connection to the server. Most thin-client labs typically use a new computer or powerful refurbished computer for the server, and older hardware for the clients. Only 10% of PS-based labs report they use a thin-client system.

Q21: Which of the following best describes the computer lab?	**FOSS N=35**	**PS N=39**	**Multi-platform N=47**
Workstations and server	40.00%	56.41%	72.34%
Non-networked workstations	8.57%	30.77%	4.26%
Thin client	45.71%	10.26%	14.89%
Other	0.00%	2.56%	4.26%
Do not know	2.86%	0.00%	0.00%
Missing	2.86%	0.00%	2.13%

Table 9: Field study -- Computer lab network configurations

A widely used thin-client solution is the FOSS-based Linux Terminal Server Project (LTSP).[65] Most organisations that promote FOSS-based computer lab solutions in Africa offer the LTSP as part of a complete package including refurbished or second-hand hardware. Another thin-client solution that is designed to further minimise the need for technical support skills is based around SUN Microsystem's Sun Ray "ultra-thin clients".[66] These devices come at a higher initial price than LTSP solutions, but contain the minimum amount of hardware required to connect to the powerful server, which could lower hardware failure rates and the related costs. As part of its Namibian Pathfinder project, Microsoft is currently developing and testing a hybrid system that combines thin-client configuration with fat-client (powerful desktop computers) functionality. The idea is that if network problems prevent the thin-client from connecting to the server, the clients can operate as standard desktop computers. Other thin-client solutions exist, including some based on proprietary software, but none enjoy the popularity of LTSP in computer labs in Africa.

4.3 Locally relevant and useful applications, content and services

In order for public-access computer labs to be meaningful to the daily lives and work of people and organisations, computers must provide locally relevant content, applications, and services. People will only embrace ICT when it is relevant to their daily lives and offers them content and services that they want (and need) to access and use. In the context of socio-economic development, local relevance means things like educational materials, health information, environmental data, or agricultural information services that are useful to people in the communities served. Government information and services provided to citizens through computers and networks (e-government) are another example of useful applications relevant in Africa.

65 http://www.ltsp.org/
66 For more information, see http://www.sun.com/software/index.jsp?cat=Desktop&tab=3&subcat=Thin%20Clients, last accessed February 2005.

At a personal level, the use of public-access computer labs for communication with loved ones or electronic penpals in other countries can be a considerable motivating factor to encourage new users to try computers. The value of relevant and appropriate applications and content is apparent immediately: applications that address a clear user need demonstrate the value of computers, create enthusiasm for computer use in the community, and stimulate further interest. Especially in schools, effective educational content is needed to show teachers and students how computers can support learning. In addition, the usefulness and relevance of software applications are the linchpin that will determine whether computer labs can become financially sustainable -- either through fees the local economy is willing to pay for these services, or through support from donors who invest in the social benefits the computers enable. But regardless of the content, application, or service, its availability in local languages is critical if computers are to be relevant and useful to the communities and groups targeted by public-access computer labs.

Software platforms and applications

Background on the issues:

An analysis of the availability of software applications relevant to the public-access environment in Africa must look at both the software platform and the applications themselves. The platform, or operating system, is the basic programme that allows applications to run on the computer, such as Microsoft Windows (which is PS), Apple OS X (a PS system that is based on a FOSS kernel), and Linux (which is FOSS).[67] There are different versions of Microsoft Windows (such as Windows 98, Windows 2000 and Windows XP), and various versions and "distributions" of Linux that have been developed by different companies (including RedHat, SuSe and Mandrake).

Generally an application is written to work with one particular operating system only: applications written for one platform will not work on another, and developers must decide which platform they write their software for.[68] Sometimes applications even run on one particular version of an operating system platform only, such as applications written for Windows XP that do not run on Windows 98. Application software that runs on top of the operating system can be FOSS or PS, to a certain degree independent of the type of operating system. For example, many FOSS applications (such as OpenOffice, Mozilla, and others) can be installed on the proprietary Windows platform. The fact that people use the terms FOSS to describe many different things -- a software development process, a philosophy of collaboration, the Linux operating system, and others -- adds to the confusion.[69]

67 Apple computers are not widely used in the public-access environment in South Africa, Namibia or Uganda, so this report does not separately consider Apple computers and software.
68 It is possible to create different versions of a software application to run on a number of platforms by using cross-platform development environments and computer languages such as Java or interfaces like .Net. Some software developers also create different "ports" of their applications that provide the same look and feel on different platforms (for example, Microsoft Office is available not only for Windows, but also for Mac OSX), but these require significant development efforts as each port is essentially a separate application.
69 For more information on terminology, see Annex 1.

African realities:

Historically, FOSS development efforts have focused on server software: applications that provide a specific service but do not require end users to directly engage with them. For example, the open source Apache web-server is the industry standard that provides web pages to users browsing the web (it is a FOSS application, but can be installed on different operating systems, including Windows and Linux). However, in the past few years more FOSS desktop applications have been developed, and now a wide range of applications exist for both FOSS and PS platforms. In addition, some of the most popular applications (for example, OpenOffice and Mozilla) are cross-platform and run on both types of operating systems.

It is difficult to estimate whether overall there is more of one kind of software relevant to public-access projects, and in most cases where a PS application serves a particular purpose (such as the Windows Office Suite that provides a word processor, spreadsheet, and other functionality), there is a FOSS alternative (such as OpenOffice, providing similar functionality). However, some PS applications that are suitable for the African public-access lab environment have no FOSS alternative. For example, accounting packages that include African regulations are rarely available as FOSS.

For most labs in this study, both FOSS and PS offer locally-relevant applications that are adequate to meet basic community needs, which generally centre on office productivity and Internet applications. However, when the focus is expanded to include needs for more specialised software -- such as educational software in school labs -- differences between FOSS and PS become more pronounced. In many specialised fields, a wide variety of proprietary applications that only run on Microsoft Windows are available, and fewer FOSS applications exist (especially where the FOSS developer community has not taken a direct interest in the applications).

Understanding users' needs

When asked direct questions about how they identify the needs of the communities they serve, most labs that participated in this study provided only vague answers, giving the impression that overall they lack awareness about how software and computers could be put to effective use in their environment. Few of the respondents had a deep understanding of the needs of their users and potential users, or how they could be addressed through different types of software. This is likely to be one of the reasons why only a small percentage of respondents from FOSS and PS labs (9% for FOSS labs, 8% for PS labs, and 30% for multi-platform labs) said that "availability of needed applications" is a factor that influenced their choice of software. Even in schools (where the needs are clear: computers are there to support education) respondents were often unsure about the kind of software that is available to meet this need and they lack the resources to find and acquire educational applications. In general, the struggle with the basic operation of equipment in most of the labs visited occupies a great deal of the time and attention of managers, leaving them little room for putting more sophisticated applications to use.[70]

70 For more on this issue, see discussion of "availability and quality of technical support" later in this section.

When users want certain software

Some FOSS labs visited in the study indicated that they are considering or planning a migration from the Linux platform to Microsoft Windows, because they want to use certain software applications that only run on Windows, and FOSS alternatives do not exist. For example, one Internet café in South Africa explained that the popular 3D games many of their customers expect can only be played on Microsoft Windows-based computers. One FOSS lab manager at a school in South Africa said she wanted to migrate to Microsoft because the school was interested in teaching computer science and the software recommended by the local Department of Education was Microsoft-based. Another said she wanted to migrate to FOSS but could not do so for fear of "upsetting" the donors who had provided software for accounting packages that only ran on Microsoft. In a different take on user needs, a few FOSS labs described how they were considering a a migration to Windows, because the use of a FOSS operating system prevents them from sharing pirated software offered by friends and colleagues (much of which is Windows-based).

Yet in some cases, the desire to migrate to Windows is fuelled by a general lack of awareness about the FOSS applications, rather than an actual lack of the applications themselves. For example, the manager of a South African FOSS lab mentioned that the lab would like to have Microsoft Frontpage for web page development. A volunteer showed the lab manager that a similar FOSS-based web page editor was already installed on the computers. After testing the application, the manager decided it was sufficient for the needs of the lab; he simply had not been aware that it existed.

Educational software in African schools

As mentioned above, educational applications are a particular example of software that is locally relevant in the African public-access context. Many initiatives are deploying computers in schools and one of their key motivations -- besides increasing e-literacy -- is to improve the quality of learning overall. However, by and large this goal remains elusive. Only 41% of school labs that participated in this study have access to and are using educational software and content. In those labs without custom applications and educational content, computers are mainly used to build general computer literacy skills and (for those labs that can afford it, Internet access). However, many school labs indicated that the interest in computers quickly wanes when there is no specific utility beyond general computer literacy. By contrast, the labs that do have educational software reported that users show a greater enthusiasm for the local relevance of computers.

> "(We) received a Namibia/Multimedia website on CD. Learners love it. They see things from their normal lives and are fascinated. It is great. Kids started showing each other things -- they were laughing as they found information about their own reality."
>
> -- Lab Manager, Namibia --

In the education area, the cost of educational applications and content is a significant hurdle limiting their widespread use. The majority of companies developing education-specific software (usually subject-specific interactive learning applications, most of which run on Windows) and high-quality educational content (generally web-based and accessible with a variety of different software applications, both FOSS and PS) license their products under proprietary or restrictive licenses and require a fee for its use. A number of online resources compile cost-free and FOSS-based educational applications and platform-independent content, but little use of these offerings was seen in the labs studied.[71]

Even labs that use FOSS operating systems turn to PS solutions for educational materials. For example, the OpenLab solution implemented by DireqLearn in a number of African countries is specifically targeted at school labs and uses almost exclusively free/open source software. However, due to lack of free alternatives it includes a proprietary educational content component from learnthings, a Guardian-backed company in the UK.[72] Lower availability of educational software that runs on the Linux platform is also an obstacle. For example, in 42% of the multi-platform labs that participated in this study, the educational software they use runs on the Microsoft Windows operating system compared to only 15% of multi-platform labs that use Linux-based educational applications. But availability of software and content for one platform or the other is not the crucial factor. Although more education-specific applications are available for the Windows platform, the use of educational applications is not more common in PS labs than in FOSS labs: 35% of PS labs and 32% of FOSS labs use educational software.

> "There is a serious lack of relevant content/applications. When the learners find something new it's neat and interesting -- then interest fades away very quickly. With relevant/educational content the interest stays. They use it to study for tests -- the computers are very popular before tests. The educational content that I provide at my school is relevant for the school work, which is key. It is directly useful to the kids. It is also designed in a way that is more fun to use, for example most of the pictures got animations. The content is organized as lessons without a teacher."
>
> -- Lab Manager, Namibia --

Internet content and services

A connection to the Internet opens doors to vast amounts of information and new methods of communication, which has a significant impact on the way people live and work in Africa. Awareness about the Internet is spreading quickly across the continent and the fascination with the content and services available via the Internet transcends social, cultural and economic boundaries. For example, the researchers for this study met children in the poorest South African township communities who are as enthusiastic about exchanging email with their friends as kids anywhere else in the world. Librarians in remote parts

71 For more information and a number of useful resources, see the online version of this report at http://www.bridges.org/foss.
72 http://www.learnthings.com

of Namibia conduct research on the World Wide Web, accessing information on topics like gender equality that would otherwise be entirely unavailable to them. There are many more examples of Internet content and services that highlight why public computer labs in Africa consider access to the Internet an absolute priority.[73]

"I do not know America, but at least I know what's going on in America. On the TV they only show part of it, on the Internet you can go through wherever you want and click."

-- Lab Manager, Namibia --

"If material/information cannot be found in the library, then the library assistant goes to search for it on the Internet. For example there are no books about law and also none about computers. On a specific legal subject for example gender equality the Internet was able to provide the needed information."

-- Librarian, Namibia --

While the majority of labs that participated in this study (78%) had some form of Internet access, practically all lab managers complained about high costs -- often combined with slow connections. Many of the newer, faster and more affordable solutions, such as wireless or asynchronous digital subscriber line (ADSL), are not available in much of Africa, and rarely in rural areas where they could be most useful. For more traditional connectivity, such as dial-up via modems, the costs are often prohibitive for telephone lines, calls and Internet service. Many labs that have Internet access cannot afford to maintain it as a permanent connection. A commonplace picture showed labs that had to restrict access to the Internet after initial funding ran out and staff came to appreciate the costs related to the use of Internet. As a result, at several schools Internet access that was originally intended for learners was only being used by teachers.

Q22: What type of Internet connection does the lab have?	**All labs combined**	**FOSS N=35**	**PS N=39**	**Multi-platform N=47**
Total with connection	78.51%	74.29%	64.10%	93.62%
Dial-up	31.40%	31.43%	35.90%	27.66%
Wireless	16.53%	25.71%	2.56%	21.28%
Satellite	14.05%	11.43%	2.56%	25.53%
Leased-line	12.40%	2.86%	17.95%	14.89%
Other	4.13%	2.86%	5.13%	4.26%
No connection	19.01%	22.86%	33.33%	4.26%
Do not know	0.83%	0.00%	2.56%	0.00%
Missing	1.65%	2.86%	0.00%	2.13%

Table 10: Field study -- Internet access in computer labs

73 The level of Internet connectivity also impacts on the use and value of online technical support, covered in more detail in the section "Availability and quality of technical support".

The differences between type of software and Internet connection are generally related to age and economic situation of labs. For example, older labs are less likely to have Internet connectivity than more recently established labs, and the PS labs in the study tend to be older.

Language and the localisation of the user interface

Localisation of software has become one of the most unanimously-agreed goals of the ICT-for-development community and in particular among FOSS proponents. Many argue that putting software into local African languages will increase universal access to ICT and some even say it could become an effective tool to help prevent the extinction of local languages and protect cultural diversity. Not long ago, most software in Africa was only available in English and French, and to a lesser degree in other European languages; however during the past few years, the user interfaces of the most common FOSS and PS applications have been made available in more local languages.

The bottom line is that a user who does not understand the language of the user interface is unable to interact with the computer effectively. Where English or French language skills are limited, non-localised user interfaces can present an almost insurmountable barrier to many potential users in Africa.

Background on the issues:

The user interface consists of all elements of an application that enable the user to interact with the software, including giving commands and displaying results. It consists of command buttons that prompt a particular action, menus that set out command options, help texts that give tips and guidelines, and icons (little pictures) that represent certain tasks. The operating system's graphical user interface (in Linux referred to as the "desktop") provides the fundamental level of interaction between user and computer. And the applications that run on top of the operating system also have their own user interfaces.

The process of localising the user interface usually only covers translation of the text-based commands and instructions in the software, including the menu text and the words on buttons. However, this task alone is often very challenging, as most interfaces were designed specifically to accommodate the structure and syntax of the English language. For example, concepts like "files" and "folders" and even the term "new" (a common menu option to create new empty documents) do not always translate directly, but software interfaces are not designed to accommodate much flexibility so sometimes imperfect translation choices are made. In addition, in most software applications the visual cues given by icons and the way that the user interaction with the computer is designed are based on a Western understanding of technology. Occasionally localisation goes beyond translation of the words, to modify icons and adjust the way software interacts with the user, to accommodate differences in local customs, culture, and the way that people frame their thoughts; however this involves an extremely complex process requiring a deep understanding of both human and technical factors.

In order to create a localised version of a software application, access to the source code is often required. The freedom to access and modify FOSS source code means that the user interface can be localised by anyone with the interest and the technical abilities to do so. As a result, a variety of international projects, initiatives and organisations have sprung up to provide more FOSS applications in more languages. Whereas proprietary software owners control the source code for the software, so they either make the changes themselves, or they decide who gets the right to translate their software.

> "Many of the icons and many of the ways Western software companies are developing software is based on Western principles. When you are waiting, the turning sand-clock appears -- for people here that does not mean anything. The icon for saving is a little floppy, something the learners can not conceptualise either."
>
> -- International Volunteer, Namibia --

African realities:

During 2004, significant breakthroughs were made in the translation of key FOSS applications into local African languages. For example, OpenOffice.org is now available in Swahili, Zulu, Northern Soto (Sepedi), Tswana and Afrikaans, due to efforts by the Kenya-based OpenOffice.org Swahili team[74] and the South African not-for-profit organisation translate.org.[75] The localisation of FOSS applications often puts pressure on PS companies to create a localised version of their applications as well and PS companies have increased their efforts to address the demand for localised software. For example, as part of the Local Language Programme (LLP), Microsoft South Africa aims to offer local language packages that can be added to an existing installation of Windows XP and Office 2003 at no cost. Packs for Setswana, isiZulu and Afrikaans were announced for December 2004.[76]

Funding of localisation

While FOSS and PS are based on fundamentally different development and economic models, the current financing mechanisms for localisation of FOSS and PS are similar. The reality is that the markets of many African languages are not large enough for most proprietary vendors to invest in local language versions of their software, at least not without financial support or incentives. Likewise, FOSS localisation for African languages also requires financial support, because the FOSS community has struggled to expand its volunteer-based software development model to local language user interfaces (although recent developments indicate that this is slowly changing). So far most localisation efforts are paid for either by directly hiring translators and sometimes user interface designers, or through grants to non-profit organisations that focus on software localisation.

Funding for both FOSS and PS localisation is often requested from the same sources, namely governments and development aid organisations. And many localisation efforts seek local partners in governments and universities. For example, localisation of the recently announced Jambo OpenOffice (the Swahili

74 http://www.kilinux.org/
75 Translate.org is a non-profit organisation affiliated with the Zuza software foundation. It has recently announced the translation of OpenOffice.org into three South-African languages. See http://www.translate.org.za/archives/permalinks/2004-08-19T12_00_00.html, last accessed October 2004. A list of languages/locales supported by OpenOffice can be found at http://l10n.openoffice.org/languages.html, last accessed November 2004.
76 http://www.microsoft.com/resources/government/locallanguage.aspx, last accessed December 2004.

version of OpenOffice.org) was funded by the Swedish International Development Agency and the University of Dar es Salaam, and coordinated by the Department of Computer Science, Institute of Swahili Research and the Swedish consultancy company IT+46.

It is unclear how many potential users are required to present sufficient economic or social incentives to fund the localisation of software. The investment of time and effort might make sense for most operating system interfaces and the major office productivity applications, but there is a vast number of smaller applications that will not attract enough local language users to draw the financial and human resources required for localisation, whether under a proprietary or free/open source model.

It is worth noting that where localisation of proprietary software is funded by an external organisation but users still have to pay for the license to use the localised software, there is, in effect, a double payment. In the case of government-funded localisation, the user pays twice: once by contributing taxes to a government budget that is used to support the localisation, and again for the software license. By contrast, localised FOSS applications can be used or modified by anyone. As pressure to avoid double payment increases, PS vendors are adding the costs of localisation into their product research and development budgets. For example, Microsoft South Africa is funding the translation of Windows XP into isiZulu, Setswana and Afrikaans without external support.

Use and usefulness of localised software in the public-access environment

While several African language versions of OpenOffice.org were recently completed (mentioned above), it is too early to assess how useful this software will prove to be in public-access projects in Africa, since at the time of this study the software had not found its way into many computer labs. For example, of the labs that participated in this study, 83% do not use local language software.

In addition, some factors could limit the usefulness of localised software. For example, one expert highlighted the relationship between literacy and language; he pointed out that in many African countries, those who can read and write are more likely to do so in English, French, or Portuguese rather than their local language, especially since some languages are purely oral. So a localised user interface might not have the desired results, because while many speak the local language, few read and write it. Moreover, a few experts are beginning to argue that localising software will have a negative effect on the use of computers for learning English, French, and Portuguese, which are the standard business languages in their respective regions of Africa, and often required subjects in school. The value of localised software to help increase universal access, preserve local languages, or improve English proficiency can only be evaluated in the local context of each lab, region and country.

> "[The computer is an] excellent tool to help learners to learn English. At [our] schools we have the unique situation where learners use computers from grade five all the way through grade ten. [Computers] build essential basic skills: typing and English. Half of the learners in grade ten do not know how to read English. If you come back five years from now you will see that their comprehension of the official language will be much better. Even from the beginning of grade five to the end of grade five I see great change."
>
> -- Teacher, Namibia --

These are some of the reasons why 59% of the labs that participated in the study answered that the lack of local language is "not a problem" for their users.[77] While no data was collected to support this impression of researchers, the lack of local language software seemed to be a more significant issue in rural areas, where typically few people read or write English well. Common sense suggests that the development of e-literacy is difficult if a user struggles with basic understanding of the interface, and some labs addressed this challenge by urging users to memorise the shapes of the icons and buttons of the interfaces rather than focusing on the words.

Local language problems that localisation does not solve

Localisation of the user interface does not address all problems related to the lack of local language information and content in electronic formats. For example, even if the user interface of a web browser is in a language the user understands, the content it presents might not be. On the World Wide Web, arguably the most important information repository of the global information society, content in African languages is almost non-existent. There are major efforts underway to increase the number of local languages represented on the Internet, but so far they merely help to preserve pockets of culture and knowledge.[78] Another example is the reality that English (and to a lesser degree other major international languages) is the dominant language in software development. As users start writing software (or explore the source code of FOSS applications to learn more about how the software works), they find that the underlying code is written in programming languages that are based on English. In addition, comments (short explanatory text embedded in the source code) that explain what certain aspects of a programme do, are usually written in English as well. Interaction with the global community of software developers also requires working knowledge of at least one major international language.

4.4 Capacity-building for end users

Any software will be insufficient if people do not understand how to put it to effective use as part of their lives or their work, either because they are not trained to use it, or they cannot imagine the possibilities for how they could use

77 The research only looked at whether labs consider the lack of local language software to be a problem. It is beyond the scope of the study to determine the impact on user satisfaction or productivity, or other quantifiable measure of the impact.

78 For more information, see Open Knowledge Network, http://www.openknowledge.net, last accessed December 2004.

it. People will be encouraged to use computers and the software that runs on them only when it is apparent to them that it will have a positive impact on their daily lives. Further, it is essential that people understand the broader potential for technology, so that users are empowered to innovate for themselves and use computers software in creative ways that may not have been envisioned by the public-access lab. In this way computers can be an enabler of broad skills development; not only related directly to the use of ICT, but as powerful tool for development.

Increasing basic e-literacy among the general population is one of the main goals of most public-access computer labs in Africa. There is a debate about whether basic computer skills (which are becoming a prerequisite for employment) can be developed equally well in FOSS and PS computer labs. Some argue that computer training should focus on specific applications that are used most often in the businesses environment -- mainly Microsoft Windows and Office. Others argue that generic skills are better suited to truly empower users.

The results of this study suggest that neither FOSS nor PS offer any significant advantages toward achieving basic e-literacy objectives. Indeed, because user interfaces and functionality of the most common FOSS and PS applications (such as text editors, email, etc.) are becoming increasingly similar, it is easy for training programmes to focus on generic skills to prepare users for different applications (and types of software). For example, a user who understands the basic principles and main functions of an application (such as the main uses for an email client or word processor, the way the menus generally work, how to look for help, and so on) will be able to adapt to almost any programme of a similar nature whether FOSS or PS. However, some external factors influence what type of training is offered. First, this study saw an overall lack of awareness about the similarity of many applications (especially office productivity tools), influencing lab managers to offer training for the applications they know are used widely. In addition, because employers often ask for particular technical skills in job advertisements (for example, specifically calling for experience with Microsoft Word or Intuit Quickbooks), end users frequently request targeted training courses.

As mentioned previously, some argue that FOSS offers users the potential to look behind the scenes at the inner workings of the software, thereby increasing the likelihood that they could become advanced computer users, system administrators or software developers -- all sought-after skills on the job market. This study found little evidence that advanced skills (programming, networking, etc.) are being developed in public-access computers labs in Africa, and where it is happening it is largely due to strong personal interest and curiosity of individuals. One local expert said that there is a certain number of "computer geeks" in the population and these will become computer experts regardless of the equipment they have access to. However, the typical user of the labs in this study has little or no computer expertise. 68% of all labs reported that their users require ongoing assistance, and some of the most basic concepts of computers are foreign to them. The most common skills that are developed by end users in the labs are basic computer literacy, and in some cases more advanced use of office productivity tools.

This study saw little indication that familiarity with certain software applications played a role in the software choices of the labs visited, to some degree influenced by the fact that often these choices were not made by the labs themselves, but donors and technology partners. However, where existing skills are taken into account -- mostly in commercial labs -- it usually results in a decision for the Microsoft Windows platform and applications.

> "[We have] no intention to change to Linux because people are not used to it".
>
> -- Commercial Internet Café, Uganda --

End-user training materials

A wealth of end-user software training materials (for both FOSS and PS) are available free of charge online, including tools for self-instruction as well as for trainers preparing to offer instruction. They range from basic computer-literacy courses that teach the use of common applications like email and word processing, to more technical instructions covering configuration and installation of software.[79] Most self-training courses require some level of prerequisite skills, and new users often need help finding and selecting the appropriate of materials. Frequently the organisations that roll out computer labs in Africa provide information to make it easier for new users to get started. For example, some include a selection of training materials on the harddrives of the computers they deliver, or store them as bookmarks in the web browsers so that users can easily find them online.

Higher-level training courses can also be found online, including course materials used in computer science departments of some of the most prestigious universities.[80] However, these often require substantial prior knowledge about computing, and in most cases they would not be suitable for the average user of the public-access computer labs reviewed in this study.

4.5 Technical capacity of computer lab staff

Employees in public-access computer labs need the basic technical skills to operate the computers and help their clients use them effectively in their daily lives. For this reason, the prior experience and existing technical capacity of staff form another key factor that influences which kind of software is the most suitable for a particular computer lab. Staff expertise with software use is linked to many aspects of successful lab operations, including the ability to solve technical problems locally and to help and train users. Within the broader community served by the lab, the level and type of local capacity for a given type of software has an impact on the availability of local technical support (which costs more if skills are scarce). Application-specific experience and skills are more relevant for staff members than general e-literacy. Skills and expertise also help computer lab staff take a realistic view of the computers' potential,

79 For more information and a number of useful resources, see the online version of this report at http://www.bridges.org/foss.
80 For example, the Massachusetts Institute of Technology (MIT) is publishing all of its course materials online as part of the OpenCourseware project. See http://ocw.mit.edu.

enabling them to think about computers as a tool to address community needs. Some of the schools in the study reported that teachers (as well as learners) who lacked basic ICT skills were initially enthusiastic about computers, but without truly understanding how the computers could be used effectively, their enthusiasm faded and they lost interest. In addition, decision-makers are more likely to choose software applications that they have the most prior experience of using themselves; before making a change they need strong evidence that an alternative is worth the effort of learning new skills and leaving the comfort of using what they know.[81]

Staff members in 93% of all the labs that participated in this study had experience using Microsoft Windows prior to working at the computer lab, and staff in 90% of the labs had prior experience using Microsoft Office (87% of PS labs, 91% of FOSS labs and 91% of multi-platform labs). By contrast, only staff in 33% of PS labs, 40% of FOSS labs, and 45% of multi-platform labs had used a FOSS operating system (such as Linux) before working at the lab where they now work. Looking at prior experience with FOSS office applications (such as OpenOffice), staff in 36% of PS labs, 31% of FOSS labs, and 38% of multi-platform labs reported this prior experience.

	FOSS N=35	**PS N=39**	**Cross-platform N=47**
Q5: Have any of the computer lab staff (including you) used Microsoft Windows before starting to work at the lab?			
Combined "Yes, all" and "Some" answers	91.43%	92.31%	95.74%
Q6: Have any of the computer lab staff (including you) used Microsoft Office applications before starting to work at the lab?			
Combined "Yes, all" and "Some" answers	91.42%	87.18%	91.48%
Q7: Have any of the computer lab staff (including you) used open source operating systems, e.g., Linux before starting to work at the lab?			
Combined "Yes, all" and "Some" answers	40.00%	33.33%	44.69%
Q8: Have any of the computer lab staff (including you) used open source office applications before starting to work at the lab?			
Combined "Yes, all" and "Some" answers	31.43%	35.90%	38.30%

Table 11: Field study -- Staff experience with different software applications

It is worth nothing that staff members' "prior experience" with the software was mentioned as a factor that influenced the software choice in only 6% of FOSS labs, 16% of PS and 25% of multi-platform labs. However during the interviews, some lab managers described that the existing expertise of staff is being taken into account in ongoing software decisions (see the statements from lab managers below). And one or two school lab managers added they would be pushing for a migration to Windows because they were simply more familiar with it and the training they had received to run their new FOSS lab was insufficient.

81 While only a small percentage of lab managers that participated in the study said prior experience influenced the choice of software (6% in FOSS, 16% in PS and 25% of respondents in multi-platform labs) this is partly due to the fact that the decisions are usually taken without involvement of the local labs..

"[We are not interested in switching to Linux]. For one main reason: When current computer teacher leaves only a Microsoft-based lab will continue to be used. About half a dozen teachers are qualified and know how to use the computer and they would freeze on a Linux system."

-- Lab Manager, Namibia --

"[We are] not getting anywhere with this Linux lab and teachers do not know how to use the [user interface] properly: the lab manager would like to revert to Microsoft, as they know how to use it, plus it is the industry standard."

-- Teacher, South Africa --

When staff training is needed

If time and/or money are available, a lack of prior experience among staff can be augmented with training. However, training courses for FOSS are less widely available than for PS. For example, of the FOSS labs that participated in this study, 41% said that training for FOSS is not availalbe locally. Of the PS labs, only 15% said that training for PS is not available locally, and 49% said that training for PS is locally available.

The involvement of skilled volunteers to provide on-the-job training to local staff members is sometimes another option, at least in the short term. Training costs and set-up time can be reduced by building on existing staff skills. Where software changes are made at later stages -- such as upgrades to new versions of software, or migration from PS to FOSS -- staff members are often reluctant to learn unfamiliar applications. Careful change management can address these issues by raising awareness among staff about the reasons for bringing in new software, providing additional training as needed, and allowing time for adjustment.

"There is a gap between the potential of using OSS and realising it, due to lack of affordable training."

-- Lab Manager, South Africa --

“The training period of 1 day did not provide enough skills and knowledge to operate on my own. ... Some of the teachers who were trained for only a day have dropped out of the project in frustration of the lack of skill and knowledge."

-- Computer Teacher and Lab Manager, South Africa --

4.6 Availability and quality of technical support

Just as staff need to be trained to use computers and software effectively, it is equally important that high-level technical support skills are available to ensure that the computer lab can be set up and maintained. Technical support is one of the key issues at ground level that determines the success or failure of public-access computer labs in Africa. The lack of technical support is a major obstacle to technology use in many African countries, and skills transfer should be an element of any development project involving ICT. There are generally two types. One is the technical support needed to ensure that the computers themselves are working. It is common for computers used in public-access labs to require constant monitoring and maintenance, and without local support to fix and upgrade systems, they can quickly become inoperable.[82] The other type, end-user support, is required to help users operate the computers and work with software applications. End users can require a great deal of assistance to empower them to use hardware and software effectively, especially when they are new to ICT and inexperienced. For example, across all labs that participated in this study, 68% reported that their users require ongoing assistance.[83] There is tremendous need for both types of support in Africa.

> "Without support the computers would break and stay broken. They would be completely useless -- just like I found them."
>
> -- International Volunteer, Namibia --

In the countries studied, a range of technical support options are available, but often there is little real choice for labs. Most public-access computer labs in Africa have to take whatever technical support they can get -- relying on what is available and affordable -- and quality varies. Technical support can be expensive, and this is one of the key obstacles that limits access to different types of support. For their part, technical support providers generally only offer support for the software that is most widely used in their market, and they require a critical mass of potential clients before they will invest in building skills and expertise for new software.

Different kinds of technical support

The computer labs in this study use a range of approaches to meet their technical support needs, but there are no fix-all solutions. The available options generally fall into these categories:

- **Internal support.** The lab staff provides different levels of support, often limited by a lack of skills or other responsibilities that do not leave sufficient time to support the lab. 70% of all the labs in this study have some kind of

82 Unless specifically noted, the term "support" refers to technical support throughout this document.
83 One assumption was that students in school labs might require more support than typical users of public-access computer labs and that the large percentage of schools might skew the result; however, when school labs are removed from the sample, the results remain very similar. The sample size after removing schools is too small to allow a reliable comparison, but in this small sample non-school labs report even higher need for ongoing assistance than school labs.

internal support. The percentage is highest -- over 80% -- for the multi-platform labs (which, again, may be because they tend to be better funded and have more experienced staff).

- **External support.** Outside organisations or individuals provide support, generally when the internal support is not able to fix a problem. A distinction can be made between commercial external support that requires payment and non-commercial external support from volunteers or donor-funded organisations. 53% of all the labs in this study use external support at least some of the time.
- **Support clusters / informal support.** A kind of external support that is provided by peers, friends, and colleagues, usually for free and without a formal agreement. One example is lab users that are helping each other. Approximately 60% of the labs reported that this was one of the support options used.[84]
- **Online support.** Support that is provided over the Internet. This can take the form of email/mailing lists (where participants post questions by email and others answer), online discussion boards (a kind of mailing-list website, which does not use email), instant messaging or "chat" conversations, and how-to documents and other resources posted by individuals or companies. 22% of the respondents said they use online support.
- **Automated support**. A support option, which is usually accomplished without user intervention. The computer software automatically upgrades itself based on new releases (often security fixes) made available by the software authors. Similar mechanisms exist for FOSS and PS, but the FOSS model generally requires some action from lab managers or technicians, while the trend for PS is moving toward less user interaction. This kind of support requires an Internet connection and is therefore less relevant to many of the labs visited during the study.

Q47: Which of the following options for support are used in the lab? (Tick all that apply)	**FOSS N = 33**	**PS N = 37**	**Cross-platform N = 46**
Internal support	60.61%	64.86%	80.43%
External support	51.52%	56.76%	50.00%
Support cluster	12.12%	5.41%	13.04%
Online support	18.18%	8.11%	34.78%
Other	0.00%	0.00%	2.17%

Table 12: Field study -- Support types used in the labs

84 When presented with different support options, only 10% of all labs (N=121) chose "support clusters". However, when asked specifically if " the lab share[s] information, experiences, and/or technical support with other labs or staff from other labs", only 39% (N=119) answered "no".

Identifying and addressing technical support needs

Some of the comments made by lab managers indicate that they only truly appreciated the need for good technical support after the computers had been set up and operations were underway. Especially in rural computer labs where support sometimes arrives a few weeks (or months) after a problem is reported, the impact of the lack of technical support quickly becomes apparent once there is a problem.

> "[A key problem is the time it takes for] service/support to arrive. Sometimes it takes 2-3 months for somebody to arrive and repair it."
>
> -- Lab Manager, Namibia --

Computer labs are generally susceptible to the availability, quality, cost -- and perhaps most importantly, the existing software-specific expertise -- of local technical support options. However, overall that was not seen in the labs studied. Only 7% of FOSS labs and 5% of PS labs mentioned the availability of technical support as a factor that affected their choice of software. One reason given was a lack of awareness at the lab level about the importance of technical support. But in the past even those who should know better failed on this score: some labs reported that the reason local technical support was not factored into the choice of software is because software decisions were not made locally, but had been made elsewhere, by donors or government departments. And some lab managers were under the mistaken impression that donor support would continue after the lab was set up, and were disappointed to learn this would not be the case.

Internal technical support

The first technical support contact in a public-access lab in Africa is likely to be a staff member: the lab manager, computer trainer, or local support staff. (In many cases the person interviewed during the field study filled more than one of these positions, and some were volunteers.) In schools, sometimes teachers step forward, get training, and take on responsibility for the computer lab.

In terms of their ability to solve the technical problems faced in the lab, FOSS labs showed a very different situation than the other labs. 26% of FOSS labs reported that their staff are "always" able to fix problems, compared to 8% of PS and 11% of multi-platform labs. However, only 17% of FOSS labs reported that their staff are "usually" able to fix problems, compared to 31% of PS and 43% multi-platform labs. And staff in 14% of FOSS labs are "never" able to resolve problems, compared to 10% of PS and 2% of multi-platform labs.[85] This indicates that it not the type of software used in a lab, but the capacities of local staff (in terms of the amount and quality of support they can provide) that most strongly determines whether technical support needs can be covered internally. Unskilled staff members sometimes fail to perform even the most basic troubleshooting tasks, such as ensuring that the power cables are plugged in.

85 These observations do not take into account the types and difficulty of problems that are encountered.

For example, in one lab a modem was reported to be faulty, but the technician found that it had simply been unplugged when one of the lab staff borrowed its adapter to charge a mobile phone.

Q49: If a technical problem occurs, are you or other staff members able to resolve the problems?	**FOSS N=35**	**PS N=39**	**multi-platform N=47**
Always	25.71%	7.69%	10.64%
Usually	17.14%	30.77%	42.55%
Sometimes	42.86%	43.59%	42.55%
Never	14.29%	10.26%	2.13%
Missing	0.00%	7.69%	2.13%

Table 13: Field study -- Ability of internal staff to resolve technical problems

The suggestion that staff experience is more important than type of software is supported by the fact that in the computer labs in this study, the technical support staff in FOSS labs had significantly less experience using computers than the staff in other lab types. Only 29% of the FOSS labs said their support staff members have more than three years experience using computers (compared to 49% of PS labs and 40% of multi-platform labs). And in 34% of the FOSS labs, support staff members have less than one year of experience using computers (compared to 3% of PS labs and 13% of multi-platform labs). In addition, more PS and multi-platform labs (42% and 45%, respectively) employ staff designated to provide technical support, with job titles of "technical support staff", or "trainer". Only 21% of the FOSS labs have designated technical support staff, and 21% of the FOSS labs draw on "unpaid volunteers" to provide technical support (compared to 15% of multi-platform labs and 7% of PS labs). On the other hand, a few FOSS labs seem to have highly-capable support staff and interviews with lab managers suggest that at least some of the FOSS labs that reported very skilled staff are those that enjoy the services of these unpaid volunteers.

Q41: How long has the staff member responsible for technical support been using computers?	**FOSS N=35**	**PS N=39**	**Cross-platform N=47**
More than three years	28.57%	48.72%	40.43%
1 - 3 years	31.43%	30.77%	38.30%
Less than one year	34.29%	2.56%	12.77%
Missing answers	5.71%	7.00%	8.51%

Table 14: Field study -- Computer experience of technical support staff

In addition, more PS and multi-platform labs (42% and 45%, respectively) employ staff designated to provide technical support, with job titles of "technical support staff", or "trainer". Only 21% of the FOSS labs have designated technical support staff, and 21% of the FOSS labs draw on "unpaid volunteers" to provide technical support (compared to 15% of multi-platform labs and 7% of

PS labs). On the other hand, a few FOSS labs seem to have highly-capable support staff and interviews with lab managers suggest that at least some of the FOSS labs that reported very skilled staff are those that enjoy the services of these unpaid volunteers.

The quality of internal support depends in large part on skills of the staff, but it also requires that staff are not overburdened with other responsibilities so they have sufficient time to service computers and support users. Providing technical support for a public-access computer lab in Africa can require significant time and effort, and balancing other job responsibilities has proven difficult for many staff members in the labs studied. Lack of time is a constraint seen especially in the FOSS labs, where more than 51% of support staff report that other responsibilities do not leave them enough time to run the computer lab effectively; interviews with staff members suggest that this is very likely linked to economic situation of FOSS labs, where staff resources are stretched thin. Only 33% of staff members in PS labs and 28% in multi-platform labs reported that they do not have enough time for technical support.

External technical support

If internal support is not available or unable to fix a certain problem, computer labs need to be able to rely on external support for help. As a result of low skill levels among staff, and not enough time for staff members to spend on support, public-access labs in Africa tend to have significant needs for external support. Only about one-quarter of the computer labs in this study reported that they "never" require external support (23% of multi-platform labs, 26% of PS labs, and 29% of FOSS labs).

While access to outside technical support is common across all lab types, the sources and types of support are very different. External support options are both commercial (provided for a fee) and non-commercial (free of charge). Commercial support is usually sourced from for-profit companies, while non-commercial support options include peer networks of friends, free online support resources, and donor-subsidised support from NGOs. In this study, 63% of the PS labs and 60% of multi-platform labs source external support from local computer companies, compared to only 32% of the FOSS labs. Based on interviews and researchers' impressions of the financial situation of the labs, it can be assumed that many computer labs can only afford a very limited amount of commercial support.

Many of the FOSS labs rely on non-commercial support: 10% source support from local and 16% from non-local NGOs, compared to only 6% of PS labs that receive support from NGOs (combining local and non-local NGOs). While support from NGOs is not necessarily free, their non-profit status and social missions usually result in lower fees. FOSS labs also often have local volunteers (13%) or non-local volunteers (19%) who provide support. The non-local volunteers usually come from a different region of the same country or another country altogether -- mostly from Europe or the United States of America -- and in the best cases these volunteers conduct on-the-job training for local staff. Especially in Namibia, many labs are supported by either local (usually from SchoolNet Namibia) or international (often from Voluntary Services Overseas or Peace Corps) volunteers.

Q47: Which of the following options for support are used in the lab? (Tick all that apply)	**FOSS N=33**	**PS N=37**	**Multi-platform N=46**
Internal support (provided by you or other employees)	60.61%	64.86%	80.43%
External support (provided by outside individuals/organisations/companies -can be free or paid for)	51.52%	56.76%	50.00%
Online support (manuals, discussion groups, email lists, etc.)	18.18%	8.11%	34.78%
Support cluster (support provided by similar labs that you share information with)	12.12%	5.41%	13.04%
Lab users helping each other	42.42%	32.43%	36.96%
Other (please specify):	0.00%	0.00%	2.17%

Table 15: Field study -- Types of support used in the lab

There is widespread anecdotal evidence indicating that the overall lack of reliable external technical support is a significant problem in Africa. However, the situation reported by the computer labs in this study reveals a more promising view. The majority of all labs reported that they have access to outside technical support when needed: 77% of FOSS labs, 82% of PS labs, and 91% of multi-platform labs. Further, the average quality of external support is also better than expected, with 40% of FOSS and multi-platform labs, respectively, and 36% of PS labs indicating that external support services are "always" able to solve the problem. And 86% of FOSS labs, 91% of multi-platform labs, and 79% of PS labs report that external support at least "sometimes" fixes the problem.

Q51: Does the lab have access to outside technical support when needed?	**FOSS N=35**	**PS N=39**	**Multi-platform N=47**
Yes	77.14%	82.05%	91.49%
No	22.86%	12.82%	6.38%
Missing	0.00%	5.13%	2.13%

Table 16: Field study -- Access to outside technical support

Q53: Is this outside technical support able to resolve the problems?	**FOSS N=35**	**PS N=39**	**Multi-platform N=47**
Always	40.00%	35.90%	40.43%
Usually	28.57%	17.95%	25.53%
Sometimes	17.14%	25.64%	25.53%
Never	5.71%	5.13%	0.00%
Missing	8.57%	15.38%	8.51%

Table 17: Field study -- Quality of outside support

Economics of commercial and non-commercial technical support

A non-commercial technical support environment has emerged around the FOSS labs seen in this study, which provides readily available and effective technical support for free, based on donations (as described above, through volunteers or donor funding). While this offers considerable cost-savings for the labs that benefit from it, the sustainability of free technical support through a donation model presents challenges in the long term. Options for maintaining this model apply equally to FOSS and PS environments.

One solution would be for public-access computer labs to be considered a public good so that a long-term commitment to subsidisation is made. This could happen if computer labs become so inextricably linked with essential social programs -- like healthcare delivery, government services, or effective education -- and they learn how to demonstrate a regular, tangible benefit to society. Another alternative could be for computer labs to generate sufficient revenue so they can afford to pay for support when needed. Given the financial situation of many public-access labs (and especially the FOSS labs), this may be easier said than done.

The impact of perceptions about technical support

As a factor informing software choices, the actual availability of support and its demonstrated worth are important, but awareness about the availability of commercial technical support and the perception of its quality (expressed in terms of ability to solve technical problems) are also relevant. If people believe good support is available for one type of software they are more likely to choose that software. While the actual availability and demonstrated success of FOSS support in the labs studied (in terms of the ability of the support technicians to solve technical problems that arise) is on par with PS support, the perception among some labs is that commercial FOSS support is not as good as comparative PS support. Several questions were asked to determine whether the technical support available to the labs actually solved the problems encountered. As noted above, 40% of FOSS labs reported that the technical support "always" solved the problem, and a further 46% said the problems were solved "usually" or "sometimes". Only 6% of labs said that the technical support available "never" solved the problems. Similar results were seen for PS and multi-platform labs. However, 23% of the same FOSS labs indicated that they consider the statement "good commercial support is available" to be more appropriate for PS (and only 14% for FOSS). But equally important is the overall lack of awareness about technical support demonstrated by the answers from all types of labs. For example, 38% of PS labs, 23% of FOSS, and 17% of multi-platform labs said they "do not know" for which type of software the statement is true.

Support clusters

Where commercial support is costly and sometimes located far away from the labs, many computer labs turn to their peers for help and advice. Several of the labs in this study described their use of "support clusters", or informal networks of lab managers, colleagues, and friends that exchange information and help each other with technical and other problems. This not only offers a good way to reduce technical support costs, it also helps build capacity and creates communities of computer users.

Support clusters usually form around common types of technology; for example, those using FOSS solutions will exchange information and share experiences with other FOSS users. The availability and quality of such informal support is therefore highly correlated with the type of skills and experience that are most widely available. The value of informal support clusters is higher for PS and multi-platform labs than for FOSS labs: when asked how often the sharing of information was helpful in for resolving technical problems, 38% of PS and 34% of multi-platform labs said "always" or "usually", compared with only 17% of FOSS labs. This is likely due to the fact that PS skills and expertise are more pervasive across the region, so it may simply be easier for the lab managers to find others (like colleagues and friends) they can talk to who have the experience to help.

Overall 56% of FOSS and PS labs and 78% of the multi-platform labs participating in this study exchange information and experience with other labs. Face-to-face meetings are the most common way of engaging in support clusters, followed by email and online discussion groups. FOSS and multi-platform labs use online discussions and email more frequently than PS labs, relying on the Internet to connect to others for support (13% of FOSS labs and 14% of multi-platform labs connect via online discussion groups, compared to 5% of PS labs; 21% of FOSS labs and 25% of multi-platform labs use email, compared to 13% of PS labs).

Online resources for technical support

Online information and services can provide a technical support mechanism that reduces the need for expensive commercial support and builds the practical expertise of lab staff. Especially in rural areas (where technicians often have to travel considerable distances to get to labs), access to support via the Internet could offer a valuable alternative, but in Africa the high costs and poor quality of Internet connections limit its usefulness. Online support can be provided in a variety of ways:

- **Email/mailing lists.** Questions are sent as email to a group of recipients and answered by other members of the list.
- **Discussion boards.** Similar to email lists, but discussions are stored as websites, and are often easier to navigate and search than mailing list archives.
- **Chat discussions groups (for example, Internet Relay Chat or IRC).** Similar to email discussions, these text-based conversations happen in real-time. Often technical chat rooms are staffed by experts.
- **Online documentation.** Instructions and background information that is hosted on a website and often compiled by volunteers, but also product documentation provided by vendors.

A wealth of online information and services are available to support both FOSS and PS solutions. This research does not provide an objective comparison of the quality of these online resources, but the value they have shown for the labs participating in this study. Many of the labs have the perception that online technical support is a greater strength of FOSS than PS. When asked about the availability of good online support 40% of the multi-platform labs (which use both kinds of software and are likely to have tried using online support for both PS and FOSS) answered "good online support is available" for both types of software, 23% indicated that it is available for FOSS, and 15% said it is available for PS.

However, overall the use of online resources among the public-access labs in this study is low. Of those labs that have Internet connections, 31% never use online support (33% of FOSS labs, 35% of PS labs and 27% of multi-platform labs), and 31% use it no more than once per month. In addition, for many of the labs online support is not useful. For example, 32% of all the labs with Internet access that participated in this study, reported that online resources "never" help them to solve a problem.

Obstacles to use of online technical support

There are many obstacles that limit the value realised from online technical support resources by public-access computer labs in Africa. Two of the most obvious and immediate constraints are the high cost of Internet connectivity and slow connection speed. When the labs in this study were asked "which of the following [options] limit the value and use of online support", the majority of FOSS and PS labs (34% and 44%, respectively) chose "cost", while the majority of multi-platform labs (39%) said that "speed" is the main constraint. However, the answers may be misleading. For example, for labs that struggle with cost, speed is less of an issue until the cost problem is solved, at which point speed might become more important. Since the multi-platform labs in this study are generally in a better financial situation -- and can afford Internet access -- they have the opportunity to experience how the slow speed of the Internet connection is an obstacle to the use online resources which those who cannot afford to connect do not have.

Some of the other factors that influence the value of online support for computer labs in Africa include:

- **Language.** Virtually all online technical support resources are only available in a very small number of international languages (including English, French, Spanish, German, Japanese and a few others). Proficiency in at least one of these languages is a prerequisite.

- **Netiquette.** The online communities of software developers and users have developed their own sets of cultural norms, acronyms and language, which they use in the discussion fora that form the backbone of online technical support. This style of communication is often difficult for many African users, especially new users, because it is foreign and very direct (and occasionally offensive). For example, when someone asks a question to a discussion list that is considered simplistic or redundant, one common shorthand response is "RTFM" (read the f***g manual), an expression that has become an accepted term in online communities, but remains highly insulting to many people.

- **Required levels of expertise.** Many online support resources are directed more to high-level users than to entry-level users, and discussions focus on challenging tasks rather than mundane installation questions. In addition users must be equipped with basic web-searching skills to identify appropriate resources that can provide answer to their questions.

- **Lack of awareness.** There is significant overall lack of awareness about the availability and use of online technical support. For example, 37% of all labs in this study have "never" used online support options.
- **Asynchronous nature of support.** Most online technical support is asynchronous: the user submits a question, and it can take hours, or sometimes days, for an answer to arrive. Other times the answer is received immediately -- or never. Waiting can be frustrating, especially when a technical problem requires an urgent solution.
- **Reliability.** Some of the lab staff that participated in the study questioned the reliability of online support information. However, upon investigation it appears that this concern is based on perception rather than experience (because there is little evidence that they have actually made extensive use of online resources and received unreliable support information).

4.7 The impact of awareness on software choices

Awareness of software options, interest and trust in certain solutions, and public support based on widespread use are important "soft" factors that influence opinions and software choices. In particular, the FOSS movement has historically been driven by a group of tremendously passionate and dedicated supporters, and it is based on a groundswell of enthusiasm. In Africa, awareness of FOSS is much lower, but initiatives are aiming to increase it and a small fervent community of supporters has emerged. These efforts seek to ensure that FOSS becomes a sustainable success story in Africa, by garnering a similar level of support and enthusiasm among the local software developers and end users as is seen elsewhere.

However, currently the awareness of FOSS is low and only a small number of African public-access labs show the same kind of passion that is characteristic for the global FOSS developer community. For 12% of FOSS labs and 14% of multi-platform labs that participated in the study, the "desire to support FOSS" is a factor influencing the choice of software. While FOSS awareness is limited, effective marketing campaigns build brand recognition, and the widespread use of PS solutions provides a popular vote of support.

Impact of lower awareness of FOSS

Lack of awareness and skills are important obstacles for increasing FOSS uptake. First, the fact that users are not familiar with the software -- in combination with the fact that PS solutions are more common -- leads to scepticism regarding the quality of FOSS. For example, some staff members in the labs visited gave the impression that they consider FOSS a lower quality product because it is available at no cost. The study also saw cases where a range of problems were interpreted by inexperienced users as flaws of FOSS applications, and even though they were actually unrelated to the software, it reduced the users' enthusiasm for the application. For example, an Internet problem caused by hardware faults of an unreliable telephone connection was blamed on the software application. The flip-side of the coin is that the widespread use of Microsoft solutions implies quality in a way that goes beyond an objective comparison of the technical merits. For example, the fact that specific PS products are used by so many people is often taken as a proof of quality that decision-makers (especially those with less skills and expertise) rely on.

> "[We are interested in switching to Microsoft Windows], because we think the software on the computers is an old system. It is very very slow. Especially when you are searching on the Internet. It is a slow process. But nobody has the skills/expertise to make an informed decision. But maybe Microsoft could provide a better solution? Because more people and most institutions are using Microsoft -- it should be reliable. But I am not sure."
>
> -- Lab Manager, Namibia --

Second, the lack of awareness among decision-makers -- especially at ground level -- limits the incentive to migrate to FOSS (and as the above quote illustrates, increases the incentive to migrate to PS). One strategy used by FOSS supporters is to encourage users that are considering an upgrade of their existing proprietary software to migrate to FOSS instead. However, the negative perception of FOSS is currently strongest among the PS labs, which means special efforts would be required before these labs would seriously consider a move to FOSS; members of computer lab staff, who are not familiar with FOSS solutions are unlikely to choose it or consider migration. In addition they are are often not skilled enough to conduct a thorough comparison between different types of software to develop their own conclusions.

There are many reasons why awareness of proprietary software is higher and lab managers (especially those in PS labs) have little knowledge of FOSS alternatives -- and often a more negative perception than the reality justifies. But some of the awareness campaigns that are trying to improve opinions of FOSS are struggling, because their arguments are not appropriate for staff in ground-level computer labs. For example, discussion of the advantages of thin-client solutions is often too complex, and the widespread use of unlicensed software makes the argument of potential cost savings less impactful. (In fact, unlicensed proprietary software is often easier to obtain in Africa than legal copies of FOSS.)

Another reason could be the marketing dominance of proprietary software vendors. Advertisement and marketing is one mechanism that increases awareness about software choices. It builds brand and product recognition and in the best case creates the perception of quality. While it is very difficult to measure the impact of advertisement -- a whole industry of market research organisation struggles to do just that -- it is clear that the majority of software-related advertisements seen by the computer labs and decision-makers who participated in this study promote proprietary software. Of all the labs in this study that mentioned they had seen advertisements, 76% said it was for Microsoft products. Only 38% said it was for "other" companies (the list of "other" answers included not only FOSS, but also smaller and local PS companies). There is no significant brand recognition in the FOSS environment, neither for the generic terms "FOSS" or "open source", nor for any specific vendors.

A few examples to highlight the impressions of PS labs are presented below. The answers specifically represent respondent's opinions, which are not necessarily based on experience, as the questions ask lab managers to also provide their opinion on the software that they are *not* using in their lab.

- Only 5.13% of PS labs believe that good online support is available for FOSS (21% consider good online support equally available for both FOSS and PS). For comparison, more multi-platform labs answer that good online support is available for FOSS (23%) than for PS (15%). 40% of multi-platform labs consider good online support available for both types of software.
- Only 3% of PS labs consider good commercial support available for FOSS (18% said it was equally available for both); and 38% say they do not know.
- No PS labs believe that FOSS is either easy to use or easy to set up.
- While significantly more FOSS and multi-platform respondents (by a ratio of more than four to one) consider FOSS more reliable than PS, PS labs more frequently consider PS to be more reliable.

Initiatives aimed at increasing FOSS awareness

A number of initiatives are underway in Africa to address the lack of awareness and promote the use of FOSS. They are driven by civil society and development aid organisations, and are typically not able to compete with the marketing efforts of large PS companies. As mentioned previously, the *Go Open Source* campaign[86] in South Africa is a notable exception, in which The Shuttleworth Foundation and Meraka have joined forces to raise awareness and lay the foundation for increased FOSS uptake through mass marketing channels including a television show. Increasingly multi-national corporations are adding their support to create business opportunities around FOSS. Other large projects that aim to support an emerging FOSS community on the continent include the Free and Open Source Foundation for Africa (FOSSFA)[87], the AVOIR[88] project based at the University of the Western Cape, and the Linux User Groups [89] across Africa. However, these large-scale efforts are seeking the support of grassroots organisations and local companies that can provide assistance to potential clients looking for guidance and support.

4.8 The effects of policy and political will on software choices

Policy positions that are agreed upon in international, regional, and national institutions are implemented through laws and regulations to form the framework for governing a country. ICT policy-making happens at the international level through processes like the International Telecommunications Union discussions on telecommunications policy, United National Commission for International Trade Law development of model laws, and World Trade Organisation negotiations on trade. And at the regional level it is seen in efforts to address cross-border issues like Internet exchange point negotiations,

86 Go OpenSource, http://www.goopensource.org.
87 FOSSFA, http://www.fossfa.net/tiki-index.php.
88 AVOIR, http://avoir.uwc.ac.za/.
89 For a list of Linux User Groups, see http://www.linux.org/groups/.

technology trade and investment cooperation, or consortium bids to roll out Internet backbone infrastructure. At the national level ICT policies cover a range of issues, from radio and television broadcasting, to the provision of telecommunications services.

Software choices affect and are affected by government policies in a variety of ways. Government decisions and actions -- even if they are not translated into legislation -- have tremendous impact on the way ICT is used throughout society and the economy. Governments also set the economic boundaries that allow businesses to develop, and Government is usually the largest client for ICT products and services. The legal and regulatory frameworks that implement government policies set parameters for ICT use and can either foster or hinder the effective, widespread use of ICT.

The current policy environment in the countries covered by this study

Public-access computer labs (especially school labs) have been at centre stage in policy discussions related to software choices in South Africa and Namibia, as proprietary vendors have made large-scale donations and local FOSS advocates urged their governments to implement their favourite solutions. Many commentators believe that the debate plays out at the lab level. While many of the labs that participated in this study indicated their awareness about government policies and how they may be affected by them, few described any concrete ways that they had participated in public dialogue or engaged with the government on the issue. Descriptions of the policy environments in Namibia, South Africa and Uganda are provided in the section "Overview of the countries studied", above.

Different ways that policy affects software choices

Public-access computer labs frequently encounter obstacles that directly or indirectly relate to the country's legal and regulatory framework and influence the way they use ICT and make software choices. So far no African government has created a law that specifically deals with the choice of software, although the South African Government has accepted a FOSS policy and several other governments are considering similar action and have started drafting strategy papers on the topic.[90] Many African governments do not have a defined position and the ICT departments in different government ministries often have different objectives. In Namibia, for example, the Ministry of Higher Education began migrating some of its users to OpenOffice. At the same time the Office of the Prime Minister, tasked with overseeing the Government's ICT strategy, was not considering FOSS solutions as an alternative for desktop computers. The following examples highlight the impact of different aspects of policy on the choice of software in public-access computer labs:

- **Copyright legislation.** Copyright legislation restricts the use of non-licensed proprietary software, if enforced. So far few African countries have stringent copyright laws, and some of those that do have them, do not enforce them effectively. Hence, the impact of copyright legislation on the widespread use

90 More information on specific policy options and the situation in African countries can be found in bridges.org's FOSS Policy Toolkit. The toolkit can be downloaded at http://www.bridges.org/foss.

of unlicensed software in Africa has been limited so far, but recent developments of African countries joining international intellectual property rights agreements indicate that this is likely to change.

- **Education policies, guidelines and curricula.** In many African countries, education policies are set by government departments at the national or provincial level, and the curriculum for computer studies (and other computer-related subjects) sometimes mention specific software applications. For example, lessons on the use of spreadsheets are based on Microsoft Excel in Namibia. Teachers with sufficient ICT skills are able to teach the curriculum with other types of spreadsheet software as well; however, not all teachers have these the skills to convert application-specific instructions to a more generic level. Some teachers interviewed as part of this study indicated that the fact that the Government-recommended curriculum mentions a specific software application is usually perceived as an endorsement, and teachers are likely to ask for the particular application in order to teach the course.
- **Laws affecting the use of the Internet.** New technologies, such as wireless, enable cheaper and faster last-mile connections to the Internet. They provide opportunities to expand access to those that could previously not afford it or did not have access to the required infrastructure. However, policy restrictions and regulations on these technologies limit their positive impact. Regulation of VSAT (Very Small Aperture Terminal -- a type of satellite dish that is suitable for Internet connectivity in Africa) is mentioned by computer labs in Uganda[91] and restrictions of wireless fidelity or WiFi technologies[92] exist in South Africa. Many respondents in South Africa think of Internet connectivity as a policy issue. Most labs attribute the high cost to lack of competition and an overly-regulated telecommunications environment.
- **Duties, Value Added Tax (VAT), and other taxes on imported computer equipment and software.** Together with transport costs and less-competitive markets, tariffs and duties can make computers in Africa more expensive than in Europe or North America. There are no import tariffs on the majority of ICT equipment in Namibia, South Africa or Uganda. However the details vary, and at least VAT is usually added to imported products.

Political will and public support

Governments can play a key role as engines for socio-economic development. Most African leaders are convinced that ICT will help their nations solve economic and social problems, and they are ready to drive the necessary changes. Government has the unique ability to lead the way and it is critical they do so effectively and bolster public confidence in the path they take.

---

91 Current regulation in Uganda awards only few Internet service providers with International Data Gateway licenses, which are required to provide Internet via VSAT. In addition only the two telecommunications companies and one ISP (Afsat) have a license to install VSAT at a client site. These regulatory barriers reduce competition and contribute to high prices. Email from James W. Lunghabo, 25 October 2004

92 The use of WiFi in South Africa is legal only if it does not cross municipal boundaries, such as streets. But it is exactly the ability to cross these boundaries, and do so without the need to create expensive infrastructure, that could most significantly increase access to the Internet. Neighbours could share their connection, schools could allow the community to use the Internet during the evening hours, etc.

The results of this study show that there is public support for government action to increase access and use of ICT. Among staff members in the labs that participated in this study tremendous enthusiasm for the use of ICT was detected -- and the interviews in schools brought out many examples where learners would wait outside the computer room for hours and use their break times for a chance to sit on the computers.

> “I see (ICT) as a ladder to progress. And this is how the community see it as well.”
>
> -- Lab Manager, South Africa --

However, some of this enthusiasm is not grounded in understanding of what the computers can do to improve social and economic conditions and not based on local realities. When problems crop up and expected benefits do not materialise, the enthusiasm can quickly give way to frustration.

Policy is not only evident in documentation, but also implementation. The political will (and the ability) to implement policy are often more important than the drafting of comprehensive strategy documents. Many public-access computer labs feel their governments are not doing enough to support them. Especially labs in schools and other public institutions expect the government to at least contribute to equipment and training costs and lower telecommunication fees for use of the Internet.

> "Teachers are frustrated with the use of computers because they cannot offer the skill and knowledge of computers to the learners. The Department of Education should complement what other stake holders such as the Department of Communication are doing to help schools."
>
> -- Teacher, South Africa --

4.9 Self-sustainability: a critical factor for computer labs

While the usefulness of ICT for socio-economic development has been demonstrated, the sustainability of public-access computer labs in Africa has proven challenging. Long-term financial sustainability of public-access computer labs is tied to the local economy. Computer labs usually hire employees from within the communities they serve. They procure supplies (such as paper and toner for printers) and services (like technical support) from local providers if possible. And they offer content and services to people or organisations in the community (like access to information, training, or printing), for a fee or sometimes free of charge. At the end of the day, the local economic environment is likely to determine the extent and frequency of computer lab use in the long term, so labs must be designed with local economic conditions in mind.

Unless computer labs and all of their activities are indefinitely and completely subsidised by outside funders with no strings attached (an unlikely scenario), one way or another computer labs need to bring in enough funds to cover their expenses. In theory, they can achieve this by generating income from the content and services they sell, and/or by effectively delivering a social benefit to the communities they serve and reporting back effectively so that funders, governments, and the communities themselves will continue to subsidise the labs' efforts.

In reality, self-sustainable public-access labs in Africa are rare. Funders talk about requiring financial sustainability in ICT projects, but so far this demand is seldom feasible in Africa. Some argue that community computer labs must simply be run like a business, and if they do not have a viable business plan they should not be funded in the first place. However, others recall the role of government and donors to support efforts for the overall good of society and the economy that would not otherwise be possible. They highlight that social returns on investment must be included in the calculation for feasibility of computer labs in Africa. Some even contend that the computers and connections provided by public-access labs should be considered public goods and that financial sustainability is not a relevant success criteria.

The bottom line is that labs struggle to cover capital expenses and ongoing costs. Most labs do not prepare a business plan and they lack the information to assess and report their progress. And funders often only support set-up stages: they expect labs to become self-sustaining, but rarely give them sufficient business, fundraising, and managerial tools to actually do so. Together these factors make financial sustainability the most important -- and unresolved -- issue facing computer labs in Africa today, and tomorrow.

It is important that the potential negative economic effects of public-access computer labs are also considered. For example, computer training programmes can be useful, but local job opportunities must be created for those who acquire ICT skills so they do not have to leave their families and communities in search of employment. Failed public-access projects can lead communities to reject future technology projects, where they feel that funds have been drained from the local economy that might have been better used for other things. And in some cases new technologies can replace human labour -- for example by "cutting out the middle-man" -- resulting in lost jobs, which also can create negative attitudes towards technology. These effects must also be carefully considered.

Current funding models for public-access computer labs

Very few computer labs in Africa receive the kind of ongoing financial support that would allow them to disregard the sustainability issue. This is one of the reasons why so many public-access initiatives fail: set-up costs are donor-funded, but ongoing costs are not covered by additional donations or financial support, and lab managers are not equipped to solve the problem on their own. Computer labs are faced with serious challenges to cover their ongoing costs, let alone budget for replacement of equipment that breaks down. The dire economic situation of most of the computer labs in this study raises the question of whether financial sustainability for public-access computer labs is even feasible.

Initial set-up costs

High initial set-up costs constitute a major obstacle for the creation of more public-access computer labs in Africa. Rarely are public-access labs able to pay the full price of the equipment. The majority of the labs in this study (53%) received donations to cover their set-up costs and only 29% of the labs said they "pay the full price" of their equipment. However, 29% seems high when compared to statements made by lab managers during interviews and researchers' observations from lab visits. Researchers' impressions are that no lab in South Africa is entirely self-supported, but all depend on at least partial donations to purchase computers and operate the lab. Similar situations were observed in Uganda and Namibia. The sources of support vary widely, from local and international businesses, to SchoolNets and other NGOs, to government programmes and international development organisations. In fact, many of the labs reported receipt of multiple donations during the life of the lab, usually to enable equipment upgrades.

But donations can be problematic. For example, one lab described how a donation of upgraded equipment caused conflict of interest, because the new equipment was not compatible with the existing computers, and to keep the donor happy, the older, but still functioning computers were put in storage. Those labs that purchase their own equipment employ various strategies to cover the costs. In one extreme example, a school principal took out a bank loan to purchase computers and software; he has since left, and the school is still paying off the loan. The volunteer working at the school was undecided about whether this is the kind of forge-ahead spirit that is needed to get computers into the hands of students, or if this kind of approach does more harm than good.

Q14: How were the initial set-up costs (those incurred getting the lab up and running) covered?	**All labs N=117**
Donations	**53.85%**
Subsidies	4.27%
Government grant	5.13%
Paid full price	29.06%
Do not know	13.68%

Table 18: Field study -- Initial set-up costs

Ongoing costs

The situation is fundamentally different for ongoing costs, which 74% of the labs that participated in the study cover themselves. However, this indicates an absence of donations rather than the labs' ability to financially sustain their operations. An appropriate reading would be that 74% of the labs are required to cover ongoing costs themselves because they do not have access to donations, government grants or subsidies. Equipment donations (when the lab is set up or to add/upgrade computers later) that come without any financial assistance for operations have led to dozens of decrepit computer labs across Southern and East Africa.

15. How are the ongoing costs (day to day costs) for the computer lab covered?	**All labs N=120**
Donations	21.67%
Subsidies	4.17%
Government grant	4.17%
Paid full price	74.17%
Do not know	4.17%

Table 19: Field study -- Ongoing costs

Revenue generation

The computer labs in this study are well aware of the financial pressure they are under, and most are investigating ways to generate revenue. But there is little evidence of breakthrough solutions that will fundamentally change the economic context for public-access labs in Africa, at least in the short term. Computer labs are economic operations and should use proper planning, accounting and business practices. Yet very few of the labs studied are run on the basis of a written business plan. The majority of labs have not even conducted a needs analysis to investigate what services they could provide for a fee. The computer labs in Uganda tend to spend more time considering their business strategies and developing basic planning tools; however, so far this has not made them significantly more successful than their counterparts in Namibia or South Africa in terms of becoming financially sustainable.

> "No [the lab is not run on the basis of a business plan], but we do have a set of objectives. The vision is to ensure local youth are skilled enough to be competitive in society and are not scared to use technology."
>
> -- Lab Manager, South Africa --

Even without a business plan, a number of the labs follow broad strategies to generate revenue. But exact information on how much income labs generate could not be collected as part of this study; some labs were reluctant to provide financial data, others simply do not have the data because they do not track their expenditure and income. Given the financial hardships faced by the labs and the difficulties they have in covering costs of equipment and operations, it can be concluded that even in labs that charge for services, the income is not sufficient to achieve sustainability.

> "[Our business plan is] to train teachers in the community and to train administration staff from churches in the community to train ordinary people from the community for no charges to begin with, then we start charging when we have trained enough teachers"
>
> -- Lab Manager, South Africa --

37% of the labs in this study offer services to their users for a fee (34% of FOSS labs, 41% of multi-platform labs, and 52% of PS labs). The slightly lower number for FOSS labs is probably related to the age of the labs, where most FOSS labs were recently set up and many indicated that they may start charging for services in the future. The FOSS and multi-platform labs tend to offer infrastructure services like web browsing and email access (following from the fact that more FOSS labs have Internet connectivity), and PS labs usually offer more value-added services like CV writing (likely linked to the fact that the staff in more PS labs are more experienced computer users with the skills to offer these kinds of services).

In many African countries it is common for public-access labs to offer telephone services alongside the computers, however, only very few of the computer labs in this study also offer telephone service (probably because many of the labs studied here were located in schools). Training is another service commonly offered in public-access labs. 53% of the labs in this study currently offer free training (and 28% charge for training), and many labs mentioned their intention of providing fee-based training services in future. Some of the school labs charge students for optional computer courses, use of the Internet, and in some cases even for access to and basic use of the lab. A few of the labs have developed unique services to put on offer. Some provide opportunities for users to send mobile phone-based text messages, and a few labs transcribe voice memos or offer computer repair services on the side. One school lab in Namibia covered its full set-up costs by partnering with a local training provider who provided the computers in exchange for free use of the facilities for commercial classes during the evenings. Another runs monthly fundraising events, such as jumble sales, to cover its ongoing costs.

5 OBSERVATIONS AND RECOMMENDATIONS

This section outlines key observations gleaned from the study, and makes recommendations for the way forward. It recalls key findings; considers the software-related obstacles to optimising public access to computers and Internet in Africa; highlights key lessons learned during the field study; and presents a set of recommendations targeted to ground-level lab staff and policy-level decision-makers. Two topics -- the lack of ground-level involvement in software choices, and open standards -- are covered in more detail, as they require additional background information to be fully understood in the context of this report.

5.1 Key observations of the study

- **Both free/open source software and proprietary software can be used to offer technology solutions appropriate for African public-access computer labs**. Good FOSS and PS applications exist to meet the minimum requirements of most public-access labs in Africa, including a word processor, spreadsheet, web browser and email client. The overall usability of the latest versions of these common applications is very similar for both FOSS and PS software, although set-up and configuration of FOSS applications are still more challenging for non-skilled users. Reliable solutions (meaning that computers work more or less reliably and do not crash extensively) can be provided with both FOSS and PS. Document exchange and compatibility problems still exist, but are decreasing because of better conversion tools.

- **The thin-client model provides a reliable, cost-effective and popular solution for public-access computer labs in Africa.** Thin-client networks are already the most popular model deployed in FOSS labs in Africa, and interest is growing. Reliability and stability of thin-client computer labs are at least on par with other alternatives, such as desktop computers running FOSS or PS. The most common thin-client solution in African public-access labs is based on FOSS (the Linux Terminal Server Project) in combination with old (or refurbished) hardware. Alternative solutions exist, but have not found widespread application in the environment this study focuses on.

- **Software license costs for proprietary software are significant in principle, but in practice they are not borne by many of the public-access computer labs in Africa.** While the cost of software licenses in Africa constitutes a more significant portion of the Total Cost of Ownership than in developed countries (especially when compared against GDP), two factors limit the actual expenditure for software licenses: widespread use of unlicensed or "pirated" software (this may change as many African governments step up efforts to combat software piracy), and the availability of donationed software.

- **At ground level in Africa, the potential for cost savings gained from the use of FOSS is reduced by many factors.** For many ground-level computer labs, the use of FOSS might not result in cost savings, at least in the short term. This is directly affected by a number of factors: the costs associated with downloading software on slow and expensive Internet connections; the more widespread availability of PS skills; and a competitive environment for PS technical support and other services (which helps keep associated costs down) that further reduces the potential cost savings from using FOSS in Africa, when compared to PS.

- **General ICT skills levels -- especially for installation and maintenance of software -- are low overall, and experience with proprietary software is more pervasive.** While there are many initiatives that are aimed at increasing FOSS-specific skills, significant results on a large scale are yet to be seen. Widespread experience with PS means that it is easier to get informal support from friends or neighbours, and commercial support for PS is more widely available.

- **Training courses for proprietary software applications are more widely available than for free/open source software applications.** Training courses are an important way for lab staff and users to gain experience with software. However, PS training is more widely available, mostly driven by demand of local businesses.

- **The fact that FOSS makes source code available and encourages modifications is not exploited by the vast majority of public-access lab staff or users in Africa because they lack the necessary skills. However, it does offers an opportunity for local service providers to create customised applications.** Overall, lab staff and users lack the skills to read and modify source code in order to adjust FOSS applications so they better address specific local needs. However, some examples are emerging where service providers (such as local software developers or organisations supporting the use of ICT in schools) are taking advantage of the availability of FOSS building blocks to develop relevant applications, especially in niche markets that might not attract attention from proprietary software companies.

- **The availability and quality of local technical support in Africa (for both free/open source and proprietary software) is reasonably high overall, although FOSS support tends to rely on free services.** Although there is a certain level of discrepancy between researchers' impressions and the data collected from questionnaires, the study found that technical support for both FOSS and PS computer labs is more readily available and of a higher quality than originally expected. The main difference observed is that FOSS technical support is most often provided by volunteers or free of charge by NGOs, and PS support is generally provided by a company for a fee.

- **While there are a number of projects underway to translate software into African languages, these localised versions are not yet widely used in public-access labs and there is some disagreement about the value of local language software.** The main productivity application suites (FOSS and PS) are being translated into African languages aiming to enable access to ICT to those previously excluded. However few public-access labs reported the use of local versions and the lack of local language content (for example, on the Internet) remains unsolved. In addition, some respondents consider the standard user interfaces a great way to learn English at the same time as learning to use a computer, and they feel that localisation of software is not important.

- **Most FOSS labs in Africa are set up and supported by a small group of enthusiastic implementing organisations, so the success of FOSS labs relies heavily on their efforts.** The majority of FOSS labs are implemented by a small group of organisations with a mission to drive the uptake of FOSS in Africa. FOSS is not a mass phenomenon in Africa, and skills are still relatively concentrated. The success of FOSS labs depends more heavily on the quality of the work these few implementing organisations do, compared to PS where more organisations offer support and implement solutions.

- **General trends with regard to economic situation, age of the lab, and staff experience can be identified among labs that use the same type of software.** There are correlations between the type of software used and the economic situation of the lab that can be identified as overall trends. The FOSS labs generally were set up most recently (and many are still enjoying good start-up support), they have the lowest economic resources and the least-experienced staff, and they therefore usually depend on outsiders for help (again, volunteers and NGOs). Multi-platform labs generally have more highly-skilled staff, are more involved in software choices, and show greater awareness of the issues. PS labs are somewhere in the middle: they usually have greater financial resources (and many purchase their equipment rather than rely on donations), but their labs tend to be older and few labs have the resources to continually upgrade the computers.

5.2 Key challenges and lessons learned in public access computer labs in Africa

- **Financial constraints are the key obstacle for African public-access labs, regardless of whether they are using free/open source or proprietary software.** The overall lack of funds -- to procure appropriate hardware, train local staff, identify and use suitable software applications -- is limiting the labs' ability to use ICT for socio-economic development. Especially in the education sector, the shortage of funds has resulted in computer labs that are not able to support effective learning. Applications are missing and the learner-to-computer ratio is too high. Where up to 15 children have to share one computer, the development of computer skills is limited and the computers have virtually no impact on the quality of learning.

- **With only rare exceptions, the current models for public-access computer labs in Africa are not self-sustainable, regardless of whether they are using free/open source or proprietary software. In the absence of profits, sustainability plans that are linked with the effective delivery of social services could make public-access labs worth subsidising over the long term.** Many donor-funded public-access labs are expected to be self-sustainable after an initial set-up phase and donations rarely cover ongoing costs. However, based on impressions from the field study, sustainability is likely to remain an elusive goal for a considerable number of the labs; even those that are charging for services are at best able to cover ongoing cost, but they are unlikely to be in a position to budget for replacement of outdated hardware. Continued government and/or donor subsidies are likely to be needed; linking public-access labs with the more effective delivery of social services like education, healthcare, and e-government offers one solution for justifying ongoing subsidisation.

- **Overall, local lab staff members lack the skills and expertise to help users and communities realise the potential benefits of computers for their lives.** Even when computers and connections are working, their benefit to local users are often not obvious. Local staff play an important role in working with users to identify useful applications and integrating computers into daily lives. However, many local staff members lack the necessary expertise (and training) and confidence. In the worst cases this even leads to locked-up computer rooms and underused equipment out of fear that something might break.
- **Most African public-access computer labs lack a full understanding of their users' needs. This also reduces the labs' opportunities to charge for services.** Very few of the respondents in this study had a deep understanding of their users' needs or how they could be addressed through different types of software. This also has economic implications; the computer labs often do not know what specific services their clients want and are willing to pay for, and they lack the ability to develop and market new services to generate income.
- **While the limited number of locally-relevant applications is an issue in some African public-access labs, the overall lack of awareness among lab staff about software alternatives is a more significant factor limiting the availability of locally-relevant applications.** Many African public-access labs lack useful and suitable software applications appropriate to the communities they serve because lab staff do not know what software applications exist, how to evaluate which is the most appropriate, or how to procure them. They also often lack the skills to install and use them. Overall lab staff lack expertise, access to information regarding new products, and a community of peers with whom to exchange opinions and experience. Especially in the case of FOSS, many applications exist that could be useful in public-access computer labs, but are simply not well known enough -- and there is no service that advertises new software applications specifically for public-access labs. FOSS product information is almost solely available online and FOSS sales channels that actively promote the best new FOSS applications are only emerging slowly and not targeted to cash-strapped public-access computer labs.
- **Confidence in and enthusiasm for certain kinds of software are important success factors for public-access labs in the African context, but many are hindered by the effects of low levels of technical expertise among staff and users.** Most people in Africa are familiar with the most popular proprietary software applications, and this familiarity together with the market dominance of certain applications instils confidence in their quality, regardless of how well they actually perform in the field. When unfamiliar software is used, it is common for technical problems that are unrelated to the software -- such as problems due to poor configuration or other external factors -- to be blamed on the new software. In African public-access computer labs, the circumstances are primed for criticism of FOSS: external factors commonly lead to technical failures, and FOSS is often unfamiliar to lab staff and users with low levels of technical expertise, who lack the ability to differentiate the real source of technical problems.

- **In addition to cost and lack of awareness, the absence of effective local channels for obtaining software make procurement difficult for many public-access labs in Africa.** In addition to cost factors and the lack of awareness about software options (covered above), the absence of effective local channels for obtaining software (including resellers and distributors) is a significant factor hampering the procurement of software by public-access labs in Africa. For FOSS, few companies or organisations fill the typical roles of a sales channel for software, to raise awareness of existing and new applications and distribute software to local labs. But procurement channels for PS are also deficient in Africa, especially in smaller markets (beyond South Africa) and rural areas.

- **The availability of effective technical support (to ensure that the computers remain operational) has a significant impact on the success of public-access labs, but support for proprietary software (usually commercial) is more readily available in Africa than support for free/open source software (usually non-commercial).** Technical support is one of the key issues at ground level that determines the success or failure of public-access computer labs in Africa. The lack of technical support is a major obstacle to technology use in many African countries, and skills transfer is a crucial element of any development project involving software. While the availability and quality of technical support for both PS and FOSS is relatively good across Africa, it is not uniform. Especially in rural areas, public-access labs often have a limited choice of support providers, and many FOSS labs rely on volunteers and donor-funded support. The impact of poor technical support is tremendous, especially since in many labs even basic problems cannot be addressed by local staff. Where support is not timely, efficient and affordable the effective use of software is not sustainable in the long-term.

- **Local staff and users are rarely involved in the software choices made in African public-access labs.** Most local staff are largely not party to the decision-making for the technology used in their labs, including the software choice. In the case of FOSS labs, this is mainly due to lack of skills (lab staff members often do not have the expertise to make informed software choices), and in the case of all labs a lack of local funds is the decisive factor (the labs rely on donations and are rarely in a situation to influence the technology that is being donated). Decisions are taken at the policy level, by donors and other organisations that are providing the technology. See the box below for more details.

Lack of ground-level involvement in software choices

In most of the computer labs in this study, local staff members -- including the individuals responsible for operating and managing the labs -- had only limited involvement in the decisions about the technology and software used in the labs. Most of these decisions are taken by outsiders such as funders and donors, and usually in a top-down approach that relegates labs to mere recipients of technology. This is largely due to the limited resources -- mainly capital and expertise -- of computer labs. Lack of capital creates dependence: donations are the most common way for labs to obtain ICT and usually the organisations that provide capital, equipment and expertise for free decide what software is used. Donors include local private sector companies making their old equipment available, and governments or international development agencies that are rolling out computer lab projects. In addition, when local staff members are not fully aware of the different options that exist, they can not participate in discussions of the costs and benefits of these options.

The lack of involvement is most strongly observed in FOSS labs: 66% of lab managers that participated in this study said they had "no influence" on the software that was chosen for their lab, and more than 39% of the FOSS labs stated that they are not aware that a choice between software applications exists.

The fact that outsiders are making technology decisions on behalf of local public-access computer labs is not necessarily a negative point, as long as the decision-makers understand and respect local needs and conduct sufficient training to enable true empowerment of the recipients. However in many cases, outside decisions are not implemented in a way that builds local expertise. Computers are "dumped" on the local staff, who lack the skills to resolve even minor technical problems let alone use the technology effectively. The computer lab staff interviewed in this study offered many anecdotes about misguided donations and implementations. For example, new computer equipment was donated and installed instead of a much more cost-effective upgrade of the existing technology. Working computers were removed and replaced by different equipment, because the implementing organisation did not support the old type of equipment. Multiple donations over the years resulted in different types of computers and software systems being used simultaneously, causing higher need for support and leading to problems of incompatibility.

Understanding the issues around this situation is relevant for two reasons. First, because better understanding of who is involved in hardware and software choices can help target awareness campaigns more directly to the actual decision-makers and support them with the information they need to make sound judgements. Often awareness campaigns are directed at ground-level staff, who might not have any control over the actual decisions. Second, this should be taken as a starting point to determine how local staff can be effectively brought into the decision-making that affects them. Building skills is the first step, because skilled local staff can work with donors to at least provide feedback on the local context that can inform the decision-making. This level of participation in decision-making can increase the feeling of empowerment and ownership, two substantial factors that make computer labs succeed or fail.

5.3 Recommendations to public-access computer labs in Africa

Computer labs in Africa are faced with very significant obstacles to providing public access to ICT in the communities they serve. The software choice is one important decision, which is intrinsically linked to cost, technical support and many other factors that determine the lab's success in the short and long term. Pragmatic choices are necessary to overcome these obstacles and ensure that people are able to use ICT effectively to improve their lives, especially those users whose only opportunity to access ICT is through public labs.

The following recommendations are directed to the staff and managers of public-access computer labs in Africa, with the intention of focusing their attention on the most critical areas related to the effective choice and implementation of software in their labs.

- **Analyse user needs by listening to your stakeholders and educating them about the possibilities that computers offer.** Analysing user needs in the public-access context in Africa is not always simple, because often users lack a basic understanding of what computers can do for them and they are unable to articulate their needs. Therefore, the analysis of user needs must consist of one-part listening to potential users and one-part educating them about what is possible with ICT.

- **To the greatest extent possible, provide software applications that are the most relevant in the local context.** Basic office productivity applications are available as both FOSS and PS, but special applications, including games or multi-media production software, are more frequently available as PS only and for PS operating systems such as Microsoft Windows. Some technology service providers that focus on public-access computer labs offer educational or other software applications as part of their package. Test and compare the software before making a decision and keep in mind that the computers must provide a real and tangible benefit to the users.

- **Ensure that the lab has access to appropriate, affordable, and effective technical support to keep the computers operational.** Support is a crucial factor for the success of a public-access computer lab and labs must ensure that affordable, effective and timely technical support is available that can respond to their needs. The decision on how much and what type of technical support is appropriate (for example, telephone support might be sufficient in some labs) depends on the expertise and workload of the local staff members. Often organisations that provide support can also advise on software choices; however, they can be expected to promote the type of software that they can support, so talking to more than one technical support company, if possible, is advised.

- **Do not let persuasive sales pitches or strong advocacy make the choice of software seem like it has to be an either-or scenario: consider a mix of FOSS and PS if that best suits lab needs.** There is no reason why public-access computer labs need to choose between one or the other type of software -- both can work well together. For example, one popular solution consists of Windows-based desktops together with a Linux-based server. In addition, many FOSS applications run on PS operating systems including Microsoft Windows, and vice-versa for Linux-based desktop computers.

- **Keep it simple by using standard software applications, and make the most of staff experience.** Many interesting and useful things can be done with a small set of standard software applications and basic hardware. Most office workers around the world use a word processor, spreadsheet, web browser and email software, all of which are available as FOSS and PS for the common types of operating systems. Especially labs that have few local skills and want to focus on increasing basic computer literacy in their communities initially should think twice about whether they really need powerful hardware and the latest software applications. The usefulness of lab computers is more likely to be determined by a well-trained and motivated lab manager than by the age and type of the software used. Using a small number of basic applications can reduce the time required to keep the computers running and more time can be spent working with users to ensure that the technology is useful for their daily lives.

- **Be smart about proprietary software donations, and consider both short- and long-term costs and benefits.** To weigh the value and effect of a potential proprietary software donation over the short and long term, public-access computer labs should consider a number of factors, including: the likelihood that they will need to upgrade the software and the timeframe for an upgrade; the investment that will need to be made to implement the software (training, installation costs, etc.); how much of the initial investment would be lost if the lab later choose migration to FOSS instead of an upgrade (for example, if they could not afford the upgrade later on); and the costs of a later migration to FOSS compared to the costs of an upgrade to a newer version of the proprietary software (including the possibility that the upgrade is covered by the donation).

- **Analyse all costs associated with owning and using software over the short and long term, and within the context of the particular lab and the local economic environment.** The Total Cost of Ownership model is useful to assess the costs for the most important aspects of operating a computer lab. As a first step, public-access projects should get quotes from local technology and service providers, investigate costs for training, and ensure that sufficient funds are available to cover the costs. However, these projects should also be mindful that general statements about the cost of FOSS or PS are often not directly applicable to the public-access lab environment in Africa. Cost is affected by many variables, including the labs willingness to use unlicensed software and/or access to software donations. Carefully analysing all cost factors in the actual context of the lab and the local economic environment is crucial to determining which option is really more affordable.

- **When making software choices, choose reliability and stability over functionality.** In the African context, the reliability and stability of the software are often more important than advanced functionality. In an environment where external support is very often prohibitively expensive or simply unavailable locally, anything that minimises the need for technical support and maintenance is crucial.

- **Balance the cost and effort required to upgrade software with the expected benefits to the user. If the lab chooses free/open source software, the incentive to use the latest versions of applications might be higher, due to frequent improvements in usability, functionality and reliability. Regardless of the type of software chosen, always apply the latest patches and updates.** The FOSS development model is built around the idea of releasing applications early (when not all of the problems and errors have been fixed yet) and using feedback received from the early-adopters to improve the software. New and improved versions are frequently released. The increases in usability, functionality and reliability that come with the latest versions of free/open source software are sometimes significant, so labs that choose FOSS should use up-to-date versions to the greatest extent possible. This is less important for PS applications, which are usually tested before they are released so that every version available to the public should be sufficiently stable, and the effects of upgrading on reliability and crashes are often less significant. However, regardless of the type of software used, updates and patches (less significant changes than upgrades, that typically address small errors, bugs and security problems) should always be applied. They can usually be installed without advanced technical expertise.

- **If the lab chooses the Microsoft Windows operating system, adequately protect computers against computer viruses and other malicious software to minimise support costs.** Especially on computers running Windows 95 or Windows 98, the default configuration of the Outlook and Outlook Express email clients represent a considerable security risk and is a popular target of computer viruses.[93] Impressions from the field study suggest that labs using this combination of software more frequently suffer from virus infections, which can lead to data loss and downtime and sometimes require outside technical support. Two simple solutions exist to address this problem: (1) adjusting the configuration of Windows and Outlook to a higher security level than the default, or (2) choosing an alternative email client that offers less opportunity for attack. A number of proprietary and FOSS alternatives (that run on Windows) exist. The effective use of anti-virus software is also a solution.

- **Allow only minimal access to the system configuration to avoid users making changes that could cause computers to stop working.** To avoid computer problems that take up the time of technical staff or require outside technical support, it is crucial that steps are taken to minimise user interference with system configuration settings. When configured appropriately, multi-user operating systems can help avoid this problems by prohibiting user access to system settings. Newer versions of Windows (after Windows 98) and all versions of Linux provide functionality to set different user accounts with adjustable access levels that can be configured to protect the system configuration settings.

93 Windows 95 and 98 are used in 64% of PS and 77% of multi-platform labs in this study. Outlook and Outlook Express are used by 47% of PS labs and in 61% of multi-platform labs in this study.

5.4 Recommendations to decision-makers setting policies that affect public-access computer labs in Africa

There are various decision-makers who lead or influence policies affecting public-access computer labs in Africa, including government officials, private sector actors, influential NGOs, bureaucrats at international institutions, and development aid organisations and donors. They carry the responsibility for long-term vision and planning to ensure effective and sustainable public access to ICT that brings concrete benefits to individuals, as well as society and the economy. Those who make determinations about the software that is rolled out in public-access computer labs are often in the position to drive change at a large scale; however, they are frequently also faced with complicated constraints that affect their choices. Usually their decisions impact more than just a few users in one local computer lab, but often tens or hundreds of computer labs and thousands of users. And their support for certain types of software can also have an impact on the broader social, economic, and political environment.

The following recommendations are directed to those decision-makers who are setting policies that affect public-access computer labs in Africa. These can be used as guidelines during the decision-making process and provide background to the key issues.

- **Address the sustainability problem for public-access computer labs by better equipping them to deliver a social return on investment (improved education, healthcare, economic opportunities, etc.) that is worth subsidising over the long term.** One solution to the sustainability problem encountered in almost all African computer labs would be to accept that public-access projects will not achieve financial sustainability. Then the focus could turn to equipping computer labs to deliver a social return on investment -- that is, the value they bring to education, health, government programmes, and other social focus areas -- which justifies the costs of subsidising their activities over the long term. Regardless of their opinion on sustainability, funders need to be aware and should help address the burden that ongoing costs place on many labs, because just providing set-up support has not succeeded overall.

- **Increase local capacity, and actively involve lab staff and managers in decision-making processes about software and other aspects of public-access labs.** Lab staff and managers are rarely involved in decision-making about software and many other aspects of public-access labs largely due to their overall lack of skills (and finances). The skills issue should be addressed through training to enable local staff members to participate effectively in discussions and decisions; skilled staff that are involved in technology choices will feel truly empowered and take ownership of the computer lab. Since skill levels for FOSS are generally lower than for PS (as is availability of training to build skills), this issue is even more important in initiatives that consider or decide to roll-out FOSS solutions.

- **When making software decisions that involve significant change (such as upgrading operating systems or migrating to FOSS applications), implement effective change management that**

addresses issues of training, process redesign, and reluctance to change in public-access labs. Local lab staff can be reluctant to learn unfamiliar software applications, such as when software is upgraded to new versions or a computer lab is migrated from one type of software to another. Change management processes can address these issues by raising awareness among staff about the reasons for bringing in new software, providing additional training as needed, and allowing time for adjustment.

- **If you choose to promote FOSS solutions across society and the economy, focus on raising awareness about software use and benefits at all levels: decision-makers, community stakeholders, lab staff, and end users.** Effective marketing campaigns for PS products have built brand awareness, and the widespread use of PS solutions provides a popular vote of support. Efforts to migrate existing PS labs to FOSS may need to overcome the negative impression of FOSS. Similarly, raising public awareness and support is crucial for FOSS to make up for the market dominance of PS. Current awareness-raising activities usually focus on the policy level, and thismakes sense, at least for the time being, as most public-access software decisions are taken by funders (local donors or international development organisations) and government officials, especially in the education sector. However, expertise, skills and awareness of the different options at the user level are equally crucial, especially when a long-term view is taken. Getting buy-in from the users is important to avoid resistance against the technology, so training and awareness efforts are necessary in computer labs into which FOSS is introduced, and awareness about the FOSS alternatives to PS products should be increased.

- **Focus on generic computer literacy skills to increase e-literacy broadly.** Most public-access projects include a component that aims to increase e-literacy of the communities served. The best way to ensure that users can effectively operate computers is to equip them with basic generic skills that are not dependent on one specific software application or environment. For example, an e-literate workforce understands the concept of email and can quickly adapt to different email applications to receive and send messages. The focus on generic skills also reduces vendor lock-in, because the need for training required by a migration to a different applications is lower.

- **Conduct high-level surveys to determine user needs and encourage public-access labs to add local information collected from the communities they serve.** Especially where a large number of public-access labs are outfitted with a standard set of software applications, the project should carefully investigate which applications are most suitable in terms of usability and functionality on the broad scale. For the main applications this will not require a choice between FOSS and PS, because similar applications are available in both types of software. The project management should also work with local labs to cater for the different needs that exist in different communities, and support the local lab to procure specific applications to address these needs.

- **Encourage and support the development of locally-relevant applications that run on multiple operating systems.** Useful and appropriate software applications are the key to demonstrating social and economic value of computer labs and can help to ensure their financial sustainability. Providing computers and connections is the first step, but without supporting developments to make the computers more useful, these efforts will be lost. Where large scale public-access projects decide to support software development, they need to choose which type of license the software will have. FOSS may make sense for small local software developers, as many standard building blocks that can be adapted already exist and do not require additional investment. On the other hand, there might be a stronger commercial incentive for local developers to create proprietary software so they can sell licenses and donor funding could help these developers get their projects off the ground. Regardless of the type of license that is chosen for the application, it is important to ensure that the software can be used on multiple operating systems (such as Microsoft Windows and Linux). For example, the success of OpenOffice is largely due to the fact that is not only works on the Linux operating system, but also on Microsoft Windows and Apple OSX.

- **Use public funds to create software that can fill specific local needs and be considered a public good**. Where public funds are used to directly support research and development of software applications for the public-access environment, the results and benefits of this investment should be available to the largest variety of users. This presents a strong incentive to choose a FOSS license. However, this is one of the few areas in this report where differences between open source and free software have significant implications. Some open source licenses allow others to modify the source code and turn the results into a proprietary product (closing the source code). A free software license however requires all software that is based on the initial development to remain free. Large parts of the private sector favour the use of less-restrictive open source licenses, but advocacy groups like the Free Software Foundation argue strongly for the use of free licenses, which they consider the most effective way to ensure that public funds benefit the largest part of society.[94]
- **Assess the need and availability of local language software and consider all available strategies for promoting its development.** Where public-access projects determine that they require localised software interfaces, which do not yet exist -- usually because large parts of the intended user group read/write in a language other than English -- the opportunities and costs of creating the needed applications should be analysed carefully. Both FOSS and PS offer opportunities to create localised software: FOSS presents a useful starting point for a localisation project as anyone can choose to add a new language to an existing application and modify the software. On the other hand PS companies might be willing to dedicate some of their financial resources to localisation, whereas the FOSS groups -- if not entirely volunteer driven -- usually require outside funding.
- **Make use of web-based content (that can be accessed from a variety of different software platforms), if possible, to reduce the impact of the software choice.** With regard to content (educational content, news, and other kinds of information), web-based content is recommended even if the computer lab does not have an Internet connection (where the "pages" of

94 Free Software Foundation -- http://www.fsf.org/

the World Wide Web could be burned onto a CD-ROM and sent by postal mail). The advantage of web-based content is that it adheres to "open standards" and can be viewed using many different web browsers (FOSS and PS options exist), so it does not require one specific application. With regard to educational content, choosing web-based solutions separates the decision of which educational content is most appropriate from the decision of what type of software is most appropriate. For large-scale projects such as those run by the Department of Education, this enables central procurement of educational content that will be useful in different types of labs, and does not exclude labs on the basis of their software choices.

- **Make smart choices about accepting software (and related) donations and keep public-private partnerships transparent.** Especially where donations are directed at governments, policy decision-makers must take additional factors into account when weighing the real value and benefits of a large-scale donation. It is critical that donations and the public-private sector relationships they foster are transparent, and that any uncompetitive procurement practices of the government are critiqued and reviewed. In addition, governments can maximise the value of donations by carefully analysing what is offered, clearly defining the terms of the agreement, and insisting on delivery of the promised goods and services. These agreements should also aim to include small and medium-sized local companies to enable skills transfer and support the local economic environment.

- **Take all steps possible to ensure that sufficient technical support is available for large projects to be rolled out.** Technical support considerations for large-scale public-access projects need to look at the availability and cost of local support providers for different types of computer labs. Often flexible and creative approaches to solving the support issue pay off. In some cases university students can provide technical support, or in the case of schools, international volunteers can be attracted who have sufficient skills to oversee the lab even if they are not computer specialists. Otherwise the existing commercial support options in the region of the project need to be assessed. In addition, large-scale projects can work with educational institutions to encourage development of certain types of skills that are needed. With regard to the current support environment for FOSS labs, the long-term sustainability of largely volunteer-based support need to be assessed. With regard to proprietary software support, local producers and sellers and resellers of proprietary software can be approached for sponsorship of support.

- **Analyse the Total Cost of Ownership in the specific local environment, rather than relying on general statements of the cost advantages of one solution.** Many arguments are brought forward that make the case for one software solution being cheaper than the other, but there are no general conclusions that are applicable across Africa. Support costs vary widely, donations might be available in one environment but not in another, and the cost of training to ensure sufficient skills levels can be significant. All of these factors and more need to be taken into account to determine the expected cost for a public-access project. The only approach that will fully inform software choices is a detailed analysis of the specific environment of each computer lab (or group of labs).

- **Consider the pros and cons of diversity in the computing environment, to make balanced choices between increased costs on the one hand and increased value and opportunity on the other.** Diversity in the computing environment is a cost factor, because uniform installations are more cost-effective to set up and support. But diversity in the social and cultural sense -- language, learning patters, and approaches to integrate technology into daily lives -- has tremendous inherent value and also offers economic opportunities. The needs of the many small and diverse user groups found in Africa are less likely to be addressed by the large software companies, offering opportunities for local developers to compete.
- **Increase Internet access to address the lack of applications and enable the use of online technical support.** Increasing Internet access can have significant impact on the success of public-access labs. It can bring down support costs if local staff make use of online support options. It can also provide communication services (email, chat, VoIP, etc.) that the labs can offer to their users. Especially for FOSS labs, fast and reliable Internet also lowers the barrier to obtaining software that has to be downloaded. At the policy level, mechanisms such as subsidised access charges (sometimes referred to as "e-rates") or public-private partnerships for sponsorship of telecommunication services should be investigated.
- **If you choose to promote FOSS solutions across society and the economy, ensure that alternative software procurement and delivery models are available where cheap and fast Internet connectivity is not an option.** The distribution costs for FOSS in Africa are considerably higher than in other parts of the world, where the majority of FOSS applications are downloaded via cheap high-speed Internet access. If provision of cheap and fast Internet connectivity is not possible, projects interested in deploying FOSS should consider the establishment of alternative software distribution models (by postal mail for example) and include the costs in budget calculations.
- **Fully understand the implications of software choices so you can leverage bargaining power to bring donations and other benefits to public-access labs.** The emergence of FOSS as an alternative to the existing dominant proprietary software applications, together with the potential loss of business from the public sector, have led many international companies to lower prices and offer substantial discounts for software to public-access labs. Especially government projects -- and in particular those in the education sector -- enjoy significantly more bargaining power now. Where these projects have publicly announced their interest in a FOSS solution, proprietary software companies have often responded with highly competitive alternative offers. Public-access projects should take this into account in their negotiations.
- **Partner with the local computer labs to evaluate existing policies with regard to software choices and ICT use for public access.** There are many examples where a nation's leadership has embraced ICT and is ready to promote a legal and regulatory environment that will enable its widespread use, including through public-access projects. To support these governments the development aid industry generates a tremendous volume of reports, advice, and analyses intended to inform the development of laws and regulations. However, these recommendations do not always show sufficient

understanding of local needs and conditions. Collaboration with public-access computer labs provides an opportunity to give leaders a realistic appreciation for what ICT can -- and cannot -- do for the country. The lessons learned in these labs can help help frame appropriate legislation, balance the needs and views of the relevant constituencies, and ensure the legal and regulatory framework is implemented effectively at the ground level.

- **Support open standards (such as open document formats) to minimise the impact of a rapidly changing technology environment.** All software choices should aim to support open standards as much as possible. Open standards avoid vendor lock-in and increase competition (which often brings down prices). More specifically, the support for open standards is relevant with regard to file formats for document exchange. Applications that save information in closed file formats (or file formats that can not easily be read by other applications) are more likely to cause problems when information and documents are exchanged with other labs.

Open standards

FOSS is often cited as an "enabler for African solutions to African problems", although many question whether FOSS is really more suited to be an African solution than locally-produced proprietary software is. However, proponents on both sides of the debate seem to agree on the relevance of open standards for helping to make ICT more accessible to the broadest range of people in Africa and beyond.

In the world of software, "standards" are protocols and definitions that describe the exchange of data or documents between different software applications. "Open standards" are developed in a non-exclusive and participatory manner, are publicly available, and can be freely used by commercial and non-commercial software developers in free/open source software as well as proprietary software.[95] The European Commission's Interchange of Data between Administrations Unit's definition of Open Standards requires the "intellectual property -- i.e. patents possibly present -- of (parts of) the standard to be made irrevocably available on a royalty-free basis" and calls for "no constraints on the re-use of the standard" to be imposed.[96] Proprietary standards are privately owned and controlled and must be licensed from the owner of the standard.

Open standards are not synonymous with FOSS and not all FOSS adheres to open standards. However the standards used by FOSS applications are always publicly available because the source code can be reviewed. The real benefit of open standards stems from a participatory mechanism of standard definition and adherence to the standard by developers of competing software applications. Because anyone is free to write a software application that implements the standard, open standards greatly lower migration barriers and increase competition. For example, if all companies that develop spreadsheet

95 There are many slightly different definitions and requirements of open standards. See the following three examples (all last accessed March 2005) for more information:
http://www.techweb.com/encyclopedia/defineterm.jhtml?term=open+standards
http://encyclopedia.laborlawtalk.com/index.php
http://www.csrstds.com/openstds.html
http://perens.com/OpenStandards/Definition.html

96 The definition is part of the European Interoperability Framework. For more information, see http://europa.eu.int/ida/servlets/Doc?id=18063, last accessed November 2004.

programs agreed to use an open standard for spreadsheet files, then the files could be shared among any of the spreadsheet programmes. That would enable users to choose from a variety of software applications -- regardless of free/open source or proprietary -- without being locked into one vendor or product, because all applications would use the same underlying format.

The processes in which open standards are developed allow participation by a variety of stakeholders and they are conducted in relatively transparent manner. Although both FOSS and proprietary software developers have declared their support for open standards, each have also been criticised for attempting to "hijack" open standards processes. The claim is that some companies extend the protocols outside of the defined standard, which leads to incompatibility and defies the purpose of open standards. An example for this is the "browser war" between Netscape Navigator and Microsoft Internet Explorer in the late 1990s. At that time, both companies tried to establish their respective products as the market leader. In order to do so, they extended the existing set of HTML tags for web pages by adding design features, which would only work in their respective browser and which displayed poorly or not at all in the competitor's browser. The goal was to attract developers of websites to take advantage of the extra features, which would then force the visitors of these sites to use a specific browser. The negative effects of these tactics can still be felt today, as complex websites often have to develop different versions for the different browsers of their users to make sure everyone has access to the same functionality.

However extending standards is not always done out of purely malign reasons. An innovative company that wants to offer new functionality to its customers could be held back by a slow moving standard development process, and open standard setting organisations provide mechanisms for extension of the standard. In other cases developers might prefer not to extend an open standard, because they believe an innovative new functionality could constitute a competitive advantage and they do not want to allow others to easily replicate the innovation.

6 CONCLUDING REMARKS

In the context of social and economic development efforts, it is difficult to resist the appeal of concepts such as information sharing, collaboration, and freedom of knowledge, which are foundations of the FOSS movement. The core FOSS community itself is an awesome phenomenon in a world largely based on the pursuit of individual gain: a self-organised group of individuals, many of whom contribute their time and effort because they enjoy the process of software development and choose the communal benefit their contribution can have over financial compensation. Perhaps the most important and surprising aspect of the FOSS movement is that is has demonstrated a fundamentally different approach to creation and innovation, and this approach has more or less been able to integrate with the existing economic models it seems so at odds with.

But in Africa, it is important to remember that FOSS is just one software option in the larger ICT toolbox. And ICT is merely a means to an end that is most valuable when it supports broad social and economic goals, such as facilitating healthcare delivery, making small businesses more competitive, or improving education and government services. In this context, the discussion of software choices necessarily moves from philosophical underpinnings to pragmatic concerns. Some would argue that failing to give FOSS a higher degree of reverence for its built-in social qualities is a dangerous omission, but in light of more pressing issues facing Africa -- like the HIV/AIDS crisis, high rates of illiteracy, and debilitating poverty -- a dogmatic focus on FOSS philosophy is considered cynical by others. After all, what is the value of computers in communities that lack clean water and struggle to provide basic education to their children, unless these computers can be integrated into strategies that will ultimately improve the living conditions of the poor?

Since the research team first set out to investigate the realities of software use in public-access labs in 2003, much has shifted in the big picture of software choices for Africa. FOSS is now enjoying widespread interest among government officials, and large international computer and software companies are contributing to its development and pushing for its adoption across the continent. These efforts mostly focus on the practical elements of the choice between FOSS and PS in the corporate and government sectors: cost, compatibility issues, migration strategies, and availability of training and technical support. This in itself is a significant development, but it has left the realities in public-access operations -- fundamentally different from the corporate or government sectors -- largely untouched. And it is disappointing that these developments in the area of FOSS have focused on the easy wins in those sectors that already use ICT effectively, rather than trying to bring the benefits of ICT to larger parts of society. Some proprietary software companies are also working to solve important problems in public computer labs, especially enabling their software to run on less powerful and cheaper hardware, and reducing or waiving licensing costs. But meanwhile, specific software applications (whether FOSS or proprietary) that could make computers more useful to local communities -- such as those putting ICT to work to improve healthcare and education and designed with cultural factors in mind -- are still missing.

In Africa, FOSS works well in large, carefully-designed and well-implemented projects. These projects usually include training for end users and managers, provide hardware (or help with procurement), and ensure that the right kind of support is available when it is needed. And these projects report cost calculations that are lower than alternatives based around proprietary software. However, it is the quality of planning, management and implementation -- rather than the quality of the technology solution -- that makes these projects successful. The use of FOSS without the safety net of a large project -- in small, independent and remote computer labs where technical skills are often low -- has proven more difficult. Especially low-cost solutions on older hardware pose configuration and administration challenges that most of the lab managers visited during this study are not able to overcome. And if the computers do not work reliably, enthusiasm for the new technology quickly fades, and the benefits of ICT in poor communities does not materialise. One notable exception are the thin-client systems found in many FOSS labs; when well configured and installed, they provide good value for money as they run on less powerful and cheaper hardware.

Once the hardware is in place, the development of useful applications becomes a priority for all supporters of ICT in Africa (regardless of which kind of software they use), because computers can only make a difference if the software on them is relevant. So what is preventing the widespread availability and use of more locally-relevant applications in African public-access labs? Some specific applications relevant to the African environment -- targeted to African needs, dealing with critical issues like health, designed with cultural factors in mind, and available in local languages -- simply do not exist. However, useful and versatile applications like text editors and spreadsheets are already available, and it is the lack of capacity and skills that prevents users from integrating them into their daily lives.

A number of key obstacles characterise the software choice for public-access labs in Africa: labs lack awareness of software choices and often do not know what applications and content are available; staff do not have the necessary skills (and training) or time to investigate and compare options; labs cannot afford to buy proprietary applications or download FOSS applications from the Internet; and often local procurement channels are not available to provide information and deliver software. And especially in rural areas, poor infrastructure (including unreliable electricity and costly Internet access) is a significant barrier to deploying ICT.

By comparison, more of the required pieces are already in place for PS, raising the stakes for FOSS projects. More people have experience using PS applications, sales channels are more developed, and there is public trust in the applications that are most common. In addition, the economic muscle and the effective marketing and political strategies of proprietary software vendors, who have much to lose from FOSS encroaching on their markets, present significant roadblocks for increased FOSS uptake.

Should PS vendors pay closer attention to the practical obstacles facing public-access labs, and build on the commitment to deliver on social and development goals, their value proposition for Africa remains high. However the momentum in Africa is currently in favour of FOSS, whose supporters are riding on a growing wave of enthusiasm based on successes in other developing countries in South

America and Asia. FOSS supporters in Africa have an opportunity to capitalise on this enthusiasm, but need to overcome serious obstacles to translate the hype surrounding FOSS into tangible benefits. Above all they need to establish communities of software developers who have the means and interest to develop and support locally-relevant applications. Otherwise the promise that FOSS will turn African nations from consumers of technology to producers will not become a reality.