

The **svninfo** package

Achim D. Brucker

brucker@freiburg.linux.de

<http://www.brucker.ch/>

April 17, 2007

Version 0.7.1

Abstract

This package is used to extract the revision and file information provided by the *Subversion* (<http://subversion.tigris.org>) and the *svk* (<http://svk.elixus.org>) revision control system.

1 Introduction

To maintain different versions of a document or to keep track on the development of one, the document is kept under a revision control system like *SCCS*, *CVS*, *Subversion*, or *svk*. This package assumes you are using either the *Subversion* or the *svk* system. To present the version information of a document, one needs to extract it from somewhere. For example, *svk* and *Subversion* offer several keywords that are expanded by the *Subversion* and *svk update* commands.¹ The most informative keyword is `Id` which contains a lot of useful information. This information is made available through this package. Additionally, the package allows one to extract also the information from the more specific keywords, e.g., `$Author$` or `$Date$`.

In this documentation we use *Subversion* in the examples; but the results should be similar if you are using *svk*. Note that you have to use the `svk` option if you use *svk*.

Moreover, one has to set the *Subversion* property for keywords expansion of that file, e.g., by executing:

```
svn propset svn:keywords "Id" svninfo.dtx
```

Please consult the manual of *Subversion* (e.g., *h*) or *svk* for further details about keyword expansion.

The information obtained from the expanded `Id` string are

¹The expansion of keywords has to be enabled manually, e.g., `svn propset svn:keywords "Id" svninfo.dtx`.

- the filename,
- the revision number,
- the date and time of the last *Subversion co* command, and
- name of the user who has done this action.

For each of these items a macro is defined. When *updating* a file, the *Subversion* keywords should be contained in the *updated* source. Further, one has to set the *Subversion* property for keyword expansion of that file, e.g.

```
svn propset svn:keywords "Id" svninfo.dtx
```

See the *Subversion* documentation for more details.

If you want to read about the implementation, put a % before `\Only-Description` in the code marked with `<*driver>` in the file `svninfo.dtx`.

2 User Interface

2.1 Options

Options for this package are:

svk Use `svk` for parsing the Id information generated by *svk*.

today Set the date of `\svnToday` to the date obtained by the *Subversion* information. If the `Id` string is not expanded (see below), the current date is used.

notoday Don't use the *Subversion* date for `\svnToday`.

revrange For multi-file documents, display minimum and maximum revision. Every file of the document must either include a `\svnInfo` or a `\svnKeyword $Revision:$` statement. This information needs two runs of L^AT_EX to be correct.

fancyhdr We use the `fancyhdr` package to fill the footline with *Subversion* information: filename, revision-number and date.

If the `fancyhdr` package is not already loaded, it will be loaded by this package. To use the footnotes, put something like:

```
\pagestyle{fancyplain}
\fancyhead[RE,LO]{\leftmark}
\fancyhead[LE,RO]{\thepage}
```

in the preamble.

nofancy We don't define the footline. The `fancyhdr` package is not loaded.

scrpage2 Use `scrpage2` instead of `fancyhdr`.

eso-foot Use `eso-pic` for showing a tiny info line at the bottom line of the paper.

margin Use `margin` for showing the version info in the margin area.

long If used together with `fancy` / `fancyhdr` the date string is printed in the long format as `\svnToday` returns it. If `fancy` / `fancyhdr` is not used, nothing.

short If used together with `fancy` / `fancyhdr` the date string is printed in the short format `YYYY-MM-DD`. If `fancy` / `fancyhdr` is not used, nothing.

final Don't define the footline. If you want the *Subversion* in the final document, please use `\usepackage{draft}`

draft If `nofancy` is not used, show a footline with *Subversion* information.

If the `babel` package is used, the `\svnToday` string is language dependent.

The default options are: *draft, today, long*. You may also configure the default fancy heading packages in the configuration file `svninfo.cfg`.

2.2 Commands

2.2.1 Defining the *Subversion* Id information

`\svnInfo` There is only one macro to define the information: `\svnInfo`. Since it must collaborate with *Subversion* it has an “unusual” calling sequence: The arguments are prefaced with a ‘\$’ and completed with a ‘\$’ (note the final space or carriage return). The following arguments are valid:

`\svnInfoId`

`\svnInfo$Id:$`

`\svnInfo$Id:$filerevYYYY-MM-DD$hh:mm:ss$owner$`

In the first two forms, *Subversion* has not expanded the `Id` string. This form will usually be used before the file is *checked in* the first time.

`YYYY-MM-DD` is a date, `hh:mm:ss` a time. Note: when you perform the *Subversion check out* command, the `Id` string is expanded by *Subversion* automatically.

To take effect, `\svnInfo` must follow the `\begin{document}`. If you are using multiple files for a document, you could place `\svnInfo` at the beginning of each file, so the information from the current file is obtained.

2.2.2 Accessing the *Subversion* Id information

There are the following macros to access the information provided by *Subversion*. If *Subversion* has not expanded the `Id` string, or no `\svnInfo` command is given, default values are returned.

`\svnInfoFile` • The name of the source file. Default: `--sourcefile--`.

<code>\svnInfoRevision</code>	• The revision number, of the <i>checked out</i> file. Default: <code>--revision--</code> .
<code>\svnInfoMinRevision</code>	• The minimum revision number of multi-file documents. Every file of the document must either include a <code>\svnInfo</code> or a <code>\svnKeyword \$Revision:\$</code> statement. This is information needs two runs of <code>L^AT_EX</code> to be correct. Default: <code>--minrevision--</code> .
<code>\svnInfoMaxRevision</code>	• The maximum revision number of multi-file documents. Every file of the document must either include a <code>\svnInfo</code> or a <code>\svnKeyword \$Revision:\$</code> statement. This is information needs two runs of <code>L^AT_EX</code> to be correct. Default: <code>--maxrevision--</code> .
<code>\svnInfoDate</code>	• The date in the form <code>YYYY-MM-DD</code> , when the file was <i>checked out</i> . Default: the current date.
<code>\svnInfoTime</code>	• The time, when the file was <i>checked out</i> . Default: <code>--time--</code> .
<code>\svnInfoOwner</code>	• The user name of the file owner. Default: <code>--owner--</code> .
<code>\svnInfoYear</code>	• The year <code>YYYY</code> of <code>\svnInfoDate</code> . Default: the current year.
<code>\svnInfoMonth</code>	• The month <code>MM</code> of <code>\svnInfoDate</code> . Default: the current month.
<code>\svnInfoDay</code>	• The day <code>DD</code> of <code>\svnInfoDate</code> . Default: the current day.
<code>\svnInfoLongDate</code>	• The date in the form of <code>\today</code> when the file was <i>checked out</i> . This is language depended. Default: the current date.
<code>\svnId</code>	• Mimics the behavior of the <code>\$Id\$</code> strings, i.e. it prints a string with a summary of the above described information.
<code>\svnToday</code>	• The date obtained from the <i>Subversion</i> information in the format <code>\today</code> is using.
<code>\svnInfoMaxToday</code>	• The date obtained from the latest <i>Subversion</i> revision in the format <code>\today</code> is using.

3 Examples

`\svnInfo` not expanded, no colon

`\svnInfo Id`

<code>\svnInfoFile</code>	<code>--sourcefile--</code>	<code>\svnInfoYear</code>	2007
<code>\svnInfoRevision</code>	<code>--revision--</code>	<code>\svnInfoMonth</code>	4
<code>\svnInfoMinRevision</code>	12	<code>\svnInfoDay</code>	17
<code>\svnInfoMaxRevision</code>	1168	<code>\svnInfoLongDate</code>	April 17, 2007
<code>\svnInfoDate</code>	2007-04-17	<code>\svnToday</code>	April 17, 2007
<code>\svnInfoTime</code>	<code>--time--</code>	<code>\svnInfoMaxYear</code>	2005
<code>\svnInfoOwner</code>	<code>--owner--</code>	<code>\svnInfoMaxDay</code>	08
		<code>\svnInfoMaxMonth</code>	04
		<code>\svnInfoMaxToday</code>	April 8, 2005

`\svnId --sourcefile-- --revision-- --time-- --owner--`

```

\svnInfo not expanded, having colon
\svnInfo $Id:$
    \svnInfoFile    -sourcefile-    \svnInfoYear    2007
    \svnInfoRevision -revision-    \svnInfoMonth    4
    \svnInfoMinRevision 12        \svnInfoDay    17
    \svnInfoMaxRevision 1168      \svnInfoLongDate April 17, 2007
    \svnInfoDate    2007-04-17    \svnToday    April 17, 2007
    \svnInfoTime    -time-        \svnInfoMaxYear 2005
    \svnInfoOwner    -owner-      \svnInfoMaxDay  08
                                   \svnInfoMaxMonth    04
                                   \svnInfoMaxToday    April 8, 2005
\svnId -sourcefile- -revision- -time- -owner-

```

```

\svnInfo standard version
\svnInfo $Id: svninfo.dtx 1168 2005-04-08 06:18:47Z brucker $
    \svnInfoFile    svninfo.dtx    \svnInfoYear    2005
    \svnInfoRevision 1168          \svnInfoMonth    4
    \svnInfoMinRevision 12        \svnInfoDay    8
    \svnInfoMaxRevision 1168      \svnInfoLongDate April 8, 2005
    \svnInfoDate    2005-04-08    \svnToday    April 8, 2005
    \svnInfoTime    06:18:47Z    \svnInfoMaxYear 2005
    \svnInfoOwner    brucker      \svnInfoMaxDay  08
                                   \svnInfoMaxMonth    04
                                   \svnInfoMaxToday    April 8, 2005
\svnId svninfo.dtx 1168 2005-04-08 06:18:47Z brucker

```

```

\svnInfo standard version without owner
\svnInfo $Id: svninfo.dtx 1168 2005-04-08 06:18:47Z $
    \svnInfoFile    svninfo.dtx    \svnInfoYear    2005
    \svnInfoRevision 1168          \svnInfoMonth    4
    \svnInfoMinRevision 12        \svnInfoDay    8
    \svnInfoMaxRevision 1168      \svnInfoLongDate April 8, 2005
    \svnInfoDate    2005-04-08    \svnToday    April 8, 2005
    \svnInfoTime    06:18:47Z    \svnInfoMaxYear 2005
    \svnInfoOwner                 \svnInfoMaxDay  08
                                   \svnInfoMaxMonth    04
                                   \svnInfoMaxToday    April 8, 2005
\svnId svninfo.dtx 1168 2005-04-08 06:18:47Z

```

```

\svnInfo standard version with
in the owner
\svnInfo $Id: svninfo.dtx 1168 2005-04-08 06:18:47Z brucker $

```

```

\svnInfoFile    svninfo.dtx          \svnInfoYear    2005
\svnInfoRevision 1168                \svnInfoMonth   4
\svnInfoMinRevision 12                \svnInfoDay     8
\svnInfoMaxRevision 1168             \svnInfoLongDate April 8, 2005
\svnInfoDate      2005-04-08          \svnToday       April 8, 2005
\svnInfoTime       06:18:47Z          \svnInfoMaxYear 2005
\svnInfoOwner      infsec“brucker     \svnInfoMaxDay   08
                                           \svnInfoMaxMonth 04
                                           \svnInfoMaxToday April 8, 2005
\svnId svninfo.dtx 1168 2005-04-08 06:18:47Z infsec“brucker

\svnInfo no leading $
\svnInfo Id: svninfo.dtx 12 2002-04-19 12:27:55 brucker $
\svnInfoFile    svninfo.dtx          \svnInfoYear    2002
\svnInfoRevision 12                  \svnInfoMonth   4
\svnInfoMinRevision 12                \svnInfoDay     19
\svnInfoMaxRevision 1168             \svnInfoLongDate April 19, 2002
\svnInfoDate      2002-04-19          \svnToday       April 19, 2002
\svnInfoTime       12:27:55           \svnInfoMaxYear 2005
\svnInfoOwner      brucker            \svnInfoMaxDay   08
                                           \svnInfoMaxMonth 04
                                           \svnInfoMaxToday April 8, 2005
\svnId svninfo.dtx 12 2002-04-19 12:27:55 brucker

```

4 The generic *Subversion* keyword interface

In addition to the already presented `\svnInfo` macro which is specially built for analyzing the `SubversionId` information we also provide a generic interface via `\svnKeyword`. This interface can be used to extract information from the remaining keywords:

- `$Date$`: The date of the last commit with changes. This information can also be accessed using the keyword `$LastChangedDate$`.
- `$Revision$`: The revision of the last commit with changes. This information can also be accessed using the keyword `$LastChangedRevision$` or `Rev`.
- `$Author$`: The author of the last commit with changes. This information can also be accessed using the keyword `$LastChangeBy$`.
- `$HeadURL$`: The the full URL to the latest version of the file in the repository. This information can also be accessed using the keyword `URL`.

These keywords can be accessed using the `\svnKeyword` macro. This macro accepts a single argument that should be of the form `$\langle Keyword \rangle$` (unexpanded) or `$\langle Keyword \rangle: \langle value \rangle$` (expanded), e.g.:

`\svnKeyword $Author: brucker $`

The `\svnKeyword` updates the corresponding *Subversion* Id information, e.g., our example updates `\svnInfoOwner`. In more detail:

- | | |
|-------------------------------|---|
| <code>\svnInfoDate</code> | • <code>\svnKeyword \$Date:\$</code> updates the date and time information, e.g., <code>\svnInfoDate</code> , <code>\svnInfoMonth</code> , or <code>\svnInfoTime</code> . |
| <code>\svnInfoMonth</code> | |
| <code>\svnInfoTime</code> | • <code>\svnKeyword \$Revision:\$</code> updates <code>\svnInfoRevision</code> . |
| <code>\svnInfoRevision</code> | • <code>\svnKeyword \$Author:\$</code> updates <code>\svnInfoOwner</code> . |
| <code>\svnInfoOwner</code> | |
| <code>\svnInfoHeadURL</code> | • <code>\svnKeyword \$HeadURL:\$</code> updates <code>\svnInfoHeadURL</code> . |

Note, the parser used by `\svnKeyword` is not that powerful than the one used by `\svnInfo`. Thus it is advisable to use `\svnKeywordId` for the `Id` keyword and `\svnKeyword` only for the keywords described in this section.

Further, the `\svnKeyword` macro does not reset all keyword definitions globally, i.e., you can combine the `$Revision$` of one file, with the `$Author$` of another file. Please take care of that!

5 Acknowledgments

This package is based on `rcsinfo` from Dr. Jürgen Vollmer. The `rcsinfo` package is distributed via the CTAN archives (`'macros/latex/contrib/rcsinfo'`).

The code for parsing generic *Subversion* keywords was taken from the `svn` packages written by Richard Lewis. The `svn` package is distributed via the CTAN archives (`'macros/latex/contrib/svn'`).

Further I want to thank all the people that reported bugs and ideas for improvements, often including patches: Andreas Haller, Christophe Jacquet, Jami Lawrence, Stefan Mann, Nigel Metherringham, Igor Nikolic, Michael Niedermair, Heiko Oberdiek, Bernd Raichle, Jörg Sommer, Arnout Standaert, Daniel Trümper, Thomas Weber, and Uwe Ziegenhagen.

6 Copyright and License

Copyright (©) 1995 Dr. Jürgen Vollmer

Copyright (©) 2003–2007 Achim D. Brucker brucker@freiburg.linux.de

This program can be redistributed and/or modified under the terms of the L^AT_EX Project Public License Distributed from CTAN archives in directory `'macros/latex/base/lppl.txt'`; either version 1 of the License, or any later version.

Change History

v0.1	General: Initial version, mainly a modified version of the rcsinfo package. 1	svnKeyword for defining arbitrary svn keywords (e.g. <i>Author</i> :). 1
v0.2	General: Removed L^AT_EX2HTML build-dependency. 1	v0.6 General: Added svnInfoMinRevision and svnInfoMaxRevision together with the option revrange . Also added support for \ in keywords. 1
v0.2.1	General: Added svnId 1	v0.7 General: Added option margin for showing version info in the margins of the document; added option svk for supporting the keyword expansion of svk ; added command svnMaxToday ; and fixed the use of fancyhdr for single-sided documents. 1
v0.3	General: Added support for \$Id:\$ keywords without owner information. 1	
v0.4	General: Added new package options draft , final , eso-foot , and scrpage . Fixed problem with ‘_’ in file names. 1	
v0.5	General: Removed support for L^AT_EX2HTML (it is non-free software and also seems to be unmaintained) and added new command	v0.7.1 General: Fixed svnMaxToday and introduced svnInfoMaxToday as an alternative name. Improved documentation 1

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

S		
\svnId	4	\svnInfoMaxToday 4
\svnInfo	3	\svnInfoMinRevision 4
\svnInfoDate	4, 7	\svnInfoMonth 4, 7
\svnInfoDay	4	\svnInfoOwner 4, 7
\svnInfoFile	3	\svnInfoRevision 4, 7
\svnInfoHeadURL	7	\svnInfoTime 4, 7
\svnInfoLongDate	4	\svnInfoYear 4
\svnInfoMaxRevision	4	\svnKeyword 6
		\svnToday 4