

ETScript 3 Readme

By Nick Thissen aka Flippy

Contents

1. Summary	2
2. Features.....	2
2.1. Standard features.....	2
2.1.1. Find & Replace (menu <i>Edit</i>)	3
2.1.2. Go to (menu <i>Edit</i>)	3
2.2. Extended features	3
2.2.1. Recent files list (menu <i>File</i>)	3
2.2.2. Templates (menu <i>File</i>).....	3
2.2.3. Auto-Indentation & Block indentation	4
2.3. Multiple Document Interface (MDI).....	4
2.4. Special features	5
2.4.1. Intellisense & AutoComplete	5
2.4.2. AutoBackups.....	6
2.4.3. Line numbering.....	6
2.5. Tools	6
2.5.1. The Scriptcommand panel, Error list and Quick Find (menu <i>View</i>).....	6
2.5.2. Bookmarks (menu <i>Tools</i>).....	7
2.5.3. Accum count (menu <i>Tools</i>).....	7
2.5.4. Bracket count (menu <i>Tools</i>)	7
2.5.5. Comment selection & Uncomment selection (menu <i>Tools</i>)	7
2.5.6. AutoBackup Manager.....	8
3. Options	9
3.1. General.....	9
3.2. Text editor	9
3.3. AutoBackup	10
3.4. Appearance	10
4. Contact	10
5. Thanks	11

1. Summary

ETScript is a script editor designed specifically for Wolfenstein: Enemy Territory map script files. These script files are usually written in simple text editors like Notepad, which offer very little extra support. ETScript is different. While it hosts *all* useful features popular text editors like Notepad have, it also hosts a ton of features designed specifically for W:ET script files.

You will probably find ETScript superior over Notepad, simply because it looks and handles identical to Notepad, plus a ton of extra features. These features are mostly aimed toward W:ET script files and help you to write your scripts quicker, easier and with less errors.

2. Features

2.1. Standard features

The standard features are the features also present in most other simple text-editors, like Notepad. A few of them however work a little differently; the differences will be explained in another subsection.

The most well-known standard features are of course the *New*, *Open*, *Save* and *Save As* functions. These functions are found in the *File* menu, or in the toolbar. Other well-known functions are the *Undo/Redo*, *Cut*, *Copy*, *Paste* and *Select All* functions, and the *Find & Replace* and *Go to* functions. These are all found in the *Edit* menu as usual, or in the toolbar.

Name / Location	Description
File – New	Opens a blank script, possibly with template (see ...)
File – Open	Open a script file of choice.
File – Save	Saves the currently active script to the last saved location.
File – Save As	Saves the currently active script to a location of choice.
Edit – Undo	Undoes the last action.
Edit – Redo	Redoes the last undone action.
Edit – Cut	Cuts the selected text onto the <i>Clipboard</i> .
Edit – Copy	Copies the selected text onto the <i>Clipboard</i> .
Edit – Paste	Pastes the text on the <i>Clipboard</i> into the script.
Edit – Select All	Selects the entire script.
Edit – Find & Replace	Opens the <i>File & Replace</i> dialog.
Edit – Go to	Opens the <i>Go to</i> dialog.

2.1.1. Find & Replace (menu *Edit*)

The *Find & Replace* dialog is not exactly the same as the usual dialog. Often, *Find* and *Replace* dialogs are separated. In ETScript 3, they are located in the same window. Use the *Find* and *Replace* tabs at the top to switch between the two.

For your convenience, ETScript 3 stores up to 50 keywords you searched for in the dropdown list.

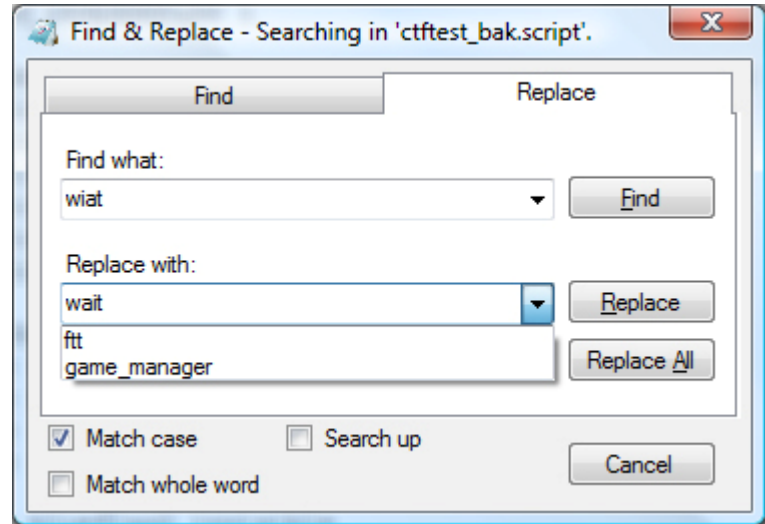


Figure 1 - Find & Replace dialog.

2.1.2. Go to (menu *Edit*)

The *Go to* dialog has also changed. Instead of merely allowing you to jump to any line number in the script, you can now also use the *Go to scriptname* tab to jump to an existing *scriptblock* name.

2.2. Extended features

The extended features are basically features that are extended from the standard features.

Name / Location	Description
File – Recent files...	Shows a list of the 5 last opened scripts for easy acces.
File – Template...	Allows you to save the currently active script as the <i>template</i> script. Also allows you to revert to the default template.
Edit – Insert date	Inserts the current date in the format specified in the <i>Options</i> .
Edit – Insert mapname	Inserts the mapname of the current script.
Tools – Indent selection	If no selection is made: Inserts a tab space. If a selection is made: Inserts a tab space in front of every selected line.
Tools – Unindent selection	If a selection is made: Removes a tab space in front of every selected line.
Tools – Convert to lowercase	Converts the selected text to lowercase.
View logfile	Shows the ETScript logfile, listing every error / info message that has occurred.

2.2.1. Recent files list (menu *File*)

The *Recent files* list simply shows you a list of the last five scripts you have opened. If you open a script that is already present in the list, it will not be added a second time.
You can clear the list by clicking the *Clear* item.

2.2.2. Templates (menu *File*)

A *template* is a default piece of script which can be opened automatically (instead of a blank script) when you choose the *File – New* option. Whether you want to open this template or open a blank script can be configured in the *Options*.

You can save the currently active script to be your template by selecting the *File – Template – Save as Template* item.

You can revert back to the default template by selecting the *Revert to default* item.

2.2.3. Auto-Indentation & Block indentation

The *Auto-Indent* feature keeps you from having to ‘re-tab’ every new line. While writing scripts, it is common to create an indent of one tab before each level in the script. If *Auto-Indentation* is enabled, ETScript will automatically add the same number of tabs as the preceding line to a new line. It essentially indents the new line automatically.

The *Block-Indent* feature is a little feature that will automatically create the usual ‘bracket-block’ when you type an opening bracket:

{	// opening bracket
	// a tab here
}	// closing bracket

It will also automatically place the typing *caret* (the blinking line that tells you where you are typing) at the end of the tab in the middle line, so you can start typing immediately.

If you would ever need to type an opening bracket ({) without the need of this ‘bracket-block’, you can use the opening square bracket ([) instead, which will type a { if enabled in the *Options*.

2.3. Multiple Document Interface (MDI)

ETScript 3 is now a Multiple Document Interface (MDI) script-editor. This means that you are able to open many scripts simultaneously. In a simple text-editor like Notepad you can usually only open one script. If you need to open another, the first script is closed. In this MDI script-editor you can open many scripts at the same time, and switch between them quickly. This way, you can easily use another script as a reference.

The MDI environment is all automated. Opening a new (blank) or existing script will automatically create a new MDI *child document*, which is simply a new window that lives inside the main window. You are able to move around the child documents separately, and you can also use the four *Alignment* functions to align them in a specific order.

The *Alignment* functions are located in the *Window* menu, under the *Alignment* submenu.

Name	Description
Alignment – Cascade	Cascades every child document downwards in the order they were opened.
Alignment – Stack	Stacks each child document neatly on top of eachother.
Alignment – Side by side	Aligns each child document neatly side-by-side.
Alignment - Squared	Aligns the child documents into a square (can only be used when 3 or 4 scripts are open).
Window – Minimize all	Minimizes every child document.
Window – Close all	Closes every child document (a warning can be enabled/disabled in the <i>Options</i>).

In addition, the *Window* menu will also host a list of every open child document.

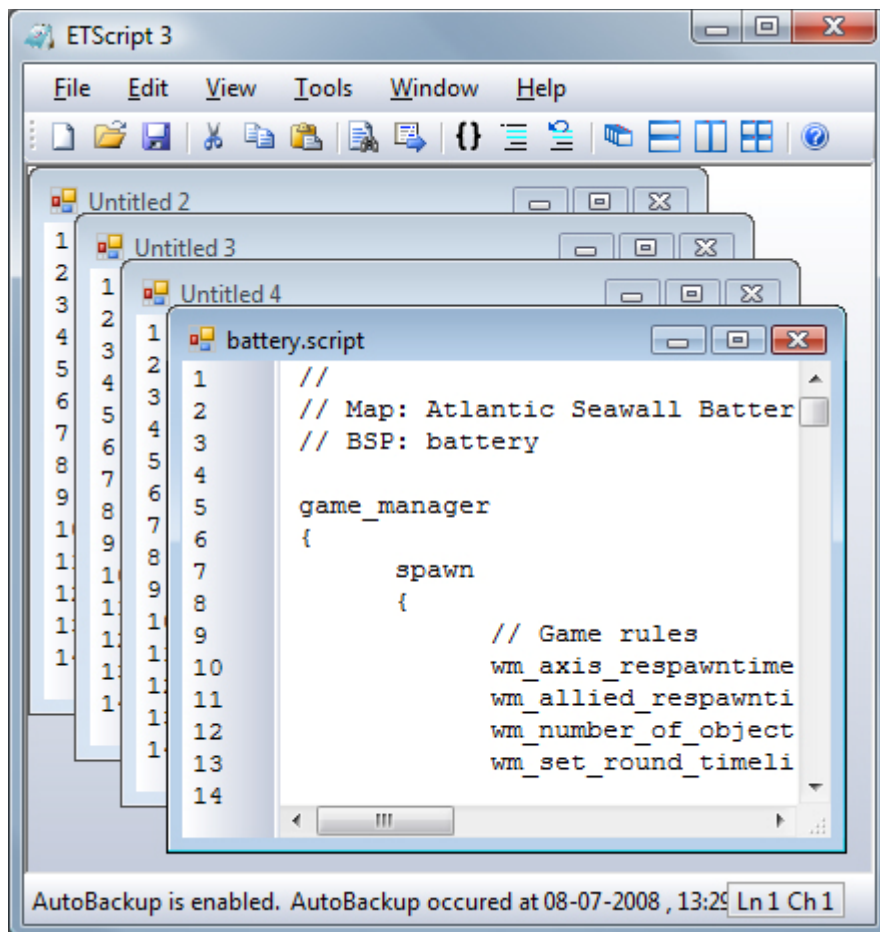


Figure 2 - Screenshot of ETScript 3 showing 4 MDI child documents in a cascaded alignment.

2.4. Special features

The special features are a few important and very handy features and functions that allow you to write script files much easier and with much less errors.

2.4.1. Intellisense & AutoComplete

The *Intellisense* and *AutoComplete* features are the most important features of ETScript 3.

As soon as you start to write some text in your script, ETScript 3 starts checking what you are typing. If you start typing some word that might be a valid W:ET script command, ETScript 3 picks up on this and displays a list of every possible command you might be typing. You can use the arrow keys to scroll through the list. Once a command in the list has been selected, a label beneath the list will show exactly how to use the command and show a short description.

Pressing Enter will AutoComplete the selected word; you don't even have to finish typing it!

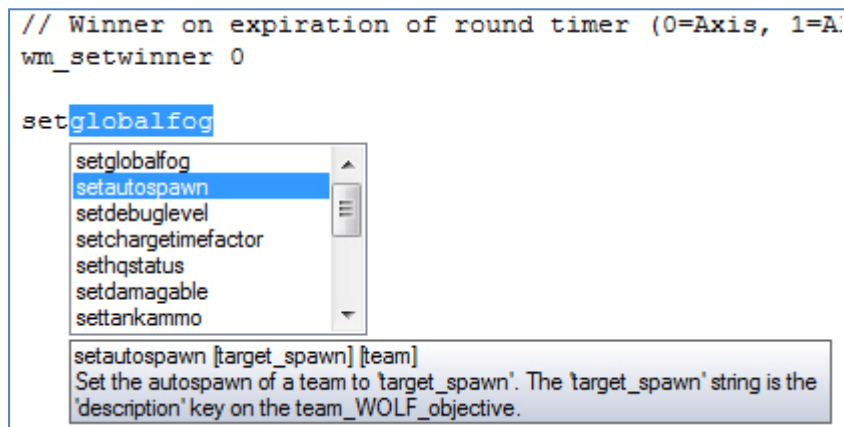


Figure 3 - Intellisense in action.

2.4.2. AutoBackups

The *AutoBackup* feature is an automated backup facility. All the active scripts are automatically saved to a backup folder in a fixed time interval which you can specify in the *Options*. Existing backups are not overwritten; instead, up to 100 backups per map can be saved. This way, you never lose any work if the program or your computer crashes, and you can also go back in time if you made a large error and you are unable to fix it.

AutoBackup can be enabled or disabled through the *Options*. You can see the status of the Autobackup tool in the StatusStrip at the bottom of the program.

To help you find your way through all these backups, you can use the *AutoBackup Manager* tool. See section 2.5.6.

2.4.3. Line numbering

To the left of the script text, a panel containing the line numbers is shown. This allows you to easily find the line number you were looking for.

In addition, the line number of a *Bookmarked* line will turn **bold blue**, see *Bookmarks*.

2.5. Tools

The new features are the most important and handy features in ETScript 3, most of them designed specifically for W:ET map scripting.

2.5.1. The Scriptcommand panel, Error list and Quick Find (menu View)

The *Scriptcommand panel* is a treeview containing every single script command you can use, grouped into different categories. You can see a short description and exactly how to use the command by clicking one in the treeview.

You can hide or show the scriptcommand panel via the *View* menu.

In addition, you can also *Dock* (attach) or *Undock* (detach) the scriptcommand panel by using the button on top.

Additionally, if you come across a certain command you wish to know the description and usage of, you can right-click the command (in the script-text) and choose *Show Intel on* The Scriptcommand panel will jump to that command instantly.

The *Error List* is a list that shows a description of any error that might have occurred. You can hide or show this list via the *View* menu as well.

The *Quick Find* feature is a little toolbar near the bottom of the window, which you can use as a replacement for the extended *Find & Replace* dialog. Simply type a keyword to find into the *Find what* textbox and press the *Next* or *Previous* buttons to find the keyword.

2.5.2. Bookmarks (menu *Tools*)

The *Bookmark* function allows you to bookmark up to 5 lines per script. Put your cursor on the line you wish to bookmark and use the *Tools – Bookmarks – Save* menu to save the line number to a slot.

Using the *Tools – Bookmarks – Load* menu you can quickly load a bookmark, which will make the script jump to the bookmarked line instantly.

Bookmarked lines are stored separately for each script, they are not shared.

Bookmarked lines will show in a bold blue face in the line numbering panel.

Note: You cannot use bookmarks in an untitled document.

2.5.3. Accum count (menu *Tools*)

The *Accum count* function displays a list of the (global)accums that are in use in your script. This way, you can easily see which accums are free to use.

In the *Globalaccums* tab, all the globalaccums are displayed.

In the *Local accums* tab, you first need to specify the scriptblock in question. Pressing Enter will then show the list.

2.5.4. Bracket count (menu *Tools*)

The *Bracket count* function counts the number of opening ({) and closing (}) brackets in your script. A message informs you if the number of opening brackets is not equal to the number of closing brackets, and how many faults have been found.

Note that brackets that are commented out (brackets in a commented line) are also counted!

2.5.5. Comment selection & Uncomment selection (menu *Tools*)

The *Comment* and *Uncomment selection* functions allow you to quickly comment out a large block of code, without having to type the // comment characters manually in front of each line.

Simply select a block of code and use the *Comment selection* function to automatically insert a comment character (//) in front of every selected line.

Using the *Uncomment selection* on a selected block of code, a comment character (//) is removed from the beginning of every selected line, if there is one present.

2.5.6. AutoBackup Manager

The *AutoBackup Manager* tool is a useful little tool that allows you to find your way through all the available *AutoBackups*.

Choose the map you want to see the *AutoBackups* from, and a list of all the *AutoBackups* will appear. You can order the list by name, size or creation date/time. Ordering by creation date/time allows you to find the last backup easily.

Select the backup you wish to open and click the *Open selected* button to open it in a new MDI child document. Note clearly that the original script file has *not yet been overwritten*! You have simply opened the backup file in an *Untitled Backup* script and you need to save it manually.

Use the *Delete selected* or *Delete all* buttons to delete selected or all scripts.

You can use the *View directory* button to have Windows Explorer browse to the folder where the *AutoBackups* are stored.

Note; it is not recommended to view/edit/delete autobackups manually through Windows Explorer. Instead, use the *AutoBackup Manager*!

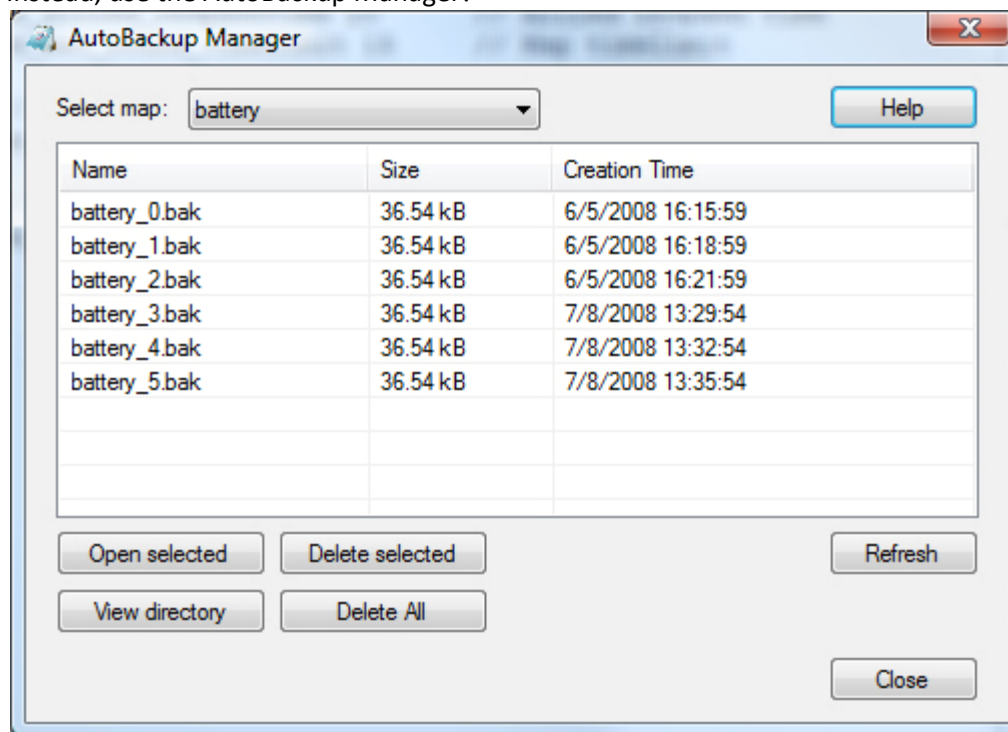


Figure 4 - The AutoBackup Manager tool.

3. Options

In this section, I will explain the different options you can set in the *Options* window, which can be accessed via the *Tools – Options* menu.

3.1. General

The *General* options have influence on the behavior of ETScript in general.

The *W:ET ‘maps’ directory* option textbox is used by the program as the default initial directory of both the *Open File* and *Save File* dialogs. It is most convenient to set this to the location where you usually store your script files (which is the *etmain/maps* directory by default).

Click the [...] button to browse to the correct folder, or click the *Default* button to revert the location back to the default: *C:\Program Files\Wolfenstein – Enemy Territory\etmain\maps*.

The *Load template on New File* checkbox determines whether the script *template* is loaded instead of a blank new document whenever you choose to open a new file.

If the *Remember open scripts on closing* checkbox is checked, the location of every open script file is stored when you close ETScript. Whenever you re-open ETScript, these script files will be opened automatically.

If the *Show Scriptcommands panel on startup* checkbox is checked, the *Scriptcommand panel* will be shown by default when you start ETScript. If it is unchecked, the panel will be hidden.

If the *Show warning message before ‘Close all’ scripts* checkbox is checked, a warning messagebox will inform you when you are about to close all scripts. While every non-saved script will still fire a ‘Do you want to save?’ message, it is probably best to realize that you are closing all the scripts if you ever click this by mistake.

Lastly, the *“Insert Date” date format* dropdown box allows you to choose between a variety of date formats which will be used by the *Edit – Insert Date* feature. Should you choose the *Custom** option, a textbox will appear in which you can type your own custom format. You can see the result of that particular format immediately on the date to the right.

3.2. Text editor

The *Text editor* options control most of the features used in the text-editor.

The *Enable Auto-Indenting*, *Enable Block-Indenting* and *Enable Intellisense* options determine whether or not these features are enabled. See section 2.2.3. and 2.4.1. for an explanation of these features.

If the *Type { instead of [* checkbox is checked, you type an opening bracket ({) instead of a square opening bracket ([). This is useful if *Block-Indenting* is enabled and you want to type an opening bracket without the automatic ‘bracket-block’.

The *Show Linenumbers* checkbox determines whether or not the line numbers in a script document are shown.

Finally, the *Texteditor font* option allows you to change the font (and color) of the text in the scripts. Click the *Select* button to choose a font and color from the *Font dialog*. The selected font will appear in the textbox. Use the *Default* button to revert back to the default black, *Courier New*, 10pt font. Note that changes in font are only visible if you re-open your scripts; they are not changed in “real-time”.

3.3. AutoBackup

In the *AutoBackup* tab, you can enable or disable the *AutoBackup* feature. You can also use the *AutoBackup time interval* option to change the wait time between two backups. Finally you can change the extension the backup files use, either *.script* or *.bak*.

3.4. Appearance

Lastly, in the *Appearance* tab you can change the color-style of the program.

4. Contact

Feel free to contact me for various things such as questions about ETScript, questions about how I created it, or to report bugs (please!).

You can reach me by email at r0fnick@gmail.com, or you can leave a reply in the appropriate ETScript 3 thread on the Splashdamage forums.

5. Thanks

I would like to thank a few people for their help. Without you, I would never have gotten this far with this program, thanks!!!

Roeltje: For the actual idea of ETScript, first created by him as 'Masterscript' but abandoned later.

Chruker: For his complete script command reference:
http://games.chruker.dk/enemy_territory/scripting_reference.php

VBForums: A forum on which I had to ask for help countless times...
<http://www.vbforums.com/>

Splashdamage: For the ideas and support of the Splashdamage mapping community.

ETScript 3 was written in Visual Basic.NET, using Visual Studio 2008.
The .NET Framework version 2.0 is required.

Feel free to distribute ETScript 3 as is, with my name as the author, without making changes and including this readme file.