The Open Document Add-in for Microsoft Office

WORD HELPER

With a little help from the ODF plugin, Microsoft Office can open up documents saved in the free Open Document format. A short test reveals just how well it works. **BY MARTIN LOSCHWITZ**

If you've worked with Microsoft Office, you may know that Office XP fails to correctly open documents created with Office 2000. The vendor does not focus on continuity – at least not with respect to Office document formats – which has been a lucrative strategy so far. An enterprise looking to upgrade to a new Office version must update every machine. Furthermore, companies that exchange documents would need the same version.

This is irritating and expensive in the long run, as many public offices have recently discovered. The Belgian government and many US states that were previously loyal Microsoft customers recently announced that in the future they would only use programs that support open standard formats. Meanwhile, Microsoft claims it will be introducing its own open document standard in Office 2007 – OpenXML.

Open Document Format

Microsoft critics have a different format in mind: a large consortium of software vendors (which includes IBM, for example) supports the Open Document Format (ODF), the format that OpenOffice already uses by default for new documents. Recently, the Joint Technical Committee accepted ODF as an ISO standard, a goal Microsoft's OpenXML was far from achieving. Then, responding to criticism from various governments, Microsoft announced that it would support the ODF standard. Parallel to this announcement, MS published an ODF plugin for Office 2000, Office XP, and Office 2007 for public testing.

Lacking Native ODF Support

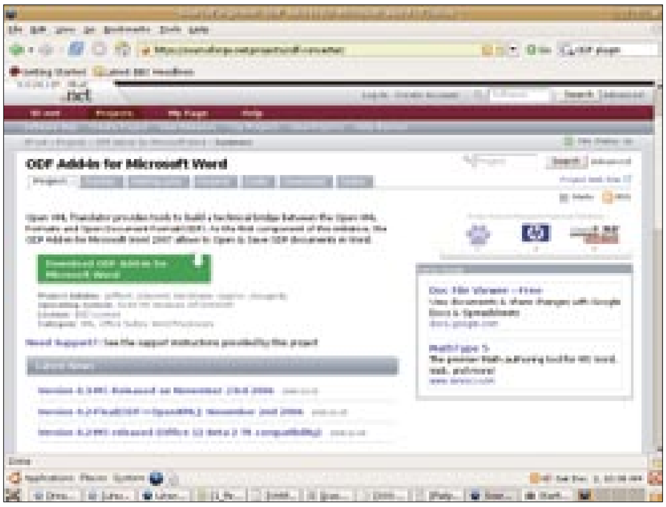
Of course, the fact that Microsoft has published an ODF plugin for Office 2007 means that the new Office version, which is due for release soon, will not have native ODF support. In fact, the plugin does not actually add genuine ODF support to Office; instead, it introduces a routine to convert OpenXML to ODF, and vice-versa. Thus, the OpenXML remains the standard format in Office 2007.

You can download the ODF Add-in for Microsoft Word from [1]. The file, which is quite small, launches into a fairly extensive download session that drops parts of the .NET Framework onto your disk. The installation program does not actually tell you the size of the download; however, the download time suggests that either the .NET Framework is fairly large or the Microsoft server is fairly slow.

After the installation, you can't open ODF documents in the way you might expect (via the *File | Open* menu); instead, you will find an entry called *Open ODF-Document* in your *File* menu.

Test Track

The first test document was a farce. Although I was able to open a document created with OpenOffice 2 by using the *Open ODF-Document* menu item, Word failed to display the text and instead distributed a bunch of weird-looking characters over no less than 69 pages. Research revealed that the ODF plugin converts ODF to OpenXML, Microsoft's proprietary standard format, and it hands an OpenXML document to Word XP after converting. Unfortunately, the older Office version has no idea what to do with the document and stubbornly displays the binary content of the XML file.



ODF Add-in for Microsoft Word

Figure 1: You'll find the ODF Add-in for Microsoft Word at the SourceForge site.

Microsoft Office Compatibility Pack for Word, Excel, and PowerPoint 2007 File Formats (Beta 2 Technical Refresh)

Figure 2: Unless you are using Office 2007 beta, you'll need the Office 2007 Format Compatibility Pack.

None of the current Office versions, which are in use on many corporate computers, understand OpenXML. However, hidden away in the depths of the Microsoft homepage, I stumbled across the "Office 2007 Format Compatibility Pack" [2], which adds OpenXML support to older Office versions. The weird characters disappeared after installing the compatibility package, and Office XP displayed the content of the ODT document without any errors.

I have no explanation for why the homepage refers to the ODF plugin, but then makes you go to another page to find the compatibility pack. It is just as surprising that the ODF plugin installation routine does not just download and install the the compatibility pack.

On With the Text

The first document in my test contained one sentence, which is insufficient to discover whether the conversion routine actually does what it promises. The next document raised the bar a bit higher – besides body text with various formatting, it also had a number of indents.

When I opened the document in Word, I discovered that most of the formatting had been kept, although the ODF plugin had not converted a few page breaks as it should have; some manual work was required. What I didn't like was that Word just ditched the hyphenation that OpenOffice had already done, but without removing the hyphens that OpenOffice had inserted, which meant that a number of words in had hyphens in strange places.

One problem that has nothing with the ODF plugin is bound to occur whenever you convert an ODT document: if the original document uses a font that Windows does not have (Helvetica, for example), the missing fonts are simply replaced by the default font in Word (typically Times New Roman or Arial). If you plan to use ODT documents in Word and OpenOffice, it makes sense to install a set of Microsoft standard fonts on your OpenOffice systems before you start.

Tables and Images

The next obstacle for the ODF plugin was that of correctly displaying tables in an ODT original. Also, some of the cells in the table contained images, just to make things more tricky. Again, the ODF converter showed a few weaknesses in the text: Word did not use the original table width and did not show line breaks in cells correctly. The whole table was not as cleanly formatted as the original.

There were no problems with character sets; no matter whether I fed the converter OpenOffice documents that used Latin1 or UTF8, Word displayed accented and special characters correctly.

Saving in ODF

Although the *File* menu had a *Save as ODF* entry, I was unable to save a file in ODT format. In fact, the entry was permanently dimmed. I could not tell whether I needed to do something else before I could save a file in ODT. This is one of the biggest problems the ODF plugin has right now. In addition to this, there is almost no documentation. The project homepage has just a couple of test documents, which the manufacturer uses to demonstrate that ODF documents can look the same in Word and in OpenOffice.

Conclusions

An ODF plugin is a step in the right direction, and it would probably be fine for most home users. However, it is not stable enough for corporate use. The bugs that were revealed in this test can be disastrous in a professional environment, where documents are required to look the same, no matter what machine you view them on.

The plugin loses a lot of points for lacking a function for saving in ODF, and the fact that the plugin either needs the new Office 2007 or a compatibility framework ("BETA2 Preview") is likely to scare off some users.

This said, there does appear to be light at the end of the tunnel. Development on the plugin is progressing quickly; in the past couple of months, there have been a number of improvements released via the public CVS repository. It remains to be seen whether the developers will at least manage to eliminate the critical bugs before the Office 2007 release hits the shops. ■

INFO

[1] ODF plugin: *http://odf-converter.sourceforge.net*

[2] Office 2007 Format Compatibility Pack: *http://www.microsoft.com/office/preview/beta/converter.mspx*