

UPGRADE is the European Journal for the Informatics Professional, published bimonthly at <http://www.upgrade-cepis.org/>

Publisher

UPGRADE is published on behalf of CEPIS (Council of European Professional Informatics Societies, <http://www.cepis.org/>) by **Novática** (<http://www.ati.es/novatica/>), journal of the Spanish CEPIS society ATI (*Asociación de Técnicos de Informática*, <http://www.ati.es/>)

UPGRADE monographs are also published in Spanish (full version printed; summary, abstracts and some articles online) by **Novática**

UPGRADE was created in October 2000 by CEPIS and was first published by **Novática** and **INFORMATIK/INFORMATIQUE**, bimonthly journal of SVI/FSI (Swiss Federation of Professional Informatics Societies, <http://www.svifsi.ch/>)

UPGRADE is the anchor point for UPENET (UPGRADE European NETWORK), the network of CEPIS member societies' publications, that currently includes the following ones:

- **Informatik-Spektrum**, journal published by Springer Verlag on behalf of the CEPIS societies GI, Germany, and SI, Switzerland
- **ITNOW**, magazine published by Oxford University Press on behalf of the British CEPIS society BCS
- **Mondo Digitale**, digital journal from the Italian CEPIS society AICA
- **Novática**, journal from the Spanish CEPIS society ATI
- **OCG Journal**, journal from the Austrian CEPIS society OCG
- **Pliroforiki**, journal from the Cyprus CEPIS society CCS
- **Pro Dialog**, journal from the Polish CEPIS society PTI-PIPS

Editorial Team

Chief Editor: Llorenç Pagés-Casas, Spain, pages@ati.es

Associate Editors:

François Louis Nicolet, Switzerland, nicolet@acm.org
Roberto Carniel, Italy, carniel@dgt.uniud.it
Zakaria Maamar, Arab Emirates, Zakaria.Maamar@zu.ac.ae
Soraya Kouadri Mostéfaoui, Switzerland, soraya.kouadrimostefaoui@gmail.com
Rafael Fernández Calvo, Spain, rfcvalvo@ati.es

Editorial Board

Prof. Wolfried Stucky, CEPIS Former President
Prof. Nello Scarabottolo, CEPIS Vice President
Fernando Piera Gómez and Llorenç Pagés-Casas, ATI (Spain)
François Louis Nicolet, SI (Switzerland)
Roberto Carniel, ALSI - Tecnoteca (Italy)

UPENET Advisory Board

Hermann Engesser (Informatik-Spektrum, Germany and Switzerland)
Brian Runciman (ITNOW, United Kingdom)
Franco Filippazzi (Mondo Digitale, Italy)
Llorenç Pagés-Casas (Novática, Spain)
Veith Risak (OCG Journal, Austria)
Panicos Masouras (Pliroforiki, Cyprus)
Andrzej Marciniak (Pro Dialog, Poland)
Rafael Fernández Calvo (Coordination)

English Language Editors: Mike Andersson, Richard Butchart, David Cash, Arthur Cook, Tracey Darch, Laura Davies, Nick Dunn, Rodney Fennemore, Hilary Green, Roger Harris, Michael Hird, Jim Holder, Alasdair MacLeod, Pat Moody, Adam David Moss, Phil Parkin, Brian Robson

Cover page designed by RFCalvo, © Rafael Fernández Calvo 2006

Layout Design: François Louis Nicolet

Composition: Jorge Llácer-Gil de Ramales

Editorial correspondence: Llorenç.Pagés-Casas@ati.es

Advertising correspondence: novatica@ati.es

UPGRADE Newslist available at

<http://www.upgrade-cepis.org/pages/editinfo.html#newslist>

Copyright

© Novática 2006 (for the monograph)

© CEPIS 2006 (for the sections UPENET and CEPIS News)

All rights reserved under otherwise stated. Abstracting is permitted with credit to the source. For copying, reprint, or republication permission, contact the Editorial Team

The opinions expressed by the authors are their exclusive responsibility

ISSN 1684-5285

Monograph of next issue (February 2007)

"Next Generation Web Search"

(The full schedule of UPGRADE is available at our website)



The European Journal for the Informatics Professional
<http://www.upgrade-cepis.org>

Vol. VII, issue No. 6, December 2006

- 2 Editorial: UPGRADE in Top Position of Google PageRank for ICT Journals — *Geoff McMullen*

Monograph: Open Document Format (ODF) (published jointly with Novática*)

Guest Editors: *Jesús Tramullas-Saz, Piedad Garrido-Picazo, Marco Fioretti*

- 3 Presentation: OpenDocument Standard for Digital Documents — *Jesús Tramullas-Saz and Piedad Garrido-Picazo*
- 5 Open by Design: The OpenDocument Format Standard for Office Applications — *Erwin Tenhumberg, Donald Harbison, and Rob Weir*
- 14 Is OpenDocument an Open Standard? Yes! — *David A. Wheeler*
- 25 OpenDocument Hidden Traps and their Side Effects on Free/Open Source Software — *Marco Fioretti*
- 29 ISO-26300 (OpenDocument) vs. MS-Office Open XML — *Alberto Barrionuevo-García*
- 38 Interoperability: Will the Real Universal File Format please Stand Up? — *Sam Hiser and Gary Edwards*
- 47 ODF: The Emerging Document Format of Choice for Governments — *Marino Marcich*
- 50 Promotion of the Use of Open Document Formats by the IDA and IDABC Programmes — *Miguel A. Amutio-Gómez*
- 53 A Brief History of Open Standards in Denmark — *John Gøtze*
- 57 Standard Open Formats and Libre Software in the Extremadura Public Administration — *Luis Millán-Vázquez de Miguel*

UPENET (UPGRADE European NETWORK)

- 60 From **Mondo Digitale** (From AICA, Italy)
Programming Languages
Programming Languages: An Introduction — *Carlo Ghezzi*
- 64 From **Pro Dialog** (PTI-PIPS, Poland)
E-Commerce
Organization and Economics of Entertainment Services Networks Exchanging Virtual Goods — *Andrzej P. Urbański*

CEPIS NEWS

- 70 Harmonise Project: Assigning Responsibilities among the Partners and Planning Next Steps — *François-Philippe Draguet*
- 71 News & Events: European Funded Projects and News Updates

* This monograph will be also published in Spanish (full version printed; summary, abstracts, and some articles online) by **Novática**, journal of the Spanish CEPIS society ATI (*Asociación de Técnicos de Informática*) at <http://www.ati.es/novatica/>.

OpenDocument Hidden Traps and their Side Effects on Free/Open Source Software

Marco Fioretti

OpenDocument is widely considered, especially within the Free/Open Source Software (FOSS) community, one of the most important tools for the promotion of FOSS itself and for a truly free Information Technology market. OpenDocument is also rightly considered essential, as far as IT is concerned, for the construction of a more free society and culture. The very nature of OpenDocument, however, is not enough to overcome some obstacles to those goals and, in any case, is very likely to have a deep and sometimes unexpected effect on the future of FOSS. This article introduces these obstacles and side effects and, when they create actual problems, briefly describes the best kind of solution to deal with them.

Keywords: Certification, Data Ownership, Free Software, Long Term Archiving, OpenDocument, Open Source, Open Standards, Public Administrations.

1 The Advantages of OpenDocument

A highly structured, metadata rich, application independent XML (eXtensible Markup Language) file format like OpenDocument can finally offer two huge advantages to all computer users and to Society as a whole. The first is complete interoperability among many software applications, regardless of their user interface, license or development model. It is worth noting that this capability is not limited to office productivity software, but will be usable and useful with a much wider range of products. Being XML, OpenDocument files can be generated on the fly, analyzed or indexed automatically by any number of independent server-side applications.

The other really important thing that OpenDocument makes possible, on a large scale and for the first time, is reliable long term archiving, without any loss of information, of digital office documents. This is something which is not possible with other standards, including those specifically developed for archiving like PDF/A (Portable Document Format Archive). The latter "*only addresses those files that might be described as a digital representation of a paper document*" and "*will ensure that a PDF document will be rendered as it was created 50 years from now*" [1]. Consequently, PDF/A is enough only when the only available open version of an already *existing* document is a PDF file, or with documents created by scanning paper. PDF/A files are not meant, for example, to preserve the history or, above all, the internal algorithms of the original document. They do not preserve different versions, or of the formulas from which the printable numbers in a spreadsheet are generated. This would make the future historical or scientific analysis of, among others, law proposals, election reports or global warming studies much harder than it could be if the original had been stored in OpenDocument format.

Of course, reliable long term archiving depends on many more variables than the file format alone, from the physical

Author

Marco Fioretti is a hardware systems engineer and a part time freelance writer for several Linux and other IT magazines. He is the co-founder (in 2002) and current coordinator of the Run Up to date Linux Everywhere Project <www.rule-project.org>, which makes modern Free Software easily installable on older computers. Marco Fioretti has always been interested in truly Open file formats and protocols, from e-books to portable databases and the Universal Business Language, and in their impacts on economy, education and civil rights. In 2004, for example, he investigated the philosophical links between Free Software and Scouting. In the same spirit, in 2006 he co-founded Elèutheros <www.eleutheros.it> which aims to promote a wider, official adoption of Free Software and non proprietary formats within the Catholic Church. More recently, Fioretti has also started to follow the attempts of the Free Software and disabled users communities to communicate more effectively. His more recent initiative to promote the importance of Free Standards and Software among all citizens is the Family Guide to Digital Freedom <<http://digifreedom.net>>. Marco Fioretti is also member and contact for Italy of the OpenDocument Fellowship <<http://opendocumentfellowship.org>>, a volunteer organization that promotes the use and development of the OpenDocument format, and the author of Everybody's Guide to OpenDocument <www.linuxjournal.com/article/8616>. <marco.fioretti@opendocumentfellowship.org>.

media to the file system specifications, but they are outside the scope of this paper.

2 Impact of OpenDocument on Free/Open Source Software

Many FOSS supporters, including politicians who hope that FOSS will create more local IT jobs, seem to have not yet realized that a general adoption of OpenDocument would *delay* the adoption of FOSS in several scenarios or, in any case, change the way in which FOSS desktop applications are to be developed and supported.

Almost any large public or private organization already has an equally large, installed and partially paid for IT infrastructure, which would take years to migrate. The only thing that really prevents interoperability with such organiza-

tions, or among them, is not which software they run. It is the fact that they (can) only produce or accept proprietary file formats, for which, by definition, FOSS filters can never be perfect.

In addition to this, in many countries only interoperability and data ownership can be realistically included as mandatory requirements in public tenders. It is immensely easier and much more coherent, at least in a Free as in Freedom [2] society, to lobby for laws which only require free formats and protocols, rather than any specific software "*product*", whatever its license might be.

Imposing the adoption of OpenDocument is the easier and most effective line of action in this context. To read or produce OpenDocument files, however, a Microsoft Office user has no need whatsoever to migrate to OpenOffice or Linux. All he or she needs is to install, or have installed, a plugin for Microsoft Office like the one being development by the OpenDocument Foundation [3]. In other words, by just adding OpenDocument support to their existing infrastructure, many private and public users may have much *less* reasons or external pressures to migrate to FOSS. Once data ownership and interoperability are guaranteed, migration does not just become much easier, it also becomes much *less urgent and attractive*. Therefore, FOSS will have to compete in other fields in order to be adopted by desktop users, regardless of license prices. Performances on limited hardware, usability, end user documentation, and free online support for non-programmers will have to be consistently better than those of proprietary software to convince those users to migrate to, and keep using, FOSS solutions.

Some FOSS advocates will probably be upset by these new trends, but it would be wrong to fight them. First of all, mandatory adoption of OpenDocument would immediately make the 100% FOSS desktops of any student or small business owner completely compatible with the proprietary ones already existing in governments or corporations. Through OpenDocument, FOSS-only desktops would become much more usable for business, research and active politics than they are today and much earlier. Secondly OpenDocument will eventually *improve* the quality of FOSS, for the reasons just mentioned.

3 Hidden Traps

It is very likely that, eventually, all major producers of both proprietary and Free software in the office files space **will** support OpenDocument. In and by itself, however, there are several ways in which the standard enables monopolies to continue, or its usefulness to be nullified for long term archiving.

Technically speaking, OpenDocument is very powerful and useful because it can be extended. Robert Weir, a Software Architect of the IBM Software Group, recently discussed with the health care standards consortium MedBiquitous [4] an OpenDocument based format for physician certification which would have to add to each file digitally signed XML metadata. However, extensibility is neutral with respect to intentions. It is possible to have a

perfectly Free as in Freedom XML container full of patent-ridden components. If anything, the fact that an office file standard is not owned and controlled by one vendor may make this even *easier*, not harder, as proprietary extensions appear to keep end users locked in, at least in some scenarios.

3.1 General Issues

Where the Open Document Format for Office Applications (OpenDocument) v1.0 specification (OpenDocument-v1.0-os,swx) allows a new namespace to be introduced for externally-defined elements is an entry point for proprietary extensions, if the namespaces are not publicized and the schemas and semantics for the elements are not entirely published and legally usable. For example, arbitrary introduction of foreign tags below the paragraph level is permitted, as well as arbitrary document metadata (section 2.2), formatting properties (section 15) and their application-determined defaults. The standard also specifies how unknown tags should be treated and, above all, states that at least some of them *should be preserved*. As a practical example, OpenOffice already preserves OLE objects and alternative renderings for images, in order to maintain compatibility with Microsoft Office if the document is exported back to Office. All these "foreign" elements, however, are intrinsically not safe for use in interchange, because there is no predictable treatment of them beyond such an optional, passive preservation [5].

3.2 Multimedia Formats

The OASIS standard "Open Document Format for Office Applications (OpenDocument) v1.0" specification obviously allows the inclusion of images, audio or any other multimedia object in texts, spreadsheets and presentations. It says nothing, however, about the *license* or any IP restriction of the corresponding file formats, so it is perfectly legal (as far as the standard is concerned) to include multimedia content in proprietary or patent-restricted formats inside OpenDocument files.

3.3 Macros

Macros are the only realistic way, for non-programmer users of office suites, to achieve two intrinsically different goals: one is to extend the functionality of one specific software program *on all the files it operates*: dictionary interfaces, spelling checkers, word counters and similar fall into this category. The other is to embed some kind of data processing capability or interactivity *into* one specific document: a real-world example may be an e-learning course, usable even without an Internet connection, made of interactive forms. This is similar to what happens with Web browsers. There are both Firefox specific extensions and web pages containing JavaScript applets that can run in any browser.

Macros of the first kind have nothing to do with OpenDocument, but the interactive course scenario is quite different. If that course consists of forms inside

OpenDocument texts or spreadsheets, then its users and distributors will expect all the buttons, form fields, checks on user input and so on, to work without problems inside *any* OpenDocument-compliant program. The standard, however, specifies how macros must be embedded in the document, but not what their language should be [6].

3.4 In-file Databases

Single-file, server-less SQL databases, which besides the data also contain schemas, indexes and forms structures are not as powerful as full-fledged RDBMS solutions, but they are extremely useful in a very common and important case. They make it possible for all computer users, including those without programming skills, root password or permission to install extra software on their computers, to create and above all exchange databases just as easily as text or images [7]. The popularity of MS Access in small offices is a proof of this fact.

OpenDocument files may include such single-file databases. OpenOffice, for example, uses HSQLDB as its default format for this. In general, OpenDocument files may also be dynamically linked to external databases. In both cases, the same issues as with multimedia objects apply: nothing in the standards prevent linked or embedded databases from being proprietary or undocumented.

3.5. Digital Signatures and other Security Related Extensions

Both governments and large corporations have interest in ways to give or deny access to different sections of the same document. An internal report, for example, may be published on the Internet with only some paragraphs encrypted. An OpenDocument file may also contain its own digital signature, to certify its authenticity. Even individual metadata may be signed, as in the MedBiquitous case. Again, any cryptographic algorithm, regardless of its license or the completeness of its public documentation, may be used for this purpose.

3.6 Formulas

Version 1.0 of the OpenDocument standard did not mandate a common format for the formulas embedded in spreadsheets, and this created well known interoperability problems [8]. This issue, however, will be fixed in version 1.2 of the standard, so it only exists for OpenDocument files already archived today or in the next 12 months.

3.7 Fonts

Fonts are referenced by name, e.g., "*Times New Roman*". Of course, they are only used for presentation, not encoding of information. They will not create substantial interoperability or archiving problems, but using fonts which are proprietary or available on only a single platform can create unnecessary trouble, for example making distribution of printed copies without modifications impossible. This is especially true for scientific or technical documents with many complex formulas.

4 Nature of the Solution

As already described, "100% ISO26300 Compliance" is not enough to guarantee that an OpenDocument report or law proposal stored today will be completely readable and usable 20 or more years from now. In order for that to happen, all the components of such files should be at least as open and documented as OpenDocument itself, but it is possible and relatively easy do the contrary, thereby keeping the end users locked into a specific software product. Both benefits of OpenDocument, that is complete interoperability and long term archiving without information loss, would be lost in such cases.

Generally speaking, only part of the solution is technical and can be placed in the applications. For example, a good option to have in OpenOffice (or any other word processor) would be to automatically convert all imported graphics into an open format like PNG (Portable Network Graphics) or SVG (Scalable Vector Graphics). The same could be done with fonts. When databases are concerned, archiving procedures should perform, when necessary, an automatic inclusion of the database linked from the OpenDocument file(s).

Automatic XML checkers which scan the content of OpenDocument files and signal any closed component, comparing its format against an accepted list, would also be easy to implement and very useful. Public Administration and libraries could run them much like antivirus software, refusing to accept any file whose interoperability, now and in the future, cannot be guaranteed.

At another, but still technical level, well defined interoperability testing is also essential to distinguish the implementations which are willing to cooperate. The University of Central Florida, for example, is already working on an OpenDocument test suite which would allow implementations to verify the conformance of a generic OpenDocument-enabled software program with the specification [9].

At the deepest level, however, this is not a format specification issue. The specification of OpenDocument or its license should not and could not contain, allow or forbid every conceivable extension, not to mention those which do not even exist yet. This is implicit in the very concept of open and extensible standard: it must *facilitate* data exchange between implementations which are *willing* to exchange them, but it can not force compatibility when this does not happen. The problem must be solved in another way, which is only partly technical.

The first step is to raise the awareness, among system administrators and policy makers, that these issues do exist and must be faced *before* they start to become a serious problem. The second is to implement the automatic checkers mentioned above. The third step may be to officially define an "OpenFile" or similar trademark which is applicable only to files in which no component has restrictive licenses or incomplete documentation. Part of the problem here is of course to define *who* should take this action. In the European Community, a likely candidate may be the

Interoperable Delivery of European eGovernment Services to public Administrations (IDABC) [10].

The last and most important step would then be to require by law that all the OpenDocument files stored or exchanged with public Administrations, libraries and so on be compliant with the "OpenFile" trademark, regardless of the software application which created them.

There are obviously several key cases where it may be unavoidable to grant exceptions to such a rule, at least in the short and medium term. Making sure that public documents are and remain completely open, of course, remains a mandatory prerequisite for a truly open society, and OpenDocument alone is not enough to reach this goal.

References

- [1] Martin Bailey. Why Would Anyone Write a PDF Standard Just for Archivists?, <<http://www.pdfzone.com/article2/0,1895,1841569,00.asp>>. July 29, 2005.
- [2] Sam Williams. Free as in Freedom: Richard Stallman's Crusade for Free Software. O'Reilly Media, 2002. ISBN: 0596002874.
- [3] The OpenDocument Foundation, Inc. <www.opendocumentfoundation.us>.
- [4] MedBiquitous. <www.medbiq.org/>.
- [5] Dennis E. Hamilton Unofficial analysis of the OpenDocument specification <<http://orcmid.com/writings/2005/06/w050601b.htm>>. October 11, 2006.
- [6] Marco Fioretti. Macros an obstacle to office suite compatibility, <<http://software.newsforge.com/article.pl?sid=05/09/09/1640253>>. September 17, 2005.
- [7] Marco Fioretti. The Lack of a Small Unified Database, <www.linuxjournal.com/article/7800>. September 28, 2004.
- [8] Marco Fioretti. OpenDocument office suites lack formula compatibility. <<http://software.newsforge.com/software/05/09/09/192250.shtml?tid=93>>. September 20, 2005.
- [9] Open Document Fellowship. Open Document Sample Documents. <<http://testsuite.opendocumentfellowship.org/>>.
- [10] European Commission. Interoperable Delivery of European eGovernment Services to public Administrations (IDABC). <<http://europa.eu.int/idabc/>>.