

Onzen

Version 0.09

© 2006-2017 Torsten Rupp

Table of Contents

Overview.....	4
Features.....	4
Requirements.....	4
Installation.....	5
Packages.....	5
Usage.....	5
Repositories.....	6
Repository lists.....	6
New repository.....	7
Check-out repository.....	7
Open repository.....	7
Close repository.....	7
Basic functions.....	7
Edit files.....	7
Commit.....	7
Add.....	8
Remove.....	8
Revert.....	8
Revisions.....	8
Diff.....	8
Changed files.....	8
Annotations.....	8
Patches.....	8
Create patch.....	8
Review patch.....	8
Preferences.....	8
Appendix.....	9
Onzen command line options summary.....	9
Compilation.....	9
Development.....	9
Frequently asked questions.....	9
Contact and bug reports.....	10
License.....	10
References.....	11

Illustration Index

Figure 1: Onzen main window.....	6
Figure 2: Repository list dialog.....	7

Index of Tables

Table 1: Compilation packages.....	9
------------------------------------	---

Overview

Onzen is a graphical front end for the revision control systems Concurrent Version System [CVS], Apache Subversion [SVN], Mercurial [HG], and GIT [GIT]. It represent the files managed by the RCS in a tree-view with detailed information and offers functions to e. g. update, commit, add, remove, rename, revert, create diffs, view files and others.

Features

- support revision control systems Concurrent Version System [CVS], Apache Subversion [SVN], Mercurial [HG], and GIT [GIT].
- show files in a tree-view with detailed information like status, revision, branch in multiple tabs
- update, commit, add, remove, rename, revert, and view files,
- show diff of files in a two-column view with highlighting of added, removed, and changed lines and lines which only have white-space changes,
- show revisions in a tree-view with details information of revision, author, commit date and commit message,
- show list of changed files,
- show annotations for each line in a file,
- history of commit message, broadcasting commit messages to other running Onzen instances in local network,
- create auto-summary line from multi-line commit message via user definable patterns,
- create and manage patches in an internal database,
- send patches for review either via mail or to a review server like ReviewBoard [ReviewBoard],
- open files via Mime type or file name extension with user definable editor program,
- create local directories, delete and rename local files,
- find files by name and content,
- user definable shell commands
- support RCS specific features:
 - view tree with branches (CVS),
 - master repository (HG),
 - push/pull command (HG),
 - forest extension (HG),
 - patch queues (HG)
- preferences to set colors, keyboard short cuts, editors, Mime types, shell commands. Optional encryption of mail and login passwords.

and more!

Requirements

To execute Onzen you need:

- installed command line tools for CVS, SVN, HG, and/or GIT
- JRE 1.6 [JRE, JDK]

Optional tools for Onzen:

- external text editor
- mail command
- diff command

Onzen is either available as a binary package for Linux, Solaris, MacOS X, and Windows or as source code for compilation. If you are not familiar with using the Java Development Kit (JDK) it is recommended to use the binary packages. Please check the website

<http://www.kigen.de/projects/onzen/index.html>

for available binary packages.

To compile Onzen you need:

- JDK 1.6 [JRE, JDK]

For more details to compile Onzen by yourself please see chapter Compilation, page 9.

Installation

Packages

Onzen is installed by simply decompress the delivered package file (compressed TAR or ZIP file):

Sub-directory	Platform
onzen-<version>-linux.tar.bz2	Linux
onzen-<version>-solaris.tar.bz2	Solaris
onzen-<version>-macosx.tar.bz2	MacOS X
onzen-<version>-windows.zip	Windows

Usage

Onzen is started with:

onzen

resp. with:

onzen.bat

or by simply clicking on the JAR file icon on the desktop resp. selecting the entry in the start menu.

Note: To execute Onzen the Java Run time Environment 1.6 [JRE, JDK] must be installed. The front end use SWT [SWT]. The required SWT libraries for Linux (32- and 64bit), Solaris (32bit), Mac OS X (32- and 64bit), and Windows (32- and 64bit) are included in the Onzen JAR files. Please use the 32bit versions on a 32bit system and the 64bit versions (file name contain "64") on 64bit systems.

Onzen can be called with some options. The general syntax is:

onzen [<options>] [<repository list>]

Onzen options are prefixed by either - (short name) or -- (long name) and may occur at any position. If an option is specified more than one time all values (for options which may occur multiple times) resp. the last value (for options which only can occur one time) are used.

The repository list name is the first non-option argument and specify the name of the repository list which is shown in the tabs in Onzen.

The main window look like in example figure 1: Onzen main window.

Note: Most input fields and buttons have a short online-help as tool tips. The tool tips are shown when you keep the mouse pointer over some field or button for a short time.

Note: To hide the shell window which is opened on Windows, create a Windows link and select in the properties settings of the Windows link the option “minimized”.

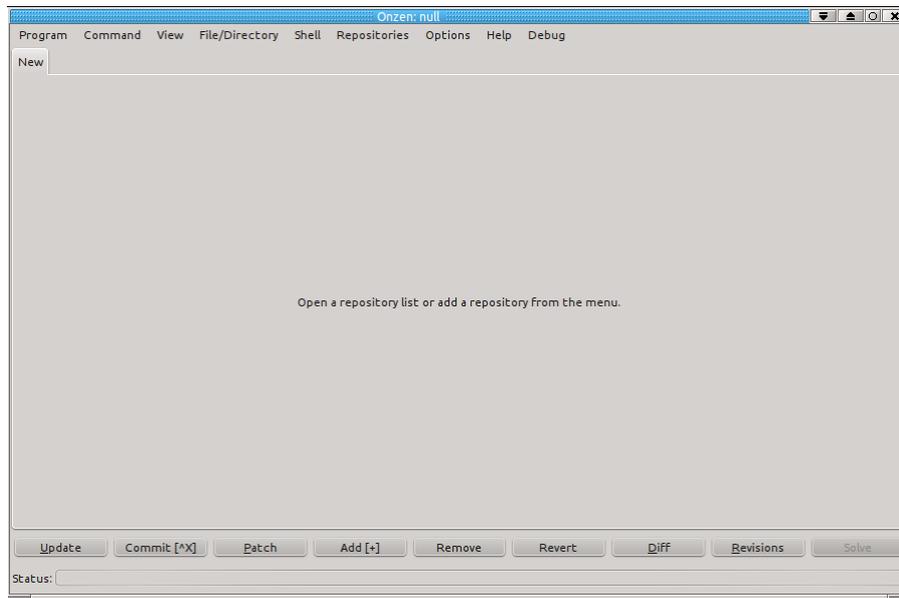


Figure 1: Onzen main window

Repositories

Repository lists

When Onzen is started the repository list dialog is opened for selecting a repository list by its name. In the repository list dialog an existing list can be selected or a new list can be created. A repository list contain a set of repositories which are organized in tabs in Onzen. For each tab the repository settings (path, tests, mail settings, patch review settings etc.) and the last state of the file tree view are stored. When the repository list is opened again these settings are restored.

Figure 2: Repository list dialog show the repository list dialog.

Note: The repository list can be edited later with the commands under menu *Program* in Onzen.

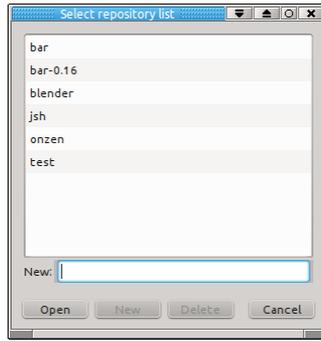


Figure 2: Repository list dialog

New repository

TODO

Check-out repository

TODO

Open repository

With the command *Open repository...* an existing repository can be opened and added to the current repository list. In the file select dialog the path name to the existing repository can be selected.

For each opened repository a new tab is added in Onzen. Single repositories can be opened multiple times and also sub-directories of a repository can be opened in a new tab.

Close repository

TODO

Basic functions

Edit files

Files can be opened for edit by a double-click on the file name. If the Mime type is known Onzen open the file with the associated editor program. If still no specific editor program is associated, a dialog open and an external program can be selected and a file type association can be created, thus all files with the same Mime type are opened with the specific editor program. If the Mime type cannot be detected, a file name pattern can be specified as an alternative approach to detect which editor should be used (e. g. required on Windows).

Note: The editor settings can be edited in the preference dialog.

Commit

With the *Commit...* command the currently selected files are prepared to be committed and the commit dialog is opened. In the commit dialog the changes of the selected files and the files to commit are listed. In a text input field the commit message can be entered. From the commit

message history list a commit message from a previous commit can be selected and reused in the new commit.

Add

TODO

Remove

TODO

Revert

TODO

Revisions

TODO

Diff

TODO

Changed files

TODO

Annotations

TODO

Patches

Create patch

TODO

Review patch

TODO

Preferences

TODO

Appendix

Onzen command line options summary

The following sections list the Onzen command line options.

--help, -h

Show help to Onzen.

--debug

Start Onzen in debug mode. If there is some serious error a Java stack trace is printed.

Compilation

To compile Onzen by yourself you need a Java Development Kit [JRE, JDK] environment and make. The following list the required packages to compile Onzen:

Package	Requirement	Comment
openjdk-6-jdk ¹	mandatory	Java Development Kit
make	mandatory	make program
ant, launch4j	optional	tool to pack Onzen into a single-binary (only needed when a single-binary exe-file should be created on Windows) ²

Table 1: Compilation packages

To compile Onzen usually the command sequence:

make

make install

will create all Onzen binary packages.

Development

The Onzen sources are currently hosted on:

<http://developer.berlios.de/projects/onzen>

There you may find the most recent development version of the sources and additional developer information.

Frequently asked questions

If you try to use Onzen or recompile Onzen by yourself and you run into some problem, please check the following questions and answers list before creating a bug report. Some problems are known and are not caused by Onzen.

¹ Instead of the OpenJDK package a standard JDK package may be usable, too.

² Currently ant/launch4j is not fully supported in the BAR make file.

1. *Onzen does not start on my system. I get the error “No more handles [gtk_init_check() failed]”. What is wrong?*

Probably Onzen cannot connect to the running X11 server. Either the environment variable DISPLAY is not set correctly or connections to the X11 server are not allowed. Please check the variable and execute `xhost <host name>` to allow an X11 server connection for `<host name>`. Further details to this problem can be found here:

http://www.eclipse.org/tptp/home/downloads/releasenotes/releasenotes4_3_0.html

2. *Instead of the SWT JAR included in BAR I like to use another one I already have. Onzen is starting, but some of the buttons does not work properly. What is wrong?*

This is probably a bug in the SWT JAR you are using in combination with the installed GTK libraries on your system. Probably SWT 3.5.x and GTK 2.18 or newer versions seems not to work together anymore. I could avoid this problem by using SWT 3.6 or newer.

Contact and bug reports

If you find a bug in Onzen, please send me a bug report including the following information:

- version number of Onzen
- description of your system environment (Linux distribution and version)
- if you used a pre-compiled version and which package or a self-compiled source-version
- steps to trigger the bug
- what is going wrong

In case of a fatal error where Onzen is crashing, please please start the program with the option `--debug`. With this option a Java stack trace should be printed with detailed information of the crash.

Please send the report to:

torsten.rupp@gmx.net

I will try to reproduce the problem and fix it as soon as possible.

License

Onzen and all included files are under the GPL version 2. The full GPL version 2 license text can be found here:

<http://www.gnu.org/licenses/gpl-2.0.html>

Onzen is using SWT [SWT]. SWT is under the "Eclipse Public License" which can be found here:

<http://www.eclipse.org/org/documents/epl-v10.php>

References

CVS <http://savannah.nongnu.org/projects/cvs>
GIT <http://git-scm.com/>
HG <http://mercurial.selenic.com/>
JRE, JDK <http://www.java.com>
launch4j <http://launch4j.sourceforge.net>
ReviewBoard <http://www.reviewboard.org>
SVN <http://subversion.apache.org/>
SWT <http://www.eclipse.org/swt>

Alphabetical Index

--.....9	-h.....9
--help.....9	