Complex singularity vs. openness

Open source impeded by incompatibilities and inconsistencies in the Office Open XML document format

Management summary

When it comes to office documents, public administrations can choose from two ISO/IEC standards. Only one of these, ODF (ISO/IEC 26300), is vendor-neutral, open and reliable across a span of years and software versions, and supported by a variety of software products.

The later OOXML standard (ISO/IEC 29500), originally developed by a single proprietary software vendor, is implemented in three different versions (‘ECMA’, ‘Transitional’ and ‘Strict’) that are not compatible with each other. Although the ‘ECMA’ and ‘Transitional’ versions are outdated – ‘Transitional’ had only been accepted as a temporary solution to give the software vendor time to implement ‘Strict’ in its products – they both continue to be used in practice. This is because older versions of the vendor’s office suite (MS Office) cannot read or write OOXML Strict and are unlikely ever to gain such abilities.

This inconsistency causes a variety of problems for developers of free and open source office solutions, including:

- loss of metadata;
- errors in graphics and other embedded objects;
- embedded links to technologies controlled and used exclusively by the proprietary vendor;
- errors caused by embedded languages like VML (Vector Markup Language);
- multiple implementations of spreadsheets.

Because of this, to date there are no free and open source solutions that fully support OOXML.

Problems with the document format also negatively affect public administrations. Archive experts warn that inconsistencies between the OOXML versions will cause problems with older documents, so that users may not be able to open or work with them. University studies show that even the proprietary vendor’s own products cannot deal with the different OOXML features properly.

- Public administrations using a mix of versions of the ubiquitous MS Office will not be able to use ISO 29500 to exchange ISO 29500 compliant documents, say open source experts including software developers and scientists working on interoperability of suites of office productivity tools;
- Experts have shown that public administrations should not rely on ISO 29500 when exchanging documents, as this is likely to create ambiguities when using office tools that do not fully

support ISO 29500;

- Such a mixed environment always results in documents that are not ISO 29500 compliant and can lead to data loss;
- To most users of such documents, the format’s ambiguity is not apparent, as the resulting documents carry the same file extension (.docx) – but this masks the fact that the document format can be encoded in several different ways;
- The various versions of OOXML that are not ISO 29500 compliant cannot be implemented by others, including open source developers, without re-implementing the proprietary office suite;
- Compared with other ISO standards, ISO 29500 is lengthy, complex and written in language that is difficult to understand. Experts say that a complete re-implementation in free and open source solutions will take up to a decade, with its success depending on the proprietary vendor’s cooperation;
- Many of the features in ISO 29500 are tied to versions of the proprietary office suite, reflecting this software’s history and development decisions.

> **“Nobody is using only OOXML Strict”**
>
> Michael Meeks, LibreOffice developer

Introduction: Why this debate won’t go away

Development of the OOXML standard began in 2004 at ECMA International (formerly the European Computer Manufacturer’s Association), a private, membership-based non-profit standards organization. Two years later, ECMA involved the international standardisation organisation ISO, which is composed of representatives from various national standards organisations. OOXML (also known as Office Open XML) became an industry standard alongside ODF, the Open Document Format used by a variety of products including LibreOffice, Apache OpenOffice and Google Docs.

OOXML’s original author, Microsoft, promises that its Office software fully supports OOXML as the ‘most-used document standard’. But there is a growing debate about the standard. Both its independence and the possibility of re-implementing it in other software products have become matters of intense discussion among free and open source software developers and public administrations.

Michael Meeks is one of the main developers of LibreOffice. Formerly at Suse, he now works for the LibreOffice support company Collabora. Meeks is convinced that OOXML (ISO 29500) is not being used ‘in the wild’, and says there will never be an OOXML standard independent of Microsoft. ODF, on the other hand, is a standard suitable for long-term document editing and archiving. ODF is supported by a variety of organisations, including many large ICT firms, whereas the only organisation claiming to support OOXML is its one and only vendor.

“Nobody is using OOXML Strict only”

Furthermore, according to Meeks, “it is extremely unlikely that any administration is really working with the ISO-compliant OOXML standard only”. Organisations will have users working with older versions of the MS Office suite, he says, or they will be working on older documents. They will be exchanging documents with citizens, companies or administrations using office tools that do not fully support ISO 29500, and that will break the chain of OOXML Strict standard documents, often without informing users. Meeks continues: “If a company or an administration uses different versions of MS Office, including old ones like MS Office 2007 or 2010, then documents exchanged through a chain of people will not stand up to the ISO Strict standard of OOXML as specified in ISO 29500.”

For the typical daily use of a connected company or a municipality working with customers, citizens or partners, the inconsistencies around the OOXML standard are likely to cause problems sooner or later. Typically, a document might be created by someone using MS Office 2013, then edited and re-saved by a colleague with MS Office 2007 or 2010. Because the old versions of Microsoft Office do not comply with the latest, ‘Strict’ OOXML standard, metadata is very likely to get lost during such round-trips. And since the ‘Strict’ standard used by Microsoft is still neither fully documented nor open (it contains references to Microsoft websites, some of which no longer exist), data loss on conversion is a widespread and well-documented phenomenon. What makes this problem even worse is the absence of error messages, and a file extension (for example .docx for text documents) that does not show the user which format has been used to save the document.

“Even though OOXML Strict is a standard which theoretically could be implemented in open source, its specification is so complex that we would have to re-implement huge amounts of detail from MS Office to fully comply with it”, says Meeks. But an even bigger problem for developers like Meeks is support for OOXML in the wild, encompassing the different dialects and how these are supported in the different office suites. “To me it remains unclear why anyone would choose to give a single, rather dominant vendor such a huge head start in the implementation of a document standard, when there are better, more open, document standards to mandate with wider implementation support”, he says.

A complex heritage

OASIS ODF TC member Svante Schubert, another of the veteran engineers from OpenOffice working on the interoperability of office suites, concurs: “You can develop an OOXML-compliant implementation, covering all the mandatory OOXML features, but this implementation will very likely not be compatible with the majority of the OOXML documents that are out there.”

This is due to the many ‘optional features’ hidden in the specification. Many of these have historical roots, and they create redundancies in OOXML, Schubert explains: “From the many available examples, just pick tables. Why are there three different table formats in OOXML? There is one for Word, one for Excel and one for PowerPoint. The answer is: Because the three different departments were working separately on the format. Microsoft either bought or created these products separately, and during the specification phase they had to integrate these three different approaches in a single XML.”

Developers and standardisation experts agree that standardisation should not work like this. Such examples show that the OOXML standard is driven by the needs of a software company more than by the procedures and ideals of normal industry standards, says Schubert. “ODF, on the contrary, has only one specification for tables. This has been developed to ensure compatibility, and is used across all the different application components.”

A report by the European Commission, part of Action 23 in the EC’s Digital Agenda, confirms this argument. The paper indicates that the OOXML format contains severe barriers to implementation. In its final report, the Guidelines for Public Procurement of ICT Goods and Services: SMART 2011/0044 declares: “Whilst standards that are set through formal standard setting organisations go through a formal development process, they may still contain barriers to implementation by all interested parties.” The report illustrates this point with the OOXML standard in particular (footnote 32): “As an example, ISO standard (ISO/IEC 29500) for document formats. The technical specifications of this ISO standard include references to proprietary technology and brand names of specific products. Further, the specification of this ISO standard is not complete (i.e. the technical specification contains references to an external web site which refers to web pages on the vendor's homepage that are currently not available."

With OOXML as a heterogeneous and ambiguous standard, and with Microsoft holding the threads and not updating old software versions, every software developer has to deal with a growing set of separate implementations, software versions and different OOXML 'flavours'. This creates a complexity of problems, with each combination behaving slightly differently on operating systems ranging from Windows XP to Windows 8, with their various sub-versions, patch levels and service packs. Free software developers trying to fix office interoperability issues must not only grapple with the OOXML variations but also test their fixes over a wide variety of operating systems, Office versions, documents and implementations. This would not be necessary with a single, unambiguous and open ISO standard.

Spreadsheet woes

According to developers, such problems can be found in many places within OOXML. And it is not only the open source software movement that is suffering from the varying implementations and interpretations of ISO 29500. Even Microsoft has distinct problems with its document formats.

This is illustrated clearly, even for non-technical readers, by recent observations from Italo Vignoli, published on his blog Marketing FLOSS. Vignoli, a journalist from Milan, is a specialist in Open/LibreOffice and the director of the Document Foundation, the home of LibreOffice.

Vignoli shows how in five steps, all common spreadsheet operations, even Microsoft's most recent Office version, Office 2013, does not properly support Open XML Strict, the declared document standard for all the vendor's products. Creating a spreadsheet in MS Excel, adding some data and some metadata such as dates, calculations or formulas, and saving the document in the ISO-standardized OOXML Strict format can result in trouble when the document is re-opened.

Problems can include data loss and features wrongly represented, which might easily remain undiscovered because they do not produce error messages in Microsoft Office. In the case of Vignoli's spreadsheet, for instance, all the user sees is some wrong dates, maybe some wrong XML tags – mostly things that are likely to go unnoticed, yet which could be disastrous to a public administration. Note, moreover, that this happened in Office 2013, which according to Microsoft has the best and most complete OOXML support of all the Office versions, and which is the only software product currently available that claims full compatibility with OOXML Strict.

This is no ordinary bug

The number of examples of errors like Vignoli's proves that the problem lies much deeper than an ordinary software bug. In January 2014, the Hungarian developer Miklos Vajna presented a preview of the newest version of LibreOffice Writer (4.3). Writer is a complex text editor with many import and export functions – the LibreOffice equivalent of MS Word. IT administrators use it to migrate documents from one format to another, not only because of the quality of its import/export filters but also because of the ease with which these filters can be created and used.

One of the major feature improvements of Writer 4.3 will be the handling of shapes in files imported from MS Office documents using the docx import filter. This is no easy task, however, because Vajna has shown that Office itself interprets the same file differently: where Office 2010 shows a green triangle, Office 2007 has a red one. And this is just one example.

Microsoft's own lack of consistency gave the LibreOffice developers a problem. As in so many other cases, again they had to resort to reverse engineering, yet at the end of the day they still had to choose whether to be compatible with the 2007 or the 2010 version of Office. Vajna admits: “At least we’re now on a par with Word 2010”, only to continue with a list of other features that do not survive the round-trip process of saving, exporting, loading, changing and exporting again. With bitter irony he explains the reverse-engineering work that open source developers have to do simply because the standard is not as open as it should be, and because different Microsoft products interpret it in different ways.

Furthermore, since there is no full and standard-compliant implementation that uses ‘Strict’ completely, there is no real testbed available. Researchers at the Fraunhofer Institute had to develop their own OOXML Strict software suite just to test the possibilities offered by the OOXML and ODF standards. That problem also makes it difficult to develop any OOXML tool – even a simple import or export filter – because developers have to implement Microsoft’s OOXML version; if they did not do this, documents would conform to the standard yet be useless to users. Even worse, as Thorsten Behrens of the Document Foundation explains: “By establishing its Markup Compatibility and Extensibility (MCE) technology in ISO 29500 Microsoft has gained the right to make changes to the document format simply by adding their own extensions, almost without limits. That makes it hard for any other company or open source project to be fully compatible.” In a paper on Microsoft's blog site Paul Lorimer and Doug Mahugh, both Microsoft employees explain what MCE is about: “ISO/IEC 29500 solves this issue [format upgrades] through Markup Compatibility and Extensibility (MCE). MCE combines several approaches to allow newer versions or other vendors to innovate through the addition of new content in a file, while at the same time annotating that content so that it can be ignored, or downgraded by applications that don’t understand the new content.” This, “as specified in in ISO/IEC 29500:2008 – Part 3” at first glance sounds helpful, but it leaves Microsoft in a position where the company basically is allowed to add any updates, upgrades, changes or extensions to an ISO standard. Furthermore, the new content will simply not be shown in older software compliant with ISO 29500 from other vendors. Only software that knows about the newly added extensions, features and their nature will be able to show the document as intended by its creator.

University of Skövde: Lack of continuity

A third example: a computer science professor at the University of Skövde in Sweden has had similar experience with MS Word’s use of OOXML. “It does not work”, says Björn Lundell, whose team at the university’s Department of Information Technology is researching compatibility and standards for office software. No other application can correctly load all the different versions of OOXML produced, Lundell says, even though LibreOffice does a fairly good job by now.

LibreOffice cannot work accurately with Microsoft docx files, especially when round-trips are involved, Lundell says, nor is there even a real chance that Microsoft’s older programs can do the same. “In our ongoing analysis of [lack of] interoperability we are also analysing the longevity of docx files initially created by some MS software”, he explains. The Skövde team is looking into what they call intra-operability between different version of the vendor’s software. They are analysing the extent to which newer versions of MS software can read, edit and re-save documents initially created by older versions.

The problem of opening files from only four years ago

“It is clear that there are several problems related to interpreting files created just four years ago”, Lundell says. “This is, of course, even more complex given the issues and confusion related to the Strict versus Transitional versions of the OOXML standard. If a user created a file in MS Office 2007 and saved it in Transitional docx format, other people should be able to open it in Office 2013, re-save it in Strict docx and finally read it in Office 2010", he says. "But you'd be surprised at the real outcome."

Such a series of round-trips is far from unusual: "Given that all new files should (according to ISO/IEC 29500) be saved in OOXML Strict, this scenario will be quite a common case for many organisations", Lundell says. The study's results are not yet final, but the fact is that for historical reasons Microsoft's programs read and write different versions of an ISO standard. This is becoming a growing problem on the desktops of public administrations.

Misleading public organisations

According to Lundell there is great danger in this situation: "I know that [in Sweden] very large public sector organisations are currently considering migrating all their files to docx since they have the (misinformed) conception that there are open source alternatives that can be used to achieve adequate interoperability. We know that this is not the case. The reality is that the format is so bad that even Microsoft cannot achieve intra-operability between different versions of their own software." Lundell thinks the whole issue needs far more discussion than there is now. "Just making a reference to proprietary functionality as it appears in different versions of MS Office, 'OOXML' doesn't stand up to analysis", he says.

Lundell's study is not the only evidence for the fact that OOXML fails on several features expected from an industry standard. On its website, Microsoft offers a product called 'OOXML Strict Converter for Office 2010'. According to the company's marketing and the commitments the corporation made to the ISO committee, this converter should not be necessary.

Daniel Melin, procurement officer at the Kammarkollegiet – a Swedish public agency concerned with legal and financial matters – asks: "Why is there a conversion tool if the support should be built-in?" On its product website the vendor explains: "OOXML Strict Converter for Office 2010 allows you to open ISO Strict documents that are created using Office 2013 in Office 2010. It will preserve the fidelity of the document. If you make any changes and save the document, the document will be saved in Transitional format." This translates to: "A user of MS Office 2010 can open OOXML Strict documents generated by Office 2013, but Office 2010 will henceforth save the file as OOXML Transitional."

ECMA, Transitional and Strict

The reason for this strange behaviour lies in the history of the ECMA/ISO/IEC document standards. Thorsten Behrens of the Document Foundation explains that OOXML actually comprises three standards or 'flavours': There is the ECMA version (that's the one MS Office 2007 writes, which was certified by ECMA International). Then there is OOXML Transitional, which is relatively close to the ECMA version, and is the format that all later versions to date write as default. Finally, there is OOXML Strict.

Ten years ago, Microsoft realised that ODF was on its way to becoming a generally accepted standard and started to push its own XML-based document format. Within a very short period the vendor managed to define XML versions of its binary document formats that were accepted as a standard by ECMA International. But the same proposal was rejected by the ISO committee because it involved too many binary, proprietary, Windows- and Office-specific dependencies. The ISO working group was convinced that nobody except the IT vendor itself would be able to implement it.

Something had to give. "In the end the ISO working group deemed OOXML Strict to be the intended standard", Behrens says. This was the compromise allowing ISO/IEC 29500 at least to pass the Ballot Resolution Meeting (BRM) stage.

For a limited period ISO allowed another standard, dubbed ISO/IEC 29500 Transitional. "Both 29500

Transitional and 29500 Strict are proper ISO standards", Behrens explains. It's just that the ISO working group does not like Transitional so much – but the compromise struck at the BRM was to have both flavours in the ISO standard. The differences between 29500 Strict and 29500 Transitional are set out in 29500 Part 4 (Transitional Migration Features), which runs to 1,464 pages.

> **"If a user created a file in MS Office 2007 and saved it in Transitional docx format, other people should be able to open it in Office 2013, re-save it in Strict docx and finally read it in Office 2010", he says. "But you'd be surprised at the real outcome."**
>
> Björn Lundell, computer science professor at the University of Skövde in Sweden

A concise history of the standardisation process can be found here. Today, what was supposed to be a single standard still consists of three different formats: ECMA, as used in Office 2007; Transitional, written by e.g. Office 2010; and Strict, which Microsoft claims to be fully supported by MS Office 2013. Most developers doubt that there is any product available which meets the ISO standard for OOXML Strict, and experiences like those of Italo Vignoli or Björn Lundell's working group seem to prove that the IT vendor's claim is untrue. And this discussion is fuelled by the absence of any test environment or test specification that could be used to prove that any piece of software, or a document it has written, complies fully with all the features of the OOXML specification. This, according to Svante Schubert, does not seem likely. The fact that the OOXML specification is already so complex prevents the development of a working test framework.

The press took a lot of notice of the whole process, and published many reports (for example: http://en.wikipedia.org/wiki/Standardization_of_Office_Open_XML#Complaints_about_the_national_bodies_process, http://www.theinquirer.net/inquirer/news/1012179/norway-iso-walk-ooxml, http://arstechnica.com/uncategorized/2008/10/norwegian-standards-body-implodes-over-ooxml-controversy/) alleging that Microsoft was trying to exert pressure on the committees, either directly or via governments. In Norway, 13 experts out of 23 even resigned from the national committee dealing with the OOXML standardisation process, explaining that "clearly commercial interest… was placed ahead of what is best for society."

British Cabinet Office heads for ODF

In early 2014, the UK Cabinet Office voted for ODF as its preferred standard. Several hundred comments were lodged from both sides in an intense discussion. Among the many prominent developers who responded were Michael Meeks, Simon Phipps (a renowned blogger and Java programmer, formerly at IBM and Sun), and Jeremy Allison of the Samba project.

Google's Vint Cerf gave a statement to the effect that Google relied on ODF, too. Microsoft also engaged in the discussion, denying any problems, claiming that OOXML is the most widespread XML document format, and saying that users are not forced to buy Microsoft software: "There are many tools (applications, apps, programs or services) available at a range of costs."

English or Greek?

In a long post, Michael Meeks gives an overview of many of the problems with the different office formats and the hidden reasons behind them. "Having seen how it is possible for a vendor to use the ECMA/ISO fast-track to essentially standardise every detail of their own implementation, it is then interesting to contrast that with ODF", he writes, adding that ODF is usable across a large number of devices and platforms, not just on Windows. The OOXML specification is almost ten times the size of that for ODF (7,000 versus 850 pages), and it still relies on Microsoft binaries.

According to Meeks, OOXML has another striking characteristic: compared to ODF it is much more difficult to understand and to program. Meeks compares OOXML to classical Greek: a highly inflected, extremely complex language “which is frequently written in capitals with no inter-word spacing”.

“ODF could be seen in contrast as the ‘English’ of document content standards: taking inspiration and heritage from many different languages, it is significantly simpler to learn, and communicate with.”

Different goals for different standards

Analysts like those at the Fraunhofer Institute agree that Microsoft wanted an XML successor to its binary formats that would help the corporation keep up the commercial success of its office products in a world where more and more public authorities recognized the necessity of an open standard. The ‘ISO standardized’ label had become an important marketing tool for the software company.

On the other hand, ODF was developed for completely different reasons and with different goals. The reason for its simplicity lies in the motives of the standard: For prominent Document Foundation members like Simon Phipps, the key motivation was that the ODF people were striving for a format as compatible and easy to understand as possible. “Among our goals with ODF we sought to create a document format that empowered citizens to engage with governments without proprietary hindrance,” he says.

Svante Schubert points out that ODF has many more supporting vendors than does OOXML. ODF support is included even in (albeit young and still somewhat experimental) modern web-document implementations. ODF is supported by a variety of free and commercial office products, including vendors such as IBM (Lotus), Google, Microsoft, Corel and Adobe, and by many free and open source implementations, for a variety of desktops and programs, including AbiWord, Scribus, Inkscape, KOffice, Okular and NeoOffice. Schubert asks: “As Microsoft claims to support ODF, is there any reason to raise the complexity and cost (for the British government) in equally supporting two ‘office standards’? What other reason – except keeping up its vendor lock-in – could there be for Microsoft not to support ODF?”

According to the developers, that should not be too much work: When Microsoft implemented the first ECMA International standard in 2007, experts were surprised how quickly the company was able to pin down thousands of pages of concise standardisation. And since the company has implemented ODF for all its Office products since 2007, full and standard-compliant support for ODF should not be a lot of work, assumes Schubert.

OOXML versions cause inevitable errors

The only solution to problems caused by the fact that there is only one vendor – who has offered incompatible versions of both software and standards – would be to buy the newest versions from Microsoft for every desktop, and to work through the whole digital archive of files to ensure compatibility for all old files. This, argue developers and IT specialists alike, is neither a convenient perspective nor a sustainable approach. It would involve lots of time and manpower, with an uncertain result if the vendor decided to change policy or format again at some point in the future.

Developers like Meeks, Schubert and Behrens are sure: It is not at all probable that a company like Microsoft would invest money in an outdated product like Office 2007, just to make sure that old file formats will work. Open source evangelists go even further, fearing that the vendor might use the incompatibility as a way to intensify pressure on those users who are not willing to upgrade. Since no other products are fully compatible with the different versions of OOXML, and since this is causing huge problems on round trips, choosing this path will mean getting stuck with Microsoft – a clear vendor-lock-in.

ODF as a ‘clear path forward’

On the other hand, developers and researchers alike favour the second, widely adopted standard (ISO/IEC 26300) that is used by many products. Björn Lundell of Skövde University, backed by other university studies, has shown that far more software and tools support ODF than is the case for OOXML. In fact, only Microsoft has continuously provided tools for its favourite standard.

The open source community favours ODF. Italo Vignoli points out the arguments on the LibreOffice mailing list: “ODF offers continuity: ODF has a clear path forward, and is actively maintained by OASIS. There is an ODF 1.0 which is an ISO standard, and an ODF 1.2 (backward compatible with ODF 1.0) which is in the process of becoming an ISO standard. Standards definitions, by their own nature, move slowly. This is the reason why LibreOffice is compatible with both ODF 1.0 and ODF 1.2.”

Open source software and non-proprietary, fully documented open standards are also demanded by archive experts when they are asked how to achieve sustainability for digital documents: “Your digital assets will always outlive your software and the vendors”, say Lundell and his colleague Jonas Gamalielsson in a university study. The Swedish archiving association TAM-Arkiv goes even further: “Never use vendor-dependent formats for long-term storage if you can avoid it, because they are often too unstable, too unstructured, and with dependencies to different suppliers’ business strategies.”

Microsoft OOXML, on the other hand, has never been fully implemented by a vendor in accordance with the standard ISO definition, and is not even actively maintained by ECMA International (because that organisation does not focus on document standards as much as OASIS does). Unfortunately, in the market there are more OOXML documents than ODF documents.

Long-term vendor-independent support

Patrick Durusau, Co-Chair of the ODF Technical Committee, points out the dangers that arise from using a standard that is not truly vendor-neutral: “Another argument is that lots of applications supporting the same format makes the format safer for legacy purposes. If some or most vendors decide to drop support for an older format, [it’s] not a problem because you still have choices for the earlier format. When Microsoft drops support for anything in its format, you simply don’t have access to that feature any more. That may be a more telling argument, especially in governments because they have lots of legacy stuff now and will have many more legacy documents in the future.”

With multiple full implementations, ODF gives users a choice of implementers, and safety in terms of future support. The huge number of recently developed import/export filters in Open/LibreOffice also bolster Durusau’s argument.

The ODF Alliance has gone even further, alleging deceit in Microsoft’s strategy: “The fact that the company is not implementing OOXML Strict, while privately extending Transitional, means that the improvements required to make OOXML acceptable to ISO are now being ignored”. According to the ODF Alliance, back in 2010 the convenor of the OOXML Ballot Resolution Meeting declared that: “the entire OOXML project is now surely heading for failure.”

“The fact that Microsoft is not implementing OOXML Strict, while privately extending Transitional, means that the improvements required to make OOXML acceptable to ISO are now being ignored.”

Working together: Community-driven

More and more, public administrations are switching to free and open source software and benefitting from the OSS development model. The trend is clear from the increasing number of examples mapped by the European Commission's Open Source Observatory and Repository:

- France: the Gendarmerie and government ministries;
- Spain: the autonomous region of Extremadura, the Canary Islands and the city of Zaragoza;
- Switzerland: the Federal Court and Federal IT Steering Unit;
- Germany: the cities of Munich, Leipzig and Jena.

Everywhere in Europe municipalities are showing how they can save taxpayers' money by rejecting Microsoft licences and using Apache OpenOffice or LibreOffice.

However, wherever public administrations rid themselves of office suite lock-ins they are confronted by interoperability issues between the free and open source office suites and the ubiquitous proprietary alternative. With the IT market dominated by a single proprietary office suite and failing to focus on an unambiguous document standard, public administrations are missing out on the benefits of standardisation. This was shown by Tineke Egyedi, a specialist on standardisation issues at Delft University of Technology in the Netherlands, in her 2012 paper on the competing document standards. Egyedi points out that the arrival of the OOXML standard prolongs vendor lock-in and increases costs to governments that are forced to support both standards.

And Juan Conde, head of the open source project in the administration of Spain's autonomous region of Andalusia, points out that the document standards aren't the only problems. There are more issues that make documents difficult to transport from one operating system to another, including font sizes and the unavailability of certain fonts on Linux. But as he says: "The font issue is not directly related to OOXML or ODF, but it shows again how important it is to have a clear standard. With text running over margins you lose information and data." In Andalusia the IT departments had tried to use OOXML in free software, and although "the last versions of Libre Office have had significant improvements", Conde was not happy. But as the national interoperability and standardisation regulation only allows ODF and OOXML Strict, Microsoft's products must not be used in the municipalities – theoretically. "As far as I know, there is no office suite that really is OOXML-Strict compliant", Conde confirms. "So, by our regulations, we should not use OOXML."

Pay it forward

Several European public administrations have articulated their struggle with office suite interoperability. One example is the German city of Freiburg, which in 2012 abandoned its move to an open source office suite. Just the year before, the city's project leader on open standards at the IT department explained how "two decades of monoculture in office applications mean there is no pressure on interoperability".

European public administrations have also in vain called on the European institutions to do the right thing. "There still is not the 'big name' weight of some EU institution that would really shake the civil service out of their conservative viewpoint", said Mark Wright, city councillor for Bristol in the UK, in 2011.

That same year, the city of Munich went as far as to send a letter to the European Commission, urging it to help the city in its document interoperability struggle. In his letter, mayor Christian Ude protested against a recommendation by the Inter-Institutional Committee for Informatics that EU institutions should continue to use the ambiguous OOXML standard. This, he wrote, hinders cooperation between public authorities.

Munich is also a good example of a public administration that is actively trying to solve the conundrum. As described in an OSOR study in 2013, Munich's Limux project included an extensive migration to LibreOffice and led to the development of WollMux, an open source tool for managing forms and document templates. Its name is a Bavarian idiom that refers to the *Eierlegende Wollmux*, a fantasy domestic animal that gives meat, wool, eggs and milk. By analogy, WollMux is supposed to do everything the city of Munich needs in the vast field of office document automation. WollMux allows employees to choose templates for their work with customers, filling them automatically with centrally stored data and thereby providing completely accurate documents and printouts within minutes.

This forward-thinking contribution from Munich allows other public administrations to benefit by taking the first of many steps needed to fix their document interoperability issues. The smart capital of Bavaria is also involved in an even more comprehensive approach to solving the riddle of document standards. The city is one of five public administrations working with the German Open Source Business Alliance (OSBA), an association for enterprise open source service providers. The city is a member of the OSBA working group on office interoperability, funding the enhancement of LibreOffice and Apache OpenOffice.

A first call for tender organised by the Alliance in 2012 focused on necessary improvements to the Open/LibreOffice OOXML import/export filter. This first round was funded by a group of public administrations including the Swiss Federal Court and France's Ministry of Culture and Communication.

Six use cases for crowd funded improvement

For this year's second round, the OSBA working group – funded by the German cities of Munich, Leipzig and Jena, plus the Swiss Federal Court and Federal IT Steering Unit (FITSU) – has selected six priorities. One of the main features to be developed concerns change tracking between open source and proprietary office suites. The public administrations issuing the call want to improve the specification of change tracking, and make this part of the Open Document Format ISO standard. This precision should clear the way for Microsoft to implement ODF's change tracking in its office suite, explains Svante Schubert, who helped write the tender specification. "The MS Office suite currently strips change tracking information from ODF documents, making the exchange of ODF documents impossible with inveterate users [of change tracking,] such as legal departments."

The new tender specification, which is called 'Major Feature Improvements for LibreOffice/ Apache OpenOffice', covers five more use cases:

- improve mail merge in Writer;
- improve paragraph handling in Writer;
- implement styles in all content elements of Writer;
- add chart styles in Calc;
- make more functions available in shared spreadsheets in Calc.

The projects are currently in an open procurement phase, with contracts due to be signed in May 2014. The work will be paid for via crowd-funding by the members of the working group, calling on other public administrations to join. All six projects are supposed to be finished by the end of 2014. All code will be licensed as open source; the results will be available to all public administrations and other users of free and open source office suites.

Conclusion

OOXML is a standard that causes problems when used in public services. For an ISO standard it shows surprising weaknesses: No single product complies with the only acceptable variant of the standard (its latest ‘Strict’ version), nor is there any alternative to Microsoft’s products. Over the years the latter have used different flavours of the standard, creating problems for archiving and document exchange. These flavours are present in the wild, making it impossible to use the ‘Strict’ standard with older versions of Microsoft Office – versions that are no longer sold by the vendor and therefore will never learn OOXML Strict.

But most recent versions of the products, and even the newest ISO standard itself, also have severe issues. These cause problems, especially for competitors and free software developers. Whereas the reason for an ISO standard should be to allow competitors access to a standardized technology and henceforth to a certain market, ISO 29500 has some surprising backdoors and pitfalls. There are MCE extensions that only Microsoft knows, and strange legacy components that have their origin in the fact that Microsoft had to adapt its document format to an open XML style format in order to get the desired ISO certificate, months after ODF had become an ISO standard.

On the other side there is a vigorous community of both free open source and enterprise-level developers. They are supported by organisations representing industry, politics and users, all of whom have to spend more time than necessary on programming features to be Microsoft compliant. They tell stories of their tedious work, step-by-step in trial-and-error style, checking the output of their work in the different versions of Microsoft Office. And in the end these filters will still comply only with version ‘X’ of Microsoft Office, not the full OOXML standard.

More and more municipalities all over Europe have acknowledged experiences like those which have plagued Munich. And the OSBA working group’s success shows clearly how powerful a crowd-funded, collaborative approach may be – if there are free and open standards that are not controlled by a single dominant vendor.

Apart from the money saved by not reinventing the wheel, or paying licence fees for proprietary software, there are more fundamental and sustainable advantages that cannot be measured in money. “Your digital assets will always outlive your software and the vendors”, says a Swedish university study, detailed above. “Never use vendor-dependent formats for long-term storage if you can avoid it, because they often are too unstable, too unstructured, and with dependencies to different suppliers’ business strategies.”

This article was written by Markus Feilner for the European Commission’s Open Source Observatory and Repository, OSOR.