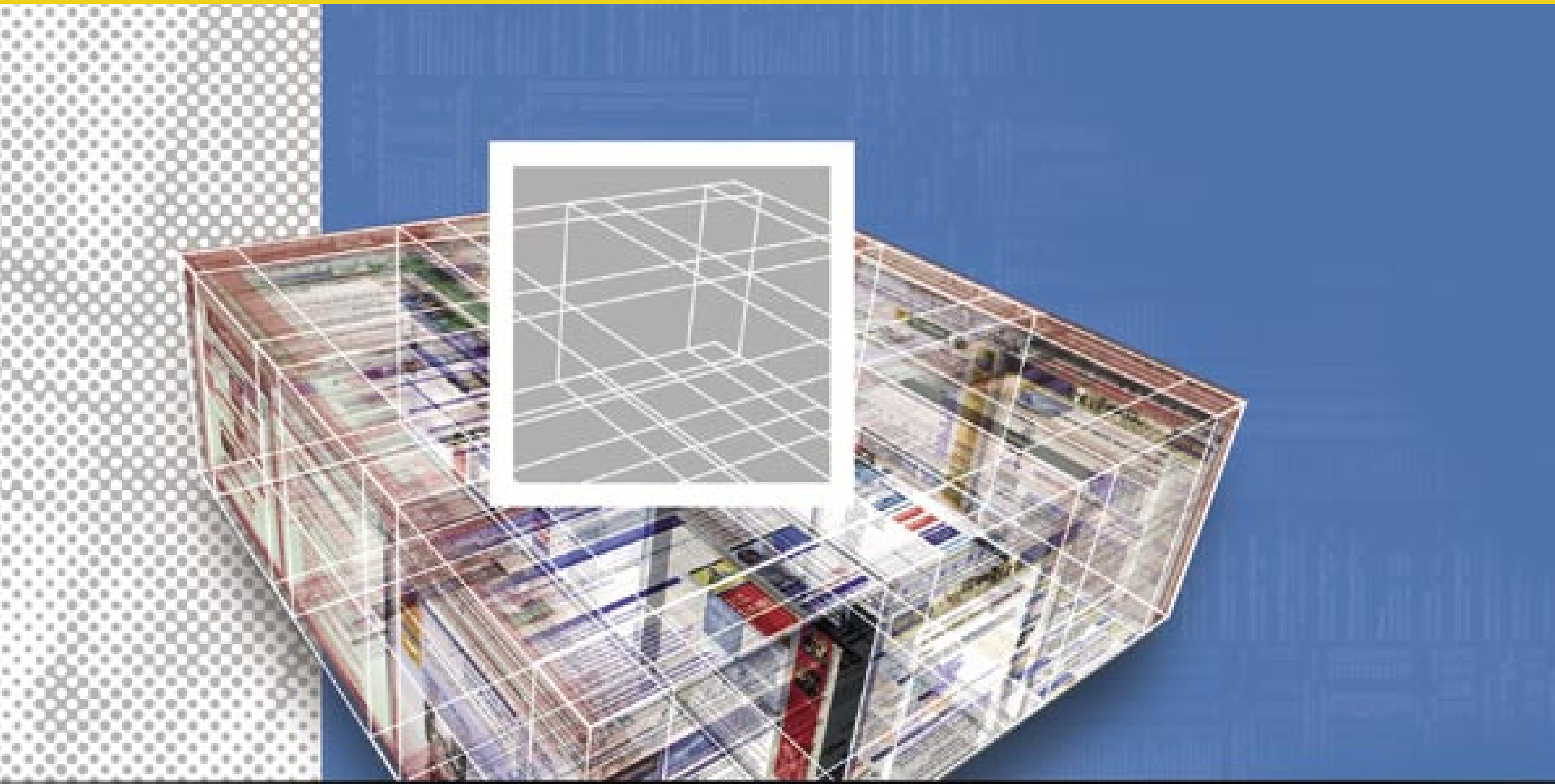


Document versioning with Subversion and OOoSVN

EYE ON CHANGE

Even if you are not a programmer, you've probably heard of Subversion, a powerful tool for managing changes to software projects. Although Subversion is designed primarily for software developers, it can be useful to mere mortals as well. **BY DMITRI POPOV**

Subversion [1] is a so-called *version control system*, which means it allows users to track changes to files. Many users do not realize you can use Subversion as a versioning tool for ordinary user files, such as word processing documents. Subversion also allows you to compare different versions of a document and roll back to a previous version. The tool is file-type agnostic, which means it can handle any file you throw at it – plain text files, Writer documents, spreadsheets, PNG graphics – and other common file types you will find on your Linux system.

Subversion Basics

Subversion is a rather advanced tool, but you do not need to understand all its intricacies to make use of it. In fact, you need to understand only three key concepts to get a grip on Subversion's basics: repository, checkout, and commit.

The key element of the system is a repository, a special directory that contains user documents and all the data that Subversion needs to track the modifications and versions of these documents.

Before you can start working with documents in a repository, you must check them out. The checkout process creates copies of the documents (often called working copies) in the specified location. For example, I have a repository called LXM_Articles containing all the articles I have written for *Linux Magazine*. When I need to edit one of them, I simply check it out to my home directory. Once a document is checked out, I can start working on it. When I'm done edit-

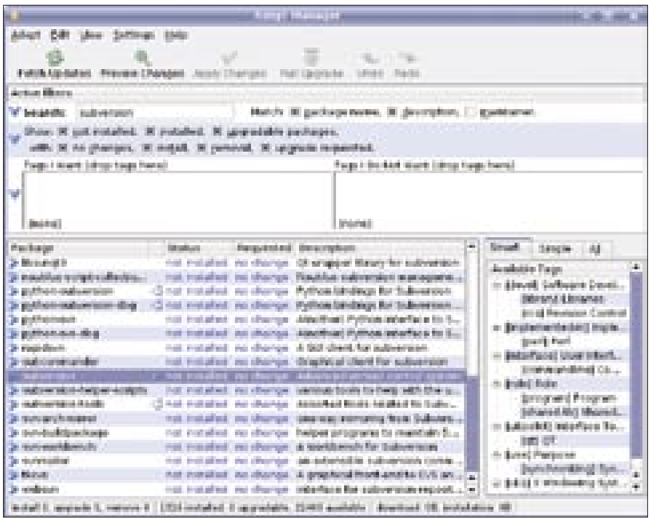
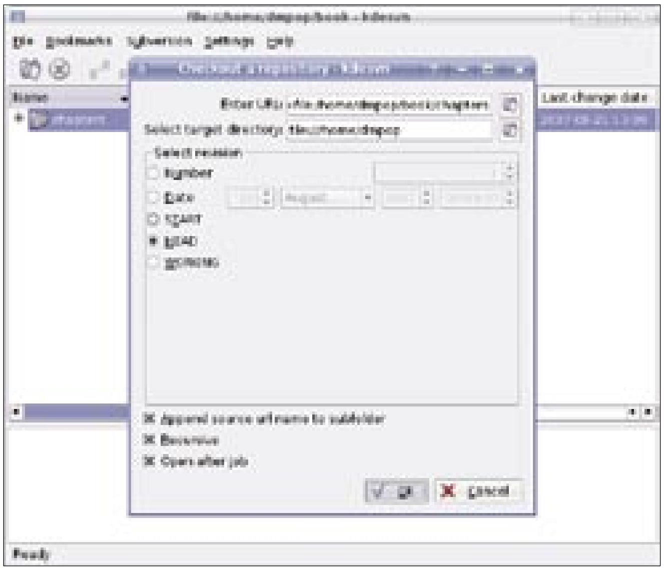


Figure 1: Installing Subversion on (K)Ubuntu is a matter of choosing the appropriate package.


file:///home/dmpop/book - kdesvn
File Bookmarks Subversion Settings Help
Name
chapters
Last change date
2007-08-21 13:08
Checkout a repository - kdesvn

Enter URL: file:///home/dmpop/book/chapters
Select target directory: file:///home/dmpop

Select revision
○ Number
○ Date
○ START
● HEAD
○ WORKING

☒ Append source url name to subfolder
☒ Recursive
☒ Open after job

Figure 2: Checking out a repository with KDESvn.

ing, I commit modifications made to the document back to the repository. At this stage, Subversion creates a new version of the document in the repository and records all the changes made to it. The good news is that all this is done transparently, with Subversion doing its job behind the scenes.

Once you've checked a document out, the important question is how often you should commit modifications to the repository. Because Subversion does not limit how many versions of a particular document you may have, it is completely up to you how often you perform commits. A rule of thumb is to commit modifications every time you make significant changes to your document that you might want to undo later.

Getting Started with Subversion

Now that you know the basics, you can put Subversion to use. First you must install the required software. Most Linux distributions have Subversion in their package repositories, so you can install the tool easily with your distribution's package manager (Figure 1). On Ubuntu, you can install Subversion with the *apt-get install subversion* command.

Besides Subversion itself, you should install a graphical SVN client, which makes it significantly easier to commit changes and manage document versions. If you are using KDE, KDESvn is your best bet. KDESvn is not only a powerful and easy-to-use SVN client, but it also integrates nicely with KDE so you can, for example, access Subversion commands in Konqueror. On some distributions, such as PCLinuxOS, you must install the kdesvn-kio-plugins package that provides this integration.

With all the pieces in place, it is time to create your first repository. As an example, you can create a repository for a book consisting of three directories for each book chapter, with a *.odt* document in each directory. First, create a temporary folder (e.g., *temp*) in your home directory. In the *temp* directory, create a *chapters* directory with three folders in it: *chapter1*, *chapter2*, and *chapter3*. Then create three Writer documents – *chapter1.odt*, *chapter2.odt*, and *chapter3.odt* – and move them into the respective chapter folders. Next, you must create an empty *book* repository in your home directory with the command

```
svnadmin create ~/book
```

and then move the contents of the *temp* directory into the *book* repository with the import command:

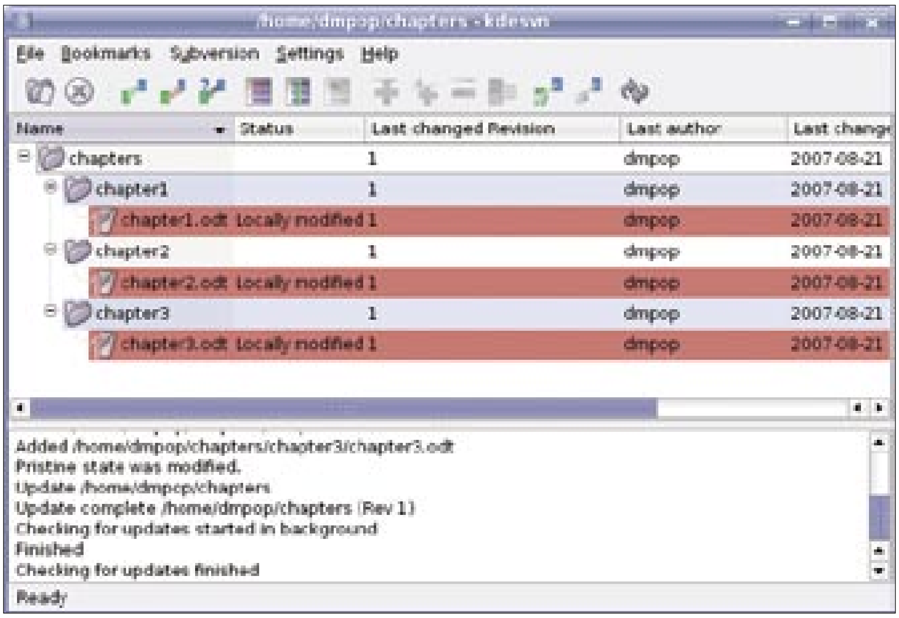
```
svn import ~/temp↩
file:///home/dmpop/book -m ↩
"Initial import"
```

Delete the *temp* directory and your first Subversion repository is ready to go. The next step is to check out one or several chapters you want to edit, which you can do from the command line or with the KDESvn client if you prefer the GUI way (Figure 2). Next, right-click on the *book* directory in Konqueror and choose *Open With* | *kdesvn* to check out all the chapters at once.

In KDESvn, right-click on the *chapters* folder and choose the *Checkout current repository* path command. Next, select the target directory (for example, your home directory) and press *OK*: You should then see the checked-out *chapters* folder in the specified directory. In a similar way, you can check out only a single chapter; to do so, select the desired chapter in KDESvn and use the described checkout procedure.

Once the documents are checked out, you can work with them in a normal manner. Use Konqueror or work directly from within KDESvn, which automatically switches to the checked-out directory. When you need to commit document modification, open the checked-out directory in KDESvn. The modified documents should be highlighted in red with their status set to "Locally modified" (Figure 3).

Selecting edited documents one by one and pressing the *Commit* button will


/home/dmpop/chapters - kdesvn
File Bookmarks Subversion Settings Help

Name	Status	Last changed Revision	Last author	Last change
chapters		1	dmpop	2007-08-21
chapter1		1	dmpop	2007-08-21
chapter1.odt	Locally modified	1	dmpop	2007-08-21
chapter2		1	dmpop	2007-08-21
chapter2.odt	Locally modified	1	dmpop	2007-08-21
chapter3		1	dmpop	2007-08-21
chapter3.odt	Locally modified	1	dmpop	2007-08-21

```
Added /home/dmpop/chapters/chapter3/chapter3.odt
Pristine state was modified.
Update /home/dmpop/chapters
Update complete /home/dmpop/chapters (Rev 1)
Checking for updates started in background
Finished
Checking for updates finished
```

Figure 3: KDESvn conveniently highlights the modified documents.

commit modifications. This allows you to enter short descriptions of the changes for each document. Alternatively, you can select the root directory (i.e., the *chapters* folder) and press the *Commit* button to commit all changes in one fell swoop. Because KDESvn integrates tightly with KDE, you can commit changes using a context menu. For example, to commit changes to the *chapter1.odt* document, right-click on it and choose *Actions* | *Subversion (kdesvn)* | *KDESvn Commit*.

OOoSVN: Using Subversion with OpenOffice.org

Once you have learned the ropes, Subversion is rather straightforward. However, if you want to use Subversion to keep track of your OpenOffice.org documents only, OOoSVN [2] is the perfect tool for the job. This extension adds a new menu containing all the essential Subversion commands in OpenOffice.org's main toolbar.

To install OOoSVN, download the latest release of the extension, choose *Tools* | *Extension Manager* in OpenOffice.org, press *Add*, point to the downloaded zip file, and press *OK*. Restart OpenOffice.org, and you should see the SVN menu in the main toolbar. To configure OOoSVN, choose *Place under version control* from the SVN menu. OOoSVN then prompts you to run a setup routine that sets up a repository and moves the currently opened document into it (Figure 4). Once OOoSVN is configured, you can use it to commit changes, and more. For example, the *Compare with older/newer version* selection allows you to compare two versions of the document using OpenOffice.org's own Compare feature (Figure 5).

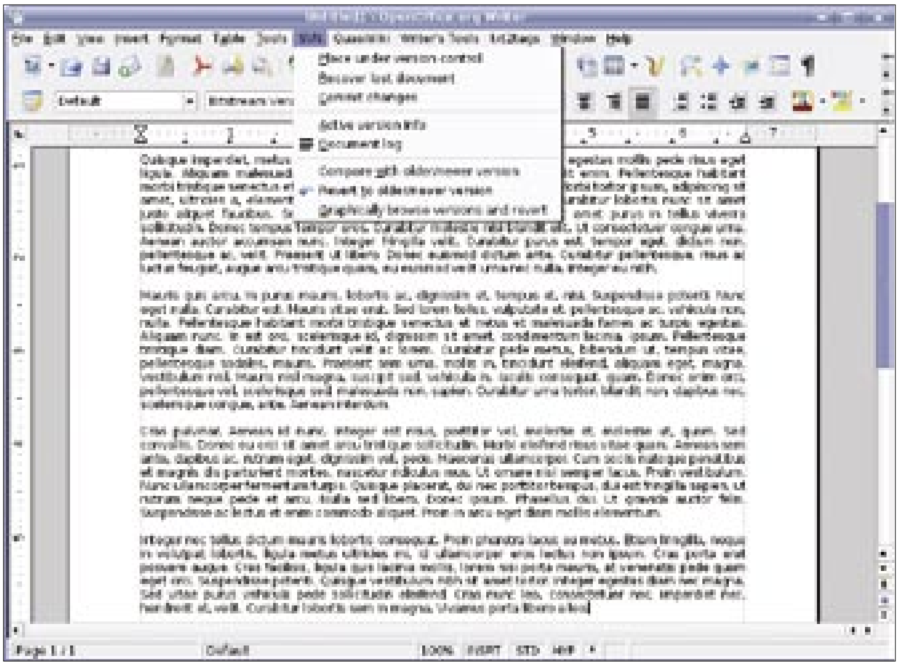

Place under version control
Recover lost document
Commit changes
Active version info
Document log
Compare with older/newer version
Revert to older/newer version
Graphically browse versions and revert

Figure 5: OOoSVN offers a few handy commands.

Final Word

In this article, I covered only the most basic features of Subversion, although you can do much more with the tool. A web front-end added to your Subversion installation will allow you to access your documents from anywhere. For a detailed description of how this can be done, check the How-To Geek's tutorial [3]. Also, you can add more users, thus turning your Subversion system into a powerful collaboration platform. No matter how you want to use Subversion, the online *Version Control with Subversion* book [4] will help you to get the most out of this software. ■

INFO

[1] Subversion: *http://subversion.tigris.org/*

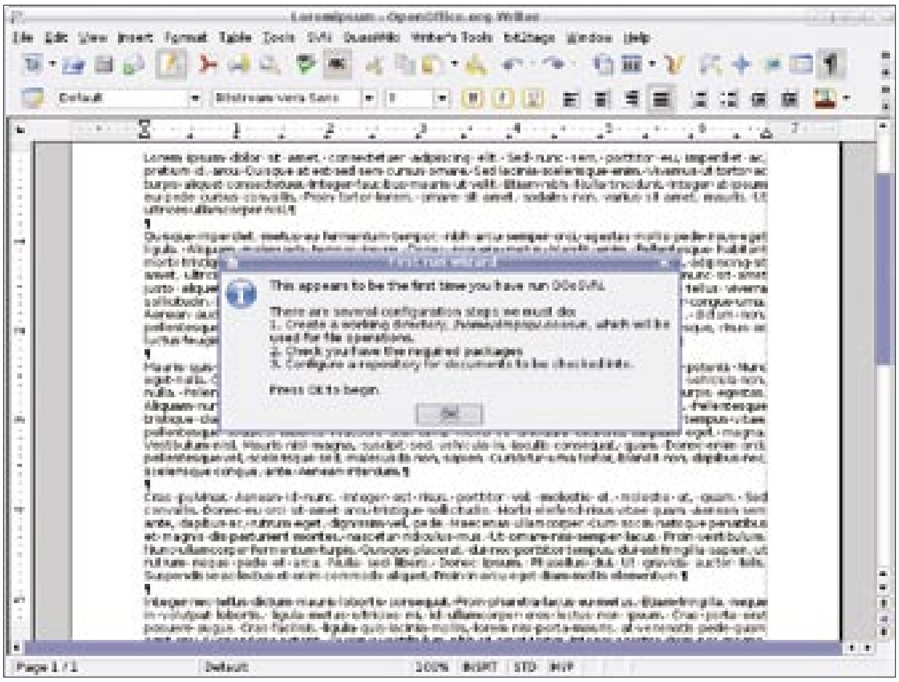
[2] OOoSVN: *http://sourceforge.net/projects/ooosvn/*

[3] Install Subversion with web access on Ubuntu: *http://tinyurl.com/2v6of8*

[4] Collins-Sussman, Ben, Fitzpatrick, Brian W., and Pilato, C. Michael. *Version Control with Subversion*. O'Reilly Media, 2004. *http://svnbook.red-bean.com/*

THE AUTHOR

Dmitri Popov holds a degree in Russian language and computer linguistics. He has been working as a technical translator and freelance contributor for several years. He has published more than 500 articles in Danish, British, US, and Russian magazines and websites.



First-run wizard

This appears to be the first time you have run OOoSVN.

There are several configuration steps we must do:
1. Create a working directory, [illegible], which will be used for file operations.
2. Check you have the required packages
3. Configure a repository for documents to be checked into.

Press OK to begin.

OK

Figure 4: OOoSVN includes a setup wizard that does all the heavy lifting for you.