

Eric 4

as
Python 2.x
Integrated Development Environment

S-PM 120200



Web Browser

Technical Report

Copyright Page

SP-M 120000

Current edition PDF file, at URL: [Eric_Deliver] <http://www.box.net/shared/k64yenrpey>
under the “Creative Commons License”



“Eric 4 — Web Browser Technical Report” by Pietro Moras (E-mail: Studio-PM@hotmail.com) is licensed
under a Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License.
Based on a work at [Eric IDE] <http://eric-ide.python-projects.org/index.html>.

No commercial uses, No modifications allowed; Jurisdiction International, Format Text

This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/3.0/> or send a letter to Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

Disclaimer The information in this document is subject to change without notice. The author and publisher have made their best efforts about the contents of this book, nevertheless the author and the publisher make no representation or warranties of any kind with regard to the completeness or accuracy of the contents herein and accept no liability of any kind including but not limited to performance, merchantability, fitness for any particular purpose, or any loss or damages of any kind caused or alleged to be caused directly or indirectly from this book.

Brand Names Brand names such as Linux, Windows, Python are assumed to be universally known, and are here used in full respect of their respective owners.

Planned edition On paper, under traditional copyright

Published by [not yet—just appointed] Town & Country Reprographics, Inc.
230 N Main St Concord, NH 03301 (U.S.A.)

All rights reserved No part of this book may be reproduced or used in any form or by any means, such as: graphic, electronic, or mechanical, including photocopying, recording, videotaping, or information storage and retrieval systems, without written permission of the publisher. The media possibly accompanying this book contain software resources to be considered as an integral part of the same book, and therefore subject to the same rules.

- = -

Foreword

S-PM 120000

Dear Reader,

This document is intended as a handy booklet focused on the Web Browsing feature of the Eric Python IDE, conceived as a member of the collection of Technical Reports built around the central volume “Eric 4 as Python 2.x Integrated Development Environment – Technical Report” [see]. This whole collection is the result of an ambitious work of test and documentation, aimed at offering the Eric's users a comprehensive, reliable and agile reference about this extraordinary development tool.

As such, for all related subjects of general scope, introductory Foreword included, you're invited to refer to the named central volume of this collection.

That said, we deem anyhow worth recalling here too the basic principles characterizing this whole work. In summary:

- All that you'll read here is based on actual experience of use, being the result of a test & documentation project carried on in fair collaboration with the Eric producer but, nevertheless, in total independence from him.*
- Besides positive technical documentation and instructions, also our possible doubts, perplexities and difficulties have been here fairly reported. Of course, in our intentions, all intended as an opportunity for a hopefully fruitful technical dialogue both with users and the producer, typically via the Eric Mailing List [see], or privately too.*
- The technical dialogue is strongly encouraged, to the obvious benefit of all involved parties, and on which we'll be glad to rely for the future editions of this work. Thank you.*

See you.

The Author

- = -

Essentials

Essential reference data, so to precisely identify the product we are talking about, its host environment and running action.

S-PM 120200

Version Reference

Nr	Item	Version	Note
1	Windows XP Pro Windows Vista Ultimate	5.1 (SP 3) 6.0 (6002:SP 2)	Host Operating System
2	\Python27	2.7.1	Host Directory
3	PyQt4	4.9-1 (Qt 4.8.0)	GUI Library
4	Eric4	4.5.0 (r4278)	IDE Application

Identifier “Eric4 Web Browser”, short form: “E4-WB”

Prerequisites Same as for the Eric4 host main product [see: “Eric4 – Technical Report”].

Run Eric4 menu command Help > Helpviewer..., or double-click the executable `eric4_webbrowser.bat` file located into the `\Python2x` installation directory [see].

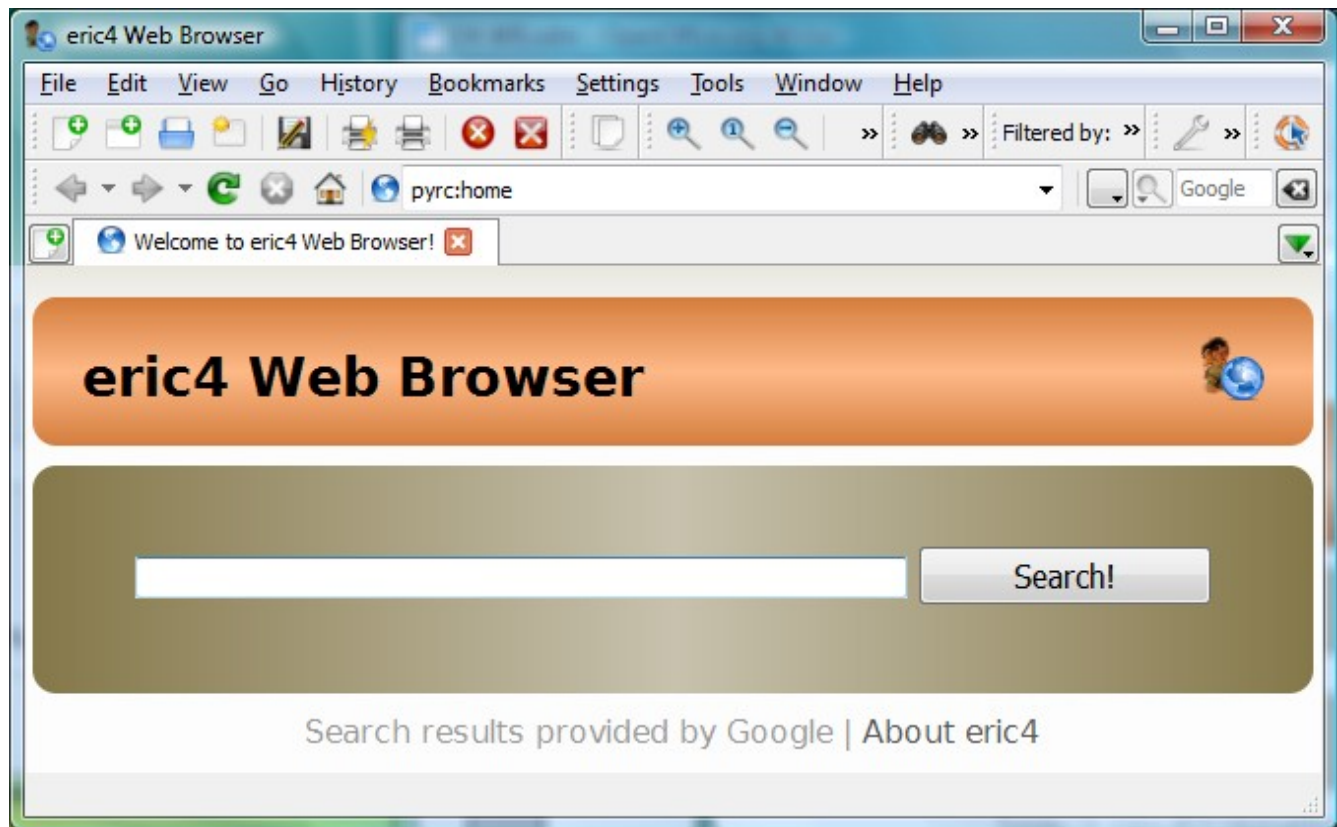
For an extensive technical reference of this action see in Appendix the Start Command section, and then, even more important, the Primary Settings & Adjustments section.

- = -

1. Introduction & Generality

SP-M 120000

In essence the Eric4 Web Browser, as declared in its `Help > About` menu command [see], is a help documentation files and HTML pages viewer, available to be run both as a utility tool embedded into the very Eric 4 IDE, or as a standalone application. Even if not comparable with other most renown web browsers available on the market, still it is a full-fledged program, well suited to complete, also under this aspect, the Eric's functionality.



Here see its initial “welcome” home page¹.

- -

¹ And in case you wander about this “pyrc:home” URL, it's for: PyRC - Python IRC Client, an extensible IRC client written in Python.

Why an Eric (4) Web Browser?

The Eric IDE program has clearly been designed to fully honor the “*Integrated*” attribute contained in its definition, in the sense that it is really meant to grant all tools a s/w development activity will require, web browser included. And this is “*why*” an Eric (4) web browser. Without, obviously, any intention of representing a real alternative to any of the major web browsers available on the market.

Of course this stated goal, both limited and ambitious, has some specific consequences, that we'll try to list hereafter under the headings *Virtues* and *Limits*.

Virtues

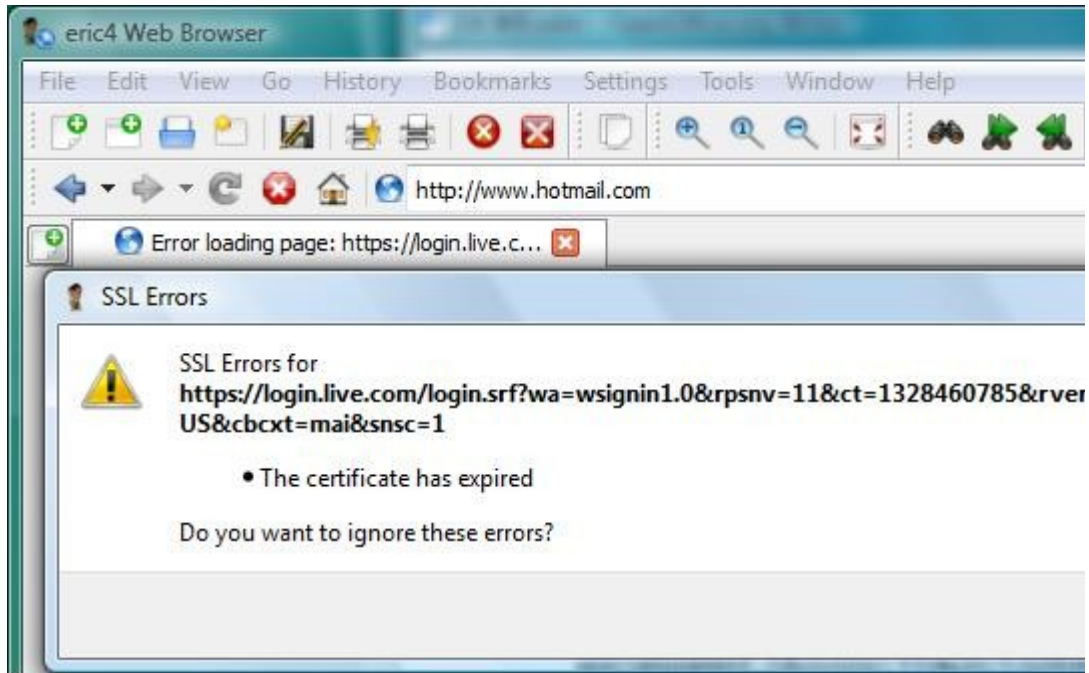
- It's there, immediately available and handy for browsing and searching all documentation files and sites of interest for Eric.
- It is meant to be precisely and suitably configurable for the stated task within Eric [*see menu command: Settings*].
- It's particularly well suited to examine Qt Help files, equipped with “Table of Contents” and “Index Table”, as hereafter described at the menu commands: Go > Sync with Table of Contents, and Window > Search [*see*].

Limits

- Do not expect to use it out of its natural scope, in place of ordinary browsers. If you do, you may encounter such obstacles:



Or:



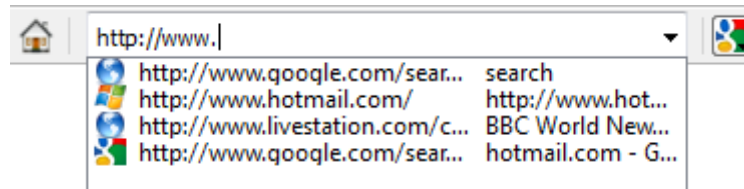
For which no solution can be reasonably expected within this very Eric browser.

- -

User Interaction

The main way for interacting with this program is through its set of menu commands, as will be thoroughly described with the next chapter. Some special actions, though, cannot be achieved through ordinary menu commands, but only in a different way, as hereafter described in detail.

Web URLs Entry



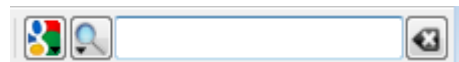
Combo-box field, where to enter the URL² of the web sites to browse.

Remarks

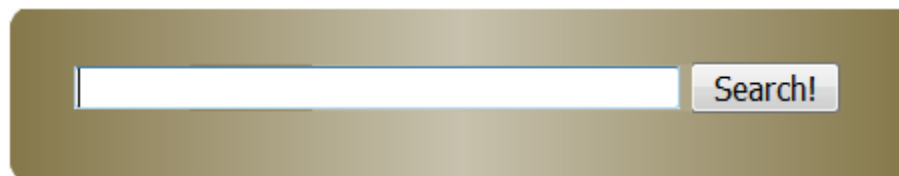
- E4-WB makes a distinction between browsing the contents of a web site, as here described, and the contents of a file. For the latter refer to File > Open File, at the next chapter 2. Menu Commands [see].

--

Web Search



Here see the controls where to set and use the web search tools [see also menu command: Settings > Configure Search Engines...].



Search results provided by Google | [About Eric](#)

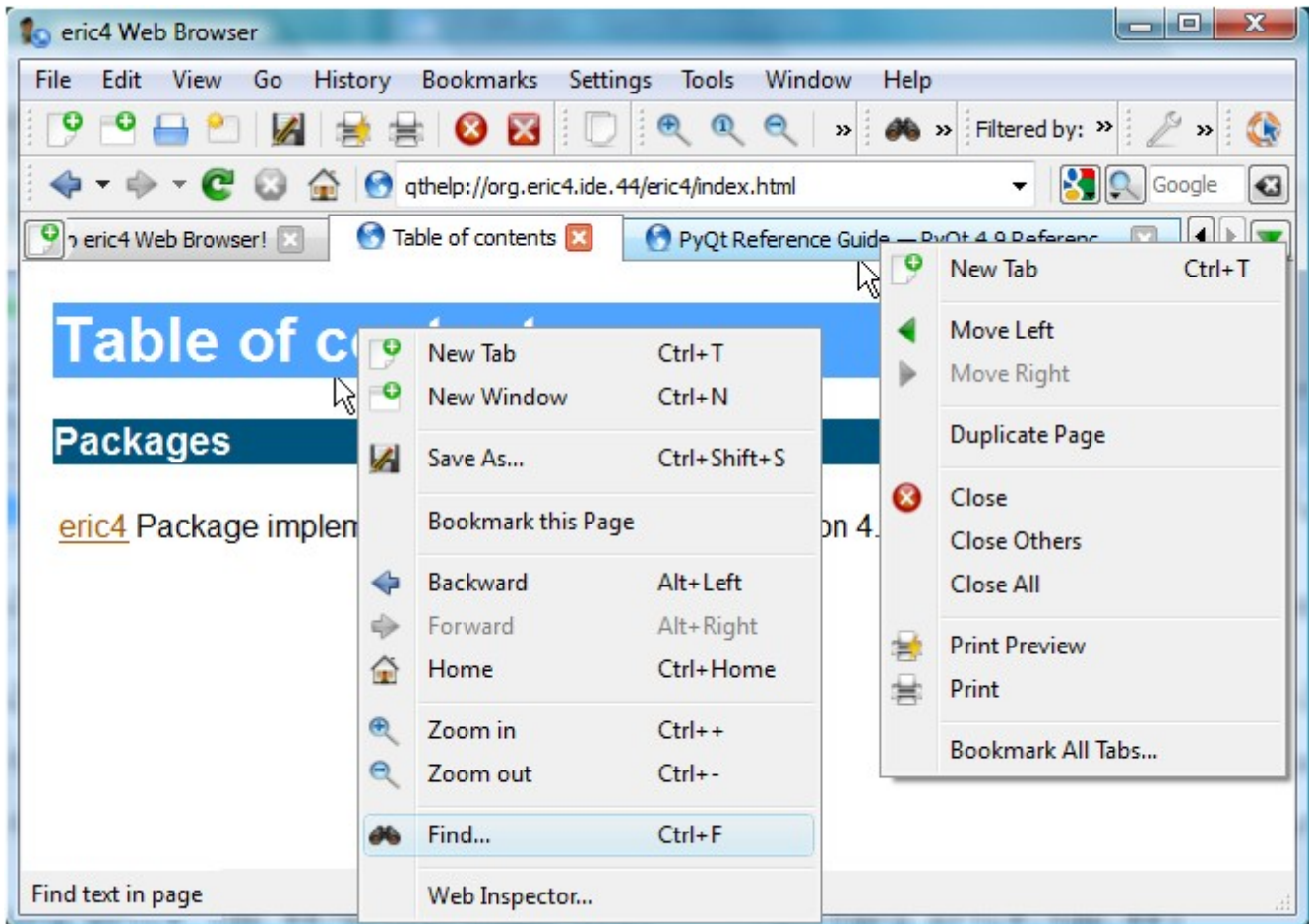
Web search can be performed on this initial form field too, where the “[About Eric](#)” is simply a link to the Eric web site home page³.

² URL – “Uniform Resource Locator”, for specifying addresses on the Internet.

³ Currently at: <http://eric-ide.python-projects.org/>

Context Menu Commands

Although the Menu Bar is the main way to interact with E4-WB, as thoroughly described with next chapter 2. Menu Commands [see], it isn't the only one. Indeed, other context-dependent Shortcut Menu⁴ can be get directly right-clicking items on the scene. As an example, see these two context menus obtained right-clicking different items on a web page:



Note also that the currently selected command (here: `Find...`) is conveniently described on the bottom information bar.

As a rule all such commands are available the ordinary way too, on the main menu⁵, but here you have the advantage of getting a selected set of them, context dependent. Therefore the user is encouraged to explore and use also this convenient way of interaction, right-clicking on different items on the scene.

- = -

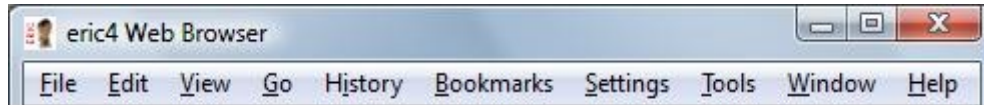
⁴ *Shortcut Menu*, Windows-defined as: A menu that you can open by using a right-click equivalent on items.

⁵ And that's why they will be described only once, at the dedicated chapter 2. Menu Commands [see].

2. Menu Commands

S-PM 120000

Reference section of all Eric4 Web Browser menu commands available for user interaction.



All menu commands will be hereafter listed and described orderly by their full names, without making also reference to their possible shortcut keys, nor command bar buttons, or context menus, all left to the direct user's interactive experience.

- -

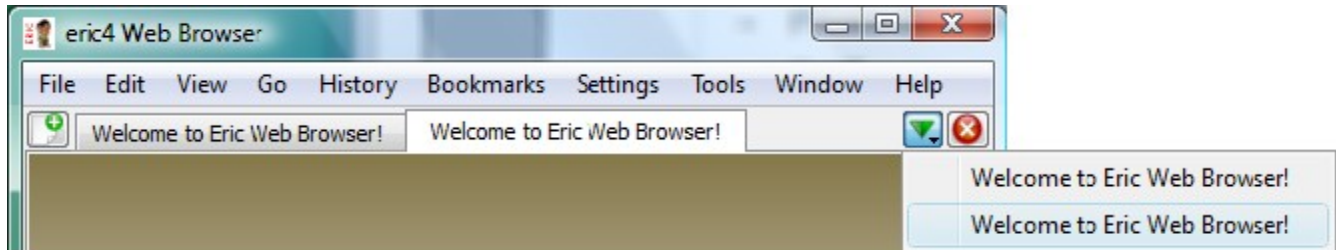
Menu Command: **File**

- New Tab
- New Window
- Open File
- Open File in New Tab
- Save As...
- Import Bookmarks...
- Export Bookmarks...
- Print Preview
- Print
- Close
- Close All
- Private Browsing
- Quit

- -

File > **New Tab**

Designed to create a new browse tab-form in display on the E4-WB workspace area.

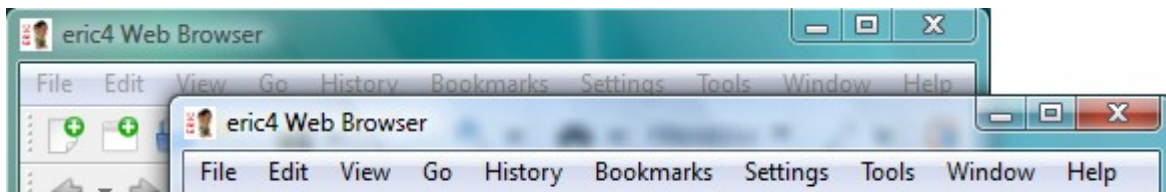


Browse tab-forms can then be closed via menu command `close` [see]. All but one, anyhow kept open in form of the initial “welcome” home page.

--

File > **New Window**

Designed to create a new full-fledged E4-WB window instance, under the same executing session.



--

File > **Open File**

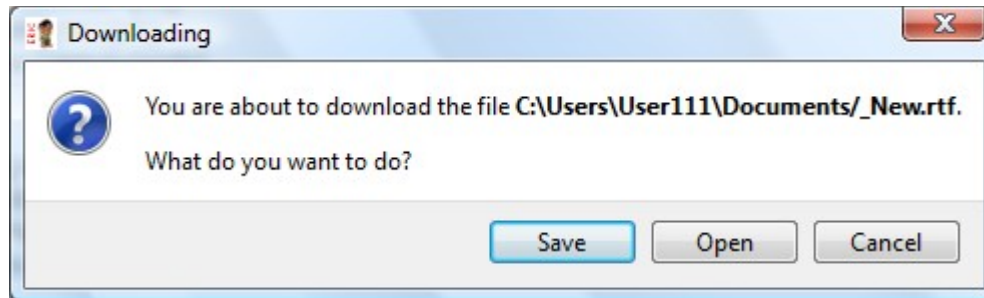
Designed to open and display a file of type: `*.html / htm`, `*.pdf`, `*.chm` or generic (i.e.: `*.*`), each one in a slightly different way. That is:

`*.html`, `*.htm` (default) opened on the currently active browse tab-form [see command: `File > New Tab`]

`*.pdf`, `*.chm` opened on an independent window, by the very application associated to that file-type

Then, with files of other generic type “`*.*`”, the behavior may vary considerably. Here some notable examples:

- *.txt, *.jpg opened directly on the current browse tab-form, as for *.htm [see]
 *.rtf instead of opening the named file, it displays such a dialog box:



Viewpoints

- Behavior that, even though modeled after that of other major browsers [e.g.: Firefox], we can't help considering rather curious.

Remarks

- Note that E4–WB makes a clear difference between browsing the contents of a file, as here described, and browsing the contents of a web site. For the latter, refer to the section *Web URLs Entry*, at the previous chapter 1. Introduction & Generality [see].
- Note that an actual file path, such as:
`C:\Python27\Lib\site-packages\eric4\Documentation\Source\index.html`
 can be entered into the “Web URLs Entry” field [see], where it is automatically re-interpreted as a URL of format⁶:
<file:///C:/Python27/Lib/site-packages/eric4/Documentation/Source/index.html>
 Whereas the inverse doesn't hold, in the sense that no web URL can be entered into this “Open File” command, which is strictly intended to operate with files.
- Anyhow, besides the usual file path format, such as: `C:\ [...] \index.html`, this command recognizes also this file-URL format: [file:///C:/ \[...\] /index.html](file:///C:/ [...] /index.html)⁷
- <?> As just a little annoying glitch, this “Open File” dialogue box doesn't keep memory of the last directory possibly selected, and keeps this same initial default directory [see]:
`C:\Python27\Lib\site-packages\eric4\`

– –

⁶ By the way, this is the same URL format used to identify a VCS Subversion repository on local disk.

⁷ By the way, note that a slash “/” and a backslash “\” separator are not less different characters then, say, a colon “:” and a semicolon “;”. Obviously not to be confused.

File > **Open File in New Tab**

Designed to open and display files as with the command `File > Open File` [see], but into a new browse tab-form [see command: `File > New Tab`], instead of the current one.

Remarks

- Here there's no difference from command `File > Open File` [see] with files of type: “*.pdf” and “*.chm”, which are anyhow opened into an independent window of their own [see].

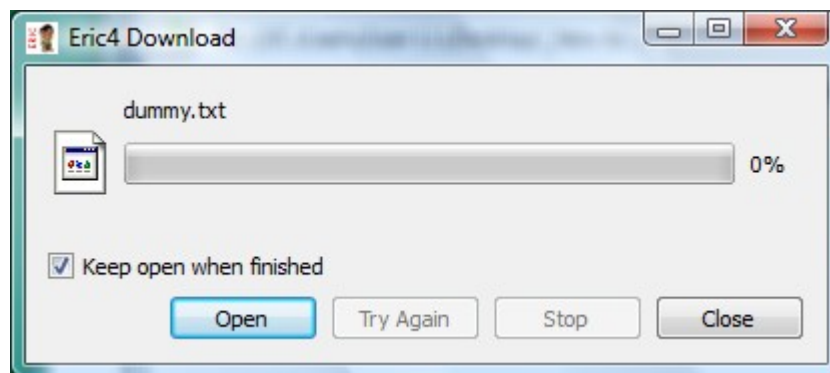
--

File > **Save As...**

Designed to save a copy file of the page in display on the current browse tab-form.

Remarks

- This command concerns only files actually in display on a browse tab-form, typically of type: *.html, *.txt, or *.jpg; not those of type such as: *.pdf or *.chm, possibly opened into other dedicated windows [see command: `File > Open File`].
- This command operates all the same both in case of a “*.html” file or of an actual web page in display on the current browse tab-form.
- <?> Alas here too no record of the last selected directory is kept for the “Save File” dialogue box.
- <?> With local files of various type such as *.txt or *.html, when clearly no download operation is involved, such curious “Download” dialogue box will appear:



Just a “Close” answer, and you'll smoothly get rid of it.

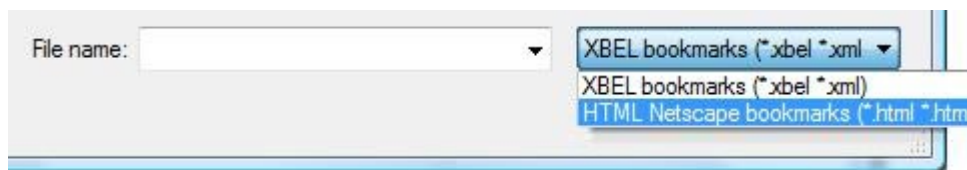
- `<?>` Some browse pages, such as the initial “Welcome” and those generated by Qt Help System documents [see `Settings > Manage QtHelp Documents`], are here ignored, no idea whether it's by design or not.

--

File > Import Bookmarks...

Designed to import a “Bookmarks” file, which is the usual tool to conveniently organize and find frequently visited web pages. Refer to menu command: `Bookmarks` to see how to make use of them.

Such files, also known as “Favorites”, can be both those generated by this same E4–WB program, via menu command: `File > Export Bookmarks...` [see], or similarly by other most popular browsers⁸. These the available import options:



--

File > Export Bookmarks...

Just the inverse of the corresponding “`File > Import Bookmarks...`” command [see].

--

File > Print Preview

File > Print

Designed to preview and print the contents of the current browse tab-form in display. Files possibly opened into a dedicated window, such as those of type “*.pdf” [see *command: File > Open File*], are here not considered, being meant to be printed with the specific tools available in their own window.

Remarks

- `<!>` It's worth recalling that this important `File > Print` feature didn't work until upgrading to PyQt ver. 4.9-1, containing the Qt ver. 4.8.0. That said to encourage constant and careful upgrading actions for all Eric components [see *Upgrade section, at chapter 5. Setup and General Management of the main Eric4 – Technical Report*].

--

⁸ Such as the Windows Internet Explorer, as we've satisfactorily verified, after overcoming some difficulties (hint: modify the exported file extension from “*.htm” to “*.html”, then adjust the URL entries order possibly altered).

File > Close**File > Close All**

Designed to close the currently active, or all the possibly opened browse tab-forms [see: `File > New Tab`]. All but at least one, that will remain anyhow open in form of the initial “Welcome” home page.

- -

File > Private Browsing

Designed to activate a rather discreet operating mode, leaving no tracks of your navigation on the web to be possibly peeped by other users using this same browser, after you did. In particular: records of web page URLs, site icons, cookies, search and browse history will be not stored; and with the `Back` and `Forward` buttons that will remain active only during the current session.

Such private browsing data may be anyhow cleared at will via menu command `Settings > Clear Private Data` [see].

- -

File > Quit

Designed to close current E4-WB session, along with all its windows possibly created.

- -

Menu Command: Edit

Copy

Find... Find Next Find Previous

--

Edit > Copy

Designed to copy into the system clipboard storage the text previously selected on the currently active browse tab-form. Once copied, the text—text only, no graphics—can be pasted into other applications with the usual “`Ctrl+V`” shortcut command.

Remarks

- Shortcut command “`Ctrl+C`” is here particularly handy [see].
- Images can be possibly copied separately, with the right-click pop-up menu `Copy Image to Clipboard` [see].

--

Edit > Find...**Edit > Find Next****Edit > Find Previous**

Designed to activate a text search on the current browse tab-form, to find `Next/Previous` occurrences of the “`Find:`” text field, with the usual options `Match case` and `Wrap around`.

Shortcut command “`Ctrl+F`” is here particularly handy [see].

--

Menu Command: View

Zoom In Zoom Reset Zoom Out
Zoom Text Only
Show Page Source
Full Screen

- -

View > Zoom In**View > Zoom Reset****View > Zoom Out**

Designed to zoom In/Out the display image, or reset it to a 100% zoom factor [*see also: View > Zoom Text Only*].

- -

View > Zoom Text Only

Designed to exclude/include graphic elements from the zooming effect [*see also: View > Zoom*].

- -

View > Show Page Source

Designed to open the current web-page in display as a source code file, into a dedicated “Eric4 Mini Editor” window.



- -

View > **Full Screen**

Designed to increase the size of the E4-WB window as much as possible on the screen. It operates a bit differently from the usual “Maximize” window button, indeed here even the title and menu bars are dropped, so that, to reset the preceding display condition, the only way left is to use the:



“Restore Window” toolbar icon, or the F11 function key⁹.

--

⁹ Note that the function key F11 here acts as a toggle, both to maximize and restore the window size.

Menu Command: Go

Backward Forward
Home
Stop
Reload
Sync with Table of Contents

- -

Go > Backward**Go > Forward**

Designed to go back/forth on the stack of the stored pages in display. Commands that are “dimmed”-disabled when no other pages are available for displaying.

- -

Go > Home

Designed to go to, and display, the “Home page”, as defined with command: `Settings > Preferences...`, form: `Help > Eric Web Browser`, parameter: `Home page [see]`.

- -

Go > Stop

Designed to interrupt a possibly lengthy process of loading and displaying of a web page.

- -

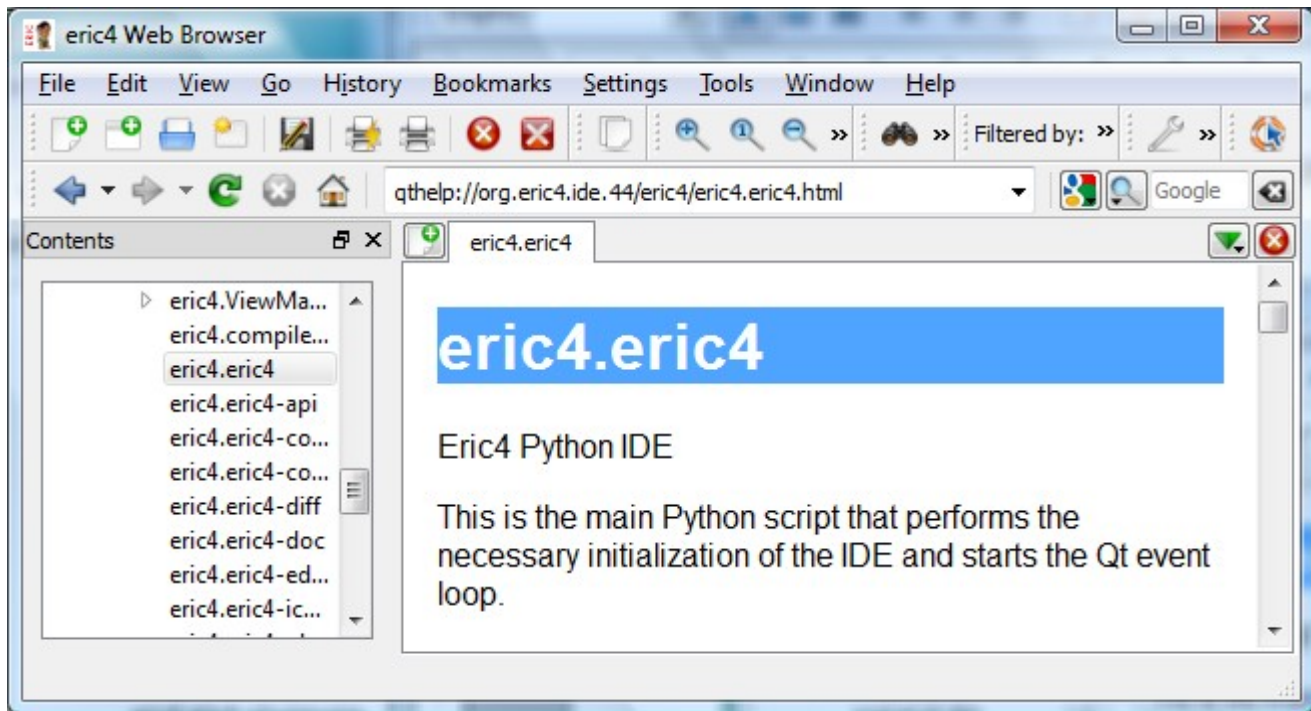
Go > Reload

Designed to refresh the process of loading and displaying of the current web page.

- -

Go > Sync with Table of Contents

When browsing a Qt Help file, it's for displaying the related “Table of Contents” form, and synchronize it with the page currently in display [*see: Settings > Manage QtHelp Filters*].



Remarks

- Note that whereas this “Contents” form is necessarily tied to show the Table of Contents of the possibly selected Qt Help document [see: Settings > Manage QtHelp Documents], the browse tab-form currently active can be independently used to show totally different file or web pages; and this, sometimes, may reveal a bit confusing.

--

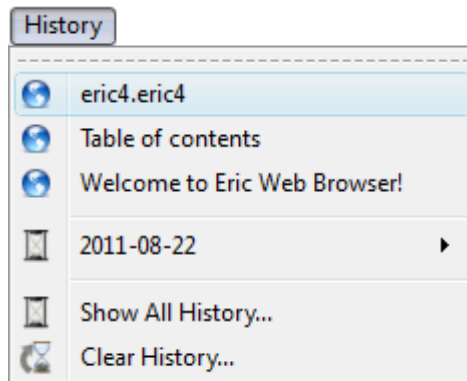
Menu Command: **History**

<URLs>

Show All History...

Clear History...

Designed to activate this kind of drop-down menu:



Where you can make use of, and manage, the browsing history as currently recorded.

--

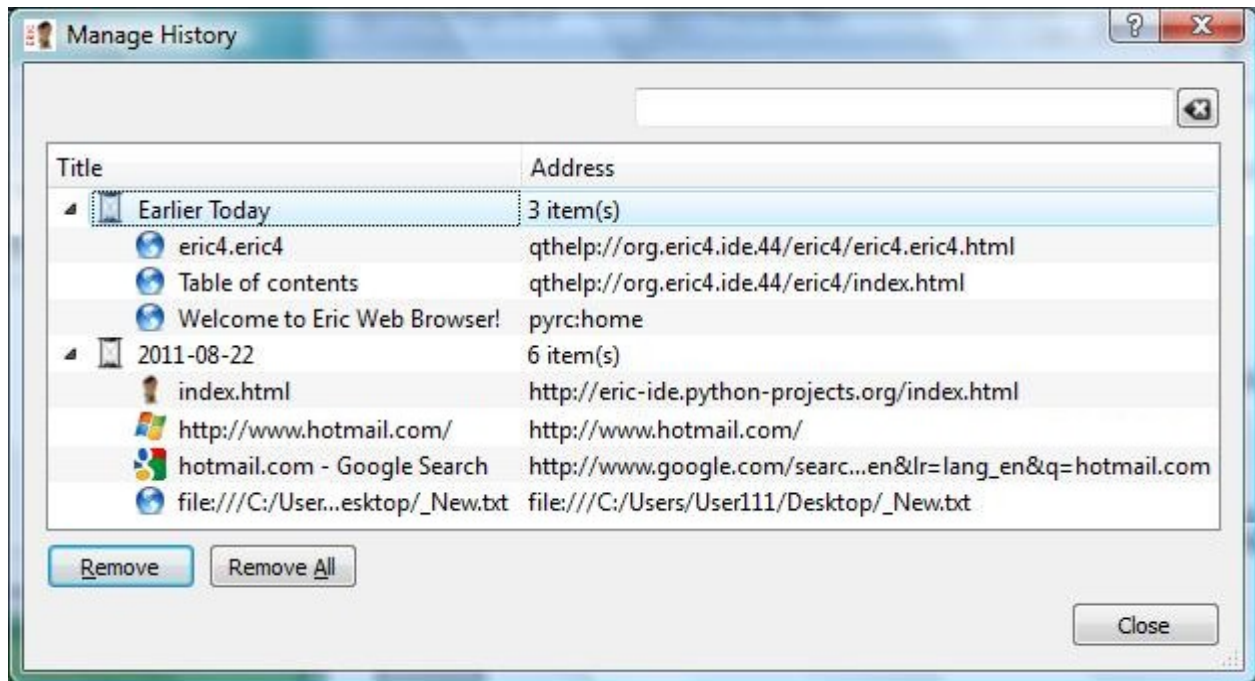
History > <URLs>

Display of the actual browsing history list as recorded on a daily basis, starting with the current one, and where to possibly call again a stored URL. History that accumulates automatically, then is conveniently ordered and collected into distinct day-lists, and that can be managed with the next menu command `History > Show All History...` [see].

--

History > **Show All History...**

Designed to show such a “Manage History” control window [see], where to manage the navigation history currently recorded.



That is, where to inspect, search and possibly remove recorded <URLs> items.

--

History > **Clear History...**

Designed to fully erase the navigation history currently recorded.

--

Menu Command: **Bookmarks**

Manage Bookmarks...

Add Bookmark...

Bookmark All Tabs...

Add Folder...

Bookmarks Bar

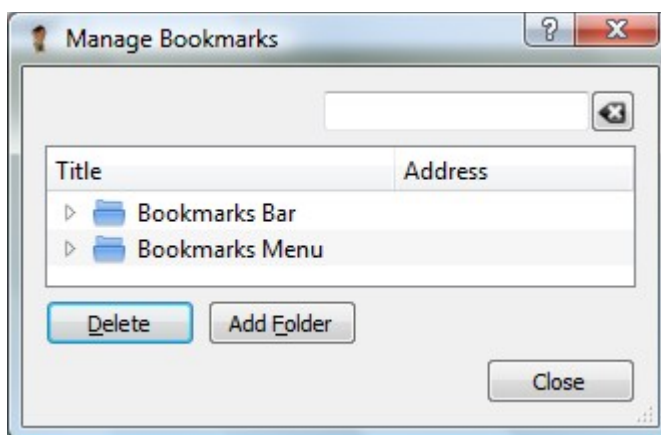
<URLs>

Open All in Tabs

--

Bookmarks > **Manage Bookmarks...**

Designed to get a control box aimed at managing the recorded set of preferred URLs, here called *Bookmarks*, but also known as *Favorites*. That is: to find, examine, move, delete and collect them into dedicated folders.



`Bookmarks Bar` and `Bookmarks Menu` are the two main folders here available, corresponding to the two subsets of items available on this very menu, as hereafter described [see].

Remarks

- Even though not without some difficulty and glitches, such Favorite/Bookmark sets can be here imported from other web browsers [see: `File > Import Bookmarks...`].

--

Bookmarks > Add Bookmark...

Designed to record the URL of the currently active browse tab-form into a given Bookmark folder [see also section: Web URLs Entry, at the chapter 1. Introduction & Generality].

- -

Bookmarks > Bookmark All Tabs...

Designed to record the URLs of all currently active browse tab-forms into a given Bookmark folder.

- -

Bookmarks > Add Folder...

Designed to create a new Bookmark folder. It does correspond to the “Add Folder” button of the “Manage Bookmark” window [see].

- -

Bookmarks > Bookmarks Bar

Standard folder, typically aimed at hosting contributed URLs, relevant to Eric.

- -

Bookmarks > <Bookmarks Menu>

Menu section typically aimed at hosting a handy list of custom URLs.

- -

Bookmarks > Open All in Tabs

Command positioned at the bottom of each different Bookmarks menu sub-set [e.g. see: Bookmarks Bar, or Bookmarks Menu], so to open at once all related URLs.

- -

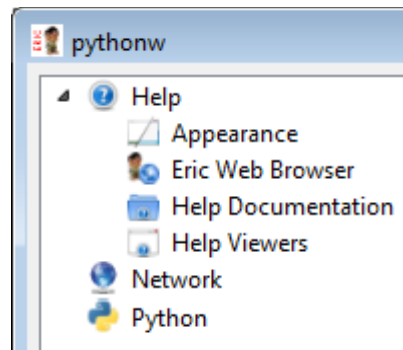
Menu Command: **Settings**

Preferences...
Languages...
Cookies...
Configure Search Engines...
Manage Saved Passwords...
Ad Block...
Manage QtHelp Documents
Manage QtHelp Filters
Reindex Documentation
Clear Private Data
Clear Icons Database

--

Settings > **Preferences...**

Designed to show a “pythonw” window aimed at managing the the specific sub-set of the Eric operative preferences related with its web browser activity. Corresponding parameters values on the main Eric program are exactly the same [see on Eric].



Viewpoints

- Here we assume that most items are self-explanatory enough not to require a general and detailed description. Instead we'd rather offer specific comments only where needed, on a case-by-case basis.

--

Settings > Languages...

Designed to show a dialog box aimed at arranging the browsing languages in the desired order of preference. Of course this setting will apply only to actually multi-language sites.

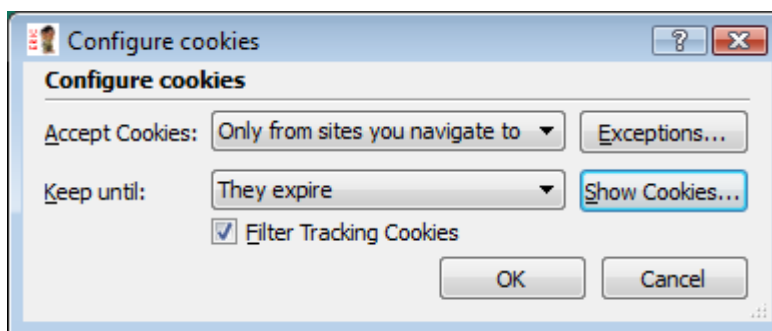
Remarks

- You are advised to give a look at this setting, given the fact that the default condition¹⁰ could be different from what you'd expect.

--

Settings > Cookies...

Designed to show a dialog box aimed at managing website “Cookies”¹¹, assumed intuitive enough not to require any specific explanation.



Remarks

- Here the “Filter Tracking Cookies” check box is meant to stop all of the so called “Tracking Cookies”, also known as “Persistent Cookies”¹².

--

Settings > Configure Search Engines...

Designed to show a dialog box where to manage the preferred web search engines, as here described at section User Interaction – Web Search, chapter 1. Introduction & Generality [see].

--

¹⁰ For instance, the default condition we've found gave priority to French [fr] over English [en] and English/UnitedStates [en-us].

¹¹ *Cookie* – A small text file that websites put on your computer to store information about you and your preferences.

¹² These special “*Persistent Cookies*”—so named as they can outlast the user sessions—could be used to record vital piece of information about the user activity, and then sent back to the server every time the user calls again.

Settings > **Manage Saved Passwords...**

This command is aimed at managing—that is: Show/Hide and Remove—the passwords possibly entered and saved during a browsing session, when `Save password check-box` condition of the `Settings > Preferences...`, `Eric Web Browser > Security is set on [see]`.

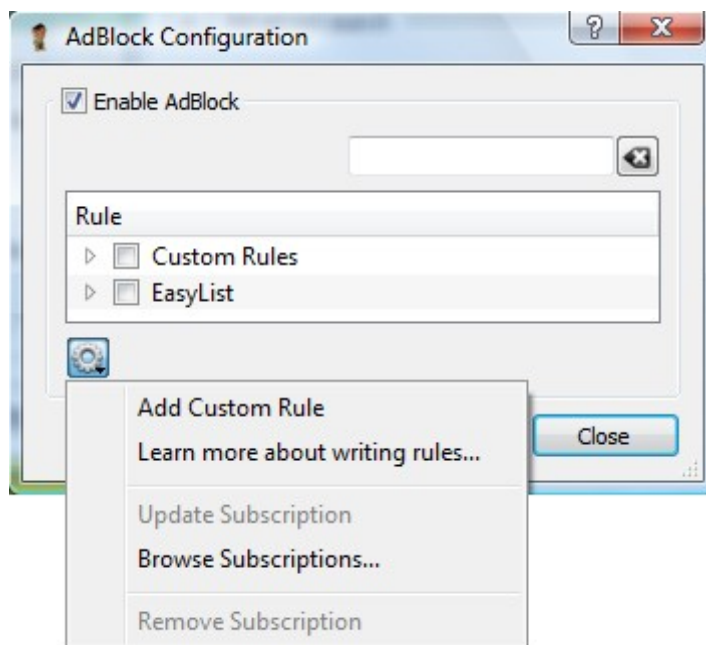
Remarks

- Security implication of this management is here not illustrated, as the user is assumed to be well aware of it.

- -

Settings > **Ad Block...**

As with Mozilla Firefox and Google Chrome, also E4-WB can make use of “AdBlock”¹³, an extension aimed at allowing users to prevent page elements, such as advertisements, from being downloaded and displayed.



And this is where to enable, disable and manage it.

13 Adblock Plus (ABP) , a forked version of AdBlock, is a content-filtering extension for web browsers.

To know more see the adblockplus.org website.

Remarks

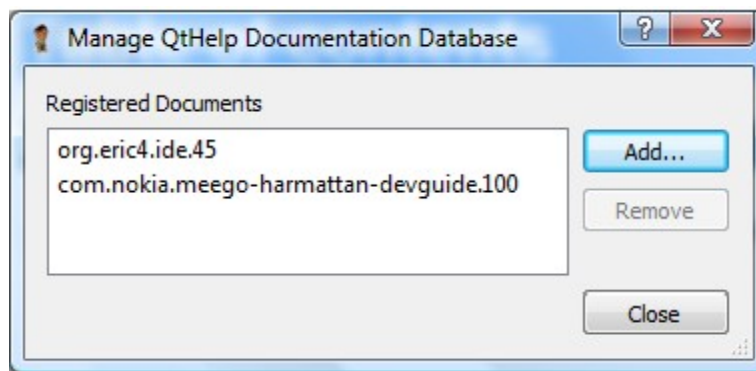
- Actual managing actions require on line connection to the `adblockplus.org` website.
- Update and Remove Subscription commands require the selection of the desired Rule item.

--

Settings > Manage QtHelp Documents

This section of the menu offers virtually the same functionality of the “Qt Assistant”, the standard browser for “Qt Help System” documents¹⁴. It's another example of a useful feature conveniently embedded into the unique Eric developing environment [see next QtHelp commands].

Here you have the control form where standard Qt Help files of type “*.qch” can be registered, so to become available for subsequent browsing and searching actions.



Standard Eric setup comes with such a “source.qch” Qt Help file, which is the documentation of the very package implementing the Eric 4 Python IDE, stored into the “..\eric4\Documentation\Help” directory, and here shown in the Registered Documents list as: “org.eric4.ide.44”.

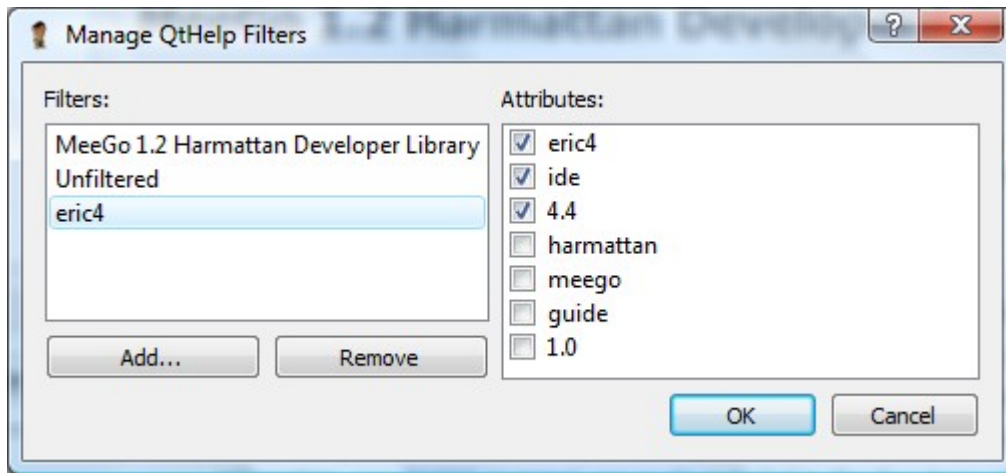
Other Qt Help files can also be added, as the next “com.nokia” example here shown.

--

Settings > Manage QtHelp Filters

Designed to show the control form where to define the “Filters” aimed at managing which ones of the registered Qt Help documents are to be actually available for browsing and searching [see former command: Settings > Manage QtHelp Documents].

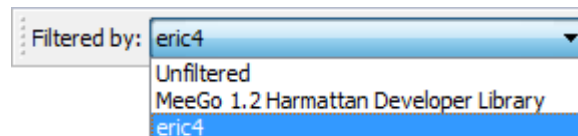
¹⁴ For more information see the article “The Qt Help Framework”, on the “`developer.qt.nokia.com`” web site.



Remarks

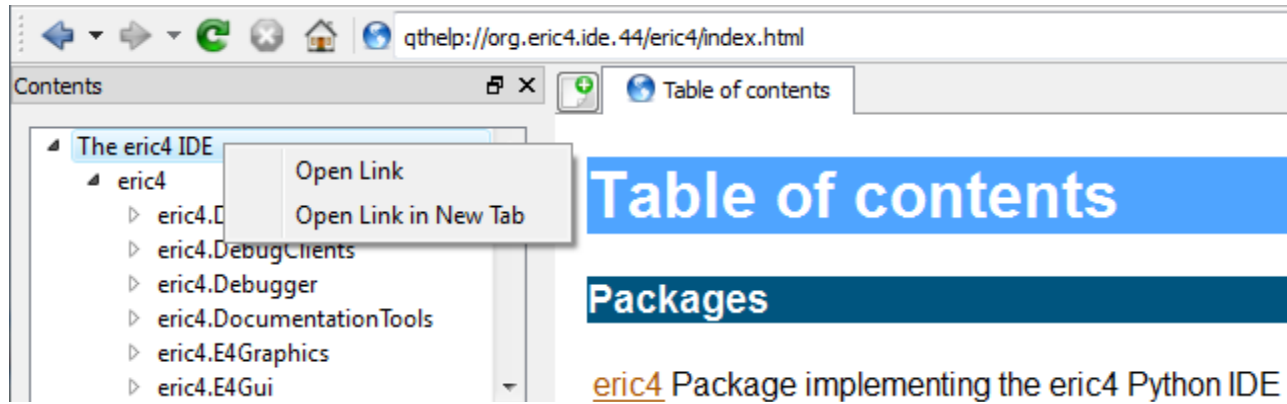
- The pre-defined “Unfiltered” dummy filter is meant to see all registered Qt Help documents.
- Filters’ “Attributes” are parameters defined at the Qt Help document creation, listed here as part of the registration process and then, curiously enough, here remaining listed even after the possible removal of the respective document from the `Manage QtHelp Documentation Database control form` [see].

Of all the available filters, that one currently in use is to be selected acting on this “Filter by:” drop-down list, available on the tool bar:



The selected filter will decide what is actually available on the `Search` and `Index` forms [see also `Menu Command: Window`].

Then a `Open Link`, or `Open Link in New Tab`, or a click on this `Contents` form will show the related item on the browse tab-form.



Remarks

- Note the corresponding “`qthelp://`” scheme shown into the URL entry combo box [*see other URL Scheme on the Eric Web Browser, Settings > Preferences...*].

--

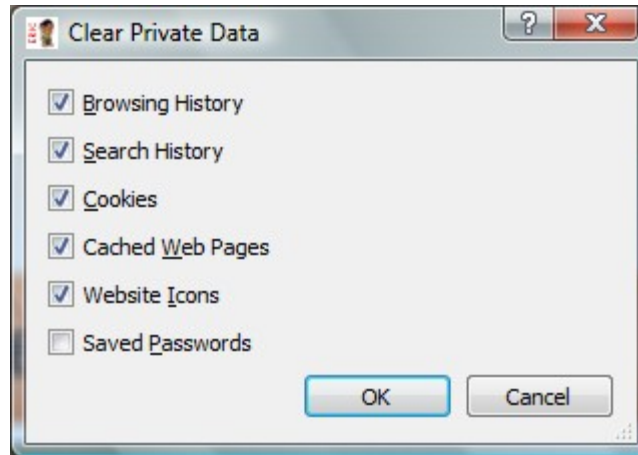
Settings > Reindex Documentation

Recovery action possibly useful to reset a corrupted index system on which is based the Qt Help search mechanism [*see: Window > Search*] as can happen, for instance, during the development of a new Qt Help documentation file.

--

Settings > **Clear Private Data**

Designed to clear the “Private Data” as listed on the hereafter form, possibly accumulated during the last browsing sessions.



Such “private” browsing data may even not be recorded at all, if operating with the `File > Private Browsing mode` on [see].

--

Settings > **Clear Icons Database**

Designed to clear the so called favorite icons, or “*favicons*”¹⁵, possibly accumulated visiting web sites that provide them, typically, to be shown next to their URL, or next to the favorite entry name [see also command: `Bookmarks`].

Remarks

- And if you wonder why “Database”, it's because these icons are stored together into the file “`WebpageIcons.db`”, into such directory: `C:\Users_eric4\browser\favicons` [see].

--

¹⁵ Also known as: bookmark, shortcut, URL or Web site icon.

Menu Command: **Tools**

Show Network Monitor

Remarks

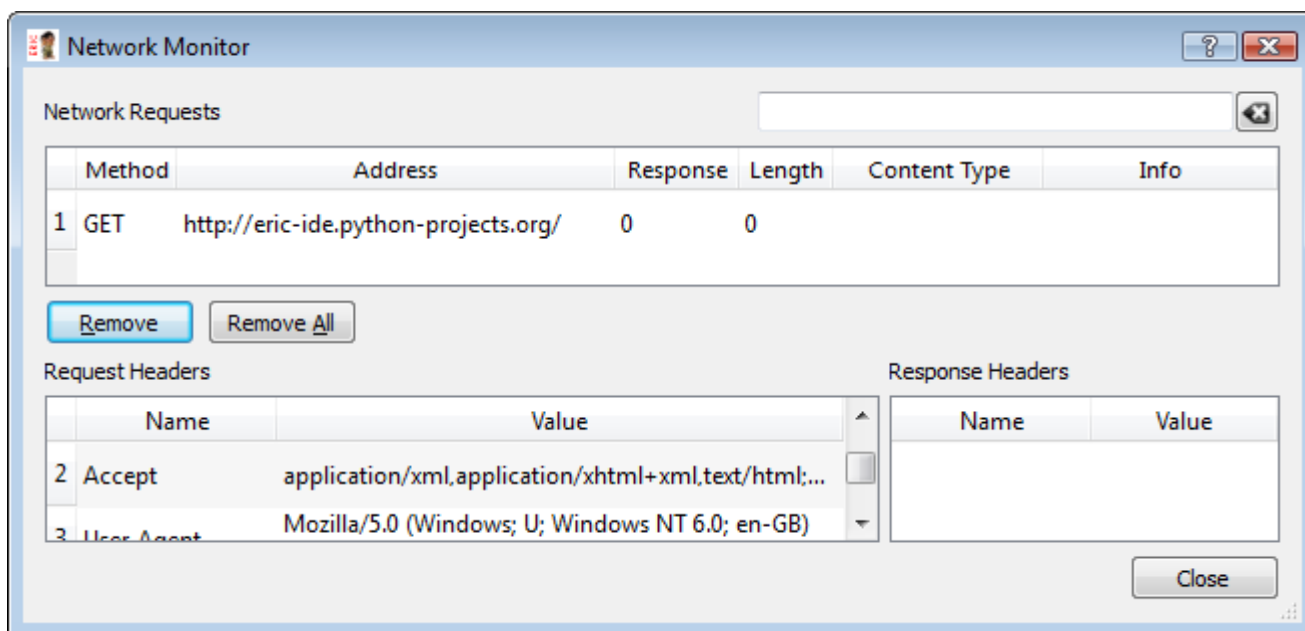
- Just one such “Tool” is currently here available, presumably more in the future.
- From the very nature of this “Tool” currently available [see], we're inclined to assume that the tools here envisioned are meant to the Eric developers rather than, strictly speaking, to this web browser users.

- -

Tools > **Show Network Monitor**

A special command, typically meant for test and debug purposes of web applications possibly developed under Eric¹⁶. Through this monitor you can check and see what's the active behavior of any web pages you're visiting, whether you are developing them, or they are totally unknown to you, and possibly suspicious.

This the on-line browsing activity monitor shown:



¹⁶ Related web application examples: Django, a high-level Python web framework, TurboGears, a ready-to-extend web application framework, Pylons, a Python open source web application framework.

Where:

Network Requests

List of methods used by the web browser to retrieve a web page, or a file

Method	Control method identifier
Address	Site URL, or file path
Response	Action return code
Length	Retrieved data length
Content Type	Retrieved data MIME ¹⁷ type
Info	Possible additional info

Request/Response Headers

Headers of data interchanged with the web server, by the method currently selected on the top pane

Name	Header name
Value	Header data

Remarks

- The “Header”, that is the blind meta-data located on top of the visible body-part of a web page.

- -

¹⁷ MIME: Multipurpose Internet Mail Extensions

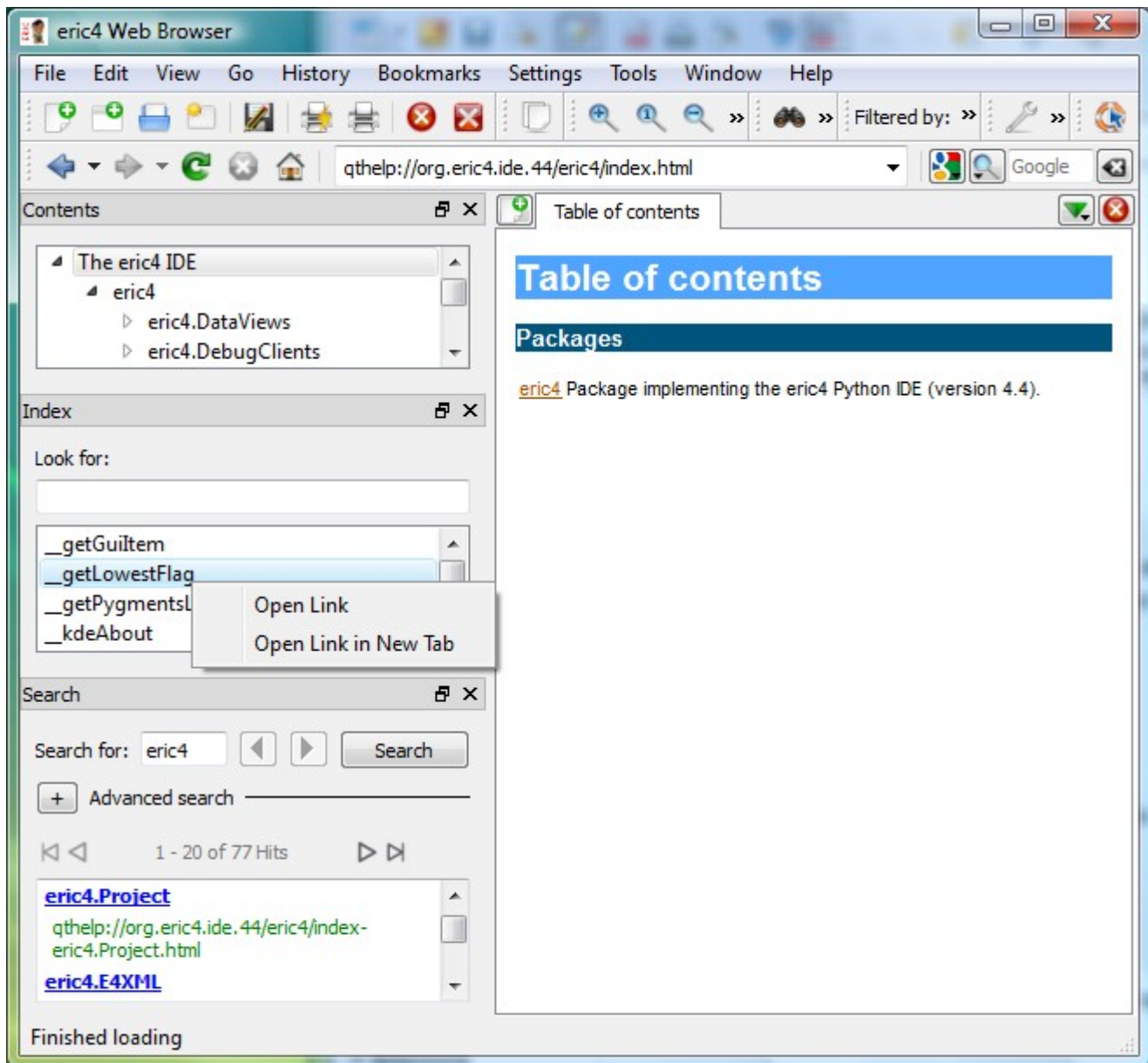
Menu Command: **Window**

Table of Contents

Index

Search

Menu of commands to show some utility control forms useful to navigate on a Qt Help document file [see *related command*: Settings > Manage QtHelp Filters].



Such forms are usually docked within the Eric workspace, along with the browse tab-forms, from where they can be un-docked and moved at will.

- -

Window > **Table of Contents**

Designed to show a “Contents” form of the Qt Help documents possibly registered via menu command `Settings > Manage QtHelp Documents` [see]. Acting on this table's items will do to display the synchronized subject on the browse tab-form [see also: `Go > Sync with Table of Contents`].

Remarks

- Even with no Qt Help file in display this “Contents” form will keep the last displayed contents. Which sometimes may reveal a bit confusing.

- -

Window > **Index**

Designed to show an “Index” form, where to operate similarly as with the above “Contents” form [see].

- -

Window > **Search**

Designed to show a “Search” form, where to operate similarly as with the above “Contents” form [see].

- -

Menu Command: **Help**

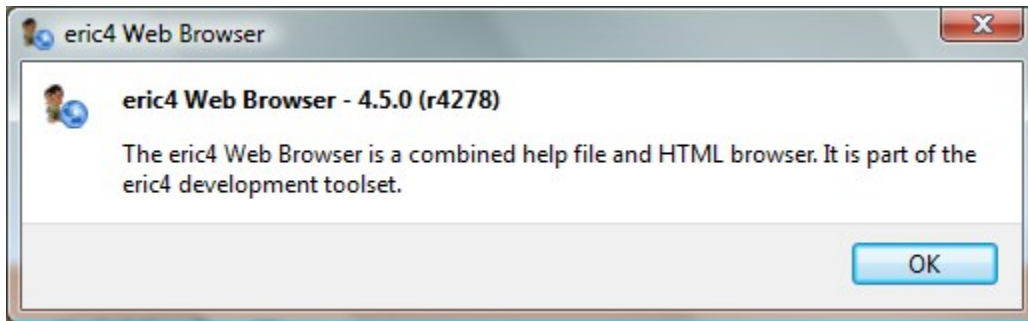
- About
- About Qt
- What's This?

--

S-PM 120200

Help > **About**

Designed to show this product's identification and definition card.



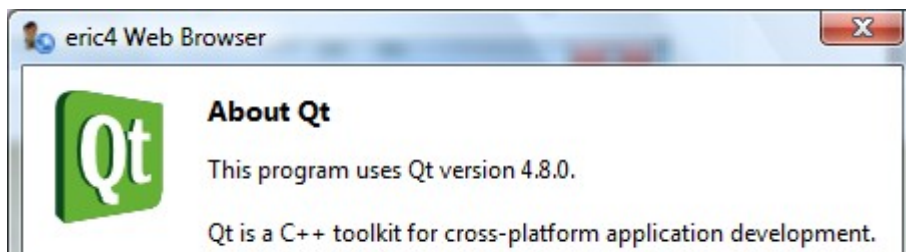
Remarks

- Note the fair definition of this tool as a “help file and HTML” viewer, conveying an implicit suggestion about what to properly use it for, and, inversely, what not (e.g.: not for general web browsing, for which you'd easily find better suited products).

--


Help > **About Qt**

Designed to show the identification card of this product's main toolkit component.



--

Help > **What's This?**

Designed to get this special mouse-pointer:  , which can be used to show a pop-up help hint about most control items on this product's window.

Remarks

- Here the corresponding function key shortcut “Shift+F1” does reveal particularly handy.

- = -

Appendix

Topics here treated should be considered as a specialized addition to the corresponding ones treated in the central volume of this collection, the “Eric4 – Technical Report” [see], which remain valid in general and are not going to be duplicated here.

Start Command

E4-WB can be started during an Eric session as any ordinary IDE feature, with the Eric menu command:

```
Help > Helpviewer...
```

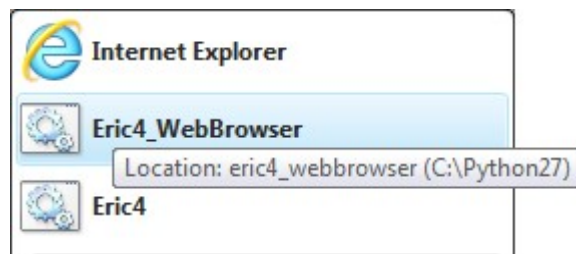
Or it can be started as an independent program, running the Python executable file:

```
eric4_webbrowser
```

A file located in the Eric installation directory `\Python2x\Lib\site-packages\eric4`, with the choice between a “*.py” source, a “*.pyc” compiled and a “*.pyw” silent-console executable file type; or via the equivalent “*.bat” file located in the `\Python2x` top directory [see].

Dedicated Start Command

<!> In case of a frequent use of this latter independent way of execution, it may be convenient to create such a dedicated *Start* command:



This way:

- 1] Locate the `eric4_webbrowser.bat` executable file on such top directory: `C:\Python2x\`
- 2] Right-click it to “Create Shortcut”, then conveniently rename it “Eric4_WebBrowser” and keep it where created;
- 3] Drag-drop a copy of this shortcut into the system menu *Start*, and/or also into the system taskbar; keeping anyhow also the original shortcut as said at the former point.

- -

Primary Settings & Adjustments

Here you have a suggested list of settings and adjustments that you'd better not to ignore, given the fact that the conditions of use you will actually encounter could not be what you'd possibly prefer.

S-PM 120000

Settings > Languages...

Where to set your preferred browsing language

Settings > Ad Block...

In case you don't precisely know what such "Ad Block" is, or simply do not intend to make use of it, here is where to disable it; and also to get rid of possibly related errors such as the declared unavailability of the "http://adblockplus.org" web site when operating off line.

"Saving History" Error

In case of this error, just erase such file¹⁸:

C:\Users\

- = -

¹⁸ Or even the whole C:\Users\

Contents

Foreword.....	3
Essentials.....	4
1. Introduction & Generality.....	5
Why an Eric (4) Web Browser?.....	6
Virtues.....	6
Limits.....	6
User Interaction.....	8
Web URLs Entry.....	8
Web Search.....	8
Context Menu Commands.....	9
2. Menu Commands.....	10
Menu Command: File.....	10
File > New Tab.....	11
File > New Window.....	11
File > Open File.....	11
File > Open File in New Tab.....	13
File > Save As.....	13
File > Import Bookmarks.....	14
File > Export Bookmarks.....	14
File > Print Preview.....	14
File > Print.....	14
File > Close.....	15
File > Close All.....	15
File > Private Browsing.....	15
File > Quit.....	15
Menu Command: Edit.....	16
Edit > Copy.....	16
Edit > Find.....	16
Edit > Find Next.....	16
Edit > Find Previous.....	16
Menu Command: View.....	17
View > Zoom In.....	17
View > Zoom Reset.....	17
View > Zoom Out.....	17
View > Zoom Text Only.....	17
View > Show Page Source.....	17
View > Full Screen.....	18
Menu Command: Go.....	19
Go > Backward.....	19

Go > Forward.....	19
Go > Home.....	19
Go > Stop.....	19
Go > Reload.....	19
Go > Sync with Table of Contents.....	19
Menu Command: History.....	21
History > <URLs>.....	21
History > Show All History.....	21
History > Clear History.....	22
Menu Command: Bookmarks.....	23
Bookmarks > Manage Bookmarks.....	23
Bookmarks > Add Bookmark.....	24
Bookmarks > Bookmark All Tabs.....	24
Bookmarks > Add Folder.....	24
Bookmarks > Bookmarks Bar.....	24
Bookmarks > <Bookmarks Menu>.....	24
Bookmarks > Open All in Tabs.....	24
Menu Command: Settings.....	25
Settings > Preferences.....	25
Settings > Languages.....	26
Settings > Cookies.....	26
Settings > Configure Search Engines.....	26
Settings > Manage Saved Passwords.....	27
Settings > Ad Block.....	27
Settings > Manage QtHelp Documents.....	28
Settings > Manage QtHelp Filters.....	28
Settings > Reindex Documentation.....	30
Settings > Clear Private Data.....	31
Settings > Clear Icons Database.....	31
Menu Command: Tools.....	32
Tools > Show Network Monitor.....	32
Menu Command: Window.....	34
Window > Table of Contents.....	35
Window > Index.....	35
Window > Search.....	35
Menu Command: Help.....	36
Help > About.....	36
Help > About Qt.....	36
Help > What's This?.....	37
Appendix.....	38
Start Command.....	38
Primary Settings & Adjustments.....	39

- = -