The Use of Open Source Technology in Elections

The Use of Open Source Technology in Elections

Contributors
Michael Clouser
Robert Krimmer
Henrik Nore
Carsten Schürmann
Peter Wolf

Editors
Peter Wolf
Nadia Handal Zander

International IDEA Resources on Electoral Processes

International IDEA
Strömsborg
SE-103 34 Stockholm
Sweden
Tel: +46 8 698 37 00, fax: +46 8 20 24 22
E-mail: info@idea.int, website: www.idea.int

Cover design by: Kristina Schollin-Borg, Sweden
Graphic design by: Eva Alkmar, Sweden
Cover photo: Pixtal/TT
Printed in Sweden
ISBN: 978-91-87729-68-3

Preface

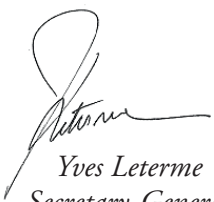
The field of technology in elections continues to develop at a rapid pace and there is a growing need for more efficient and transparent systems, not least due to the increase in the number of electoral processes implemented in response to popular movements and demands for change, as can be seen for example in West Asia and North Africa. Many countries are going through crucial transitions, and elections are being held as part of these transitions. When held in volatile situations, and in direct response to emerging or urgent political needs, such elections are often implemented within relatively short timelines and with little notice or time for preparation.

The introduction of technology to elections is not new, but what is relatively new is the debate on the role of open source technology (OST). One of the reasons for the increased debate on the issue is the demand for full transparency throughout the different phases of the electoral process. It is believed that the use of OST in implementing elections will increase transparency and therefore increase the level of trust in the results of those elections.

Another expectation from OST is that it will help reduce the overall cost of managing elections through increased efficiencies in administration, and a reduction in the total cost of ownership (TCO) of a system. Election officials are therefore interested in the financial benefits of such systems.

There are few examples of the use of OST in elections, however, as well as a great deal of scepticism, misconceptions and even mistrust in systems that use OST. Misconceptions surrounding OST include the assumption that making source code publicly accessible presents a serious threat to the security of the electronic systems that use those codes, and therefore to the overall electoral process.

The aim of this Guide is to enhance the understanding of OST among key electoral stakeholders, including electoral management bodies, governments and decision-making bodies, vendors and, of course, civil society, including the voters. We hope that this Guide will be helpful in engaging these stakeholders in a more active way in the debate on OST and on whether such technology can indeed be instrumental in enhancing the transparency and efficiency of their electoral process.



Yves Leterme
Secretary-General
International IDEA

Acknowledgements

We would like to thank all those who contributed to the content of this Guide, in particular the authors who patiently sat through an intensive four-day Book Sprint in order to generate its content. We thank Michael Clouser, OSET Foundation; Robert Krimmer, Professor of e-Governance, Tallinn University of Technology; Henrik Nore, Ministry of Local Government and Regional Development, Norway; Carsten Schürmann, Associate Professor/DemTech, IT University of Copenhagen; and Peter Wolf, Elections and Technology, International IDEA.

Thanks also to those who attended the workshop which informed the Guide: Prasana K. Dash, Director General, Election Commission of India; Gregory Miller, Co-Executive Director and Chief Development Officer, OSET Foundation; Annette Fath-Lihic, Senior Programme Manager for Electoral Processes, International IDEA; and Abdurashid Solijonov, Database and Statistics, International IDEA.

Particular thanks and appreciation are extended to the vendors who participated in the workshop in person and online, and shared their valuable points of view: Manuel Kripp, Business Development Logistics, IVU Traffic Technologies; Jordi Puiggali, SVP Research and Development; Dallas Newby, Director of Project Management (Operations), Dominion Voting; and Lori J. Steele, Chairman and Chief Executive Officer, Everyone Counts.

Last but not least, we would like to thank Faith Bosworth, facilitator of the Book Sprint workshop, for keeping the contributors on track; Andrew Mash and Eve Johansson, for their meticulous language editing of the Guide; International IDEA colleagues Rosinah Ismail-Clarke and Nuno Durao, for their logistical support; and Publications Manager Nadia Handal Zander for seeing the Guide through the editing and production stages.

Acronyms

API	application programming interface
BSD	Berkeley Software Development (an open source licence)
CI	continuous improvement
GNU	GNU Not Unix
GPL	General Public Licence (an open source licence)
GRECO	Group of States against Corruption
ECL	Educational Community Licence (an open source licence)
EMB	electoral management body
EOM	election observation mission
EVM	electronic voting machine
ICT	information and communications technologies
IP	intellectual property
IT	information technology
MIT	Massachusetts Institute of Technology
NGO	non-governmental organization
OSET	Open Source Election Technology (Foundation)
OSI	Open Source Initiative
OST	open source technology
TCO	total cost of ownership

Table of Contents

Contents

CHAPTER 1

What Is Open Source Technology?

Introduction

What makes software 'open source' is the licence under which it is released. The Open Source Initiative (OSI) provides the industry's standard definition: 'Open source software is software that can be freely used, changed, and shared (in modified or unmodified form) by anyone. Open source software is made by many people, and distributed under licences that comply with the Open Source Definition'. For the purpose of this Guide, a more 'subjective' working definition has been adopted, which we have called the 'open source state of mind'. By this, we mean a willingness to create transparency in software design.

International standards and commitments on elections include the promotion of transparency and accountability as important pillars. Open source technology (OST) would appear to support these goals and is receiving increased attention. However, software vendors tend not to embrace this open source state of mind, in part because of the stigma that is attached to OST linked to the fact that it is 'free', and that the protection of intellectual property (IP) is not the main focus and therefore cannot be sufficiently guarded.

The open source concept dates back to the early days of information technology (IT). It is based on the assumption that by opening up a software system and exposing its source code, that is, the individual instructions that tell the computer how to perform computations, the software system can achieve higher levels of quality with less effort. There are several reasons for this. These include the fact that, by permitting others to access the code, software developers can potentially draw on a larger group of people who can experiment with, test and expose errors in it. There might be programmers among these people who can fix bugs and problems, or even extend the code. In addition, the code may also be shared with other projects.

Examples include the Linux and FreeBSD operating systems, which profited greatly from the user community who designed and debugged device drivers for third-party hardware. In addition, if the software is distributed free of charge, it will almost certainly gain some kind of popularity and sometimes a larger user base. Examples include Open Office, Apache, Gimp and Sugar CRM, as well as the MySQL database.

Overview

The use of IT in elections is increasing around the world, and the number of misconceptions and myths surrounding its use is growing in parallel. These include concerns that IT provides little transparency and is difficult to understand, given the perceived complexity of many technologies. Even back in the 1920s, David T. Zuckerman (1925) was discussing this issue: 'Presumably the voting machine does require an act of faith on the part of the voter in a mechanical contrivance whose workings he cannot see. No more so, however, than is required in the case of the automobile in which he drives up to the polls. Indeed, he has even less assurance that the paper ballot [will be] counted as he intended; [or] see his vote recorded, nor does anyone else'.

In nation states that follow democratic principles, elections are a celebration of democracy and considered the backbone of a democratic processes that should ideally be trusted by everyone—not just a select few. Thus, it is important, in particular when it comes to using IT in elections, that all stakeholders—voters, political parties, election officials and so on—achieve a sufficient level of trust in the voting technology used to allow them to accept the results. While electoral management bodies (EMBs) can acquire this trust to a certain degree through discussions with vendors, with the help of experts or by using statistical methods, it is much more difficult for others, such as political parties or individual candidates, civil society and the media, to do so if they do not have access to any of the software artefacts used during the election. This problem becomes even more pronounced, however, in democracies that are used to traditional paper ballots, and to relying on a collective public effort to count them.[1] In future, it will be important for these EMBs to gain a direct and immediate understanding of the workings of the voting technology instead of trusting it by proxy.

Elections are unique in comparison with other IT projects in that they come with fixed deadlines, especially in terms of the nomination periods or the delivery of final results, which are usually stipulated by law and cannot be extended. This makes the introduction of technology to and its use in the electoral process particularly difficult. It is therefore important that electoral IT projects allow sufficient time, and that planning commences early. Elections are also an expression of local culture and national identity, which makes it

difficult to reuse software in more than one context without significant customization. There are also other external challenges for electoral IT projects, such as that elections are recurring events that take place every two to five years, by which time the underlying technology (hardware and software) might have been updated to newer versions or rendered obsolete. In either case, this will lead to changes to the election software that must be executed, tested and audited.

All these arguments have prompted a discussion of whether OST is a good model for election administration. Aspirations for the use of OST are that it will increase the transparency, security, accountability, accessibility and sustainability of the electoral process, in combination with lower overall costs. For some, OST even provides a better balance of power between electoral stakeholders as, in contrast to closed alternatives, it enables civil society to understand and assess the technology used. The idea of OST is a general one, however, and there are many different ways in which OST can be introduced.

About this Guide

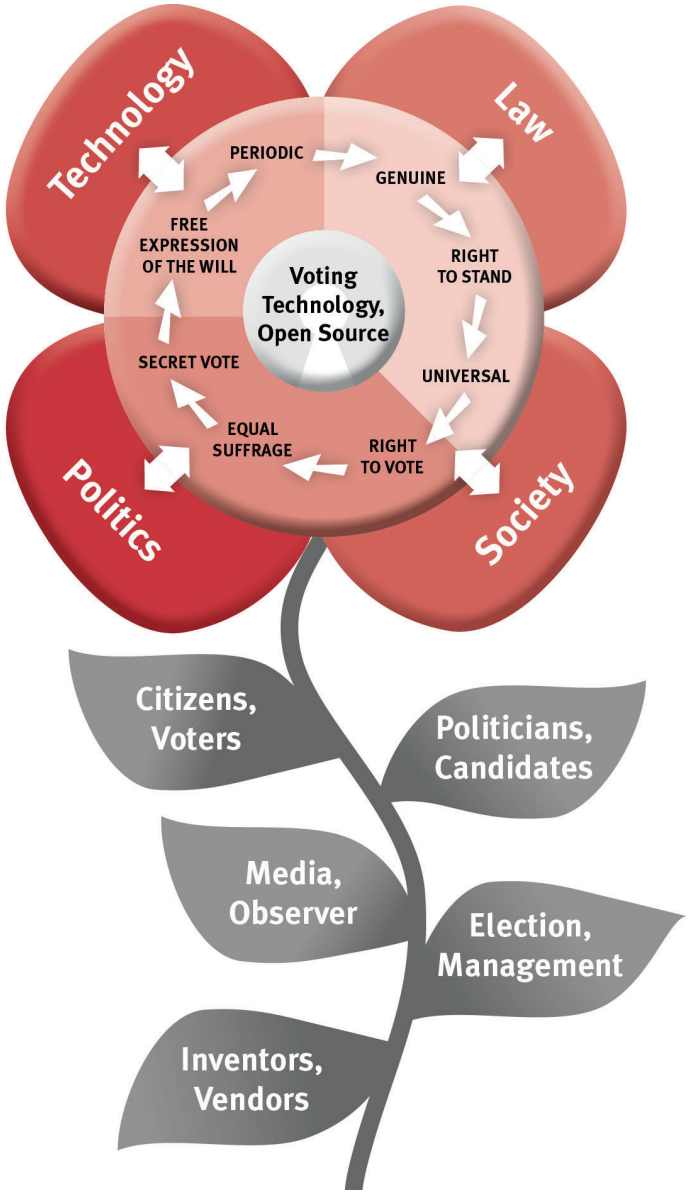
This Guide provides an introduction to the term 'open source' and its licences, and the role OST plays or could play in the administration of elections. It presents an overview of alternative business models for OST, and discusses the issues to be considered while assessing the feasibility of using OST in any given election. The Guide touches on the role that OST plays or could play in the different stages of the election administration process. It examines why OST has emerged as an alternative to traditional closed source software and endeavours to summarize the advantages of using OST, as well as its limitations. Finally, it discusses how OST can be implemented and sustained in the long run.

The Guide does not advocate a specific approach to the use of open or closed source technologies, but rather presents different options to key stakeholders in order to encourage them to take a more proactive role in making or initiating the appropriate changes. Nor does it go into the detail of the different components of and options for the use of OST.

Context

Elections do not take place in isolation or in a vacuum. They are social processes that are shaped and defined by the context in which they take place, making each election unique. In order to be able to understand the commonalities and differences between different elections, it is useful to get to know the ecosystem in which the use of OST takes place. This consists of its environment, the OST's technological properties and the electoral process, often known as the electoral cycle, in which it is used (Krimmer 2012).

Figure 1. The OST ecosystem



Source: Krimmer 2012

Elections technology, including OST, is at the centre of an ecosystem (see Figure 1) that shapes and is shaped by its surroundings, in particular the electoral process which consists of three main phases: the pre-electoral, the electoral and the post-electoral (see Figure 2). It is worth noting that the preparations for each of the different electoral phases may take longer if technology is being used, especially in the case of the pre-electoral phase where preparations for the use of technology should be expected to take longer than those for traditional paper-based voting. The other main components of the technology and OST ecosystem are: (1) the legal dimension, which includes the constitution and electoral law that govern the conduct of an election and in most cases shape the main properties of how the election technology can be applied; (2) the political dimension, which includes discussions around using this and/or similar technologies; (3) the social dimension, which includes the impact on and influence of society; and (4) the technological dimension, which consists of the available technical infrastructure and the penetration of various supporting technologies such as digital identification schemes, including smart cards. The context is further influenced by the different stakeholders, which include EMBs, citizens and voters, politicians—whether they are candidates or not—the media, election observers, and OST inventors and vendors.

Showing the electoral process in the form of an electoral cycle is commonly accepted practice (see Figure 2). It portrays the conduct of elections as a continuous, repetitive process. Within this cycle, the electoral process consists of several phases, in all of which technology can play a key role. These include the design and drafting of legislation, the recruitment and training of electoral staff, electoral planning, voter registration, the registration of political parties, the nomination of parties and candidates, the electoral campaign, polling, counting, the tabulation of results, the declaration of results, the resolution of electoral disputes, reporting, auditing and archiving. After one electoral cycle ends, preparations for the next are already expected to begin.

Open source licences

There are many kinds of open source licences, which can be amended and adapted to the specific needs of each EMB or electoral body. Box 1 provides a summary of the general requirements that make software open source as defined by the OSI, the de facto standardization organization.

Box 1. The requirements of the Open Source Definition

1. Free Redistribution

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

2. Source Code

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

3. Derived Works

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

4. Integrity of The Author's Source Code

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

5. No Discrimination Against Persons or Groups

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

6. No Discrimination Against Fields of Endeavor

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

7. Distribution of License

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

8. License Must Not Be Specific to a Product

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

9. License Must Not Restrict Other Software

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

10. License Must Be Technology-Neutral

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

Source: <http://www.opensource.org/osd>

It is important to note that these ten requirements alone do not define what open source is or how intellectual property in software is treated. These kinds of questions are settled by the individual licence, which is required to respect these requirements. There are numerous open source licences under which software can be released. The most popular is the *GNU General Public Licence* (GPL), which transitively applies to all derived software systems. The *Berkeley Software Development (BSD) Licence*, on the other hand, places only minimal restrictions on the redistribution of the software covered.

A fundamental benefit of all open source licences is the transparency created by the requirement to have the source code publicly available. Having source code on public view can greatly ease the task of electoral stakeholders when examining voting technology and determining its quality, features and benefits. It even allows the comparison of a deployed system with the published reference software, providing confidence that the system running is based on the software that was previously examined.

In spite of the high level of transparency, however, open source licences are flexible enough to allow a developer or vendor to retain proprietary rights and the ability to charge software licence fees, even after publishing their source codes.

Sample licences[2]

The five licences set out below are each examples of open source licences approved by the OSI. They are all open source but possess different requirements regarding the distribution of code.

1. *General Public Licence version 3 (GPLv3)*
   Anyone may copy, distribute and modify software released under GPLv3 as long as he or she tracks and dates changes in the source files and keeps any modifications under GPLv3. Applications can be distributed commercially using a GPL library, but these must be open source under GPLv3.

2. *Berkeley Software Development Licence—BSD Licence Version 2 (BSDv2)*
   The BSDv2 licence allows anyone to freely use, modify and distribute the software as long as the BSD copyright notice is included in the source code. This licence is considered the most liberal open source licence.

3. *Educational Community Licence—ECL Version 2 (ECLv2.0)*
   The ECL licence is an 'open/open' licence, which makes the source code available for unrestricted development by commercial or non-commercial entities, and does not impose the use of a particular licence on derivative works. It was developed specifically by the educational community for its own needs. It is based on the Apache licence.

4. *Apache Licence, Version 2.0*
   Issued by the Apache Foundation, it allows for the free use, modification and distribution of its software without regard to royalties.

5. *Massachusetts Institute of Technology (MIT) Licence*
   The MIT Licence allows free use and modification of the source code for both proprietary and non-proprietary uses. In proprietary uses, it allows proprietary properties to be kept even though the MIT source code has been used. It is also GPL-compatible. The popular Ruby on Rails uses the MIT Licence.

Further restrictions on open source licences

Approved open source licences might be considered prohibitively open by some election technology providers. If this is the case, it is recommended that the licence be further adapted, even if this means that the resulting licence is no longer accepted by the OSI or similar organizations. The key property that should be retained from an open source licence is that of transparency. Box 2 provides an example of a licence that was adopted, following customization, by the Norwegian Ministry of Local Government and Modernisation (the EMB) and a vendor.

Box 2. An extract from an adapted OST licence

'Source Code, High Level Architecture Documentation and Common Criteria Documentation Copyright © 2013 and ownership belongs to The Norwegian Ministry of Local Government and Regional Development and [the Vendor] ('Licensor').

The Norwegian Ministry of Local Government and Regional Development has the right to use, modify (whether by itself or by the use of contractors) and copy the software for the sole purposes of performing Norwegian Public Sector Elections, including to install and run the code on the necessary number of locations centrally and in any number of counties and municipalities, and to allow access to the solution from anywhere in the world by persons who have the right to participate in Norwegian national or local elections. This also applies to elections to the Longyearbyen Community Council at Svalbard and any possible future public elections in Norway arranged by the Election Authorities.

Patents, relevant to the software, are licensed by [the Vendor] to the Norwegian Ministry of Local Government and Regional Development for the purposes set out above.

[the Vendor] (or whom it appoints) has the right, inside and outside of Norway, to use, copy, modify and enhance the materials, as well as a right of licensing and transfer, internally and externally, either by itself or with the assistance of a third party, as part of the further development and customization of its own standard solutions or delivered together with its own standard solutions.

The Norwegian Ministry of Local Government and Regional Development and [the Vendor] hereby grant to you (any third party) the right to copy, modify, inspect, compile, debug and run the software for the sole purpose of testing, reviewing or evaluating the code or the system solely for non-commercial purposes. Any other use of the source code (or parts of it) for any other purpose (including but not limited to any commercial purposes) by any third party is subject to [the Vendor's] prior written approval.'

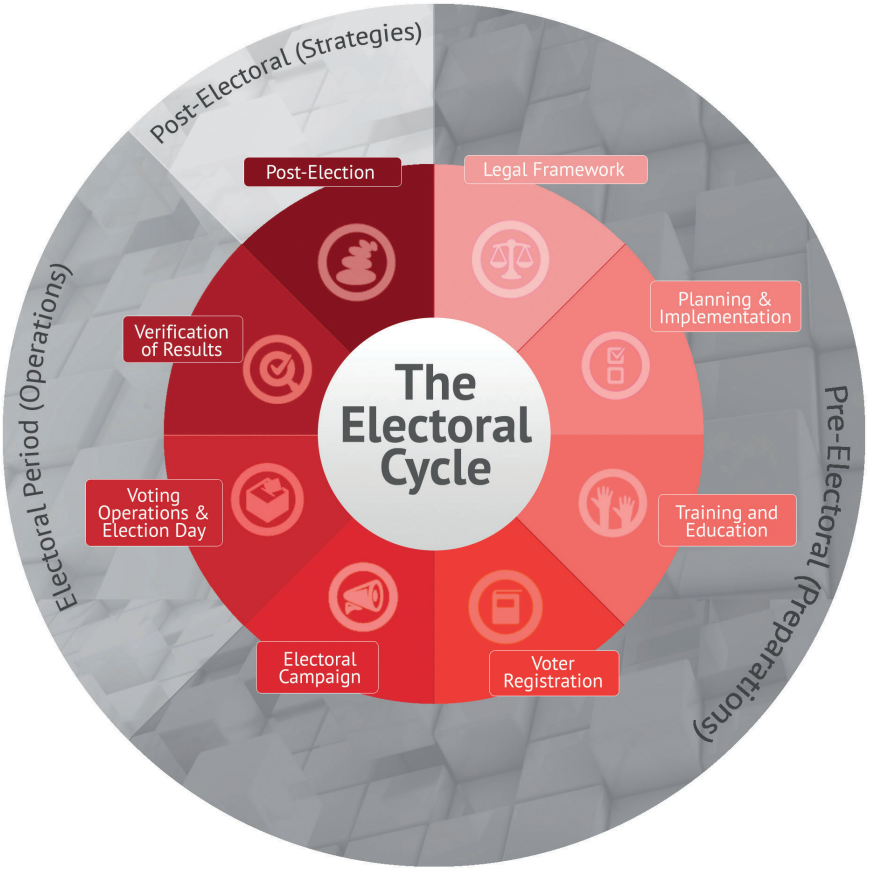
Source: <http://www.regjeringen.no/nb/dep/kmd/prosjekter/e-valg-2011-prosjektet/kildekode/tilgang-pa-kildekode.html?id=646007>

The role of OST in elections

The organization of elections can be described as an electoral cycle (see Figure 2). Every step in the organization of elections can be supported by information and communication technologies (ICT). In the 21st century, it seems hard to imagine that some countries still organize elections with little or no technology, for example, for the aggregation and tabulation of election results. Recent years have seen massive investment in voter registration processes, where many problems related to the integrity of elections have their roots.

In its cornerstone document, *Recommendation Rec* (2004) *11 on Legal, Technical and Operational Standards for e-voting* (CoE 2004), the Council of Europe defines voting technology as ICT used only in the casting of the vote. By contrast, a broader understanding of voting technologies is used for the purposes of this Guide, meaning the use of ICT to support the conduct of election administration throughout the electoral cycle. This section identifies the applications used throughout the entire electoral process, as well as the specialist applications used in the pre-voting, voting and post-voting phases, and describes how OST can be used in the implementation of these applications.

Figure 2. The electoral cycle



Source: International IDEA (2014)

A. Throughout all phases of the electoral cycle

1. *Web portals voting services portals*
   The creation of web portals to provide relevant information to the various electoral stakeholders is quite common and presents no particular challenges. The use of open source software for the operation of web servers and content management systems is widespread and a variety of OST standard solutions are readily available.

2. *Election administration systems*
   Such systems are usually designed to support all steps of election administration and there is currently no common form or definition of their functionality. Ideally, the system should be made the cornerstone of the architecture of the voting technology, as it provides the necessary interfaces for and data on the various other applications used. Because many functionalities are highly specific to the respective administrative process defined in the electoral cycle, the software is usually highly customized and hardly any OST has been developed for it, other than as a modular basis for customization. There is, therefore, a big need for standardized OST modules that allow customization. This system has considerable potential as it can automate and support many of the repetitive and time-consuming administrative tasks involved in the organization of an election.

3. *Reporting on campaign financing*
   This topic has gained increased attention since the Group of States against Corruption (GRECO) started to tackle the issue of campaign finance and the abuse of administrative resources. In connection with the need to provide more transparency in campaign spending, electronic tools have been found to be an easy solution for implementing the GRECO recommendations, in particular due to the short timelines usually imposed for the publication of campaign finance reports. Open source software lends itself to this purpose, as the issue is mainly publishing reports and providing access to data. While some customization may be needed, this might easily be implemented through minor customization of existing OST.

4. *Management of multiple voting channels, in particular registration for postal voting/mail ballots*
   The increasing mobility of citizens puts EMBs and lawmakers under pressure to increase the level of service available to citizens by offering them multiple opportunities to participate in an election, such as through advance voting in polling stations, postal voting or Internet voting. Some mechanisms require citizens to register to ensure the integrity of the election, that is, to prevent any citizen from being able to vote more than once. Like party or candidate registration, this is a natural module for the

election administration system and could be built using a standard workflow management system. OST could potentially provide a solution as this seems to be a similar task across different elections. However, no standardized solution currently exists.

5. *Inventory tracking and management*
   Conducting an election usually involves tracking a lot of materials, such as ballot boxes, polling booths and ballot papers, and requires even more management when it comes to storing and using voting technology. This could require stand-alone software or part of the election administration system. It could be easily standardized on the basis of existing OST inventory management.

6. *Records management systems*
   Election management software could include a module that allows the creation, editing, publication, storage and overall management of the EMB's records.

7. *Data analysis (data warehouse)*
   The election administration system can include a data warehouse that allows the display and analysis of election data generated throughout the election. Such a warehouse would facilitate strategic decision-making on current and future elections on the basis of past experience.

B. The pre-election phase, from calling an election to voting

1. *Voter registration, review of electoral registers*
   This is one of the applications most sought after by EMBs, as many electoral processes fail to provide accountable and transparent voter registration procedures. While it might be relatively easy to put together a database for a single election, the main problems arise in connection to data integrity, sustainability and the maintenance of the voter data. This phase of an election usually incurs relatively high costs, and in some cases[3] technology-based projects are implemented involving considerable investment in both hardware and software. Often identification technologies, such as biometric readers for fingerprint or facial recognition, are implemented together with the development of centralized databases for voter records. In many cases, these processes could benefit from a wider use of OST. However, this is seldom achieved. OST has the potential to enhance sustainability, avoid vendor lock-in, and maintain flexibility, particularly if a local community and local ownership can be built to service and maintain the relevant customized software. Voter registration systems can also integrate features that include voters as part of the quality control of voter registries, by allowing them to review relevant data in

such registries. Although this would require the data protection implications to be addressed, it could lead to better quality control of the data and greater public confidence in the accuracy of the electoral register.

2. *Digital ballot paper delivery*
   To address the issue of increased voter mobility, EMBs, in particular in the United States, have started to offer a digital service channel to deliver ballot papers for self-printing to voters living abroad. These must be returned by mail to the EMBs. Because this voting channel could have substantial security implications, there is a greater need for accountability and transparency. It could therefore benefit greatly from the use of OST.

3. *Registration of election observers*
   International and national election observation are important elements in increasing trust in the electoral process. Election observers are often required to receive official accreditation from the EMB, and a module in the election administration system could help to manage this process. While OST does not seem to be a requirement for this, it would certainly not hurt to customize existing open source workflow management systems for this purpose.

4. *Signature collection (European Citizen Initiative)*
   The signature collection tool is similar to the one developed at the European Union level by the European Citizen Initiative, which provides opportunities for citizens/voters to support a certain cause through a public website. OST could be useful in this and similar initiatives, but it has not been adopted. One reason could be fear of the transformative effects such a platform could have on the electoral process. Nonetheless, OST seems to be an obvious choice for this application as transparency and community engagement are essential to the success of any initiative in this area.

5. *Party/candidate registration and ballot paper generation*
   Candidate and party registration or nomination in an election can also be supported by OST, through either a public channel for self-registration or a platform managed by the EMB. A publicly accessible interface is usually one of the less likely options to be adopted, however, often because of fears that it could lead to the registration of too many new candidates. Nonetheless, if used correctly, this tool could provide important data for future steps in the process, such as automatic ballot paper generation, or the provision of open data to portals that use party and candidate lists for diverse purposes, such as candidate information systems. While the workflow is likely to be unique to each type of electoral process, mainly due to the relevant legal and administrative procedures, it could provide an easily standardized OST module.

C. The election phase

1. *Electronic poll books*
   There has been an increase in the use of information systems in polling stations, in particular connected to the more widespread use of tablet computers. Such information systems can be used for easier access to voter records and to verify voter eligibility. Particularly where electoral registers and eligibility checks are contentious, such systems can be put under particular scrutiny, which makes the use of OST even more useful.

2. *Electronic voting machines (EVMs)*
   Machines to support the electronic casting of votes have been under discussion since the middle of the 19th century, and have been proposed by many reputable inventors. With the increased debate around the use of such machines comes increased criticism of their programming and heightened calls for the use of OST. However, apart from sharing source code, very few existing EVMs use OST licences.

3. *Ballot paper scanners for polling station and central counting*
   The use of optical scanners for digital recognition of voters' intentions on ballot papers has received increased interest following challenges related to the use of EVMs. While there has been some attempt to develop EMB-owned software, for example, in Latvia, only closed source software currently exists on the market. The need for transparency and accountability means that such software would greatly benefit from becoming an OST.

4. *Ballot paper marking devices*
   Often in combination with ballot paper scanners, EMBs will want to assist voters with disabilities to mark their ballot papers independently through the use of a computer. Such voters would complete the ballot paper electronically and print it out. Apart from making it possible for people with support needs to vote without assistance, this would also reduce the number of unintentionally spoilt ballot papers. The use of OST could be of great value in this context.

5. *Internet voting systems*
   Internet voting systems are designed to verify the identity and eligibility of a voter, allow him or her to register and cast a ballot, and tabulate the results. While many proposals for algorithms exist in the academic sphere, including several OST-based implementations such as Helios, few elections have been conducted using OST-based Internet voting systems—an interesting fact considering the need for transparency and accountability, which speaks clearly in favour of using OST in such a system.

6. *Voter turnout reporting*
   The voter turnout at a given polling station or voting channel can be easily calculated and reported to a centralized location and/or election administration system using a digital poll book system or an alternative voter eligibility check system. Such a system could be realized using open source data entry and visualization applications.

D. The post-election phase

1. *Result transmission, aggregation and tabulation software*
   Apart from the EMB's website, one of the most common areas where ICT is used in an electoral process is the transmission, aggregation and tabulation of election results. Such systems facilitate the transmission of results from polling stations or counting centres, and their aggregation and tabulation in regional and/or national tabulation centres. Depending on the workflow, OST could be used, among other things, to allow for parallel tabulations and provide the necessary amount of transparency and accountability for the public, who would want to be certain that their votes are tabulated and counted accurately.

2. *Calculation of mandates*
   In an extension of the results transmission system, the system can also calculate the assigned mandates on the basis of the election results, in particular when more complicated mandate assignment calculations are needed. The principles of OST can be particularly helpful in this context for establishing trust in the system, as interested members of the public can verify the correct implementation of the algorithm and ensure that appropriate checks and balances are in place.

3. *Systems for publishing election results*
   Systems for the publication of election results are usually standard extensions of the EMB's website which automatically display preliminary and final results, and corresponding information graphics. Standard OST can help implement such a functionality.

4. *Information for successful candidates*
   A module of the election administration system can automatically inform the winning candidates by printing letters, sending e-mails and performing other standard or routine steps. Standard OST workflow systems would be suitable for such uses.

The role of OST in election technology

In sum, OST can essentially be used in all forms of voting technology to support the implementation of an election. While in some areas it is already widely used, for example for the provision of information on EMB websites, there is a strong need to further develop specialized software, in particular for election administration systems and EVMs, Internet voting systems and ballot paper scanners, which require high levels of transparency and accountability.

Table 1. Availability and needs assessments for open source technology in election administration

Voting technology	OST readily available	Need for transparency and accountability
Web portal/voting services portal	Yes	Low
Election administration software	No	Medium
Reporting on election campaign financing	Yes	Low
Management of multiple voting channels, in particular registration for postal voting/mail ballots	No	Medium
Inventory tracking and management	Yes	Low
Record management system	Yes	Low
Data analysis (data warehouse)	Yes	Low
Voter registration, review of the electoral register	No	High
Digital ballot paper delivery	No	High
Registration of election observers	Yes	Low
Signature collection	No	Medium
Party/candidate registration and ballot paper generation	No	Medium
Poll books	No	High
Ballot paper scanners for polling station and central counting	No	High
Ballot marking devices	No	High
Electronic voting machines	No	High

Voting technology	OST readily available	Need for transparency and accountability
Internet voting systems	Yes	High
Voter turnout reporting	Yes	Low
Result transmission, aggregation and tabulation	Yes	Medium
Calculation of mandates	Yes	Medium
Results publication system	Yes	Medium
Information for successful candidates	Yes	Low

CHAPTER 2

Why (or Why Not) Use OST?

Introduction

This chapter analyses the strengths, weaknesses and current uses of OST in elections. It examines the various advantages that an open source approach can have in terms of increased transparency, potential cost savings and opportunities to build local capacity and competence through a culture of openness. This is followed by a presentation of the preliminary results of a recent survey conducted by International IDEA in more than 130 countries on the current adoption of OST in elections, which seem to indicate a lack of awareness of OSTs and a low rate of adoption of such technologies.

The reasons for this seemingly low adoption rate are also discussed. These include a lack of awareness and several misconceptions about the technology, the challenges of a fragmented market that requires highly customized solutions, and existing vendor business models that depend to a large extent on closed source technology.

The strengths of open source technology

Transparency

Based on current discourse, it appears that the central benefit of using OSTs in electoral ICT systems is transparency. OSTs are required to be in the public domain, regardless of the software licence under which the source code may be used. By definition, open source licensing means that anyone, including electoral staff, political parties, civil society and voters, has the right to see the source code of the systems that drive the electoral process. Furthermore, anyone, including vendors, can retain proprietary rights and the ability to

charge software licence fees, even after sharing their proprietary software's source code with the public.

Placing software source code in the public domain can greatly ease the task of an electoral stakeholder when examining a voting system to determine exactly what features have been implemented and how. It also guarantees auditors and civil society access to the source code to evaluate the degree to which it meets their requirements. This helps to reinforce confidence in the electoral process and/or in generating proposals on possible avenues for improvement. Either way, the transparency in voting technology implied by open source licensing fosters trust among the electorate.

This *system validation* function is particularly important for voting system technology, where election officials are bound by state law only to operate voting systems that have passed detailed testing, and been approved or certified for use. Election officials must be able to validate a voting system, that is, to see it in use during an election and determine whether it consists of tested software and configurations. Part of the validation process can also be an assessment of how likely it is that the voting system will be available for the entire course of an election, can keep the vote secret and can compute the correct results.

The transparency derived from open source-licensed systems is a necessary first step towards confidence in the operational behaviour of voting systems. By releasing voting technology software into the public domain, EMBs introduce a shared ownership with the electorate, and this openness generates increased trust in the election. To strengthen this openness further, EMBs might consider deploying techniques that can guarantee that the correct version, that is, the published and audited version of the software stack, is running. They might also consider introducing a measure of verifiability to the electoral process in order to reassure voters, election officials, political parties, election observation missions, and possibly even courts in cases where the vote is contested. Such measures can complement OSTs and include facilities for secure logging or other certificate generation.

Blind trust in voting technology is considered harmful. Calls for more transparent elections technology often include a demand for the source codes of mission-critical elections software to be opened up. However, relying on OST alone cannot make a system fully transparent, as its operational behaviour usually depends on particular versions of external libraries, compilers, run-time and operating systems, and last but not least hardware. The availability of code alone does not guarantee that the shared code will be used during the election. Nonetheless, making the source code available remains an important confidence-building measure that demonstrates the election administration's

willingness to be as transparent as possible and share details about the type and quality of the systems used.

Cost and sustainability

The cost of election technology is often high and, especially in developing democracies, prompts questions about the long-term sustainability of these solutions. In some cases, ICT solutions have already been discontinued as they proved too costly.

Elections management ICT solutions have to be looked at from a total cost of ownership (TCO) perspective, that is, the initial investment plus the ongoing expenses related to licensing fees, continuous improvement (CI), maintenance, testing, and the development and deployment of the system for its entire lifetime. The largest and most obvious savings from OST come from the absence of software licensing fees, which are non-existent in the open source model. Table 2 highlights the cost components of the TCO model.

At this point it is difficult to predict the exact TCO impact of increased OST use in elections administration. There are expectations that the overall costs of election ICT solutions will fall and that it will be easier to sustain such solutions. This change could come about through a transformation in parts of the voting solutions industry to a more service-based business, from increased competition as new vendors find it easier to enter the elections market, as well as through a more efficient joint community pooling of efforts. While the positive impact of the increased use of OST in elections is as yet unproven, and comprehensive TCO assessments have yet to be made, one example of the potential for OST to change the market and make available enterprise-strength systems at a much lower cost than comparable proprietary systems is the development of MySQL, a database management system that is now being used, among other things, to drive websites of any size from small private efforts to large endeavours such as Twitter and Facebook.

Table 2. Components of total cost of ownership

	Cost of acquisition	Operating costs
Software	Initial licence fees, development costs	Licence subscription fees
Hardware	Acquisition of required hardware	Hardware maintenance, upgrades, security, decommissioning
Support	Initial migration, installation, set-up	System maintenance, reconfiguration, support, security, upgrade, replacement
Human resources	Initial staff training	Personnel costs (management, operation), ongoing professional development
Telecommunications	Initial acquisition, set-up	Hosting and network recurring fees, upgrades, security
Facilities	Initial acquisition, adaptation	Ongoing operating costs, renovation, security

Box 3. Norway—preliminary cost implications of adopting OST

According to the Norwegian Ministry of Local Government and Modernisation, the main consideration when moving to OST was transparency—not reducing the costs in the system. Nonetheless, the TCO of the elections management system, including an Internet voting and a ballot paper scanning system, has been reduced by the use of an open source program stack for both development and operations. The upfront investment was about €20 million over three years. Annual maintenance costs have been significantly reduced due to the very low licence cost of open source software. (A support fee is charged on some products but these are also available in a free version.) Maintenance costs have also been reduced due to the lower cost and better availability of skilled developers linked to the use of an OST stack.

Box 4. Cost savings in the government sector through OST

In 2007, Transport For London moved to an open source software stack to administer its transport network. Significant cost savings have been realized since then. Thus far, the TCO saving has been 80%, and the project is expected to save over £20 million over a ten-year period. Part of this cost saving comes from the increase in transparency and auditability, which allows for better system management (Shaikh and Cornford 2011).

Maintainability

The code base of OSTs is in perpetual motion and only comes to rest during code freezes and code releases. This is inherent in the open source software licensing scheme. Anyone can extend the capabilities of a particular set of open source-licensed software, but in many cases the terms of the licence require them to offer their extensions and modifications back to the custodian of the software so that other adopters can choose whether to accept the new versions. Typical updates to open source software include bug fixes, the removal of features or the addition of new functionality, which sometimes triggers changes in the application programming interfaces (APIs). For this reason, OST can be a blessing because over time the quality of the software can be expected to improve. It can also be a curse, however, as changes to the API bring cascading effects for those systems that are built on top of the OST.

OSTs guarantee transparency, which is a good starting point for EMBs in terms of a justification for their adoption. OSTs make continuous change easier to manage by giving operational control to the owner and user of the technology. This, in turn, increases auditability, which further empowers the organization's ability to make election technology transparent. This is especially important for the testing and debugging of voting systems prior to the election process, where mistakes and problems can be costly for election officials, and may affect the public's views on the elections, and their level of trust.

OSTs could empower EMBs around the world to share a broad substrate of common functionality while permitting individual EMBs to tailor the OST system to their own local and cultural needs. On the flip side, due to the changes that can affect an electoral system, it is the responsibility of the EMBs to invest wisely in continuous maintenance to guarantee the stability of their systems. To respond to this maintenance challenge, EMBs need to set up appropriate institutions and define their mandates. Some EMBs choose to build up in-house capacities, while others outsource tasks to the private sector or to public-private partnerships. A more general overview on this is given in Chapter 3 under 'Community development and maintenance'.

Box 5. Maintainability: the example of the Commonwealth of Virginia[4]

The Commonwealth of Virginia has extended the capabilities of its Voter Services Portal to include extension of its online voter registration (OVR) capabilities so that voter registration applications can be submitted online using a paperless system. These capabilities have been made available to other US states. The Presidential Commission on Election Assistance has recommended that those states which are already moving to OVR should adopt the technology, while those states considering a move to OVR should bear in mind that much of the technology required can be adopted and adapted, rather than having to be developed from scratch using a lengthy and expensive public tender for a new IT system.

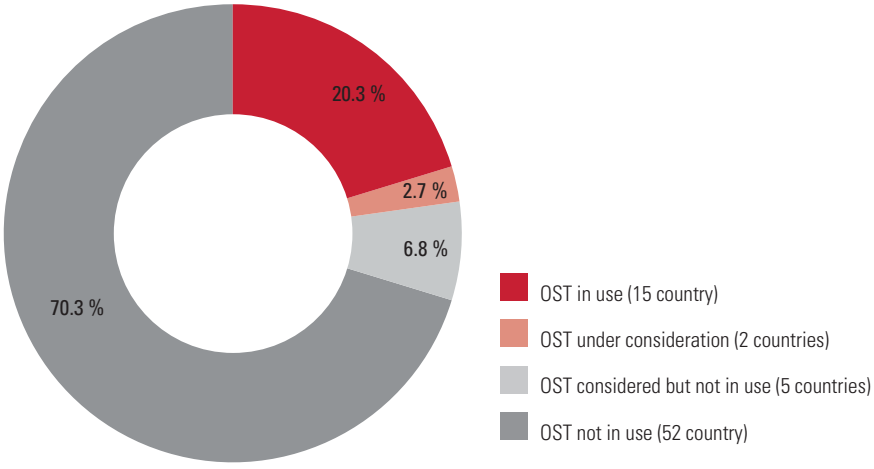
A culture of openness, sharing and learning

OST can also lead to the development of a culture of openness, sharing and learning. This is often an unexpected positive side effect of using open systems. OST is conducive to an environment where local ownership and the development of local expertise within and outside the EMB are seen as important for the acceptance of new technical solutions. Where OST is used, this creates an environment where interested stakeholders are encouraged to investigate, experiment and integrate other technologies in an attempt to understand and possibly even improve and innovate their current offering. These activities increase the likelihood of learning about and building trust in technology solutions.

Who is using OST?

As is noted above, there are many advantages to using OST in the different stages of the electoral process. However, there are few examples of countries that are actually using such technology. In 2014 International IDEA conducted a survey on the use of OST in electoral processes. Over 130 countries were included in the survey, of which 73 responded in time to be included in this publication. Preliminary results from this survey show that:

- 52 countries (70.3%) have never considered the use of OST in their electoral processes.
- 15 countries (20.3%) have introduced and/or are using OST in their electoral processes.
- Five countries (6.8%) have considered using OST, but have not yet started to do so.
- Two countries (2.7%) are still considering the use of OST in their electoral processes.



The main uses of OST in the 15 countries that have introduced it include voter registration, processing results, general data management systems and e-voting.

Figure 3. An overview of coverage and preliminary results of the survey

Updated survey data available at International IDEA's Unified Database, <http://www.idea.int/uid/>

The reasons given for using OST include transparency and the fact that it can be developed by a larger community. The reasons given for not using OST included security-related concerns, a lack of the necessary infrastructure and the belief that only commercial software is reliable. There were also concerns about the possible lack of support from vendors, and assumptions that OST might not be suitable for enterprise-level software, as well as that it is developed by amateur developers and thus insecure and unsustainable.

Box 6. The use of OST in Egyptian elections

In addition to the survey, a question about the use of OST was posted on the ACE Practitioners Network (http://www.aceproject.org). A response from Egypt explained that in the seven elections since 2011, the Egyptian EMB has relied heavily on the use of OSTs, especially in the following stages of the electoral process:

- Polling stations inquiry service: over 54 million voters found the name, address and location of their polling station using OST.
- External voting: an online system was used to manage external voting.
- OST was also used on the elections website to register provincial voters, for candidate registration, for the consolidation and transmission of results and for the registration of observers and members of the media.

Overcoming barriers

If open source technologies deliver increased transparency, sustainability and trust in the electoral process, this raises questions about why they have not been more widely adopted in the administration and implementation of electoral processes worldwide. This section presents a number of observations on why OST has not yet been more broadly adopted in practice, in the hope that EMBs can identify their particular reasons and find insights from this Guide that might induce them to reconsider.

Barriers

Limited awareness

The decision on whether to adopt open source technologies or closed source technologies is perceived by many practitioners in the electoral sphere—as well as others—to be a technical decision that can only be made by specialists. As such, the decision is often delegated to the contractors, or left open in tender documents. As a consequence, most systems in the electoral domain are closed source, because vendors tend to prefer closed source technologies.

A fragmented market

Although elections are a regular occurrence in most countries, they are socio-technical processes and as such unique events. Voter registration systems, for example, need to connect to national registration databases or require the active registration of voters. Countries use a wide variety of electoral systems, such as proportional, preferential and majority voting systems, each of which require different allocation algorithms. This means that one standard technology solution cannot fit all needs. The market for election technologies is therefore highly dependent on local context, culture and national identity, which means that there is much less opportunity for shared ownership and shared development compared to other areas in which open source technologies have proved successful. In some administrative cultures, such as federalism where the authority to run elections and procure election materials has been delegated to several lower levels of government, the market is even more fragmented. This provides even less incentive for the formation of an open source community willing to evaluate and contribute to the effort.

Proprietary solutions

Information technology companies often seek to protect their intellectual property by making their solutions proprietary, that is, closed to third-party inspection or contributions. This is also true for the current market in elections technology, where vendors have successfully marketed and implemented proprietary solutions in many countries. There are even cases of solutions developed by university research projects that were initially released under an open source licence, but reverted to providing proprietary solutions after the university set up start-up company spin-offs to market their research.

Open source charges

Although software vendors usually insist on their software being proprietary, some are willing to release their software under an open source licence, but only for an additional open source charge which increases the upfront cost of the initial software purchase.

EMBs' need to protect themselves against potential criticism

Often, countries that introduce IT have faced criticism from the press, activists and scientists, who have claimed that such technologies are harmful to the democratic process. It is therefore understandable that governmental institutions and EMBs should seek ways to reduce exposure to such criticism. However, this should not deter EMBs from considering the use of OST, as the potential benefits outweigh these concerns. Instead, EMBs should focus on educating the relevant stakeholders on the advantages that the use of OST might bring to the electoral process.

OST software is difficult to maintain
One concern is that electoral system development that stems from an existing OST development may require additional efforts on the part of the EMB to synchronize the code bases. This is indeed the case. Merging subsequent releases of the OST software into the main branch can be difficult. The associated costs are determined by the frequency with which these changes occur, and how pervasive they are. To mitigate the costs, it might be a good idea to contract a vendor to take responsibility for this.

Misconceptions

In addition, possibly the main reasons behind scepticism about OSTs are some of the 'myths' or misconceptions surrounding the issue. These include the following.

OSTs are insecure
This remains one of the most common misconceptions, derived from the assumption that by publishing source code one is also publishing the security-relevant secrets that are contained within it. Good software engineering techniques, however, demand even from proprietary systems that the operational part of the source code is separate from the part that needs to remain secret. Attempting to achieve security by hiding the source code, which is often referred to as *security by obscurity*, although practised by many, is not endorsed by the security community, which argues that obscurity is not a preventive method against attacks. It is important to note that cryptographic keys, user names and passwords, which need to be kept secret, are protected even in an OST. For example, if the algorithms that are used to encrypt and decrypt data are open, and the relevant stakeholders can access and review them, this will increase the trust of the election stakeholder that only secure and widely accepted algorithms have been used. In addition, in the age of surveillance, it is safer to assume that nation states have the capacity to access the source code of any mission-critical system, including electoral systems, and therefore security-relevant details should not be contained in those codes.

OSTs are not mature enough to be used in deployment
There is a stigma attached to open source developments, which can in part be explained by how open software projects are executed. Everything is done in the public domain. Therefore, intermediate, unstable and untested versions are available for download, and bad experiences have been associated with these versions. There is also a misconception that the API of an open source system is less stable than that of enterprise-level systems. There are however many examples of open source projects that work in a reliable and dependable manner. Given sufficient resources and expertise, it is possible to achieve the desired levels of maturity, code quality, reliability, professionalism and publicly shared ownership, especially if the resources in question can be provided by

vendors that embrace the open source approach. There is nothing to prevent OSTs from undergoing the same levels of quality control as any enterprise-level system would.

OSTs automatically mean a loss of intellectual property

It is true that all IT systems (hardware or software) contain algorithms, technologies and software artefacts that vendors and EMBs view as IP, and which therefore require protection. It is, however, a misconception that using OSTs is tantamount to not protecting IP. Although the IP is made transparent in OSTs, there are other means to protect it, for example, by formulating restricted open source licences or by registering patents.

OSTs lack the vendor support required for enterprise-level use

This assumption goes back to the early days of OST when companies were reluctant to use such technology due to the lack of professional services guaranteeing support and maintenance. However, many open source products are now supported not only by a volunteer developer community, but also by a growing number of professional open source companies that generate revenue from professional services provided to enterprise users of open source software. OST is commonly used by a wide variety of adopter, commercial and government organizations, among others. In government IT, one familiar model involves a government adopter organization selecting a familiar government-IT oriented system integrator or other IT services organization. The government organization contracts the system integrator to assemble a particular system from commodity hardware, commercial software, open source-licensed software and sometimes custom software.

OST software can be changed by anyone at any time

There is no requirement to accept any modifications made to OS-licensed software. It is the custodian organization that chooses which modifications are selected. It is true that anyone can obtain the source code and propose modifications or extensions as they see fit, but only the custodian organization can agree to accept these modifications. In many open source projects, there are participants who make extensions or improvements that are not taken up into the public repository of OS-licensed software. Just because an OS-licensed source is 'publicly readable' does not mean that it is 'publicly writable'.

CHAPTER 3

How Can OST Be Used?

Introduction

In many business sectors, OST has become a competitive alternative to closed source software. However, little progress has been achieved so far in the elections field. This chapter gives a brief description of how to develop and use OSTs in elections, and how to assess the feasibility of adopting OSTs. It also provides broad guidelines on the procurement process.

This chapter investigates the different models of closed and open source service provisioning, as well as how a community for the development of OST could be developed. It also examines some of the measures needed to make the implementation of OST sustainable.

Feasibility

The procurement of any new ICT to be used in the electoral process should be preceded by a feasibility study. The outcomes of the feasibility study should be an analysis of the potential of and threats posed by the proposed technology, and a thorough investigation of the basic assumptions and requirements or features required to facilitate decision-making and procuring the system. As the general area of feasibility studies is well understood, this Guide focuses on those elements which are relevant to the feasibility of adopting OST in elections, especially with respect to technical, legal, economic and political feasibility.

Technical feasibility
If the use of a particular OST is required, for example, systems that are released under a GPL, the entire software system may also have to be released under a GPL. Conversely, if a vendor bases a system design on proprietary systems,

then the entire system may have to be proprietary. Finally, the use of proprietary data formats, for example in the use of bar codes on voter cards, may lead to the undesirable situation in which an EMB is locked in with a particular vendor.

Legal feasibility
Although OST provides additional levels of transparency, it might still be incompatible with national law.

Political feasibility
It may require some additional education for political stakeholders to embrace OST.

Economic feasibility
As is noted above, vendors may charge an open source charge for publishing the source code online, due to the perceived loss of intellectual property.

Mapping needs

IT is one of the main drivers of the design of new systems to be acquired for elections. Voter registration systems, for example, build on existing database technologies and technologies for capturing biometrics, such as fingerprints or iris scans. Internet voting technologies cannot simply be deployed in many nations, but require a legal framework that legitimizes their use.

It is useful to consider the framework in which the feasibility study takes place. When computer systems are used in elections, it becomes harder for election observation missions (EOMs) to evaluate internal processes and operational steps. Closed source systems, also referred to as black box systems, have to be trusted. OSTs, on the other hand, have some potential to be audited and for subsequent evaluation. Many countries, for example, Germany, are bound by constitutional law to allow any citizen to gain confidence in the accuracy of the election result, which means that the law plays an essential role when mapping needs or requirements. Countries that require this level of openness must not procure proprietary technology that would be in violation of the law. In addition, reports from EOMs can play an important role when assessing the need to improve an electoral process. For example, if an EOM final report recommends improving the voter registration system, there is obviously a need to be addressed. As a rule of thumb, when mapping needs it is useful to consider the role of OSTs as an alternative to proprietary systems, simply because OSTs provide higher levels of transparency.

Mapping capacities

If, while mapping needs, OSTs are identified as part of the solution, the EMB needs to map available capacities and existing infrastructures. Capacities include

human, financial and logistical resources including transport and storage facilities. Mapping also needs to include technical capacities, such as access to the Internet, mobile phone networks, and access to servers to publish artefacts.

Understanding the market

Once needs and capacities have been mapped, it is recommended that the EMB collect information about the vendors and service providers available to implement the project. Vendors will include software and hardware vendors, supplemented by in-house capacities. Additional resources may also be available from intergovernmental organizations and non-governmental organizations (NGOs), which may already have built capacities or even made available OSTs that could be incorporated.

Furthermore, we recommend that the EMB carefully evaluate the different tendering options. Some tendering options may fare better with the design of OSTs than others. Norway, for example, tendered its Internet voting platform using a competitive dialogue method (see Box 7).

Decision-making

The final step in the feasibility study is decision-making. Decision-makers armed with the results of these analyses will have to weigh the different concerns and arrive at a solution. It could be argued that the decision on whether to use OSTs, and more specifically which precise formulation of the licence to use, should be taken at this point, and neither presented to another committee nor left to the vendor. The results of the decision-making should provide clear and coherent recommendations intended to inform the procurement process. Ideally, this decision-making will be done in an inclusive way, involving all the electoral stakeholders.

Procurement

The procurement process for an election can be a very complex and long process. OST is only one, albeit important, component of the procurement process. This section focuses on the important issues to consider if the procurement includes OST. As procurement regulations are very different in different parts of the world, this section provides a brief description of a 'generic' method that can be adapted to different situations, allowing some flexibility and taking into consideration different laws and regulations.

One of the main challenges in the procurement of OST for elections is that the vendor market is changing rapidly. This makes it difficult to know who is delivering what, and what is available under what conditions. Therefore,

much valuable information will have to be collected before and during the procurement process. This is one of the reasons why the competitive dialogue method is recommended for such procurements, as it enables the EMB to have a dialogue and gather crucial information from the vendors before the official tendering process starts.

Box 7. Competitive Dialogue

Competitive Dialogue is a European Union process introduced in 2006. It permits discussion of different options before choosing a particular solution. It can be used in complex contracts where technical solutions are difficult to define or where the development of the best solution is required. For more information see:
<http://admin.exeter.ac.uk/corporate/procurement/manual/definitions/c.shtml>.

Considerations ahead of the procurement process

Demanding OST
Software vendors tend to protect their intellectual property by avoiding open source licences. This means that it is up to the EMB or the relevant electoral body to propose the use of OST during tender processes and contract negotiations. Experience has shown that the advantages of OST justify efforts to identify vendors willing to produce open source products.

Define the requirements as early as possible in the process
As the market for OST in elections is relatively new and immature, it is recommended that developing the procurement requirements should not be undertaken by EMBs, or the relevant electoral body, behind closed doors. If allowed by the relevant government procurement laws, competitive dialogue is recommended in order to define the necessary requirements as early as possible in the process. If this is not possible due to restrictions related to these laws, thorough market research and a wide consultation process are recommended.

Other important considerations include:

1. *Sourcing strategy—in-house vs outsourced*
   It is important to decide whether software development and maintenance are to be carried out in-house or outsourced, or to decide on a balance between what is to be done in-house and what should or can be outsourced. Factors that could influence this decision include the budget, and the availability of human resources and other internal capacities. Existing government regulations could also influence the decision. It is important

to remember that regardless of which model is chosen—in-house or outsourced—the procurement of OST is not a one-off deal, and software and systems need to be regularly revised and updated. For example, changes in the electoral law or other relevant laws and regulations will have effects on the systems in place. In general, regular updates and revisions are important in order to improve the process and enhance the quality of the system. It usually takes a few election cycles for the EMB to really get to know how the system works and what improvements should be made to fit the specific context. When making a decision, it is therefore important to take a long-term perspective of at least ten years.

2. *Contract strategy—fixed price vs open book (pay as you go) or a combination*
   A fixed price strategy could be appropriate if the EMB is working with a fixed and predefined budget. However, it might prove difficult and time-consuming to decide upfront on all the formal specifications, and the EMB will then be tied to these specifications. This solution might also imply a greater risk that the final product will not be exactly what is required, or will not fully serve its intended purpose. Assuming some flexibility in the budget, the pay as you go option might be a better fit for the procurement of OST, especially if combined with the option of agile software development (see below). This combined solution could work well in certain contexts, but it is important to keep in mind that the combined solution option might result in highly complicated contracts.

3. *Software development strategy—open source vs closed source, and agile software development vs fixed software*
   When it comes to deciding on open or closed source, it is important that the EMB pose the question based on the reality in its country. The advantages of using OST, and in which contexts it could be most useful, are described above. The other decision to be made in the context of a software development strategy is whether the software development will be agile, that is, done in-house in collaboration with the vendor, or all the formal specifications will be set out during the procurement process and the EMB will receive a final product that it will have to test.

4. *Requirements for the solution—technical aspects, including security and functionality*
   These requirements are usually decided on before the start of the procurement process either in a competitive dialogue process or through market research and consultations with key actors and stakeholders. If an agile software development strategy has been chosen, these requirements can be kept to a minimum in the contract. Otherwise, especially if a fixed software development strategy has been chosen, it will be important to go into detail regarding these requirements.

Setting out the contractual specifications and the legal framework

Once a vendor has been chosen, it is important to think carefully about the contract's specifications and the legal framework. One important decision to be made is on the type of licence to be used. OST has many licence types, and licences can be amended and adapted to specific needs. A decision should be made on which licences are suitable for OST designed for elections. For example, Norway used a modified version of an existing licence, which gave the Norwegian Ministry of Local Government and Regional Development full rights in Norway while the vendor retained rights in the rest of the world (see Box 2). Other contractual specifications include software development and maintenance specifications and technical requirements for the solution, as discussed above.

Community development and maintenance

Vendors of voting technology rely on the returns they get from selling the products they have developed. However, the development of OST requires a different business model to ensure continuous software development, reliable functionality and long-term maintenance. While it would seem that such a new model would present many opportunities, vendors are currently reluctant to voluntarily and proactively take the risk of fully embracing an open source approach. However, it is likely that vendors would take this leap of faith if there were a strong enough demand from their customers or if the market shifted in this direction through the establishment of an elections OST project and community. Embracing an open source approach in the election technology community would not only increase transparency but also consolidate the valuable, but limited, research and development resources in this field. An overview of different business models and their relationship to the development of OSTs is set out below.

Community models

The community model (or volunteer development) is the most commonly used model. Often, developers work in a grass-roots, decentralized manner to build a particular product with a shared vision. However, software developers do not usually 'donate' their efforts, time and knowledge to the project. They generally seek some sort of gain for themselves or their organizations. For example, software developers might join efforts to build a product that they can then turn around and use in their own businesses or organizational pursuits. Alternatively, they might build products in common that they can then use to achieve recognition in the community, and thus attract future clients.

Corporate models

Corporate, for-profit models of OST development are usually service-based. The software is developed and distributed at zero cost to clients and customers who want it. Revenue streams are realized through services provided to the client, and usually related to the implementation and testing of the software.

NGO models

NGO business models generally rely on philanthropic contributions to develop, distribute and maintain OST. While there is an aspect of community involvement—mainly due to the not-for-profit nature of the initiative—these organizations generally compensate the developers they contract for their efforts.

Box 8. The Open Source Election Technology Foundation: building an open source development community

The Open Source Election Technology (OSET) Foundation is a Silicon Valley-based NGO founded and funded by experienced software development professionals from firms such as Netscape, Apple and Facebook. The foundation has built a network of 200 development community members who are building or have agreed to build open source solutions for election management bodies. In the light of the mission-critical nature of democratic election technology, the OSET Foundation's organizational structure for the coordination of such a large effort was developed based on the model used by the Mozilla Foundation, to ensure a central screening and testing function. However, the organization also empowers developers to work on their own projects, including those which will be flexible as components of an election management system. A key goal is the interoperability of these components. OSET is seeking to grow its open source developer community into a global network for sharing interests and work.

Source: <http://www.regjeringen.no/nb/dep/kmd/prosjekter/e-valg-2011-prosjektet/kildekode/tilgang-pa-kildekode.html?id=646007>

Election organizations models

Election organization models are similar to NGO models. Generally, there is some level of funding for software development, open source or otherwise, contributed, for example, by the government. Software developers are paid a wage to develop, and possibly to test and implement, the systems. Under this model, there is a tendency for election organizations and governments to maintain ownership of the IP developed, and a reluctance to share this IP with others. In a sense, a governmental body might reason that since it paid for the development of the software with public money, it is entitled—and even obliged—to keep the code secure and secret, and not share it with other people or governments. Thus, the election organizations models can result in 'pseudo-open source software' that is not truly shared with others, and cannot be built on for the benefit of other countries.

University models

University OST research and development models are driven by research and educational goals, and are usually publicly funded. Therefore, computer science departments are generally in favour of the use of open source software and contribute to OST development. However, once research ideas have matured enough, they are often commercialized by start-up companies. While universities have not ventured much into OST for election administration, they remain a promising potential future partner.

There is no ban on for-profit companies

Any type of legal entity can adopt and use open source-licensed software, including for-profit companies. The Red Hat Licence is probably the best example of a thriving business of IT products and services based in part on open source-licensed technology. The terms of the open source licence must be honoured by a licensee that is for-profit, but these licence terms do not prevent use by a for-profit company. Indeed, open source technology has become so prevalent that it is relatively rare for a for-profit company's proprietary software not to incorporate the use of open source-licensed software for several common functions, such as cryptographic algorithms, network security protocols, HTML-rendering engines and operating systems.

New partnerships and cooperation

There is great potential in the possibility of international partnerships between the different actors and organizations involved in the development of election-related OSTs. Such partnerships will enable the consolidation of resources and the further development, distribution and deployment of OST systems for use in a larger number of electoral processes. Such partnerships could improve

collaboration between NGO/not-for-profit developers, universities, EMBs and vendors, and allow the leverage of research and the integration of existing code sets, leading to the improvement of existing and the creation of new products that respond to the needs of diverse electoral contexts. We have yet to see such a maturation of an open source community on a wider scale in the area of elections management.

A specific possibility for new partnerships could be to foster cooperation between universities, government and industry, also known as the Triple Helix (Etzkowitz and Leydesdorf 2000), where there currently appears to be little interaction on electoral systems and specifically on open source technology. Most public universities are obliged by their mission statements and funding mechanisms to benefit the public. Many public universities have schools of public administration, political and social sciences, informatics (computer science) and electrical engineering. Electoral OST involvement is a cross-disciplinary endeavour, which makes it an ideal area of focus for universities today as the goal of many is to break down or at least better integrate knowledge silos. For example, the department of public administration could work in conjunction with the department of computer science. For some universities, this would be an unprecedented interaction on campus.

There is considerable potential for a university to specialize in open source elections technology. Such a specialization might sit between political science, electrical engineering, information science and even management studies, which offers operations and logistics education. Master's-level courses might be on offer, as well as short continuing professional development courses to train election officials so they can gain the technical know-how to oversee projects, and build and administer these complex systems.

Last but not least, there are opportunities for collaboration between industry, government and universities on research and the co-development of new technology for voting systems that can then be shared across the three realms. The use of the term 'industry' in this context does not refer to vendors of election technology systems, but to an industry that has a demand for voting technology. For example, corporations, both privately and publicly traded, might have a need for shareholder voting systems or customer voting systems. These industry partners might fund some of the open source software development components through a foundation partner and a university. This software could then be shared.

Box 9. Examples of successful OST communities

The Mozilla Foundation[5]

Established in 2003 in Silicon Valley, the Mozilla Foundation is a non-profit organization that seeks to support and lead the open source Mozilla Project. It describes itself as 'a non-profit organization that promotes openness, innovation and participation on the Internet'. The initial contributions for its start-up came from America Online (AOL) and Mitch Kapor. Together they donated 2.3 million US dollars to launch the Mozilla Project. The first project successfully launched by the team was the Firefox browser. The Mozilla Developers Network currently has over 3400 members, who contribute their efforts to a variety of projects. Most of these are peripheral to the core product of the Firefox browser. A centralized decision-making team decides which projects will be implemented in the core products of Mozilla, and which will remain catalogued as side projects.

The Moodle Community

The Modular Object-Oriented Dynamic Learning Environment (Moodle) was developed off the back of a PhD study on open source learning software by Martin Dougiamas in 2002. As the original developer, he sought to break the monopoly of incumbent vendors such as Blackboard and WebCT, which he believed had a stranglehold on universities. Today, Moodle has 55,000 sites in 230 countries and 67 million users.[6]

Moodle is freely distributed under the GNU General Public Licence and has an open source development community of approximately 1 million developers. Members of the Moodle Community may freely share ideas, code and courses online. Organizationally, these developers are classified as either core or non-core. There is also a commercial arm, the Moodle Partner Network, made up of developers and partners who act as consultants and are paid royalties and consulting fees through the entity.

The SAKAI community

Founded in 2005 by four universities, the SAKAI Project is a community of educational institutions, developers, instructors and others who have worked together to develop a common Collaboration and Learning Environment (CLE) that is free and open sourced. Its outputs are distributed under the ECL, an open source licence. The software is now on its tenth iteration and is widely used among the education community.[7] SAKAI, originally funded philanthropically by the Mellon Foundation, currently has over 350 educational institutions on board. The network is administered by the university partners themselves and uses the SAKAI confluence wiki to distribute information (see <https://confluence.sakaiproject.org/dashboard.action>).

Like the motives behind the creation of the Moodle Community, the SAKAI Project has sought to break the monopoly that incumbent vendors have on the marketplace for learning management systems and courseware. The founding universities were motivated by lowering the TCO, and increasing flexibility and the ability to integrate new developments, use platforms more creatively and increase interoperability.

Each university partner agrees to dedicate one staff member to the open source community. In return, it receives 'free use' of the software under the ECL and is entitled to future upgrades. Now that the development momentum has been set in motion, other educational institutions that wish to join, including state-funded schools, among others, may do so without necessarily making resource commitments to the network in terms of development talent.

Capacity-building and sustainability

In order to ensure sustainability, once a voting technology system has been commissioned, EMBs must plan for further customization, future maintenance and other system improvements in between electoral cycles, without heavy dependency on developers and vendors which may for any number of reasons be unable or unwilling to continue their cooperation with the EMB.

Independence from vendors will be a key challenge for any EMB that is considering outsourcing voting technology under a closed source licence. Ideally, dedicated IT staff will be employed and trained over time. Alternatively, the EMB might consider integration and coordination with an open source community or communities. EMBs can also find software updates and assistance with systems integration through various communication channels, such as bulletin boards, online forums, mailing lists, social media sites, and so on. *Building local capacity* is key, as this will enable EMBs to be less reliant on proprietary software vendors and build more sustainable systems. Ways to accomplish this include:

Adopting CI to achieve the long-term sustainability of OST systems. Election systems need improving, testing, debugging, upgrading, and so on in-between election cycles. This can be accomplished by building local capacity and through interaction with a community. EMBs are advised to familiarize themselves with CI and build skills in this regard.

Building capacity through collaboration between and networks of EMBs, as is mentioned above, will be required to share expertise, resources, code, systems and manpower. This will help make the adoption, customization, development and deployment of open source software more viable in the long term. The creation of an elections-specific open source software licence should also be considered.

Involvement in local open source communities, in addition to online open source communities, is another way to build capacity. EMBs could contribute to these communities by creating forums on OST and elections to address, among other things, the specific problems and challenges faced by the EMB.

In addition to the provision of new solutions, such forums might also assist with recruiting valuable expertise in the form of employees or consultants.

In conclusion, in order to create a more sustainable electoral system in the long run, EMBs must interact with open source communities, while also collaborating among themselves, in order to share resources on the development and improvement of systems and implement CI.

Provision of services in election administration

The development and maintenance of quality election administration software is a complex and resource-intensive undertaking. Moreover, the market for buying and selling such technology is limited, in terms of both suppliers and users. In addition, the world of research and development of election technology is fragmented and isolated. Vendors tend to develop independent commercial solutions, while some EMBs develop their own solutions in-house. Much of these efforts addresses similar issues, and would therefore benefit from joining forces in order to maximize the results.

Vendors

EMBs usually enter into a relationship with a vendor in order to procure professional services related to the provision of a technical solution that 'gets the job done', in part because they lack the required technical expertise and knowledge to manage a complex system in the long term, but also because they need an external partner that can take responsibility for the technical aspects of the system. Many commercial vendors use business models whose economics are based on paid, long-term licensing for the use of elections software and systems. Their systems are usually closed, often referred to as 'black box systems', mainly because the vendors believe that the source code is their IP and the basis of the advantage they have over their competitors. They are therefore often reluctant to provide full access to EMBs, political parties or relevant auditors. While the vendors' point of view is understandable to a certain extent, their reluctance to give full access to elections stakeholders often leads to a lack of trust in the elections technology, which can lead to a lack of trust in the electoral process as a whole. An important side effect of this is that an EMB might end up locked in, in a situation in which it is so dependent on a particular vendor and the vendor's particular solution that it can neither maintain the system nor respond to changes in the political and social environment without that vendor.

Compromises and trade-offs

One approach to increasing transparency in closed solutions might be to allow public testing and auditing of the technology through evaluation and certification exercises, and mock elections that use the technology. While these steps may not fully disperse all doubts about what goes into the black box, they would be a good first step, especially if combined with a sound technology that performs accurately and reliably in the 'real' elections.

As the need for transparency increases, some vendors have moved towards a more open model of software licence by granting access to selected groups of stakeholders, which usually have to sign non-disclosure agreements, to conduct code reviews, audits and overall quality assessments.

A few vendors have developed and released software as open source, albeit under certain restrictions. One example is the software licence for the Norwegian Internet voting system, which is open source and grants anybody permission to review it. However, the licence limits the rights to *use* the system to the Norwegian Ministry of Local Government and Modernisation for the specific purpose of conducting elections in Norway. Nonetheless, this licence, however limited, clearly adds to transparency in the Norwegian Internet voting system (see Box 2).

Conclusions

As part of an effort to explain the opportunities and challenges of open source technology in election administration, this Guide provides a general overview of open source approaches and discusses which tools in election administration could benefit from using OST. The Guide also addresses arguments for and against the adoption of OST in voting technologies and attempts to dispel some of the misconceptions surrounding OSTs.

The general aim is to provide a practical hands-on guide to considering OST from an electoral point of view, and provide concrete leads on how an *open source state of mind* can affect feasibility studies, procurement, development, maintenance and sustainability.

In sum, the Guide draws a number of conclusions on the use of OST in election administration.

1. Elections should be transparent. Thus, voting technology should be transparent, which in turn requires the source code of voting technologies to be accessible and transparent.

2. The decision on whether to adopt OST for elections should not be left to vendors or technical experts, but rest with the EMB, which is responsible for the transparency of the electoral process.

3. Intellectual property associated with voting technology can be protected without endangering the transparency of elections. The protection of intellectual property is not in violation of open source licences.

4. There is a need to define an open source licence for voting technology that is readily understandable, deployable and usable in the electoral context.

5. More widespread use of OST in elections would require vendors to adopt flexible business models that incorporate OST.

6. Awareness must be raised among EMBs. EMBs, decision-makers, political parties, civil society and the media should be encouraged to build capacity on the benefits of the use of OST in elections, which in turn will create increased demand for accessible and transparent source codes in election technologies.

7. OST considerations must play a more prominent role in feasibility studies, which weigh alternative options for voting technologies, and must subsequently be properly reflected in the procurement process. This will give the use and development of OSTs in elections a critically required impetus.

8. Open source voting technology would greatly benefit from the establishment of a global electoral OST community. Such a community would make releasing existing and newly developed voting technology under an open source licence more feasible.

Notes

[1] See for example an analysis on France, available at <http://www.e-voting.cc/wp-content/uploads/Proceedings%202006/Proceedings2006_GESAMT.pdf>.

[2] For more information see <http://opensource.org/licenses/alphabetical>.

[3] See for example <http://www.content.eisa.org.za/sites/eisa.org.za/files/imports/import-data/PDF/vrafrica.pdf> and <http://www.ifes.org/~/media/Files/Publications/Books/2011/Civil_and_Voter_Registries_final.pdf>.

[4] See also <https://www.supportthevoter.gov/files/2014/01/Amer-Voting-Exper-final-draft-01-09-14-508.pdf>.

[5] See also <https://www.mozilla.org/en-US/foundation/>.

[6] See Moodle Statistics, <http://moodle.net/stats/>.

[7] See <https://sakaiproject.org>.

Glossary

Algorithm	a step-by-step procedure for calculations
Apache	an open source server-side software application that is partially credited for the initial rapid growth in the worldwide web. It is developed and maintained by a network of open source developers who are organized in the Apache Foundation
Bugs	errors in computer programmes
Ciphertext	the result of an encryption
Code set	a set of source code from several computer programmes
Competitive Dialogue	a European Union pre-tendering process that permits discussion of different options before choosing a particular solution (<http://admin.exeter.ac.uk/corporate/procurement/manual/definitions/c.shtml>)
Continuous improvement	an ongoing effort to improve products, services and processes
Cryptographic algorithms	encoded mathematical equations, used in this context to protect data and the privacy of voters
Cryptographic method	the use of mathematical means to protect a message from unauthorized access
Custodian organization	an organization that maintains and monitors an open source technology project

De/encryption	the process of using cryptographic methods on a given message to make it un/readable
e-voting	the use of electronic means in elections to cast or count votes
FreeBSD	an operating system that is assembled under the model of open source software development and distribution
Free distribution	publicly accessible software that is royalty-free
Gimp	an open source-based graphics package
GNU General Public Licence (GNU GPL, GPL)	a form of open source licence, where intellectual property is made available for reuse, modification and distribution without regard to purpose and without discrimination
Government adopter organization	a government organization that is intending to use a particular service or software
Government-IT-oriented system integrator	a company that provides integrated IT services that can, but do not have to, come from several other companies to public sector stakeholders
HTML-rendering engines	software that makes code visible on a computer monitor
Intellectual property	the legally recognized exclusive rights to creations of the mind
Linux	an operating system that is assembled under the model of open source software development and distribution
Majoritarian voting systems	electoral systems designed to produce an absolute majority (50 per cent plus 1) of votes
Mission-critical system	any system, or part thereof, the failure of which would result in the failure of the business process
MySQL database server	open source database software

Network security protocol	processes and methods to secure network data from illegitimate attempts to review them or extract data from the network
Obfuscation	an automated way of making the source code of a computer programme hard to understand
Open source charge	an extra fee charged by a vendor for making a closed source programme public under an open source licence
Open Office	an open source-based package that contains word processing, spreadsheet and presentation software
Open Source Initiative	a global non-profit that supports and promotes the open source movement. Among other things, it maintains the Open Source Definition, and a list of licences that comply with that definition (see <http://opensource.org/>).
Operating system	software that forms the essential base for managing the resources provided by computer hardware and software and the management of computer programmes that 'sit on top of it'
OST stack	an open source technology programme comprising at least two open source software solutions interacting with each other
Patch file	a file that only contains the difference between the source code of two different computer programmes
Preferential voting systems	electoral systems in which voters rank political parties or candidates on the ballot paper in order of preference
Pre-processor	a source code transformer
Proportional voting system or proportional representation	an electoral system family based on the principle of the conscious translation of the overall votes of a party or grouping into a corresponding proportion of seats in an elected body
Proprietary rights	a computer software licence that does not disclose its source code
Proprietary software	a synonym for closed source software that is not released under an open source licence

Shared development	when more than one person, most commonly not located in the same place, develops a programme
Strategic business unit	a line of business that a particular firm deals and trades in strategically. For example, a software company might be in the business of selling enterprise software to car makers and electoral management software to EMBs. These would comprise two distinct strategic business units because each requires its own marketing strategy.
SugarCRM	an open source customer relationship management system based in Silicon Valley that offers a free, open source version of its software as well as a proprietary version
System validation	a process of evaluating a computer system for compliance with a set of requirements
Third-party hardware	hardware that is not directly related to the technology in use
Translator	a computer programme that translates a programme written in a given programming language into a functionally equivalent program in a different programming language
Triple Helix	a sociological theory that posits that interaction, collaboration and the changing roles of government, universities and industry lead to innovation and the economic development of regions. It proposes that universities are becoming the central players as knowledge and its places of creation grow in importance.
Unix	an operating system originally developed at AT&T's Bell Laboratories. Early versions were proprietary. This led to criticism and eventually the creation of Linux, which was distributed as a free and open source competitor operating system.
Vendors	for-profit firms that sell proprietary voting systems to EMBs. These organizations may have other lines of business too, but for the purposes of this Guide they sell electoral management systems.

References and Further Reading

ACE Electoral Knowledge Network (ACE project), <http://www.aceproject.org>

Council of Europe, *Legal, Operational and Technical Standards for e-voting. Recommendation Rec(2004)11 and Explanatory Memorandum*, Strasbourg, 2004

Etzkowitz, Henry and Leydesdorf, Loet, 'The Dynamics of Innovation: From National Systems and "Mode 2" to a Triple Helix of University-Industry-Government Relations', *Research Policy*, 9/23 (2000)

International IDEA Unified Database, <http://www.idea.int/uid>

International Institute for Democracy and Electoral Assistance, *International Obligations for Elections: Guidelines for Legal Frameworks* (Stockholm: International IDEA, 2014)

Krimmer, Robert, *The Evolution of E-voting: Why Voting Technology is Used and How it Affects Democracy*, Institute for Public Administration (Tallinn: Tallinn University of Technology, 2012)

Open Source Initiative, <http://opensource.org/osd>

Shaikh, Maha and Cornford, Tony, *Total Cost of Ownership of Open Source Software: A Report for the UK Cabinet Office Supported by Openforum Europe* (London: UK Cabinet Office, 2011)

Zuckerman, David T., *The Voting Machine: Report on the History, Use and Advantages of Mechanical Means for Casting and Counting Ballots* (New York: Political Research Bureau of the Republican County Committee of New York, 1925)

About the Contributors

Michael Clouser, BS/MBA (Cornell University), MSc/PhD (University of Edinburgh), is the European Director of the OSET Foundation and an academic entrepreneur. Michael is an e-learning developer and a former venture capitalist in Silicon Valley who has maintained a keen interest in open source technology, especially in the area of higher education. He taught technology entrepreneurship at the University of Edinburgh, where he led the Edinburgh-Stanford Link collaboration for seven years, and currently teaches courses at various universities in Canada, the USA and the UK. He also provides consultancy services at universities on innovation and new venture creation, including spin-outs, start-ups and venture capital funds. Michael's research interests are universities, public policy and innovation, and his blog, *The Academic Entrepreneur*, covers these subjects. Contact: mclouser@osetfoundation.org.

Professor Robert Krimmer, MBA (Vienna University of Economics and Business—Wirtschaftsuniversität Wien, WU), PhD (Tallinn University of Technology), is Professor of e-Governance at Tallinn University, focusing on the transformation of the public sector, electronic democracy and related issues. Previously, he served as Senior Adviser on New Voting Technologies in the Election Department of the Organization for Security and Co-operation in Europe's Office for Democratic Institutions and Human Rights (OSCE/ODIHR) in Warsaw. His role was to coordinate and support election-related activities where new technologies are used in elections, and contribute to developing the methodology in this respect. Professor Krimmer founded and chairs E-Voting.CC, and initiated the biannual EVOTE conference series held in Bregenz, Austria. In 2003, he received the BMW Scientific Award. Contact: robert.krimmer@ttu.ee.

Henrik Nore, MS (Norges teknisk-naturvitenskapelige universitet), MBA (University of Navarra), is a Project Manager at the Ministry of Local Government and Modernisation, Norway, responsible for the central ICT systems supporting the Norwegian elections. He initially joined the ministry for the e-vote project in 2008. Prior to that, he worked as a manager for a private-sector IT consultancy in Norway. Contact: henrik.j.nore@gmail.com.

Associate Professor Carsten Schürmann, PhD (Carnegie Mellon University), is Associate Professor at the IT University of Copenhagen, focusing on information security, cryptography, language-based security, formal methods, computational social choice and trust. He leads the DemTech research project on trustworthy democratic technologies, an interdisciplinary research effort between computer science and social science that studies trust in the use of technology in elections. In 2002, he received the NSF CAREER award. Contact: carsten@itu.dk.

Peter Wolf, MS (Graz University of Technology), is a member of the Elections Processes Programme at International IDEA. He has focused on ICT applications in electoral processes since joining the Elections Department of the OSCE Mission in post-war Bosnia and Herzegovina, where he worked on voter registration and results databases. He has been a consultant on voter registration projects in Albania, the Democratic Republic of Congo and Iraq, and served in various international election observation missions, including as an electronic voting expert in France, Kazakhstan, Kyrgyzstan, Venezuela and the Philippines.

About International IDEA

What is International IDEA?

The International Institute for Democracy and Electoral Assistance (International IDEA) is an intergovernmental organization with a mission to support sustainable democracy worldwide. The objectives of the Institute are to support stronger democratic institutions and processes, and more sustainable, effective and legitimate democracy.

What does International IDEA do?

The Institute's work is organized at the global, regional and country levels, focusing on the citizen as the driver of change. International IDEA produces comparative knowledge in its key areas of expertise: electoral processes, constitution building, political participation and representation, and democracy and development, as well as on democracy as it relates to gender, diversity, and conflict and security. International IDEA brings this knowledge to national and local actors who are working for democratic reform, and facilitates dialogue in support of democratic change. In its work, International IDEA aims for:

- increased capacity, legitimacy and credibility of democracy;
- more inclusive participation and accountable representation; and
- more effective and legitimate democracy cooperation.

Where does International IDEA work?

International IDEA works worldwide. Based in Stockholm, the Institute has offices in the Africa, Asia and the Pacific, Latin America and the Caribbean, and West Asia and North Africa regions.