WebSite-Watcher

This manual contains parts of the
WebSite-Watcher help file.

Table of Contents

Part II User Interface   95

Part III Scripting language   139

Part IV Plugin System   145

Index 0

Part

I

1 WebSite-Watcher HELP

1.1 Welcome to WebSite-Watcher

WebSite-Watcher allows you to monitor your favorite websites for updates and changes with a minimum of time and effort. This prevents you from visiting each page manually looking for changes. When changes in a page are detected, WebSite-Watcher saves the last two versions to your hard disk and highlights all changes in the text.

Many features give you control, when and how a page should be monitored for updates. For example, to only receive an update notification if a specific keyword is available in the changed content.

WebSite-Watcher works as a client on your computer, similar to an e-mail client. All bookmarks and configurations are stored locally on your hard disk, WebSite-Watcher does not need our servers to check pages for updates and changes.

Main features

- Monitor web pages
- Monitor password protected pages
- Monitor forums for new postings and replies
- Monitor RSS feeds
- Monitor local files
- Monitor PDF/Word/Excel files
- Highlight changes in a page
- Powerful filter system to ignore unwanted content
- Many more powerful features to stay up-to-date!

1.2 Overview

1.2.1 The main window

The main window of WebSite-Watcher is divided into three areas.
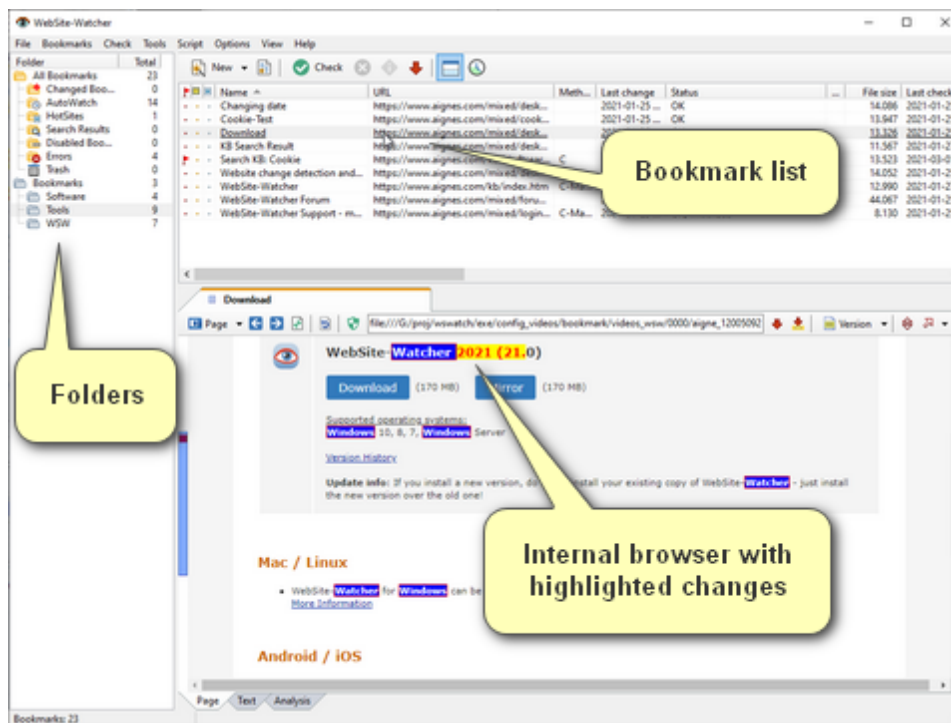
1. The **folder tree** on the left side contains several predefined folders (such as AutoWatch, HotSites or Trash) and user defined folders which can be created by the user.

2. The **bookmark list** shows all bookmarks of the selected folder. Each bookmark has its own configuration which can be opened via the context menu.

3. A double click onto a bookmark opens the web page with highlighted changes in the **internal browser**. After starting WebSite-Watcher, this internal browser is closed and will be opened automatically as soon as you open a bookmark. The internal browser can optionally be placed at the bottom or on the right side of the bookmark list.

1.2.2 Main menu

File

The File menu contains all related commands for creating, opening and maintaining bookmark files (databases). A bookmark database has the file extension *.wsw and contains all folders, bookmarks and configurations. Cache files are saved in the related *_WSW folder.

The menu "**Database**" contains all commands to maintain, optimize or repair a database. You only have to run these commands when experiencing problems with your database, such as crashes, performance problems etc.

If your database is damaged, then running the "**Repair database**" command tries to restore as many data as possible. If you have a working and up to date backup file, then you should restore

that backup instead.

A further alternative to repair a damaged database is to use "**Re-create database from cache folder**". This command rebuilds the entire database from files that are available in the cache folder. You should use this command as last alternative if the command "Repair database" fails and if you don't have a working and up to date backup file.

Bookmarks

This menu contains all commands for managing bookmarks.

The commands "**Find**" and "**Find next**" let you search for text in all bookmark configurations of the selected folder. If you want to search for text in checked web pages, you have to use the Extended Search feature.

With the "**Replace**" feature you're able to change text within selected bookmark configurations.

Check

This menu contains all commands to check web pages.

The command "**Check HotSites**" selects and checks all bookmarks in the "HotSites" folder. The command "**Check selected bookmarks (extended)**" gives you more options which bookmarks should be checked, for example to ignore entered time configurations.

Tools

The Tools menu contains several helper tools, for example to import/export bookmarks from/to various formats, synchronize databases via Cloud, search for text in monitored web pages, etc.

Script

WebSite-Watcher includes a simple internal scripting language that lets you automate many things and user interactions. For example to check your bookmarks, send a generated report by e-mail and repeat the whole procedure after a specified time.

Options

This menu contains all WebSite-Watcher options and configurations.

View

This menu contains commands to show/hide main window elements, bookmark statistics, log files, etc.

Help

Help topics and of course the About box.

1.2.3 Toolbar

Main Toolbar

	Open a bookmark file (database). Bookmark lists are saved in files with the extension *.wsw, monitored web pages in the related *_WSW cache folder
	Wizard to add new bookmarks
	Add new bookmark
	Check all bookmarks in the selected folder
	Abort check
	Show information window (check status, etc)
	Go to previous bookmark
	Go to next bookmark
	Go to next changed/unread bookmark (changed bookmarks are displayed in bold and red)
	Open/Close internal browser
	Open a window with extended search features
	Turn AutoWatch on/off
	Move WebSite-Watcher to the system tray
	Exit WebSite-Watcher

Browser Toolbar Buttons

	Browse back
	Browse forward
	Abort
	Reload page in internal browser
	Find text in internal browser
	Additional page commands and features
	Edit bookmark properties
	Create an Auto-Filter (Ignore highlighted changes)

↺	Restore previous version
↓	Open next changed bookmark (changed bookmarks are displayed in bold and red)
↓	Go to next change within the opened page
🌐	Open online version of the bookmark
📄	Open new version of the bookmark
📄	Open old version of the bookmark
📄	Open version with highlighted changes
↕	Open both, version with highlighted changes and old version
⇕	Maximize internal browser
↗	Open current page with external browser

Toolbars can be configured in the Options menu.

1.2.4   Symbols in bookmark list

Each bookmark can have one or more of the following icons:

⚑	Each bookmark can be assigned a small flag. Main purpose is to have a simple marker that helps to easily see important bookmarks at first glance. There are also some other functionalities based on these flags, for example the "Don't auto-delete bookmarks with flags" in the folder properties. You can assign a flag with a double click onto the first bookmark column or via the context menu where flags with more colors are available.
📝	This icon appears when notes have been entered in the bookmark properties. A double click onto that icon jumps into a separate "Notes" dialog without having to open the bookmark properties.
H	Each bookmark can be marked as a HotSite. All bookmarks with that flag are listed in the "HotSites" folder.
🔗	Bookmark is a Master-Bookmark (that means that the Follow-Links feature is enabled in the bookmark properties).
🔗	Bookmark is a "child" of a master bookmark and was inserted automatically via the Follow-Links feature.
⚙	Bookmark is queued for checking.
⚙	An error occurred during the bookmark-check, the bookmark was re-queued for a second check
🌐	Bookmark is currently checked
🌐	An error occurred during the bookmark-check, the bookmark is currently checked the second time

1.2.5   Changes side bar

When you open a bookmark in the internal browser of WebSite-Watcher, you can see the changes bar on the left side. This bar shows where changes and keywords are located and makes it easy to scroll to these locations.

The area where changes are located is displayed in <u>yellow</u>. The area before and after that yellow area is displayed in <u>white</u>.

Highlighted changes are only located within the yellow area and displayed in red.

Highlighted keywords within highlighted changes are displayed in blue. The changes bar doesn't show keywords that are placed outside of highlighted changes.

If a page contains highlighted changes that are hidden or cannot be displayed due to CSS definitions, then WebSite-Watcher shows them as small lilac markers. You can then switch to the "**Text**" tab of the internal browser, this view will always show all highlighted changes.
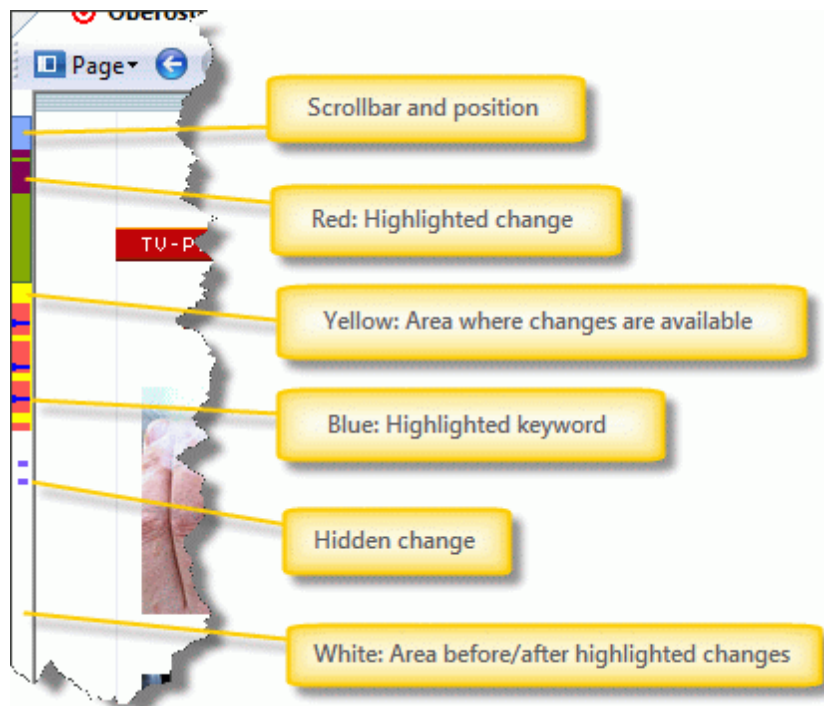
If the changes side bar contains xxx characters instead the change markers, then it is not possible to extract the correct position of the changes (WebSite-Watcher is dependent on some Chromium functions which sometimes report wrong positions). This is also the case if you use the Check by screenshot method where it is not possible to extract the position of changes.

1.2.6 What can be checked

WebSite-Watcher can monitor a lot of different file types and pages:

- Any kind of **web page** with text content. WebSite-Watcher supports both, static web pages (pages with file extension .htm or .html) and dynamic web pages (pages with file extension .asp, .php, etc).

- **RSS/Atom feeds**: RSS/Atom feeds are converted into a readable web page.

- **Pages behind a login**: For password protected pages you can record a Check-Macro that is

executed when the page is checked for updates.

- **Discussion forums**: Forums are handled as normal web page with various optimizations, for example a special method to highlight changes.

- **Pages with Javascript**: For pages where the content is displayed with Javascript, WebSite-Watcher supports a separate and more complex technology to convert the dynamically generated page content into a static content.

- **PDF files**: An integrated Plugin automatically converts PDF documents into web pages. Then PDF files can be monitored like normal web pages and WebSite-Watcher is able to highlight changes in the text.

- **Microsoft Word files**: An integrated Plugin automatically converts Microsoft Word documents into web pages. Then Word files can be monitored like normal web pages and WebSite-Watcher is able to highlight changes in the text.

- **Microsoft Excel files**: An integrated Plugin automatically converts Microsoft Excel documents into web pages. Then Excel files can be monitored like normal web pages and WebSite-Watcher is able to highlight changes in the text.

- **Images**: If you monitor an image file directly, WebSite-Watcher is also able to highlight changes within an image.

- **Binary files**: WebSite-Watcher is able to detect updates of binary files (for example zip or exe files) by comparing the file date, the file size and/or parts of the content (binary files won't be downloaded completely when they are checked for updates).

- **FTP files and directories**: WebSite-Watcher supports monitoring of FTP files and directories. FTP files can be checked by file date/size or by content.

- **Locally saved files**: WebSite-Watcher can monitor files on your local hard disk.

1.2.7   Useful Shortcuts

Navigate through changed bookmarks

F12	Open next changed bookmark
Ctrl+F12	Jump to next change
Enter	Open selected bookmark (if the bookmark list is focussed)
Shift+Enter	Open selected bookmark and maximize internal browser (if the bookmark list is focussed)

Ctrl+Enter	Set the focus to the bookmark list
Ctrl+Alt+Enter	Open selected bookmark in external browser

Internal Browser

F11	Maximize internal browser
Space	Scroll down the internal browser
Ctrl+Tab	Select next browser tab
Ctrl+W	Close selected browser tab
Ctrl+Shift+W	Close all read browser tabs
Shift+Click on 🌐📄📄📄 or *Ctrl+Click* on 🌐📄📄📄	Open page in new browser tab (foreground tab)
Shift+Click on a link or *Ctrl+Click* on a link	Open link in new browser tab (background tab)
Shift+Ctrl+Click on a link	Open link in new browser tab (foreground tab)
Ctrl+Alt+Click on 🌐📄📄📄	Open version in external browser
Ctrl+Alt+Click on a link	Open link in external browser

1.3   Add new bookmarks

1.3.1   Using the Assistant

Clicking the ✳ button in the main toolbar lets you add new bookmarks via the wizard. In a few steps the wizard helps to choose the best configurations dependent from the page type.

In the first step you have to decide, if a web page can be accessed directly or if the page is placed behind a web based login.

- **Page can be accessed directly**
  Simply said, a page can be accessed directly when you can open it in your web browser by just entering the URL (without further actions such a pressing a button or something like that). In that case you can enter the page address into the URL field.

- **Page is password protected**
  If a page is password protected and located behind a web based login, then you have to record a Check-Macro. A Check-Macro contains all steps of the login and is executed when you check the bookmark for updates.

- **Page is the result of a search query**
  If a page can only be displayed after filling out a form (for example after performing a search query), you have to record a Check-Macro. Technically said, in many cases the result of a search

query is displayed using the POST command instead of the GET command, a Check-Macro saves all these steps and POST commands.

After clicking the Next button, WebSite-Watcher performs a check and initializes the bookmark. In the next step you have to choose the page type which is the reference for all further wizard actions to choose the best options and configurations for the bookmark.

> 1. Click the Wizard button ☀ in the main toolbar
> 2. Enter the URL you want to monitor (or create a Check-Macro for password protected pages)
> 3. Click "Next" to initialize the page
> 4. Select the page type (if not detected automatically)
> 5. Follow the next steps

1.3.2   Add bookmarks manually

Alternatively to the Wizard you can add bookmarks directly by clicking the Add button 📄 in the main toolbar (or by pressing the "Ins" key).

You only have to enter the page address into the URL field and press the 'Check Now' button to initialize the bookmark, all other settings have typical default values. If you leave the bookmark name empty, WebSite-Watcher assigns it automatically with the page title.

> 1. Click the Add button 📄 in the main toolbar (or press the "Ins" key)
> 2. Enter the page address into the URL field
> 3. Click the "Check Now" button

1.3.3   Add bookmarks directly from your Browser

A further and comfortable method to add new bookmarks directly from your browser is WebSite-Watcher's browser integration:

- **WebSite-Watcher button in the title bar of your browser**

If WebSite-Watcher is running, then a small button is displayed in the title bar of your browser. Pressing this button pops up a menu from where you can add the currently opened page to WebSite-Watcher. This method works with all main browsers such as Chrome, Firefox and Microsoft Edge.

- **Drag&Drop browser tabs to folder**
A quick and comfortable way to insert new bookmarks within WebSite-Watcher is to drag&drop a browser tab from the internal browser to the folder list.

1.3.4 Import bookmarks

Bookmarks can be imported from different sources ("Tools" menu):

- **Import bookmarks from Excel / CSV**
  See chapter Import from Excel / CSV

- **Import bookmarks from a text file**
  WebSite-Watcher can import URLs from a text file, a description of supported formats can be found in chapter Import from Text file.

- **Import RSS feeds from OPML file**
  RSS/Atom feeds can be imported from an OPML file that can be saved locally on your disk or somewhere in the web.

---

1. Click "Tools" the main menu

2. Select "Im/Export"

3. Select the source from where you want to import bookmarks

1.3.5 Import URLs from Text file

WebSite-Watcher can import URLs from text files that have one of the following formats:

- **One URL per line**
  Each URL must be placed in a single line, for example:
  ```
  https://www.website-watcher.com
  https://www.aignes.com/support.htm
  https://www.aignes.com/videos.htm
  ```

- **One Name and URL per line**
  Each line contains Name and URL which are separated by a tabulator character, for example:
  ```
  WebSite-Watcher{TAB}https://www.website-watcher.com
  WSW Support{TAB}https://www.aignes.com/support.htm
  Videos{TAB}https://www.aignes.com/videos.htm
  ```

The file must be stored in Ansi or UTF8 format.

---

1. Click "Tools" in the main menu

2. Select "Im/Export"

3. Select "Import all/new bookmarks from Text file"

4. Select the location and the format of the text file

5. Press OK

1.3.6   Import from Excel / CSV

Import from Excel / CSV

Bookmarks (incl. bookmark properties) can be imported from the first sheet of an Excel workbook or an CSV file. Supported file extensions are **xls**, **xlsx** and **csv**.

The first row must contain the field names that should be imported. The following rows must contain the values of the bookmark properties (one bookmark per row).

The following screenshot shows a typical Excel sheet:

You only have to specify the fields that you want to import. The smallest sheet would be a table with only one column that contains the URL field.

Supported fields:

General	*Topic*
Field	*Description*
wswid	Unique WebSite-Watcher bookmark ID that is written when you export bookmarks. If this ID is available when importing bookmarks, then WebSite-Watcher will use that ID to identify a bookmark. Otherwise a combination of URL/check_macro/postdata is used.
name	Bookmark name
url	URL of the bookmark. If no wswid is available, then this field must be available, otherwise the row will not be imported.
url2	Alternative check URL
check_macro	Source code of a check macro
postdata	PostData if the bookmark is checked via the POST command

Filter	*Topic*
ignore_filter	Manual ignore filter definitions with one filter per line.
watch_filter	Manual watch filter definitions with one filter per line.
ignore_removed_elements	Special filter: Ignore removed elements. Valid values: 0 or 1
ignore_html_tags	Special filter: Ignore all HTML tags. Valid values: 0 or 1
ignore_upper_lower	Special filter: Ignore upper/lowercase. Valid values: 0 or 1
ignore_dropdown	Special filter: Ignore entries in dropdown and list boxes
filter_sort_content	Special filter: Sort page content. Valid values: 0 = no sorting 1 = sort page content by words 2 = sort page content by sentences

Keywords	*Topic*
highlight_keywords	Highlight keywords. One keyword per line.
same_keywords_as_above	Use same keywords as above. Valid values: 0 or 1
update_on_keywords	Updates on keywords. One keyword per line.

AutoWatch	*Topic*
autowatch	Enable AutoWatch. <u>Valid values</u>: 0 or 1
autowatch_check_every_value	Value of the "Check every..." time configuration
autowatch_check_every_unit	Unit of the "Check every..." time configuration <u>Valid values</u>: s = seconds m = minutes h = hours d = days
autowatch_check_every_time	Time value for "Check every X days"
autowatch_max_one_update_per_day	Option "Max one update per day". <u>Valid values</u>: 0 or 1
autowatch_only_check_between	Only check in specified time frame. <u>Format</u>: HH:MM-HH:MM If a cell doesn't use exactly that syntax, then the cell value is ignored.
autowatch_dont_check_between	Don't check in specified time frame. <u>Format</u>: HH:MM-HH:MM If a cell doesn't use exactly that syntax, then the cell value is ignored.
autowatch_check_on_days	Check only on certain days. <u>Valid values</u>: x = don't check, 1=check <u>Format</u>: 7 characters with x or 1, the first character is Monday, the second character is Tuesday, etc. <u>Example</u>: xxxxx11 (check only on Sat and Sun)
autowatch_automatic	Automatic check interval

Check	*Topic*
alert_on_initialization	Alert/Actions on initialization. <u>Valid values</u>: 0 or 1
alert_on_update	Alert/Actions on update. <u>Valid values</u>: 0 or 1
alert_on_error	Alert/Actions on error. <u>Valid values</u>: 0 or 1
alert_on_error_count	Alert after X errors in a row. <u>Valid values</u>: 1-99
check_technology	Technology to check web pages <u>Valid values</u>: 1=Internal 2=Browser: Internet Explorer 3=Screenshot: Chromium 4=Browser: Chromium 5=Browser: Edge 6=Screenshot: Edge
delay_after_page_loaded	Number of seconds for option "Delay after page has loaded". Only in combination with check technology "IE/Browser" or "Screenshot".
scroll_pages_down	Number of pages that should be scrolled down during a check. Only in combination with check technology "IE/Browser" or "Screenshot".

Actions	*Topic*
action_sendmail	Enable bookmark action **Send e-mail**. <u>Valid values</u>: 0 or 1
action_sendmail_type	<u>Valid values</u>: 0 = Send full page 1 = Send simple version 2 = Send only changes
action_sendmail_attach_screenshot	Attach screenshot with highlighted changes. <u>Valid values</u>: 0 or 1
action_sendmail_emails	Alternative e-mails. Multiple e-mails must be separated with a comma or a blank.
action_sendmail_queue_with_priority	Send before other e-mails. <u>Valid values</u>: 0 or 1
action_sendmail_subject	Alternative subject. The subject field supports the same variables as available in the program configuration.
action_sendmail_template	Optional e-mail template.
action_mark_as_read	Enable bookmark action **Mark as read**. If you don't open bookmarks within WebSite-Watcher (eg. when reading changed bookmarks only in your e-mail client), then you should also enable the Mark-As-Read action.
action_export	Enable the bookmark action **Export**. <u>Valid values</u>: 0 or 1
action_export_filename	Export filename. Supported variables can be found in topic Actions
action_export_pageversion	<u>Valid values</u>: 0 = Version with highlighted changes 1 = New version
action_export_format	<u>Valid values</u>: 0 = HTML format 1 = Text format 2 = Complete web page

Options	*Topic*
jump_to_first_change	Jump to the first highlighted change. <u>Valid values</u>: 0 or 1
jump_to_first_keyword	Jump to the first keyword (within highlighted changes). <u>Valid values</u>: 0 or 1
prevent_horizontal_scrolling	Prevent horizontal scrolling. <u>Valid values</u>: 0 or 1
offset_before_change	Offset before the first highlighted change in pixel.
show_removed_content	Show removed elements at the bottom of the page. <u>Valid values</u>: 0 or 1
show_last_change_marker	Show last change marker. <u>Valid values</u>: 0 or 1
dont_check	Enable the option "Don't check". <u>Valid values</u>: 0 or 1

Ignore updates	*Topic*
ignore_updates_is_blacklist	Defines if the entered words/phrases are a blacklist (1) or a whitelist (0). <u>Valid values</u>: 0 or 1
ignore_updates_words	Words/phrases
ignore_updates_min_char_count	Required number of characters. A value larger than zero means that this option is active.

Troubleshooting	*Topic*
troubleshooting_ignore_css	Ignore document CSS definitions. <u>Valid values</u>: 0 or 1
troubleshooting_ignore_text_color	Ignore document text color. <u>Valid values</u>: 0 or 1
troubleshooting_ignore_background_color	Ignore document background color. <u>Valid values</u>: 0 or 1
troubleshooting_dont_ignore_noscript	Don't ignore NOSCRIPT parts. <u>Valid values</u>: 0 or 1
troubleshooting_dont_optimize_offlinebrowsing	Don't optimize for offline browsing. <u>Valid values</u>: 0 or 1

Notes	*Topic*
note	Bookmark note. Multiple lines are supported.
infofield1	Additional info field #1
infofield2	Additional info field #2
infofield3	Additional info field #3

Export to Excel / CSV

WebSite-Watcher can export bookmarks to an Excel or CSV file with the same fields as described above. This makes it for example possible to export bookmarks to Excel, change them in Excel and re-import them into WebSite-Watcher. Some additional fields are only available when exporting bookmarks (for example last_check_date or last_change_date).

1.4   Check bookmarks

1.4.1   Check pages manually

There are various methods how bookmarks can be checked:

1. manually via menu commands or shortcuts
2. automatically with the AutoWatch feature
3. via the integrated scripting language
4. via command line parameters

One of the following methods can be used to check bookmarks manually:

Check all bookmarks in the selected folder

1. Select the folder you want to check

2. Press the "Check" button ✅ (or shortcut F9)
   (Ctrl+Shift+Click onto a folder will also start the check)

Check all bookmarks

1. Select the folder "All bookmarks"

2. Press the "Check" button ✅ (or shortcut F9)

Check selected bookmarks

1. Select the bookmarks you want to check
2. Right click the selection and choose "Check Bookmarks" (or shortcut Ctrl+K)

Check all Hotsites

The shortcut Ctrl+Shift+H selects the folder "Hotsites" and checks all bookmarks in it.

When you start a check, the bookmarks are placed in a check queue and a grey earth ⊚ icon appears in the first column of the bookmark list. If a bookmark is actually checked, a blue globe 🌐 icon is displayed. If the check has finished, the earth symbol will be removed.

1.4.2 Check pages automatically (with AutoWatch)

Alternatively to manual bookmark checks, WebSite-Watcher supports an automated checking method (called **AutoWatch**) to monitor bookmarks automatically via predefined time configurations.

How to use AutoWatch

1. Enable AutoWatch for specified bookmarks
2. Start the AutoWatch mode

How to enable AutoWatch for a bookmark

1. **Bookmark properties**
   Open the bookmark properties (by right clicking a bookmark) and select the AutoWatch tab. In that section you can enable AutoWatch and optionally enter additional time restrictions when and how often a bookmark should be checked. To perform that task for multiple bookmarks, right click a bookmark selection and run "Set properties in all selected bookmarks..."

2. **Drag&Drop**
   Drag&Drop bookmarks to the AutoWatch folder to enable AutoWatch with the already defined time configurations. This action will not move bookmarks into that folder, just enable the AutoWatch feature.

3. **Pre-defined configuration**
   In the program configuration you can pre-define an AutoWatch configuration for new bookmarks (button "Assign bookmark properties"). This can also be done per folder via the folder properties.

All bookmarks with AutoWatch enabled are listed in the AutoWatch folder.

Starting the AutoWatch mode

Click the AutoWatch button 🕐 in the main toolbar to start the AutoWatch mode. Bookmarks are only monitored automatically when this button is checked.

1.4.3   Check disabled bookmarks

A bookmark can be disabled in the bookmark properties, that means it won't be checked during a normal check run. A bookmark check can also be skipped if you have entered some time configurations, for example the configuration "Check every 7 days".

The feature "Check selected bookmarks (extended)" offers you a way to bypass these restrictions.

> 1. Select the bookmarks you want to check
> 2. Click "Check + Check selected bookmarks (extended)"
>   or use the shortcut Ctrl+F9
> 3. Click the OK button

1.4.4   Status and error messages

The status column in the bookmark list shows the result of the last check. If there was an error, the status is displayed in blue.

Successful check

- **OK**
  If "OK" is displayed, then the check was successful. OK can also be displayed in combination with other information.

Successful check with information

If "OK" is displayed in combination with at least one of the following messages, then the page could be

checked correctly.

- **Initialized**
  When a bookmark is checked the first time, it is initialized. WebSite-Watcher downloads the page and collects all data which are necessary to detect updates with each further check.

- **Warning: Frameset**
  The checked page has a frameset and WebSite-Watcher must download all frames which are defined in that frameset. It is only a warning and indicates that it would be more efficient when you check only the frame you are interested in (if possible, it does not work with all framesets). This warning can also be turned off in the bookmark properties.

- **Redirection**
  A redirection happens when you enter an address into the URL field of your web browser and it automatically "jumps" to another page address (the address in the URL field will also be changed automatically). If such a redirection happens, this warning message is displayed.

- **Warning download size**
  A typical web page has a size between 10 and 100 KB, sometimes even more. If WebSite-Watcher checks and downloads a page with a very large size (for example 1 MB or more), it could be possible that the page redirects to a binary file that would be handled as normal web page then. If the downloaded amount of data is larger than a predefined size (entered in the program configuration), then you get such a warning. Please note that this is only a warning that won't abort the check.

- **Skipped**
  The bookmark was not checked due to some time restrictions in the bookmark properties. You can use "Check Bookmarks (extended)" to check such a bookmark.

- **Disabled**
  The bookmark has been disabled in the bookmark properties. You can use "Check Bookmarks (extended)" to check disabled bookmarks.

Check with errors

If at least one of the following messages is displayed in the status column, then the page could not be checked correctly.

- **Warning: whole content filtered (please verify your filter settings)**
  Normally WebSite-Watcher compares the filtered text content of the new version with the filtered text content of the old version of a page. If the whole content is filtered, then WebSite-Watcher has nothing to compare and is not able to detect updates. In such a case this error message is displayed. Open the bookmark properties and verify/change your filter definitions to fix that problem. With the feature "Test filter" you can see the filtered content that is used to detect

updates.
Related topic: What are filter settings (Overview).

- **Redirection via HTML commands not followed**
  HTML redirections are supported, but not enabled by default. If that message is displayed, you can try to change the bookmark URL to the address of the redirected page, or you enable HTML redirections in the bookmark properties. In rare circumstances HTML redirections are required, for example when a permanent URL redirects to an URL that contains a daily date or a session id.

- **Page not available?**
  The page was not available when it was checked. The question mark indicates that it is not ensured that it is a permanent error. You have to verify and decide yourself if if the page was temporarily not available or if the page was completely removed.

- **Access forbidden?**
  If you receive this error message, then a page is usually protected via Server Authentication and you need username and password to access the page. Username/password can be entered in the bookmark properties, see also chapter Server Authentication.

- **Timeout**
  The page could not be checked within the defined time frame. This can for example happen when a page is temporarily not available or when you check a very large page that cannot be transferred within the given time. The timeout value can be changed in the program configuration or individually for each bookmark in the bookmark properties.

- **Error downloading page**
  The page is available but could not be downloaded or the download was truncated. This is mostly a temporary server problem and this issue should be resolved automatically when you re-check a bookmark at a later time.

- **Error: Page or file not found**
  The page was not available for an unknown reason. For example due to server problems, problems of your internet service provider, etc.

1.4.5   Exclude bookmarks from checking

Bookmarks can be "disabled". Then these bookmarks are stored in the bookmark list but not checked.

- **Exclude from checking permanently**
  To disable bookmarks and exclude them permanently from checking, open the Advanced section in the bookmark properties and enable the option "Disable bookmark (Don't check)" under Options.

- **Exclude from checking temporarily**
  To exclude bookmarks temporarily from checking (for example for the next 7 days), right click a

bookmark selection, then chose "Extras" and "Disable check". Here you can enter for how long a bookmark should not be checked.

How to re-enable disabled bookmarks

- **Permanently excluded bookmarks**
  Permanently excluded bookmarks can be re-enabled in the bookmark properties by unchecking the option "Disable bookmark (Don't check)".

- **Temporarily excluded bookmarks**
  Temporarily excluded bookmarks can be re-enabled the same way as they were disabled. Right click a bookmark selection, then choose "Extras" and "Disable check". Then just enter "0 minutes" and click OK.

1.5 Open changed bookmarks

1.5.1 How are changed bookmarks displayed

Changed bookmarks are displayed in red and bold, they are also automatically sorted to the top of the bookmark list. Opening a changed bookmark will change the status back to "read" and the bookmark will be displayed in black and normal font.

Folders with changed bookmarks are displayed in red and bold, the number of changed bookmarks are displayed within brackets.

1.5.2 How to open (changed) bookmarks

Open the next changed/unread bookmark

- The shortcut F12 opens the next unread bookmark
- The button ⬇ opens the next unread bookmark
- A double click opens the selected bookmark

Navigation within opened bookmarks

- The shortcut Ctrl+F12 jumps to the next highlighted change
- The button 📥 jumps to the next highlighted change
- The Space key scrolls the opened page down
- Typical navigation keys such as example PageUp/Down, arrow keys, etc.

1.5.3 Internal Tabbed Browser

When you start WebSite-Watcher, the internal browser is closed by default and automatically opened as soon as you open a bookmark. The internal browser is an embedded Chromium browser, you can open checked bookmarks as well as online web pages.

WebSite-Watcher will always use the first browser tab to display changed bookmarks. Links can be opened in additional browser tabs.

Open pages in new browser tabs

Pages will be opened in new browser tabs when:

- Holding the Shift key pressed while clicking a link
- Via the context menu when right-clicking a link
- Holding the Shift key pressed while clicking on one of the following buttons:
- Opening a new empty browser tab with the shortcut Ctrl+T

New browser tabs can optionally be opened in foreground or in background. This behavior can be changed in the program configuration ("Internal Browser" tab) or quickly by right clicking a browser tab.

If **Ctrl+Shift** is pressed while clicking a link, then the new tab will be opened in foreground if the option "Open new tabs in background" is enabled and vice versa.

Close opened browser tabs

Opened browser tabs can be closed with:

- Right clicking a browser tab
- Using the shortcut Ctrl+W
- All read browser tabs can be closed with Ctrl+Shift+W

Open in external web browser

The button opens the currently opened page in your web browser.

If you hold Ctrl+Alt pressed while clicking onto a link or one of the buttons , then the page will also be opened in your external default web browser.

1.6 Filter: Ignore / Watch specific parts of a page

1.6.1 What are filter settings (Overview)

To detect updates, WebSite-Watcher compares the text content of the new version of a page with the text content of the old version of a page. If these contents are different, WebSite-Watcher alerts an update. If a page contains changing parts (for example a daily date, some kind of advertisement, ...), an update would be reported with each check. A filter can ignore specific parts of a page to prevent false update notifications.

If a filter is defined, WebSite-Watcher will compare the **filtered text content of the new version** with the **filtered text content of the old version** of a page to detect updates.

Filter definitions are stored in the bookmark properties. The button "Test filter" is available on various places and opens a dialog that displays the filtered text content of the new version and the filtered text content of the old version.
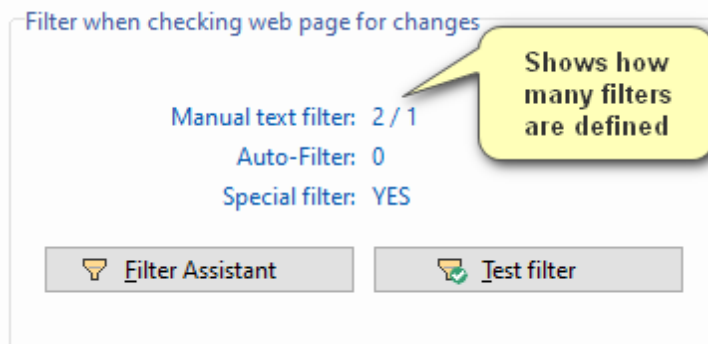
Filter types

1. **Automatically created Ignore filters (Auto-Filter)**
   The Auto-Filter feature is a self learning filter system and provides an easy and comfortable way to automatically create a set of Ignore filters based on highlighted changes. You should only call this filter when you get a false positive, for example caused by a changing date, a changing visitor counter, etc. Don't call this feature when a page contains any other highlighted news or changes.

2. **Ignore filters**
   This filter must be created manually and ignores specified parts of a page, for example a text with changing numbers, a daily date or everything from the page beginning to a defined text.

3. **Watch filters**
   This filter must be created manually and watches specified parts of a page and ignores everything else. It can for example be used to only monitor the software version number on a download page.

4. **Special Filters**
   WebSite-Watcher has also some built-in filter options that must only be turned on/off, for example to sort all words in a page.

In most cases it works best to define a set of Ignore filters, either with the Auto-Filter system or manually via the Filter-Assistant.

How to create Ignore/Watch filters

1. **Using the Auto-Filter system**
   Auto-Filters can be created directly in the internal browser by clicking the 🌿 toolbar button. This button is also available in the "Page" sub menu of the browser toolbar.

2. **Using the Filter-Assistant**
   The Filter-Assistant can be started directly in the internal browser (simply hold the Alt key pressed while selecting the text you want to filter) or in the bookmark properties (by clicking the "Filter-Assistant" button). The Filter-Assistant lets you create Ignore/Watch filters interactively in a very comfortably way. Main steps are to select the text you want to filter and then to classify the selected text (eg. the text a permanent text, a text with changing numbers, etc).

3. **Create the filter manually**
   Alternatively to the Auto-Filter system and the Filter Assistant you can create Ignore/Watch filters manually with simple wildcard definitions and regular expressions.

Cosmetic Filter

A further way to simplify the page (before WebSite-Watcher applies Ignore/Watch filters) are Cosmetic Filters. They will completely delete unwanted content from a page and produce a new page which is then processed by WebSite-Watcher.

1.6.2   Auto-Filter

The Auto-Filter feature is a self learning filter system and provides an easy and comfortable way to automatically create a set of Ignore filters based on highlighted changes.

When can the Auto-Filter system be used?

You should only call this filter when you get a false positive, for example caused by a changing date, a changing visitor counter, etc. Don't call this feature when a page contains any other highlighted news or changes.

Use the Auto-Filter system if you have changes like:

Today is May 29th, 2006 - 13:41:56 am

You are visitor number 8,736

Comments (7), Trackback (3)

Don't use the Auto-Filter system if you have changes like:

26-Apr-2006
AM-Notebook 4.0 has been released.
This major update comes with many new formatting features,
table support, formula based spreadsheets.

How to create an Auto-Filter?

Auto-Filters can be created directly in the internal browser by clicking the Auto-Filter button. This button can be found in the Page submenu of the browser toolbar (or directly in the toolbar if configured).
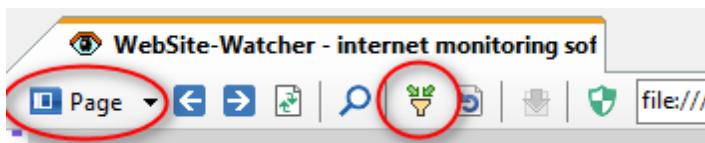
Alternatively you can open the Filter-Assistant in the bookmark properties, select the Auto-Filter tab and click the button "Create Auto-Filter (Ignore highlighted changes)".

The learning process

After clicking the Auto-Filter button, WebSite-Watcher displays the following dialog that shows the number of learned changes and the number of learned filters:

Auto-Filter (self learning filter system)   ✕

| Info | Filter (6) | Changes (23) | Options |

Number of learned changes: 23
Number of learned filters: 6

When should I create an Auto-Filter?

🌱 Save learned items   Cancel   Help
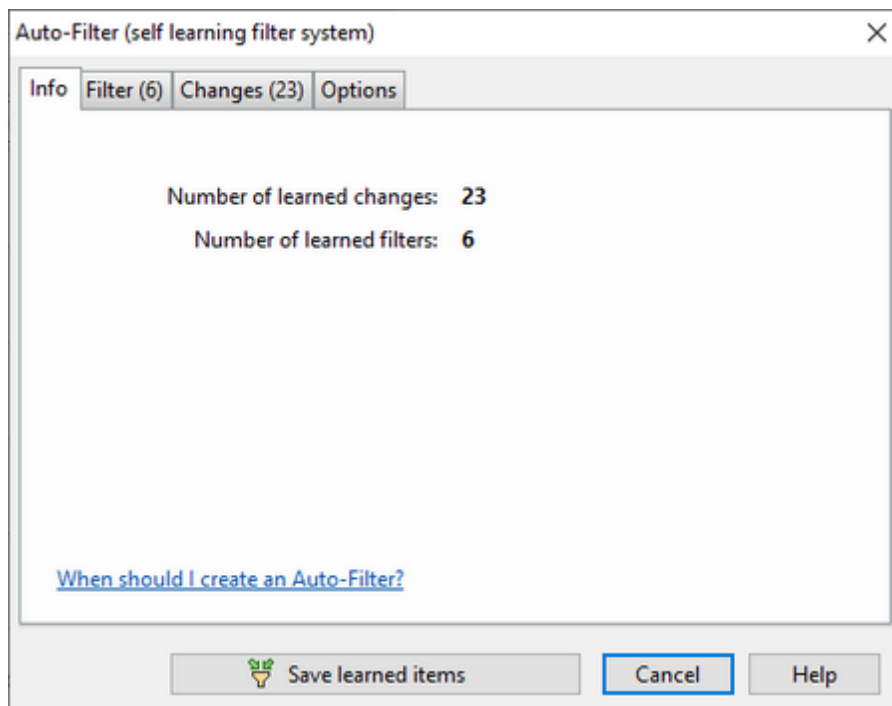
Sometimes a filter can be created right after calling the Auto-Filter feature. In other cases, changes are only learned and a filter might be created after a few more learning processes. It's dependent from the type of a change if and when a filter can be created. Under certain circumstances, WebSite-Watcher is unable to create an Auto-Filter.

Auto-Filter cannot be created

If WebSite-Watcher isn't able to create a filter automatically, you have to use the Filter Assistant to create a filter manually. Manually defined filters have more priority and are executed before Auto-Filters.

Automatically created filters are deleted automatically when they are no longer be used. There's no need to maintain these kind of filters manually.

1.6.3   Filter-Assistant

The Filter-Assistant lets you create Ignore/Watch filters interactively in a very comfortably way. Main steps are to select the text you want to filter and then to classify the selected text (eg. the text a permanent text, a text with changing numbers, etc).

The Filter-Assistant can be started directly in the internal browser (simply hold the Alt key pressed while selecting the text you want to filter) or in the bookmark properties (by clicking the "Filter-Assistant" button).

Filter types

There are two types of filters that can be created with the Filter-Assistant:

- **Ignore filters**
  An Ignore filter ignores specified parts of a page, for example a text with changing numbers, a daily date or everything from the page beginning to a defined text.

- **Watch filters**
  A Watch filter watches specified parts of a page and ignores everything else. It can for example be used to only monitor the software version number on a download page.

Creating a filter in the internal browser

- **Step 1: Select the text you want to filter**
  (for example: "Downloads today: 3,333")

  The selected text should be as unique as possible. A unique text can easily be identified by the filter system.

  If you would only select "3,333", then WebSite-Watcher would find and filter all numbers with separators. The combination of the text "Downloads today:" and the number makes the selection unique and can exactly be identified and filtered.
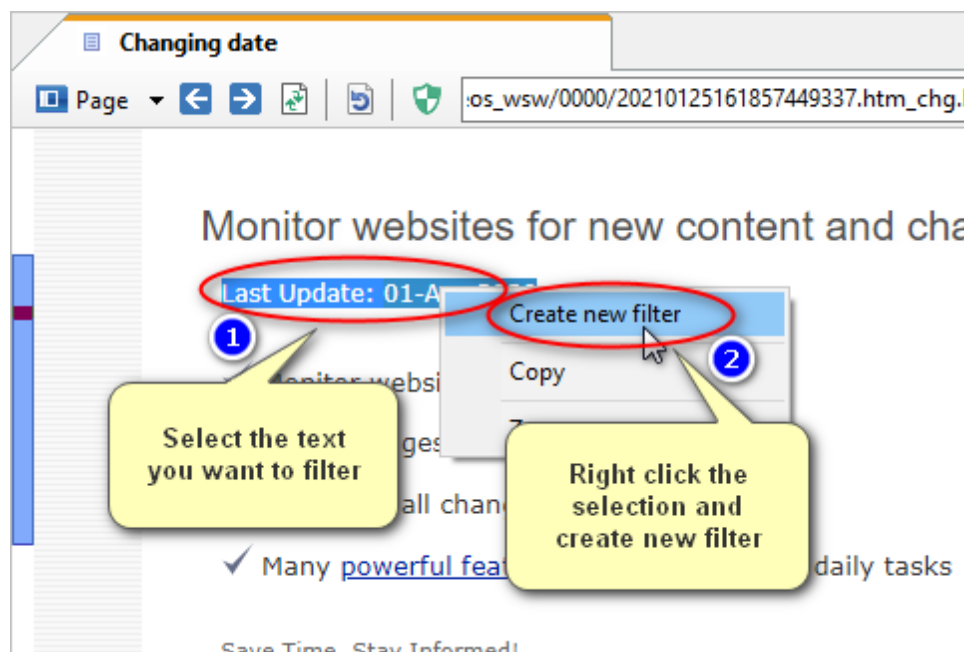
After selecting the text, click the "Page" button and click "Create new filter" to start the Filter-Assistant. If you hold the "Alt" key pressed while selecting the text, the Filter-Assistant will be started automatically.

- **Step 2: Select the filter type you want to create**
  (in our case: Ignore selected text with changing numbers)

  After the Filter-Assistant popped up, you have to select the filter type you want to create. You can change between the Ignore/Watch-Filter view and select the filter type by clicking onto one of the buttons.
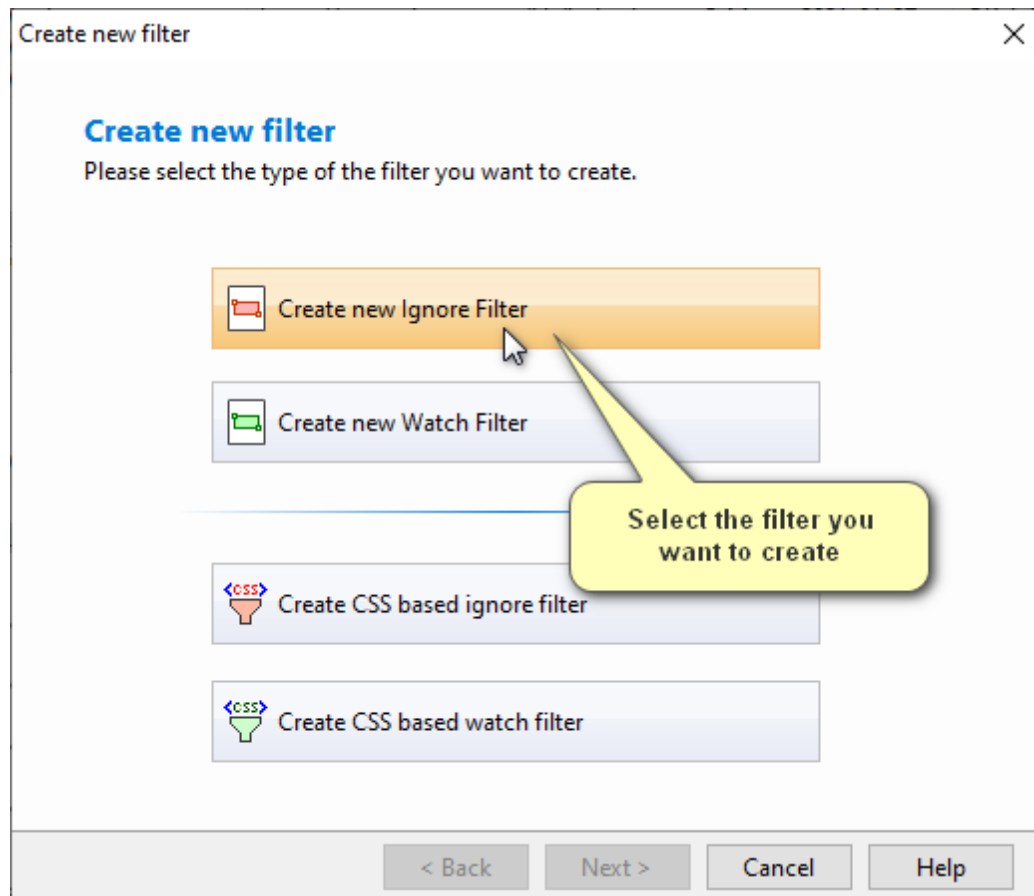
Based on the selected filter type, the assistant will guide you through the rest of the creation process. The next step is to classify the selected text, on our case it is a text with changing numbers.
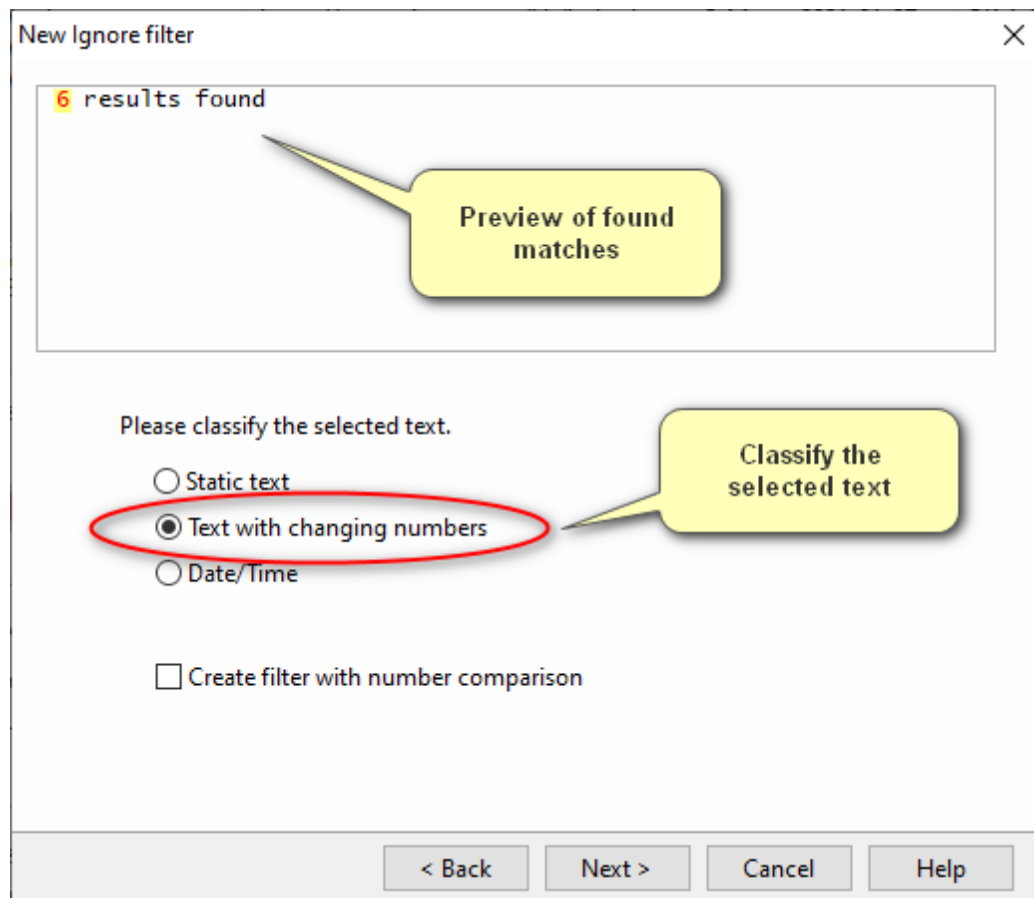
After you have classified the selected text, WebSite-Watcher will display all matching results for verification before the filter is finally created.

There can be zero results when WebSite-Watcher was not able to find the selected text, one result if the selection was a unique text, or more results if WebSite-Watcher was able to find more matches which are "compatible" with the created filter.

If WebSite-Watcher doesn't find a result, then you should try to make a more unique selection.

The order of filters is important!

Filters are always executed from top to bottom, that means that the filter in the first line is executed before the filter in the second line, and so on.

The content that is filtered with the first filter will then no longer be available for filter definitions in the following lines.

Example:

You have defined the following two ignore filters:

```
Watcher
WebSite-Watcher
```

The first filter ignores (deletes) all words "Watcher". The second filter will never find a match since "Watcher" is no longer available. The correct order for these two filters would be:

```
WebSite-Watcher
Watcher
```

Here the first filter ignores (deletes) all words "WebSite-Watcher". The second filter can then ignore all remaining words "Watcher".

Options

- **Convert links to text**
  If that option is enabled, WebSite-Watcher will convert all links to text. This might change the layout a bit, but it makes it much easier to select text within links.

Outdated filters

Outdated ignore filters are automatically detected and deleted after some time when they no longer find any matches. There's no need to delete outdated ignore filters manually. That behavior can be disabled in the Tweaks section (although we do not recommend it).

Outdated watch filters are not deleted automatically, you have to maintain these kind of filters manually.

Don't forget to test your filter settings

The feature "Test filter" lets you always verify your filter definitions by comparing the filtered text content of the new page with the filtered text content of the old page.

1.6.4   Create Ignore / Watch filter manually

Alternatively to the Auto-Filter system and the Filter-Assistant, you can create all Ignore/Watch filters manually.

Enter one filter expression per line. All filter definitions are case insensitive, they are converted to lowercase internally.
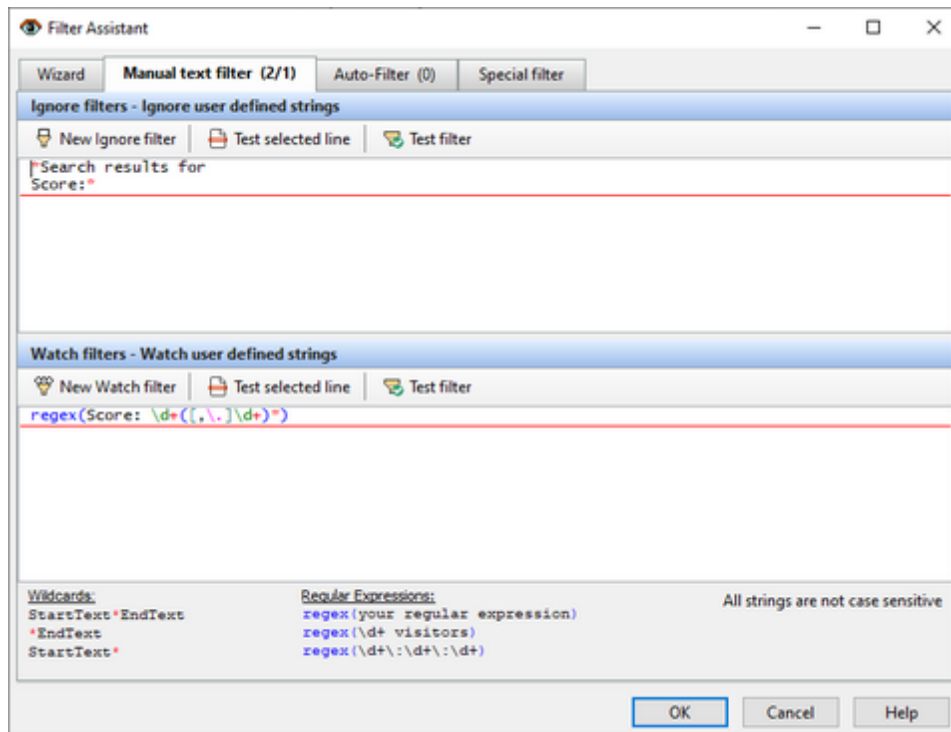
The button "**New Ignore Filter**" (or "New Watch Filter") opens a helper dialog where you can enter and validate a new filter expression. The button "**Test selected line**" lets you validate the filter expression in the active line. The "**Test filter**" button tests all filter definitions by comparing the filtered text content of the new page with the filtered text content of the old page.

Each line can contain one of the following elements:

1. **Static text phrase**
   You can use that type if you want to ignore specific phrases in a page.

2. **Text with a Wildcard**
   A wildcard filter is typically used to ignore areas of a page with a specified start/end text.

3. **Regular Expression**
   With regular expressions you can define complex filter expressions which must be placed in special function names, for example "regex(....)".

Wildcard filter

WebSite-Watcher supports three types of wildcard filters with the restriction that only one Wildcard is allowed per filter expression:

- ***EndText**
  This form filters everything from the page beginning to the first occurrence "EndText".
  For example: *Daily News

- **StartText*EndText**
  This form filters all text areas which begin with "StartText" and end with "EndText".
  For example: Downloads:*Publisher

- **StartText***
  This form filters everything from the last occurrence of "StartText" to the end of the page.
  For example: Users online*

Regular Expressions

WebSite-Watcher supports PERL5 compatible regular expressions which can be used to create complex filter definitions. Regular Expressions must be placed in one of the following functions:

- **regex(...)**
  Filters the defined regular expression
  For example: regex(\d+ downloads)

- **FirstRegex(...)**
  Filters only the first occurrence of the defined regular expression
  For example: FirstRegex(\d+ downloads)

- **StartToRegex(...)**
  Filters everything from the page beginning to the first occurrence of the defined regular expression
  For example: StartToRegex(\d+ visitors)

- **RegexToRegex(... , ...)**
  Filters everything between two regular expressions
  For example: RegexToRegex(Downloads\: \d+,License\:)

- **RegexToEnd(...)**
  Filters everything from the last occurrence of the defined regular expression to the end of the page
  For example: RegexToEnd(\d+ users online)

- **RegexCmp(...)**
  Finds a defined regular expression, extracts all digits from the result and compares them with a pre-defined number. This can for example be used to extract and compare prices. Eg. to only find a match when a certain price is higher than 1000.
  For example: RegexCmp(\d+([,\.]\d+)* Euro;,; > 1000)

CSS based filters

Page content can also be ignored or watched by CSS class names.

- **css(...)**
  Ignores/Watches all text content that is formatted with the defined CSS class or id. Concrete syntax and behavior of css based filters see table below.

Syntax of CSS filters

WebSite-Watcher supports different type of CSS filters, the main syntax is **css(NAME).** This type is also created and used by the Filter Assistant.

Syntax	Description
c	Matches all elements with id="name" or class="name"
	This is the default filter type that is also used by the Filter Assistant. It should be sufficient for most use cases.
	Example: css(header-menu)
	will match: <div id="header-menu"> or <h1 class="header-menu">
c	Matches all elements with id="name"
	Example: css(#header-menu)
	will match: <div id="header-menu">...</div>
c	Matches all elements with class="name"
	Example: css(.top-bar)
	will match: <div class="top-bar">...</div>
c	Matches all elements with class attribute containing "name" (will match class="name name2")
	Example: css(.dropdown*)
	will match: <div class="left dropdown">...</div>

The order of filters is important!

Filters are always executed from top to bottom, that means that the filter in the first line is executed before the filter in the second line, and so on.

The content that is filtered with the first filter will then no longer be available for filter definitions in the following lines.

Example:

You have defined the following two ignore filters:

```
Watcher
WebSite-Watcher
```

The first filter ignores (deletes) all words "Watcher". The second filter will never find a match since "Watcher" is no longer available. The correct order for these two filters would be:

```
WebSite-Watcher
Watcher
```

Here the first filter ignores (deletes) all words "WebSite-Watcher". The second filter can then ignore all remaining words "Watcher".

Outdated filters

Outdated ignore filters are automatically detected and deleted after some time when they no longer find any matches. There's no need to delete ignore filters manually. That behavior can be disabled in the Tweaks section (although we do not recommend it).

Outdated watch filters are not deleted automatically, you have to maintain these kind of filters manually.

Don't forget to test your filter settings

The feature "Test filter" lets you always verify your filter definitions by comparing the filtered text content of the new page with the filtered text content of the old page.

> 1. Open the bookmark properties
> 2. Click the "Filter-Assistant" button (or the text "Manual text filter")
> 3. Enter the filter definitions

Related topics
  • Wildcards
  • Regular Expressions

1.6.5   Special Filter

WebSite-Watcher has also some built-in filter options that must only be turned on/off. These filters work for the whole page and not only for specified parts as the Ignore/Watch filters do.

  • **Ignore all HTML Tags**

    This filter is enabled by default and returns only the real, readable text content of a page while all HTML tags are ignored. This filter should only be disabled under special circumstances, for

example if you want to monitor any changes in the page source code.

The source code "this is a bold word" contains the HTML-Tags and which let your browser display the word "bold" in bold. This filter removes the HTML-Tags, the result is the text "this is a bold word".

- **Ignore Upper/Lowercase**

  This filter is enabled by default and ignores all upper/lowercase changes. If you want to be notified on capitalization changes, then you have to disable this filter.

- **Ignore entries in dropdown and list boxes**

  If this filter is enabled, then WebSite-Watcher ignores all entries in drop down boxes and in list boxes. Disable this filter if you want to monitor these entries. Be aware that there is no way to highlight new or changes entries. <u>Technical description</u>: Ignore everything between the HTML tags <option> and </option>

- **Sort page content**

  This filter sorts all words or sentences in a page and was implemented for some special situations. If you watch for example a search result where the order of found items is changed each time, then WebSite-Watcher would alert an update with each check. This filter sorts all words or sentences in the page and you won't get an update notification when only the order of the found items is changed.

- **Ignore removed elements (alert only if content has been added/changed)**

  If this option is enabled, you will be notified only if page content has been added or modified but not, when content has only been removed.

- **Watch all changes in link addresses**

  If you want to know if link addresses in a page are changed, then you have to use this filter. Link addresses are usually ignored by the filter "Ignore all HTML-Tags". Extracted link addresses are automatically inserted at the end of the page.

- **Watch all changes in image addresses**

  If you want to know if image addresses in a page are changed, then you have to use this filter. Image addresses are usually ignored by the filter "Ignore all HTML-Tags". Extracted image addresses are automatically inserted at the end of the page.

- **Minimum number of new/changed words**

  By default, WebSite-Watcher alerts an update if at least one change is detected. With that option you can define the number of new/changed words that must reached to get an update notification. The window "Test filter" versions" displays the number of new/changed words. If configured, all removed content is ignored and only new and changed words are used to detect an update.

1.6.6   Cosmetic Filter

This is a very powerful feature that allows you to completely remove unwanted content from a page. Cosmetic filters are applied before the page is processed by WebSite-Watcher. Result is or can be a (much) smaller page with only the content you want to monitor.

This feature can also be used to remove floating banners that hide other elements, for example cookie banners, login panels, etc.

Include/Remove filters

To remove content from a page, you have to create include and remove filters. Remove filters will remove all found matches. Ignore filters will use all found matches and ignore everything else.

These filters are based on the used CSS definitions, so you typically have to analyze the HTML source code to get these definitions. WebSite-Watcher simplifies this part with a simple 3-step assistant:

1. **Select text**
   Search and select words in a text for which you want to create a filter. This works best with some unique words (or very short phrases which are not splitted by HTML tags). Clicking "Next" will find all related CSS definitions.

2. **Select CSS definition**
   Select and test the found CSS definitions. It should include the whole text you want to remove or include.

3. **Create the filter**
   Select if you want to create a Remove filter or an Include filter. Test your filter carefully, then click "Add filter" to create it.

This is a very powerful feature that removes content from the page. Please test the selected CSS filters carefully so that no important content will be removed.

2-step execution

In the first step, WebSite-Watcher will execute all Include filters.
In the second step, WebSite-Watcher will execute all Remove filters.

This allows you to define parts of the page you want to monitor and then remove unwanted content from these parts.

Syntax of cosmetic filters

Remove filters can by entered directly, Include filters have to be placed between brackets.

There are mainly 3 type of rules:

- **#name**
  Matches all elements with id="name"
  Example: #header-menu
  will match: <div id="header-menu">...</div>

- **.name**
  Matches all elements with class="name"
  Example: .top-bar
  will match: <div class="top-bar">...</div>

- **.name***
  Matches all elements with class attribute containing "name" (will match class="name name2")
  Example: .dropdown*
  will match: <div class="left dropdown">...</div>

Any line that starts with an exclamation mark is a comment.

Examples

- Example of Remove filters:
  #left-menu
  !this is a comment, the next filter will remove the cookie banner
  .consentbanner

- Example of Include filters:
  (#breaking-news)
  (.top-stories)

See also:
- Ignore Cookie Banner

1.6.7 Screenshot filter

If you monitor a page with the Screenshot method, you can use the following filters:

1. Crop an area of the page that you want to monitor

2. Define one or more ignore filters to exclude certain parts

3. Define how many pixels must have changed to trigger an update notification

See also:
- Check technologies
- Check by screenshot

1.7   Page type dependent settings

1.7.1   RSS / Atom feeds

WebSite-Watcher automatically detects and converts RSS/Atom feeds into a readable format. You can then use the same features that are available for web pages, for example to monitor RSS feeds for specified keywords.

By default, no special configuration is required. If WebSite-Watcher cannot identify an RSS/Atom feed automatically, then you can try to change the file type setting in the bookmark properties from "auto detect" to "rss".

1.7.2   Facebook/Twitter/Instagram pages

Facebook, Twitter and Instagram pages can automatically be converted to RSS feeds which makes it much more easier to monitor these kind of pages. The bookmark properties will show a "Convert page to RSS" option that is automatically enabled for new Facebook/Twitter/Instagram pages.

For Facebook pages, you have to login to your Facebook account once. This will write a cookie with your login credentials that will then be used to monitor Facebook pages.

1. Right click a Facebook bookmark in the main window of WebSite-Watcher
2. Select "Browser", this will open a browser window with the Facebook page
3. Login to your account
4. Close this browser window

Hints:
- Checking a Facebook page can take up to approx. 15 seconds, there is unfortunately no way to improve the performance.
- WebSite-Watcher will only alert new articles, not changes in existing articles.
- WebSite-Watcher can only convert articles that are visible in the page. If you don't check a Facebook page frequently, the RSS feed might not display all postings (but you'll at least know that new postings are available).

1.7.3   Table based forums

WebSite-Watcher supports two different methods to handle web based forums:

1. Forum-Plugins
2. Forum-Templates

Forum-Plugins

Pre-defined Plugins are available for major forums with default layout (such as phpBB, SMF, etc) and can be selected in the bookmark properties or via the browser information bar.
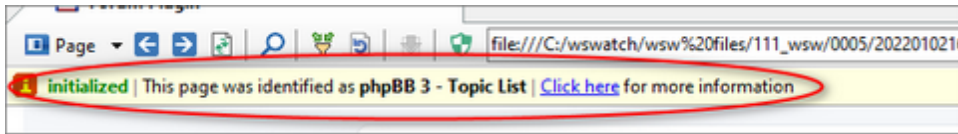
WebSite-Watcher performs various compatibility tests when selecting a Forum-Plugin to ensure that it can be used with the selected bookmark.

Forum-Templates

WebSite-Watcher supports Forum-Templates for table based forums, especially for topic lists. Such a topic list must contain:

1. **A table header**
   The header tells WebSite-Watcher, where the topic list starts, for example the line:
   Topics | Replies | Author | Views | Last Post

2. **A "Last Posting" time (per topic)**
   Each thread/topic must have a last posting time, for example "09:13 AM". It does not work when a forum shows "13 minutes ago" instead of a time.

3. **A table footer**
   The footer tells WebSite-Watcher, where the topic list ends. The footer may contain changing numbers, for example "Mark forum read" or "Page 1 of 13".

If a forum fulfills all these requirements, you can use the Forum-Wizard to create the Forum-Template and the required filter definitions, no further configuration is required in that case.

> 1. Create a new bookmark
> 2. Enter the forum URL
> 3. Click the "Extras" button and select "Forum Wizard" from the sub menu
> 4. Follow the steps in the Forum Wizard

See also chapter Forum

1.7.4   PDF files

Monitor the text content of PDF files

With a special PDF-Plugin, WebSite-Watcher can extract and convert the text content of a PDF file into a HTML page. Then PDF files can be checked and handled like normal web pages and WebSite-Watcher is able to highlight changes in the text. This PDF-Plugin is assigned automatically when adding new PDF documents to the bookmark list.

Unfortunately it's not always possible to extract text from a PDF file. The PDF-Plugins in WebSite-

Watcher use different solutions/methods to extract text. Should one method fail, there's still a good chance that the other methods can extract text from the PDF file.

Monitor PDF files without a Plugin

If no Plugin is used to monitor PDF files, then they are handled as binary files. Dependent from the server log, one or a combination of the following checking methods is used:

- Check the PDF document by file date
- Check the PDF document by file size
- Check parts of the file content

In that case, it's not possible to highlight text changes or to use the filter system to ignore unwanted content.

1.7.5 Word/Excel files

Monitor the content of Word/Excel documents

With a special Plugin, WebSite-Watcher can convert Word/Excel documents into a webpage. Then they can be checked and handled like normal web pages and WebSite-Watcher is able to highlight changes in the text. This Plugin is assigned automatically when adding new Word/Excel documents to the bookmark list.

Use Microsoft Office for the conversion

By default, WebSite-Watcher uses internal conversion routines to convert Word and Excel files into web pages. If Microsoft Office 2003 or higher is installed on your system, then you can also use Microsoft Office to convert these files (better quality especially for Word files). Separate Plugins to use Microsoft Office for the conversion are available.

Monitor Word/Excel files without a Plugin

If no Plugin is used to monitor Word/Excel documents, then they are handled as binary files. Dependent from the server log, one or a combination of the following checking methods is used:

- Check the Word/Excel document by file date
- Check the Word/Excel document by file size
- Check parts of the file content

In that case, it's not possible to highlight text changes or to use the filter system to ignore unwanted content.

1.7.6 FTP files and directories

FTP files and directories

FTP files and directories can be monitored by entering the FTP address into the URL field of a bookmark. Supported are the FTP protocol and the FTPS protocol. FTP addresses must start with ftp:// and FTPS addresses must start with ftps://

If you want to check a single file, you have use the complete address with path and filename. WebSite-Watcher will check the file by file-date and file-size, no content is transferred.

If you want to monitor a complete folder, the last character of the URL must be a Slash / (without a filename)!

How to check the content of a (text) file

With the checking method "automatic" a file is checked by file-date and file-size. If you want to check it by content (for example a text file), you have to change the checking method from "automatic" to "by content". Then the file is downloaded and handled like a web page.

Password protected FTP files / directories

By default WebSite-Watcher performs an anonymous FTP login. If an FTP server is password protected, you have to enter username and password in the bookmark properties (Advanced + Server Authentication).

1.7.7 Images

If the address of an image file is entered into the URL field of a bookmark, the image will be checked by content and WebSite-Watcher is able to highlight changes within the image. No special settings are required in that case.

1.7.8 Binary files

WebSite-Watcher can monitor binary files (for example ZIP or EXE files) without downloading them. Dependent from the server log file, one or a combination of the following checking methods is used:

- Check by file date
- Check by file size
- Check parts of the file content

Normally no special settings are required for binary files. If WebSite-Watcher cannot identify a binary file automatically (for example when a script redirects to a binary file), then you have to change the file type from "auto detect" to "binary".

1.7.9   Local files and folders

WebSite-Watcher is able to monitor local files and folders by entering the absolute path into the URL field of a bookmark, for example "c:\docs\judo.doc" or "d:\docs\".

Monitor a directory with subdirectories

If you want to check a folder with all files and subfolders, you have to use the Follow-Links feature. It's recommended to start with link depth 1 and increase that value if needed.

Directories and the modified-date

If a file in a directory is changed, then the date of that directory is also changed. This will probably give you redundant update notifications that can be avoided as follows:

1. **Create an appropriate filter**
   Create an appropriate filter by using the Filter-Assistant.

2. **Use the folder creation-date instead of the modified-date**
   Alternatively to the default behavior you can display the creation-date instead of the modified-date by entering the following line to the Tweaks section of the program configuration:
   CheckLocalFolderUseCreationDate=1

1.7.10 Check by screenshot

If a page contains content that cannot be monitored with all the available options (eg. content that is displayed with Java), you can try to use the feature "Check by screenshot".

This check technology opens a page in a hidden browser window, makes a screenshot and saves this screenshot as PNG file. This screenshot will then be used to detect updates. WebSite-Watcher is also able to highlight changes in the saved screenshot.
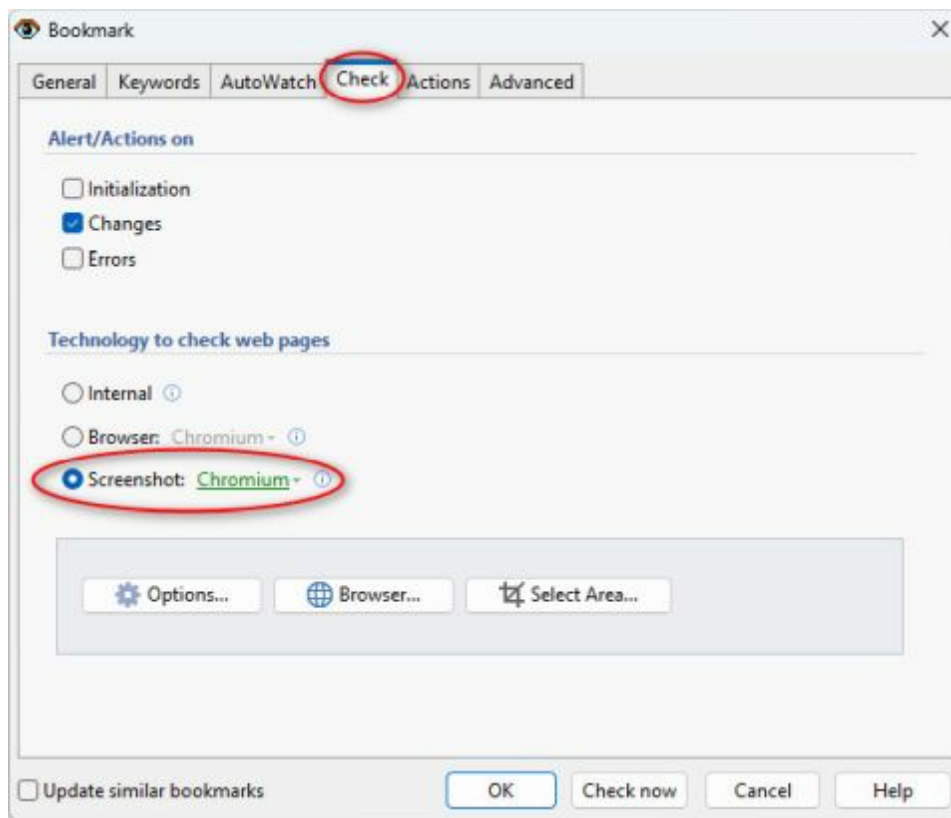
You can optionally define a custom window size and/or the region you want to capture.

Many options will of course not work with that method (eg. the keywords feature), but you have an additional "chance" to monitor content that would not work otherwise.

---

1. Open bookmark properties
2. Select the tab "Check"
3. Select the technology "Screenshot"

1.8   Password protected pages

1.8.1   Login via Server Authentication

Server Authentication is a special method to protect pages from public access. You can identify that type of login, when a small dialog pops up where you have to enter username and password, for example:
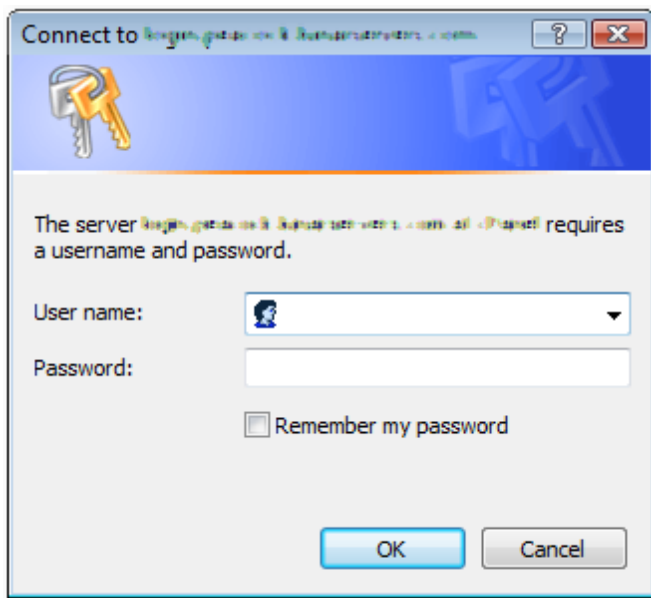
If you get such a dialog, you have to enter username and password in the bookmark properties (Advanced + Server Authentication)

1. Open bookmark properties
2. Select the "Advanced" tab
3. Select "Login" on the left side
4. Under "Login via Server Authentication", enter username and password

If the fields for username and password are placed within a web page, then it is a web based login and you have to record a Check-Macro.

1.8.2   Web based logins / Check-Macros

Web based logins are logins where the fields for username and password are placed within a web page (other than logins with Server Authentication where a small window pops up to enter username and password).

WebSite-Watcher supports a technique called Check-Macros to automate various steps and operations in a page. For example, you can record the steps to login to a page, surf to the page you want to monitor, and perform a logout afterwards. Then, when the bookmark is being checked, these steps will be executed.

Check-Macros can also be used to automate search queries where you enter a search term and press a search button.

Please note: It's unfortunately not possible to automate all kind of logins and operations, this is dependent from the technology used in the page.

See topic Check Macros for more information.

1.8.3 Password protected FTP files

By default WebSite-Watcher performs an anonymous FTP login.

If an FTP server is password protected, you have to enter username and password in the bookmark properties (Advanced + Server Authentication).

> 1. Open the bookmark properties
> 2. Select the Advanced tab
> 3. Select "Login" on the left side
> 4. Under "Login via Server Authentication", enter username and password

1.9 Highlight changes

1.9.1 Highlight methods

When an update is detected, WebSite-Watcher highlights all text changes in the page. WebSite-Watcher supports different methods how text will be highlighted:

- **Standard**

   The standard method is a general purpose method that should work with nearly all types of pages. It has been designed to ignore modifications in the page structure (for example when content is moved to another place) and works very well for news pages. It is a general purpose method while the other methods are optimized for special situations or page types.

- **Table based (highlight whole, changed rows)**
   You can use this highlight method if you want to highlight whole table rows when at least the content of one cell has been changed. To prevent highlighting of a row when only columns with numbers have been changed, you can select the option "Cells with a changed number don't highlight the whole line".

- **Exact**
   The exact method uses a more detailed analysis to detect changes. It may work well with pages where only some details are updated, for example single numbers within a table. It's dependent from the page if that method works better than the Standard method.

> 1. Open the bookmark properties
> 2. Select the Advanced tab
> 3. Select "Highlight changes" on the left side
> 4. Enter the highlight configuration

1.9.2 Highlight changed images in a page

WebSite-Watcher can optionally paint borders around changed images.

How are changed images detected?

It's dependent from the image configuration if WebSite-Watcher is able to paint a border around changed images. WebSite-Watcher supports three different configurations to download and display images:

1. **Download all images (recommended)**

   If an update is detected, WebSite-Watcher downloads all images to your hard disk. With that option WebSite-Watcher uses the image content to detect if an image has been changed. This is the most reliable method to put highlight-borders around changed images.

2. **Use absolute image addresses (not recommended)**
   With that setting, WebSite-Watcher downloads only the web page and loads images from the server as soon as you open the page in your browser. If you use that method, WebSite-Watcher will use the image filename to detect new or changed images.

3. **Display grey boxes instead of images (not supported)**
   This is the fastest method to check and open pages, but you will see grey boxes instead of images. With that option it is not possible to highlight changed images.

Please note that a web page is only marked as updated when a text change has been detected, and not when only an image has been changed!

If you want to watch a concrete image, you can put the image address into the URL field of a bookmark.

1. Open the bookmark properties
2. Select the Advanced tab
3. Select "Images" on the left side
4. Enable the option "Download all images"
5. Select "Highlight changes" on the left side
6. Enable the option "Highlight images"

See also topic Image configuration

1.10 Report / Export

1.10.1 Overview / Templates

The Report/Export feature lets you create your own reports in HTML, XML or any other TEXT format.

With report templates you can create your own individual reports. A report template is a HTML file

(for HTML reports) or a text file (for text exports) that contains several predefined datafields. These datafields are replaced with real values when the report is created.

Reports are always created in UTF-8 format!

How to create a report?

1. Call "Report/Export" in the Tools menu
2. Select the bookmarks that should be included in the report
3. Select a report template
4. Select the output file into which the report should be created
5. Click the button "Start report"

Structure of report templates

Structure of a HTML report template (without folders):

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>your title</title>
</head>
<body>
Your header text

{wsw_begin_url}
  <a href="{wsw_url_url}" target="_top">{wsw_url_name}</a><br>
  {wsw_url_text}<br>
  date: {wsw_url_date_changed}<br>
{wsw_end_url}

<br>
<br>
report date: {wsw_report_date}<br>
</body>
</html>
```

Structure of a HTML report template (with folders):

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>your title</title>
</head>
<body>
Your header text

{wsw_begin_folder}
  Folder: <b>{wsw_folder_name}</b><br><br>
  {wsw_begin_url}
    <a href="{wsw_url_url}" target="_top">{wsw_url_name}</a><br>
```

```
    {wsw_url_text}<br>
    date: {wsw_url_date_changed}<br>
  {wsw_end_url}
{wsw_end_folder}


<br>
<br>
report date: {wsw_report_date}<br>
</body>
</html>
```

1.11 Working with Wildcards and Regular Expressions

1.11.1 Overview

Wildcards and Regular Expressions are supported from various features in WebSite-Watcher, especially by the filter system to ignore or watch specified page content.

WebSite-Watcher supports three methods to filter or to search for text:

1. Concrete words and phrases
2. Wildcards
3. Regular Expressions

All three methods work case insensitive ("HELLO" is the same as "hello").

1.11.2 Wildcards

WebSite-Watcher supports the following three types of wildcard filters with the restriction that only one wildcard is allowed per filter expression:

- ***EndText**

  This form filters everything from the page beginning to the first occurrence of "EndText".
  For example: *****Daily News

- **StartText*EndText**

  This form filters all text areas which begin with "StartText" and end with "EndText".
  For example: Downloads:*****Publisher

- **StartText***

  This form filters everything from the last occurrence of "StartText" to the end of the page.
  For example: Users online*****

All defined wildcard filters are case insensitive, they are converted to lowercase internally.

1.11.3 Regular Expressions

With Regular Expressions you can define complex search and filter expressions. All regular expressions are case insensitive by default.

Regex functions

Regular Expressions must be placed into one of the following functions:

- **regex(...)**

  Filters the given regular expression
  For example: regex(\d+ downloads)

- **FirstRegex(...)**

  Filters only the first occurrence of the defined regular expression
  For example: FirstRegex(\d+ downloads)

- **StartToRegex(...)**

  Filters everything from the page beginning to the first occurrence of the given Regular Expression
  For example: StartToRegex(\d+ visitors)

- **RegexToRegex(... , ...)**

  Filters everything between two Regular Expressions
  For example: RegexToRegex(Downloads\: \d+,License\:)

- **RegexToEnd(...)**

  Filters everything from the last occurrence of the given Regular Expression to the end of the page
  For example: RegexToEnd(\d+ users online)

- **RegexCmp(...)**

  Finds a defined regular expression, extracts all digits from the result and compares them with a
  pre-defined number. This can for example be used to extract and compare prices. Eg. to only find
  a match when a certain price is higher than 1000.
  For example: RegexCmp(\d+([,\.]\d+)* Euro;,; > 1000)
  The regexcmp function can be used in the Keywords functionality, Ignore filters and Watch filters.
  A detailed description can be found below.

Tokens of Regular Expressions

Below you can find a list of useful Regular Expression tokens.

\	The backslash escapes any character and can therefore be used to force characters to be matched as literals instead of being treated as characters with special meaning. For example, '\[' matches '[' and '\\' matches '\'.
.	A dot matches any character. For example, 'go.d' matches 'gold' and 'good'.

{ }	{n} ... Match exactly n times {n,} ... Match at least n times {n,m} ... Match at least n but not more than m times
[ ]	A string enclosed in square brackets matches any character in that string, but no others. For example, '[xyz]' matches only 'x', 'y', or 'z', a range of characters may be specified by two characters separated by '-'. Note that '[a-z]' matches alphabetic characters, while '[z-a]' never matches.
[-]	A hyphen within the brackets signifies a range of characters. For example, [b-o] matches any character from b through o.
|	A vertical bar matches either expression on either side of the vertical bar. For example, bar|car will match either bar or car.
*	An asterisk after a string matches any number of occurrences of that string, including zero characters. For example, bo* matches: bo, boo and booo but not b.
+	A plus sign after a string matches any number of occurrences of that string, except zero characters. For example, bo+ matches: boo, and booo, but not bo or be.
\d+	matches all numbers with one or more digits
\d*	matches all numbers with zero or more digits
\w+	matches all words with one or more characters containing a-z, A-Z and 0-9. \w+ will find title, border, width etc. Please note that \w matches only numbers and characters (a-z, A-Z, 0-9) lower than ordinal value 128.
\s	matches a whitespace (space, tab and carriage return/line feed)
.*?	find as few characters as possible. a.*?b means: "find "a", followed by as few characters as possible, followed by "b
[a-zA-Z\xA1-\xFF]+	matches all words with one or more characters containing a-z, A-Z and characters larger than ordinal value 161 (eg. ä or Ü). If you want to find words with numbers, then add 0-9 to the expression: [0-9a-zA-Z\xA1-\xFF]+

RegexCmp(...)

The RegexCmp function finds a defined regular expression, extracts all digits from the result and compares them with a pre-defined number. If the comparison returns true, the match will be accepted.
This function requires 3 parameters (divided by the ; character), the exact syntax is:

```
regexcmp(regular expression; decimal point character; operator number)
```

Parameters:
- **regular expression**

This regular expression extracts defined numbers from a page. The result can contain characters and numbers, for example a regular expression that finds "Price: 49,00 Euro". The regexcmp function will then extract all digits from the found result and compare the extracted number.

- **decimal point character**
  Defines if a dot or a coma is used as decimal point character in the page. Valid parameter characters are "." and "," (without quotes).

- **operator number**
  valid operators:
  = ... equal
  < ... less than
  <= ... less or equal than
  > ... greater than
  >= ... greater or equal than
  <> ... not equal
  The number defines the number for the comparison and can optionally contain a decimal point character, for example 49,95 or 49.95. Thousands separators are not allowed.

Example:
regexcmp(\d+([,\.]\d+)* Euro;,; > 49.95)

- The first parameter searches the regular expression "\d+([,\.]\d+)* Euro" and extracts all digits from the found result (incl. decimal point character). For example 1449,95
- The second parameter defines which character is used as decimal point character, in that example it's the character ","
- The third parameter compares if the price is higher than 49.95
- If the extracted price is lower or equal than 49.95, then the found match is omitted. If the extracted price is higher than 49.95, then the found match is accepted.

Typical examples

- **regex(bo*)**
  will find "b", "bo", "boo", "booooo"

- **regex(bx+)**
  will find "bxxxxxxxx", "bxx", "bx" but not "b"

- **regex(\d+)**
  will find all numbers

- **regex(\d+ visitors)**
  will find "3 visitors" or "243234 visitors" or "2763816 visitors"

- **regex(\d+ of \d+ messages)**

will find "2 of 1200 messages" or "1 of 10 messages"

- **RegexToEnd(\d+ of \d+ messages)**

  will filter everything from the last occurrence of "2 of 1200 messages" or "1 of 10 messages" to the end of the page

- **regex(MyText.{0,20})**

  will find "MyText" and the next 20 characters after "MyText"

- **regex(\d\d.\d\d.\d\d\d\d)**

  will find date-strings with format 99.99.9999 or 99-99-9999 (the dot in the regex matches any character)

- **regex(\d\d\.\d\d\.\d\d\d\d)**

  will find date-strings with format 99.99.9999

- **regex(([_a-zA-Z\d\-\.]+@[_a-zA-Z\d\-]+(\.[_a-zA-Z\d\-]+)+))**

  will find all e-mail addresses

- **regexcmp(\d+([,\.]\d+)* Euro;,; > 49.95)**

  will find all prices with format "9.999,99 Euro" and only accept results with prices higher than 49,95 Euro

1.12   Bookmark files (database)

1.12.1   Where are bookmarks stored

Bookmark files

Bookmarks are stored in database files with the extension *.wsw, downloaded cache files in the related sub folder *_WSW.

If your bookmark file has the filename bookmarks.wsw, then all downloaded files are stored in the related sub folder bookmarks_WSW.

Bookmark folder

Bookmark databases can be stored in any folder. The default bookmark location is:

- c:\Users\%USERNAME%\AppData\Roaming\aignes\WebSite-Watcher\bookmarks\

To see the exact folder of the currently opened bookmark file, you can call "File + Database + Move bookmark file". This dialog shows the folder path, the bookmark filename and the cache folder.

1.12.2 Move bookmark files to another location

You can use the feature "Move bookmark file" to move a bookmark database to a new location.

1. Click "File" in the main menu
2. Select "Database", then "Move bookmark file"
3. Enter the new location
4. Press OK

This will move all bookmarks, configurations and downloaded cache files to the new location. Dependent from the number of bookmarks, this operation can take a few minutes.

1.12.3 Export/Import bookmarks between bookmark files

WebSite-Watcher offers an integrated Import/Export functionality to exchange single bookmarks between bookmark files. This feature exports/imports bookmarks with all configurations and cache files, but without the folder structure.

This feature is located in the Tools menu under Import/Export:

- Import bookmarks from Export file (*.zip)
- Export selected bookmarks to Export file (*.zip)

To keep two bookmark databases in sync, you should use the Synchronization feature.

1.12.4 Synchronize / Cloud Sync

The Synchronize feature synchronizes the currently opened bookmark database with a local Cloud folder or a user defined local folder on your PC or network. If you want to synchronize your database via Cloud, you have to install the client of the used Cloud service (or alternatively the tool "rclone") that synchronizes a pre-defined local folder with the Cloud. All data saved in the Cloud is encrypted.

When do I need that feature?
For example, if you use WebSite-Watcher on your PC and your notebook, you can easily synchronize your two systems via the Cloud.

Options

- **Sync downloaded images**
  If enabled, WebSite-Watcher will synchronize all images that were downloaded and saved into the local cache folder. Dependent on your configurations, this can dramatically increase the size of the synchronized data.

- **only for unread bookmarks**
  If enabled, WebSite-Watcher will only synchronize downloaded images for unread bookmarks. So unread bookmarks will show correctly on synchronized PCs while read bookmarks will be displayed without downloaded images. As soon as a bookmark check detects an update, WebSite-Watcher

will automatically download missing images. This option can dramatically reduce the size of synchronized data.

Rclone

WebSite-Watcher can use the tool "rclone" to synchronize data with the Cloud. If rclone is installed on your PC, you can enter the path to rclone.exe and the preferred "Remote". No other Cloud client will be required in that case.

"**Remote**" is the profile name of a cloud that you have configured in rclone. If synchronization with rclone does not work correctly, you can temporarily enable the Debug-Mode to get the results and error messages from rclone.

More information and help about rclone can be found at https://rclone.org

If you have not heard about rclone before, we can highly recommend this tool, especially in combination with the CloudSync feature in WebSite-Watcher. Rclone supports many Cloud services, can add extra encryption to your data and works very reliable to synchronize bookmarks between PCs.

1.12.5 Transfer everything from one PC to another PC

Transfer files manually

Bookmarks are stored in database files with the extension *.wsw, downloaded cache files in the related sub folder *_WSW. If your bookmark file has the filename "bookmarks.wsw", then all downloaded files are stored in the related sub folder "bookmarks_WSW".

To transfer all bookmarks between two PCs, you have to copy that file and cache folder to your second PC with a file manager.

Transferred bookmark files can be opened with File+Open on the second PC.

Transfer via Backup/Restore

WebSite-Watcher has an internal Backup/Restore functionality that can be used to copy or transfer program configurations and/or bookmark databases between PCs.

Steps to copy bookmarks and program configurations from PC1 to PC2:

1. On PC1: Click "Tools + Backup/Restore"
2. On PC1: Select "Backup" and follow the assistant. This will create a single ZIP file, in our example it's named wsw_bookmark.zip.
3. Copy the file wsw_bookmark.zip from PC1 to PC2
4. On PC2: Click "Tools + Backup/Restore"

5. <u>On PC2</u>: Select "Restore" and follow the assistant. In a further step the assistant asks for the "Backup file", here you have to enter the file wsw_bookmark.zip from step #2

If you no longer need WebSite-Watcher on PC1, you can deactivate the license in the Help menu (Deactivate license).

Transfer via Synchronization

To keep two existing bookmark databases in sync, you can use the Synchronization feature.

1.12.6 Database errors

Before you apply one of the following steps, it's highly recommended to make a backup of the damaged database (***.wsw** file) and the related cache folder (***_WSW** folder). Just to ensure that you can return should one of these steps fail.

If your bookmark database has physical damages, then you have to restore or repair your database with one of the following steps:

1. Restore a backup

If you have a working and up to date backup of your bookmark database, then it's recommended to restore that backup. This is always the best and most reliable solution. By default, WebSite-Watcher creates an AutoBackup once a day that can be restored with Tools + Backup/Restore.

1. Click "Tools" in the main menu
2. Select "Backup/Restore"
3. Select "Restore AutoBackup" and follow the assistant

2. Run the command "Repair database"

If you don't have a working backup file or your backup file outdated, then you can run the command "Repair database". This will repair your database and restore as many data as possible. After using that command, the next check run might produce some false update notifications, especially if you have disabled the special filter "Ignore all HTML tags".

1. Click "File" in the main menu
2. Select "Database"
3. Run the command "Repair database"

3. Run the command "Re-Create database from cache folder"

This command can be used if the command "Repair database" fails and you don't have a working and up to date backup file. This command re-creates the whole database from files that are stored in your cache folder. After using that command, the next check run might produce some false update notifications, especially if you have disabled the special filter "Ignore all HTML tags".

1. Click "File" in the main menu

> 2. Select "Database"
> 3. Run the command "Re-Create database from cache folder"

After running any of these commands, you can run the command "Optimize database" that will also perform an integrity check.

Database Repair Tool

If it is no longer possible to start WebSite-Watcher or to open a database due to database corruptions and crashes, you can use the application "**dbtool.exe**" that can be found in the WebSite-Watcher setup folder. With that tool you're also able to run the steps described above at a lower level.

1.13   Additional tools and features

1.13.1   Check Macros

WebSite-Watcher supports a technique called Check-Macros to automate various steps and operations in a page. For example, you can record the steps to login to a page, surf to the page you want to monitor, and perform a logout afterwards. Then, when the bookmark is being checked, these steps will be executed.

Check-Macros can also be used to automate search queries where you enter a search term and press a search button.

Please note: It's unfortunately not possible to automate all kind of logins and operations, this is dependent from the technology used in the page.

Technologies

WebSite-Watcher supports different technologies to record and execute check macros. Simply start with one of these methods, if it fails then try the next one.

The "Static macros" are still available for legacy reasons and work only with very simple logins. It's not recommended to use this method for new macros.

Demos and Videos

At https://www.aignes.com/videos.htm you can find some videos which show how to record and use Check-Macros.

> 1. Create a new bookmark
>
> 2. Press the "Check-Macro" button in the bookmark properties
>
> 3. A window pops up, enter the Start-URL of the macro

4. Click the button to start recording

5. Surf to the page you want to monitor

6. Perform a logout if required (and check the appropriate check box to tell WebSite-Watcher that you are performing a logout)

7. Click the OK button at the bottom of the window, this will stop recording and return to the bookmark properties

8. Click the "Check Now" button to check the bookmark

1.13.2 Check Macros / Additional commands

Check-Macros support additional commands that can be added manually to the recorded macro source.

While the syntax of the different macro technologies is different, these additional commands have the same syntax for all technologies.

Command	Description
WAIT: seconds=XXX	Waits the defined number of seconds. Valid values are between 5 and 60 Example (wait 10 seconds): WAIT: seconds=10
SCROLLPAGEDOWN:	Scrolls the page down. If a page should be scrolled down multiple times, you have to use one command per scroll operation. Example (scroll 2 pages down): SCROLLPAGEDOWN: SCROLLPAGEDOWN:
RELOAD:	Reloads the currently opened page.

1.13.3 Extended Search

The Extended Search feature lets you search text information in the page content.

The button "Display in Main Window" transfers the search result to the "Search" folder in the main window, selects that folder and closes the Extended Search.

1. Open "Extended Search" with the toolbar button (or shortcut F7)

2. Enter the search terms

3. Click the "Search" button

Related topics
  • Regular Expressions

1.13.4 Backup / Restore

The integrated Backup/Restore feature lets you backup and restore your program configuration and currently opened bookmark file (inclusive downloaded web pages) to a ZIP file.

> 1. Select Tools in the main menu
>
> 2. Select "Backup / Restore..."
>
> 3. Follow the steps in the wizard

1.13.5 Download Manager

WebSite-Watcher includes an integrated download manager that can be used to automatically download RSS feed enclosures. Supported download protocols are HTTP and HTTPS.

How to download files

1. **Download RSS feed enclosures**
   RSS feed enclosures can automatically be downloaded when an RSS feed has been changed. This option can be enabled in the bookmark properties ("Advanced" section). By default, WebSite-Watcher downloads mp3 and wav files, these extensions can be changed or enhanced in the program configuration ("Advanced" section).

2. **Add a URL manually**
   A download URL can also be added manually by starting the download manager and clicking the "Add Download" button. In WebSite-Watcher, the download manager can be started via the Tools menu.

3. **Download within Plugins**
   The Plugin system supports the command AddToDownloadManager to add a download URL to the download manager. That means that you can extract a specific URL from a page and download it with the download manager.

Where are downloaded files stored?

By default, WebSite-Watcher saves the downloaded files to the following folder:

  • c:\Users\<user>\AppData\Roaming\aignes\WebSite-Watcher\config\downloads\

  -or-

  • c:\Documents and Settings\<user>\Application Data\aignes\WebSite-Watcher\config\downloads\

Automatically delete files after X days

The download manager keeps all downloaded files for a pre-defined number of days, then they are automatically deleted from the list and from your hard disk. Right clicking a file (or file selection) lets you copy or move all selected files to another folder. You can also automatically copy downloaded files to another location based on configured rules (see program configuration within the Download Manager).

Downloaded files will only be deleted from the default download folder. If downloaded files were copied/moved to another location (manually or automatically with the "Copy To" feature), then these copies will of course remain.

1.13.6 Run multiple instances

If you install the desktop edition of WebSite-Watcher, it's only possible to run one instance of WebSite-Watcher.

If you want to run a second instance of Website-Watcher on the same PC, you have to install a second copy of WebSite-Watcher as Portable Edition into a separate folder on your hard disk (eg. c:\wsw2). Then this second installation can be started and executed at the same time as the desktop edition. For each further instance you have to install a separate portable edition to your hard disk.

The option for the "Portable Edition" can be selected during installation in the setup wizard.

You don't need an additional license key when installing additional copies of WebSite-Watcher, just enter the same license key that you have used for your desktop edition.

1.13.7 Command line

Usage

```
wswatch.exe [Options] filename
```

Options

/R	Run a manual check of your bookmarks. A bookmark file must be defined in the command line in combination with the /R parameter.
/Q	Exit WebSite-Watcher when the check has finished. Works only in combination with /R.
/C	Connect (Dialup). Works only in combination with /R.
/D	Disconnect (Dialup). Works only in combination with /R.
/L	Write log file (wswatch.log). Works only in combination with /R.
/SND	Use sound. Works only in combination with /R.

/BatchMode	Suppress specific user input messages, for example to automatically check for new versions. Works only in combination with /R.
/MINIMIZE	Start WebSite-Watcher minimized
/SEARCH	Open the "Extended Search" window
/SCRIPT="..."	Execute the specified script (in quotes). If the script isn't stored in the default script directory, you have to use the full path of the script file.
/NODB	Starts WebSite-Watcher without opening the last opened database
/EXIT	Exits a running WebSite-Watcher. (this parameter will abort a running check)

Examples

```
wswatch.exe "c:\wsw\magazines.wsw"

wswatch.exe /c /d /r /q "c:\wsw\magazines.wsw"
```

1.13.8 Date variables

If an URL contains a daily date, you can use a **date variable** that will be replaced with the real date when the bookmark is checked.

Example on an URL with a date variable:
- http://www.domain.com/file<$date$,'yyyy-mm-dd',0>.htm

Resulting URL:
- http://www.domain.com/file2009-03-15.htm

Usage

```
<$date$,format,DayDifference>
<$date$,format,DayDifference,MonthNames>
```

If the date format contains the variable 'mmm' or 'mmmm' for month names, WebSite-Watcher will automatically use English names. If you need localized month names, you can define them with the parameter `MonthNames`. You have to define exactly 12 names separated with a colon, for example jan:feb:mae:apr:mai:jun:jul:aug:sep:okt:nov:dez.

Date format

yy	current year with 2 digits
yyyy	current year with 4 digits
m	current month with 1 or 2 digits, no leading zero (1-12)

mm	current month with 2 digits, leading zero (01-12)
mmm	name of current month, 3 characters (jan, feb...)
mmmm	full name of current month
d	current day with 1 or 2 digits, no leading zero (1-31)
dd	current day with 2 digits, leading zero (01-31)
hh	hours
nn	minutes
ss	seconds
"xx"	Characters enclosed in quotes are displayed as-is, and do not affect formatting.

Day difference

The parameter `DayDifference` lets you add/subtract the specified number of days to/from the current date.

- 0 ... current date
- 1 ... tomorrow
- -1 ... yesterday
- -2 ... the day before yesterday

If you append the character "h", "m" or "s" to the parameter, then the value will added to hours, minutes or seconds.

- **1h** ... plus one hour
- **-100m** ... minus 100 minutes
- **5s** ... plus 5 seconds

Examples:

```
http://www.domain.com/file<$date$,'yyyy-mm-dd',0>.htm
http://www.domain.com/changes<$date$,'yy-mm',0>.txt
http://www.domain.com/events-<$date$,'mmm-dd',1>.html
http://www.domain.com/news_<$date$,'mmmmyyyy',0,jan:feb:mae:apr:mai:jun:jul:aug:sep:o
```

1.13.9 Cookies

Dependent on the feature you use, WebSite-Watcher uses different databases to save Cookies.

There are a total of 4 different Cookie databases:

1. **View pages in browser tabs**
   WebSite-Watcher uses an embedded Chromium browser to display bookmarks and online pages. Cookies that are used when displaying pages in the internal browser will not be used when

checking bookmarks, the check process uses isolated Cookie databases. So changing something in a browser tab will not influence bookmark checks.

2. **Check technology "Default"**
If you check bookmarks with the technology Default, WebSite-Watcher uses a separate Cookie database that is only used by this check technology. To determine Cookies for a bookmark, right click the bookmark in the main window and select the menu item "Browser".

3. **Check technology "Browser: Chromium", Chromium-Macros and "Screenshot: Chromium"**
If you check bookmarks with one of these technologies, WebSite-Watcher will use a separate Cookie database that is only used by these check technologies. To determine Cookies for a bookmark, right click the bookmark in the main window and select the menu item "Browser".

4. **Check technology "Browser: Edge", Edge-Macros and "Screenshot: Edge"**
If you check bookmarks with one of these technologies, WebSite-Watcher will use a separate Cookie database that is only used by these check technologies. To determine Cookies for a bookmark, right click the bookmark in the main window and select the menu item "Browser".

5. **Check technology "Internet Explorer/Browser" and Internet Explorer Macros**
If you check bookmarks with one of these technologies, WebSite-Watcher will use the Internet Explorer cookie database. To determine Cookies for a bookmark, right click the bookmark in the main window and select the menu item "Browser".

These Cookie databases work independently of each other, Cookies are not shared by these technologies.

See also:
• Ignore Cookie Banner

1.13.10 Silent Install/Uninstall

The WebSite-Watcher setup program accepts optional command line parameters to perform silent installations.

Silent Install

Syntax:

    wswsetup.exe /SILENT /DIR="x:\dirname"

Parameters:

• /SILENT
Instructs the setup to be silent, no user interaction will be required. Only the installation progress window will be displayed, also error messages during the installation.

- /DIR="x:\dirname"
  Defines the default setup directory (which is usually displayed on the Select Destination Location wizard page). A fully qualified pathname must be specified.

- /x32
  By default, the setup will install the 64bit version of WebSite-Watcher on 64 bit systems, and the 32bit version on 32bit systems. If that command line parameter is set, the setup will install the 32bit version on 64bit systems.

Silent Uninstall

Syntax:

unins000.exe /SILENT /REMOVEDATA

Parameters:

- /SILENT
  When specified, the uninstaller will not ask the user for startup confirmation or display a message stating that uninstall is complete. Critical error messages will however be displayed.

- /REMOVEDATA
  When specified, the uninstaller will also remove the WebSite-Watcher configuration and the bookmark database (if stored in the default bookmark folder).

Advanced setup options

The topic Advanced setup options shows how to automatically install the license and pre-define certain program configurations.

1.13.11 Advanced setup options

The features described below are only available in the registered Business Edition and will not work in the Trial version nor in the Free/Personal Editions.

This topic shows how to automatically install the license key and how to pre-define certain program configurations. This can be helpful when installing and maintaining WebSite-Watcher on many PCs or workstations.

Automatic installation (for all users of a computer)

Before installing WebSite-Watcher, create an INI-File with the name **wswconfig.ini** and save it in the same folder as the file **wswsetup.exe**. When installing WebSite-Watcher, the file wswconfig.ini will be copied into the setup folder.

All options defined in the file wswconfig.ini file will be imported each time you start WebSite-Watcher, so it's not possible that a user can overwrite these options in the program configuration of WebSite-Watcher.

Manual installation (for a single user of a computer)

Alternatively you can manually copy the file wswconfig.ini into the users config folder. When WebSite-Watcher is started, this file will be imported once. This file will only be imported if the file wswconfig.ini is not available in the setup folder.
The exact path of this file is::
c:\Users\%USERNAME%\AppData\Roaming\aignes\WebSite-Watcher\config\wswconfig.ini

wswconfig.ini

The file wswconfig.ini can have the following sections and fields:

[License]
Enter the license key into the next line. Line breaks are allowed, an example can be found at the end of this topic. Please note that the license key of a Business Edition is required for all following options.

[General]

Language=	Language for the user interface. Available language files can be found in the sub folder "lang". If not defined, then English will be used. Example see at the end of this topic.
ShowWelcomeWindow=	0 ... Do not show the welcome window when starting WebSite-Watcher the first time
CheckForNewVersion=	0 ... Check periodically, ask the user 1 ... Check periodically in background, don't ask the user 2 ... Don't check for updates
BookmarkFile=	Absolute path and filename of the bookmark file Example: BookmarkFile=f:\website-watcher\bookmarks.wsw
StartAutoWatchAfterLoading=	1 ... Start AutoWatch after loading WebSite-Watcher 0 ... Don't start AutoWatch after loading WebSite-Watcher
DateFormat=	Date format for columns "last check", "last update", etc. See also Appearance
CurrentDateFormat=	Date format of the current day See also Appearance
ShowButtonInBrowserTitleBar=	1/0 ... Show button in browser title bar (Browser integration) See also General
ShowUpdateTicker=	1/0 ... Show Update-Ticker See also Update-Ticker

[Proxy]

http_proxy_config=	0 ... automatic proxy configuration 1 ... don't use proxy 2 ... manual proxy configuration
http_addr=	HTTP proxy address
http_port=	HTTP proxy port
http_username=	username, if required
http_password=	password, if required (plaintext)
ftp_noproxy=	1 ... don't use FTP proxy 0 ... use FTP proxy
ftp_direct_type=	use one of the following values: • not passive • passive
ftp_type=	use one of the following values: • Proxy not passive • Proxy passive • Socks 4 • Socks 4a • Socks 5
ftp_login_type=	Login type that can be determined automatically in the program configuration. Valid values are 0-8.
ftp_addr=	FTP proxy address
ftp_port=	FTP proxy port
ftp_username=	username, if required
ftp_password=	password, if required (plaintext)
ftp_ftpoverhttp=	1 ... use FTP-over-HTTP 0 ... don't use FTP-over-HTTP
noproxy=	Enter the domains here which should not be checked through a proxy. Separate domains with **{crlf}** <u>Example</u>: domain.com{crlf}domain2.com{crlf}.domain2.com

[Connection]

threads=	Number of simultaneous connections
timeout=	Timeout in seconds

[Dialup]

usedialup=	1 ... dialup connection to the internet 0 ... permanent connection to the internet
dial=	1/0 ... dial before checking
hangup=	1/0 ... hangup after checking

dun=	name of connection
username=	username of connection
password=	password of connection

[EMail]

smtp_server=	host address of the SMTP-Server
smtp_port=	port number
smtp_username=	Username for SMTP server
smtp_password=	Password for SMTP server
smtp_authentification=	1/0 ... Perform SMTP authentification
smtp_connection=	fullssl ... Full SSL autossl ... Auto SSL
smtp_after_pop=	1/0 ... use smtp-after-pop
pop3_server=	host address of the POP3-Server (only necessary when you use smtp-after-pop)
pop3_port=	port number
pop3_username=	username for POP3-Server
pop3_password=	password for POP3-Server
pop3_connection=	fullssl ... Full SSL autossl ... Auto SSL
to_email=	recipient e-mail addresses (separated by blank or ,)
to_email_bcc=	BCC e-mail addresses (separated by blank or ,)
from_email=	sender e-mail address
advanced_rules=	Use advanced rules for sending e-mails (instead of sending after each check run). Valid values are 1 or 0.
max_mails_per_session=	Max. e-mails per session (SMTP login).
sec_between_sessions=	Seconds between SMTP sessions.
blacklist=	Black listed phrases to suppress e-mail sending. Lines must be separated with {crlf} Example: phrase1{crlf}phrase2{crlf}phrase3

[Sounds]

Updates=	1/0 ... Play sound when a bookmark has been changed
CheckEnd=	1/0 ... Play sound when a check run has finished
DefaultSounds=	1/0 ... Play default sounds, for example when a message box is displayed or when initialized bookmarks are opened

Example

The following example will apply the following options:

1. Automatically use the entered license

2. Hide the welcome window

3. Don't check for new WebSite-Watcher versions

4. Use the program with German user interface

```
[general]
CheckForNewVersion=2
ShowWelcomeWindow=0
Language=german.lng

[license]
VkdobElHWnZiR3h2ZDJsdVp5QmxlR0Z0Y0d4bElIZHBiR3c2RFFwaGRYUnZiV0YwYVdO
aGJHeDVJSFZ6WlNCMGFHVWcNClpXNTBaWEpsWkNc2FXTmxibk5sRFFwb2FXUmxJSFJv
WlNCM1pXeGpiMjFsSUhkcGJtUnZkzBLWkc5dUozQWdZWFYwDQpiMjFoZEdsallXeHNl
U0JqYUdWamF5Qm1iM0lnYm1WM01GGZGxZbE5wZEdVdFYyRjBZMmhsY2lCMlpYSnphVzl1
Y3cwKVlhObElIUm9aU0J3Y205bmNtNtRnRJR2x1SUVkbGNtMWhiaUJzWWVc1bmRXRm5a
VGhlIGZvbGxvd2luZyBleGFtcGxlIHdpbGw6DQphdXRvbWF0aWNhbGx5IHVzZSB0aGUg
ZW50ZXJlZCBsaWNlbnNlDQpoaWRlIHRoZSB3ZWxjb21lIHdpbmRvdw0KZG9uJ3QgYXV0
b21hdGljYWxseSBjaGVjayBmb3IgbmV3IFdlYlNpdGVtV2F0Y2hlciB2ZXJzaW9ucw0K
VXNlIHRoZSBwcm9ncmFtIGluIEdlcm1hbiBsYW5ndWFnZW==
```

1.14   App (Andorid / iOS)

1.14.1   WebSite-Watcher App

The WebSite-Watcher App can be used to read changed bookmarks on your Smartphone.

WebSite-Watcher for Windows must be running and will use the configured WebDAV server (or alternatively your personal Dropbox account) to send changed bookmarks or status messages (eg. e-mails could not be sent) to the App.

Requirements

- WebSite-Watcher for Windows
- WebSite-Watcher App (Free or Pro Edition)
- WebDAV server or Dropbox account

Limitations:
- Sending data to the App only works with the Desktop Edition. The Portable Edition is not able send data to the App.

Setup required on your PC

1. In WebSite-Watcher, open the program configuration

2. Click the button "WSW App"

3. Enter the WebDAV configuration or connect WebSite-Watcher with your Dropbox account

Setup required on your Phone

1. Open the module "WSW for Windows"
2. Click the button "Connect to Cloud"
3. Select if you want to use a WebDAV server or your personal Dropbox account
4. Enter the WebDAV configuration or connect WebSite-Watcher with your Dropbox account

Upload bookmarks and cache files

WebSite-Watcher will periodically send a list with updated bookmarks to the cloud so you can see on your phone which bookmarks have been changed. To be able to view highlighted changes on your phone, you also have to upload the bookmark cache files by turning on the Phone Mode (optional, disabled by default). The topic Upload Cache Files contains more information about which data is sent to the cloud and how cache files can be uploaded.

See also topic WSW App

1.14.2 Upload Cache Files

Changed bookmarks can be sent to the WebSite-Watcher App (Android / iOS) via any WebDAV supported cloud and file hosting service (or alternatively your personal Dropbox account). Sending data to the App only works when the Windows version of WebSite-Watcher is running. It only makes sense if used in combination with AutoWatch or a script that periodically monitors bookmarks.

Below you can find some information about which data is sent to your WebDAV server or to your Dropbox.

Data uploaded to the Cloud

The following data is uploaded to the cloud:

1. The bookmark and status feed (always)
   The bookmark feed mainly contains the name, URL and change-date of the last 30 updated bookmarks. No page content with highlighted changes is uploaded with that feed. The status feed contains data about the last 5 errors, for example if an e-mail could not be sent. This feed is uploaded every 10 minutes, the size of that feed is around 20 KB.

2. The bookmark cache files (optional)
   The bookmark cache files contain the checked pages with highlighted changes that you typically see when you open a red/changed bookmark. The size depends on the page and is approx. 50-150KB per bookmark. Similar to the bookmark feed, WebSite-Watcher keeps the last 30 cache files in the cloud. Cache files are only uploaded to the cloud when the Phone-Mode is turned on.

By default, uploading cache files is disabled and WebSite-Watcher will only upload the bookmark and

status feed. With this minimal information, you will see which bookmarks have been updated, you are not able to open a changed bookmark with highlighted changes in the App. For this, the optional cache files are required. Long pressing a bookmark will always open the online version of a page.

Uploading cache files / Phone-Mode

To see the changed bookmarks with highlighted changes in the WebSite-Watcher App, you also have to upload the cache files to the Cloud by turning on the Phone-Mode.

The Phone-Mode is a special mode that will, as long as activated, upload cache files with highlighted changes to the cloud. The Phone-Mode can be enabled in the Tools menu of WebSite-Watcher, or via the WebSite-Watcher App.

1.14.3 Add to phone bookmarks

This feature can be used to add URLs from WebSite-Watcher for Windows to the bookmark list on your smartphone.

This feature will add URLs to a temporary list (this list keeps the last 20 URLs). In the WebSite-Watcher App you can then add new URLs from that list.

WebSite-Watcher for Windows

Add bookmarks from the WebSite-Watcher main window to the temporary list:

1. Right click a bookmark
2. Select "Extras"
3. Click "Add to Phone Bookmarks"

Furthermore you can use the Website-Watcher button in your web browser to add URLs from the browser to the temporary list.

WebSite-Watcher App

1. Add new bookmark
2. Click "Windows"
3. Select URLs from the temporary list
4. Click the "Add" button

1.15 How to...

1.15.1 Display images instead of grey boxes

By default, WebSite-Watcher replaces all images in a page with local, grey boxes. This is the fastest way to check and open bookmarks.

Alternatively to the ugly grey boxes you can also download and display the real images of a page. The image configuration can be changed globally in the program configuration, or individually for each bookmark.

Options to download images

WebSite-Watcher supports the following options to download and display images:

1. **Display grey boxes instead of images (fastest)**
   If this option is enabled, all images will be replaced with a local, grey image. This is the fastest way to check and open bookmarks, no images are downloaded and displayed.

2. **Don't display images**
   Replaces all images with a transparent image, so no image is displayed in the page. Use that option if images overlap other areas of the page. It is as fast as option #1.

3. **Use absolute addresses for all images**
   If this option is enabled, all image addresses will be converted to absolute addresses. No images will be downloaded during a check. As soon as you open a bookmark, the images will be downloaded from the server and displayed in your browser.

4. **Save images to disk**
   If this option is enabled, WebSite-Watcher downloads images as soon as an update is detected. All images are saved to your hard disk.

   <u>Re-download already saved images</u>
   If this option is enabled, WebSite-Watcher will always download all images instead of only new images. Enable this option if a web page always uses the same filename for changed images.

<u>Show embedded videos</u>
WebSite-Watcher can optionally show certain types of videos if the option for absolute image addresses or downloading images is enabled. This only works if the used technology to display embedded videos is supported by WebSite-Watcher.

<u>Don't show images within filtered content</u>
If enabled, WebSite-Watcher will not download and display images within filtered content. This option only works in combination with Ignore filters, not with Watch filters.

Global image configuration

1. Click Options in the main menu
2. Open the "Program configuration"
3. Select the "Bookmarks" tab

4. Click the "Download/Display Images..." button

5. Enter the image configuration

Individual image configuration per bookmark

1. Open bookmark properties

2. Select the "Advanced" tab

3. Select "Images" on the left side

4. Enter the image configuration

1.15.2 Monitor pages for specified words (keywords)

To only get an update notification when specified keywords are available within new or changed page content, you have to use the "Update on keywords" feature in the bookmark properties. To highlight specified keywords, you have to use the feature "Highlight keywords". Both features work independently. If you want to highlight the same keywords you're looking for, then you have to enter the keywords twice or use the option "Use same keywords as above".

1. Open bookmark properties

2. Select the "Keywords" tab

3. Enter the keywords in "Highlight keywords" and/or "Update on keywords". Enter one word/ phrase/regex per line.

Keywords are only highlighted in the version with highlighted changes, not within the new/old version of a page.

If you use the "Update on keywords" feature, then the new version will only be saved when at least one of the entered keywords is found within changed content.

Regular expressions must be defined with regex() or regexcmp(), for example: regex(WebSite. Watcher)

Different keywords can also be highlighted with different colors, just add #COLORNAME on the right side of the keyword. Supported color names are: red, green, olive, lime, teal, blue, aqua, magenta, orange
Example:
website #blue
watcher #red

See also topic Keywords

1.15.3 Check pages that require Javascript

By default, WebSite-Watcher filters all Javascript content from a page and only works with the static, textual content.

For pages that can only be displayed with Javascript enabled, WebSite-Watcher supports the check technology "Browser".

> 1. Open bookmark properties
> 2. Select the tab "Check"
> 3. Select the technology "Browser" (and change the browser engine if necessary)

This technology opens the page in a hidden browser window and converts the "dynamic Javascript content" into a static, textual content that can be used and processed by WebSite-Watcher. WebSite-Watcher supports the browser engines Chromium, Edge and Internet Explorer.

Delay after page has loaded (in seconds):
This option must be used for pages that dynamically change the content after the page has been loaded. For example when it takes a few seconds until the final content is displayed.

1.15.4 Monitor a whole site (Follow Links)

By default, you have to create a separate bookmark for each page you want to monitor.

If a whole site instead of single pages should be monitored, you can use the **Follow Links** functionality. This feature will automatically insert all links of a page as new bookmarks. With several options you can specify how many or which links should be inserted as new bookmarks, for example the link depth.

The Follow Links requires a Master-Bookmark that is the root for all other automatically inserted child bookmarks. Master bookmarks displays the image 🐛 before the bookmark name and are usually sorted before all other bookmarks.

For all child bookmarks the Follow Links configuration is disabled. Changes to the Follow Links configuration can only be made in master bookmarks.

> 1. Create a new folder (it's recommended to put each Master-Bookmark into a separate folder)
> 2. Create a new bookmark and enter the URL
> 3. Select the "Advanced" tab
> 4. Select "Follow Links" on the left side
> 5. Enable the "Follow-Links" option and enter the configuration

1.15.5 Send e-mails when changes are detected

If changes in a page are detected, WebSite-Watcher can send an e-mail to one or more recipients. The setup of that feature consists of two steps:

1. SMTP Server Setup
2. Enable the Send-Mail feature for specified bookmarks

SMTP Server Setup

Before you can send e-mails, you have to enter the mail server configuration.

1. Open the program configuration
2. Select the "Check" tab
3. Click the "E-mail" button
4. Enter the server configuration

- **Smtp Server**
  SMTP server for outgoing e-mails.

- **Smtp after Pop**
  Depended from the server configuration, a POP3 login may be required before you can send e-mails (also called SMTP-after-POP or POP-before-SMTP authentication). If this option is enabled, WebSite-Watcher will only perform a login and a logout to the POP3 server, no mails will be downloaded.

- **Pop Server**
  Is only required if you use the Smtp-after-Pop feature.

- **Connection**
  WebSite-Watcher supports different methods for Server Authentication:

➢ **Regular -** Standard login without SSL/TLS
➢ **Full SSL -** Uses SSL/TLS starting with the first contact to the server. Servers with full SSL/TLS mode usually don't use the standard TCP port (25).
➢ **Auto SSL -** Uses SSL/TLS only if it is supported by the remote server.

Template:

The Template section contains basic e-mail configurations such as name and mail address of the sender or recipient. To enter more than one e-mail address to the Recipient field, you have to separate them with a comma or a blank.

The Subject and Body fields can contain variables which will be replaced with real values when e-mails are sent.

Variables in the e-mail subject:

{name}	Name of the bookmark
{url}	Url of the bookmark
{check_url}	Url that was finally checked, recognizes changes to the URL via the Plugin system (event function Wsw_BeforeCheck) and redirections.
{folder}	Absolute folder path
{foldername}	Folder name
{status}	Status of the last check
{infofield1}	Info field #1
{infofield2}	Info field #2
{infofield3}	Info field #3
{changes50}	First 50 characters of highlighted changes
{changes50a}	Same as {changes50} but removes the text "NEW ITEMS(n):" from RSS feeds
{changes50b}	Same as {changes50} but removes the text "NEW ITEMS" from RSS feeds while keeping the number of new items
{date_changed}	Date of last update
{counter}	Incrementing counter. This counter increments by 1 with every e-mail sent. In your mailbox you could then see at a glance if a message did not come through because of a missing number.

Variables in the message body:

{name}	Name of the bookmark
{url}	Url of the bookmark
{check_url}	Url that was finally checked, recognizes changes to the URL via the Plugin system (event function Wsw_BeforeCheck) and redirections.
{folder}	Absolute folder path
{foldername}	Folder name

{status}	Status of the last check
{note}	Bookmark note
{infofield1}	Info field #1
{infofield2}	Info field #2
{infofield3}	Info field #3
{date_changed}	Date of last update

Max E-Mail Size:
If you send e-mails with embedded images, you can limit the max size of the e-mails. WebSite-Watcher will remove images from the e-mail as long as the size larger than the configured max value.

Enable the "Send e-mail" feature in the bookmark properties

If an e-mail should be sent for a bookmark, you have to enable the "Send Mail" feature in the bookmark properties.

1. Open bookmark properties
2. Select the "Actions" tab
3. Select "Send e-mail" on the left side
4. Enable the Send e-mail feature

If the Recipient field is entered, then it will be used instead of the recipient that was entered in the program configuration. If you enter more than one e-mail address, you have to separate them with a comma or a blank. The subject field supports the same variables as available in the program configuration.

- **Send full page**
  Sends the full page with all CSS definitions and downloaded images. Main goal is that the e-mail will look like similar to the page in your Internet Browser. Please note that many e-mail clients or Webmail services do not support all formatting commands, so the layout may be broken in your e-mail client. In this happens, you can choose one of the other sending options.

- **Send simple version**
  This sending option strips the layout of the page and will only send a simplified version. This is similar to the TEXT tab of the internal browser. Use this option if your e-mail client has problems displaying the full layout or formatting of the page.

- **Send only changes**
  This feature extracts and sends only the highlighted text of the page. This option provides the best compatibility with various e-mail clients, it also reduces the size of the e-mails.

- **Attach screenshot**
  This option attaches a screenshot with the highlighted changes. This option can be used in

combination with the feature "Send only highlighted text" and might be useful when the layout of the page is required to understand the highlighted changes. Please note that the size of the screenshot is limited and you might not see all changes.

- **E-Mail Template**
  Lets you assign a customized e-mail template (optional). If assigned, then the template configuration will override the global e-mail configuration and can be used to send e-mails with an alternative "From" e-mail or an alternative body.

- **Send before other e-mails**
  The option "Send before other e-mails" will insert an e-mail to the top of the outbox. So these e-mails have higher priority in the outbox and will be sent before e-mails without that flag. Use this option only for single (critical) bookmarks, not for most or all bookmarks.

1.15.6 Ignore updates

The "Ignore updates" feature allows you to

1. Prevent an update notification when a page contains a certain pre-defined text (**blacklist mode**), or
2. Only allow an update notification when a page contains a certain pre-defined text (**whitelist mode**)

Typical example: Page shows certain error message
If a page regularly displays a certain error message, you can prevent false update notifications when such a message appears. Typical examples for such error messages are "Too many users online, please try again later" or "Could not connect to database".

> 1. Open bookmark properties
> 2. Select the "Advanced" tab
> 3. Select "Ignore Updates" on the left side
> 4. Enter the text for which WebSite-Watcher should prevent an update notification (one phrase per line)

If a black listed word is found in the page, WebSite-Watcher will ignore the update but not list the bookmark in the error folder nor alert an update after X errors in a row.

If WebSite-Watcher should list the bookmark in the error folder and alert an update after the configured number of errors in a row, you have to use the program program Tweak "HandleIgnoreUpdateAsError=1".

1.15.7   Limit connections to a single server/URL

If a server doesn't allow simultaneous connections, you can reduce the number of connections to that server or to a specified URL.

---

1. Open the program configuration

2. Select the "Check" tab

3. Click the "Connection..." button

4. Select the tab "Advanced Connection Rules"

5. Click the "New" button

6. Enter the domain name, for example "aignes.com" (without quotes) and the connection limitations you want to use

---

1.15.8   Make a bookmark a Hotsite

What are HotSites?

HotSites are bookmarks with a certain flag enabled. All HotSites are listed in the virtual folder "HotSites" independently from the folder they are really saved.

This gives you the advantage to display your favorite bookmarks within a single folder.

How to enable the HotSite flag

There are several ways to make a bookmark a HotSite:

- Enable the check box "HotSite" in the bookmark properties
- Select a bookmark and double click the HotSite column
- Drag&Drop a bookmark into the "HotSites" folder

1.15.9   Change properties in multiple bookmarks

WebSite-Watcher offers different ways to change specified properties in multiple bookmarks with a single procedure.

1. Change properties in selected bookmarks

---

1. In the main window, select the bookmarks you want to change

2. Right click the selection

3. Select "Set properties in all selected bookmarks..."

This will open a window that lets you configure specified bookmark properties. To change a specific

configuration, enable the checkbox "Assign" and configure the related feature.

Disable an already configured option

To disable an already configured feature, you have to check the "Assign" box but uncheck the related feature (as shown in the following example):
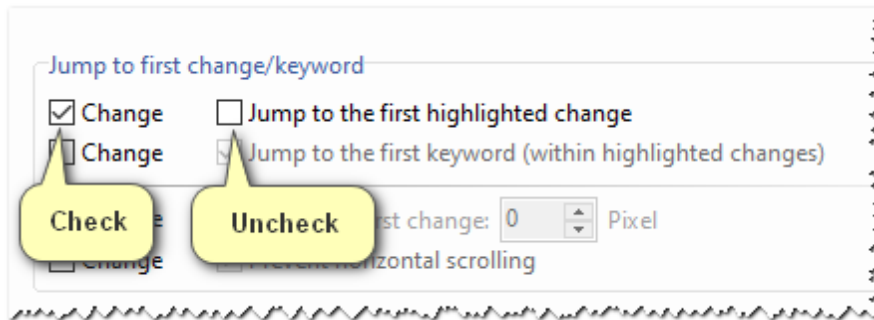
2. Update similar bookmarks

If you have changed a configuration in the bookmark properties directly and want to apply these changes to other bookmarks, you can check the option "Update similar bookmarks" before you press the "OK" button.
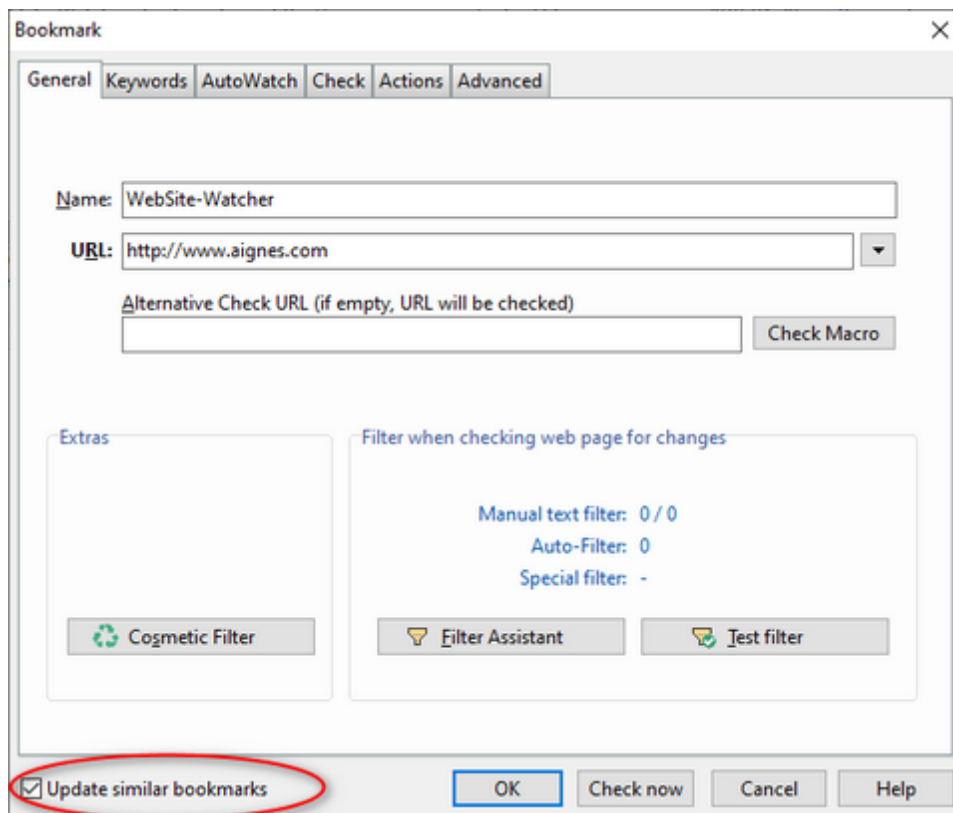
This will display a window that lets you select the target bookmarks and the options you want to copy from the actively opened bookmark.

1.15.10 Monitor the full HTML code of a page

By default, WebSite-Watcher filters all HTML tags and uses only the real, readable text content of a page. To monitor a page with all HTML tags, you have to disable the "Ignore all HTML tags" filter in the bookmark properties.

1. Open bookmark properties
2. Click the "Filter-Assistant" button
3. Select the "Special filter" tab
4. Uncheck the filter "Ignore all HTML tags"

If you want to see the source code of the page (instead of the rendered page) when you open the bookmark, then change the option "file type" from "auto detect" to "text".

5. Select the "Advanced" tab
6. Select "Check options (1)" on the left side
7. Change the option "file type" from "auto detect" to "text"

1.15.11 Ignore Cookie Banner

Preferred solution

If a checked bookmark shows a Cookie consent banner that you have to accept, then in most cases you can use these steps:

1. Right click the bookmark
2. Select "Browser". This will load the page in a browser window.
3. Accept the Cookies
4. Close this window

Then the next detected update should no longer display the cookie message. To test these steps immediately, right click the bookmark and select "Extras + Re-Initialize bookmark".

This is the preferred method for dealing with Cookie consent banners and should be used whenever possible.

Alternative: Cosmetic Filters

If the steps above do not work correctly or the Cookie banner reappears after a short while, you can use Cosmetic Filters to remove it from the page.

1. Open the bookmark properties
2. Click "Cosmetic filter"
3. Click the Wizard symbol
4. Select "Cookie Banner". This tries to automatically determine the correct cosmetic filter.

Create Cosmetic Filter manually

If no cosmetic filters are found with the steps above, you have to create the cosmetic filter manually.

Typical steps:

1. Click "Create new filter"
2. Enter a short phrase or word that is part of the Cookie consent text, for example: cookie
3. Search the correct occurrence of the entered word or phrase
4. Click Next, this will find related CSS definitions
5. Select a CSS definition that includes all the text of the Cookie banner
6. Click Next
7. Test the filter. If the Cookie banner is removed, click "Add filter"

1.15.12 Pre-define bookmark properties

Ways to pre-define bookmark properties

- **Globally in program configuration**
  Configurations for new bookmarks can be pre-defined globally in the program configuration. These pre-configurations are valid for all databases and applied when you create new bookmarks.

  1. Click "Options" in the main menu
  2. Select "Program configuration"
  3. Select the tab "Bookmarks"
  4. Click the button "Assign bookmark properties"
  5. Enter the properties you want to pre-define for new bookmarks

- **Per folder**
  It's also possible to pre-define bookmark properties for new/existing bookmarks per folder. These

pre-defined bookmark properties are stored per bookmark database.

> 1. Right click a folder
>
> 2. Select "Properties"
>
> 3. Select the tab "Assign bookmark properties"
>
> 4. Enter when and which bookmark properties should be assigned

- **Rule based for certain URLs**
  A third way is a rule based system that lets you pre-define properties for bookmarks with certain URLs, for example: use the checking method "Browser: Chromium" for all URLs with domain "aignes.com". These pre-defined bookmark properties are stored per bookmark database and assigned to new/existing bookmarks.
  Define these rules as unique as possible. If a pre-definition is found and the option "Continue processing with other pre-definitions" is disabled, then further rules won't be assigned to a bookmark. Otherwise it's possible to assign multiple pre-definitions to a single bookmark.

> 1. Click "Options" in the main menu
>
> 2. Select "Global bookmark pre-predefinitions"
>
> 3. Enter the rules and the properties for new/existing bookmarks

Order how pre-defined bookmark properties are assigned

1. Pre-definitions from program configuration (new bookmarks only)

2. Pre-definitions from folder properties

3. Rule based pre-definitions from "Global bookmark pre-predefinitions" in the Options menu. Define these rules as unique as possible. If a pre-definition is found and the option "Continue processing with other pre-definitions" is disabled, then further rules won't be assigned to a bookmark. Otherwise it's possible to assign multiple pre-definitions to a single bookmark.

If pre-definitions are available in #1 #2 and #3, then they will be assigned from all of them whereas properties from step #2 will override properties from #1, and properties from #3 will override properties from #2 and #1.

1.15.13 Using Local Website Archive

WebSite-Watcher stores the last two version of a web page. If you want to archive a page permanently for future reference, you can use our tool Local Website Archive (LWA). Local Website Archive can be used as WebSite-Watcher add-on and/or as stand alone solution to archive web pages from your browser.

There are different ways how WebSite-Watcher and Local Website Archive can work together.

1. Archive pages manually

Similar to archiving pages from your Internet Browser (eg. from Chrome, Edge or Firefox), you can archive web pages from the internal WebSite-Watcher browser to Local Website Archive.
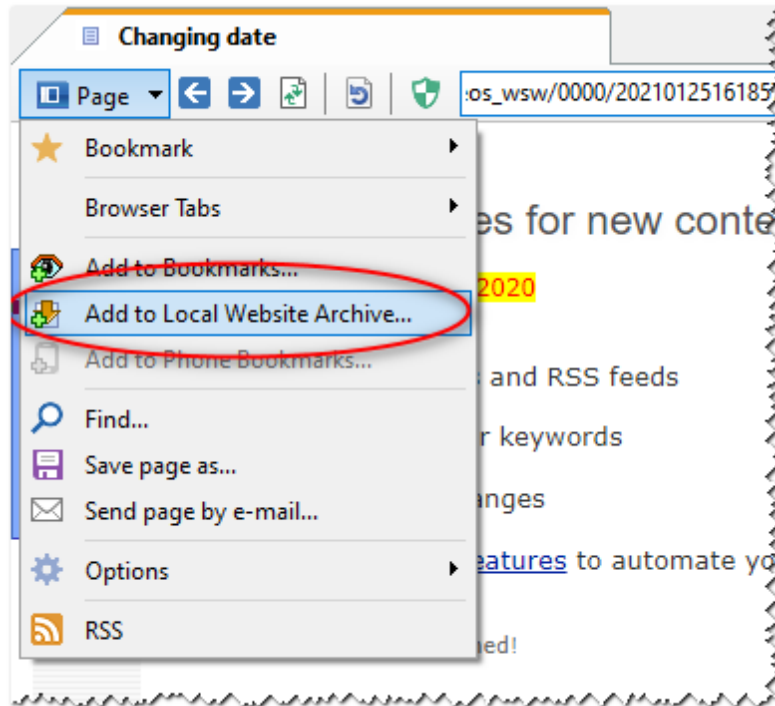
2. Archive pages automatically

Updated pages can automatically be archived to Local Website Archive via the bookmark actions. Once an update has been detected, you can either archive the page with highlighted changes or the new version of the page.
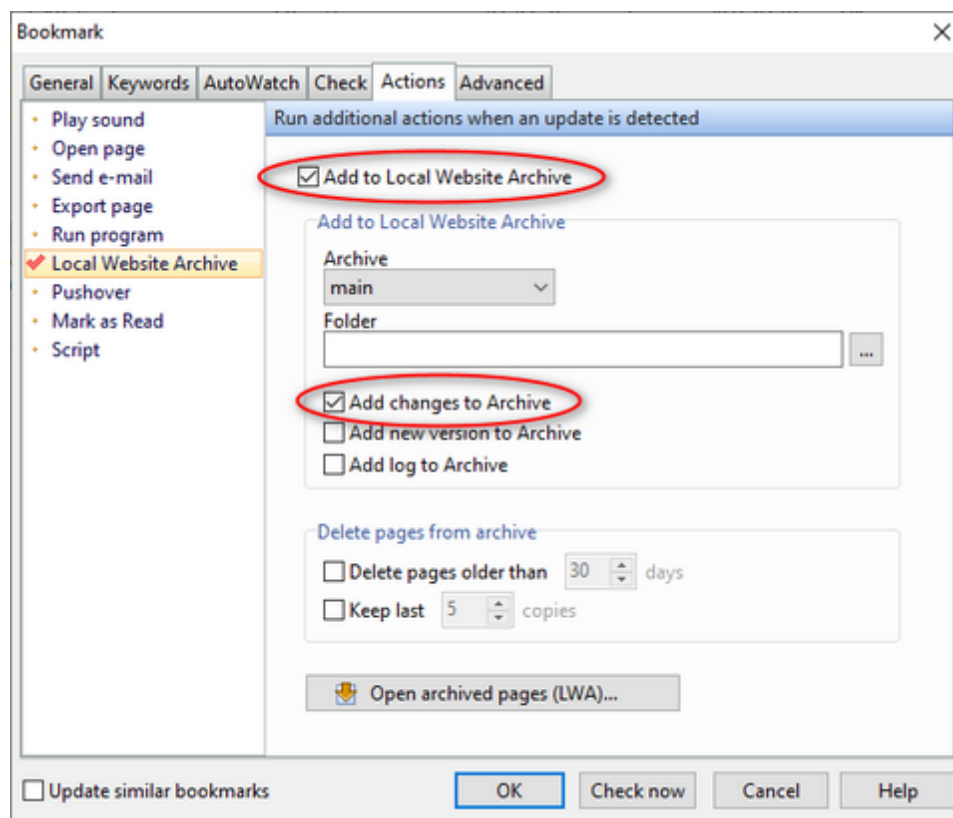
3. Open archived pages

If you work with the automatic method from #2, you can open the archived pages from various places:

- Bookmark properties
- Bookmark context menu
- Clicking the "Version" button in the browser toolbar
- Via a shortcut that can be configured in Options + Shortcuts
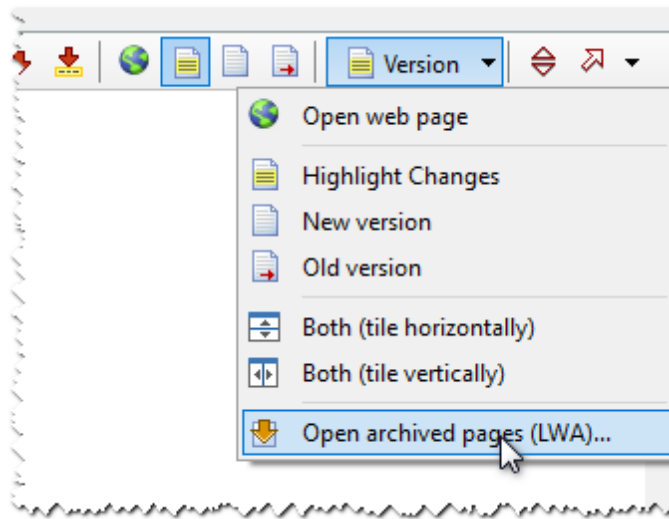
This will open an extra window with all archived pages of the selected/opened bookmark.

1.16 What to do when...

1.16.1 A page cannot be displayed

If a page cannot be displayed correctly, then you can try the check technology Browser. With this technology, WebSite-Watcher creates a hidden browser window and checks the page in that window.

> 1. Open bookmark properties
> 2. Select the tab "Check"
> 3. Select the technology "Browser" (and change the browser engine if necessary)

If this method does not work, you can try to check the page with the Screenshot technology.

> 1. Open bookmark properties
> 2. Select the tab "Check"
> 3. Select the technology "Screenshot" (and change the browser engine if necessary)

1.16.2 All bookmarks report an error

If a check run reports an error for all bookmarks, then one of the following reasons may cause the problem:

- **Proxy settings are not correct**
  If you need a proxy to connect to the internet, then verify that the proxy configuration is correctly entered in WebSite-Watcher.

  > 1. Open the program configuration
  > 2. Select the "Check" tab
  > 3. Click the "Proxy" button

> 4. Enter your proxy configuration

- **WebSite-Watcher is blocked by a Firewall**

  If you have a firewall or another Internet Security Suite running, then WebSite-Watcher is probably blocked by that solution. In that case you have to setup your firewall to let WebSite-Watcher access the internet, the problem cannot be fixed within WebSite-Watcher.

1.16.3 A check has status "Whole content filtered (verify your filter settings)"

If a check has the status message "Warning: whole content filtered (please verify your filter settings)", then there might be something wrong with your filter definitions and WebSite-Watcher isn't able to detect updates.

Normally WebSite-Watcher compares the filtered text content of the new version with the filtered text content of the old version of a page. If the whole content is filtered, then WebSite-Watcher has nothing to compare and is not able to detect updates. In such a case this error message is displayed.

Open the bookmark properties and verify/change your filter definitions to fix that behavior. With the function "Test filter" you can see the filtered content that is used to detect updates.

Related topic: What are filter settings (Overview)

1.16.4 A web page contains only cryptic characters

If a web page contains only cryptic characters, then a web page has probably no character set definition and WebSite-Watcher converts language specific characters wrongly to Unicode.

To fix that behavior, you have to define the character set manually.

> 1. Open bookmark properties
> 2. Select the "Advanced" tab
> 3. Select "Check options (1)" on the left side
> 4. Click onto "character set"
> 5. Select the correct character set from the sub menu
> 6. Re-check the bookmark

Part II
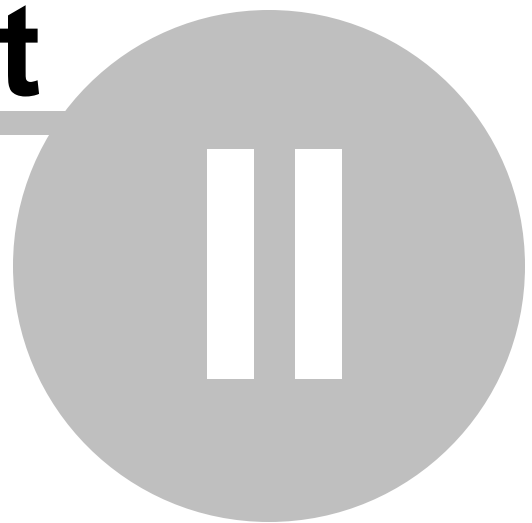
2   User Interface

2.1   Program configuration

2.1.1   General

Tray Icon

The tray icon section lets you define if and when WebSite-Watcher should be displayed in the tray bar.

- **Always show icon in system tray**
  If enabled, then WebSite-Watcher is always displayed in the system tray, independently if the application is minimized or not.

- **Move to system tray when minimizing WebSite-Watcher**
  WebSite-Watcher is moved to the system tray when the minimize button in the window title bar is pressed.

- **Move to system tray when closing WebSite-Watcher**
  WebSite-Watcher will be moved to the system tray when the close button in the window title bar is pressed. In that case you can close WebSite-Watcher only with the menu commands File+Exit or TrayIcon+Exit.

Independently from the Tray Icon configuration, WebSite-Watcher can always be moved to the system tray with the shortcut F8 or the toolbar button .

Browser Integration

- **WebSite-Watcher button in the title bar of your browser**

  If WebSite-Watcher is running, then a WebSite-Watcher button is displayed in the browser title bar. Clicking this button lets you add the currently opened page to WebSite-Watcher. It works with all main browsers such as Chrome, Edge and Firefox. This button can be moved horizontally via a Tweak.

See also topic Add bookmarks directly from your Browser

2.1.2 Bookmarks

Bookmarks

- **Automatically sort changed bookmarks before unchanged bookmarks**
  If this option is enabled, WebSite-Watcher will sort all changed bookmarks to the top of the list (independently from the sort direction).

- **Hide bookmarks in "Changed" folder after opening it**
  A bookmark will be removed automatically from the folder "Changed Bookmarks" as soon as you open it.

- **Open the next unread bookmark jumps also into the next folder**
  When opening the next unread bookmark, WebSite-Watcher will automatically jump to the next folder if the selected folder no longer contains changed bookmarks.

Download/Display images

WebSite-Watcher supports different methods to download or display images. Images can be replaced with local grey boxes (fastest method), images can be downloaded from the internet as soon as the page is opened (with absolute image addresses), or images can be downloaded immediately and stored on your local hard disk.

See also:
- Images
- Display images instead of grey boxes
- Highlight images

Predefined settings for new bookmarks

Here you can pre-defined properties which are assigned to new bookmarks. These pre-defined properties can be overridden for each folder (via the folder properties).

Plugins for: PDF/Word/Excel

By default, WebSite-Watcher converts PDF, Word and Excel files into web pages via the integrated Plugin system. In that section you can define the default Plugins which should be used for the conversions.

WebSite-Watcher contains internal conversion routines for these files, no additional tools are required. Alternatively to these internal solutions, WebSite-Watcher can use third party tools to convert these file types. For example, if Microsoft Office 2003 or higher is installed on your system, then you can also use Microsoft Office to convert Word/Excel files (better quality incl. layout, images,

etc).

2.1.3 Check

Bookmark Check

- **Alert with the first initialization of a page**
  With the first check a page is initialized, with each further check updates can be detected. If you enable this option you will also get an update notification when a page is initialized with the first check. No bookmark actions (eg. "Send e-mail") are performed in that case.

- **Alert after XX errors in a row**
  If a bookmark reaches the specified number of errors in a row, you will get an update notification. As soon as you open the page, the number of errors is reset and the error counter restarts from zero. No bookmark actions (eg. "Send e-mail") are performed in that case.

- **Re-Check bookmark when an error occurred**
  If a bookmark check failed with an error, then the bookmark will be re-inserted into the check queue for a second check.

AutoWatch

WebSite-Watcher supports an automated checking method (called AutoWatch) to monitor bookmarks automatically with predefined time intervals. You can turn on this mode with the AutoWatch button

in the toolbar.

- **Start AutoWatch when starting WebSite-Watcher**
  If this checkbox is enabled, AutoWatch will be turned on automatically when you start WebSite-Watcher.

- **Delay before first check**
  Enter the countdown in seconds that should be performed before running the first check. After the first check, WebSite-Watcher calculates the countdown automatically based on bookmark time configurations.

Automatic AutoWatch Interval

The "Automatic AutoWatch Interval" allows WebSite-Watcher to automatically use optimized time configurations based on previous update frequency.

Active pages will be checked more often, pages that lay dormant for months will be checked less often. This option is especially for bookmarks where it isn't really important to get an update notification as soon as possible.

If a bookmark check doesn't find changes, the time interval can grow up to the specified maximum. If changes are found, the time interval resets to the specified minimum and a new "growing process" can begin.

Automatic time configuration and Keywords:
If you have keywords specified in the bookmark properties (Update On Keywords), the time interval will also reset to the specified minimum when the page has been changed but no keywords were found.

See also:
• Check pages automatically (with AutoWatch)

2.1.4   Browser

- **Use internal browser**
  By default, WebSite-Watcher always opens bookmarks in the internal browser (Chromium browser). If you disable this option, then WebSite-Watcher will open bookmarks in the external default browser.

- **Close browser tabs when closing the internal browser**
  If this option is enabled, WebSite-Watcher will close all opened browser tabs when you close the internal browser. That means that you will see an empty document when you reopen the internal browser.

Tabbed Browsing

- **Use Tabbed Browsing**
  Tabbed browsing allows you to display pages in multiple tabs. If you click a link while pressing the Shift Key, the link is opened in a new browser tab. If this option is disabled, links are opened in your default web browser instead of a new browser tab.

- **Tab #1: Open links in new tabs**
  If enabled, then clicking on a link in the first browser tab will automatically open the link in the new tab. Otherwise online pages can also be opened in the first browser tab.

- **Restore last visited browser tabs**
  If the internal browser is closed with opened browser tabs, WebSite-Watcher can reopen these tabs automatically.

Ad-Blocker Whitelist

WebSite-Watcher includes a DNS based ad blocker that is turned on by default. If a server detects an ad blocker and does not display a web page, you can whitelist and turn off the ad blocker for that

domain.
Enter one domain (without protocol, www and path) per line.

Example:
aignes.com
website-watcher.com

2.1.5   AutoBackup

If the option "AutoBackup" is enabled, then WebSite-Watcher will backup your bookmark database to the configured AutoBackup folder. Only the bookmark database will be saved, no cache files or program configurations will be included.

You can restore these AutoBackups with "Tools + Backup/Restore", then select "Restore AutoBackup" and follow the assistant. This will copy back your bookmark database and re-create all important cache files.

2.1.6   Appearance

General

- **Show grid lines**
  Show grid lines in the bookmark list.

- **Start minimized**
  Starts WebSite-Watcher minimized. Alternatively to that option you can use the command line parameter /MINIMIZE to start WebSite-Watcher minimized.

Date format

Defines how a date is displayed in the columns "last checked" and "last changed". For the current day you can use a separate date/time definition, for example to display only the time without the date.

Date format:

yyyy	year with 4 digits
mmm	name of the month, format: Jan, Feb, Mar, ...
mm	month with 2 digits
dd	day with 2 digits
hh	hours
nn	minutes
ss	seconds
am/pm	adds am or pm to the current time

Examples:

```
dd-mmm-yyyy hh:nn
hh:nn am/pm
```

Fonts

Gives you the possibility to change the font of the folder tree and the bookmark list.

Changed Bookmarks

Gives you the possibility to change the color and the bold flag for changed bookmarks.

If you disable the bold flag, then selected changed bookmarks cannot be identified since the text color of selected bookmarks is always the same. Disabling the bold flag is only recommended when you view changed bookmarks via the "Changed Bookmarks" folder.

2.1.7 Messages

In that section you can specify when WebSite-Watcher should display messages or warnings.

2.1.8 Folder

Show folders

Specify which special folders should be displayed.

Trash

- **Empty Trash after NN days**
  Specify how long deleted bookmarks should remain in the trash. After that period, deleted bookmarks and cache files are finally deleted.

2.1.9 Update-Ticker

WebSite-Watcher can show a small floating Information Window (InfoWin) when:

1. Changed bookmarks are detected
2. WebSite-Watcher is not the foreground window.

A double click on that window brings Website-Watcher into foreground.

2.1.10 Sound

All sounds must be available in wav format.

- **Updates**
  is played when a page has been changed

- **Check-End**
  is played when a check run is finished

- **Allow default sounds**
  plays default sounds, for example when a message box is displayed or when initialized bookmarks are opened.

2.1.11 External Browser

Specify the browser that should be used as external browser.

2.1.12 RSS Feed Enclosures

WebSite-Watcher comes with an internal Download Manager that can be used to automatically download RSS feed enclosures (podcasts).

By default, WebSite-Watcher uses this download manager for MP3 or WAV enclosures. To add further file extensions, simply separate them with a coma.

Example:
mp3,wav,exe,msi,zip,rar,7z

2.1.13 Colors

Last change marker

WebSite-Watcher can display a marker after the last change in a page. Here you can define the text and the color of that marker. For each bookmark, the "Show last change marker" option can be enabled in the bookmark properties.

Removed elements

Define the color of removed elements. For each bookmark, the "Show removed elements" option can be enabled in the bookmark properties.

2.1.14 Local Website Archive

WebSite-Watcher stores the last two version of a web page. If you want to archive checked pages permanently for future reference, you can do this with the additional tool "Local Website Archive".

By default, the tool "Local Website Archive" is installed automatically with WebSite-Watcher. If you want to use a separate installation of Local Website Archive (instead of the version that comes with WebSite-Watcher), you have to enter full path+filename of the file "wsarc.exe" (EG: c:\mytools\lwa\wsarc.exe). The option "Use separate installation" is mainly available for legacy reasons

as starting with WebSite-Watcher 2023, Local WebSite Archive is automatically installed and updated with WebSite-Watcher.

Homepage of Local Website Archive: https://www.aignes.com/lwa.htm

See also topic Using Local Website Archive

2.1.15 User Agent

By default, WebSite-Watcher is identified as Chrome.

If you want that WebSite-Watcher is identified as another browser or crawler, you can add a customized browser identification.

2.1.16 Check for new version

WebSite-Watcher can check itself for new versions on the internet. This feature is only available in the registered version.

2.1.17 Images

Download/Display images

By default, WebSite-Watcher replaces all images in a page with local, grey boxes. This is the fastest way to check and open bookmarks. Alternatively to the ugly grey boxes you can also download and display the real images of a page.

For each bookmark, the global image configuration can be overridden in the bookmark properties.

See chapter Display images instead of grey boxes for more information.

Don't download images which contain one of the following strings

Enter one entry per line. If an image URL contains one of the entered strings, the image will not be downloaded.

Example:
```
banner
counter.pl
/ads
```

Show embedded videos

WebSite-Watcher can optionally show certain types of videos if the option for absolute image addresses or downloading images is enabled. This only works if the used technology to display embedded videos is supported by WebSite-Watcher.

Don't show images within filtered content

If enabled, WebSite-Watcher will not download and display images within filtered content. This option only works in combination with Ignore filters, not with Watch filters.

2.1.18 Proxy

If you are connected to the internet through a proxy, you have to enter the proxy configuration in that section.

To access certain sites without going through a proxy, you have to enter the domain names of those sites (one entry per line).

Example:
www.aignes.com
website-watcher.com
345.32.123.21

If an entered line starts with a dot, then no proxy will be used for all sub-domains:

- domain.com ... will not use a proxy for domain.com
- .domain.com ... will not use a proxy for news.domain.com, download.domain.com, etc, but will use a proxy for domain.com

FTP Proxy

WebSite-Watcher supports different login types for FTP proxies, you can specify them with numbers from 0 to 8. To determine the correct number for your FTP proxy, simply click the button "Detect Login Type automatically".

FTP over HTTP

Use the FTP-over-HTTP method only if you have to connect through a proxy that doesn't support the FTP protocol.

A typical example where FTP-over-HTTP is required is the proxy solution "Squid". The FTP-over-HTTP method sends a HTTP-like request (GET "ftp://ftp.something.com") to Squid, Squid connects to the FTP server and gets a file listing. Then Squid converts this file listing into a webpage and returns it to your browser or to WebSite-Watcher.

Disadvantages and limitations:

- No direct support for password protected FTP servers or directories. The only solution to monitor a password protected directory is the insecure URL format: ftp://USERNAME:PASSWORD@ftp.something.com/

This format is not supported by all proxies, so you have to test yourself if it works.

- It's not possible to monitor a single file via directory entry, it's only possible to monitor a whole directory listing (all files in the directory will be listed)
- A proxy change or proxy update can cause a false positive

Tip: Use the FTP-over-HTTP method only if direct FTP access is not possible!

Proxy Rotation

If two or more proxies are entered, then the proxy rotation feature will change the proxy with each check run. This will make it harder for a web server to detect WebSite-Watcher as bot.

This feature is only available in the Business Edition.

Proxy List

The Proxy List section allows you to define further HTTP proxies that can be selected per bookmark (see also chapter Check Options (1)). So it will be possible to monitor different bookmarks with different proxies. A custom proxy works only for bookmarks that use the internal checking technology.

2.1.19 E-mail

When a page has been changed, WebSite-Watcher can automatically send e-mails to one or more recipients. In that section you have to enter your mail server configuration.

Server

- **Smtp Server**
  SMTP server for outgoing e-mails.

- **Smtp after Pop**
  Depended from the server configuration, a POP3 login may be required before you can send e-mails (also called SMTP-after-POP or POP-before-SMTP authentication). If this option is enabled, WebSite-Watcher will only perform a login and a logout to the POP3 server, no mails will be downloaded.

- **Pop Server**
  Is only required if you use the Smtp-after-Pop feature.

- **Connection**
  ➤ Regular - Standard login without SSL/TLS
  ➤ Full SSL - Uses SSL/TLS starting with the first contact to the server. Servers with full SSL/TLS mode usually don't use the standard TCP port (25).
  ➤ Auto SSL - Uses SSL/TLS only if it is supported by the remote server.

Template

The Template section contains basic e-mail configurations such as name and mail address of the sender or recipient. To enter more than one e-mail address to the Recipient field, you have to separate them with a comma or a blank.

Variables:

The Subject and Body fields can contain variables which will be replaced with real values when e-mails are sent. Supported variables can be found in topic Send e-mails when changes are detected

Options

- **Use absolute image URLs in HTML mails**
  If this option is enabled, WebSite-Watcher will use absolute URLs for images in HTML mails, independently if you download images to your hard disk when checking bookmarks. This option is enabled by default and can dramatically reduce the size of HTML mails. Also Facebook/Twitter/ Instagram bookmarks will benefit from that option.

- **Max E-Mail Size**
  If you send e-mails with embedded images, you can limit the max size of the e-mails. WebSite-Watcher will remove images from the e-mail as long as the size larger than the configured max value.

- **Variable {counter}**
  Incrementing counter that can be used in the e-mail subject. This counter increments by 1 with every e-mail sent. In your mailbox you could then see at a glance if a message did not come through because of a missing number.

Advanced

By default, WebSite-Watcher performs a bookmark check run and sends queued e-mails afterwards. That means, that checking bookmarks and sending e-mails must not share available resources.

With the advanced sending rules it is possible to send queued e-mails independently from bookmark check runs. With that option you have more flexibility and e-mails can be sent earlier if you have many bookmarks and a check run takes a long time.

- **Max. mails per SMTP login**
  If at least one e-mail exists in the outbox, then WebSite-Watcher will login to the SMTP server and send the queued e-mails (but not more than the max. number of configured e-mails). If there are still queued e-mails in the outbox available, then WebSite-Watcher will wait the configured time between SMTP logins, make a new login and send the next bunch of e-mails. This is

repeated until all bookmarks are sent.

- **Min. time between SMTP logins**
  Defines the time in seconds between SMTP logins.

Black list

The black list can be used to suppress sending e-mails if the page contains a pre-defined black listed text (eg. "could not connect to database").

Enter one phrase per line. Entries are case-insensitive. Wildcards or regular expressions are not supported.

Example:
```
database error
site under maintenance
```

2.1.20 WSW App

The WebSite-Watcher App can be used to read changed bookmarks on your Smartphone.

WebSite-Watcher can use any WebDAV supported cloud and file hosting service (or alternatively your personal Dropbox account) to send changed bookmarks or status messages (eg. e-mails could not be sent) to the App.

See also topic App (Android / iOS)

WebDAV

WebSite-Watcher can use any WebDAV supported clouds and file hosting services to synchronize data between the Windows version and the app.

WebSite-Watcher will save all required files in the folder "wswapp" that you can store anywhere on your server.

1. On your WebDAV server, create the folder "wswapp" (this folder can be created in the root folder or in any sub folder).

   Examples:
   /wswapp
   /sync/wswapp
   /Username/wswapp

   This is the only folder that will be used by WebSite-Watcher. Do not save any other files in that folder, they will be deleted automatically by WebSite- Watcher.

   If you enter a WebDAV URL without "wswapp" into WebSite-Watcher, then WebSite-Watcher will

ask if this folder should created automatically.

2. In WebSite-Watcher, open the program configuration

3. Click the button "WSW App"

4. Select "WebDAV"

5. Enter the WebDAV URL including the folder that you have created in step #1

   Examples:
   https://webdav.server.com/wswapp
   https://www.server.com/webdav/sync/wswapp
   http://192.168.1.101:8181/Username/wswapp

6. Enter the login credentials

7. Click "OK"
   This will perform first test if the entered configuration is correct and can be used for synchronization.

2.1.21 Pushover

WebSite-Watcher can be configured to send Pushover messages for certain actions, for example when e-mails could not be sent. Pushover messages are similar to SMS but without costs per message. Pushover messages can be received on a wide range of devices, eg. iOS, Android, macOS and Windows. If you use WebSite-Watcher 24/7 on a server and if you're a Pushover user, then this is a very quick way to get notified on your phone when a problem appears.

To be able to use this feature you need your own Pushover account. More information about Pushover can be found at www.pushover.net

Pushover configuration

To setup Pushover in Website-Watcher, you have enter the following three fields. All these data can be found in your Pushover account.

- **User Key**
  The user key that you can find in your Pushover account. It's a string with ~30 characters.

- **API-Token**
  The API-Token that you can find in your Pushover account. It's a string with ~30 characters.

- **Device**
  This setting is optional and can be used if you want to send notifications only to certain devices. If this field is empty, WebSite-Watcher will send notifications to all your Pushover

devices.

After entering these fields, you have to send a test message to make sure that your configuration is correct.

2.1.22 Ransomware Detection

Ransomware is a type of malicious program that will encrypt many of your documents (text documents, photos, videos, etc), make them inaccessible and demands a payment to provide a decryption key (or a program) to remove ransomware from the infected computer. If you're sitting in front of your PC, you'll know very quickly if you are infected with Ransomware because it will likely prompt you for payment to decrypt your files. If you have WebSite-Watcher running 24/7 on a server or separate PC, you'll probably not realize that something is wrong. With the "Ransomware Detection" feature, WebSite-Watcher will regularly check if a certain reference file has been changed or deleted. And if so, you can get notified. This feature only works while WebSite-Watcher is running.

How does it work concretely

When WebSite-Watcher is started, a unique RTF file will be created on your hard disk. As long as WebSite-Watcher is running, this file will be monitored for changes. If this file is changed or deleted, the following will happen:

1. WebSite-Watcher will stop all running tasks, for example to check bookmarks, send emails, etc.

2. WebSite-Watcher will lock and protect as many important files in the WSW folder as possible to prevent encryption (these files will be locked as long as WebSite-Watcher is running). For example bookmark databases, configuration files, AutoBackups, etc.

3. A window will be displayed on top of all other applications

4. Optional alert by e-mail (the message will be sent on-the-fly without saving it into an Outbox on your hard disk)

5. Optional alert via Pushover (the push message will be sent on-the-fly without saving it into an Outbox on your hard disk)

6. If you use the WebSite-Watcher App and check the special bookmark WSW-for-Windows, you'll also get a notification that your PC was infected with Ransomware.

How to test this feature

The "Ransomware Detection" configuration includes a "Test this feature..." button. If you click this button, the reference file is deleted as soon as you close the program configuration. You should then see an alert after a couple of seconds.

How secure is it?

There's no guarantee that WebSite-Watcher will be able to detect all types of Ransomware infections using this technique. WebSite-Watcher is only able to detect an infection if the reference file is changed (encrypted) or deleted, this only works as long as WebSite-Watcher is running. WebSite-Watcher cannot stop encryption of documents on your hard disk nor prevent infection with Ransomware. If Ransomware doesn't encrypt the mentioned RTF file or if WebSite-Watcher is not running, then WebSite-Watcher is not able to detect an infection.

2.1.23 Connections

General check configuration

- **Max. total connections**
  Number of simultaneous update checks.

- **Max. HTTPS connections per server**
  Maximal number of simultaneous connections to a single server via the HTTPS protocol.

- **Max. FTP connections per server**
  Maximal number of simultaneous connections to a single server via the FTP protocol.

- **Timeout**
  If the check of a single bookmark takes longer then the specified number of seconds, the check is aborted. This value can be overridden for each bookmark.

- **Max. download size**
  If the downloaded size of a page is larger than this specified value, WebSite-Watcher will display a warning message in the status column. It is only an information message that won't abort the check.

Advanced connection rules

The advanced connection rules offers the possibility to limit the number of simultaneous connections to specified servers or URLs.

"Pause between checks" can be used if a server allows for example only two requests per minute (the search feature in forums often has such a limitation against automated hack attacks).

Rules that are displayed in blue are fixed rules that are automatically created by WebSite-Watcher. You can increase the limits for these fixed rules via Tweaks but you cannot remove them.

Advanced connection rules and Check-Macros:

URLs within macros are not recognized, this functionality is based on the bookmark fields "URL" and "Alternative URL". If an alternative URL is entered, then this field is used, URL otherwise.

HTTP-Header

- **Accept-Language**
  Lets you configure the Accept-Language entry in the HTTP header (default value is en).
  For more information about that header search the web for "HTTP Accept-Language".

2.1.24 Dialup

If you have a dialup connection to the internet, you can enter the dialup configuration in that section. WebSite-Watcher can optionally perform a connect and a disconnect.

2.1.25 Tweaks

The Tweaks section allows you to fine tune WebSite-Watcher, to enable special features or workarounds.

The Tweaks section uses the format of ini files but without sections (names between "[" and "]"), simply make one or more entries without specifying a section. Changes in the Tweak section require a restart of WebSite-Watcher to take effect.

- **BrowserButtonMoveLeft=nnn**

  With this setting you can horizontally move the WebSite-Watcher button that can be displayed in the browser title bar. nnn is the horizontal value in pixel. A positive value moves the button to the left, a negative value moves the button to the right. This is useful to avoid overlapping when other programs display buttons in the browser toolbar, too.

- **UnfocusedColorRGB=**
  Define the color (in RGB format) of selected, unfocused folders and bookmarks.
  For example, UnfocusedColorRGB=ff0000 displays the unfocused selections in red.

- **UseProxyIfUrlExists=**
  If an URL is entered, WebSite-Watcher tries to access this URL before the first check run. If it is reachable, WebSite-Watcher will use the defined proxy settings (general options), otherwise WebSite-Watcher uses direct internet connection. If you place a "+" (without quotes) before this URL, it is validated with each check, otherwise this URL is only checked once until you restart WebSite-Watcher. This setting is used for the http and https protocol.

  Examples:
  ```
  UseProxyIfUrlExists=http://www.DOMAIN.com
  UseProxyIfUrlExists=+http://www.DOMAIN.com
  ```

- **PerformCheckOnlyIfUrlExists=URL;ValidText**
WebSite-Watcher checks the entered URL before a bookmark-check run is performed. The checked URL must contain the specified "ValidText" to ensure that this page is really available. If the specified text is not available in the page, a bookmark check run will not be performed. This feature is useful when you use different internet connections (eg. at different locations) and want to ensure that WebSite-Watcher can connect to the internet. If you place a "+" (without quotes) before this URL, it is validated before each check run, otherwise this URL is only checked once after starting WebSite-Watcher.

  Examples:
  ```
  PerformCheckOnlyIfUrlExists=http://www.google.com;Advanced Search
  PerformCheckOnlyIfUrlExists=+http://www.google.com;Advanced Search
  ```

  This example downloads the page www.google.com and starts a bookmark check only if the page contains the phrase "Advanced Search".

- **BookmarkDblClickOpens=web|new**
Alternative bookmark double click operation to open the online or new version of a page. Valid values are "web" and "new".

- **OpenUrlInsteadOfAlternativeUrl=1**
If an "Alternative URL" is entered, then WebSite-Watcher will open the Bookmark URL instead of the "Alternative URL" when you open the online version of the bookmark.

- **CheckSortedBookmarkList=name**
Sorts the bookmarks by name before adding them to the checking queue.

- **BrowserAcceptLanguage=**
Lets you configure the "accept-language" HTTP header. This tweak will be used for the internal browser and for bookmark checks that are based on Chromium (Browser: Chromium, Chromium-Macros and Screenshot: Chromium). For more information about this header search the web for "HTTP Accept-Language".
Default value: en-US,en
Example: BrowserAcceptLanguage=de-DE,de,en-US,en

- **BrowserActiveTabColorRGB=**
Defines the color of the active browser tab (in RGB format).
For example: BrowserActiveTabColorRGB=f0f0f0

- **CreateHighlightedChangesInBackground=1**
By default, WebSite-Watcher calculates and generates the file with highlighted changes as soon as you open a bookmark. If this option is enabled, then WebSite-Watcher creates the file with highlighted changes immediately while a bookmark is checked. This will speed up opening changed bookmarks, but the check run will consume more memory and resources. It's **not**

recommended to use this tweak if you often re-check changed/unread bookmarks without opening them after each check.

- **FolderCheckAndShowRecursive=1**
This tweak is an enhancement to the command "Check Folder and Subfolders" (folder context menu) and will automatically show bookmarks in subfolders.

- **FollowLinksShowDepth=1**
Shows the link depth on the right side of the Follow-Links symbols .

- **SuppressInitializedInfoBar=1**
This option can be used to suppress the yellow browser information bar that is displayed when you open an initialized bookmark.

- **BookmarkPropertiesDefaultButton=check**
Sets the "Check Now" button in the bookmark properties as default button. This is the button that is used when closing the bookmark properties with the Enter key.

- **CheckLocalFolderUseCreationDate=1**
If local directories are checked for changes, WebSite-Watcher uses the modification-date for folders (folders are displayed as changed when files in these folders are changed). If this option is set, then the folder creation-date is used.

- **ReopenTabsAfterCrash=0**
If WebSite-Watcher crashes, then the next start will show a dialog to re-open the last opened browser tabs. This tweak suppresses that dialog and don't re-open browser tabs.

- **AutoWatchGuard=1**
Checks every 5 minutes whether AutoWatch is running. If AutoWatch is disabled, then it will be started automatically. Useful if you sometimes need to disable AutoWatch and forget to enable it again.

- **MinAutoWatchSecondsAfterHibernation=**
If an AutoWatch countdown is active while the system turns into hibernation, WebSite-Watcher will wait at least 15 seconds before the next AutoWatch check is started after the system returns from hibernation. This gives Windows some time to re-establish network connections before the next AutoWatch check run is started after hibernation. Default value is 15 seconds.
Example: `MinAutoWatchSecondsAfterHibernation=60`

- **PauseCheckIfWindowExists=**
Pauses AutoWatch or a running bookmark check if one of the defined windows is in foreground. You can, for example, use this feature when working with a certain program or playing a game and you don't want to check bookmarks in that time. You can use the Windows Task Manager to figure out the window title.

Define one or more window titles, separated by the ; character. WebSite-Watcher will check if one of the defined strings is part of the title of foreground window to trigger a pause. All definitions are case insensitive.
<u>Example</u>: PauseCheckIfWindowExists=notepad;My Game;Video Player

- **SendMailOnlyNewRssArticles=0**
  By default, the bookmark action "Send Mail" sends only new articles of an RSS feed. If this tweak is set, then all articles of an RSS feed (as you can see them when opening the bookmark) will be included in the e-mails.

- **SendMailRemoveWswTags=0**
  By default, WebSite-Watcher removes help tags for highlighting changes from the e-mails to keep e-mail sizes as small as possible. With that tweak, no tags are removed. This can be useful if you want to post process e-mails and extract changes via these tags (for example wswchange1, wswchange2, etc).

- **ConvertRssTitleLength=**
  Length of the article titles when WebSite-Watcher converts Facebook/Twitter/Instagram pages to RSS feeds. Can be a number between 30 and 200, default value is 40.

- **HighlightChangesExactHigh=1**
  Improves accuracy of the highlight method "Excact". This tweak uses a more exact algorithm to highlight different words in similar paragraphs. This method requires more resources and might work a bit slower. Use it only if the default behavior of the "Excact" method is not accurate enough. When using that tweak, WebSite-Watcher is not able to strike out filtered content.

- **UptimeControlFile=**
  This tweak will write a control file once a minute and can be used to verify if WebSite-Watcher is up and running (Business Edition only).
  <u>Example</u>: UptimeControlFile=c:\path\file.dat

- **ExtSearchHighlightCSS=**
  CSS definition for highlighting found words in the Extended Search.
  Default CSS definition is: `color:#ff0000;background:#00ff00;`
  To put a border around the highlighted words, the following line can be used:
  `ExtSearchHighlightCSS=color:#ff0000;background:#00ff00;border:thin;border-color:#ff0000;border-style:solid;border-width:1px`

- **AutoDeleteOutdatedIgnoreFilters=0**
  By default, WebSite-Watcher automatically detects and removes outdated ignore filters. With that tweak you can disable that mechanism but you can get very long ignore filter lists over the years if you only use the filter assistant to create new filters interactively. It's <u>not</u> recommended to disable that system unless you want to maintain your filter definitions manually. Keep in mind that many defined filters need more resources and time to be processed, even if they are outdated.

- **HandleIgnoreUpdateAsError=1**
  This tweak refers to the Ignore updates feature in the bookmark properties. If a black listed word/text is found in the page, WebSite-Watcher will ignore the update but not list the bookmark in the error folder nor alert an update after X errors in a row. If WebSite-Watcher should list the bookmark in the error folder and alert an update after the configured number of errors in a row, you have to use this program program tweak.

- **HandleWarningWholeContentFilteredAsError=1**
  If a check reports the message "Warning: whole content filtered (please verify your filter settings)", then this is handled as warning by default. If you enable that tweak, this status message is handled as error (listed in Error folder, the bookmark option "Alert/Actions-on-Error" is performed, etc).

- **QuickInsertFromWebBrowserContextMenu=1**
  If you insert a bookmark via the webbrowser context menu, WebSite-Watcher usually shows a dialog with name/URL/folder selection. If this tweak is set, bookmarks are silently inserted to the root folder of the opened bookmark list without displaying that dialog. If a bookmark already exists, then the URL is not inserted.

- **CheckIEBrowserSingleConnection=1**
  That tweak limits the simultaneous connections of bookmarks with the checking method "Internet Explorer/Browser" to 1 (same behavior as in WebSite-Watcher 2015 and lower). You can use that tweak if simulatenaous connections with the checking method "IE/Browser" cause problems on your system.

- **UseStaticTrayIconHint=1**
  Shows the static text "WebSite-Watcher" as tray icon hint (tooltip) instead of a dynamically generated info about the current check/update status.

- **crTwitterPauseSec=**
  Advanced connection rule for Twitter bookmarks. Defines the number of seconds between checks. Valid values are 0..600.

- **crInstagramConnections=**
  Advanced connection rule for Instagram bookmarks. Defines the number of simultaneous connections. Valid values are 1..3.

- **crInstagramPauseSec=**
  Advanced connection rule for Instagram bookmarks. Defines the number of seconds between checks. Valid values are 0..600. Will only take effect if the number of simultaneous connections is set to 1.

- **crFacebookConnections=**
  Advanced connection rule for Facebook bookmarks. Defines the number of simultaneous connections. Valid values are 1..3.

- **crFacebookPauseSec=**

  Advanced connection rule for Facebook bookmarks. Defines the number of seconds between checks. Valid values are 0..600. Will only take effect if the number of simultaneous connections is set to 1.

- **BlindUser=1**

  Optimized user interface for blind users.

Example:

```
BrowserButtonMoveLeft=100
BookmarkSort=-date_changed
```

2.2   Bookmark properties

2.2.1   General

General

- **Name and URL**
  The URL field is required. If an URL contains a changing daily date, you can use date variables to get this solved. See chapter Date variables for more information. If the Name field is empty, the webpage name will automatically be assigned with the next bookmark check.

- **Alternative Check-URL**
  If an address is entered, WebSite-Watcher will monitor this URL instead of the entered address in the URL field.

- **"Check-Macro" button for password protected pages**
  If you want to monitor a password protected page that is located behind a login page, you have to use a Check-Macro. Check-Macros can also be used for pages that require more steps to get to a certain page, for example search queries. See chapter Check Macros for more information.

- **Convert page to RSS**
  This option is only available for certain pages, for example Facebook, Twitter or Instagram pages. If enabled, WebSite-Watcher will convert the page to an RSS feed what makes it much more easier to monitor these pages. WebSite-Watcher will only alert new articles, not changes in existing articles. See also Facebook/Twitter/Instagram pages

See also:
- Using the Wizard
- Add bookmarks manually
- Facebook/Twitter/Instagram pages
- RSS / Atom feeds

- FTP files and directories
- Binary files
- PDF files
- Images

Cosmetic Filter

Cosmetic filters will completely delete unwanted content from a page and produce a new page which is then processed by WebSite-Watcher. Ignore/Watch filters are then applied to the page that is created via the cosmetic filters.

See also:
- Cosmetic Filter

Filter when checking web page for changes

To detect updates, WebSite-Watcher compares the text content of the new version of a page with the text content of the old version of a page. If these contents are different, WebSite-Watcher alerts an update. If a page contains changing parts (for example a daily date, some kind of advertisement, ...), an update would be reported with each check. A filter can ignore specific parts of a page to prevent false update notifications.

If a filter is defined, WebSite-Watcher will compare the **filtered text content of the new version** with the **filtered text content of the old version** of a page to detect updates.

Filters can be created with the Auto-Filter system, the Filter-Assistant or manually. By default, the filter "Ignore all HTML-Tags" is enabled and returns only the real, readable text content of a page. This filter should only be disabled under special circumstances, for example if you want to monitor any changes in the page source code.

The feature "Test filter" gives you always the possibility to verify your filter settings by comparing the filtered content of the new version with the filtered content of the old version of the page.

See also:
- What are filter settings (Overview)
  - Auto-Filter
- Filter Assistant
- Create Ignore / Watch filter manually
- Special Filter
- Filter with Wildcards
- Working with Regular Expressions
- Cosmetic Filter

Update similar bookmarks

If you have changed a configuration in the bookmark properties directly and want to apply these changes to other bookmarks, you can check the option "Update similar bookmarks" before you press the "OK" button. This will display a window that lets you select the target bookmarks and the options you want to copy from the actively opened bookmark. See also topic Change properties in multiple bookmarks.

2.2.2   Keywords

Highlight keywords

This feature lets you highlight entered keywords in a page. Whole keywords must be placed in quotes, for example "watch". All entered keywords are case insensitive.

Keywords are only highlighted in the version with highlighted changes, not in the new or old version of a page.

Regular Expressions must be placed between "`regex(`" and "`)`" (without quotes).
For example: `regex(WebSite.Watcher)` which will find "WebSite-Watcher", "WebSite Watcher", "WebSite_Watcher", ...

Restriction: Keywords can only be highlighted when no HTML-Tags are placed within these words (for example the text "`<b>H</b>ello`" contains HTML-Tags, it's not possible to highlight the keyword "Hello"). If WebSite-Watcher cannot highlight all entered words, a warning is displayed at the beginning of the page.

Different keywords can also be highlighted with different colors, just add #COLORNAME on the right side of the keyword. Supported color names are: red, green, olive, lime, teal, blue, aqua, magenta, orange
Example:
website #blue
watcher #red

Update on keywords

The Update-on-keywords feature only alerts an update, if at least one of the entered keywords is available in new or changed content. If none of the entered keywords is found in a page, WebSite-Watcher will not save the checked page to your hard disk. When you open the bookmark, you'll always see the latest version that contained at least one of the entered keywords.

Enter one keyword per line. All entered keywords are case insensitive.

- **Parts of a word**

By default, WebSite-Watcher searches keywords anywhere in new and changed text content, also when they are part of a word. If you enter the keyword **watch**, you will also get a positive result when the word **WebSite-Watcher** is available.

- **Whole words**

  If WebSite-Watcher should only find whole words, then you have to place them in quotes. The keyword **"watch"** (with quotes) won't give you a positive result when the page contains **WebSite-Watcher**, but **"watch"** will be found in the text **you can watch a page for**.

- **Combination of words**

  Words in a single line can be combined with a logical AND, so that all words must be found in new/changed content. Keywords can be combined by placing a " + " (space-plus-space) between them. Whole keywords are supported by placing the keywords into quotes.
  Examples:
  keyword1 + keyword2
  "keyword1" + keyword2 + "keyword3"
  "website" + "watcher"

- **Regular Expressions**

  The keywords you are looking for can also be defined with Regular Expressions. Regular Expressions must be placed between "`regex(`" and "`)`" (without quotes).
  For example: `regex(WebSite.Watcher)` will find "WebSite-Watcher", "WebSite Watcher", "WebSite_Watcher", ...
  A special function is regexcmp(...) that allows you to compare prices with a specified number.

Save page even if no keywords found:
This option is turned off by by default and WebSite-Watcher does not save the page when no keywords are found. So you'll always see the last version where keywords have been found. That behavior works for pages where keywords are searched in new content.
Turn that option on if a certain keyword always appears/disappears at the same place, for example the keyword "in stock" in a shop page.

Defined Filters:
WebSite-Watcher does not search keywords in content which is filtered with Ignore filters. Defined Watch filters are not used when searching for keywords.

Plugins:
With the integrated Plugin system you can also define complex keyword queries with logical operators. More details can be found in topic Wsw_CheckKeywords. If Keywords and a Plugin are defined, then both tasks are executed and WebSite-Watcher will alert an update if at least one of these two features find a keyword.

See also:
- Monitor pages for keywords
- Regular Expressions

2.2.3   AutoWatch

WebSite-Watcher supports an automated checking method (called AutoWatch) to monitor bookmarks automatically via predefined time configurations. You can turn on this mode with the AutoWatch button ⏱ in the toolbar.

Here you can define, if and how often a bookmark should be checked when WebSite-Watcher is in AutoWatch mode.

Automatic check interval

The option "Automatic check interval" allows WebSite-Watcher to automatically use optimized check intervals based on previous updates.

Active pages will be checked more often, pages that lay dormant for months will be checked less often. This option is especially for bookmarks where it isn't really important to get an update notification as soon as possible.

The minimum/maximum time can be defined in the program configuration.

See also:
- Check pages automatically (with AutoWatch)

2.2.4   Check

Alert/Actions on

With the "**Alert/Actions on**" option you can define when an update notification and bookmark actions (eg. "Send e-mail") should be performed. Default value is "Alert-On-Changes", that means that you get only update notifications when a page is changed.

See also "Alert with the first initialization of a page" and global error handling in the program configuration.

Technology to check web pages

- **Internal**
  This is the default method to check pages. You should only try alternative methods if this method fails.

- **Browser**
  Use this method if WebSite-Watcher cannot check a page with the internal check technology, for example when a web page can only be displayed with Javascript.
  By default, WebSite-Watcher filters all Javascript content from a page and works only with the

static, textual content. This technology opens the page in a hidden browser window and converts the "dynamic Javascript content" to a static, textual content that can be used and processed by WebSite-Watcher. Supported browser engines are Chromium, Edge and Internet Explorer.

Delay after page has loaded (in seconds):
This option must be used for pages which change the content dynamically after the page has loaded. For example when it takes a few seconds until the final content is displayed.

Scroll pages down:
Number of pages that should be scrolled down during a check. Use this option if a page displays information dynamically when the page is scrolled down.

Alternative method to get page content:
This method analyzes all transferred documents and collects them to a single page. This method might be useful if WebSite-Watcher cannot resolve IFRAMES or other embedded elements. There are different methods that can be used on a try-and-decide basis.

- **Screenshot**

  If a page contains content that cannot be monitored with the other methods (eg. content that is displayed with Javascript), you can try to create a screenshot of that page and compare that screenshot to detect changes. See chapter Check by screenshot for more information. Supported browser engines are Chromium and Edge.

See also:
- Check pages which require Javascript
- A page cannot be displayed
- Check by screenshot

2.2.5   Actions

When an update is detected, WebSite-Watcher can optionally perform additional tasks.

Play sound

Enter a sound file in WAV format that should be played when an update is detected.

Open page

When a page has been updated, you can automatically open it in the internal or an external browser. This feature is only useful for single pages (not for all pages in the list).

Send e-mail

The "Send e-mail" feature gives you the possibility to send e-mails to one or more recipients. WebSite-Watcher will send a separate e-mail for each changed bookmark.

- **Send full page**
  Sends the full page with all CSS definitions and downloaded images. Main goal is that the e-mail will look like similar to the page in your Internet Browser. Please note that many e-mail clients or Webmail services do not support all formatting commands, so the layout may be broken in your e-mail client. In this happens, you can choose one of the other sending options.

- **Send simple version**
  This sending option strips the layout of the page and will only send a simplified version. This is similar to the TEXT tab of the internal browser. Use this option if your e-mail client has problems displaying the full layout or formatting of the page.

- **Send only changes**
  This feature extracts and sends only the highlighted text of the page. This option provides the best compatibility with various e-mail clients, it also reduces the size of the e-mails.

- **Attach screenshot**
  This option attaches a screenshot with the highlighted changes. This option can be used in combination with the feature "Send only highlighted text" and might be useful when the layout of the page is required to understand the highlighted changes. Please note that the size of the screenshot is limited and you might not see all changes.

- **E-Mail Template**
  Lets you assign a customized e-mail template (optional). If assigned, then the template configuration will override the global e-mail configuration and can be used to send e-mails with an alternative "From" e-mail or an alternative body.

- **Send before other e-mails**
  The option "Send before other e-mails" will insert an e-mail to the top of the outbox. So these e-mails have higher priority in the outbox and will be sent before e-mails without that flag. Use this option only for single (critical) bookmarks, not for most or all bookmarks.

See also:
- Send e-mails when changes are detected

Export page

The export feature saves the updated page (new version without highlighted changes) to a specified filename.

Variables in the filename:

{id}	Bookmark ID
{datetime}	Current date/time with format yyyy-mm-dd hh.mm.ss (ms)
{infofield1}	Info field #1

{infofield2}	Info field #2
{infofield3}	Info field #3
<$date$, 'format'>	Date-Variable as described in topic Date variables

Format:

Website, only HTML	WebSite-Watcher exports the page without CSS and image files.
Website, complete	WebSite-Watcher can automatically export related CSS and image files to the sub folder with the extension ".files". If the export filename is "test.htm", then exported CSS/image files are stored in the sub folder"test.files". WebSite-Watcher doesn't automatically remove outdated files from that folder when you re-export bookmarks to the same filename.
Text	Optionally you can strip all HTML tags, then the HTML page will be saved as text file.

Run program

This action can start a program with the following variables as parameter:

{url_new}	new version of the downloaded page
{url_changes}	version with highlighted changes
{url_old}	old version of the downloaded page
{url}	URL of the bookmark
{postdata}	PostData field of the bookmark
{infofield1}	Info field #1
{infofield2}	Info field #2
{infofield3}	Info field #3

If more than one program should be executed, you can create a DOS-Batch file and start these programs via the batch file.

Local Website Archive

This action exports a page to the tool Local Website Archive. Local Website Archive must be installed separately and can archive web pages and documents permanently for future reference. More information about Local Website Archive can be found at https://www.aignes.com/lwa.htm

See also topic Using Local Website Archive

Pushover

This action will send a Pushover message to your Pushover client. To be able to use this feature you need your own Pushover account. More information about Pushover can be found at www.pushover.net

The size of Pushover message is limited, so you might not see all changes.

Mark as Read

If you don't open bookmarks within WebSite-Watcher (eg. when reading changed bookmarks only in your e-mail client), then you should also enable the Mark-As-Read action. Otherwise the old version will not be replaced with the new version and you will always see all changes between the (not updated) old version and the new version from the web.
Starting with WebSite-Watcher 2017, the Mark-as-Read action is no longer required for the actions "Send e-mail" and "Open bookmark".

Script

Allows you to execute certain commands when a bookmark is updated. Scripts are line oriented, that means you have to put each command into a single line.

The following commands are supported:

CheckFolder	Allows you to check a bookmark folder when a certain bookmark is updated. Examples: CheckFolder "News" CheckFolder "Software\Mail"

2.2.6 Options

Jump to the first change / keyword

- **Jump to the first highlighted change**
  If enabled, WebSite-Watcher automatically jumps to the first highlighted change when a bookmark is opened.

- **Jump to the first keyword (within highlighted changes)**
  If you have keywords specified, you can automatically jump to the first keyword (within highlighted changes) when opening the page.

- **Offset before first change**
  When WebSite-Watcher jumps to the first highlighted change, then it is displayed directly at the top border of the internal browser. With that option you can automatically scroll back the specified number of pixels. This can make reading the first highlighted change a bit more comfortable, but might take a bit longer to scroll to that position. Default value is 0.

- **Prevent horizontal scrolling**
  If WebSite-Watcher jumps to the first change/keyword and the page width is larger than the with of the browser window, then a horizontal scroll might be performed. Enabling this option will prevent horizontal scrolling. This works only for changes that are completely or partially visible. If a change is placed completely outside the visible browser window, then a horizontal scroll will always be performed to see this change.

Show elements

- **Show removed elements at the bottom of the page**
  If this option is enabled, WebSite-Watcher displays removed elements at the bottom of the page.

- **Show last change marker**
  If enabled WebSite-Watcher shows a marker after the last change. The text and color of that marker can be defined in the program configuration.

Options

- **Disable bookmark (Don't check)**
  With that option you can disable a bookmark and exclude it from checking. Disabled bookmarks can only be checked with the feature Check selected bookmarks (extended).

- **HotSite**
  HotSites are bookmarks with a certain flag enabled. All HotSites are listed in the virtual folder "HotSites" independently from the folder they are really saved. This gives you the advantage to display your favorite bookmarks within a single folder. See chapter HotSites for more information.

2.2.7 Open bookmark

Select which version of the page should be opened and if a bookmark should be opened in the internal browser or in your external default browser. You can for example open a bookmark in your external browser if the internal browser causes problems or crashes for that bookmark.

2.2.8 Time Settings

You can use these options to enter some time configurations for manual bookmark checks (shortcut F9 or the toolbar Check button ✅).

If no configuration is entered, the bookmark will be monitored with each check run.

2.2.9 Highlight changes

Highlight method

Dependent from the page type, WebSite-Watcher supports several methods to highlight text changes.

See chapter Highlight methods for more information.

Options

- **Highlight changed images**
  Dependent from the image configuration, WebSite-Watcher can place a border around changed images. This option works only when you download images or use absolute image addresses to display images. Please note that a web page is only marked as updated when a text change has been detected, and not when only an image has been changed!

- **Watch filter: Highlight only matched results**
  If you create an ignore filter, then WebSite-Watcher tries to strike out the filtered content. If you create a watch filter, then nothing is striked out by default and WebSite-Watcher highlights all changes in the page. With that option enabled, WebSite-Watcher only highlights the changes that are found within the watch filter and strikes out everything else. Then you'll have the same behavior as in the filter assistant.

See topic How are changed images detected? for more information.

Colors

Here you can define color and style of highlighted changes

2.2.10 Images

The Image section allows you to override the global image configuration.

See topic Display images instead of grey boxes for more information.

2.2.11 Plugin

The Plugin system provides a powerful way in which the functionality of bookmark checks can be changed or enhanced by writing small "Basic" scripts.

WebSite-Watcher supports Private Plugins (which are created and saved per bookmark) and Shared Plugins (which are saved in separate files and can be shared between bookmarks).

Plugin specific data

This section contains fields that may be used by a plugin. For example the password field to check

password protected Excel files. A Plugin can read this field with the function Bookmark_GetProperty.

See also:
- Plugin System - Overview
- Shared/Private Plugins
- Event functions

2.2.12 Follow Links

By default, you have to create a separate bookmark for each page you want to monitor.

If a whole site instead of single pages should be monitored, you can use the **Follow Links** functionality. This feature will automatically insert all links of a page as new bookmarks. With several options you can specify how many or which links should be inserted as new bookmarks, for example the link depth.

The Follow Links requires a Master-Bookmark that is the root for all other automatically inserted child bookmarks. Master bookmarks displays the image 🔗 before the bookmark name and are usually sorted before all other bookmarks.

For all child bookmarks the Follow Links configuration is disabled. Changes to the Follow Links configuration can only be made in master bookmarks.

It's recommended to put each Master-Bookmark into a separate folder!

Options

- **Link depth**
  The link depth defines the depth that WebSite-Watcher should follow. A link depth of 1 means that WebSite-Watcher will follow only links from the current page. With a link depth of 2 WebSite-Watcher will also follow links from linked pages, etc...

- **Copy all bookmark properties when inserting new bookmarks**
  By default (without that option) only the most important bookmark properties are copied to automatically inserted bookmarks. If this option is enabled, nearly all bookmark properties from the source bookmark are used in the new, inserted bookmarks.

- **Automatically delete no longer linked pages every X days**
  This option allows you to automatically detect and delete no longer linked pages from the bookmark list. Since this operation takes some time you can configure how often it should be performed. If you right click a master bookmark in your bookmark list, you can also manually perform this operation via the context menu.

- **Don't follow links which contain one of the following strings**
  With that option, new bookmarks will NOT be inserted if a link address contains at least one of the entered strings. Enter one string per line, all strings are case insensitive. The entered strings

are "sub-strings". Regular Expressions (with function *regex(...)*) and date variables are supported. Wildcards or other regular expression functions than regex(...) are <u>not</u> supported (eg. StartToRegex).

<u>Examples are:</u>
```
.zip
.exe
/adv/
regex(/example\d+/)
```

- **Only follow links which contain one of the following strings**
  With that option, new bookmarks will only be inserted if a link address contains at least one of the entered strings. Enter one string per line, all strings are case insensitive. The entered strings are "sub-strings". Regular Expressions (with function *regex(...)*) and date variables are supported. Wildcards or other regular expression functions than regex(...) are <u>not</u> supported (eg. StartToRegex).

  <u>Examples are:</u>
  ```
  techinfo
  forum_data
  regex(/article\d+/)
  ```

- **Name Prefix**
  If entered, then automatically inserted bookmarks will have the defined text as prefix in the bookmark name. Example: [wsw]

- **Don't follow links inside filtered parts**
  With that option, the Follow-Links feature uses defined Ignore filters to exclude links from filtered areas of a page. This option only works with Ignore filters, not with Watch filters.

Advanced Options

- **Check if an URL already exists in WebSite-Watcher**
  If an URL contains a changing Session-ID, it will be inserted each time you check the site. To prevent that behavior, you can define regular expressions to exclude parts of an URL when WebSite-Watcher checks if it already exists. Enter one regular expression per line. Entering the function text "regex(" and ")" is optional, not required.

  <u>Examples are:</u>
  ➢ `?.*`
    filters everything after the question mark
  ➢ `&s=.*?&`
    filters the parameter &s= until the next ampersand, for example &s=hasdqwe79a8sd798&
  ➢ `&sid=[a-z|1-9]*`
    filters a session id with characters a-z and digits

See also:

- Monitor a whole site (Follow Links)
- Regular Expressions

2.2.13 Merge pages

If a page consists of multiple pages (eg. search results) and the URL provides a parameter that defines the page number, then WebSite-Watcher can merge these multiple pages to a single page.

Typical examples of supported URLs:

- http://www.domain.com/search.php?s=text&**page=3**&tid=792837492387
- http://www.domain.com/topic.php?**start=15**&sort=asc

There are two different ways how pages can be merged

- **Merge pages with a permanent start page**
  Use this option if you want to monitor for example the first five pages of a search result.

- **Merge the latest pages with new content**
  Use this option if you want to monitor for example new posts in a forum thread that are added to higher page numbers. WebSite-Watcher will then automatically monitor the latest pages which have new postings.

To setup the merge feature, simply start and follow the assistant.

Alternative solution via plugins

You can also write a plugin to automatically merge pages, this will then work also for URLs where the page number is part of the URL/filename and not available as parameter. See example Merge search result pages and Merge pages in forum thread.

2.2.14 Ignore updates

Ignore updates via Blacklist/Whitelist

The "Ignore updates" feature supports two different modes:

1. **Blacklist mode**
   This method will prevent an update notification when a page contains a certain pre-defined text. This feature can be used when a page regularly shows a certain error message, for example "Too many users online, please try again later" or "Could not connect to database".

2. **Whitelist mode**
   This method only allows an update notification when a page contains a certain pre-defined text. If this pre-defined (white listed) text is not found in the page, then WebSite-Watcher will not

alert an update.

See also topic Ignore updates

Required number of characters

If a server sometimes sends an empty page, then you can use this option to prevent false positives. Enter the minimum number of readable characters that must be available in a page.

2.2.15 IFRAMES

IFRAMES (inlineframes, embedded pages) are often used to display advertisements within a web page.

WebSite-Watcher doesn't check IFRAMES automatically. If a page contains IFRAMES, the internal browser displays the browser information bar and you can optionally enable IFRAME checking or suppress the IFRAME warning.

2.2.16 Login

Login via Server Authentication

Server Authentication is a special method to protect pages from public access. You can identify that type of login, when a small dialog pops up where you have to enter username and password, for example:
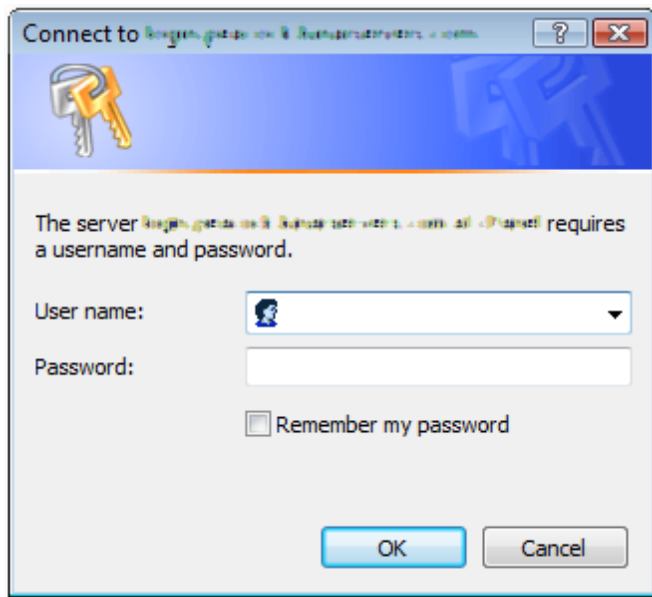
If you get such a dialog, you have to enter username and password here.

Web based login

If the fields for username and password are placed within a web page, then it is a web based login and you have to record a Check-Macro.

See also:
- Server Authentication
- Web based logins / Check macros
- Password protected FTP files

2.2.17 Check Options (1)

Options

- **Ignore error codes from server**
  Enable this option if a misconfigured server permanently sends an error code for an existing page, for example "404 file not found".

- **Allow redirection via HTML commands**
  If a web page is only reachable via HTML redirections, you can enable this option. WebSite-Watcher will download the first page, scan the source code for HTML redirections and check the target page afterwards.

- **Suppress redirection warnings**
  If you monitor a page that redirects to another address, WebSite-Watcher always shows "redirection" in the status column. With that option you can suppress this status message.

Checking Method and file type

- **Checking method**
  WebSite-Watcher supports several checking methods to monitor a page. The default method is "Automatic" that lets WebSite-Watcher choose the best method automatically, or dependent from the page type a combination of them. Only in special cases you should select another checking method manually.

  ➢ Automatic
    Dependent from the page type, this method decides automatically which method should be used to check a page. This can also be a combination of the available methods.

  ➢ Check-by-content
    This method downloads the entire page and compares the filtered content of the new version with the filtered content of the old version of the page. If differences are found, WebSite-Watcher alerts an update.

➢ Check-by-filedate
This method uses the file date of a page to detect updates. It's dependent from the page type, if a server reports that date. This method is typically used for binary files.

➢ Check-by-filesize
This method works similar to the Check-by-filedate method and uses the file size that is reported by the server. This method should only be used for binary files.

- **File type**
If WebSite-Watcher is not able to detect the file type automatically, you have to select the correct file type manually. If WebSite-Watcher is for example not able to identify an RSS feed, you have to select the file type "rss". A reason for such a situation could be that the server identifies an RSS feed falsely as binary file.

- **Character set**
By default, WebSite-Watcher determines the character set of a page automatically. If neither, the page nor the server log contains any information about the character set, WebSite-Watcher is not able to display the page correctly and you have to choose the codepage manually. See chapter A web page contains only cryptic characters for more information.

Alternative Proxy

If you want to use a separate proxy to check a bookmark, you can select a configured proxy from the list. See also chapter Proxy. This setting is only used for the internal checking method, the check technologies "Chromium", "Edge" and "Internet Explorer" will always use the proxy configuration of your system.

2.2.18 Check Options (2)

Cookies

The Cookies configuration is only used by the internal check technology. The technologies "Chromium", "Edge" and "Internet Explorer" have their own Cookie system.

WebSite-Watcher supports the following methods to handle cookies:

- **Use Cookies**
This is the default method, WebSite-Watcher uses cookies from an internal Cookie database. To get Cookies for a bookmark, right click the bookmark in the main window of WebSite-Watcher, select "Browser" to load the page (perform logins etc), then close the window.

- **Use custom Cookies**
Alternatively to the Cookie database, WebSite-Watcher can manage Cookies individually for each bookmark. The Cookies are stored in the bookmark properties and are not shared with other

bookmarks.

- **Don't use cookies**
As the caption says, WebSite-Watcher doesn't use cookies.

Referrer

In some cases a web page can only be opened when you open it from another specific page. If a web page needs a specific previous page (referrer), you can enter this URL into that field.

GET / POST

The POST feature is available for professional users only and is usually a bit faster than using a Check-Macro. Normally you should record a Check-Macro for pages where the POST command is involved.

By default, WebSite-Watcher uses the GET command to monitor web pages. If a page requires the POST command (for example after filling out a form), then you can use the tool MiniBrowser to determined the POST URL and PostData. The field PostData can also contain date variables.

See also:
- Web based logins / Check macros

User Agent / Timeout

- **Use customized Browser-ID**
By default, WebSite-Watcher is identified as Chrome. This identification can be overridden optionally per bookmark.

- **Override Timeout**
Here you can override the default timeout value from the program configuration.

2.2.19 Forum

The topic list of discussion forums can be checked with pre-defined forum plugins or with **Forum-Templates**.

Forum plugins are available for well known forums (eg. phpBB, SMF, vBulletin, etc). It's enough to assign a forum plugin (if available) no further configurations are required in that case.

If a forum is not supported by a forum plugin, you can create a **Forum-Template** by selecting the Forum-Header, the last posting time (of a thread/topic) and the forum footer. Then WebSite-Watcher will create all required settings automatically, no further configurations are needed.

See also topic Table based forums.

2.2.20 RSS/Atom

These section can be used to enter some basic RSS/Atom feed configurations.

- **RSS-Style**
  WebSite-Watcher supports different layouts to convert RSS feeds into a readable HTML page.

- **Ignore changes in old postings**
  By default, WebSite-Watcher alerts an update if new or changed postings are found in an RSS feed. If you are only interested in new posts but not in updated articles, then you have to enable this option.

- **Keep the most recent X articles**
  By default, WebSite-Watcher shows only the RSS feed items that are really saved in the RSS feed file. Enabling that feature allows you to keep a small archive of older postings in the feed. This feature can also be used when an RSS feed dynamically adds and strips old articles (seen in rare cases).

- **Ignore articles older than X days**
  By default, WebSite-Watcher shows all articles that are stored in an RSS feed. If an RSS feed contains thousands of old articles, then the result would be a huge HTML file that would slow down WebSite-Watcher. With that feature you can ignore articles older than the specified number of days to keep RSS feeds small and fast.

- **RSS item identification**
  Defines which fields should be used to detect new RSS feed postings. By default, WebSite-Watcher uses an automatic system that can fail under certain circumstances, for example when an RSS feed changes the GUID (=Globally Unique Identifier) regularly for existing postings. In such a case you can select the fields manually which should be used to detect new items. The automatic system should however work in nearly all cases.

- **Ignore article text**
  If enabled, WebSite-Watcher will completely ignore and remove the text bodies of articles, you'll only see the fields titles, pubdate, etc.

- **Blacklist / Whitelist**
  It's possible to define either a Blacklist or a Whitelist per RSS feed. The Blacklist mode allows you to ignore articles that contain at least one of the specified words. If a blacklisted word is found, WebSite-Watcher will completely remove the article from the list. The Whitelist mode will only monitor articles that contain at least one of the defined words, all other articles are ignored and removed. Enter one word per line. Wildcards and regular expressions with syntax regex(...) are supported.

2.2.21 Report

In that section you can enter additional fields for the Report/Export feature.

If the fields "URL" and "Name" are entered, the report feature will use them instead of the main URL and name fields.

See also:
- Report Overview / Templates
- Datafields for bookmarks
- Datafields for folder
- General datafields/options
- Example: Simple URL list in HTML format
- Example: Export to CSV format

2.2.22 Troubleshooting

The following options can be used if a downloaded page cannot be displayed correctly:

- **Ignore document CSS definitions**
  If this option is enabled, all CSS definitions in the downloaded page are ignored.

- **Ignore document text color**
  If this option is enabled, all text colors in the downloaded page are ignored.

- **Ignore document background color**
  If this option is enabled, all background colors in the downloaded page are ignored.

- **Don't ignore NOSCRIPT parts**
  By default WebSite-Watcher doesn't support Javascript and related NOSCRIPT parts. There are often errors in NOSCRIPT parts because they are mostly never tested. If a page can only be displayed with Javascript, you have to use the check technology Browser.

- **Don't optimize for offline browsing**
  You should only enable this option if WebSite-Watcher has problems to optimize a page for offline browsing!

Alternatively to these options you can also try to use the check technologies Browser or Screenshot.

See also:
- A page cannot be displayed

2.2.23 HTML

This section contains options to convert the page into a simple HTML version and to extract parameters from HTML tags which are not readable by default. This makes it easy to monitor content

that is only available in parameters of HTML tags (for example the title parameter of the IMG tag).

- **Convert to simple HTML version**
  This option converts the page into a very simple version. You can use that option if the checked page cannot be displayed or if opening the bookmark causes a crash.

- **Extract DATA parameter from HTML tags**
  Extracts the text content from various HTML tag parameters and inserts it into the page. For example text content from ALT, TITLE or DATA-CONTENT parameters. This lets you monitor content that you typically see after moving the mouse over certain elements.

- **Extract TITLE parameter from HTML tags**
  Extracts the text content from the ALT and TITLE parameters (from any HTML tag) and inserts it into the page.

- **Extract image title**
  Extracts the text content from the Image ALT and TITLE parameters and inserts it into the page.

- **Extract image addresses**
  Extracts the URL of the image tag and inserts it into the page.

- **Extract links addresses**
  Extracts the URL of a link and inserts it into the page.

2.2.24 Misc.

This section contains various configurations that do not fit into other sections.

- **Filename prefix for download manager**
  If you download a file via the download manager (either via RSS feed enclosures or via a Plugin), you can enter a prefix that the downloaded filename should have. This field is optional, the max length is limited to 10 characters.

2.2.25 Tweaks

The Tweaks section allows you to fine tune bookmark properties, to enable special features or to use special workarounds. The Tweaks section uses the format of ini files but without sections (names between "[" and "]").

- **HttpAcceptLanguage=**
  Lets you configure the Accept-Language HTTP header. For more information about that header search the web for "HTTP Accept-Language".
  Example: HttpAcceptLanguage=de

2.2.26 Note / Info fields

For each bookmark you can enter a note and 3 additional info fields.

These fields can be displayed as separate columns in the bookmark list. You can also quickly change these fields when you double click onto the note image 🟨 in the bookmark list.

2.3 Folder properties

2.3.1 Folder properties

General

Bookmark name and user defined icon.

Options

- **Delete bookmarks automatically**
  You can use this feature if you want to monitor bookmarks only for a given time. Optionally you can exclude bookmarks with flags 🚩 from automatic deletion.

Assign bookmark properties

In that section you can define, if and when specific properties should be assigned to bookmarks in the selected folder.

2.4 Virtual folder

2.4.1 Virtual folder properties

Virtual folders contain only bookmarks that meet certain criteria which can be defined in the virtual folder dialog.

Process bookmarks in folders

For each Virtual Folder you can select the folders that should be processed. Default is to process all folders, alternatively you can only select specific folders.

Rules

To get a bookmark listed in a Virtual Folder, entered rules must be met.

1. **Keywords**
   A bookmark is listed in a virtual folder, if an update has found and at least one of the entered keywords is available within new/changed content. If keywords are found, then it's possible to

also run additional actions, for example to send an e-mail or to archive a page permanently to Local Website Archive.

Beside searching in new/changed content, WebSite-Watcher also searches keywords in the following elements: bookmark name, bookmark URL, in the TITLE tag of the page and in the META tags keywords and description.

WebSite-Watcher only searches for words/phrases, wildcards and regular expressions are not supported.

2. **Advanced Rules**
   A bookmark is listed in a virtual folder if the configured rules are met. If only advanced rules are configured (without any keywords defined), then it's not possible to perform additional actions from the "Actions" tab.

3. **Combination: Keywords AND Advanced Rules**
   If keywords AND advanced rules are entered, then both criteria must be met to get a bookmark listed.

If you add or change advanced rules, then the changes are applied immediately and the virtual folder is re-filled with matching bookmarks. Changes in the keywords configuration cannot be applied immediately and are only processed during a bookmark check. To process changes in the keywords configuration immediately, right click a virtual folder and run the command "Re-process virtual folders".

Actions

If entered keywords are found within new/changed content, Website-Watcher is able to run additional actions similar to the Actions available in the bookmark properties. Find a full description of these actions in topic Bookmark Actions.

Part

III
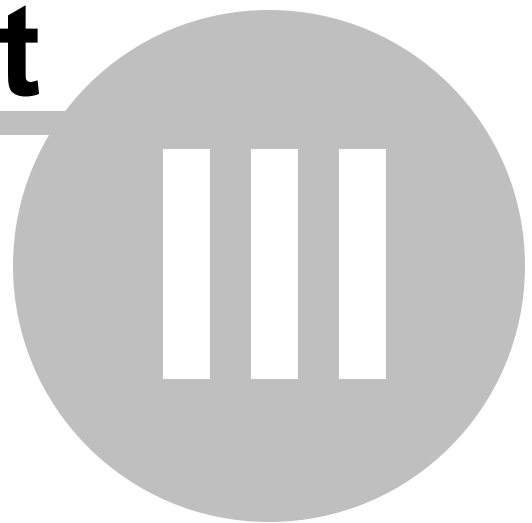
3   Scripting language

3.1   Scripting language (Overview)

The built-in scripting language provides an easy method to automate several tasks in WebSite-Watcher.

The scripting language is line oriented, that means you have to put each command into a single line. See also chapter Syntax overview.

Edit scripts

WebSite-Watcher comes with a built-in script editor that can be started in the "Script" menu.

To quickly open a script in the editor, simply hold the Shift key pressed while selecting a script from the "Script" menu.

Special scripts

Special scripts are executed automatically at at appropriate times, for example the "Autostart" script when WebSite-Watcher is launched. You can find and edit these special scripts in the menu "Edit special script". All other scripts must be started manually via the "Script" menu or the command line.

- **Autostart script**
  This script is executed when WebSite-Watcher is started.

- **Scripts for toolbar buttons**
  WebSite-Watcher can optionally display three toolbar buttons to execute special scripts. Additionally you can assign shortcuts to these scripts.

- **Scripts executed after each check run**
  WebSite-Watcher can optionally execute a script after each check run. Not all commands are allowed in those scripts, the script editor will give you a notification if you are using disallowed commands.

3.2   Example 1: Check bookmarks

The following example checks the bookmarks in three specified folders. All defined bookmark actions are temporarily disabled during the bookmark check.

At the beginning, WebSite-Watcher asks if you want to run the script.

```
// Ask if the script should be executed
QuestionBox "Run the script ?" /ContinueOnYes /beep

// disable bookmark actions
SetBookmarkActionSendMail /off
SetBookmarkActionOpenWebsite /off
SetBookmarkActionRunPrg /off
SetBookmarkActionLWA /off
SetBookmarkActionExport /off

// Check the following folders
CheckFolder "Tools\Internet"
CheckFolder "Blogs" /subfolder
CheckFolder "Magazines"
```

3.3   Example 2: Simulate AutoWatch

The following example simulates the AutoWatch feature for all Hotsites from two different bookmark files. The delay between checks is one minute.

The script uses the time settings that have been defined for manual checks in the bookmark properties.

The script can be aborted with the "Abort" button in the main toolbar. Defined bookmark actions will be executed (for example "Send e-mail") because they have not been disabled in the script.

```
// Check all Hotsites from bookmark file tools.wsw
OpenBookmarkFile "f:\wswatch\tools.wsw"
CheckBookmarks /hotsites

// Check all Hotsites from bookmark file magazines.wsw
OpenBookmarkFile "{bookmarkdir}magazines.wsw"
CheckBookmarks /hotsites

// Wait 60 seconds, then restart the script
WaitDelay 60
RestartScript
```

3.4   Example 3: Create and send a report

The following example checks all bookmarks. Afterwards it creates a report and opens it in the internal browser.

Optionally you can send it to two different e-mail addresses, the script will ask before sending an e-mail.

```
// Ask if the script should be executed
QuestionBox "Run the script ?" /ContinueOnYes /beep

// Check all bookmarks
CheckBookmarks /all

// Create a report and open it in the internal browser
CreateReport /template="{programdir}examples\report_template_changes.htm" (+)
             /output="c:\report.htm" (+)
             /bookmarks=changed
OpenWithInternalBrowser "c:\report.htm"

// Ask if an e-mail should be sent
// If the user does not abort the script, then send the e-mail
QuestionBox "Send the report by e-mail?" /ContinueOnYes /beep
SendMail /recipient="support@domain.com,info@domain2.com" (+)
         /subject="WSW-Report" (+)
         /body="Attached a WSW report" (+)
         /attachfile="c:\report.htm"
MessageBox "Done." /beep
```

3.5   Example 4: Run CloudSync

The following example runs CloudSync, but not more often than every 12 hours.

To execute this script after an AutoWatch check run, insert the following script into the special script "autowatch_after.wws" (Main menu + Script + Edit special scripts + autowatch_after.wws).

```
//Run CloudSync every 12 hours

//Load last sync date
$Date = FileToString("c:\myfiles\CloudSync.dat")

//Calc difference in hours
$Diff = HoursAgo($Date)
if $Diff < 12 then goto End

  CloudSync

  //Save last sync date
  $Now = Now
  StringToFile("c:\myfiles\CloudSync.dat", $Now)

label End
```

To run CloudSync only when the database "sync.wsw" is loaded, you can extend the script as follows:

```
//Run CloudSync every 12 hours (only if database sync.wsw is opened)
$DB = GetBookmarkFilename
$DB = Lowercase($DB)
if $DB <> "sync.wsw" then goto End

  //Load last sync date
```

```
$Date = FileToString("c:\myfiles\CloudSync.dat")

//Calc difference in hours
$Diff = HoursAgo($Date)
if $Diff < 12 then goto End

CloudSync

//Save last sync date
$Now = Now
StringToFile("c:\myfiles\CloudSync.dat", $Now)

label End
```

3.6   Advanced Examples / Code snippets

1. Check all bookmarks and display a message box with the number of changed bookmarks

```
CheckBookmarks /all
$Updated = GetUpdatedBookmarkCount
MessageBox "Bookmarks changed: {$Updated}" /beep
```

2. Check all bookmarks. If updates are available, then create a report with a dynamic/unique filename and send it by e-mail

```
CheckBookmarks /all
if GetUpdatedBookmarkCount=0 then goto NoUpdates

  $filename = GetUniqueID + ".htm"
  CreateReport /template="{programdir}examples\report_template_changes.htm" /output="
  SendMail /recipient="support@domain.com,info@domain2.com" /subject="WSW-Report" /bo

Label NoUpdates
```

3. Display a QuestionBox and perform different actions dependent from the button that has been clicked

```
QuestionBox "Click yes or no"
if GetQuestionBoxResult="yes" then goto yes_has_been_clicked
if GetQuestionBoxResult="no" then goto no_has_been_clicked

label yes_has_been_clicked
  // insert commands here
  goto continue

label no_has_been_clicked
  // insert commands here

label continue
```

4. Loop with 10 iterations

```
$i=1
Label BeginLoop1
  if $i > 10 then goto EndLoop1

  // commands, for example a Messagebox which displays the iteration number
  MessageBox "Iteration #: {$i}"

  $i = $i + 1
  goto BeginLoop1
Label EndLoop1
```

5. Open a bookmark file if it is not currently opened

```
// Open the bookmark file "c:\wsw\bm.wsw" if it is not the opened bookmark file
$s1 = "{bookmarkdir}" + GetBookmarkFilename
$s1 = Lowercase($s1)
if $s1 <> "c:\wsw\bm.wsw" then OpenBookmarkFile "c:\wsw\bm.wsw"
```

6. Check bookmarks every full hour

```
Label myStart

$sHH = GetDateTime("hh")
$nHH = StrToInt($sHH)
$nHH = $nHH + 1
if $nHH > 23 then $nHH = 0

WaitUntilTime "{$nHH}:00"
CheckBookmarks /all

goto myStart
```

Part IV

4   Plugin System

4.1   Overview

The Plugin system provides a way in which the functionality of bookmark checks can be changed or enhanced by writing small scripts.

WebSite-Watcher contains a powerful Integrated Development Environment (IDE) to write, debug and test these plugins.

A plugin script must contain one or more Event functions. During a bookmark check, WebSite-Watcher calls these event functions (if available) at appropriate times. That allows you, for instance, to extract a specific price from a web page, compare it with the price from the previous version and only alert an update if the price is lower than USD 100 and lower than the previous price.

WebSite-Watcher supports Private Plugins (which are created and saved per bookmark) and Shared Plugins (which are saved in separate files and can be used by more than one bookmark).

4.2   Shared/Private Plugins

Plugin types

WebSite-Watcher supports two types of plugins:

- **Private Plugins**
  Private Plugins are created and stored per bookmark and cannot be shared between bookmarks.

- **Shared Plugins**
  Shared Plugins are stored in separate files and can be used by multiple bookmarks.

Both plugin types use the same programming language and have the same set of features, only the way how they are stored is different.

Identification of Shared Plugins

While Private Plugins are stored per bookmark, Shared Plugins require an identification mechanism to test if they are compatible with specific bookmarks.

For each Shared Plugin, unique identification strings of the HTML source must be entered to make them compatible with specific pages. These identification strings can be entered in the "Identification" tab of Plugin IDE.

Automatic suggestion of Shared Plugins

During a bookmark check, WebSite-Watcher tests (based on the identification strings) if a Shared Plugin is compatible with a page. If a compatible plugin was found and you open the page in the internal browser, the browser information bar is displayed.
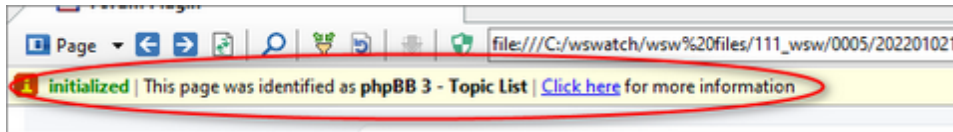
Clicking onto that information bar opens the list of compatible plugins.

4.3 Event functions

A plugin script must contain one or more event functions that are called by WebSite-Watcher at suitable times. All Event functions start with "Wsw_"

- Wsw_SelectPlugin

  This function is called when a user selects a plugin. This function can check whether all the required conditions are met (e.g. the existence of an external converter) and cancel the selection if necessary.

- Wsw_BeforeCheck

  This function is called before a bookmark check is started. You can use this function to change some bookmark properties before the check is started.

- Wsw_MergePages

  This function is called during a bookmark check and offers a way to merge different pages to a single page.

- Wsw_PreProcessPage

  This function is called immediately after downloading a page and before any other processing begins. You can use this function to change the page source or even convert it to another format.

- Wsw_CompareVersions

  This function is called to compare the web version of a page with the locally saved version. You can compare the page content (or parts of it) and decide when a bookmark should alert an update, e.g. if a certain price in the web version is lower than in the locally saved version.

- Wsw_CheckKeywords

  This function is called when WebSite-Watcher searches for keywords. It can be used for an advanced search for keywords with additional logic.

- Wsw_ActionsOnUpdate

  This function is called to perform actions after a detected update, after initialization or after an error.

- Wsw_AfterCheck

  This function is called after a bookmark has been checked.

4.4 Event functions

4.4.1 Wsw_SelectPlugin

```
Sub Wsw_SelectPlugin(ByRef $bCancelSelect)
```

This function is called when a user selects a Plugin. This function can test if all conditions are met (for example the existence of an external converter) and cancel the selection if required.

Parameters

- *$bCancelSelect* ... Boolean. Default value is False. Set it to True if the Plugin selection should be canceled.

Example:

```
Sub Wsw_SelectPlugin(ByRef $bCancelSelect)

    If Not FileExists("c:\converter\doc2txt.exe") Then
        MsgBox("The file ""doc2txt.exe"" was Not found!")
        $bCancelSelect = True
    End If
End Sub
```

4.4.2 Wsw_BeforeCheck

```
Sub Wsw_BeforeCheck()
```

This function is called before a bookmark check is started. You can use this function to change some bookmark properties before the check is started.

The bookmark properties are not changed permanently, only temporarily for the current bookmark check. You will not see these changes in the bookmark properties.

Example:

```
Sub Wsw_BeforeCheck()

    ' Add entry to the bookmark check log
    Bookmark_AddLog("Wsw_BeforeCheck> Adjust pagetype and checkmethod")

    ' Change bookmark properties
    Bookmark_SetProperty("pagetype", "html")
    Bookmark_SetProperty("checkmethod", "automatic")
```

```
End Sub
```

4.4.3   Wsw_MergePages

```
Sub Wsw_MergePages($sMem, $nPageNumber, $sUrl, ByRef $sNewUrl, ByRef $sNewPostData, B
```

This function is called during a bookmark check and offers a way to merge different pages to a single page.

Parameters

- *$sMem* … String expression. HTML source code of the last checked page. All links are available as absolute URLs.
- *$nPageNumber* … Integer value. Amount of checked pages, value starts with 1. Helps to limit the number of merged pages. WebSite-Watcher can merge maximal 10 pages.
- *$sUrl* … String expression. URL of the last checked page.
- *$sNewUrl* … String expression. Return value, that contains the new URL that should be checked and merged with the previously checked page. Initial value is an empty string. If an empty string is returned, then the merge operations will be terminated.
- *$sNewPostData* … String expression. Return value, that contains PostData if the page should be checked via the POST command. If that parameter is empty, then the page will be checked with the GET command. Default value is an empty string.
- *$bChangeBookmarkUrl* … Boolean value. Return value, default value is FALSE. Set this variable to TRUE if the bookmark URL should replaced with the returned parameter $sNewUrl. This is only useful if you monitor a forum thread and always want to read the latest posts/pages, then the bookmark URL is always updated to the last checked page. If you monitor a search result, then simply don't assign a value to that parameter.
- *$sStatusMessage* … String expression. Status message that will be displayed in the status column. Default value is an empty string.
- *$iStatusCode* … Integer value. Returns the success status of that function.

<u>Valid values of $iStatusCode:</u>

0 …   OK, event function could be processed successfully.
        Default value, must not be assigned manually.

1 …   Error.
        Should be returned if errors are detected, the bookmark check will be aborted.

Example:

```
Sub Wsw_MergePages($sMem, $nPageNumber, $sUrl, ByRef $sNewUrl, ByRef $sNewPostData, B

   Dim $nPageParamNumber, $sPageParam, $sNewPageParam

   ' Limit the number of merged pages
   If $nPageNumber > 3 Then
      Return
   End If
```

```
    ' Extract page number
    $sPageParam = GetFirstRegexMatch($sUrl, "\&start=\d+")
    If $sPageParam = "" Then
        ' first page, continue with page 2 (solved with start=10)
        $sNewPageParam = "&start=10"
        $sUrl = $sUrl + $sNewPageParam
    Else
        ' increase page number
        $nPageParamNumber = CInt(ExtractDigits($sPageParam)) + 10
        $sNewPageParam = "&start=" + CStr($nPageParamNumber)
        $sUrl = Replace($sUrl, $sPageParam, $sNewPageParam)
    End If

    ' check (only) via Page-Param, if the next page is already available in the page s
    If Pos($sNewPageParam, $sMem) > 0 Then
        ' the next page exists
        $sNewUrl = $sUrl
        $sStatusMessage = "Merge pages via FOLLOW-Plugin"
    End If

End Sub
```

See also examples Merge search result pages and Merge pages in forum thread.

4.4.4   Wsw_PreProcessPage

```
Sub Wsw_PreprocessPage(ByRef $sMem, ByRef $sStatusMessage, ByRef $iStatusCode)
```

This function is called immediately after a page has been downloaded and before any other processing starts. This function can be used to modify the page source or even to convert it into another format.

Parameters

- *$sMem* ... String expression. Memory string of the page source.
- *$sStatusMessage* ... String expression. The assigned text will be displayed in the status column of the bookmark.
- *$iStatusCode* ... Integer value. Returns the success status of that function.

Valid values of $iStatusCode:

0 ...   OK.
         Default value, must not be assigned manually.

2 ...   Error.
         If 2 is returned, then the check of that bookmark will be aborted and the text assigned to $sStatusMessage will be displayed in the status column.

Example:
```
Sub Wsw_PreprocessPage(ByRef $sMem, ByRef $sStatusMessage, ByRef $iStatusCode)

    Dim $nStart, $nLen
```

```
    ' Exit if no session ids are available (small performance improvement)
    If InStr($sMem, "sid=") = 0 Then
        Return
    End If

    ' Remove all session ids found in the page source
    While FindRegex($sMem, "&(amp;)*sid=[a-z,0-9]*", $nStart, $nLen)
        Delete($sMem, $nStart, $nLen)
    End While
End Sub
```

4.4.5 Wsw_CompareVersions

```
Sub Wsw_CompareVersions(ByRef $sMemWeb, ByRef $sMemLocal, ByRef $sStatusMessage, ByRe
```

This function is called to compare the web version of a page with the locally saved version. You can filter the content and decide when a bookmark should alert an update, for example when a specific price in the web version is lower than in the locally saved version. The parameter $iStatusCode is a numeric value and indicates if a page has been changed or if an error has occurred.

Parameters

- *$sMemWeb* ... String expression. Page source code of the web version.
- *$sMemLocal* ... String expression. Page source code of the locally saved version.
- *$sStatusMessage* ... String expression. Status message that will be displayed in the status column. Default value is an empty string.
- *$iStatusCode* ... Integer value. Indicates if a page has been changed or if an error has occurred. Default value is 0.

Valid values of $iStatusCode:

0 ...   OK, page is unchanged.
        Default value, must not be assigned manually.

1 ...   OK, page has been changed.
        Bookmark will be marked as updated.

2 ...   Error.
        If 2 is returned, then the check of that bookmark will be aborted with an error.

Examples:

Simple example to see if the filtered page content has been changed.

```
Sub Wsw_CompareVersions(ByRef $sMemWeb, ByRef $sMemLocal, ByRef $sStatusMessage, ByRe

    ' Apply the defined filter definitions
    $sMemWeb = Bookmark_ApplyFilter($sMemWeb)
    $sMemLocal = Bookmark_ApplyFilter($sMemLocal)

    ' Check if the page has been changed by comparing the filtered content
```

```
    If $sMemWeb <> $sMemLocal Then
        $iStatusCode = 1 ' updates detected
    Else
        $iStatusCode = 0 ' no updates available
    End If

End Sub
```

More advanced example to detect price changes and alert if a price is lower than a pre-defined value. This price extraction is dependent from the page, you can use the code below as a starting point.

```
Sub ExtractPrice($sMem, ByRef $nPrice)

    Dim $p, $nLen

    $nPrice = -1
    $sMem = DeleteHtmlTags($sMem)
    ' Extract price without decimal places
    If FindRegex($sMem, "Price:\s*EUR\s*\d+,", $p, $nLen) Then
        $sMem = ExtractDigits(Copy($sMem, $p, $nLen))
        $nPrice = CInt($sMem)
    End If
End Sub

'*******************************************************************************

Sub Wsw_CompareVersions(ByRef $sMemWeb, ByRef $sMemLocal, ByRef $sStatusMessage, ByRe

    Dim $nPriceNew, $nPriceOld, $nPriceRef

    ' Define reference price (100 EUR)
    $nPriceRef = 100

    ' Extract price from new/local version
    ExtractPrice($sMemWeb, $nPriceNew)
    ExtractPrice($sMemLocal, $nPriceOld) ' only needed to speed up several WSW routine

    ' Return only price - speed up several WSW routines, eg. "Test filter" dialog or "
    $sMemWeb = CStr($nPriceNew)
    $sMemLocal = CStr($nPriceOld)

    If $nPriceNew = -1 Then
        $iStatusCode = 2
        $sStatusMessage = "Error extracting price"
    ElseIf ($nPriceNew <> $nPriceOld) And ($nPriceNew <= $nPriceRef) Then
        $iStatusCode = 1
        $sStatusMessage = "Price changed and lower than EUR " + CStr($nPriceNew)
    ElseIf ($nPriceNew <> $nPriceOld) And ($nPriceNew > $nPriceRef) Then
        $iStatusCode = 0
        $sStatusMessage = "Price too high"
    Else
        $iStatusCode = 0
        $sStatusMessage = "Price unchanged"
    End If
End Sub
```

4.4.6 Wsw_CheckKeywords

```
Sub Wsw_CheckKeywords($sMemChanges, ByRef $sStatusMessage, ByRef $iKeywordFound)
```

This function is called when WebSite-Watcher looks for keywords. It can be used for advanced keyword processing with logical operators. The parameter $sMemChanges contains the new and changed text without HTML tags, carriage return linefeeds and duplicate blanks. $iKeywordFound is a numeric value and indicates whether keywords have been found.

If keywords are also entered in the bookmark properties (Update-On-Keywords), then this function is not called if one of the entered keywords has already been found. If no keywords are entered or none of the entered keywords were found (Update-On-Keywords), then WebSite-Watcher will call this function.

Parameters

- *$sMemChanges* ... String expression. New/changed text of the web page (no HTML tags, no CRLFs, no duplicate blanks).
- *$sStatusMessage* ... String expression. Status message that will be displayed in the status column. Default value is an empty string.
- *$iKeywordFound* ... Integer value. Indicates if keywords were found. Default value is 0.

Valid values of $iKeywordFound:

0 ...   No keywords found.
        Default value, must not be assigned manually.

1 ...   Keywords were found.

Example:

```
Sub Wsw_CheckKeywords($sMemChanges, ByRef $sStatusMessage, ByRef $iKeywordFound)

   Dim $s = $sMemChanges

   ' Look for WebSite Watcher but exclude weight watcher
   If KeywordExists($s, "website", True) And KeywordExists($s, "watcher", True) And N
      $iKeywordFound = 1
      Return
   End If

   If KeywordExists($s, "aignesberger", True) And KeywordExists($s, "watcher", True)
      $iKeywordFound = 1
      $sStatusMessage = "Keyword 'aignesberger + watcher' found"
      Return
   End If

End Sub
```

4.4.7 Wsw_ActionsOnUpdate

```
Sub Wsw_ActionsOnUpdate($iUpdateStatus, ByRef $sStatusMessage, ByRef $iStatusCode)
```

This function is called to perform actions after an update, initialization or an error.

This function is only called when the "Alert/Actions on initialization/update/error" feature in the bookmark properties is configured.

This event function is called after all other bookmark actions (which are defined in the bookmark properties) are executed.

Parameters

- *$iUpdateStatus* ... Integer value. Indicates if an initialization, update or error occurred.
- *$sStatusMessage* ... String expression. Status message that will be displayed in the status column. Default value is an empty string.
- *$iStatusCode* ... Integer value. Indicates if actions could be performed or if an error has occurred. Default value is 0.

Valid values of $iUpdateStatus:

1 ...   Bookmark has been initialized

2 ...   Bookmark has been updated

3 ...   Error happened while checking the bookmark

Valid values of $iStatusCode:

0 ...   OK, actions could be performed.
         Default value, must not be assigned manually.

1 ...   Error.
         Should be returned if actions could not be performed, the bookmark check will however not be aborted. The status code is currently only written to the bookmark log file and doesn't affect the bookmark check.

Example:

```
Sub Wsw_ActionsOnUpdate($iUpdateStatus, ByRef $sStatusMessage, ByRef $iStatusCode)

    Dim $sMem

    If $iUpdateStatus = 2 Then
        $sMem = Bookmark_LoadCacheFile(1)
        StringToFile("c:\data\changed.htm", $sMem)
    End If
End Sub
```

4.4.8   Wsw_AfterCheck

```
Sub Wsw_AfterCheck(ByRef $sStatusMessage)
```

This function is called after a bookmark has been checked. For example, you can use this function if you want to write the time of a bookmark check to an external log file.

Parameters

- *$sStatusMessage* ... String expression. Status message that will be displayed in the status column. Default value is an empty string.

Example:

```
Dim $dStart = ""  'global variable that can be used in all event functions


'********************************************************************************


Sub Wsw_BeforeCheck()
    Bookmark_SetProperty("checkmethod", "content")
    $dStart = Now
End Sub


'********************************************************************************


Sub Wsw_AfterCheck(ByRef $sStatusMessage)
    AddStringToFile("c:\wswlog\checktime.txt", "Seconds: " + DateDiff(Now, $dStart, "s
End Sub
```

4.5 How to...

4.5.1 Create a Private Plugin

Private Plugins are created and stored per bookmark and cannot be shared between bookmarks.

1. Open the bookmark properties
2. Select the Advanced tab
3. Select "Plugin" on the left side
4. Click the button "Create Private Plugin"

This will open the Plugin IDE and you have to implement an Event function. You can use the "Insert" menu to paste the framework of an event function into the script editior.

4.5.2 Create a Shared Plugin

Shared Plugins are stored in separate files and can be shared between multiple bookmarks.

1. Open the bookmark properties
2. Select the Advanced tab
3. Select "Plugin" on the left side
4. Click the button "Create Shared Plugin"

This will open the Plugin Editor where you have to enter/implement the following parts:

- Plugin name and description
- Basic Plugin configuration (eg. when the Plugin should be suggested)
- The Plugin identification strings
- The Plugin script (Event functions)

4.6   Examples

4.6.1   Check for specific price

The following example extracts a specified price from a certain page and only alerts an update if the price is lower than a predefined price. This price extraction is dependent from the page, you can use the code below as a starting point.

An alternative to such a plugin might be the regexcmp function that can be used in Ignore filters, Watch filters and the Keywords functionality.

```
Sub ExtractPrice($sMem, ByRef $nPrice)

   Dim $p, $nLen

   $nPrice = -1
   $sMem = DeleteHtmlTags($sMem)
   ' Extract price without decimal places
   If FindRegex($sMem, "Price:\s*EUR\s*\d+,", $p, $nLen) Then
      $sMem = ExtractDigits(Copy($sMem, $p, $nLen))
      $nPrice = CInt($sMem)
   End If
End Sub

'*******************************************************************************

Sub Wsw_CompareVersions(ByRef $sMemWeb, ByRef $sMemLocal, ByRef $sStatusMessage, ByRe

   Dim $nPriceNew, $nPriceOld, $nPriceRef

   ' Define reference price (100 EUR)
   $nPriceRef = 100

   ' Extract price from new/local version
   ExtractPrice($sMemWeb, $nPriceNew)
   ExtractPrice($sMemLocal, $nPriceOld) ' only needed to speed up several WSW routine

   ' Return only price - speed up several WSW routines, eg. "Test filter" dialog or "
   $sMemWeb = CStr($nPriceNew)
   $sMemLocal = CStr($nPriceOld)

   If $nPriceNew = -1 Then
      $iStatusCode = 2
      $sStatusMessage = "Error extracting price"
   ElseIf ($nPriceNew <> $nPriceOld) And ($nPriceNew <= $nPriceRef) Then
      $iStatusCode = 1
      $sStatusMessage = "Price changed and lower than EUR " + CStr($nPriceNew)
   ElseIf ($nPriceNew <> $nPriceOld) And ($nPriceNew > $nPriceRef) Then
```

```
        $iStatusCode = 0
        $sStatusMessage = "Price too high"
    Else
        $iStatusCode = 0
        $sStatusMessage = "Price unchanged"
    End If
End Sub
```

4.6.2 Alert if no update

The following example only alerts an update if there was <u>no</u> change.

```
Sub Wsw_BeforeCheck()

   ' disable optimizations and always perform a download to compare content
   Bookmark_SetProperty("checkmethod", "content")

End Sub

'*******************************************************************************

Sub Wsw_CompareVersions(ByRef $sMemWeb, ByRef $sMemLocal, ByRef $sStatusMessage, ByRe

   ' Apply the defined filter definitions
   $sMemWeb = Bookmark_ApplyFilter($sMemWeb)
   $sMemLocal = Bookmark_ApplyFilter($sMemLocal)

   ' Compare the filtered content
   If $sMemWeb <> $sMemLocal Then
      ' Page is updated -> do NOT mark as changed
      $iStatusCode = 0
   ElseIf $sMemWeb = $sMemLocal Then
      ' Page has not been updated -> mark as changed
      $iStatusCode = 1
   End If

End Sub
```

4.6.3 Find keywords

The following example alerts an update if specific keywords are found within new or changed content.

```
Sub Wsw_CheckKeywords($sMemChanges, ByRef $sStatusMessage, ByRef $iKeywordFound)

   Dim $s = $sMemChanges

   ' Look for WebSite Watcher but exclude weight watcher
   If KeywordExists($s, "website", True) And KeywordExists($s, "watcher", True) And N
      $iKeywordFound = 1
      Return
   End If

   If KeywordExists($s, "aigne", True) And KeywordExists($s, "watcher", True) Then
      $iKeywordFound = 1
      $sStatusMessage = "Keyword 'aignes + watcher' found"
```

```
      Return
   End If

End Sub
```

Please read topic Wsw_CheckKeywords to see under which circumstances this event function is called or not called.

4.6.4 Alert if keywords are NOT found

The Update-On-Keywords feature in the bookmark properties only alerts an update if one of the entered keywords is found within new or changed content.

If you want to get an update notification when a keyword is NOT found in new/changed content, you can use the following plugin.

```
Sub Wsw_CheckKeywords($sMemChanges, ByRef $sStatusMessage, ByRef $iKeywordFound)

   If KeywordExists($sMemChanges, "your keyword", True) Then
      ' keyword found
      $iKeywordFound = 0
      $sStatusMessage = "No alert because keyword found"
   Else
      ' keyword NOT found
      $iKeywordFound = 1
      $sStatusMessage = "Alert because keyword NOT found"
   End If

End Sub
```

For a negative keyword search you should not use any keywords in the Update-On-Keywords section. Please read topic Wsw_CheckKeywords to see under which circumstances this event function is called or not called.

4.6.5 Merge search result pages

This example shows how to merge pages from a search result.

In that example, the page URLs contain a &start= parameter, whereas the first page doesn't have that parameter, following pages have steps of 10 (&start=10, &start=20, etc.)

```
Sub Wsw_MergePages($sMem, $nPageNumber, $sUrl, ByRef $sNewUrl, ByRef $sNewPostData, B

   Dim $nPageParamNumber, $sPageParam, $sNewPageParam

   ' Limit the number of merged pages
   If $nPageNumber > 3 Then
      Return
   End If

   ' Extract page number
   $sPageParam = GetFirstRegexMatch($sUrl, "\&start=\d+")
```

```
    If $sPageParam = "" Then
        ' first page, continue with page 2 (solved with start=10)
        $sNewPageParam = "&start=10"
        $sUrl = $sUrl + $sNewPageParam
    Else
        ' increase page number
        $nPageParamNumber = CInt(ExtractDigits($sPageParam)) + 10
        $sNewPageParam = "&start=" + CStr($nPageParamNumber)
        $sUrl = Replace($sUrl, $sPageParam, $sNewPageParam)
    End If

    ' check (only) via Page-Param, if the next page is already available in the page s
    If Pos($sNewPageParam, $sMem) > 0 Then
        ' the next page exists
        $sNewUrl = $sUrl
        $sStatusMessage = "Merge pages via FOLLOW-Plugin"
    End If

End Sub
```

4.6.6   Merge pages in forum thread

This example shows how to follow to the next page in a forum thread once it is available.

The links to the previous and next pages are typically displayed as:
 First … 13, **14**, 15 … Last

If we are for example on page 14 and a link to page 15 already exists in the page, then this plugin will return the new URL so that WebSite-Watcher can merge the new page with the current page.

Links of the pages have the following format:
   http://domain.com/forum/topic123x-14.html
where 14 is the page number. The URL of the first page doesn't have the page number included, just topic123x.html

Each posting in that forum thread has a posting number that is displayed as **# 12** for posting number 12. We will define a watch filter to monitor only the post numbers, then we will not get false positives if other content is changed, for example forum statistics, a changing date, advertisements, etc.

```
Sub Wsw_BeforeCheck()
    ' Set a watch filter to monitor only posting numbers
    ' so we will only get an update notification if a new posting number appears
    Bookmark_SetProperty("watch_filter", "regex(#\s*\d+)")
    Bookmark_SetProperty("filter_ignore_removed_content", "1")
End Sub

'********************************************************************************

Sub Wsw_MergePages($sMem, $nPageNumber, $sUrl, ByRef $sNewUrl, ByRef $sNewPostData, B

    Dim $nPageParamNumber, $sFilename
```

```
    ' Limit the number of merged pages
    If $nPageNumber > 3 Then
        Return
    End If

    ' Extract page number
    $sFilename = GetFirstRegexMatch($sUrl, "index\d+\.html")
    If $sFilename = "" Then
        $nPageParamNumber = 2
    Else
        $sUrl = Replace($sUrl, $sFilename, "")
        $nPageParamNumber = CInt(ExtractDigits($sFilename)) + 1
    End If
    $sFilename = "index" + CStr($nPageParamNumber) + ".html"

    ' Generate possible next page
    If Right($sUrl, 1) <> "/" Then
        $sUrl = $sUrl + "/"
    End If
    $sUrl = $sUrl + $sFilename

    ' check, if the next page is already available in the page source
    If Pos($sUrl, $sMem) > 0 Then
        ' the next page already exists
        $sStatusMessage = "New page via Merge-Plugin"
        $sNewUrl = $sUrl 'return new URL
        $bChangeBookmarkUrl = True 'change URL in bookmark properties
    End If
End Sub
```

If a forum thread displays the page number via a parameter in the URL (for example **&page=13**), then the following code might be a starting point:

```
Sub Wsw_BeforeCheck()
    ' optional watch filter if each post has a posting number, eg. #14
    Bookmark_SetProperty("watch_filter", "regex(#\s*\d+)")
    Bookmark_SetProperty("filter_ignore_removed_content", "1")
End Sub

'*********************************************************************************

Sub Wsw_MergePages($sMem, $nPageNumber, $sUrl, ByRef $sNewUrl, ByRef $sNewPostData, B

    Dim $nPageParamNumber, $sPageParam

    ' Limit the number of merged pages
    If $nPageNumber > 5 Then
        Return
    End If

    ' Extract page number
    $sPageParam = GetFirstRegexMatch($sUrl, "\&page=\d+")
```

```
   If $sPageParam = "" Then
      ' first page, continue with page 2
      $sUrl = $sUrl + "&page=2"
   Else
      ' increase page number
      $nPageParamNumber = CInt(ExtractDigits($sPageParam)) + 1
      $sUrl = Replace($sUrl, $sPageParam, "&page=" + CStr($nPageParamNumber))
   End If

   ' check, if the next page is already available in the page source
   If Pos($sUrl, $sMem) > 0 Then
      ' the next page already exists
      $sNewUrl = $sUrl
      $bChangeBookmarkUrl = True ' change bookmark URL to latest merged URL
      $sStatusMessage = "New page via FOLLOW-Plugin"
   End If

End Sub
```

4.6.7 Log the bookmark check time to a file

This example calculates the time of a check and writes the number of seconds to a csv log file (c:\wswlogs\log23.txt).

The number of seconds are also written into the additional info field #1 that can be displayed as column in the bookmark list.

Global variables can be defined at the beginning of the plugin.

```
' Global variables that can be used in all event functions
Dim $sLogfile = "c:\wswlogs\log23.txt"
Dim $dStart = ""

'*******************************************************************************

Sub Wsw_BeforeCheck()
   Bookmark_SetProperty("checkmethod", "content")
   $dStart = Now
End Sub

'*******************************************************************************

Sub Wsw_AfterCheck(ByRef $sStatusMessage)

   Dim $sSeconds = DateDiff(Now, $dStart, "s")

   ' Write seconds into additional info field/column
   Bookmark_SetProperty("infofield1", $sSeconds + " sec")

   ' Write URL and seconds into log file
   If $sLogfile <> "" Then
      AddStringToFile($sLogfile, """" + Bookmark_GetProperty("check_url") + """;""" +
   End If
```

```
End Sub
```

4.6.8   Alert only if there is more than 1 change

By default, WebSite-Watcher alerts an update as soon as a change is detected.

If you only want an update notification if there is for example more than one change, you can use the plugin below. You could also increase the value 1, for example to only get an update notification if there are more than 3 changes in the page.

```
Sub Wsw_CompareVersions(ByRef $sMemWeb, ByRef $sMemLocal, ByRef $sStatusMessage, ByRe

   Dim $nNew, $nChg, $nDel

   ' Apply entered filter definitions
   $sMemWeb = Bookmark_ApplyFilter($sMemWeb)
   $sMemLocal = Bookmark_ApplyFilter($sMemLocal)

   ' Calculate the number of new/changed/deleted content
   GetComparisonInfo($sMemWeb, $sMemLocal, 1, $nNew, $nChg, $nDel)

   ' Alert only if there is more than 1 change
   If ($nNew + $nChg > 1) Then
      $iStatusCode = 1
   End If

End Sub
```

4.6.9   Extract and show PDF links

Extracts the URLs of all links to PDF files and displays them in the web page.

```
Sub Wsw_PreprocessPage(ByRef $sMem, ByRef $sStatusMessage, ByRef $iStatusCode)

   Dim $sl, $i, $sItem

   $sl = GetAllWildcardMatches($sMem, "<a *>")
   For $i = 0 To StringList_ItemCount($sl) - 1
      $sItem = StringList_GetItem($sl, $i)
      If Pos(".pdf", $sItem) > 0 Then
         $sMem = Replace($sMem, $sItem, $sItem + HtmlGetParam($sItem, "href"))
      End If
   Next

End Sub
```

4.6.10   Write check status to log file

This example writes the date, URL and status of the last check to a log file in CSV format.

```
Sub Wsw_AfterCheck(ByRef $sStatusMessage)

   Dim $sUrl, $sDateTime, $sStatus
```

```
    $sUrl = Bookmark_GetProperty("check_url")
    $sStatus = Bookmark_GetProperty("status")
    $sDateTime = Now

    AddStringToFile("c:\logs\log.csv", $sDateTime + ";" + $sUrl + ";" + $sStatus)

End Sub
```

4.6.11 Error if not all ignore filters found a match

This example gives an error if not all ignore filter definitions found a match.

```
Sub Wsw_CompareVersions(ByRef $sMemWeb, ByRef $sMemLocal, ByRef $sStatusMessage, ByRe

    'Apply the defined filter definitions
    $sMemWeb = Bookmark_ApplyFilter($sMemWeb)
    $sMemLocal = Bookmark_ApplyFilter($sMemLocal)

    Dim $sUnused = Bookmark_GetProperty("ignore_filter_unused")

    If Len($sUnused) > 0 Then
        $iStatusCode = 2 'error
        $sStatusMessage = "Filters broken"
        Bookmark_AddLog("Unused filters: " + $sUnused)
    ElseIf $sMemWeb <> $sMemLocal Then
        $iStatusCode = 1 'updates detected
    Else
        $iStatusCode = 0 'no updates available
    End If
End Sub
```