

Main Page

Welcome to the Aegisub manual. See our [about](#) page for more information on the program, or browse the topics on the left bar.

Aegisub also has [another wiki](#) , used for keeping track of more development-related issues. It might eventually be merged into this one?

Starting points

[Quick overview of Aegisub](#)

Common tasks:

- [Timing a translated script to audio](#)
- [Basic typesetting of on-screen signs](#)
- [Putting your finished subtitles onto the video](#) (for viewing or distribution)

- [Translating a subtitle file from one language to another](#)
- [Correcting a subtitle that's not properly synchronised to the video](#)

- [Timing karaoke to a song](#)
- [Creating fancy karaoke effects](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

About

Aegisub was originally created as a tool to make typesetting, particularly in anime fansubs, a less painful experience. At the time of the start of the project, many other programs that supported the Advanced Substation Alpha format lacked (and in many cases, still lack; development on several competing programs have since been dropped for various reasons completely unrelated to Aegisub) many vital functions, or were too buggy and/or unreliable to be really useful.

Since then, Aegisub has grown into a fully fledged, highly customizable subtitle editor. It features a lot of convenient tools to help you with timing, typesetting, editing and translating subtitles, as well as a powerful [scripting environment](#) called Automation (originally mostly intended for creating karaoke effects, Automation can now be used much else, including creating macros and various other convenient tools).

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Overview

Navigation

» Main Page

Introduction

- » What is Aegisub?
- » Highlights
- » Credits
- » Support Aegisub
- » FAQ
- » Tutorials

Working with Subtitles

- » Editing Subtitles
- » Exporting Subtitles
- » Applying Subtitles
- » Spell Checker
- » Translation Assistant
- » Paste Over
- » Select Lines

Typesetting

- » Introduction
- » Working with Video
- » Editing styles
- » Visual Typesetting
- » ASS Override Tags
- » Colour Picker
- » Styling Assistant
- » Resolution Resampler
- » Fonts Collector

Timing

- » Working with Audio
- » Shift times
- » Timing Post-Processor
- » Kanji Timer

Automation

- » Overview
- » Karaoke Templater
- » Lua Reference

Miscellaneous

- » Aegisub Options
- » Script Properties
- » Attachment Manager

The screenshot displays the Aegisub v2.2.0 interface. The main window is titled "clampazar-wish.base.ass - Aegisub v2.2.0". The interface includes a menu bar (File, Edit, Subtitles, Timing, Video, Audio, Automation, View, Help) and a toolbar. The central area is divided into several sections:

- Video Box:** Shows a video frame with subtitles. The text on screen is "meguriaeru you ni 巡り逢えるように" and "...was for us to stay together.".
- Audio Box:** Shows a waveform and playback controls. The time displayed is 0:00:23.04.
- Edit Box:** Contains fields for Comment, Romaji, Actor, and Effect. Below these are buttons for Bold (B), Italic (I), Underline (U), Strikethrough (ABC), and other formatting options. The text area contains a line of ASS code: `{\k15}me{\k17}gu{\k31}ri{\k30}a{\k31}e{\k24}ru{\k6} {\k93}yo{\k17}u{\k9} {\k241}ni`.
- Subtitles Grid:** A table showing subtitle lines with columns for #, Start, End, Style, Effect, and Text.
- Status Bar:** Shows the current time and frame: 0:00:20.311 - 487, +11ms; -5129ms.

#	Start	End	Style	Effect	Text
1	0:00:00.00	0:00:00.00	Default		--- Karaoke effect ---
2	0:00:00.00	0:00:00.00	Romaji	template pre-line romaji	!retime("set",line.start_time-200,line.end_time+200)!{\fad(150,200)}pos(\$x,\$y) k20}
3	0:00:00.00	0:03:00.85	Romaji	template line romaji	{\kf\$kdur}
4	0:00:00.00	0:00:00.00	Kanji	template pre-line kanji	!retime("set",line.start_time-200,line.end_time+200)!{\fad(150,200)}pos(\$x,\$y) k20}
5	0:00:00.00	0:03:00.85	Kanji	template line kanji	{\kf\$kdur}
6	0:00:00.00	0:00:00.00	Translation	template pre-line main	!retime("set",line.start_time-200,line.end_time+200)!{\fad(150,200)}pos(\$x,\$y)}
7	0:00:00.00	0:00:00.00	Default		--- Wish ---
8	0:00:05.61	0:00:07.75	Romaji		{\k29}I{\k0} {\k51}wish{\k12}, {\k54}my{\k9} {\k59}wish
9	0:00:08.06	0:00:10.20	Romaji		{\k17}When{\k0} {\k18}I{\k0} {\k30}wish{\k0} {\k30}up{\k32}on{\k0} {\k19}a{\k0} {\k70}star
10	0:00:10.50	0:00:15.12	Romaji		{\k10}ho{\k18}shi{\k7} {\k18}ni{\k11} {\k30}ne{\k30}ga{\k28}h{\k0} {\k51}o{\k42} {\k32}ka{\k62}ke{\k46}ta{\k7} {\k70}no
11	0:00:15.41	0:00:17.54	Romaji		{\k27}I{\k6} {\k48}wish{\k9}, {\k64}my{\k0} {\k59}wish
12	0:00:17.86	0:00:19.98	Romaji		{\k25}My{\k7} {\k51}dream{\k10} {\k55}comes{\k0} {\k64>true
13	0:00:20.30	0:00:25.44	Romaji		{\k15}me{\k17}gu{\k31}ri{\k30}a{\k31}e{\k24}ru{\k6} {\k93}yo{\k17}u{\k9} {\k241}ni
14	0:00:27.97	0:00:31.99	Romaji		{\k90}kou{\k30}ha{\k60}ku{\k29} {\k22}na{\k11} {\k33}ku{\k22}mo{\k8} {\k15}ni{\k4} {\k44}sa{\k33}ku
15	0:00:32.87	0:00:36.83	Romaji		{\k88}hi{\k32}ka{\k54}ri{\k29} {\k27}no{\k11} {\k35}ha{\k22}na{\k37}zo{\k61}no
16	0:00:37.46	0:00:39.58	Romaji		{\k14}me{\k0} {\k11}ni{\k6} {\k28}mi{\k32}e{\k31}na{\k18}h{\k6} {\k23}ha{\k11}na{\k4} {\k28}ga
17	0:00:39.86	0:00:42.03	Romaji		{\k19}shi{\k17}ro{\k21}h{\k9} {\k29}ha{\k28}ne{\k4} {\k23}ni{\k7} {\k15}na{\k17}t{\k28}te
18	0:00:42.32	0:00:45.81	Romaji		{\k14}so{\k19}t{\k18}to{\k14} {\k30}ka{\k27}ta{\k6} {\k24}ni{\k6} {\k23}o{\k36}chi{\k23}te{\k9} {\k24}ku{\k76}ru

This screenshot shows the Aegisub main window with everything open:

The two most important areas are the Subtitles Grid and Edit Box.

The grid shows all the subtitle lines in the file you're working with. Lines are marked with different colours depending on their properties. For example, lines with green background are selected. Lines with blue background are marked as comments and won't be shown on screen. Lines with a light yellow

background are visible on the current video frame. Lines with red text overlap in time with a currently selected line. Click a line here to select it for editing.

- [Editing Subtitles](#)

The **edit box** is where you can edit the text and other properties of a subtitle line directly. While you can enter times in the time boxes here, you will rarely need to, as it's much faster to time lines using the **audio box**. You can also time to video. You can use the formatting toolbar to insert formatting codes into the current line. The four "AB" buttons change the colours of different elements of the line. Change the text for the subtitle and press Enter to commit the changes and go to the next line. Holding Ctrl while pressing Enter will stay on the line instead of going to the next, but still commit the changes.

- [Editing Subtitles](#)

Use the **audio box** to time subtitles (synchronize them to audio) in a fast and accurate way. The **audio display** can show the audio you have loaded in two different ways, waveform and spectrum, the screenshot above shows it in spectrum mode.

- [Working with Audio](#)

The **video box** allows you to see how the subtitles will look on the video, but it also allows you to select specific video frames a subtitle line should start/end on, and it allows you to use the mouse to position, rotate and crop a subtitle line to for example match an onscreen sign in the video.

- [Working with Video](#)

Finally, there are three bars in the program: the **status bar** is used to show status messages to you, and the **menu bar** and **tool bar** are used to access Aegisub's myriad of tools:

- [Export](#)
- [Spell Checker](#)
- [Translation Assistant](#)
- [Paste Over](#)
- [Select Lines](#)
- [Styles Manager](#)
- [Colour Picker](#)
- [Styling Assistant](#)
- [Resolution Resampler](#)
- [Fonts Collector](#)
- [Shift Times](#)
- [Timing Post-Processor](#)
- [Kanji Timer](#)
- [Automation Manager](#)
- [Options](#)
- [Script Properties](#)
- [Attachment Manager](#)

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Audio

(Redirected from [Audio Timing Tutorial](#))

Aegisub has a fairly advanced, customizable audio mode with both the traditional waveform display as well as an alternative spectrum display. Several different timing modes are available for both normal dialog timing and karaoke timing.

Todo: This page should probably be split into several smaller ones to make it easier to digest, easier to link, less confusing and wall-of-text and to promote going more in depth with the separate topics.

Opening audio

To load an audio file into Aegisub, just go to the *Audio* menu and press *Open audio file*. If you have a video file (with an included audio track) already loaded, you can use *Open audio from video* instead, which obviously will load the audio track from the video file you currently have loaded. You can open any type of audio file that your [audio provider](#) can decode, more on this below.

Supported formats: Windows

Under Microsoft Windows, your audio provider is *Avisynth* by default, which means that any audio format that your DirectShow environment knows how to decode is supported (at least in theory). For example, if you want to load an AC3 file, you will need an AC3 DirectShow decoder (e.g. AC3filter or ffdshow). *Note:* some formats seem pretty buggy at the moment. Ones more or less guaranteed to work are (16-bit) PCM-WAV, MP3 and Vorbis, so if your audio doesn't work, try transcoding to one of them, at least temporarily.

Warning: If you have opened a video file with more than one audio track (most commonly an MKV or OGM file), and try to open audio from it, Aegisub is completely at the mercy of the splitter when it comes to what audio stream is delivered. Some splitters may deliver both audio streams at once (this will happen for dual audio AVI's, when using the default Windows splitter), and since Aegisub very much doesn't expect that, you will get weird results (and probably crashes). Just remux the file to single audio, or better yet, decompress the desired audio stream to WAV.

Supported formats: non-Windows

On all other operating systems (MacOS X, GNU/Linux, the BSD variants etc.) your audio provider is *ffmpeg*, which means you can use any audio format that *ffmpeg* supports (and was compiled with).

Audio caching

If you're loading any audio format that isn't an uncompressed (PCM) Microsoft WAV file, Aegisub needs to decode and cache it first. When loaded, the audio is downmixed to mono (see the [audio downmixer option](#) if you want to grab one channel only instead), decompressed to PCM (a.k.a. WAV), and (by default) loaded into a RAM cache. This means that you will need a *large amount* of RAM to open a long compressed audio file. If your computer doesn't have a lot of RAM, or if you're working with a full-length movie, refer to the [audio cache option](#) for instructions on how to make Aegisub use its (slower) hard drive cache instead; or decompress the file to WAV first since Aegisub can read from WAV's directly without need for caching.

The exact amount of memory used for any given audio file can be calculated with the following

Contents [\[hide\]](#)

- 1 Opening audio
 - 1.1 Supported formats: Windows
 - 1.2 Supported formats: non-Windows
 - 1.3 Audio caching
- 2 The main audio view
- 3 Basic audio timing
 - 3.1 Timing protips
 - 3.2 The spectrum analyzer mode
- 4 Karaoke timing

formula:

$$s = (b * r * l) / 8$$

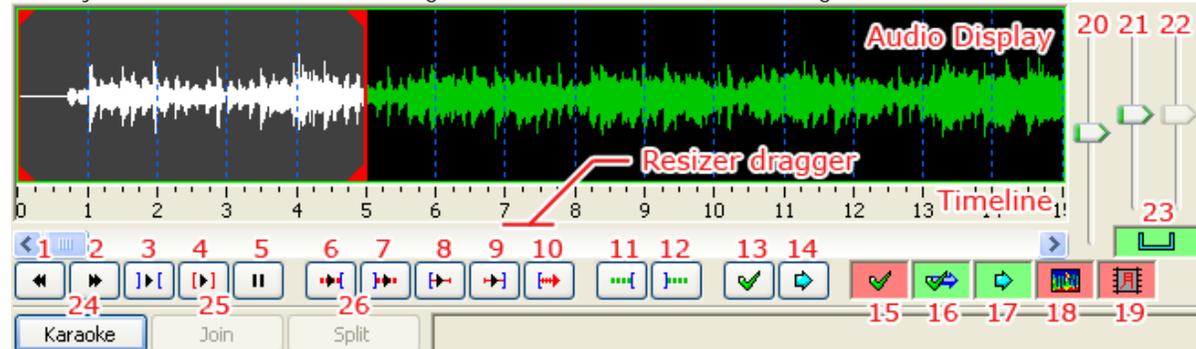
where s is the amount of memory (in bytes - divide by 1024 to get kB), b is the number of bits per sample (always 16 in the current implementation), r is the sample rate in Hz (usually 48000, or 44100 in some cases), and l is the length of the audio (in seconds).

For example, for a 25 minute audio clip at 48 kHz, you will need $(16 * 48000 * 25 * 60) / 8 = 144000000$ bytes \approx 137 MB.

Loading and decompressing the audio into the cache will take a few seconds; Aegisub will display a progress indicator while loading the audio.

The main audio view

When your audio file has loaded, Aegisub will transform into something like the screenshot below:



You can click and drag just below the audio timeline to change the height of the audio waveform/spectrum display.

Green and red buttons are toggle buttons. A green background indicates that the option is turned on, while a red background indicates that the option is turned off. The buttons and controls are as follows (many of these have [keyboard shortcuts](#) associated with them by default):

1. Go to previous line, discarding any unsaved changes (previous syllable when in [karaoke mode](#))
2. Go to next line, discarding any unsaved changes (next syllable when in [karaoke mode](#))
3. Play selected area of the audio waveform
4. Play currently selected line
5. Pause playback
6. Play 500ms before selection start
7. Play 500ms after selection end
8. Play first 500ms of selection
9. Play last 500ms of selection
10. Play from selection start to end of file (or until pause is pressed)
11. Add lead-in (how much is determined by the [audio lead in setting](#))
12. Add lead-out (exactly like the above, but the setting is called [audio lead out](#), logically enough)
13. Commit (save) changes
14. Scroll view to selection/go to selection
15. Toggle auto-commit (all timing changes will be committed immediately, without the user pressing commit, if this is enabled)
16. Toggle auto next line on commit (if this is enabled, Aegisub will automatically select the next line when the current line is committed; enabling both this and auto-commit at the same time is strongly discouraged)
17. Toggle auto-scrolling (will center waveform on the currently selected line automatically when enabled)
18. Toggle spectrum analyzer mode (see below)
19. Toggle Medusa-style timing shortcuts
20. Audio display zoom (horizontal)
21. Audio display zoom (vertical)
22. Audio volume
23. Toggle linking of vertical audio zoom slider with volume slider

24. Toggle karaoke mode
25. Join selected syllables (karaoke mode only)
26. Split selected syllables (karaoke mode only)

Basic audio timing

When you click on a line in the subtitles grid, Aegisub will highlight it in the audio display and, if you have auto-scrolling enabled, scroll the audio display so it's centered on the line (during normal timing, it's probably a good idea to disable auto-scrolling). You'll notice various vertical lines in the audio display; the dark blue ones indicate second boundaries, the pink ones indicate keyframes in the video if you have it loaded (see the [Working with video](#) section), the white broken line indicates the currently visible video frame, and the thick red and orange ones are the line start and end markers (respectively) for the current line. To (re-)define the start and end times of the line, you can either left-click to set the start time and right-click to set the end time, or just drag-and-drop the line boundaries. The selection background will turn red and display the word "Modified" in the top left corner of the audio display when you've changed the timing but haven't saved the changes yet. It will remain red until you either press the commit button (*enter* or *g* by default) or go to another line (discards changes). If you have auto-commit on, the background will never turn red since all changes will be saved immediately. Press the *play* button (keyboard shortcut *s* by default) to listen to the selection, or the various other playing buttons to listen to parts of the selection or the audio surrounding it. When you are satisfied with the timing, press commit. Then repeat once for every line; it's as simple as that.

Timing protips

If you want to finish timing your movie or episode within any reasonable amount of time, there's some things you should note:

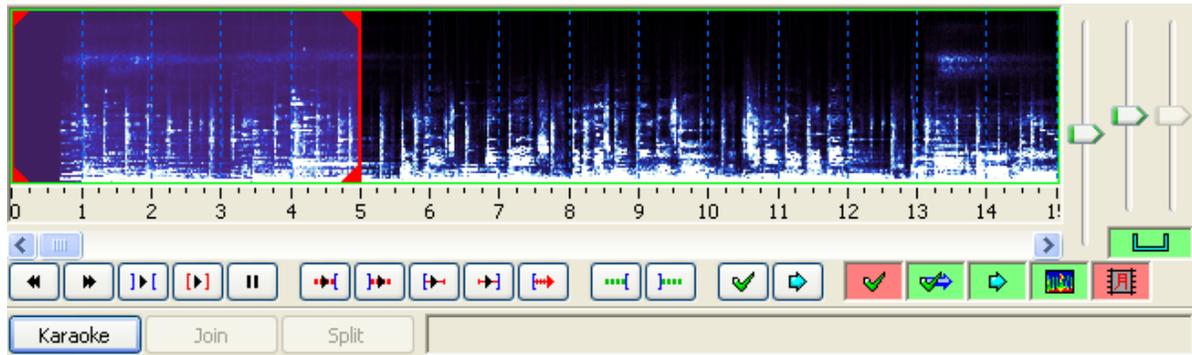
- Use keyboard shortcuts! They speed up your work by several orders of magnitude.
- You don't need to have video displayed while timing. Scene-timing, i.e. syncing line start/ends to scene changes, can be done later. Either manually, or with the [timing postprocessor](#).
- Use "go to next line on commit".
- Experiment with different timing styles when you're new and stick to one that suits you. Then practice. Lots.
- Aegisub heavily relies on the concept of "focus", and doing things in a way that require you to switch back and forth between video/audio/subtitle edit box a lot will cost you a lot of time. Do it in several "passes" instead.
- The spectrum analyzer mode can make it a lot easier to "see" where lines start and end.

One common timing style (preferred by the author of this page) goes something like the following: Turn on "go to next line on commit" but disable auto-committing, auto-scrolling and Medusa timing shortcuts. Keep the four main fingers of your left hand on *s/d/f/g*. You won't be using the thumb so do whatever you want with it. Keep your right hand on the mouse. Now select (by left- and right-clicking) an area in the waveform that seems likely to contain a line of speech matching the current subtitle line, and hit *s* to play it back. While it's playing, adjust the start time if necessary. When the playback marker has passed the end time mark, adjust the end time as well. If greater accuracy is needed, play the last 250ms of the selection by pressing *d*, 250ms before the selection start by pressing *q*, 250ms after the selection end by pressing *w*, or the first 250ms of the selection by pressing *e*. As you grow more experienced, you won't be using anything else than *s* very much, except maybe *d* and *q*. When you're satisfied with the timing, hit *g* to commit changes and go on to the next line. Scroll the audio display forward by pressing *f*. If you need to scroll it backwards, use *a*. To go to next or previous line without committing changes, use *z* and *x*.

This style has the advantage that you never need to move your hands at all. With some training, it can also be very fast; audio timing 350-400 lines of dialog to a 25-minute episode can easily be done in less than 40 minutes.

Of course, this style may not feel comfortable for all people; you should experiment with other timing styles before deciding which one is best for you.

The spectrum analyzer mode



When you press the spectrum analyzer button, the waveform does no longer show amplitude (signal strength) on the vertical axis - instead it shows frequency. The higher up, the higher the frequency. The colors instead indicate amplitude, with black/dark blue being silence and white being the strongest sound. This may seem confusing, but since the frequency window is set to fit human voices rather well, it can make it easy to tell where a line (or a word in karaoke mode) starts and ends when there's a lot of background noise (or music) that makes it hard to tell from the normal waveform. It can be especially useful when timing karaoke. Play around with it for a little while, and you'll understand how it works.

Note that in spectrum analyzer mode, the "vertical zoom" slider is redefined to control color intensity instead, since the colors indicate signal strength.

Because calculating the spectrum data is very CPU intensive, it is initially set to be in a medium quality. You can increase the quality of the spectrum in the [audio options](#).

Karaoke timing

Todo: here be dragons

Category: [Pages with Todo items](#)

Tutorials

The following tutorials are available for Aegisub:

Video Tutorials (Screencasts)

Dialogue Timing

Todo: actually do this

- By: Karl Blomster

Kanji Timer

- Description: A video tutorial demonstrating how the kanji timer can be used to synchronize kanji timing to roomaji timing.
- By: Niels Martin Hansen
- Details: XviD MP3 AVI, 5 min 20 sec, 12 MB
- Link: <http://jfs.itcamefromjapan.net/aegisub/demovids/kanji-timer.avi> 

Karaoke Timing

Todo: Actually do this

- By: Niels Martin Hansen

Visual Typesetting

- Description: A video tutorial demonstrating how to use Aegisub 2's new Visual Typesetting features.
- By: Rodrigo Braz Monteiro
- Details: Matroska/XviD/Vorbis, 10 min 22 sec, 25 MB
- Subtitles: English, Portuguese (Brazil) (by amz), French (by shawurai), German (by Sqall)
- Link: malakith.net 

Text tutorials

- [Karaoke timing](#) (work in progress) – Enter the words for a song and add time-codes to the words
- [Karaoke Templater](#) (incomplete) – Easy way to add special effects to karaoke, ranging from the most basic effects to very advanced ones.

Category: [Pages with Todo items](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Contents [\[hide\]](#)

- 1 Video Tutorials (Screencasts)
 - 1.1 Dialogue Timing
 - 1.2 Kanji Timer
 - 1.3 Karaoke Timing
 - 1.4 Visual Typesetting
- 2 Text tutorials

Attaching subtitles to video

In digital encoding, there are two main ways of including subtitles in a video. There is softsubbing, and hardsubbing. Both methods has unique advantages and disadvantages, along with various arguments both for and against each method.

Hardsubbing

Hardsubbing is a method that "burns in" subtitles into the actual video portion of a movie. Digital hardsubs are much like subtitled VHS tapes; the subtitles cannot be turned off.

Advantages of Hardsubbing

Hardsubbing is usually much less demanding on the playback device. Since the text is already part of the video, it will only take as much processing as the unsubtitled video would. You are also often able to make special effects that would be difficult to replicate in a soft subtitle format, because of the large amount of CPU usage required to render them. Even in softsubbed anime fansubs, the opening and closing karaoke are often hardsubbed because of the special effects used.

Some people argue that with hardsubs, scripts are harder to steal, since the text is embedded in the image - thieves cannot simply extract subtitles as in a softsub. However, the presence of very good subtitle extractors designed for the purpose of extracting this embedded text removes much of the argument that hardsubs prevent script stealing.

Many playback devices and computer platforms cannot display the special fonts and formattings that softsubs contain, but this problem is removed with hardsubs, where the style is preserved. Also, these stylings will show back exactly the same on any device, unlike softsubs which depend on the playback device to properly intepret and display the stylings.

Disadvantages of Hardsubbing

Despite what some may call numerous advantages for hardsubbing, there are several distinct disadvantages that should be evaluated before making a decision.

The method of hardsubbing requires that the source video is re-encoded so the subtitles can be written on the image. This, by the nature of lossy video encoding, causes a reduction in video quality.

Subtitles add a sharp contrast in a video image due to their nature. This will cause compression artifacts along the edges of the encoded subtitle, and blurring of the subtitle. This effect is especially evident at lower bitrates.

Under typical circumstances, the inclusion of the subtitles will cause an increase in the bitrate needed for the video to keep the same quality. This, of course, means an

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Contents [hide]

- 1 Hardsubbing
 - 1.1 Advantages of Hardsubbing
 - 1.2 Disadvantages of Hardsubbing
- 2 Softsubbing
 - 2.1 Advantages of Softsubbing
 - 2.2 Disadvantages of Softsubbing
- 3 What method do I choose?
- 4 Hardsubbing with Avisynth
- 5 Hardsubbing with VirtualDub
- 6 Softsubbing
 - 6.1 Variant 1: softsubs inside the video container
 - 6.2 Variant 2: distributing script files

increased filesize, or lower quality at the same size. The increase in bitrate necessary is typically around 3 to 10%.

Changing the subtitles requires a re-encode of the video source, which can add a lot of time and extra work to the release process.

Softsubbing

Softsubbing is a method that keeps subtitles separate from the video and relies on the playback device to combine the two when the video is being played. This method can be best compared to subtitles on most DVDs. The subtitling can be turned on or off as needed, and multiple languages can be supported with just one combined media file. Unlike with a DVD though, digital softsubs are actually text (DVD subtitles are pictures) which adds many nice features at the cost of complexity.

Advantages of Softsubbing

Softsubs are much clearer on display. Since they are not part of the video image, video compression does not affect them, and with a good subtitle renderer, they are sharp and crisp - a huge benefit to readability.

Softsubs can be smaller. Since the subtitle is just a text file, it can take up less room because it isn't hogging video bitrate. This allows for an encoder to either make a smaller file with the same video quality, or a same-sized file with higher video quality.

People with vision problems have an opportunity to adjust how the subtitles look on-screen.

Without a huge impact on size, multiple languages can be supported in one video file.

If you find a subtitling mistake in a file, you can fix it without having to re-encode the video - saving a lot of time.

Disadvantages of Softsubbing

Softsubs add processing complexity to the video. The playback device has to render and overlay the text before displaying the video, as a result, this means that low-powered devices will not be able to play the video.

Since the subtitles are bundled as straight text they are very easy to extract and use. This makes things easier on bootleggers or other script stealers. Note that grabbing subtitles from a hardsub is very easy currently, so this argument doesn't hold much weight.

The playback device is responsible for rendering the subtitles on screen. As a result, they might not look the same as the subtitler intended. In some cases, the playback device might not support the subtitle format, or might have bugs with it.

The AVI file format is not reliable for supporting softsubs, if you plan to use it. Please note that Matroska (MKV) is very well supported by computers, so this isn't a major negative.

Subtitles with effects added (usually for karaoke) take up a lot of processing time, and may cause playback issues if the device cannot handle the processing requirements. A solution for this is to hardsub the complex parts such as opening and ending karaoke, and softsub the normal dialog.

What method do I choose?

The method you should choose depends greatly on your audience. Will they have relatively new and powerful playback devices? Will they possibly be able to install something to play back softsubs if they don't have it? Is your destination a digital format (Matroska, DVD, etc.) or will you be printing to tape?

While every situation will be different, you can use some of the following suggestions to guide you. These are based on making a digital format for playback on a computer system.

If you want your file playable on the largest range of computers and operating systems, you will want to hardsub.

If your audience will have relatively new machines (500 MHz or greater) and will be running on a platform where your subtitle format is well-supported, softsubs are a good idea.

If you want to speed up your release process, use softsubs. They are faster to fix if an error is found.

Hardsubbing with Avisynth

Many people use the Avisynth package to add filters to their video to clean up defects, or otherwise manipulate the video image before encoding it. It is a very flexible tool, and can be also used to add subtitles directly to the video stream, allowing an easy and scriptable method to hardsub a video.

If you are unfamiliar with Avisynth, it is recommended that you look into it, as it has lots of nice features and a large community contributing video filters, allowing easy video fixes for any source. This tutorial assumes you have some basic knowledge of Avisynth.

To allow adding subtitles to the video stream, you need the VSFilter plugin (you may also use [Asa](#), but at the time of writing, it's not as complete as VSFilter is). You can find this on the "csri" folder in your Aegisub installation.

To just add subtitles, you will want to make a simple AVS file containing the script lines you need. Simply create a plain-text file in notepad (or your favourite text editor) and save it with the .avs extension (beware that Windows might be hiding your extension, and you might actually be making a .avs.txt file). Here is an example:

```
LoadPlugin("c:\program files\aegisub\csri\vsfilter.dll")
AVISource("c:\projects\project1\video\mycoolvideo.avi")
TextSub("c:\projects\project1\subs\mainsubtitles.ass")
TextSub("c:\projects\project1\subs\endkaraoke.ass")
```

The above script will take an AVI file (mycoolvideo.avi), and then draw the contents of two subtitle files on the video. You can then encode this video in any program that supports AVS, such as [VirtualDub](#). To do so, just open the .avs file in the program, and follow the normal encoding procedure for it.

Keep in mind that, due to a bug in VSFilter, the path to the subtitle files MUST be absolute.

Hardsubbing with VirtualDub

If you're already familiar with VirtualDub filters, and don't intend to do any other video processing, you should note that it's possible to use VSFilter as a VirtualDub filter as well. Just rename the .dll to .vdf and copy it to the VirtualDub plugins folder. The filter will then be available as "TextSub".

Warning: VirtualDub comes with a TextSub of its own, that is called "TextSub 2.23". This is a very old version that, amongst many other issues, cannot parse UTF-8 (the default Aegisub encoding) files properly. This will result in any non-ASCII characters being rendered as gibberish. NEVER USE THIS FILTER.

Softsubbing

Softsubbing a video can be done in several ways. On Windows using a DirectShow player, such as Media Player Classic, ZoomPlayer or even Windows Media Player, you need VSFilter installed to view the subtitles. If you use MPlayer, you need libass and FontConfig compiled to correctly view all the formatting. Please note that VLC does currently not support softsubs very well; it is strongly recommended that you avoid it for viewing softsubbed files.

Variant 1: softsubs inside the video container

Matroska Video (MKV) is currently the best container for this method (MP4, OGM and even AVI can technically contain softsubs, but none supports font attachments, and all of them has various other issues). Using a muxer that supports attachments (i.e. [mkvmerge GUI](#)), you simply add your subtitle files to the Matroska file as separate tracks (just like you add audio and video tracks), and any fonts as attachments (make sure they have the MIME type application/x-truetype-font). The fonts will then be installed temporarily by Haali Media Splitter (on Windows) or MPlayer (on *nix and MacOS X) during playback.

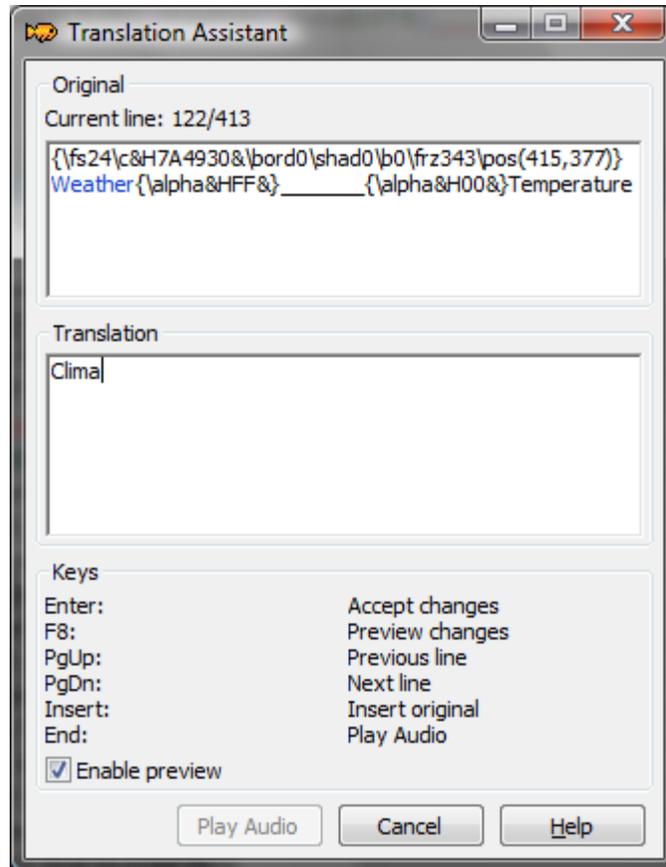
Variant 2: distributing script files

This method works best when you want to encode the video in an AVI wrapper. You simply send the raw subtitle files along with the video. The viewer then needs to load them in a player that supports external subtitles. When using this method, you either need to make sure you use fonts that everyone can be expected to have installed, or distribute a separate ZIP archive with the fonts. For obvious reasons, this method isn't recommended.

Translation Assistant

The translation assistant is a valuable tool for translating subtitles from one language to the other. It allows you to translate in the most efficient possible manner, by letting you type the translated text without having to worry about selecting the original, or forgetting what was written in it. It also parses ASS override tags, and will skip over them.

Overview



The assistant window is simple: it shows you the original raw line, and a box where you can type the translation. There is also a list of hotkeys specific to this window, and a checkbox to enable previewing, if you have the corresponding video set up. Having this option enabled will automatically jump to the current line when you change it.

How to use

Using the assistant is very easy. First, start it by clicking its icon in the toolbar or going to Subtitles->Translation Assistant. It will start at the currently selected line. The original line will have the text that you have to translate highlighted in blue. Simply type the translation to it, and hit enter. If there are override tags in the line, you may have to translate more than one block per line.

In the above screenshot, supposing that you are translating to Portuguese, you would first type "Clima" (Portuguese for Weather) and hit enter. It would then highlight "_____". Since I don't want to translate that, I just hit Insert in the keyboard, and it

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

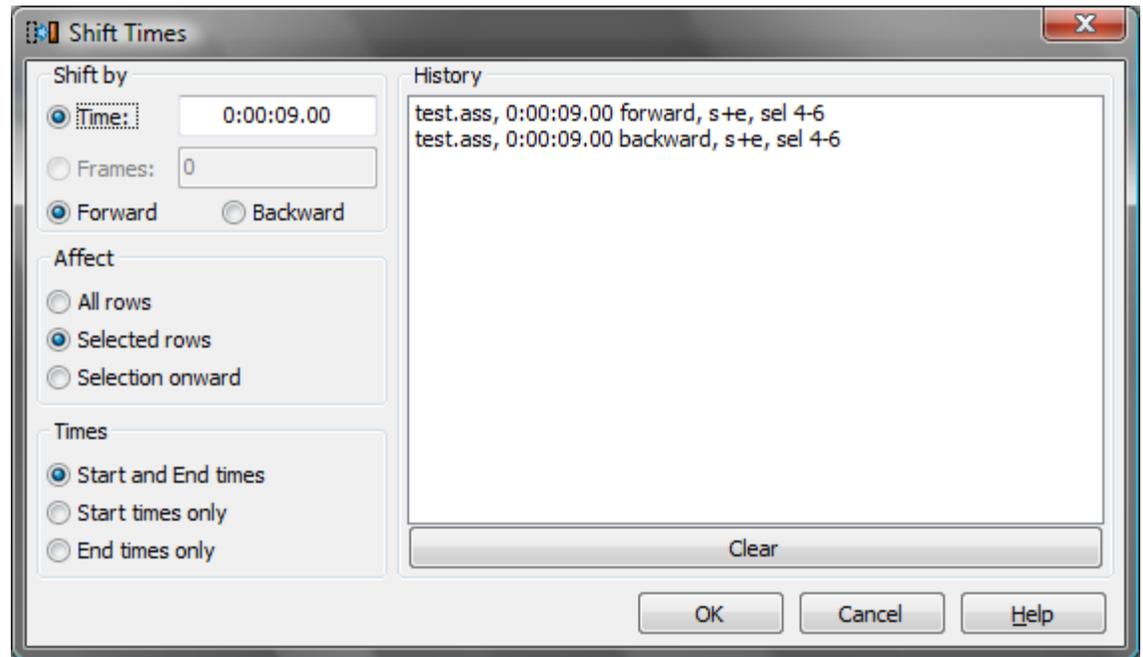
will copy the text over. So I hit enter, and it will ask for the last bit, "Temperature", so I enter "Temperatura" and hit enter again. Then it will jump to the next line, where I will repeat the process. If I make a mistake, I can use Page Up and Page Down keys to move to the next or previous block/line.

Shift Times

(Redirected from Fix Timing Tutorial)

The **shift times** tool is a batch processor for timestamps; it adjusts the start/end timestamps of many subtitle lines at once in various ways. It is located under *Timing* -> *Shift Times*.

It looks like this:



The left part of the window contain the options.

Shift by

These controls decide in which direction and by how much each timestamp will be modified.

- **Time** - How much you want to adjust each timestamp, in hours:minutes:seconds.centiseconds.
- **Frames** - If you have video loaded, you can specify the adjustment time as a number of frames instead.
- **Forward** or **Backward** - controls in which direction the timestamps are adjusted.

Affect

These controls decide what lines will be processed.

- **All rows** - Applies the time shifting to all lines in the script.
- **Selected rows** - Applies the time shifting only to the selected lines.
- **Selection onward** - Applies the time shifting to the selected line(s) and all lines below (in the grid) the last selected line.

Times

Contents [hide]

- 1 Shift by
- 2 Affect
- 3 Times
- 4 History

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

These controls decide what timestamps will be processed.

- **Start and End times** - Both start and end times of the affected lines will be modified by the given amount.
- **Start times only** - Only the start times of the affected lines will be modified. Note that this makes the lines longer (if you shift backwards) or shorter (if you shift forwards) and can even make them have a duration of zero.
- **End times only** - Only the end times of the affected lines will be modified. Note that this makes the lines longer (if you shift forwards) or shorter (if you shift backwards) and can even make them have a duration of zero.

Note that if a line is shifted so that its start or end time stamp would be negative, that timestamp is zeroed instead. This can be used to clear all timings from an entire script, by shifting backwards by longer than the latest timestamp in the script.

History

This is a history of all time shiftings you have done since last time you cleared the shift history (with the clear button). The format is a number of fields separated by commas. The fields are:

- Filename of the script (e.g. "example.ass")
- Shift amount and direction (e.g. "0:00:05.00 forward")
- What times were affected, "s" for start, "e" for end, "s+e" for both
- What rows were affected; "sel start-end" for selections, "all" for all rows (e.g. "sel 1-40")

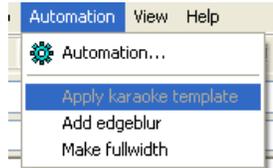
Karaoke Templater Tutorial 1

Welcome to the first tutorial in using Karaoke Templater to create [karaoke effects](#). We'll start out with something simple and then continue on to more advanced effects throughout the series.

Finding Karaoke Templater

You can run Karaoke Templater in two different ways. For now we'll just concentrate on one of them.

Look under the Automation menu in Aegisub.



You should see the option called *Apply karaoke template*, though it will be unavailable. This is what you'll select to use the "templates" we'll be writing throughout these tutorials. Right now it's unavailable because you haven't entered any templates. We'll get to that in a moment.

If you don't have the *Apply karaoke template* option, your Aegisub installation is either incomplete or damaged. You need to repair your installation in that case, to be able to use Karaoke Templater.

But there is something else to make sure of first.

Getting timed karaoke

Karaoke Templater can do many things for you, but it can't guess the lyrics for your song, or how they're synchronised with the music. You'll have to time the lyrics yourself, or get someone else to do it. You can learn how to time karaoke at the [Karaoke Timing Tutorial](#).

We'll call song lyrics with basic karaoke timing (\backslash k timing), but no other special effects, for *timed karaoke*. When an effect has been applied to the karaoke we'll call it *styled karaoke*.

If you don't have any song to work with here are two lines from a song. You can select them in your browser, select *Edit*→*Copy*, then go to Aegisub and select *Edit*→*Paste*, to get them into Aegisub.

```
Dialogue: 0,0:00:01.85,0:00:09.06,Default,,0000,0000,0000,,{\k97}shi{\k41}ta{\k0} {\k20}no{\k10} {\k30}u{\k80}e{\k53}
{\k23}a{\k21}ma{\k39}ku{\k7} {\k24}to{\k24}ke{\k31}ru{\k0} {\k37}wa{\k23}ta{\k92}gu{\k69}mo
Dialogue: 0,0:00:09.28,0:00:16.21,Default,,0000,0000,0000,,{\k79}ki{\k61}su{\k0} {\k9}o{\k0} {\k37}shi{\k98}te{\k40}
{\k23}ku{\k25}ro{\k40}i{\k0} {\k28}tsu{\k19}ba{\k51}sa{\k0} {\k11}no{\k0} {\k34}shi{\k138}ta
```

Either way you have some karaoke timed lyrics now, so we can start.

Load a video

You don't need to load any audio, but it can be a good idea to have some video open. If you don't have any video files ready you can use, select *Video*→*Use dummy video* and just say OK. It's not a very interesting video you will get, but it will show you how the subtitles and the effect we're creating will look.

Writing a *k-replacer* style template

Now we have everything else set up it's time to make the actual template. First, here's how to add it. The explanation of what every part means will follow.

1. Select the very first subtitle line in the file.
2. Select *Subtitles*→*Insert Lines*→*Before Current* to get a new line before it. This will become our karaoke template line. It doesn't *need* to be first, but it usually makes it easier to keep track of for yourself.
3. Make sure the newly created line has the same *style* as your timed karaoke.
4. Click the *Comment* check box for the new line. It should change colour in the subtitle grid.
5. Find the *Effect* field, it's to the right of the *Style* and *Actor* fields. Put the text "template line" into it. (Without the quotes!) Press the *Enter* key on your keyboard to save the Effect field.
6. Finally, enter this text for the main text of your template line. Finish off with *Enter* again.
`{\r\t($start,$mid,\fscyl20)\t($mid,$end,\fscyl00)}`

Contents [\[hide\]](#)

- 1 Finding Karaoke Templater
- 2 Getting timed karaoke
 - 2.1 Load a video
- 3 Writing a *k-replacer* style template
- 4 Extending the effect a bit
- 5 So what does it all mean?

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Todo: Insert a screenshot of how it should look after the steps.

Now look at the *Automation* menu again. If you made the template line correctly, *Apply karaoke template* will now be available. If it isn't review the steps above again.

Select *Apply karaoke templates* and watch Karaoke Templater do its work.

Todo: Another screenshot, this time after applying templates.

If you have a video open you can view the effect right now, inside Aegisub.

Also notice how the template line was left in place, and the timed karaoke was changed to comment lines and had *karaoke* filled into their Effect fields. Karaoke Templater preserves your timed karaoke in the styled karaoke so you won't lose it. But it also has another function...

Extending the effect a bit

Continuing right on from above, now try doing this:

1. Change the text of the template line to:
`{\r\k$kdur\t($start,$end,\lc&H00FF00&)\t($start,$mid,\fscy120)\t($mid,$end,\fscy100)}`
2. Apply templates again

Todo: More screenshot

Karaoke Templater re-used the commented out timed karaoke and changed the styled karaoke to match the new effect instead. You can also try changing the commented out timed karaoke and apply templates again.

Like this, you can work gradually on your effect and preview it as you go.

So what does it all mean?

To finish off this first part of the tutorial, here's what each part means. This is not the full explanation of everything, but it should be plenty for now.

- *Template lines* are lines in the subtitle file marked in a special way. They must always be Comment lines, and the first word in their Effect field must be `template`.
- There are several kinds of template lines. We only used one in this tutorial, this template line type, or *template class*, is called a *line template*. Yes, it might be a bit confusing. It's called this because it creates one line of styled karaoke from one line of timed karaoke. The second word in the Effect field of a template line tells what template class it is. For line templates, this is `line`.
- So, the `template line` text in the Effect field means this is a *template line* of *line class*.
- A template only does something to timed karaoke lines that have the same Style as the template line.
- Styled karaoke produced by Karaoke Templater all has `fx` in the Effect field. This is used as a reminder for Karaoke Templater that this line should be replaced, if templates are applied once again.
- The main text for a template line is called the *template text*. In *line* templates, every `\k` tag is replaced with the template text.
- Template text can use several *variables*. These are short words that start with a dollar-sign, like `$start`, `$end`, `$mid` and `$kdur`. Variables are replaced with the information they store for every syllable that's replaced.
 - `$start` gets replaced with the start time of the syllable. This is in milliseconds from the start of the line, ie. a time code suitable for putting into the `\t`, `\move` and `\fade` tags.
 - Similarly, `$end` is the end time of the syllable, also in milliseconds.
 - A bit more special is `$mid`, this is the *midway time* of the syllable, the time right in the middle of `$start` and `$end`. Here, we used it to grow each syllable taller the first half of its duration, and back to normal height for the second half. Also in milliseconds.
 - The `$kdur` variable, however is in centiseconds. This is the original time from the `\k` tag, and is almost only useful for putting back into a `\k` tag, like we did here.

With this knowledge, you should already be able to create lots of effects. You may also want to refer to the page on [ASS override tags](#).

You can also [continue to the next tutorial](#), where we'll look at how to do math with variables to get more variation.

Category: [Pages with Todo items](#)

Main Page

Welcome to the Aegisub manual. See our [about](#) page for more information on the program, or browse the topics on the left bar.

Aegisub also has [another wiki](#) , used for keeping track of more development-related issues. It might eventually be merged into this one?

Starting points

[Quick overview of Aegisub](#)

Common tasks:

- [Timing a translated script to audio](#)
- [Basic typesetting of on-screen signs](#)
- [Putting your finished subtitles onto the video](#) (for viewing or distribution)

- [Translating a subtitle file from one language to another](#)
- [Correcting a subtitle that's not properly synchronised to the video](#)

- [Timing karaoke to a song](#)
- [Creating fancy karaoke effects](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Highlights

Some highlights of Aegisub:

- **Simple and intuitive yet powerful interface for editing subtitles**
 - With help from such features as spell checking/thesaurus, a translation assistant, call tips for override tags, syntax highlighting and various macros, translating, editing and quality assuring subtitles is now easier than ever.
 - Did a mistake? Had a system crash? Don't worry, Aegisub saves your .ass with features like multi-level undo/redo, auto saving, backups and emergency recovery saving in case of system or program crashes.
- **Support for many formats and character sets**
 - Aegisub doesn't lock you into using any specific format; importing and exporting from and to several different formats (ranging from MPEG-4 Timed Text to MicroDVD) is supported.
 - Multiple language subtitles? Legacy programs without Unicode support? No problem, Aegisub has full support for both Unicode and most common legacy encodings; both reading and writing.
- **Powerful video mode**
 - Most video common video formats are supported out of the box, but if that isn't enough, Aegisub can also use your system's DirectShow environment to open any format you can play in Windows Media Player.
 - Full support for variable framerate video and Matroska timecodes. Capable of reading timecodes directly from Matroska files for seamless VFR work.
 - Problems with anamorphic video? Overscan? Aegisub has the tools to handle it.
- **Visual typesetting tools**
 - Typesetting on easy mode! No need to manually edit override tags; drag, rotate or clip the subtitles directly on the video frame using your mouse.
 - Want to re-use typesetting on a video of different resolution or aspect ratio? Can't figure out what color that sign is using? Aegisub has the tools to help you deal with it.
- **Intuitive and customizable audio timing mode**
 - Time dialogue or karaoke in any way you want with the heavily customizable audio interface; less painful than ever! Use features such as the spectrum analyzer mode or the automatic timing post-processor for even more efficient timing.
 - Support for any audio format your DirectShow environment knows how to decode.
- **Fully scriptable through the Automation module**
 - Is all the above not enough? Extend the program with your own macros or export filters using scripting in Lua or Perl.
 - Comes with several ready-made scripts for quick and easy generation of your very own karaoke effects.

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Credits

Todo: Make sure that nobody is missing from the list - if you are or can think of someone, please let us know.

Contents [\[hide\]](#)

- 1 Programming
- 2 Installer
- 3 Manual
- 4 Hosting
- 5 Localization files
- 6 Additional thanks
- 7 About

Programming

- Main Programmers:
 -  **Rodrigo Braz Monteiro** (ArchMage ZeratuL) - Lead developer, video system, audio system, subtitles I/O, tools, PRS library
 -  **Niels Martin Hansen** (jfs) - Automation, Lua automation, audio system, tools
- Staff Contributors:
 -  **Karl Blomster** (TheFluff) - General code, FfmpegSource video provider, Ffmpeg-related patches, Perl hacking
 -  **Fredrik Mellbin** (Myrsloik) - FfmpegSource and Avisynth-related code
 -  **Amar Takhar** (verm) - New unix build system
 -  **David Lamparter** (equinox) - Unix port, FFMPEG-related programming, asa
 -  **Alysson Souza e Silva** (demi_alucard) - General code
- Other contributors:
 -  **Dan Donovan** (Dansolo) - Miscellaneous tools
 -  **Simone Cociancich** (shb) - Perl automation
 -  **Patryk Pomykalski** (Pomyk) - Ruby automation, Miscellaneous patching
 -  **ai-chan** - ASSDraw3
 -  **Evgeniy Stepanov** (Azzy) - Some Linux patches, libass
 -  **Mike Matsnev** (Haali) - Some DirectShow code and Matroska Parser library
 - **2points** - Linux patches
 -  **p-static** - Linux patches
 - **David Conrad** (Yuvi) - Mac patches
 -  **Daniel Moscoviter** (Mosc) - General patches
 -  **Plorkyeran** - Various patches
 -  **Harukalover** - Usability, various patches

Installer

- 2.x:
 -  **Niels Martin Hansen** (jfs)
 -  **Fredrik Mellbin** (Myrsloik)
 -  **Rodrigo Braz Monteiro** (ArchMage ZeratuL)
- Previous:
 -  movax

Manual

- 2.x:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

-  Karl Blomster (TheFluff)
-  Niels Martin Hansen (jfs)
-  Rodrigo Braz Monteiro (ArchMage ZeratuL)
- Previous:
 -  Motoko-chan
 -  Kobi
 -  movax
 -  Jcubed

Hosting

- Current:
 -  DeathWolf: SVN repository for versions 2.1.2+
 -  Sigurd Tao Lyngse (Bot1): Forums and Main Wiki
 -  Niels Martin Hansen (jfs): Bug Tracker
 -  Marcello Bastéa-Forte: Documentation Wiki
 -  Rodrigo Braz Monteiro (ArchMage ZeratuL): Domain administration
- Previous:
 -  David Lamparter (equinox): SVN repository for versions 2.00 - 2.1.2
 -  BerliOS: SVN repository for version 1.10
 -  Mentar: SVN repository up to version 1.09
 -  Bot1: Bug Tracker

Localization files

Todo: This is outdated

- 2.00 translations:
 -  ArchMage ZeratuL: Brazilian Portuguese
 -  Ereza: Catalan
 -  Yuri: Hungarian
 -  Hiroshi: Japanese
 -  Jeroi: Finnish
- 1.11 translations:
 -  Karasu: Traditional Chinese
- 1.10 translations:
 -  TechNiko: French
 -  Crysral: French
 -  thrash-sensei: Russian
 -  equinox: German
 -  Nesukun: Spanish
 -  EmBolo: Italian
 -  Mazinga: Italian
 -  oblisk: Korean
 -  mulrich: Danish

Additional thanks

-  Combined Community Codec Pack [↗](#) staff
-  Gabest for VSFilter

- squid_80 for the x64 builds of Avisynth and CSRI-enabled VSFilter
-  Firebird for some of the toolbar icons
-  Kayle for help with Win32 Vfw interface
-  ender for yawning a lot
- Alpha testers:  b0nk,  Bot1,  deathbygirl,  DoGfOoD,  grimlock,  Jagobah,  Jcubed,  Kintaro,  Kobi,  kodachrome,  maxx-,  mASSIVe,  Maya,  Mentar,  movax,  msb,  neo2sonic,  nich,  nwa,  omgifos,  SOzuken,  TheFluff,  thrash_sensei, 
Vincent,  zegnat

About

Aegisub 2 was developed using Microsoft Visual Studio 2005 and wxWidgets 2.8. The Automation module also uses the Lua, Perl and Ruby programming languages. The Regular Expressions code used in this program was written by Henry Spencer and included in wxWidgets. Some builds of Aegisub also uses the PortAudio, OpenAL, PulseAudio, libpng, zlib, Hunspell, universalchardet, CSRI, asa, libass, Freetype 2 and ffmpeg libraries.

Category: [Pages with Todo items](#)

Support

Do you want to support Aegisub? Well, it's easy!

Feedback

You can give us feedback - comments, critiques, suggestions, etc. Bug reports and feature requests are always welcome. Check out our [forums](#) and the [bug tracker](#), or stop by for a chat in the [IRC channel](#).

Contents [\[hide\]](#)

- 1 Feedback
- 2 Spreading the word
- 3 Donating
- 4 Programming

Spreading the word

Like Aegisub? Tell your friends about it! Spreading the word is a good way of helping Aegisub to be the best subtitle editor around.

Donating

Feeling generous? Consider giving us a donation! We do this in our spare time, you know.

Programming

"Given enough eyes, all bugs are shallow."

--Linus Torvalds

Feel like really helping out, or do you just have some code you want to donate? Some advice from readme.txt in the source tree:

First, some of the code is pretty readable, some is decent, and some is patched up crap. Good luck. ;) Second, if you want to code anything for Aegisub, you will need to agree to these terms:

1. You will release the patch to the public domain or give its copyright to one of the developers. This is to stop a source file from being owned by too many people. (Exception: MAJOR changes might be accepted under BSD license under your name. Consult the developers)
2. Make SURE it compiles and works fine before submitting to developers.
3. Stick to the coding standards. That is, no GNU-style indenting and crap.

Third, this is all available under the BSD license. According to GNU itself, BSD is GPL-compatible, meaning that you can link GPL code to BSD code. Keep in mind, though, that if a source file has mixed BSD and GPL content, it becomes ruled by GPL.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

FAQ

A small collection of Frequently Asked Questions about Aegisub - mostly stuff that didn't fit anywhere else.

Contents [hide]

- 1 Karaoke effects?
- 2 Can I create DVD subtitles with Aegisub?
- 3 Does Aegisub allow saving to SRT?
- 4 What's this PRS stuff?
- 5 I've found a bug!?
- 6 Why doesn't Aegisub have <feature X>? <Program Y> has it!
- 7 Where can I find more information and/or get help?
- 8 I click the "Play video" button, and the video and audio aren't synchronized at all!
- 9 Are there any VFilter bugs I should know about?

Karaoke effects?

See the [Karaoke Templater](#) tutorials.

Can I create DVD subtitles with Aegisub?

Not directly, but there's a nifty program called [MaestroSBT](#) that can convert SSA to VOBSubs. It does have quite a bunch of restrictions on what tags and other things may be used, so reading its manual first is advised. Also note that it does not accept ASS - only SSA. You can use Aegisub's File -> Export... dialog to save real SSA files.

Does Aegisub allow saving to SRT?

Yes, but only if it means that no information will be lost. In other words, if you have any override tags that aren't \1c, \b, or \i, Aegisub won't allow saving directly to SRT. However, you can still export to SRT by using the File -> Export... dialog box. Just uncheck all the checkboxes (clean script info, VFR transform etc.).

What's this PRS stuff?

Pre-Rendered Subtitles. Basically a way to "encode" ASS (text) subtitles into pictures, which can then be encoded into the video using an Avisynth filter. There's also a DirectShow renderer, but it's in early development and not ready for public use yet. PRS has a few advantages over ASS (and a few disadvantages too). One advantage is that it doesn't require much CPU power to render onto the video - all the effort is spent when "encoding" to PRS. Another is that the subtitles will look exactly as the one who rendered them intended, which is not guaranteed with ASS (not even when rendering with VFilter, as it has had, and still has, quite a few rendering bugs). The main disadvantage is that PRS takes more space than ASS and other text-based subtitle formats do.

I've found a bug!?

Report it on the [bug tracker](#). Please include as many details as possible in your report! Remember that if a bug is not on the bug tracker, it *does not exist* as far as we are

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

concerned.

Why doesn't Aegisub have <feature X>? <Program Y> has it!

Quite possibly because we didn't know you wanted it. Request it on the [bug tracker](#) and see what happens.

Where can I find more information and/or get help?

For Aegisub-related stuff, the [forums](#) and the [IRC channel](#) are good places to ask questions. The Aegisub [wiki](#) also contains some more obscure information not included in the manual for various reasons, and so does the forums. For general video-related questions, [Doom9.org](#) and [its forums](#) is generally the place to go.

I click the "Play video" button, and the video and audio aren't synchronized at all!

That's not a question.

But anyway, this is because of the way Aegisub displays video, it does not guarantee sync with the audio at all. Also, sometimes AviSynth can be quite slow at delivering frames, which would make real sync playback impossible. The "play video" feature is intended to display frames in rapid succession to check that some typesetting looks right, not as a replacement for a media player application. You can be sure that, if it sounds right when you click the "play line" button for the audio, and the subtitle looks like it's at the right frames, it is timed correctly to the audio and it is timed correctly to the video. If you don't believe this, just test it in a real media player (or hardsub it) and see for yourself.

If you absolutely must have this, feel free to implement it yourself.

Are there any VSFilter bugs I should know about?

In one word: [yes](#).

Editing Subtitles

Editing subtitles is what Aegisub is made for. This page will deal with basic text editing of subtitle lines; for more information on the typography of subtitles, see [typesetting](#). For information on the timing of subtitle lines, see [working with audio](#).

Todo: This page should probably be split into several smaller ones to make it easier to digest, easier to link, less confusing and wall-of-text and to promote going more in depth with the separate topics.

Opening subtitles

In the *File* menu, there are three menu choices that relate to opening or creating subtitles:

- **New subtitles** - Creates a new, blank script.
- **Open subtitles** - Opens an existing subtitles file or imports subtitles from a [Matroska container file](#).
- **Open subtitles with charset** - Opens subtitles but lets you choose what character set Aegisub will use to interpret the file. Usually not needed anymore, see below.

When you open a subtitles file that is not detected as Unicode, Aegisub will attempt to guess what character set it is encoded with. If it is unsure, it will ask you to choose from two or more likely alternatives. If the result looks garbled or otherwise incorrect, try reopening it with another character set.

Supported formats

Aegisub supports reading the following subtitle formats:

- Advanced Substation Alpha, also known as SSA v4+ (.ass)
- Advanced Substation Alpha v2, also known as SSA v4++ or ASS2
- Substation Alpha v4 (.ssa)
- [SubRip](#) Text (.srt)
- MPEG4 Timed Text (limited support at best; broken at worst), also known as ISO/IEC 14496-17, MPEG-4 Part 17 or just TTXT (.ttx)
- MicroDVD (.sub)
- Plain "dialog script" formatted text (see below)

Importing subtitles from MKV

Loading subtitles directly from Matroska files can also be done. The following CodecID's are supported:

- S_TEXT/UTF8 (SRT)
- S_TEXT/ASS (ASS/SSA v4+)
- S_TEXT/SSA (SSA v4)

Importing plain text scripts

Aegisub also supports importing "dialogue-formatted" plain text scripts. For example:

```

Actor 1:      Well do I understand your speech, yet few strangers do so.
              Why then do you not speak in the Common Tongue,
              as is the custom in the West, if you wish to be answered?
# TL check: The above seems to be a quote from the lord of the rings, look it up later
Actor 2:      What are you babbling about?

```

This will result in five subtitle lines, one being commented out. The first three will have the actor field set to "Actor 1", and the fifth will have it set to "Actor 2" (the comment line's actor field will be blank).

When you open a file with the extension .txt, Aegisub will ask you about what characters it should use as the actor separator and comment starter, respectively. In the example above, the actor separator is a colon (":") and the comment starter is a hash("#").

Editing subtitles

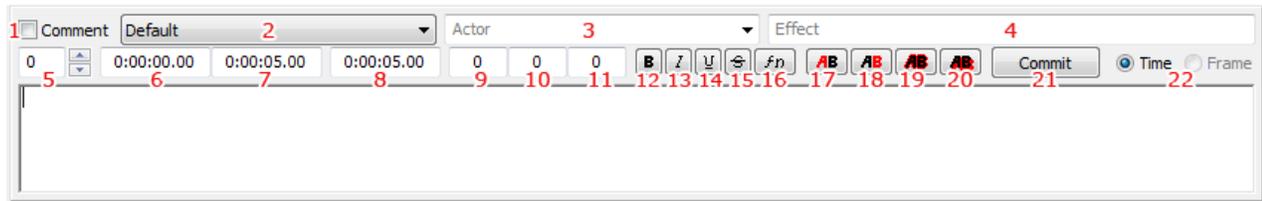
Editing subtitles in Aegisub is done in two areas: the subtitle edit box (where you type in or edit text) and the subtitles grid, where you do things to one or more lines at the same time.

The subtitles edit box

Contents [hide]

- 1 Opening subtitles
 - 1.1 Supported formats
 - 1.2 Importing subtitles from MKV
 - 1.3 Importing plain text scripts
- 2 Editing subtitles
 - 2.1 The subtitles edit box
 - 2.2 The subtitles grid

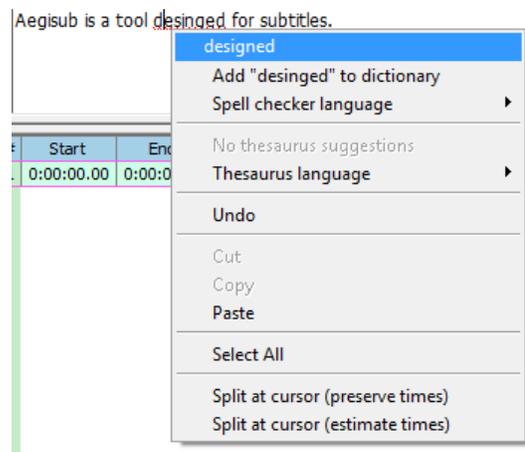
- Navigation**
 - » [Main Page](#)
- Introduction**
 - » [What is Aegisub?](#)
 - » [Highlights](#)
 - » [Credits](#)
 - » [Support Aegisub](#)
 - » [FAQ](#)
 - » [Tutorials](#)
- Working with Subtitles**
 - » [Editing Subtitles](#)
 - » [Exporting Subtitles](#)
 - » [Applying Subtitles](#)
 - » [Spell Checker](#)
 - » [Translation Assistant](#)
 - » [Paste Over](#)
 - » [Select Lines](#)
- Typesetting**
 - » [Introduction](#)
 - » [Working with Video](#)
 - » [Editing styles](#)
 - » [Visual Typesetting](#)
 - » [ASS Override Tags](#)
 - » [Colour Picker](#)
 - » [Styling Assistant](#)
 - » [Resolution Resampler](#)
 - » [Fonts Collector](#)
- Timing**
 - » [Working with Audio](#)
 - » [Shift times](#)
 - » [Timing Post-Processor](#)
 - » [Kanji Timer](#)
- Automation**
 - » [Overview](#)
 - » [Karaoke Templater](#)
 - » [Lua Reference](#)
- Miscellaneous**
 - » [Aegisub Options](#)
 - » [Script Properties](#)
 - » [Attachment Manager](#)



The edit box is just a plain editing area with a number of associated controls. They are:

1. Flags the line as a comment. Comment lines will not be displayed on the video.
2. The [style](#) used for this line.
3. The actor speaking this line. Has no actual effect on subtitle display but can be useful for editing purposes.
4. Effect for this line. Mostly for format historical reasons; it is not actually used for any rendering. Some utilities store various data in this field.
5. Layer for this line. If you override positioning with an [override tag](#) so that two or more lines are displayed on top of each other, this field controls which one is drawn where; higher layer numbers are drawn on top of lower ones.
6. Start time for the line.
7. End time for the line.
8. Duration for the line. If you modify this field, the end time will be modified as a result.
9. Left margin for this line. 0 means use the margin specified in the style.
10. Right margin for this line. 0 means use the margin specified in the style.
11. Vertical margin for this line. 0 means use the margin specified in the style.
12. Inserts a bold override tag (`\b1`) at the cursor position. If the text is already bold, inserts a corresponding closing tag (`\b0`).
13. Inserts an italics override tag (`\i1`) at the cursor position. If the text is already italic, inserts a corresponding closing tag (`\i0`).
14. Inserts an underline override tag (`\u1`) at the cursor position. If the text is already italic, inserts a corresponding closing tag (`\u0`).
15. Inserts a strikethrough override tag (`\s1`) at the cursor position. If the text is already italic, inserts a corresponding closing tag (`\s0`).
16. Brings up a font selection window and inserts a font face name tag (`\fnFontName`) with the given font name, as well as the chosen effect tags.
17. Brings up the [color picker](#) and lets you choose a color; then inserts a primary color override tag (`\c`) with the chosen color at the cursor position.
18. Brings up the [color picker](#) and lets you choose a color; then inserts a secondary color override tag (`\2c`) with the chosen color at the cursor position.
19. Brings up the [color picker](#) and lets you choose a color; then inserts an outline color override tag (`\3c`) with the chosen color at the cursor position.
20. Brings up the [color picker](#) and lets you choose a color; then inserts a shadow color override tag (`\4c`) with the chosen color at the cursor position.
21. Commits the current text in the edit box to the script. Any changes that have not been committed will be discarded when you go to another line.
22. Changes display between times and frames. Note that this does not change how times are actually stored in the script.

If you right-click anywhere in the edit box, you get the following menu:



Select all, undo, copy, cut and paste all do what you'd expect them to.

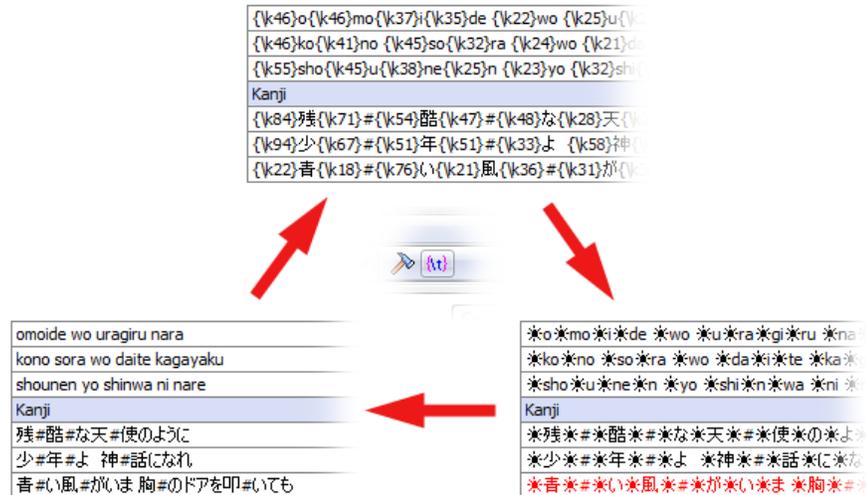
- **Spell checker:** if you right-click on a word that has been detected as misspelled, the spell checker will suggest some likely alternative. You can also set which language it will use for checking from this menu, or add words that it doesn't recognize but you know to be correctly spelled to the dictionary.
- **Thesaurus:** suggests alternative ways to write the highlighted word.
- **Split line:** splits the line into two new lines at the cursor position. Preserve times keeps the old line's timing for both lines, estimate times tries to guess where the split is based on the length of the text on each side of the cursor.

The subtitles grid

#	Start	End	Style	Text
13	0:01:18.41	0:01:21.87	Eva Op Roomaji	omoi de wo uragiru nara
14	0:01:22.13	0:01:25.63	Eva Op Roomaji	kono sora wo daite kagayaku
15	0:01:25.80	0:01:29.34	Eva Op Roomaji	shounen yo shinwa ni nare
16	0:00:00.00	0:00:00.00	Default	Kanji
17	0:00:01.03	0:00:06.77	Eva Op Kanji	残#酷#な天#使のように
18	0:00:07.01	0:00:14.48	Eva Op Kanji	少#年#よ 神#話になれ
19	0:00:22.91	0:00:29.23	Eva Op Kanji	青#い風#がいま 胸#のドアを叩#いても
20	0:00:29.68	0:00:36.93	Eva Op Kanji	私##だけをただ見つめて 微笑#んでるあなた
21	0:00:37.70	0:00:44.23	Eva Op Kanji	そっとふれるものもとめることに夢中#で
22	0:00:44.60	0:00:51.36	Eva Op Kanji	運#命#さえまだ知らないいたいけな瞳##
23	0:00:52.26	0:00:58.34	Eva Op Kanji	だけどこいつか気付くでしょう その背中#には
24	0:00:59.75	0:01:06.26	Eva Op Kanji	運#か未来# あざすための 羽根#があること
25	0:01:07.14	0:01:10.70	Eva Op Kanji	残#酷#な天#使のテーゼ
26	0:01:10.94	0:01:14.43	Eva Op Kanji	窓#辺からやがて飛び立つ
27	0:01:14.65	0:01:18.07	Eva Op Kanji	ほとばしる熱#いリフトで
28	0:01:18.41	0:01:21.87	Eva Op Kanji	思#い出を裏#切るなら
29	0:01:22.13	0:01:25.63	Eva Op Kanji	この宇宙を抱いて輝##く
30	0:01:25.80	0:01:29.34	Eva Op Kanji	少#年#よ 神#話になれ
31	0:00:00.00	0:00:00.00	Default	Translation
32	0:00:01.03	0:00:06.77	Eva Op English	Just like the Cruel Angel,
33	0:00:07.01	0:00:14.48	Eva Op English	Young boy, rise as a legend!
34	0:00:22.91	0:00:29.23	Eva Op English	Even as the tender wind knocks at the door to your heart,
35	0:00:29.68	0:00:36.93	Eva Op English	You merely look at me and smile.

The subtitles grid shows all lines (comments and otherwise) in the entire file. Some common controls:

- To move lines up or down in the grid, select them, hold down the Alt key and press the up- or down-arrow keys.
- To select multiple lines, hold down either Ctrl or Shift and click. Ctrl-click selects one more line per click; Shift-click selects all the lines between the first clicked and the last clicked.
- To sort all lines in the grid by their start time, open the *Timing* menu and click "Sort by time".
- To change the way *override tags* are displayed in the grid, click the "cycle through tag hiding modes" button on the toolbar.



The lines have different (configurable) colors representing different things; see the [subtitles grid](#) section of the options page for details on what the colors mean.

By default, the following columns are visible:

- # - The line number.
- Start - The start time of the line.
- End - The end time of the line.
- Style - The style used for this line.
- Text - The text of the line (this is what will be displayed on the video).

The following columns will be displayed if any line in the script uses them:

- L - The layer of the line (see above).
- Actor - The actor speaking the line.
- Effect - The effect for this line (not actually used for rendering).
- Left - The left margin.
- Right - The right margin.
- Vert - The vertical margin.

You can also right-click the top line of the grid (the one with the column names) to manually select which columns you want to be visible.

Right-clicking any other line in the grid gives you the following menu (many of the options are also available in other menus):

Insert (before)
Insert (after)
Insert at video time (before)
Insert at video time (after)
Duplicate
Duplicate and shift by 1 frame
Split (by karaoke)
Swap
Join (concatenate)
Join (keep first)
Join (as Karaoke)
Make times continuous (change start)
Make times continuous (change end)
Recombine Lines
Create audio clip
Copy
Cut
Paste
Delete

- **Insert (before/after)** - Inserts a new empty line before or after the selected line. The new line will be timed start at 0:00:00.00 and go to 0:00:05.00.
- **Insert at video time (before/after)** - Same as the above, but the new line will be timed to start at the current video frame. Not enabled unless you have video loaded.
- **Duplicate** - Duplicates the selected line(s).
- **Duplicate and shift by 1 frame** - Duplicates the selected line(s) and shifts the start and end times one frame forward. Useful for frame-by-frame typesetting. Not enabled unless you have video loaded.
- **Split (by karaoke)** - Splits the line into one new line per syllable, as delimited by karaoke override tags (\k and its relatives). The timing of the first line will start at the original line's start time and end at that time plus the length of the first syllable; the following lines will start at the end of the previous and last for the duration of the syllable.
- **Swap** - Swaps the places (in the grid) of two selected lines.
- **Join (keep first)** - Joins two or more lines, discarding the text of all but the first. The new line will be timed to start at the first line's start time and end at the last line's end time. Only enabled if you have more than one line selected.
- **Join (concatenate)** - Same as above, but concatenates the text of all selected lines instead. A newline override (\n) is inserted between the texts of each source line.
- **Join (as karaoke)** - Does the inverse of *Split (by karaoke)*, i.e. the same as *Join (concatenate)* but inserts \k tags with the timing of each source line in the joined line.
- **Make times continuous (change start/change end)** - Modifies the timing of the selected lines so that the end time of each line is the same as the start time of the next line. Change start/change end determines whether the function changes the end time or the start time of each line. Not enabled unless you have more than one line selected.
- **Recombine lines** - Given two or more lines with the same text being partially present in all of them, creates one line per text fragment instead. This is mostly useful for correcting subs ripped from DVD's, which frequently look something like this:

31	0:02:35.02	0:02:36.01	Style1	It's no use.
32	0:02:37.05	0:02:39.35	Style1	I shouldn't have come here after all.
33	0:02:43.22	0:02:43.66	Style1	To Shinji\N'I'll be coming\nto get you,\nso wait for me.\NAttention here please!!
34	0:02:43.66	0:02:45.66	Style1	To Shinji\N'I'll be coming\nto get you,\nso wait for me.\NAttention here please!!\NWell, I guess we won't be meeting here.
35	0:02:45.66	0:02:46.02	Style1	Well, I guess we won't be meeting here.
36	0:02:47.16	0:02:49.76	Style1	Can't be helped. I'll go to a shelter.
37	0:03:15.02	0:03:15.42	Style1	Estimated Path

After recombine lines, the result is:

31	0:02:35.02	0:02:36.01	Style1	It's no use.
32	0:02:37.05	0:02:39.35	Style1	I shouldn't have come here after all.
33	0:02:43.22	0:02:45.66	Style1	To Shinji\N'I'll be coming\nto get you,\nso wait for me.\NAttention here please!!
34	0:02:43.66	0:02:46.02	Style1	Well, I guess we won't be meeting here.
35	0:02:47.16	0:02:49.76	Style1	Can't be helped. I'll go to a shelter.
36	0:03:15.02	0:03:15.42	Style1	Estimated Path

- **Create audio clip** - Saves a segment of the loaded audio corresponding to the timing of the selected lines (starting at the earliest start time and ending at the latest end time) as an uncompressed WAV file. Only enabled if you have audio loaded.
- **Cut/Copy/Paste** - Cuts/copies/pastes entire lines. Note that the lines are copied as plain text and can be copied and pasted freely between text editors, chat programs, web browsers, other instances of Aegisub etc etc.
- **Delete** - Deletes the selected lines.

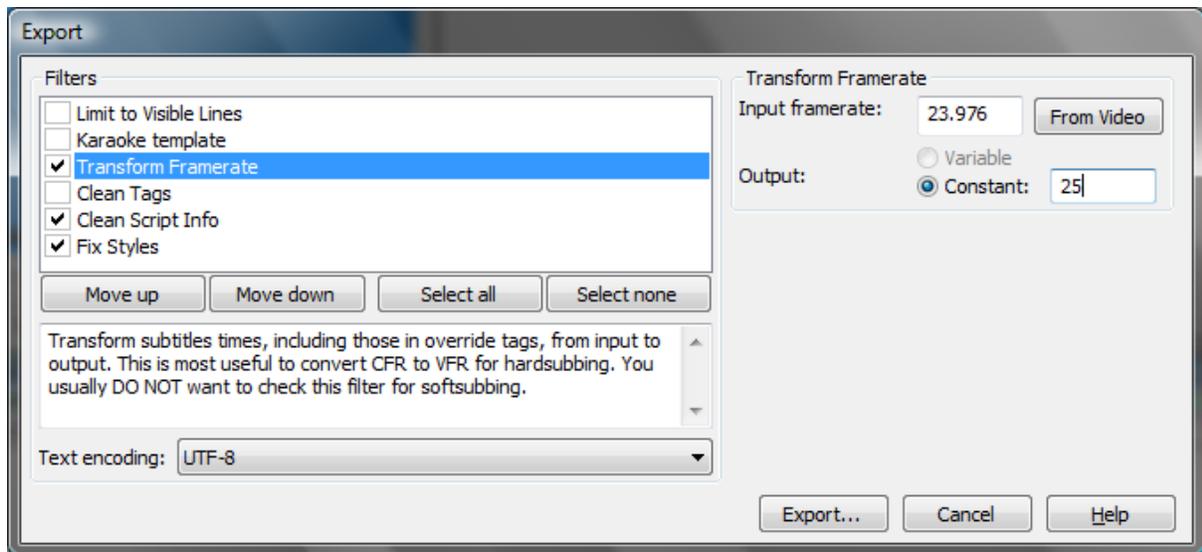
Exporting

In addition to the normal "Save" and "Save as" functions, Aegisub also has an "Export" feature, which can transform the entire script through various export filters. This is used for a number of things ranging from framerate conversions over karaoke effect generation to just saving to other formats and/or character sets.

The export dialog

Contents [hide]

- 1 The export dialog
- 2 Filters
 - 2.1 Limit to visible lines
 - 2.2 Karaoke template
 - 2.3 Transform framerate
 - 2.4 Clean tags
 - 2.5 Clean script info
 - 2.6 Fix styles



The upper half of the window contains the available filters. Ticking one or more of them will apply them in the order they are listed; use the move up/down buttons to change the order. The lower half contains a brief description of the selected filter.

Some filters have configuration parameters; those that do will extend the window to the right and put any settings controls there.

The dropdown menu at the bottom controls what text encoding will be used for the exported file. May be useful for exporting to legacy programs that do not support Unicode.

When you click the "Export" button, do note that you can choose other formats than ASS to save to. Also note that this will almost always mean that a lot of formatting tags will be stripped.

Filters

The following filters are available in the default installation:

Limit to visible lines

Exports only the lines that are currently visible on the active video frame. Does nothing if you do not have video loaded. Script headers and styles etc. are also exported.

- Navigation
 - » Main Page
- Introduction
 - » What is Aegisub?
 - » Highlights
 - » Credits
 - » Support Aegisub
 - » FAQ
 - » Tutorials
- Working with Subtitles
 - » Editing Subtitles
 - » Exporting Subtitles
 - » Applying Subtitles
 - » Spell Checker
 - » Translation Assistant
 - » Paste Over
 - » Select Lines
- Typesetting
 - » Introduction
 - » Working with Video
 - » Editing styles
 - » Visual Typesetting
 - » ASS Override Tags
 - » Colour Picker
 - » Styling Assistant
 - » Resolution Resampler
 - » Fonts Collector
- Timing
 - » Working with Audio
 - » Shift times
 - » Timing Post-Processor
 - » Kanji Timer
- Automation
 - » Overview
 - » Karaoke Templater
 - » Lua Reference
- Miscellaneous
 - » Aegisub Options
 - » Script Properties
 - » Attachment Manager

Karaoke template

Filters the script through the automation script "karaoke templater" to generate karaoke effects. See the [karaoke templater](#) and [automation overview](#) pages for further details.

Transform framerate

In "constant" output mode, recalculates every single timestamp in the script (including ones contained in [override tags](#)) to work with a new framerate. Note that this means that the entire script will be "sped up" or "slowed down". Can be used for NTSC->PAL conversions or vice versa.

In "variable" output mode, uses the framerate of the loaded video (or the specified one, if different from the video's) and the loaded timecodes to recalculate every single timestamp in the script so that the exported subtitles can be hardsubbed on the loaded video and still sync up after timecodes are taken into account. Does nothing if you do not have timecodes loaded. See [variable framerate video](#) for further details.

Clean tags

Filters the script through the "clean tags" automation script, which tries to clean up the override tag blocks by concatenating adjacent blocks and removing redundant tags (more specifically, the second instance of tags that can only be specified once per line).

Clean script info

Cleans the script headers by removing all lines that are not absolutely essential for proper display of the script. If you're paranoid, you should consider using this for scripts that you plan on distributing in the original form, since Aegisub stores things like the path to the last opened video/audio etc. in the script headers.

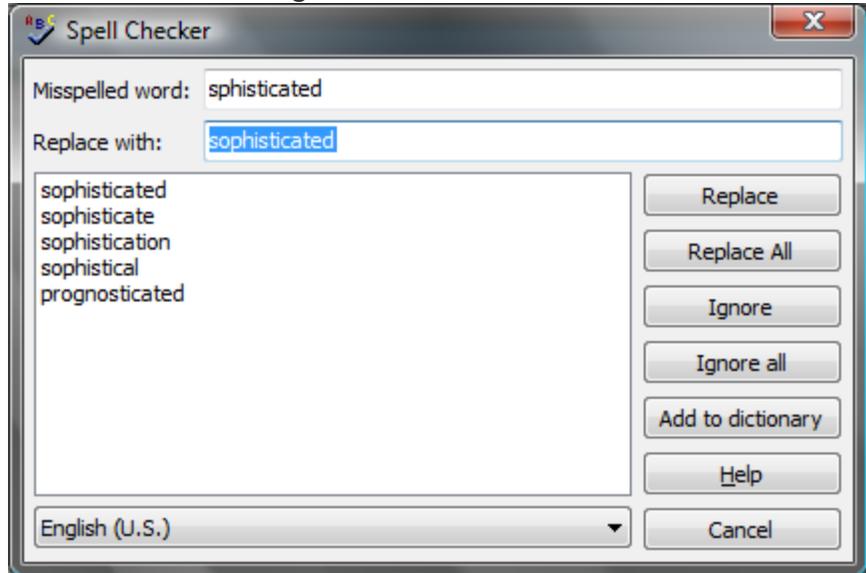
Fix styles

Goes through all the lines of the script and checks what style they use; any lines that use a style that isn't available in the current script is replaced with Default.

Spell Checker

To help you with quality assurance of your subtitle scripts, Aegisub features a complete spell checker. It uses the Hunspell library, which is based on OpenOffice.org's MySpell, which means it's at least as good as one you'd find in a sophisticated word processing program. It can be found in the *Subtitles* menu -> *Spell Checker*. It also spell-checks as you type and can to some degree be controlled from the subtitle edit box's right-click menu (see [editing subtitles](#)).

The spell checker dialog



The spell checker dialog will go through all misspelled words it can find in your script. The area in the lower left of the box displays suggestions to replace the misspelled word with; to the right there are a number of buttons:

- **Replace** - replaces the found misspelled word with the chosen replacement.
- **Replace All** - replaces *all instances* of the found misspelled word *in the entire script* with the chosen replacement.
- **Ignore** - ignores *this instance* of the misspelled word.
- **Ignore All** - ignores *all instances* of this word.
- **Add to dictionary** - adds the found word to the dictionary so it will not be detected as misspelled again.

At the bottom of the dialog box there is a dropdown box to choose the spell checker language.

Dictionaries

Aegisub comes with a whole library of dictionaries, but don't lose hope if your language isn't included with the default installation. You can download more dictionaries at:

<http://wiki.mozilla.org/L10n:Dictionaries>

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

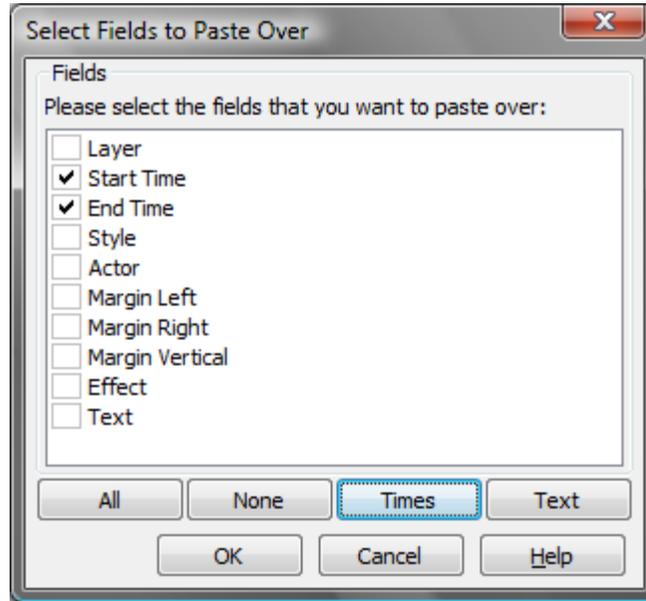
Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Paste Over

The **paste over** tool is a "smart" variant of the good old Ctrl-c Ctrl-v combo. It allows you to paste any given field(s) of one or more lines to the corresponding fields of other lines, without affecting the other fields. It can be found under the *Edit* menu -> *Paste lines over*.

To use it, first copy one or more lines with the familiar copy function (right-click -> copy, or Ctrl-c). Then select the line(s) you want to paste to, and click *paste lines over* (or press Ctrl-Shift-v). You will then get a dialog box that looks like this:



Tick the fields you want to paste (the buttons below the checkboxes can be used to select all fields (in which case paste over is almost the same as normal pasting except it replaces existing lines instead of inserting new ones), no fields (in which case it does nothing), just the text field, or both time fields) and press OK. The fields you selected will then be pasted. It's as simple as that.

Do note that the paste over function isn't limited to using lines copied from Aegisub itself; you are free to copy correctly formatted lines from any text editor, chat program, web browser etc. etc.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

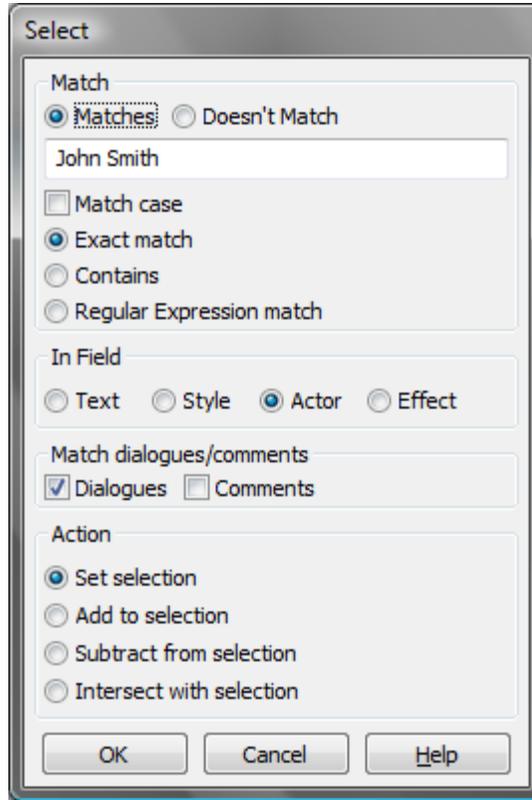
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Select Lines

The `select lines` tool is useful for finding and selecting all lines in the script that matches certain criteria. This can be useful for a number of things ranging from deleting all comment lines to finding all lines said by a certain actor. The tool can be found in the *Subtitles* menu -> *Select lines*.



Match

These controls what criteria will be used for selecting the lines.

- **Match** or **Doesn't match** - Controls whether a "positive" or "negative" matching is done. If you use *match*, all lines that does match the criteria will be considered matching. If you use *doesn't match*, all lines that does *not* match the criteria will be considered matching.
- **Match case** - Controls case sensitivity, i.e. if the tool should care about the difference between lower and upper case or not.
- **Exact match** - Matches the lines that matches the given string *exactly*.
- **Contains** - Matches the lines that *contains* the given string.
- **Regular Expression match** - Treats the string as a [regular expression](#) or "regex", and matches if said expression matches. For a regular expressions tutorial, try reading the [perlretut manual page](#) or googling. For a reference of the exact syntax supported by Aegisub, see the [wxWidgets regular expressions reference page](#).

In field

This option controls what field of each line will be used for the matching mentioned above. Possible alternatives are:

Contents [hide]

- 1 Match
- 2 In field
- 3 Match dialogues/comments
- 4 Action

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

- **Text** - the main text field
- **Style** - the style field
- **Actor** - the actor field
- **Effect** - the effect field

Match dialogues/comments

Here, you can choose if you want to select from comment lines, dialogue lines or both.

Action

Decides what the tool should do with the lines that matches the given criteria. You can choose between:

- **Set selection** - Your current selection will be discarded and all lines in the script matching the criteria will be selected instead.
- **Add to selection** - Adds all lines in the script that matches the criteria to your current selection.
- **Subtract from selection** - Deselects all currently selected lines that match the criteria.
- **Intersect with selection** - Does the inverse of subtract from selection. That is to say, all lines in the current selection that matches the criteria are kept selected, but everything else is deselected.

Typesetting

Wikipedia's definition of [typesetting](#) [↗]: "Typesetting involves the presentation of textual material in graphic form on paper or some other medium." In a subtitling context, this means three things:

- Presenting (a translation of) the spoken dialogue to the viewer in an easily readable and visually pleasing form.
- Translating any foreign-language background texts (such as episode titles, scene information like time/place and other plot-important writings on things in the background) and presenting the translation to the viewer, preferably so that it appears that the translation was there all along (yet retaining readability).
- Designing other texts (logotypes, karaoke, etc. etc.) that should be displayed on the video.

Particularly in the fansub community, the word "typesetting" usually refers to the second or third of these three.

This page will attempt to give you an introduction to subtitle typesetting. Unfortunately good taste is hard to teach, but we will do our best.

Dialogue typesetting

The main goal of dialogue typesetting is **legibility**; everything else is just icing on the cake. There's no point in having subtitles if the viewer can't read them. Remember that subtitles frequently require quite fast reading, so ease of recognition is vital. Some guidelines for good legibility:

- Use a plain, uncluttered, no-nonsense typeface. Sans-serif fonts (or if you insist on serifs, "slab serif" fonts) are generally preferable, especially at low video resolutions, since the serifs tend get blurred out, which naturally looks bad. Too thin fonts are also to be avoided. Small caps fonts are bad because the ascenders/descenders of help recognition of words. Using tried and trusted faces such as Helvetica, Arial or Verdana is not a bad thing; if you want something that looks less "boring" but still readable try a sans-serif font of the "humanist" variety, such as Calibri.
- Use a well-defined but not too thick border. The color is important; the higher the contrast against the main color the better. The white main color/black border combo is well tried and trusted. Shadow is optional; if you use it consider setting it to half-transparent black; solid black tends to hurt legibility.
- Use big friendly letters with comfortable margins; usually subtitles need to be bigger than you think to be easily readable, particularly on a TV screen (since the viewing distance on a TV is usually 2-3 meters, compared to 0.3-0.6m for a computer screen). If you plan on showing your subtitles on a TV you also need to take [overscan](#) [↗] into account; Aegisub has an overscan mask feature (see [working with video](#)) that can help you with this. Also consider the aspect ratio of the video; 16:9 or wider aspect ratios give you the opportunity to have longer lines at the expense of vertical space.
- Make sure that no more than two lines of text are visible at a time (exceptions can be made at times where there are a lot of people talking at the same time). Sometimes you can compress the text horizontally a bit to squeeze it in; at other times you just have to split the text block into two different lines and display one after the other.

Todo: examples

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

"Sign" typesetting

Sign typesetting (translating various background text) can usually be accomplished with ASS, but commercial programs like Adobe AfterEffects are becoming an increasingly popular alternative, since the holy grail of sign typesetting is to make the subtitle look like it's a part of the image that was always there. How to get there from here will not be discussed in detail on this page (see the typesetting tutorial), but here are some pointers:

- Is the sign important to the plot? You shouldn't try to go around translating every single sign there is, or you'll go insane (and the end result will probably be unreadable anyway).
- Frequently, the text of a plot-important sign (such as a letter) is actually read out loud in the dialogue; if this is the case you should consider skipping the sign typeset because having both text and dialogue may overload the viewer with information.
- Will you be able to make the translation easily legible? For example, if you have very little space available to squeeze it in on, consider putting a translation note as a normal "toptitle" instead.

Todo: images

Further reading

The Wikipedia page on [typography](#) has a lot of useful links and explanation of various typesetting-related terms.

Category: [Pages with Todo items](#)

Video

Aegisub isn't a video (or media) player, but it still supports loading and working with video in various ways. This page is a reference of how Aegisub's loading and handling of video works; for a gentler introduction on how to just get things to display properly you may want to read the [Typesetting Tutorial](#) or [Visual Typesetting](#) pages.

Opening video

To load a video file, go to the *Video* menu and press *Open video file*. What video files you can open depends on your [video provider](#). To use a dummy (mockup, blank) video, press *Use dummy video*.

Supported formats: Windows

On Windows, your video provider is *Avisynth* by default, which in practice means Aegisub will try its best to load your video in the most reliable way possible. There are currently no less than six Avisynth functions it can try:

- **Import()** - Avisynth builtin function. Used to load Avisynth scripts, never used for anything else.
- **AviSource()** - Avisynth builtin function. Only works on AVI files, obviously. Requires a working Video for Windows (VFW) decoder to open the video. Tried first for .avi files, never for anything else. If it fails, Aegisub will try `ffmpegSource()` instead, if that fails try `DSS2()`, and if that too fails try `DirectShowSource()` (see below).
- **MPEG2Source()** - Only used to load .d2v files (DVD2AVI project files; otherwise known as indexed .VOB's ripped from DVD's). Tries to use neuron2's [DGDecode](#) to open the file; if Aegisub can't find or load that it'll try the old `mpeg2dec3.dll` instead; if that too fails, returns an error.
- **ffmpegSource()** - Tries to use Myrsloik's [ffmpegsource plugin](#) (comes with the default Aegisub install) to try to load the file; see its manual for information on what formats and codecs are supported and what problems they may suffer from. Tried first for anything that isn't .avs, .avi or .d2v. If Aegisub can't find it or it returns an error, falls back on `DSS2()` or if that isn't available, `DirectShowSource()`. Note that for VFR videos that are not in the Matroska (MKV) container, you will have to extract and load timecodes yourself or subtitle synchronization will be off! See the section about [VFR](#) below.
- **DSS2()** - Tries to find, load and use Haali's `DirectShowSource2` plugin (comes with the [Haali Media Splitter](#) package and with the [CCCP](#); the Avisynth plugin is `avss.dll`, you need to manually put it somewhere where Aegisub or Avisynth can find it, see below). That will in turn try to use your computer's `DirectShow` environment to open the file. Theoretically any file renderable by `DirectShow`

Contents [hide]

- 1 Opening video
 - 1.1 Supported formats: Windows
 - 1.2 Supported formats: Unix
 - 1.3 Dummy video
- 2 Playing video
 - 2.1 Before you press the "play" button
 - 2.2 Video playing quirks
- 3 Keyframes
 - 3.1 Loading and saving of keyframe data
 - 3.2 Keyframe file specification
- 4 Variable framerate video
 - 4.1 Timecodes
 - 4.2 VFR and hardsubbing
- 5 Anamorphic video
 - 5.1 Image stretching and subtitle rendering
 - 5.2 Compensating for the stretching
 - 5.3 Caveats
 - 5.4 Further reading
- 6 The video menu
 - 6.1 Sourcefile-related
 - 6.2 Timecodes-related
 - 6.3 Keyframes-related
 - 6.4 Display-related
 - 6.5 Seeking-related

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

should be supported, but since you are at the mercy of third-party splitters and decoders, your mileage may vary. As a rule of thumb, if it's playable in Windows Media Player, Aegisub should be able to load it. Do note that DSS2 converts variable framerate files to constant. Usually this is what you want and expect but if you're doing proper VFR subtitles, it isn't.

- **DirectShowSource()** - Uses `DirectShowSource()` (which ships with Avisynth) to try to load the file. Basically the same as DSS2, with the exception that it doesn't convert VFR to CFR. **Warning:** DSS is known to have problems with frame-accurate seeking. Do not use it if you can avoid it.

Aegisub will look for Avisynth plugins in its `?data` directory (generally the folder where `Aegisub.exe` is, on Windows). You can also put them directly in your Avisynth plugins folder to get them autoloaded.

Anyway, what does all this mean to you as the average user? It means Aegisub can in theory open ANY video format as long as you are capable of playing it in Windows Media Player (or render it in GraphEdit), plus some that aren't playable there (such as `.d2v`'s). At least it'll try its best to load it. You may however run into trouble with some of the more unusual formats and just because you CAN load something it doesn't mean it's a good idea to use it. Formats more or less guaranteed to work in a stable fashion with the default install include but is not limited to:

- XviD/DivX/other MPEG4 ASP codecs in AVI
- Most stuff stored in Matroska (except maybe some really oddball `V_MS/VFW/FOURCC` streams)
- Pretty much any standards-compliant MP4

If you change the video provider to `dshow`, Aegisub will try to interface directly with DirectShow to open the file. May help with some really oddball files, but is usually not recommended.

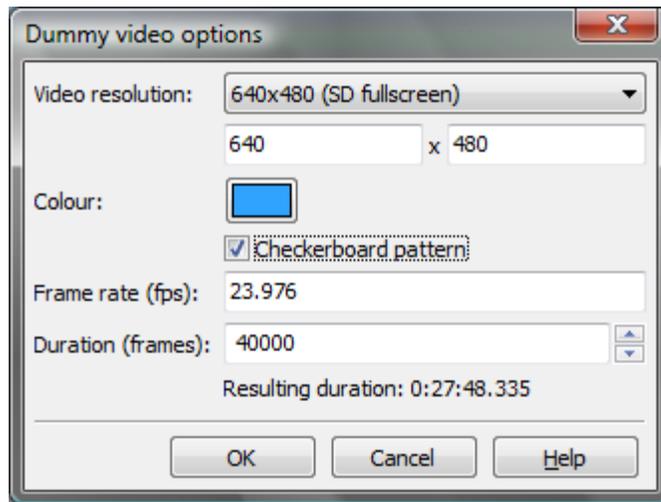
Supported formats: Unix

Under Unix or Unix-like systems (including but not limited to *BSD, GNU/Linux, and MacOS X) your video provider is `ffmpeg`, which means Aegisub will attempt to load the video using, well, `ffmpeg`, which should mean that anything supported by `ffmpeg` is supported; see its documentation for details. There are a few quirks however; for example `libavformat` is not used for Matroska (`.mkv` `.mka` `.mks` etc) files since its Matroska splitter has a lot of issues; instead Aegisub does its own parsing of such files (by using code borrowed from Haali's Matroska parser).

At the time of writing the `ffmpeg` interface isn't 100% stable (much like Aegisub in its entirety isn't 100% stable under Unix yet) but at least most AVI's, MKV's and MP4's seem to work in a decently reliable fashion.

Dummy video

Aegisub also supports providing a bogus video surface for you to render subtitles on without having any actual video loaded. The dummy video dialog box looks like this:



- **Resolution**

The resolution of the dummy video in pixels. There are a few pre-defined choices in the dropdown menu; you can also specify the size yourself. Note that since the dummy video is RGB there are no particular restrictions with regards to odd widths/heights.

- **Colour**

The colour of the video surface. The color is solid by default; tick the box "checkerboard pattern" if you want some more variation.

- **Frame rate**

Determines how long each frame is displayed by setting the number of frames per second (fps). Note that it is in fact possible to load VFR timecodes with dummy video.

- **Duration**

The duration of the video in frames. Below this number is shown the resulting duration in hours/minutes/seconds.

Playing video

Aegisub does support playing back video, but it has been widely debated if it is a misfeature or not.

Before you press the "play" button

Think it through. Do you really want to play the video? (Hint: the answer is "no", you don't want to do that, at least not in Aegisub.) If you're trying to check if a subtitle matches up to something in the video, wouldn't it be easier to just step through the video frame-by-frame with the arrow keys? If you're proofwatching, wouldn't it be more convenient to do so in a real media player?

To conclude: in real world situations, you will never need to play the video from inside Aegisub. The mantra is: Aegisub isn't a media player; if you want a media player use a media player.

Video playing quirks

When you play video, Aegisub will try to play the audio along with the video, even if you haven't explicitly loaded it. This feature is currently somewhat unreliable, especially on non-Windows systems, so if you *really* want to play video and audio, pre-load the audio first (see [working with audio](#)).

Note that Aegisub does **NOT** guarantee audio/video synchronization. It does attempt it, but since the video rendering plus subtitle drawing is slower than in a real media player your computer may not be fast enough to keep up. Even if it *is* fast enough, you will probably still not get perfect synch. If you want that, use a media player.

Keyframes

A brief, highly simplified explanation of what a keyframe is, for the people who missed Video 101:

Modern video codecs compress video by storing as little information as possible. Instead of storing all frames in full (like a large amount of JPG pictures), they take occasionally take a keyframe, which is stored fully, and for each frame after that until the next keyframe, they only store what changed in the picture since the last frame. Hence, to seek to a given frame the decoder needs to find the previous keyframe and decode all frames between that frame and the sought one, which means seeking to keyframes is faster than to other frames. Keyframes are interesting mostly for scene timing reasons, because they usually appear at scenechanges. The first frame in a scene is almost invariably a keyframe, but beware - far from every keyframe is a scenechange! Most encoders insert a keyframe at least once every 250-300 frames, even if there hasn't been a scenechange.

Aegisub uses keyframe data at several places in the program. They are drawn as black lines on the video seek bar, and (by default) also displayed in the audio waveform/spectrum analyzer.

Loading and saving of keyframe data

Aegisub supports loading keyframe data directly from AVI and Matroska files, regardless of what video provider was used to open them. If you are using any video format other than AVI or MKV, you can still load keyframe data separately. Aegisub currently supports reading from two formats; its own keyframe file format (see specification below) and XviD first-pass .stat files. If you don't know what the latter is, ask your local encoding guru.

Aegisub can also write the currently loaded keyframe data to a keyframes file, which can be useful to avoid passing video files around in certain situations (audio timing for example).

Keyframe file specification

A keyframes file is a plain ASCII-encoded text file; both `\n` and `\r\n` is understood to mean a linebreak. Syntax example:

```
# keyframe format v1
fps 23.976000
0
30
70
```

```
82
130
131
```

The first line is the format specification; the string "`# keyframe format v1`". The next line consists of the lower-case string "`fps`" followed by the the FPS of the video file the keyframe data was generated from as a double-precision floating point number ("`.`", `0x2E`, full stop, is used as decimal separator regardless of system locale). Finally, after the `fps` line follows an arbitrary number of long integers (one per line), each representing the frame number of a keyframe. The frame numbers are zero-indexed; i.e. the first frame of the video is frame number 0.

Variable framerate video

Aegisub supports loading and working with variable framerate video. The how's and why's of VFR is far outside the scope of this manual (see for example [the VFR thread on AnimeSuki](#) or the [Avisynth manual page](#) for more information about VFR), but we will cover what you need to know about how Aegisub handles it.

Timecodes

Loading of Matroska timecode files (v1 and v2) is supported and useful if you have a VFRaC (Variable FrameRate assumed Constant; for example the frames of an VFR MKV stored in an AVI at a constant FPS, see the VFR thread linked above for details) video file but want the subs to sync to it. If you load a Matroska file, Aegisub will automatically read the timecodes from it directly. Do however note that this is **NOT** supported for any other potentially VFR formats like MP4 or WMV; for those formats you **MUST** demux and load timecodes yourself or the audio/video/subtitle synchronization will be wrong. Be aware that WMV in particular is frequently VFR even if you do not expect it.

VFR and hardsubbing

As long as you're not encoding with GDSMux, the encoding environment your subtitling filter is working in (i.e. Avisynth, VirtualDub or mencoder) will assume that the world is CFR, and hence the input file is VFRaC. This is obviously a problem since it messes up subtitle synchronization. Aegisub provides a way around this through the Framerate Transformation export filter, which takes the framerate of the VFRaC input file and the timecodes, and then changes every time code and override tag in the script so it can be hardsubbed on the VFRaC video and still sync up perfectly after timecodes are applied. To prepare a script for hardsubbing, make sure you have the timecodes loaded, then go to the *File* menu and press *Export*. Untick everything except the *Transform Framerate* filter. Choose the *Variable* output mode. It needs to know the FPS of the video you're going to hardsub the subtitles on; if you have video loaded Aegisub will assume that's it and insert it in the box for you. **Note:** if you have a Matroska or other VFR file loaded, remember that the FPS value Aegisub reports might not be the same as the one of the video you are going to hardsub the subtitles on.

Anamorphic video

OR: How to get your subtitles to not look stretched

The following paragraphs assumes that you have a working knowledge of what anamorphic video is and how aspect ratios work. If you feel unsure, you may want to



consult a [gentle but fairly complete introduction](#) .

Image stretching and subtitle rendering

An anamorphic image needs to be stretched to the proper aspect ratio before it is presented to the viewer. On a computer, this is usually done by the video renderer, and thereby lies a problem. Most subtitle renderers (including VSFILTER, the current "standard" renderer) do the subtitle drawing on the image before it is stretched, and since the renderer is not aware of any aspect ratio issues, the subtitles will be stretched along with the image when the video is displayed to the viewer, and therefore they will look stretched. Aegisub does its subtitle rendering this way as well (since that way it will be WYSIWYG with most players); you can test how it looks by using the "Override aspect ratio" in the video menu.

Compensating for the stretching

Fortunately it's easy to compensate for the stretching, since you know by how much the image will be stretched (since you know its original dimensions and the display aspect ratio). You just calculate how many percent the image will be stretched in either the X or the Y direction, and then set the ScaleX or ScaleY parameter in the [style](#) (or use the `\fscx` or `\fscy` [overrides](#)) to the same amount but in the other direction.

Example: we have a 704x480 image that we know will be displayed as 16:9 (or 1.7777...:1). If we assume that the player will stretch the width but leave the height untouched, this means that the new width will be:

$$(16 / 9) * 480 = 853.333...$$

which in percent is:

$$853.333... / 704 = 1.212121...$$

i.e. 121%. Hence, to compensate for this horizontal (X-direction) stretch we set ScaleY in all our styles to 121% to stretch the subtitles by the same amount, and after rendering they now no longer look stretched.

OR we can do the stretching in the other direction, assuming that the player changes the height of the image instead. Assuming the same image:

$$704 / (16 / 9) = 396$$

which corresponds to:

$$396 / 480 = 0.825$$

or 82.5%, which means we can compensate for the vertical (Y) compression by setting ScaleX to 82.5%.

Caveats

Both of the above methods give the subtitles the proper aspect ratio, but depending on how the player does the stretching, you may get subtle variations in subtitle size. Aegisub (and indeed most video players and renderers) always changes the image width and never the height, except if you pick the "custom" aspect ratio and specify a resolution. Note that if you use the Matroska container, you may specify the display resolution directly, but player support of this varies.

Note that some odd subtitle renderers (most notably Media Player Classic's builtin renderer) are actually a part of the video renderer and will do the subtitle rendering *after* anamorphic stretching, resulting in stretched subtitles and much annoyance. With MPlayer's libass renderer, you can actually specify whether subtitles should be drawn before or after stretching by using the `-vf` parameter to move the subtitle renderer in the filter chain.

Further reading

For more information about anamorphic video and aspect ratios in general (a deeply obfuscated topic that appears simple at a glance) the following links may be of interest:

- [A Quick Guide to Digital Video Resolution and Aspect Ratio Conversions](#)  - Absolutely essential reading for anyone who wants to really understand the topic, but unfortunately far more than everything most people ever wanted to know about it.
- [Widescreen.org: Aspect Ratios](#)  - the history and reasons for some common aspect ratios
- [Wikipedia: Aspect Ratio \(image\)](#) 
- [Wikipedia: Anamorphic Widescreen](#) 

The video menu

The following options are available from the video menu:

Sourcefile-related

- **Open video**

Opens video. Note that if you try to open another video with video already loaded, the original video will be closed first.

- **Close video**

Unloads the currently open video.

- **Recent**

Shows list of recently opened videos.

- **Use dummy video**

Opens a dummy video (see above).

- **Show video details**

Shows some information about the currently open video. Details shown are filename, frames per second (for VFR files the average FPS is displayed), resolution and aspect ratio, length and decoder. Decoder being what filter/method Aegisub used to open the file.

Timecodes-related

- **Open timecodes file**

Loads a timecodes file and applies it to the video, changing the video/subtitle

synchronization.

- **Save timecodes file**

Saves the currently loaded timecodes as a new v2 timecodes file.

- **Close timecodes file**

Unloads the currently loaded timecodes.

- **Recent**

Shows list of recently opened timecodes files.

Keyframes-related

- **Open keyframes**

Loads keyframe data from the given file. If you already have keyframe data loaded, it will be replaced by the one read from the file.

- **Save keyframes**

Saves the currently loaded keyframes data to a keyframes file.

- **Close keyframes**

Unloads the currently loaded keyframe data, if any. Note that it is not possible to unload keyframe data that was directly loaded from the video file; if you want to get rid of that for some reason, load a keyframes file with only frame 0 being marked as a keyframe.

- **Recent**

Shows list of recently loaded keyframe files.

Display-related

- **Detach video**

Detaches the video display and related controls from the Aegisub main window and moves it into its own window. To reattach the video to the main window, close the detached window. This feature can be particularly useful on multi-monitor setups.

- **Set zoom**

Sets the video zoom level.

- **Override aspect ratio**

Stretches the video to the specified aspect ratio by modifying the video width. Useful for anamorphic video (see above).

- **Show overscan mask**

Draws a blue "mask" over the image, indicating the edges of the action safe (dark blue) and title safe (light blue) areas. Useful if you plan on showing your subtitles on a TV without adjustable overscan correction. For more information, see the Wikipedia pages on [overscan](#), [safe areas](#) and [overscan amounts](#). Aegisub follows the [BBC guidelines](#) on how big the safe areas should be.

Seeking-related

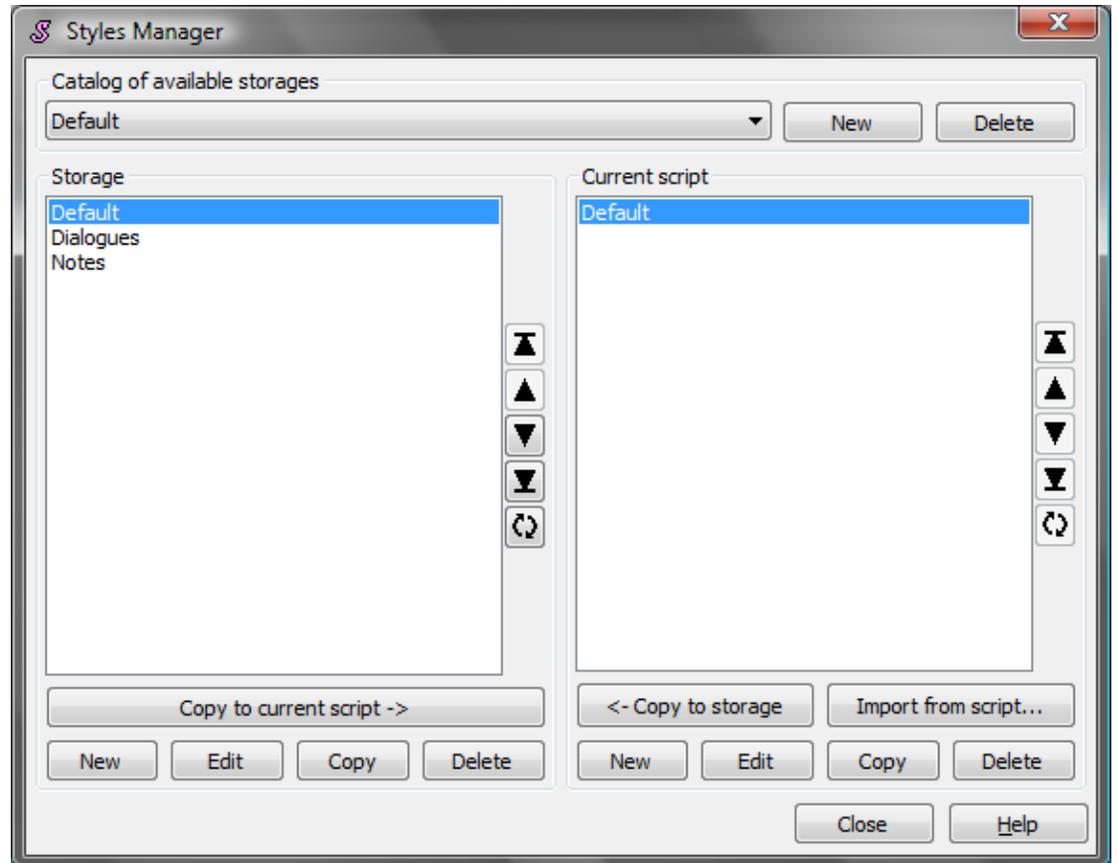
- **Jump to**
Seeks the video to the given time or frame.
- **Jump video to start**
Seeks the video to the start time of the currently active line.
- **Jump video to end**
Seeks the video to the end time of the currently active line.

Styles

A style in the ASS format is a set of typographic formatting rules that is applied to dialogue lines. The style parameters can be overridden with [override tags](#); the styles exist so that you won't have to type out all the overrides for each line.

The styles manager

Aegisub's styles manager tool (accessed from the Subtitles menu) provides various means of organizing, saving and editing styles. It looks like this:



As you can see, the window is divided into two halves. The right half displays all styles that you currently have accessible in your currently loaded script; the left represents a style storage. Style storages are used by Aegisub to let you save styles and quickly import them into scripts. You can have as many storages as you want; some people prefer to keep all their styles in one storage, others sort their styles by font, by show or alphabetically. The dropdown menu at the top lets you choose what storage you want to view.

In the bottom part of the window there's two almost identical sets of buttons; one set for the storage and one for the current script. They are:

- **New** - Create a new style, either in the storage or in the current script.
- **Edit** - Edit the selected style using the style editor (see below).
- **Copy** - Create a copy of the selected style and bring up the style editor for it.
- **Delete** - Delete the selected style(s).
- **Copy to script** and **Copy to storage** - copies the selected styles between the storage and the current script.
- **Import from script** - Lets you import one or more styles from another script to the current one.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

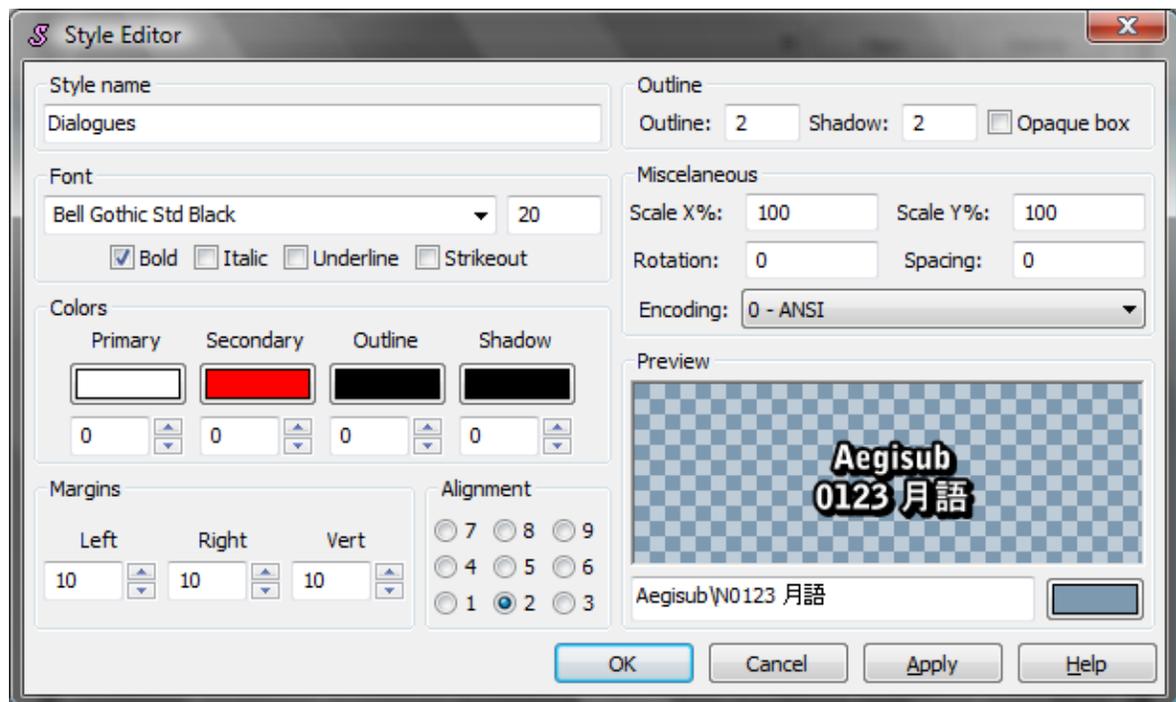
- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Pressing Ctrl-C in the styles manager will copy the selected style(s) to the clipboard as a text string. This also works the other way around; you can copy one or more style lines from another program and press Ctrl-V (paste) in the styles manager; this will insert them into the current script.

The style editor

Pressing the "Edit" button or doubleclicking a style name will bring up the style editor, which lets you tweak all the parameters available. **Note:** some parameters that are available as override tags (i.e. `\be`) are *not* available as style parameters; conversely, some parameters that are available in the styles editor (i.e. the "opaque box" outline option) are *not* available as a style override. This is an unfortunate and occasionally annoying format limitation.

To return to the style editor:



- **Style name** - The name of the style. You can't have two styles with the same name in the same script.
- **Font** - This section controls the font settings. The dropdown menu lets you choose the typeface (any font installed on your system will show up in this list) and the number to the right controls the size in points. The checkboxes below set the bold/italic/underline/strikeout parameters.
- **Colors** - This controls the four text colors (primary, secondary, outline and shadow). The meaning of each of them is as follows:
 - *Primary* - used as the "fill" (main) color of the text.
 - *Secondary* - secondary fill color, only used for karaoke effects (see `\k` and its relatives on the [override tags page](#)).
 - *Outline* - used as outline (border) color.
 - *Shadow* - used as shadow color.

The four colored boxes show you the current color for each of the four text colors; clicking on them opens the [color picker](#). The spin controls/number boxes below each colored box controls the transparency of that color. The transparency (or "alpha") is given as a number between 0 (opaque) and 255 (completely transparent).

- **Margins** - Controls how close to the borders of the video frame the text will be positioned (and hence also when automatic linebreaking will kick in; but do note that there is no option for text justification). Each value (left, right, vertical) is given in

script resolution pixels (see [script properties](#)). For alignments (`\an`) 1-3, the vertical margin is relative to the bottom of the video frame; for 4-6 it doesn't have any meaning, and for 7-9 it is relative to the top of the video frame.

- **Alignment** - Controls the alignment of the text. The numbers correspond to the arguments to the `\an` tag. Alignments 1, 4 and 7 are flush left; 3, 6 and 9 are flush right; 2, 5 and 8 are centered. 1, 2 and 3 are "subtitles" (i.e. drawn at the bottom of the frame); 4, 5 and 6 are "midtitles" (i.e. centered vertically on the frame); 7, 8 and 9 are "toptitles" (i.e. drawn at the top of the frame). Do note that there is no such thing as a justified alignment; this is a format limitation.
- **Outline** - Controls the outline (border) thickness (and style) and the shadow offset.
 - For the outline, the number is the thickness of the border in script pixels if `ScaledBorderAndShadow` is enabled (see [script properties](#)); if it is disabled the thickness is given in video pixels instead. Setting it to 0 disables the outline.
 - The shadow on the other hand is basically a copy of the main text, and the number controls how far downwards and to the right the shadow is drawn from the main fill. It's given in script pixels if `ScaledBorderAndShadow` is enabled, video pixels otherwise. Setting it to 0 disables shadow.
 - Finally, the checkbox marked "Opaque box" controls whether the outline should be drawn as described above (which happens if it is unticked), or replaced by an opaque background bounding box.
- **Miscellaneous** - this section has various other parameters, namely:
 - *Scale X* and *Scale Y* - Controls text stretching in the horizontal (X) and vertical (Y) direction respectively. The value is given in percent, so 100 means no stretching is done. You should not use this for normal text sizing; use the point value in the Font area instead since that uses hinting information from the font.
 - *Rotation* - Controls rotation of the text. The value is the angle of the rotation in degrees (360 degrees to a full circle) and may be negative or larger than 360 (setting it to 360, 720 etc is the same as setting it to 0).
 - *Spacing* - Controls the horizontal spacing between letters. 0 means the font defaults are used. May be negative for less spacing, or positive for more. It is currently unknown what unit is used.
 - *Encoding* - Controls which codepage is used to map codepoints to glyphs; it has nothing to do with the actual text encoding of the script. This is only meaningful on Windows using VSFilter, where it is used to get some old (particularly Japanese) fonts without proper Unicode mappings to render properly. On other systems and renderers, FreeType2 provides the proper mappings. If you didn't understand a word of the above, pretend this setting doesn't exist.
- **Preview** - Shows a preview of what the text will look like using the current style parameters. The text field lets you input some sample text to preview, and the colored box controls the color of the background.

Visual Typesetting

One of the most significant changes from Aegisub 1.10 to Aegisub 2 is the inclusion of Visual Typesetting tools, which allow you to visually drag, rotate and otherwise transform subtitles on the video screen, using your mouse.

A video tutorial for this feature is available at [the tutorials section](#).

Overview

With the exception of the drag tool, every other tool affects only the active line (that is, the line whose text you see on the edit box). The line must be displayed on the current frame (that is, the current frame time must lie between the subtitle duration span) to be edited by any of the tools.

There are currently seven different visual typesetting tools: crosshair, drag, rotate on z axis, rotate on xy axes, scale, rectangular clip and vectorial clip.

Contents [\[hide\]](#)

- 1 Overview
- 2 Tool Descriptions
 - 2.1 Crosshair
 - 2.2 Drag
 - 2.3 Rotate on Z axis
 - 2.4 Rotate on XY axes
 - 2.5 Scale
 - 2.6 Rectangular clip
 - 2.7 Vectorial clip

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

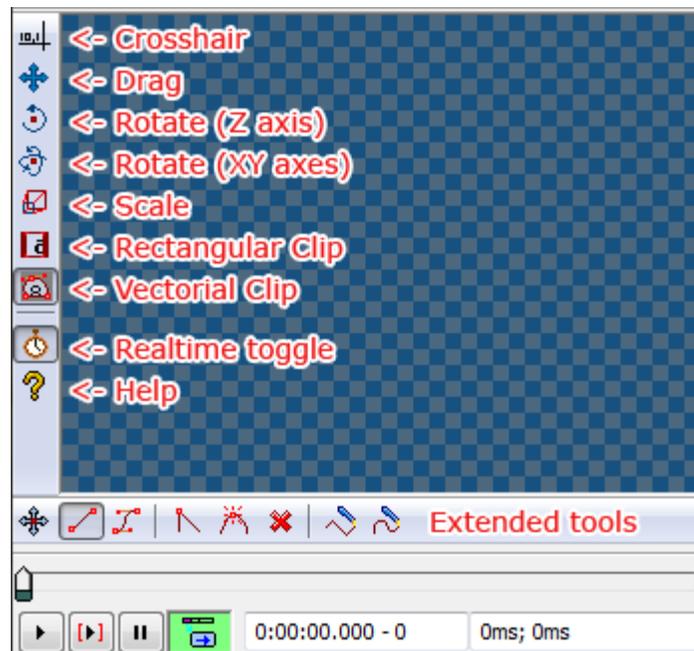
- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

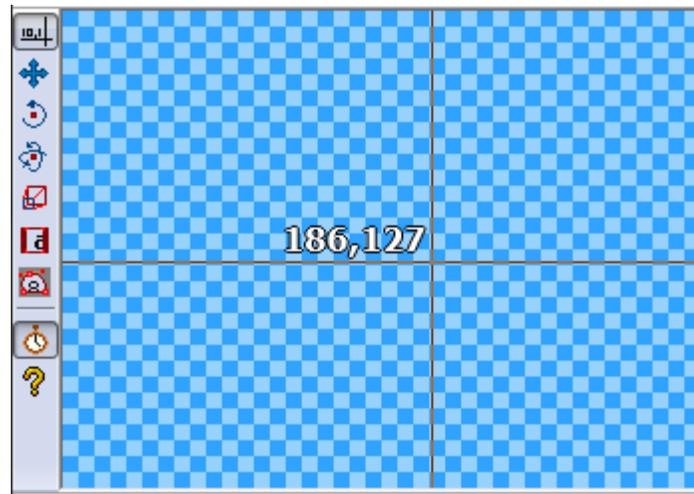


The preview of all tools can either be real-time or delayed. This behaviour can be toggled by pressing the "Toggle Real-time" button at the end of visual typesetting toolbar. When in real-time, all changes you made are immediately visible on subtitles, at the cost of slower response. Otherwise, you will only see the subtitles themselves updating when you release the mouse. However, you will still see the visual positioning aids moving in real time.

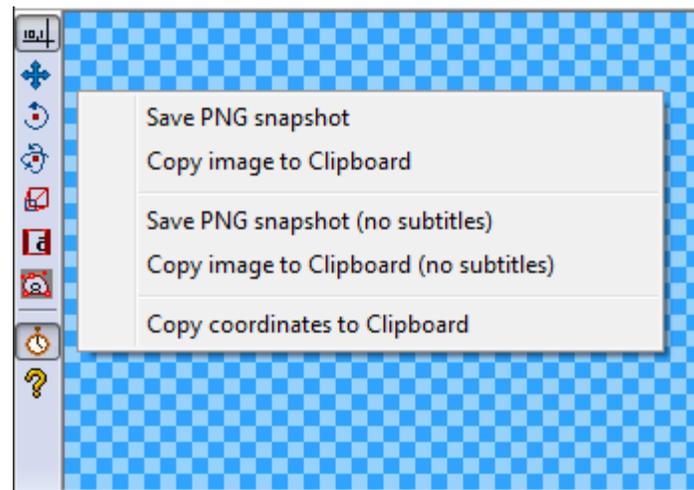
Tool Descriptions

Crosshair

This is the standard mode. Hovering the mouse over the video will show the script coordinates under the cursor, as well as a crosshair indicating the exact point that you are looking at. Double clicking a point will position the current line (using the \pos tag) on that position.



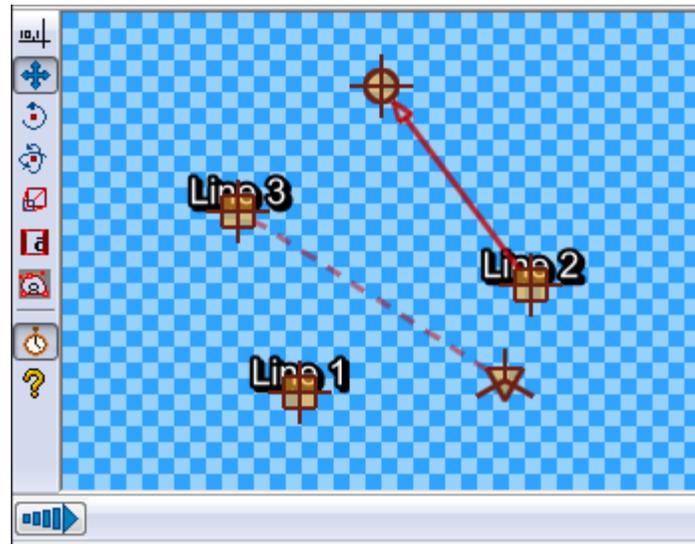
Also, while on this mode, you can right click to bring a context menu with other options:



- Save PNG snapshot: Saves a PNG snapshot of the current frame to the path specified in the [options](#). This snapshot will be in the real video size, and WON'T be affected by zooming or aspect ratio overriding.
- Copy image to clipboard: Same as above, but copies the frame to the clipboard, instead of saving as a PNG. You can then paste it in any image editing software.
- Save PNG snapshot (no subtitles): Same as the previous option, but subtitles won't be visible in the image. This is only available if you're using a detached subtitles provider.
- Copy image to clipboard (no subtitles): Same as above, but copying to clipboard.
- Copy coordinates to clipboard: Copies the current mouse coordinates to clipboard, e.g. "230,152"

Drag

The drag tool has two modes. You can toggle between them by clicking the button on the auxiliary visual typesetting bar. This tool is special because, unlike every other, it can affect EVERY subtitle that is currently visible on the current frame, and not just the active one. This makes the positioning of signs composed of many lines easier.



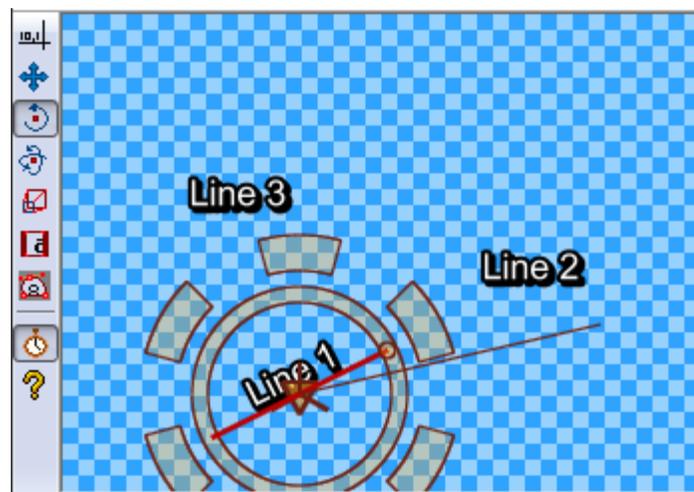
On the positioning mode, you can simply click-and-drag subtitles across the video surface by dragging their anchor (the square). They will be positioned where you release them with the `\pos` command.

On the movement mode, there is another anchor, the end-of-movement anchor, which is represented by a circle. There will be an arrow pointing from the start-of-movement to the end-of-movement anchor. In order to set the times where the movement will start or end, move the corresponding anchor on the time where you want the subtitle to be at that anchor. For example, if you want movement to start at 5000 milliseconds after the start of the line, seek the video to 5000 milliseconds after the start of the line, and drag the start anchor there. The same goes for the end-of-movement anchor.

If your line has an origin point specified, you will see a third anchor, represented by a triangle, connected to the square anchor by a dotted line. You can also drag this to affect the origin position. This same anchor will be visible in the two rotation tools.

Rotate on Z axis

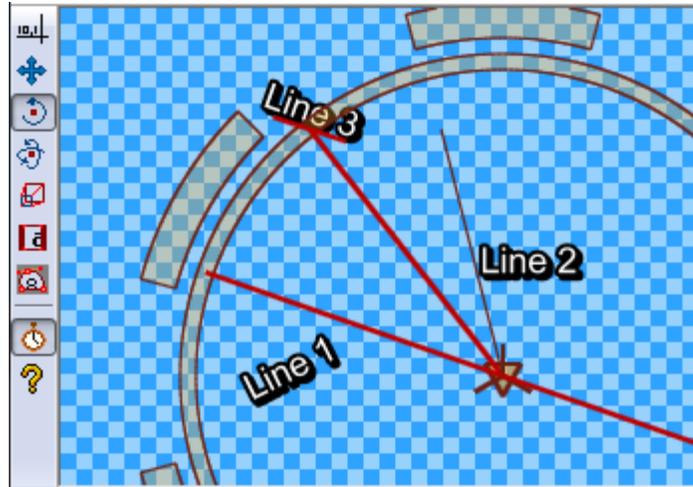
On this mode, you will see a circle centered around the subtitle's pivot (either its position, or the origin point, if defined). The circle is surrounded by 6 arcs that help mark and measure angles.



There are two functionalities in this mode, you can either drag the origin point (the triangle at the center of the circle) to move it (and insert a `\org` tag, if none exists), or you can click and drag anywhere else to rotate the line.

You will notice that there is a line connecting the center of the circle to the mouse cursor. When you click and drag, the subtitle line rotate following that line, that is, you must rotate the cursor around the center to rotate the line. Just release the mouse to finish positioning. You can also use the shift key to limit the rotation to increments of 30 degrees.

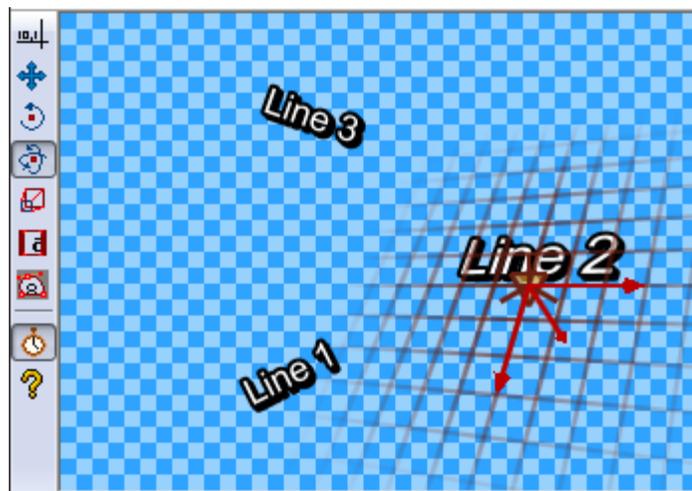
If the origin is far from the center, you will see a little helper line indicating where the center of the subtitles will be positioned as you rotate it.



Rotate on XY axes

This mode is somewhat similar to the previous mode, although there are a few important differences. Since this rotates on the two screen axes, the rotation is three-dimensional, and therefore harder to visualize.

In order to make this easier, there is a grid representing the plane in which the subtitles lie, and you rotate that grid. There are also three arrows pointing from the center, indicating the direction and orientation of the three axes.



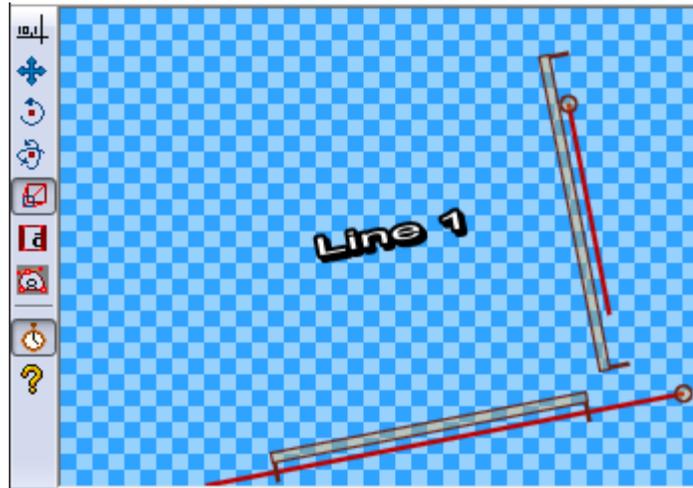
To use this tool, simply hold the mouse button anywhere on the screen and move it. As you move it left and right, it will rotate the line on the Y axis, and as you move it up and down, it will rotate the line on the X axis.

If you hold down the ctrl key while rotating, the rotation will be limited to only one of the two axes - whichever has the greatest movement. If you hold down the shift key, rotation will happen in steps of 30 degrees.

As with the Z rotation tool, you can also drag the origin anchor here.

Scale

This is the simplest tool, and allows you to scale subtitles on the X and Y axes. It will show one bar for each axis, showing not only the 100% size, but also the current scale.



To use this tool, simply hold down the mouse button and drag the mouse up and down (to scale on the Y axis) or left and right (to scale on the X axis). You can hold down the ctrl key to limit scaling to the axis where the biggest change happened, and shift to limit it to increments of 25%.

Rectangular clip

The rectangular clip tool allows you to clip the subtitles so that nothing OUTSIDE an axis-aligned rectangle can be displayed (in essence, the `\clip(x1,y1,x2,y2)` tag).

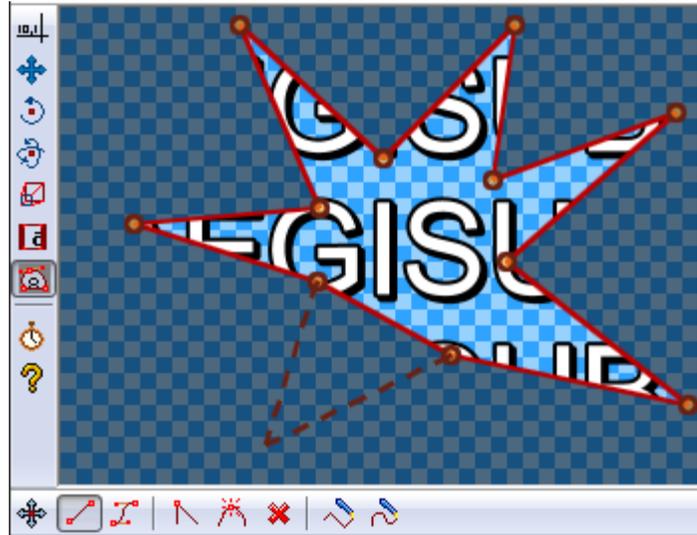


There are two ways to use this tool. You can either click and grab one of the four vertices of the rectangle, to resize an already-existing clip, or you can click-and-drag in empty space to create a new rectangle from scratch. The areas that will be invisible will be darkened.

Vectorial clip

Similarly to the last tool, the vectorial clip tool allows you to draw an area, so that nothing outside it will be rendered. The difference, however, is that this area can have

any arbitrary shape defined by a path of lines and bézier curves.



This mode has 8 sub-tools:



1. Drag - Allows you to drag a control point
2. Insert line - Allows you to insert a straight line from the last point to the current mouse position by clicking the point.
3. Insert bézier bicubic curve - The same as above, but it instead inserts a bicubic curve. You can then use the two control points to adjust the shape of the curve.
4. Convert between line and curve - Click on a line segment or bicubic curve to convert it to the other type.
5. Split curve - Click on a line segment or bicubic curve to split it in two, at the marked point.
6. Remove point - Click on a point to delete it.
7. Freehand shape - Click and drag with the mouse over the video and move the mouse to draw a freehand shape composed of line segments. This shape will automatically be closed, with the last point connecting to the first.
8. Freehand smooth shape - Same as above, but the shape will be smoothed with bicubic curves.

ASS Tags

The following is a list of every tag supported by the Advanced Substation Alpha format. This is basically a detailed version of `ass-quickref.txt`. See the tutorial for an introduction to typesetting, using some basic tags.

Override tags always follow the same form: They start with a backslash `\` character, then a name, and after the name a parameter to the tag.

Some tags are "complex" and take more than one parameter, in that case the parameters are put inside parentheses with commas between the parameters.

Note on typography:

On this page, everything written in *italics* with `<` angle brackets `>` around, means it is a parameter and you must enter a value instead of it. The angle brackets are not part of the value you should enter. Use the examples as a guide to how the tags should be entered. In general, the same rules apply to all tags, in how they look.

Special characters

The following tags are written in the middle of the text, and not inside override blocks (i.e. not between `{` and `}`).

Soft line break

`\n`

Insert a forced line break, but only when in wrapping mode 2. (See [the `\q` tag](#).) Note that this is a lowercase `n`.

In all other wrapping modes, this is replaced by a regular space.

Hard line break

`\N`

Insert a forced line break, regardless of wrapping mode. Note that this is an uppercase `N`.

Hard space

`\h`

Insert a non-breaking "hard" space. The line will never break automatically right before or after a hard space, and hard spaces are not folded when they appear at the start or end of a displayed line.

Override tags

These tags must be in an override block, that is, they must be between a `{` and a `}`, and will affect all text following the block, unless they get re-overridden.

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Italics

```
\i1  
\i0
```

Switch *italics* text on or off. Use `\i1` to enable italics for the following text and `\i0` to disable italics again.

Bold

```
\b1  
\b0  
\b<weight>
```

Switch **boldface** text on or off. Use `\b1` to enable boldface for the following text and `\b0` to disable boldface again.

The `\b<weight>` form allows you to specify an explicit weight to use. Note that most fonts only support one or two weights so you rarely need to use this. Font weights are multiples of 100, such that 100 is the lowest, 400 is "normal", 700 is "bold" and 900 is the heaviest.

Example

```
I am {\b1}not{\b0} amused.
```

The word "not" is written in boldface.

Example

```
{\b100}How {\b300}bold {\b500}can {\b700}you {\b900}get?
```

The words are written with increasingly greater weight. Note that most fonts do not have more than one or two different weights and you will only be able to see "not bold" and "bold" in that case.

Underline

```
\u1  
\u0
```

Switch underlined text on or off. Use `\u1` to enable underlining for the following text and `\u0` to disable underlining again.

Strikeout

```
\s1  
\s0
```

Switch ~~striked-out~~ text on or off. Use `\s1` to enable strikeout for the following text and `\s0` to disable strikeout again.

Border size

```
\bord<size>
```

Change the width of the border around the text. Set the size to 0 (zero) to disable the border entirely.

If "scale border and shadow" (see [script properties](#)) is enabled, the value is given in script resolution pixels, otherwise it is given in video resolution pixels (which means the border thickness will vary depending on the resolution of the video on which the subtitles are

rendered.)

The value is not limited to whole integer pixels, it can have decimal places.

Example

```
\bord0
```

Disable border entirely.

Example

```
\bord3.7
```

Set the border width to 3.7 pixels

Border size (extended)

```
\xbord<size>
```

```
\ybord<size>
```

These tags are only available in VSFilter 2.39 and later, and might not be safe for softsubbing.

Use the `\xbord` `\ybord` tags to set the border size in X and Y direction separately. This can be useful for correcting the border size for anamorphic rendering of subtitles.

Note that if you use `\bord` after `\xbord` or `\ybord` on a line, it will override both of them.

You can set the border width to 0 (zero) in one of the directions to entirely disable border in that direction.

Shadow distance

```
\shad<depth>
```

Set the distance from the text to position the shadow. Set the depth to 0 (zero) to disable shadow entirely. Works similar to `\bord`.

The shadow distance can not be negative with this tag.

Shadow distance (extended)

```
\xshad<depth>
```

```
\yshad<depth>
```

These tags are only available in VSFilter 2.39 and later, and might not be safe for softsubbing.

Set the distance from the text to position the shadow at, with X and Y position set separately. Shadow is only disabled if both X and Y distance is 0.

Note that you can set the distance negative with these tags to position the shadow to the top or left of the text.

Blur edges

```
\be0
```

```
\be1
```

```
\be<strength>
```

The `\be<strength>` version is only available in VSFilter 2.39 and later and might

not be safe for softsubbing.

Enable or disable a subtle softening-effect for the edges of the text. The effect isn't always very visible, but it can in some cases make the text look better. It is usually more visible at smaller text sizes.

Be aware that this tag blurs the *edges* of the text, not everything. This means that if the text has a border (set with `\bord`) the border will be blurred, but if there is no border, the main text will be blurred instead.

In the extended version, *strength* is the number of times to apply the regular effect. Note that at high values the effect de-generates into nothingness, and generally isn't very useful. The *strength* must be an integer number.

Blur edges (gaussian kernel)

`\blur<strength>`

This tag is only available in VSFilter 2.39 and later, and might not be safe for softsubbing.

In general, this has the same function as the `\be` tag, but uses a more advanced algorithm that looks better at high strengths. Unlike `\be`, the *strength* can be non-integer here. Set *strength* to 0 (zero) to disable the effect. Be careful, setting *strength* too high can take a lot of CPU time to render.

Be aware that this tag blurs the *edges* of the text, not everything. This means that if the text has a border (set with `\bord`) the border will be blurred, but if there is no border, the main text will be blurred instead.

Font name

`\fn<name>`

Set the font face to use for the following text. There should be no space between `\fn` and the font name, and you should not put parentheses or similar around the font name either.

Example

```
\fnArial
```

The text following this tag will be in Arial font.

Example

```
\fnTimes New Roman
```

The text following this tag will be in Times New Roman font.

Font size

`\fs<size>`

Set the size of the font. The size specified is the height in script pixels, so at font size 40 one line of text is 40 pixels tall. (Technical note: it's really typographic (desktop publishing) points, not script pixels, but since the rendering is always done at 72 DPI (as per the de facto standard), one point ends up being exactly equal to one script resolution pixel.)

You can only specify integer font sizes.

Example

```
\fs10
```

The following text will use a size 10 font.

Font scale

```
\fscx<scale>
```

```
\fscy<scale>
```

Adjust the size of the text in X (`\fscx` or Y (`\fscy`) direction. The *scale* given is in percent, so 100 means "original size".

This is not the same as setting the font size, as setting the size is subject to [font hinting](#) while scaling the text modifies the text shape after hinting.

These tags also affect [vector drawings](#).

You can use font scaling to correct for anamorphic rendering and to specify text size more precisely than with `\fs`.

Example

```
\fscx150
```

Make the text 50% wider than normal.

Example

```
\fscy50
```

Make the text half height.

Example

```
\fscx200\fscy200
```

Make the text double size.

Letter spacing

```
\fsp<spacing>
```

Changes the spacing between the individual letters in the text. You can use this to spread the text more out visually. The *spacing* is given in script resolution pixels.

Text rotation

```
\frx<amount>
```

```
\fry<amount>
```

```
\frz<amount>
```

```
\fr<amount>
```

Rotates the text along the X, Y or Z axis. The `\fr` tag is a shortcut for `\frz`.

- The **X axis** runs horizontally on the screen. Rotating on it (with positive values) causes an effect where the top of the text moves farther "into" the screen while the bottom moves "out" of the screen.
- The **Y axis** runs vertically on the screen. Rotating on it (with positive values) causes the text to rotate so that the left moves "outside" the screen, when the right moves "into" the screen.
- The **Z axis** runs perpendicular to the screen. Rotating on it (with positive values) causes the text to rotate in 2D, counterclockwise (as standard for degrees).

The rotation *amount* is given in mathematical degrees, such that 360 degrees is a full rotation, and rotating any multiple of 360 is the same as not rotating. It is legal to

specify negative rotation amounts, as well as amounts larger than 360 degrees.

The rotation is performed around the subtitle line origin point, this is described with the [\org](#) tag.

These tags also affect [vector drawings](#).

Example

```
\frx45
```

Rotate the text 45 degrees on the X axis.

Example

```
\fry-45
```

Rotate the text 45 degrees in opposite direction on the Y axis.

Example

```
\frz180
```

Rotate the text 180 degrees on the Z axis, making it upside-down.

Example

The following two rotations produce the same result:

```
\frz-30
```

```
\frz330
```

This is because 330 degrees is 30 degrees less than a full rotation of 360 degrees.

Example

```
\t(\frz3600)
```

Perform an animation where the text performs 10 full revolutions on the Z axis. Also see the [\t](#) tag.

Example

The following screenshots illustrate the effect of rotating on the different axes:





Text shearing

```
\fax<factor>  
\fay<factor>
```

These tags are only available in VSFilter 2.39 and later, and might not be safe for softsubbing.

Perform a shearing (perspective distortion) transformation of the text. A *factor* of 0 (zero) means no distortion.

Usually *factor* will be a small number, not larger than 2 as that creates a very strong distortion.

Shearing is performed after rotation, on the rotated coordinates. The coordinate system used for shearing is not affected by the [rotation origin](#).

Todo: We need a screenshot illustrating the effect.

Font encoding

```
\fe<id>
```

Set the Windows font encoding used to select the font mapping table used to translate Unicode codepoints to glyph indices in the font. For some fonts without a Unicode mapping table this might be required to get text in certain languages to work. For fonts that do have a Unicode mapping table, it might be used to pick regional variations, such as picking the correct glyph for a Han ideogram that is different in Simplified Chinese, Traditional Chinese, Japanese and Korean.

Some common font encoding ID's are:

- 0 - ANSI, Windows CP-1252 for Western-European languages.
- 1 - Default, depends on the configuration of the user's system, but also allows the font sub-system to dynamically pick a different mapping table in some circumstances.

Todo: Is that really correct?

- 2 - Symbol, codepoints in the 0-255 range are translated to per-font defined symbol glyphs, this is used for fonts such as Wingdings.
- 128 - Shift-JIS, used for Japanese.
- 129 and 130, respectively Hangeul and Johab, two encoding schemes for Korean.
- 134 - GB2312, used for Simplified Chinese.
- 136 - BIG5, used for Traditional Chinese.
- 162 - Turkish.
- 163 - Vietnamese.
- 177 - Hebrew.
- 178 - Arabic.

A more complete list can be seen the [style editor](#) dialog box.

In ASS files stored in non-Unicode encodings, this tag also affects what codepage the text following it should be interpreted in. Aegisub doesn't support this use and some renderers might not support it either, it is recommended you do not rely on this and instead always store your files in a Unicode encoding. (Aegisub stores files in Unicode UTF-8 by default.)

Set color

```
\c&H<bb><gg><rr>&  
\1c&H<bb><gg><rr>&  
\2c&H<bb><gg><rr>&  
\3c&H<bb><gg><rr>&  
\4c&H<bb><gg><rr>&
```

Set the color of the following text. The `\c` tag is an abbreviation of `\1c`.

- `\1c` sets the primary fill color.
- `\2c` sets the secondary fill color, this is only used for pre-highlight in standard karaoke.
- `\3c` sets the border color.
- `\4c` sets the shadow color.

The color codes are given in [hexadecimal](#)  in Blue Green Red order. Note that this is the opposite order of eg. HTML color codes. Color codes must always start with `&H` and end with `&`.

The Pick Color toolbar buttons  can assist in picking colors and entering the color codes.

Set alpha

```
\alpha&H<aa>  
\1a&H<aa>  
\2a&H<aa>  
\3a&H<aa>  
\4a&H<aa>
```

Set the alpha (transparency) of the text.

- `\alpha` sets the alpha of all components at once.

- `\1a` sets the primary fill alpha.
- `\2a` sets the secondary fill alpha, this is only used for pre-highlight in standard karaoke.
- `\3a` sets the border alpha.
- `\4a` sets the shadow alpha.

An alpha of 00 (zero) means opaque/fully visible, and an alpha of FF (ie. 255 in decimal) is fully transparent/invisible.

Example

```
\alpha&H80&
```

Set the alpha of all components to hexadecimal 80, decimal 128, making the text 50% transparent in general.

Example

```
\1a&HFF&
```

Set the primary fill alpha to hexadecimal FF, decimal 255, making it invisible and effectively leaving only the border and shadow.

Line alignment

```
\an<pos>
```

Specify the alignment of the line. The alignment specifies the position of the line when no [position override](#) or [movement](#) is in effect, and otherwise specifies the anchor point of positioning.

The `\an` tag uses "numpad" values for the *pos*, ie. the alignment values correspond to the positions of the digits on the numeric keypad on a regular keyboard:

- 1: Bottom left
- 2: Bottom center
- 3: Bottom right
- 4: Middle left
- 5: Middle center
- 6: Middle right
- 7: Top left
- 8: Top center
- 9: Top right

Line alignment (legacy)

```
\a<pos>
```

Specify the alignment of the line using legacy alignment codes from SubStation Alpha. This tag is supported but considered deprecated; you should usually use `\an` in new scripts instead, as it is more intuitive.

Calculate *pos* as follows: Use 1 for left-alignment, 2 for center alignment and 3 for right-alignment. If you want sub-titles you're done. To get top-titles, add 4 to the number, to get mid-titles add 8 to the number:

- 1: Bottom left
- 2: Bottom center
- 3: Bottom right
- 5: Top left
- 6: Top center
- 7: Top right
- 9: Middle left

- 10: Middle center
- 11: Middle right

Karaoke effect

```
\k<duration>
\K<duration>
\kf<duration>
\ko<duration>
```

Please note that these tags alone only create some very specific effects and all other effects are created with a combination of multiple different tags.

The `\k` family of tags mark up subtitles for karaoke effects by specifying the duration of each syllable. You place a `\k` tag before each syllable in the line.

The *duration* is given in centiseconds, ie. a *duration* of 100 is equivalent to 1 second. You generally don't enter `\k` tags manually but rather use karaoke timing tools such as [Aegisub's karaoke mode](#).

The different `\k` tags create various effects:

- `\k`: Before highlight, the syllable is filled with the secondary color and alpha. When the syllable starts, the fill is instantly changed to use primary color and alpha.
- `\K` and `\kf`: These two are identical. Note that `\K` is an uppercase K and is different from lowercase `\k`. The syllable fill starts out secondary color, when the syllable starts, the fill changes from secondary to primary with a sweep from left to right, so the sweep ends when the syllable time is over.
- `\ko`: Similar to `\k`, except that before highlight, the border/outline of the syllable is removed, and appears instantly when the syllable starts.

Note: There is an additional karaoke tag, `\kt`, which is very different from the other ones. Aegisub does not support that tag and so it is not documented.

Wrap style

```
\q<style>
```

Determine how line breaking is applied to the subtitle line. The following *styles* are available:

- 0: Smart wrapping, make each line approximately equally long, but top line wider when equal width is impossible. Only `\N` forces line breaks.
- 1: End-of-line wrapping, fill as much text in a line as possible, then break to next line. Only `\N` forces line breaks.
- 2: No word wrapping, wide lines will extend beyond the edges of the screen. Both `\n` and `\N` force line breaks.
- 3: Smart wrapping, similar to style 0, but bottom lines are made wider.

Reset style

```
\r
\r<style>
```

Reset the style. This cancels all style overrides in effect, including [animations](#), for all following text.

The first form that does not specify a *style* will reset to the style defined for the entire line, while the second form, that specifies the name of a *style*, will reset the style to that specific style.

Example

```
-Hey\n{\rAlternate}-Huh?\n{\r}-Who are you?
```

Assuming the current line style is "Default", this has first "Hey" in the Default style, then follows on next line "Huh?" in the style "Alternate", and on the third line the style is reset to "Default" for the "Who are you?" text.

Set position

```
\pos (<X>, <Y>)
```

Set the position of the line. The X and Y coordinates must be integers and are given in the script resolution coordinate system. The meaning of X and Y changes slightly depending on [alignment](#).

In VSFilter 2.39 and later, you can use non-integer numbers for coordinates. Note that this might not be safe for softsubbing.

The alignment of the subtitle line is used as anchor point for the position, eg. when you have a line with alignment top-left, the top-left corner of the subtitle is placed at the coordinates given to `\pos`, and for bottom-center alignment, the bottom center of the subtitle is placed at the coordinates given.

Example

The following screenshots illustrate the way alignment affects positioning. The green cross marks the point (320,240) on the video.





Movement

```
\move(<x1>,<y1>,<x2>,<y2>)  
\move(<x1>,<y1>,<x2>,<y2>,<t1>,<t2>)
```

The `\move` tag works similar to `\pos` in that it positions the subtitle line, the difference is that `\move` makes the subtitle move.

The two versions of `\move` differ in that one makes the movement occur over the entire duration of the subtitle, while on the other you specify the time over which the movement occurs.

The coordinates $x1$, $y1$, $x2$ and $y2$ are given in the script resolution coordinate system, like `\pos`. The subtitle starts out at point $(x1,y1)$ and moves with constant speed so it ends up at $(x2,y2)$. **Alignment** influences movement coordinates the same way as it

influences `\pos` coordinates.

In VFilter 2.39 and later, you can use non-integer numbers for the X and Y coordinates. Note that this might not be safe for softsubbing.

In the second version, the times $t1$ and $t2$ are given in milliseconds, ie. one thousandth of a second, and are relative to the start time of the subtitle. For example, a $t1$ value of 1500 means that the movement begins 1.5 seconds (one and a half second) after the line has appeared on screen. When you specify times for the movement, the position of the subtitle is as follows:

1. Before $t1$, the subtitle is stationary at point $(x1, y1)$.
2. Between $t1$ and $t2$, the subtitle moves with constant speed from $(x1, y1)$ to $(x2, y2)$.
3. After $t2$ the subtitle is stationary at point $(x2, y2)$.

Note that it is legal to have $t1$ and $t2$ specify times greater than the duration of the line, but it might not be very useful to do so. Specifying both $t1$ and $t2$ as 0 (zero) is the same as using the first version of `\move`, ie. the movement will occur from the start time of the line to the end time of the line.

There are some things `\move` can not do:

- Non-constant-speed movement is not possible. The movement can not, for example, start out slow and end fast.
- There can only be one positioning or movement tag in a line. Putting both a `\pos` and a `\move` tag in a line will not work. It will also not work to put two or more `\move` tags in a single line.

If you need to do do any of those two you will need to split the movement into segments that are done on separate subtitle lines. (How to do this is outside the scope of this page.)

Example

```
\move(100,150,300,350)
```

When the line appears on screen, the subtitle is at (100,150). While the subtitle is displayed, it moves at constant speed such that it will arrive at point (300,350) at the same time it disappears.

Example

```
\move(100,150,300,350,500,1500)
```

The line appears at (100,150). After the line has been displayed for half a second (500 milliseconds) it begins moving towards (300,350) such that it will arrive at the point a second and a half (1500 milliseconds) after the line first appeared on screen.

Rotation origin

```
\org(<X>,<Y>)
```

Set the origin point used for [rotation](#). This affects all rotations of the line. The X and Y coordinates are given in integer script resolution pixels.

When there is no `\org` tag in a line, the rotation origin is implicitly the same as the [position anchor point](#). This means that the rotation origin will move if the line moves and there is no origin set with `\org`. Note that you can *not* animate the `\org` tag, you are limited to a fixed origin if you use it.

If the rotation origin is placed on the vanishing point in a 3D scene, 3D rotations of subtitle line will produce the correct perspective to match the scene.

It's perfectly possible (and occasionally useful) to place the origin point far outside the actual image; if it is sufficiently far away, doing appropriately calculated small rotations will seem to move the text along a straight (or almost straight) line through the image. This is somewhat hard to control, but can be used to work around format limitations with `\move`, such as the inability to do accelerated moves or several moves per line.

There can be at most one `\org` tag in a single line, if you put more than one in a single line, only the first is used.

Example

```
\org(320,240)
```

Fix the rotation origin at point (320,240).

Example

```
\org(10000,0)
```

Placing the rotation origin at a far away point allows you to use slight `\frz` rotations to produce "jumping" effects; the text will move up or down without seeming to rotate.

Fade

```
\fad(<fadein>,<fadeout>)
```

Produce a fade-in and fade-out effect. The *fadein* and *fadeout* times are given in milliseconds, ie. 1000 means one second. You can specify *fadein* or *fadeout* as 0 (zero) to not have any fade effect on that end.

Adding a fade effect does not extend the duration of the line, rather the start or end of the line's display time is used for the fade effect. For this reason, you should be careful that *fadein+fadeout* is not greater than the duration of the line. For example, for a line displayed for 4 seconds, the sum of *fadein+fadeout* should not be greater than 4000.

Example

```
\fad(1200,250)
```

Fade in the line in the first 1.2 seconds it is to be displayed, and fade it out for the last one quarter second it is displayed.

Fade (complex)

```
\fade(<a1>,<a2>,<a3>,<t1>,<t2>,<t3>,<t4>)
```

Perform a five-part fade using three alpha values *a1*, *a2* and *a3* and four times *t1*, *t2*, *t3* and *t4*.

The alpha values are given in *decimal* and are between 0 and 255, with 0 being fully visible and 255 being invisible. The time values are given in milliseconds after the start of the line. All seven parameters are required. (For most common fade effects the `\fad` tag works fine.)

- Before *t1*, the line has alpha *a1*.
- Between *t1* and *t2* the line fades from alpha *a1* to alpha *a2*.
- Between *t2* and *t3* the line has alpha *a2* constantly.
- Between *t3* and *t4* the line fades from alpha *a2* to alpha *a3*.
- After *t4* the line has alpha *a3*.

Example

```
\fade(255,32,224,0,500,2000,2200)
```

Starts invisible, fades to almost totally opaque, then fades to almost totally

invisible. First fade starts when the line starts and lasts 500 milliseconds. Second fade starts 1500 milliseconds later, and lasts 200 milliseconds.

Animated transform

```
\t(<style modifiers>)  
\t(<accel>,<style modifiers>)  
\t(<t1>,<t2>,<style modifiers>)  
\t(<t1>,<t2>,<accel>,<style modifiers>)
```

Perform a gradual, animated transformation from one style to another. The *style modifiers* are other override tags as specified in this reference. Only a limited set of the override tags are animateable with `\t`:

Font	Geometry	Other effects
<code>\fs</code>	<code>\fscx</code>	<code>\bord</code>
<code>\fsp</code>	<code>\fscy</code>	<code>\shad</code>
<code>\c</code>	<code>\frx</code>	<code>\clip</code>
<code>\1c</code>	<code>\fry</code>	
<code>\2c</code>	<code>\frz</code>	
<code>\3c</code>	<code>\fr</code>	
<code>\4c</code>		
<code>\alpha</code>		
<code>\1a</code>		
<code>\2a</code>		
<code>\3a</code>		
<code>\4a</code>		

In VFilter 2.39 and later, the following additional tags can also be animated with `\t`:

`\fax \fay \be \blur \xbord \ybord \xshad \yshad \iclip`

Note: For `\clip` and `\iclip`, only the rectangle versions can be animated. The vector drawing versions cannot be animated.

The *t1* and *t2* parameters specify the time interval to perform the transformation over. In the versions without *t1* and *t2* the transformation is performed over the entire duration of the line. The times are given in milliseconds and are relative to the start time of the line. (For the rest of the description of `\t`, *t1* and *t2* are assumed to be specified, or implicitly be respectively 0 and the duration of the line.)

The *accel* parameter can be used to make the animation non-linear and instead follow an exponential curve. An *accel* parameter of 1 (one) causes the animation speed to be linear. A value between 0 and 1 causes the animation to start fast and end slow. A value greater than 1 causes the animation to start slow and end fast. (For the mathematically inclined, the function is $y = x^{accel}$ with $x \in [0;1] = (t-t1)/(t2-t1)$, *t* being the current time.)

Before *t1*, the style is as all tags before the `\t` tag specify. After *t2* the style is as all tags before the `\t` tag, and further overridden by the given *style overrides*. Between *t1* and *t2* the style is gradually animated between those two points, following the acceleration function described above.

Example

```
{\1c&HFF0000&\t(\1c&H0000FF&)}Hello!
```

The text starts out blue, but fades towards red so it is completely red when the line ends.

Example

```
{\an5\t(0,5000,\frz3600)}Wheee
```

Makes the text rotate 10 times, counterclockwise, lasting for 5 seconds.

Example

```
{\an5\t(0,5000,0.5,\frz3600)}Wheee
```

Same as above, but it will start fast and slow down, still doing the 10 rotations in 5 seconds.

Example

```
{\an5\fscx0\fscy0\t(0,500,\fscx100\fscy100)}Boo!
```

Text starts at zero size, ie. invisible, then grows to 100% size in both X and Y direction.

Clip (rectangle)

```
\clip(<x1>,<y1>,<x2>,<y2>)
```

```
\iclip(<x1>,<y1>,<x2>,<y2>)
```

The \iclip tag is only available in VSFilter 2.39 and later, and might not be safe for softsubbing.

Define a rectangle to clip the line, only the part of the line that is inside the rectangle is visible. The \iclip tag has the opposite effect, it defines a rectangle where the line is not shown.

The *x1*, *y1*, *x2* and *y2* coordinates are given in script resolution pixels and are relative to the top-left corner of the video. The coordinates must be integers, there is no possibility to use non-integer coordinates. (Increasing the script resolution will not increase the precision, the clipping always happens on video pixel boundaries.)

Note: If you use the \iclip tag anywhere on a line, all clips on that line will be treated as "inverse" regardless of whether a regular \clip occurs later or the \iclip has taken effect yet through \t or similar.

Example

```
\clip(0,0,320,240)
```

Assuming 640x480 script resolution, only the part of the line within the top left quadrant is visible.

Example

```
\iclip(0,0,320,240)
```

Similar to above, but instead the part of the line within the top left quadrant is hidden.

Example

Example of \clip(0,0,704,245) on a 704x480 video:



Clip (vector drawing)

```
\clip(<drawing commands>)
\clip(<scale>,<drawing commands>)
\iclip(<drawing commands>)
\iclip(<scale>,<drawing commands>)
```

The `\iclip` tags are only available in *VFilter 2.39* and later, and may not be safe for softsubbing.

Use the shape defined by a vector drawing to selectively display (`\clip`) or hide (`\iclip`) parts of the line.

The *drawing commands* are drawing commands as those used with the `\p` tag, the coordinates are given in script resolution pixels and are relative to the top left corner of the video.

If the *scale* is not specified it is assumed to be 1 (one), meaning that coordinates correspond directly to pixels. The *scale* works the same way as the *scale* for `\p` drawings.

Unlike the rectangular clip, the vector drawing clip can *not* be animated with `\t`. If you need to animate a vector drawing clip, you must create multiple similar subtitle lines with each their own "frame" of the clipping animation.

Example

```
\clip(1,m 50 0 b 100 0 100 100 50 100 b 0 100 0 0 50 0)
```

Only show the portion of the line within the defined pseudo-circle.

Beware: VFilter 2.37 has a bug that causes vector drawing clips to be rendered incorrectly. The solution is to use a newer or older version.

Note: If you use the `\iclip` tag anywhere on a line, all clips on that line will be treated as "inverse" regardless of whether a regular `\clip` occurs later or the `\iclip` has taken effect yet through `\t` or similar.

Drawing tags

Advanced Substation Alpha also supports some advanced drawing tags that allow you to draw with vectorial graphics. Certain familiarity with vectors and splines will make the understanding of this much simpler.

`\p<0/1/..>` - Toggle drawing mode

Setting this tag to 1 or above enables drawing mode. Text after this override block will then be interpreted as drawing instructions, and not as actually visible text. Setting this to zero disables drawing mode, restoring normal behavior. When turning on, the value might be any integer larger than zero, and will be interpreted as the scale, in $2^{(\text{value}-1)}$ mode. This is done to allow sub-pixel accuracy. e.g.:

```
\p1
```

(Enables drawing with normal coordinates)

```
\p0
```

(Disables drawing)

```
\p2
```

(Enables drawing, and resolution is doubled. So drawing to 200,200 will actually draw to 100,100)

```
\p4
```

(Enables drawing, and resolution is 8x larger ($2^{(4-1)}$). So drawing to 400,400 will actually draw to 50,50)

`\pbo<y>` - Baseline offset

Defines baseline offset for drawing. This is basically an Y offset to all coordinates. e.g.:

```
\pbo-50
```

(Draws everything 50 pixels above specified)

```
\pbo100
```

(Draws everything 100 pixels below specified)

Drawing commands

These commands should appear either in a `\clip` tag (vectorial overload) or between `\p#` and `\p0`, outside override blocks. For example (taken straight from the ASS specs):

- Square:

```
{\p1}m 0 0 1 100 0 100 100 0 100{\p0}
```

- Rounded square:

```
{\p1}m 0 0 s 100 0 100 100 0 100 c{\p0}
```

(c equals to "p 0 0 100 0 100 100" in this case)

- Circle (almost):

```
{\p1}m 50 0 b 100 0 100 100 50 100 b 0 100 0 0 50 0{\p0}
```

(note that the 2nd 'b' is optional here)

Drawing commands use the primary color for fill and outline color for borders. They also display shadow. The idea of drawing vectors is that there is an invisible "cursor" (think of it as the mouse pointer in a drawing program, or as a pen moving through the image) on the video frame, and you tell it to move to other positions. As it moves, it draws on the area behind it, and when you close the line formed, it fills it with the primary color.

m <x> <y> - Move

Moves the cursor to x,y. If you have an unclosed shape, it will automatically be closed, as the program assumes that you are now drawing a new, independent shape. All drawing routines must start with this command.

n <x> <y> - Move (no closing)

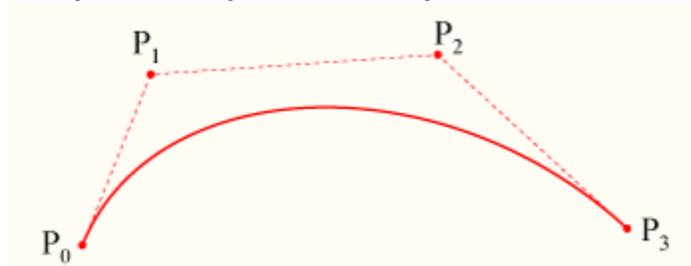
Moves the cursor to x,y, without closing the current shape.

l <x> <y> - Line

Draws a line from the current cursor position to x,y, and moves the cursor there afterwards.

b <x1> <y1> <x2> <y2> <x3> <y3> - Cubic Bézier curve

Draws a cubic (3rd degree) Bézier curve from the cursor position to (x3,y3), using (x1,y1) and (x2,y2) as the control points. Check the [article on Wikipedia](#) for more information about Bézier curves. In this picture taken from that article, P0 is the cursor position, P1 is x1,y1, P2 is x2,y2 and P3 is x3,y3:



Note that the curve begins at P0, heads towards P1, then arrives at P3 coming from P2's direction.

s <x1> <y1> <x2> <y2> <x3> <y3> .. <xN> <yN> - Cubic b-spline

Draws a cubic (3rd degree) uniform b-spline to point N. This must contain at least 3 coordinates (and is, in that case, the same as b). This basically lets you chain several cubic Bézier curves together. Check this other article on Wikipedia for more information.

p <x> <y> - Extend b-spline

Extends the b-spline to x,y . This is essentially the same as adding another pair of coordinates at the end of s .

c - Close b-spline

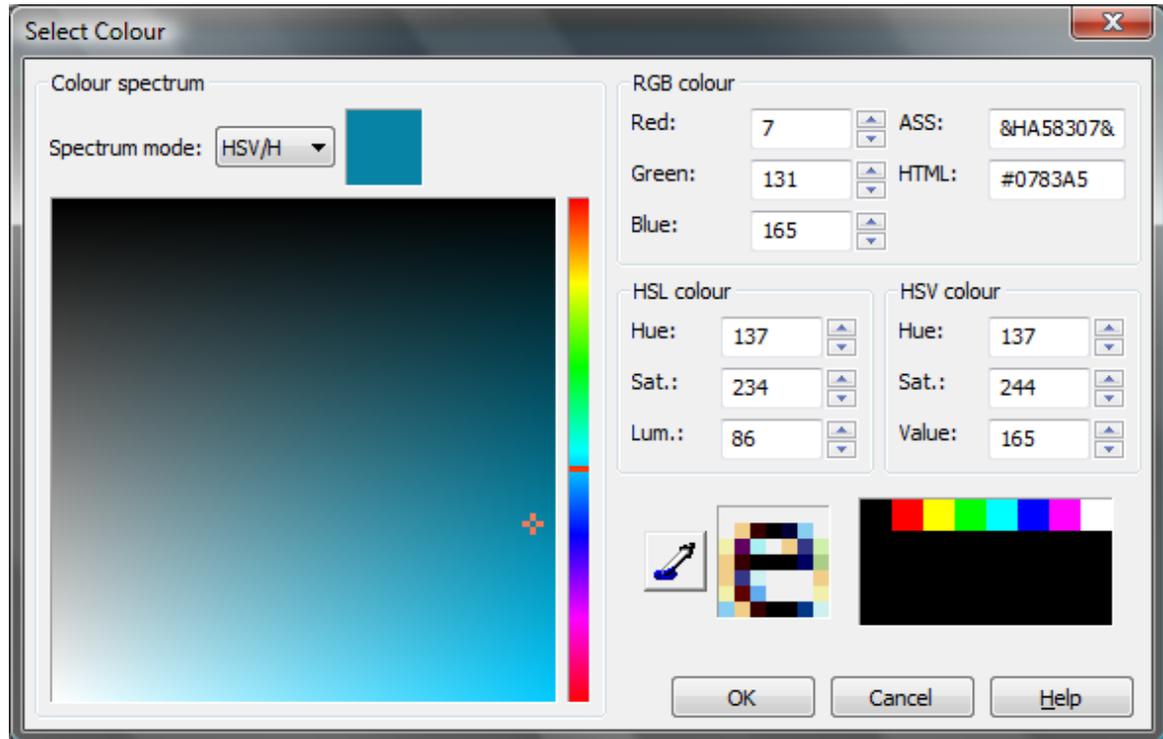
Closes the b-spline.

Category: Pages with Todo items

Colour Picker

Aegisub introduces a more advanced color picker than the default Windows one. It allows you to enter color values using various color spaces, in both HTML and ASS hex, pick colors from a mini-screenshot and use three different color spectrums for graphically picking colors.

Overview



The color picker has the following components:

- Color spectrum
- Parametric settings for four color-spaces
- HEX input for ASS and HTML formats
- Pick-from-screen control, for taking a mini-screenshot and picking a color from it
- List of 32 last picked colors

The last used mode of the color spectrum is remembered.

Using the pick-from-screen function

1. The pick-from-screen control is located to the left of the last-used colors, in the lower part of the window.
2. To start using it, click on the "pick color" symbol.
3. As you drag over the screen, you'll notice that the square box beside the pick color button turns into a magnification of the area the cursor is hovering over.
4. When you've found the area you want to grab a color from, click the left mouse button to fixate the magnification.
5. Now click on the pixel you want to grab a color from in the magnification.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Descriptions of the color spaces

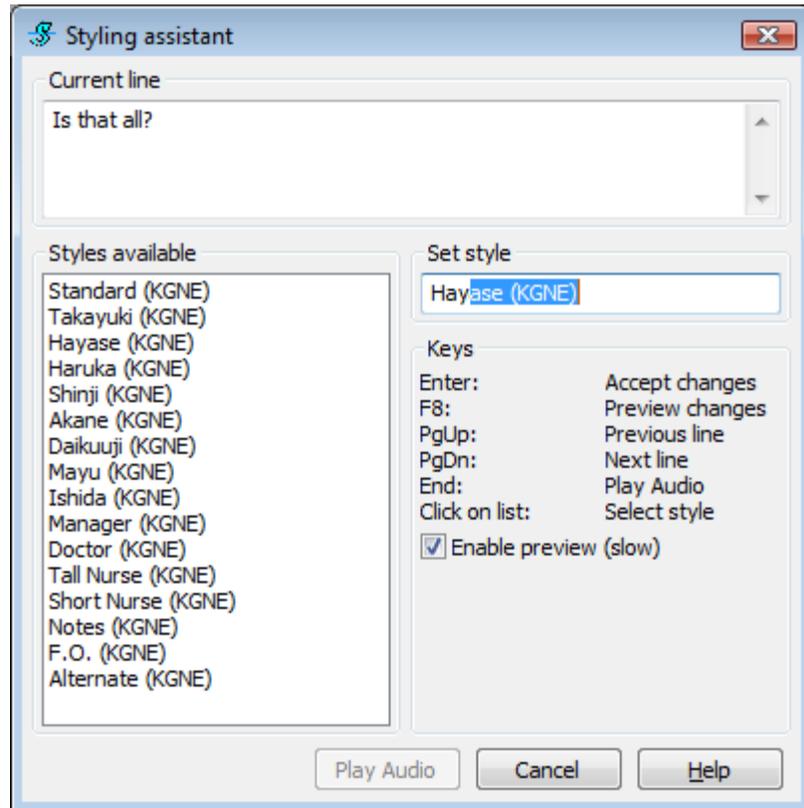
Here's a short round-up of the three available color spaces:

- *RGB* - Red, Green and Blue, the three component colors a computer monitor uses to display images. In RGB mode, you specify the intensity of each of those three components.
- *HSL* - Hue, Saturation and Luminance. Hue is the actual "color of the color", that is, red, green, blue or a mix of those. Saturation is the "greyness" of the color, the lower the saturation, the closer to a shade of grey, the color becomes. Luminance is the lightness, max luminance means pure white, while zero luminance means pure black.
- *HSV* - Hue, Saturation and Value. Hue and Saturation mean the same as in HSL. Value, however, is different from the luminance in HSL. Value is the "non-blackness" of the color, the smaller the value, the closer the color is to pure black.

Styling Assistant

The Styling assistant is a powerful tool for subtitles that have multiple styles, such as cases when several actors have their own style, or for some other reason you have many different styles for dialog lines.

Overview



As seen from the image above, the assistant is composed of the following:

- Current line field
- Styles available field
- Set style window
- Keys field
- Enable preview checkbox
- Play audio button

The Current line field displays the line in the subtitles for which the style will be set. The text in it cannot be edited.

Styles available field displays the styles available in the script. The styles have to be imported through the Styles Manager or created with its help. The styles can then be set for the selected line in the subtitles by clicking on the desired style.

The set style window is the main aspect of the Styling Assistant. Through it, styles can be applied to the line in the subtitles easily and without much work. It support autocomplete, so it helps to select the desired style after some characters are written in the window.

The Keys fields displays the keyboard shortcuts that are used for work with the assistant.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Enable preview checkbox is to enable a video preview for the subtitles. The video jumps to the first frame of the selected line.

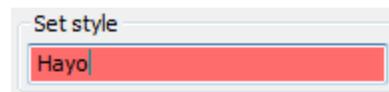
Play audio button helps the typesetter to determine to which actor a line goes. It plays the audio of the line for which the style will be applied.

The styling assistant displays the video for the line if the checkbox is selected. When the style for a line is set via the Styles available or with the Set style window the Styling Assistant moves to the next line as seen in the image.

The autocomplete function



This window displays how the autocomplete function works. The characters that were typed are on the white background and the rest of the name is displayed as being selected. The selected part is what autocomplete sets as the style that matches with the typed characters.



This image shows the response to a name that doesn't exist. The background of the window is colored red to show that no style on the list of selected styles matches the entered text.

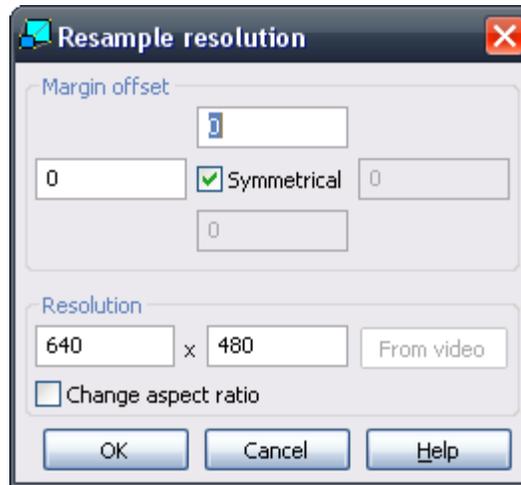
How to use

Usage is simple. Select the first line you want to style and open the assistant. Then you look at text, video and/or audio to determine who is speaking it and type first few letters of style name, after which autocomplete should kick in. Hit enter to go to next line and proceed until you're finished. The other way is to double click on a style in the Styles available list and the style will be applied to the selected line.

Resolution Resampler

The Resolution Resampler is a built-in tool designed to change the resolution of the script while transforming all affected tags so it ends up looking the same. It can also offset all margins (and absolute positions) by a specific value. Uses include merging two scripts with different resolutions, converting a script for a 4:3 video to its 16:9 equivalent, and converting between 1:1 and anamorphic pixel formats.

Overview



There are two static boxes in the dialog, "margin offset" and "resolution". The upper box controls how much you want to increase each margin by. The second box indicates the final resolution to which the script is resized.

It's important to keep in mind that margin offset happens BEFORE the resolution is changed, so the values are relative to the current script resolution. If you check the "symmetrical" checkbox, then the left and right values will be identical, and the same for top and bottom. Positive values indicate an increase in margin (that is, subtitles farther from the video border), and negative values a decrease.

After the margins were offset, the resolution is then resampled to the final value provided. You can click the "From video" button to copy the actual video resolution to the two fields. The "Change aspect ratio" check box indicates whether the horizontal size of subtitles should be affected to conform to the aspect ratio change. This is useful when converting between anamorphic and 1:1 formats.

Examples

4:3 SD to 16:9 HD

For example, if you have subtitles typeset to a 640x480 video, and want to apply the same subtitles to a 1280x720 video (which is widescreen, therefore showing more video on the left and right margins), you would set margin left to 107, while leaving top at 0 and "symmetrical" checked. Then you would set final resolution to 1280x720 (leave "Change aspect ratio" unchecked), and hit OK.

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)



The above 107 value for the left/right margins was obtained with the following formula:

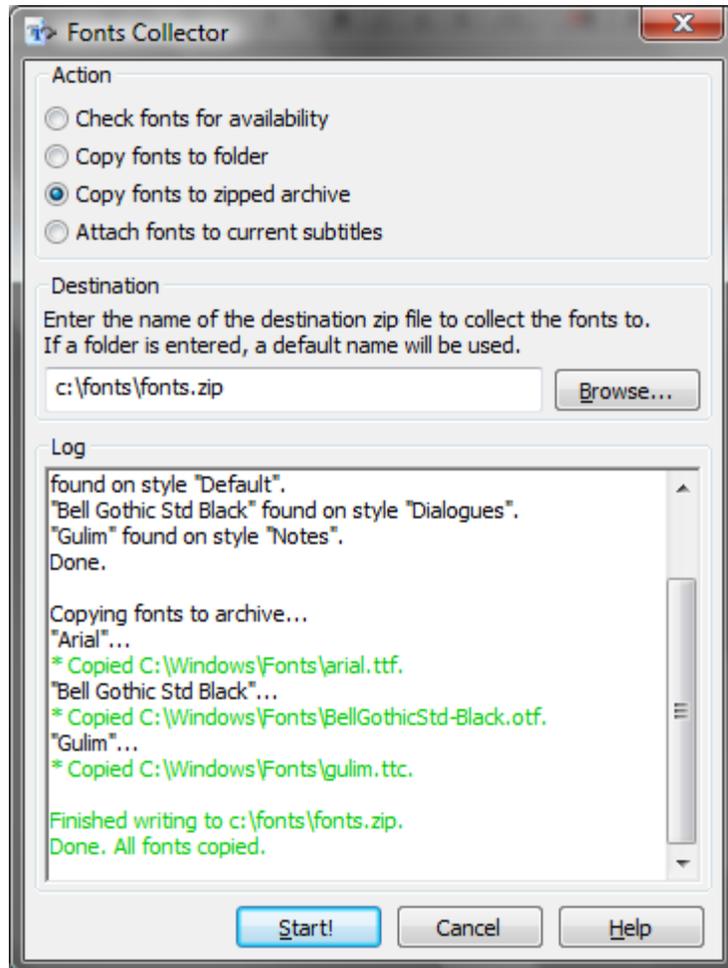
$$(\text{OrigH} * \text{FinalW}/\text{FinalH} - \text{OrigW})/2$$

Which results in:

$$(480 * 1280/720 - 640)/2 = 107$$

Fonts Collector

The fonts collector is an automated tool that finds all fonts used in the current script, scans your system's fonts folder(s) and copies all the relevant fonts to a destination of your choosing. The dialog box looks like so:



The collector can perform any of the following actions:

- **Check fonts for availability** - Checks the fonts folder(s) to see if you have all the fonts used in the current script. Do note that for technical reasons, the collector can't detect or collect fonts that aren't actually installed (like for example fonts temporarily loaded by a Matroska splitter or a font viewing program).
- **Copy fonts to folder** - Copies all the fonts used on the current script to the given folder.
- **Copy fonts to zipped archive** - Copies all the fonts used on the current script to a compressed .ZIP archive.
- **Attach fonts to current subtitles** - Attempts to attach all used fonts to the currently open subtitles file. See the [attachment manager](#) page for more information about the quirks and limitations of this feature.

The fonts collector will output diagnostic information about its doings in the lower half of the window.

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Audio

Aegisub has a fairly advanced, customizable audio mode with both the traditional waveform display as well as an alternative spectrum display. Several different timing modes are available for both normal dialog timing and karaoke timing.

Todo: This page should probably be split into several smaller ones to make it easier to digest, easier to link, less confusing and wall-of-text and to promote going more in depth with the separate topics.

Opening audio

To load an audio file into Aegisub, just go to the *Audio* menu and press *Open audio file*. If you have a video file (with an included audio track) already loaded, you can use *Open audio from video* instead, which obviously will load the audio track from the video file you currently have loaded. You can open any type of audio file that your [audio provider](#) can decode, more on this below.

Supported formats: Windows

Under Microsoft Windows, your audio provider is *Avisynth* by default, which means that any audio format that your DirectShow environment knows how to decode is supported (at least in theory). For example, if you want to load an AC3 file, you will need an AC3 DirectShow decoder (e.g. AC3filter or ffdshow). *Note:* some formats seem pretty buggy at the moment. Ones more or less guaranteed to work are (16-bit) PCM-WAV, MP3 and Vorbis, so if your audio doesn't work, try transcoding to one of them, at least temporarily.

Warning: If you have opened a video file with more than one audio track (most commonly an MKV or OGM file), and try to open audio from it, Aegisub is completely at the mercy of the splitter when it comes to what audio stream is delivered. Some splitters may deliver both audio streams at once (this will happen for dual audio AVI's, when using the default Windows splitter), and since Aegisub very much doesn't expect that, you will get weird results (and probably crashes). Just remux the file to single audio, or better yet, decompress the desired audio stream to WAV.

Supported formats: non-Windows

On all other operating systems (MacOS X, GNU/Linux, the BSD variants etc.) your audio provider is *ffmpeg*, which means you can use any audio format that *ffmpeg* supports (and was compiled with).

Audio caching

If you're loading any audio format that isn't an uncompressed (PCM) Microsoft WAV file, Aegisub needs to decode and cache it first. When loaded, the audio is downmixed to mono (see the [audio downmixer option](#) if you want to grab one channel only instead), decompressed to PCM (a.k.a. WAV), and (by default) loaded into a RAM cache. This means that you will need a *large amount* of RAM to open a long compressed audio file. If your computer doesn't have a lot of RAM, or if you're working with a full-length movie, refer to the [audio cache option](#) for instructions on how to make Aegisub use its (slower) hard drive cache instead; or decompress the file to WAV first since Aegisub can read from WAV's directly without need for caching.

The exact amount of memory used for any given audio file can be calculated with the following formula:

Contents [hide]

- 1 Opening audio
 - 1.1 Supported formats: Windows
 - 1.2 Supported formats: non-Windows
 - 1.3 Audio caching
- 2 The main audio view
- 3 Basic audio timing
 - 3.1 Timing protips
 - 3.2 The spectrum analyzer mode
- 4 Karaoke timing

$$s = (b * r * l) / 8$$

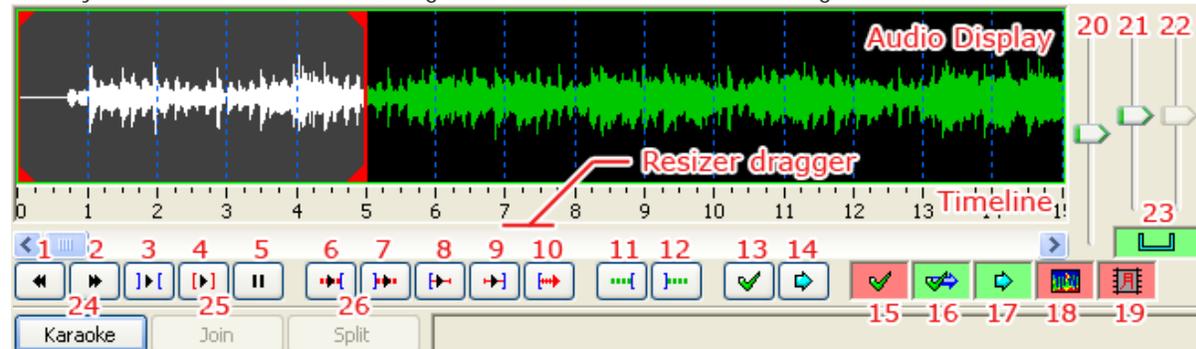
where s is the amount of memory (in bytes - divide by 1024 to get kB), b is the number of bits per sample (always 16 in the current implementation), r is the sample rate in Hz (usually 48000, or 44100 in some cases), and l is the length of the audio (in seconds).

For example, for a 25 minute audio clip at 48 kHz, you will need $(16 * 48000 * 25 * 60) / 8 = 144000000$ bytes \approx 137 MB.

Loading and decompressing the audio into the cache will take a few seconds; Aegisub will display a progress indicator while loading the audio.

The main audio view

When your audio file has loaded, Aegisub will transform into something like the screenshot below:



You can click and drag just below the audio timeline to change the height of the audio waveform/spectrum display.

Green and red buttons are toggle buttons. A green background indicates that the option is turned on, while a red background indicates that the option is turned off. The buttons and controls are as follows (many of these have [keyboard shortcuts](#) associated with them by default):

1. Go to previous line, discarding any unsaved changes (previous syllable when in [karaoke mode](#))
2. Go to next line, discarding any unsaved changes (next syllable when in [karaoke mode](#))
3. Play selected area of the audio waveform
4. Play currently selected line
5. Pause playback
6. Play 500ms before selection start
7. Play 500ms after selection end
8. Play first 500ms of selection
9. Play last 500ms of selection
10. Play from selection start to end of file (or until pause is pressed)
11. Add lead-in (how much is determined by the [audio lead in setting](#))
12. Add lead-out (exactly like the above, but the setting is called [audio lead out](#), logically enough)
13. Commit (save) changes
14. Scroll view to selection/go to selection
15. Toggle auto-commit (all timing changes will be committed immediately, without the user pressing commit, if this is enabled)
16. Toggle auto next line on commit (if this is enabled, Aegisub will automatically select the next line when the current line is committed; enabling both this and auto-commit at the same time is strongly discouraged)
17. Toggle auto-scrolling (will center waveform on the currently selected line automatically when enabled)
18. Toggle spectrum analyzer mode (see below)
19. Toggle Medusa-style timing shortcuts
20. Audio display zoom (horizontal)
21. Audio display zoom (vertical)
22. Audio volume
23. Toggle linking of vertical audio zoom slider with volume slider
24. Toggle [karaoke mode](#)
25. Join selected syllables ([karaoke mode](#) only)

Basic audio timing

When you click on a line in the subtitles grid, Aegisub will highlight it in the audio display and, if you have auto-scrolling enabled, scroll the audio display so it's centered on the line (during normal timing, it's probably a good idea to disable auto-scrolling). You'll notice various vertical lines in the audio display; the dark blue ones indicate second boundaries, the pink ones indicate keyframes in the video if you have it loaded (see the [Working with video](#) section), the white broken line indicates the currently visible video frame, and the thick red and orange ones are the line start and end markers (respectively) for the current line. To (re-)define the start and end times of the line, you can either left-click to set the start time and right-click to set the end time, or just drag-and-drop the line boundaries. The selection background will turn red and display the word "Modified" in the top left corner of the audio display when you've changed the timing but haven't saved the changes yet. It will remain red until you either press the commit button (*enter* or *g* by default) or go to another line (discards changes). If you have auto-commit on, the background will never turn red since all changes will be saved immediately. Press the *play* button (keyboard shortcut *s* by default) to listen to the selection, or the various other playing buttons to listen to parts of the selection or the audio surrounding it. When you are satisfied with the timing, press commit. Then repeat once for every line; it's as simple as that.

Timing protips

If you want to finish timing your movie or episode within any reasonable amount of time, there's some things you should note:

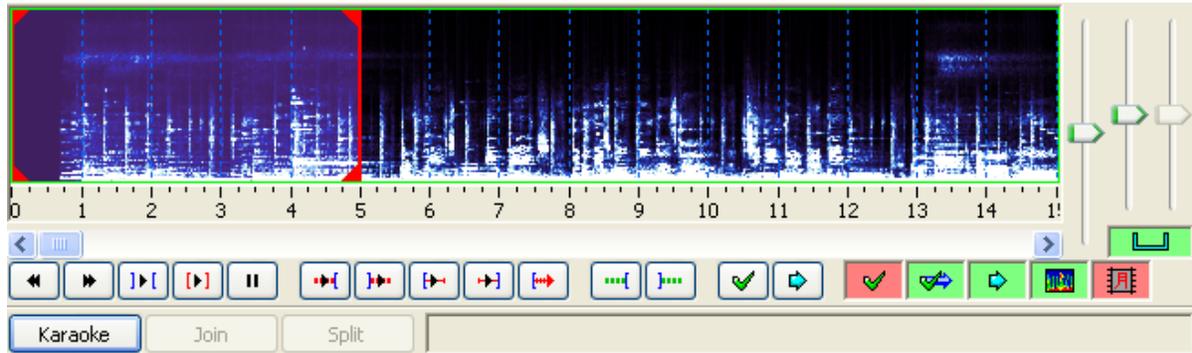
- Use keyboard shortcuts! They speed up your work by several orders of magnitude.
- You don't need to have video displayed while timing. Scene-timing, i.e. syncing line start/ends to scene changes, can be done later. Either manually, or with the [timing postprocessor](#).
- Use "go to next line on commit".
- Experiment with different timing styles when you're new and stick to one that suits you. Then practice. Lots.
- Aegisub heavily relies on the concept of "focus", and doing things in a way that require you to switch back and forth between video/audio/subtitle edit box a lot will cost you a lot of time. Do it in several "passes" instead.
- The spectrum analyzer mode can make it a lot easier to "see" where lines start and end.

One common timing style (preferred by the author of this page) goes something like the following: Turn on "go to next line on commit" but disable auto-committing, auto-scrolling and Medusa timing shortcuts. Keep the four main fingers of your left hand on *s/d/f/g*. You won't be using the thumb so do whatever you want with it. Keep your right hand on the mouse. Now select (by left- and right-clicking) an area in the waveform that seems likely to contain a line of speech matching the current subtitle line, and hit *s* to play it back. While it's playing, adjust the start time if necessary. When the playback marker has passed the end time mark, adjust the end time as well. If greater accuracy is needed, play the last 250ms of the selection by pressing *d*, 250ms before the selection start by pressing *q*, 250ms after the selection end by pressing *w*, or the first 250ms of the selection by pressing *e*. As you grow more experienced, you won't be using anything else than *s* very much, except maybe *d* and *q*. When you're satisfied with the timing, hit *g* to commit changes and go on to the next line. Scroll the audio display forward by pressing *f*. If you need to scroll it backwards, use *a*. To go to next or previous line without committing changes, use *z* and *x*.

This style has the advantage that you never need to move your hands at all. With some training, it can also be very fast; audio timing 350-400 lines of dialog to a 25-minute episode can easily be done in less than 40 minutes.

Of course, this style may not feel comfortable for all people; you should experiment with other timing styles before deciding which one is best for you.

The spectrum analyzer mode



When you press the spectrum analyzer button, the waveform does no longer show amplitude (signal strength) on the vertical axis - instead it shows frequency. The higher up, the higher the frequency. The colors instead indicate amplitude, with black/dark blue being silence and white being the strongest sound. This may seem confusing, but since the frequency window is set to fit human voices rather well, it can make it easy to tell where a line (or a word in karaoke mode) starts and ends when there's a lot of background noise (or music) that makes it hard to tell from the normal waveform. It can be especially useful when timing karaoke. Play around with it for a little while, and you'll understand how it works.

Note that in spectrum analyzer mode, the "vertical zoom" slider is redefined to control color intensity instead, since the colors indicate signal strength.

Because calculating the spectrum data is very CPU intensive, it is initially set to be in a medium quality. You can increase the quality of the spectrum in the [audio options](#).

Karaoke timing

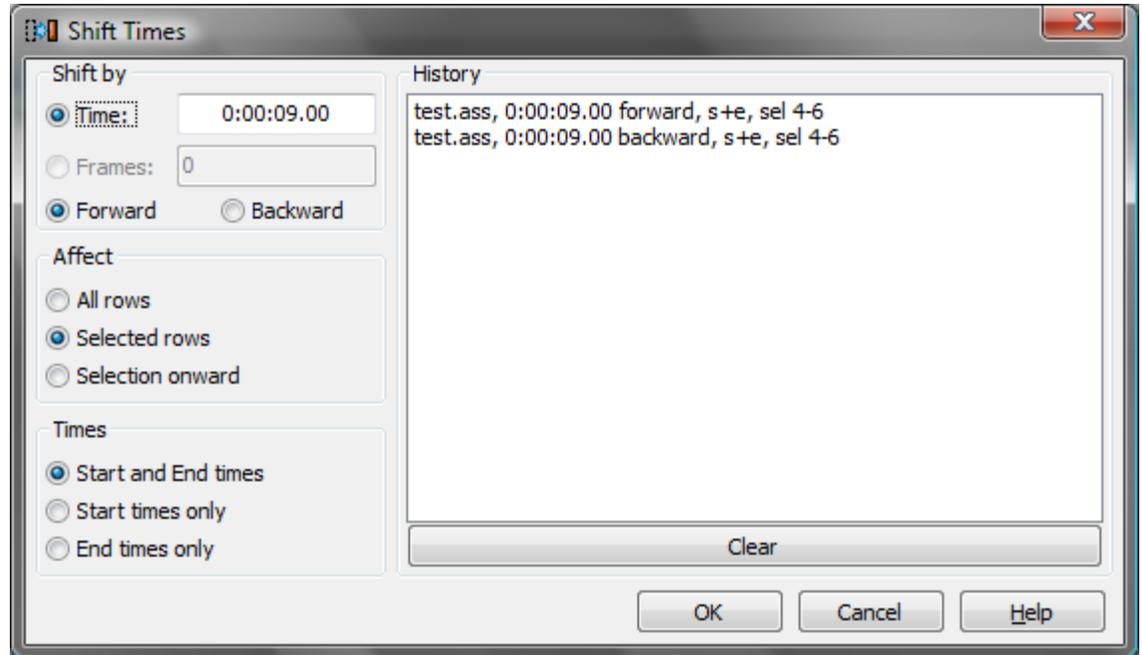
Todo: here be dragons

Category: [Pages with Todo items](#)

Shift Times

The **shift times** tool is a batch processor for timestamps; it adjusts the start/end timestamps of many subtitle lines at once in various ways. It is located under *Timing* -> *Shift Times*.

It looks like this:



The left part of the window contain the options.

Shift by

These controls decide in which direction and by how much each timestamp will be modified.

- **Time** - How much you want to adjust each timestamp, in hours:minutes:seconds.centiseconds.
- **Frames** - If you have video loaded, you can specify the adjustment time as a number of frames instead.
- **Forward** or **Backward** - controls in which direction the timestamps are adjusted.

Affect

These controls decide what lines will be processed.

- **All rows** - Applies the time shifting to all lines in the script.
- **Selected rows** - Applies the time shifting only to the selected lines.
- **Selection onward** - Applies the time shifting to the selected line(s) and all lines below (in the grid) the last selected line.

Times

These controls decide what timestamps will be processed.

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Contents [\[hide\]](#)

- 1 [Shift by](#)
- 2 [Affect](#)
- 3 [Times](#)
- 4 [History](#)

- **Start and End times** - Both start and end times of the affected lines will be modified by the given amount.
- **Start times only** - Only the start times of the affected lines will be modified. Note that this makes the lines longer (if you shift backwards) or shorter (if you shift forwards) and can even make them have a duration of zero.
- **End times only** - Only the end times of the affected lines will be modified. Note that this makes the lines longer (if you shift forwards) or shorter (if you shift backwards) and can even make them have a duration of zero.

Note that if a line is shifted so that its start or end time stamp would be negative, that timestamp is zeroed instead. This can be used to clear all timings from an entire script, by shifting backwards by longer than the latest timestamp in the script.

History

This is a history of all time shiftings you have done since last time you cleared the shift history (with the clear button). The format is a number of fields separated by commas. The fields are:

- Filename of the script (e.g. "example.ass")
- Shift amount and direction (e.g. "0:00:05.00 forward")
- What times were affected, "s" for start, "e" for end, "s+e" for both
- What rows were affected; "sel start-end" for selections, "all" for all rows (e.g. "sel 1-40")

Timing Post-Processor

The timing postprocessor is a highly useful tool for automatically correcting timing in various ways.

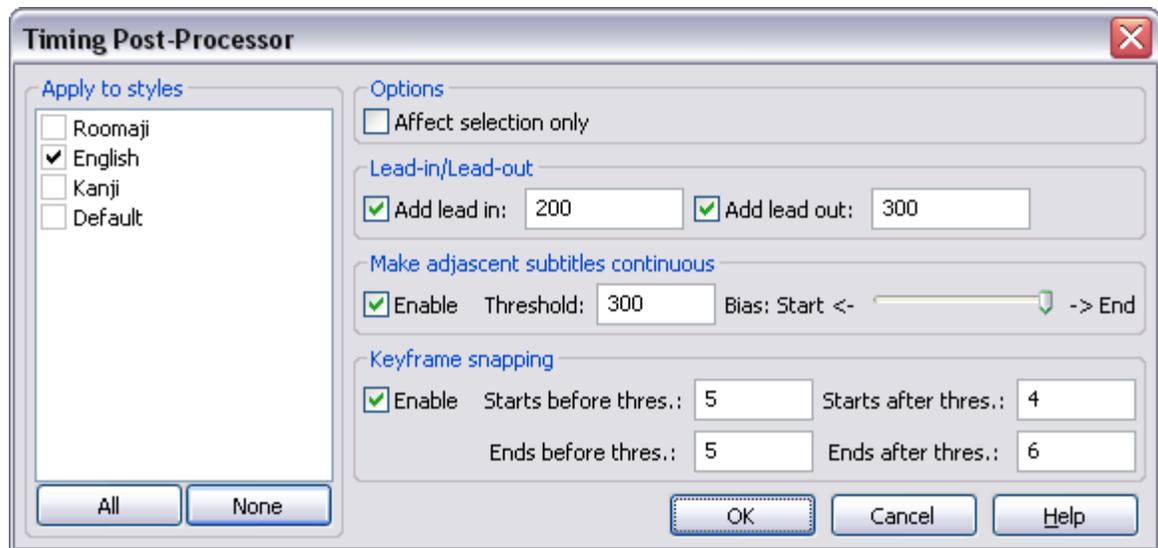
Overview

As shown in the screenshot, there are three functions:

- Add lead-in and/or lead-out
- Make lines that start close to each other continuous by extending their start and/or end times
- Snapping line starts/ends to video keyframes (only available if a video is loaded)

Contents [hide]

- 1 Overview
- 2 Apply to styles
- 3 Options
- 4 Lead-in/Lead-out
- 5 Make adjacent subtitles continuous
- 6 Keyframe snapping



The processing gets applied in the order it is displayed in the dialogue box. That is to say, first lead-in/out are added, then a check for adjacent lines is performed and lines that are closer than the given threshold are made continuous, and lastly, line start/ends get snapped to keyframes.

Apply to styles

This field determines which styles will get processed - check all you want to process. This is useful for only processing dialogue lines while leaving signs and/or karaoke alone.

Options

Check "Affect Selection Only" to restrict the operation to the selected lines. If this is left unchecked, all lines in the file whose style match the selected ones will be affected.

Lead-in/Lead-out

Navigation

» [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

This function extends the start/end times of the line, a procedure known as adding lead-in and lead-out. The postprocessor will add the given time (in milliseconds) to each line start and end, respectively. You can add both lead-in and lead-out, only one, or none, by checking and unchecking the boxes as appropriate.

Make adjacent subtitles continuous

This function will check if any two lines' starts and ends are closer in time than the given threshold (in milliseconds). If they are, one or both will get their start and/or end time moved so that they are continuous (i.e. one appears directly following the next, without any subtitle-less frames in between).

The *Bias* slider determines how the lines are extended. Sliding it all the way to the right will extend the end time of the first line all the way to the start time of the second, without touching the second line at all. Sliding it all the way to the left will instead make the start time of the second line extend backwards to the end of the first line, without touching the first line at all. Putting it in the middle will extend the end time of the first line and the start time of the second equally, so that they meet in the middle. Anything in between will makes the lines "meet" where the slider is, so to speak. For example, if the threshold was 1000, and the slider was 3/4ths of the way to the right (roughly as shown in the screenshot), the end time of the first line would get extended by 750 ms, and the start time of the second extended backwards with 250 ms.

Keyframe snapping

The keyframe snapping function is a kind of automatic scenetimer. It is probably the most useful of the three, but will only work if there is a video or keyframe loaded, because of its dependency on keyframes. See the [keyframes section of the working with video page](#).

The keyframe snapping function will look at how close the start and end of lines are to the nearest keyframe, and if they are closer than the given threshold, it will get extended or shortened to the keyframe.

There are four thresholds to consider:

- *Starts before* - if the line starts less than this many frames (inclusive) *before* a keyframe, its start time will get moved forward so that the line starts on the keyframe.
- *Starts after* - if the line starts less than this many frames (inclusive) *after* a keyframe, its start time will get moved backward so that the line starts on the keyframe.
- *Ends before* - if the line ends less than this many frames (inclusive) *before* a keyframe, its end time will get moved forward so that the line ends on the frame before the keyframe.
- *Ends after* - if the line ends less than this many frames (inclusive) *after* a keyframe, its end time will get moved backward so that the line ends on the frame before the keyframe.

When using this feature, remember your lead-in/out times, and the framerate of the video! At 23.976 frames per second, each frame lasts for about 42 ms. This means that if you have a lead-in of 120 ms, you can safely set the "starts before" threshold to at least 3 ($3 * 42 = 126$) without fear of lines starting showing up after someone has started speaking. Using the same reasoning, a lead-out of 250 ms allows you to safely have at least 6 as your "Ends after" threshold, without fear of lines ending before people stop talking. Figuring out good values for the other thresholds and/or framerates is left as an

exercise for the reader.

Another thing you can do with the keyframe snapping feature is using it to correct one-frame bleeds really, really fast. If your script is full of them, just set all the thresholds to 1 or 2, disable the lead-in/out adding and the adjacent line snapping, choose your dialogue style, and hit Apply. Problem solved.

Kanji Timer

The kanji timer makes it easy to copy timed karaoke from one set of already karaoke timed lines to another set of yet untimed lines. Primarily intended for adding karaoke to Japanese songs.

Here is a video demonstration of the kanji timer in action: [Download demonstration video](#) (XviD MP3 AVI, 5 min 20 sec, 12 MB)

Before you start

The kanji timer attempts to pair syllables in one (timed) input line with a given style with syllables in a corresponding (untimed) line with another style. In other words, the timed input lines should all have same style ("romaji" for example) and the untimed ones another ("kanji" for example). Having more timed source lines than untimed destination ones or vice versa can confuse the tool, but it's usually possible to un-confuse it, see below. In order to correctly pair source lines with destination ones, both sets must be in the same order in the grid. For example, this works:

```
Timed line 1
Untimed line 1
Timed line 2
Untimed line 2
```

And so does this:

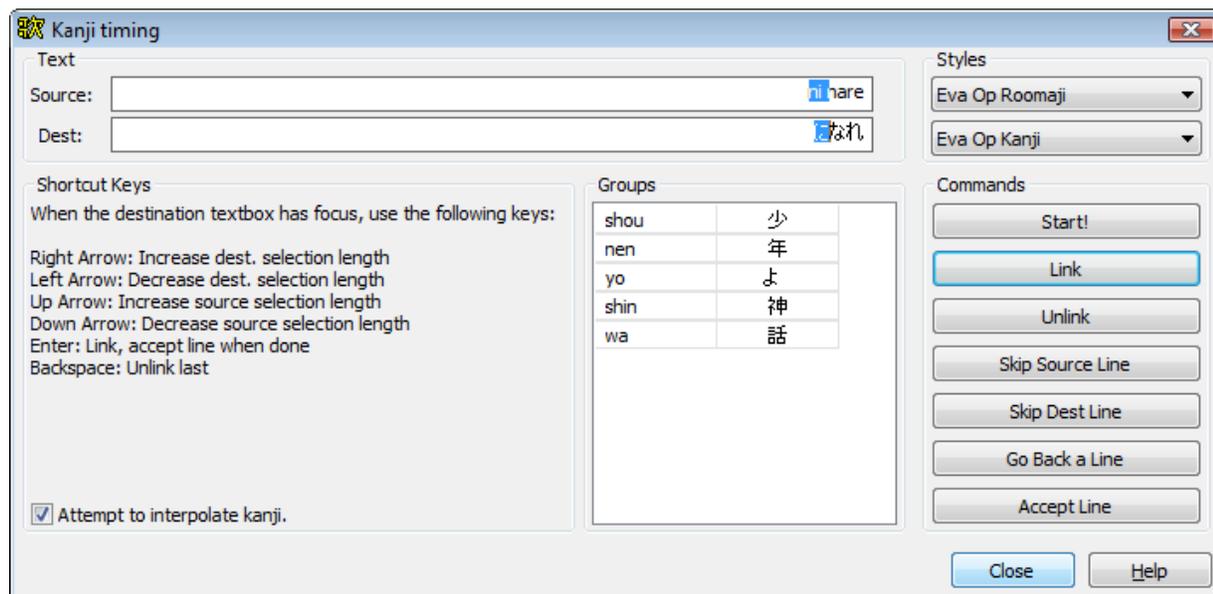
```
Timed line 1
Timed line 2
Untimed line 1
Untimed line 2
```

But this does not (pairs lines wrong):

```
Timed line 1
Timed line 2
Untimed line 2
Untimed line 1
```

Using the kanji timer

The kanji timer dialog looks like this:



The first thing you need to do is to select which styles are used for the timed input lines and the untimed output lines respectively. This is done in the upper right corner of the dialog; the top dropdown is the source

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

style and the bottom one is the destination. When you've done this, click the start button.

Now, you'll see the first syllable of the first source line highlighted in the source text field, and a suggestion for the destination syllable highlighted in the destination field. What you do now is "group" each source syllable with one or more kanji (or other syllables) in the destination. This is done using the following keyboard shortcuts:

- **Enter** - accept the highlighted grouping (and go to next line if all syllables are grouped).
- **Right arrow** - increase the destination highlight length.
- **Left arrow** - decrease the destination highlight length.
- **Up arrow** - increase the source highlight length.
- **Down arrow** - decrease the source highlight length.
- **Backspace** - un-groups (or unlinks) the last accepted syllable and lets you try to group it again (useful if you make a mistake).

Things to note

- Don't use the mouse to change the highlights; it confuses the tool a lot. Use the keyboard shortcuts instead, they're much faster.
- The destination line can already be k-timed, but if it is, the kanji timer will overwrite those timings.
- Empty syllables will be copied alone, or will be combined with the surrounding syllables if those are to be combined.
- Any ASS override tags appearing before each \k will be copied directly without modification, but tags after each \k is currently not copied at all.
- If you have more source than destination lines or vice versa, you can use the "Skip source line" or "Skip destination line" to make sure the pairing of source/destination lines is correctly done.

Automation

The name **Automation** covers the entire scripting functionality of Aegisub.

Contents [hide]

- 1 About Automation
- 2 Using Automation
- 3 Automation for programmers
 - 3.1 Future goals

About Automation

The single purpose of Automation is — as the name implies — to automate various aspects of subtitle creation and editing. To date this has mainly been for creating **karaoke effects** but Aegisub 2 introduces **Automation 4** which also allows creating more general purpose **macros** and also changing the style definitions and other meta data for the subtitles.

Some of the goals of Automation:

- Macros to automate complex subtitle editing tasks
- Export filters to generate complex effects from simpler input
 - Karaoke effects
 - Translation note boxes
- Probably several yet-undiscovered uses

Using Automation

Aegisub comes with several Automation scripts already packaged and ready to use. This includes the advanced **Karaoke Templater** script (which covers everything *simple-k-replacer*, *line-per-syllable* and *multi-template* did in Automation 3) and a **collection of macros** to simplify some editing tasks.

Aegisub almost always has several Automation scripts loaded at a time. You can see what scripts are loaded, and load/unload more from the **Automation Manager** window.

All Automation scripts also present themselves in Aegisub in one way or another. Some appear as **macros in the Automation menu** and others appear as **filters in the Export dialog**. Some scripts even appear in both places.

Automation for programmers

Automation offers several scripting languages that can be used. The main "featured" language is **Lua 5.1** but **Perl** and **Ruby** engines are also available.

Other languages have also been proposed but haven't been implemented yet: JavaScript, Python, Haskell and perhaps support for native dynamic library code (DLL's, so's, dylib's.)

There is several example scripts included with Aegisub to give you a head start on writing your own ones. A word of warning: Unless you are an experienced programmer the *kara-templater.lua* script is a very bad place to start!

In addition, the Automation 3 scripting interface used in Aegisub 1.10 and earlier is also supported but using the Automation 4 Lua interface is recommended. The Automation 3 environment is reproduced as closely as possible, meaning Lua 5.0 is used instead of Lua 5.1 and all data structures are kept the same. There are however a few minor differences, see more at **Moving from Automation 3**.

Navigation

» [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

If you have previous experience with writing Automation 3 scripts, there is also an [overview of changes from Automation 3](#).

Future goals

Apart from the addition of more languages as mentioned above, one major future goal of Automation is to allow *custom file format handlers* enabling you to make Aegisub read and write almost every imaginable file format so they can be edited with Aegisub.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Karaoke Templater

Karaoke Templater is an [Automation](#) script that ships with Aegisub. Its primary purpose is to help creating [karaoke effects](#) with a specially designed template language. Karaoke Templater is already installed and ready to use along with Aegisub.

Tutorials: Introducing Karaoke Templater

- [A simple example](#)
- [Using math expressions](#)
- Using multiple template lines
- More advanced effects with positioned syllables

Todo: Plan more tutorials. Also actually write those above.

Reference

- [Declaring template and code lines](#)
- [Rules for when and in what order templates are run](#)
- [Template modifiers](#)
- [Inline variables \(dollar-variables\)](#)
- [Rules for code blocks and code lines](#)
- [Contents of the code block/line execution environment](#)

Also see the [Automation 4 karaskel.lua](#) section for more information on what's in the *line* and *syl* variables, and more.

For users of *multi-template*

If you have used the *multi-template* script from Aegisub 1.10 you should recognise several similar concepts in the karaoke templater, but there are also several pitfalls. Some of them are:

- You no longer declare template lines in the Actor field but in the Effect field instead. You can also put much more than just *template* in there. Read the tutorials above for an introduction, or the reference below if you feel adventurous.
- Instead of using percent-signs to write Lua code blocks you use exclamation marks. So write *!\$start+\$i*30!* instead of *%%\$start+\$i*30%*.
- The *A* global is gone, but *line* and *syl* are directly accesible. The escaped Lua code is no longer run in the true global environment but instead in its own environment, so clashes between your templates and Karaoke Templater itself is much less probable.
- The *return false* hack to cancel execution of a template no longer works. Neither does having multi-statement Lua blocks and returning from them in general. For the first purpose the *fxgroup* functionality has been introduced, and for your multi-statement needs code lines have been introduced.
- Instead of working with *newline* and *line* (for being-generated and original line) you now work with *line* and *orgline* for being-generated and original lines.
- The *retime* function has been introduced to make it much easier to control the start and end times of your generated lines.
- Lots of more fancy features. Check the tutorials or read the reference to learn about it all.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Automation 4

Overview:

[Automation Manager](#) • [Running macros](#) • [Using export filters](#) • [Standard macros](#) • [Changes from Automation 3](#) • [Moving from Automation 3](#)

Karaoke
Templater
reference:

[Declaring templates](#) • [Execution order](#) • [Modifiers](#) • [Inline-variables \(\\$-variables\)](#) • [Code lines and blocks](#) • [Execution environment](#)

Lua reference:

[Registration](#) • [Subtitles object](#) • [Progress reporting](#) • [Config dialogues](#) • [Misc. APIs](#) • [karaskel.lua](#) • [utils.lua](#) • [unicode.lua](#) • [cleantags.lua](#)

Karaskel
concepts:

[Style tables](#) • [Dialogue line tables](#) • [Syllable tables](#) • [Inline effects](#) • [Furigana](#)

Category: [Pages with Todo items](#)

Lua Reference

The Automation 4 Lua scripting engine is based on the version 5.1 series of [the Lua scripting language](#).

This manual will not deal with the Lua language itself nor the standard libraries shipping with Lua, but only the additional functions and data structures provided by the Aegisub Automation 4 Lua interface. Please see [the Lua 5.1 manual](#) for details on the language itself and its standard libraries.

General overview of the organisation of Automation 4 Lua

The smallest legal Automation 4 Lua script is an empty file, but that won't be able to do anything interesting.

There's a number of global variables a script can set to provide information about itself. This information will be displayed in the [Automation Manager](#) window: `script_name`, `script_description`, `script_author` and `script_version`.

Automation 4 Lua implements both of the currently defined "features" of Automation 4: Macro and Export Filter. One script (one file) can define zero, one or many of each of those features. For example, the Karaoke Templater script defines one macro and one export filter.

When an Automation 4 Lua script is loaded, its top-level code is executed once. You can put variable initialisations and such at the top level, but what you usually will do is define some of the script information globals, write some functions that do the script's work and then call the feature registration functions. This is described on the [Registration](#) page.

When the user activates a feature from the Aegisub interface (such as by selecting a macro from the Automation menu) the registered script function is run. One of the parameters passed to the function is a *subtitles object*, the primary interface to the subtitle data the script will manipulate. To some extent, the subtitles object works as an integer-indexed array, but it exposes some special interfaces to add, remove and modify subtitle lines. The subtitles object allows you to access every line in the subtitle file, including headers, style definitions, dialogue lines and comment lines. This is described on the [Subtitle file interface](#) page.

Automation 4 Lua also provides a number of helper functions in the core API for getting information on eg. the video frame timestamps and how large a piece of text will be when rendered with a given style.

Most things that can be implemented in clean Lua code, ie. that doesn't depend directly on Aegisub internal data structures, has been implemented outside the core API as include files. While it is possible to write Automation 4 Lua scripts without using the provided standard include files you will find that for anything but the simplest scripts you will need some of the functions provided by the includes. See below for an overview of the standard include files.

The Automation 4 Lua core API

Automation 4 Lua provides various APIs that can be grouped in these general categories.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [Script and feature registration](#) - deals with advertising what features a script provides and a few other script meta data.
- [Subtitle file interface](#) - deals with use of the *subtitles* object, the principal way of accessing and modifying the subtitle data.
- [Progress reporting and debug output](#) - providing feedback to the user while a script is running, outputting hints and warnings to the user and printing debug information.
- [Displaying dialogue boxes and getting user input](#) - requesting user input during macro execution by dialogue boxes and providing a configuration interface for export filters.
- [Miscellaneous APIs](#) - for eg. getting the rendered size of text and getting video frame rate information.

Standard include files

A number of standard include files are provided. These aren't part of the core API, but should still be considered almost essential for writing scripts.

- [utils.lua](#) - A large collection of various utility functions, especially for handling colours.
- [karaskel.lua](#) - The karaoke skeleton is a collection of functions mainly intended to do text layout of timed karaoke for creating advanced karaoke effects, as well as a number of other helper functions.
- [unicode.lua](#) - All data passed in and out of Aegisub through the Automation 4 Lua interface are encoded in UTF-8, but Lua doesn't natively provide support for this. A number of helper functions are provided here.
- [cleantags.lua](#) - A function to clean up ASS tags in a line.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Category: [Lua Reference](#)

Options

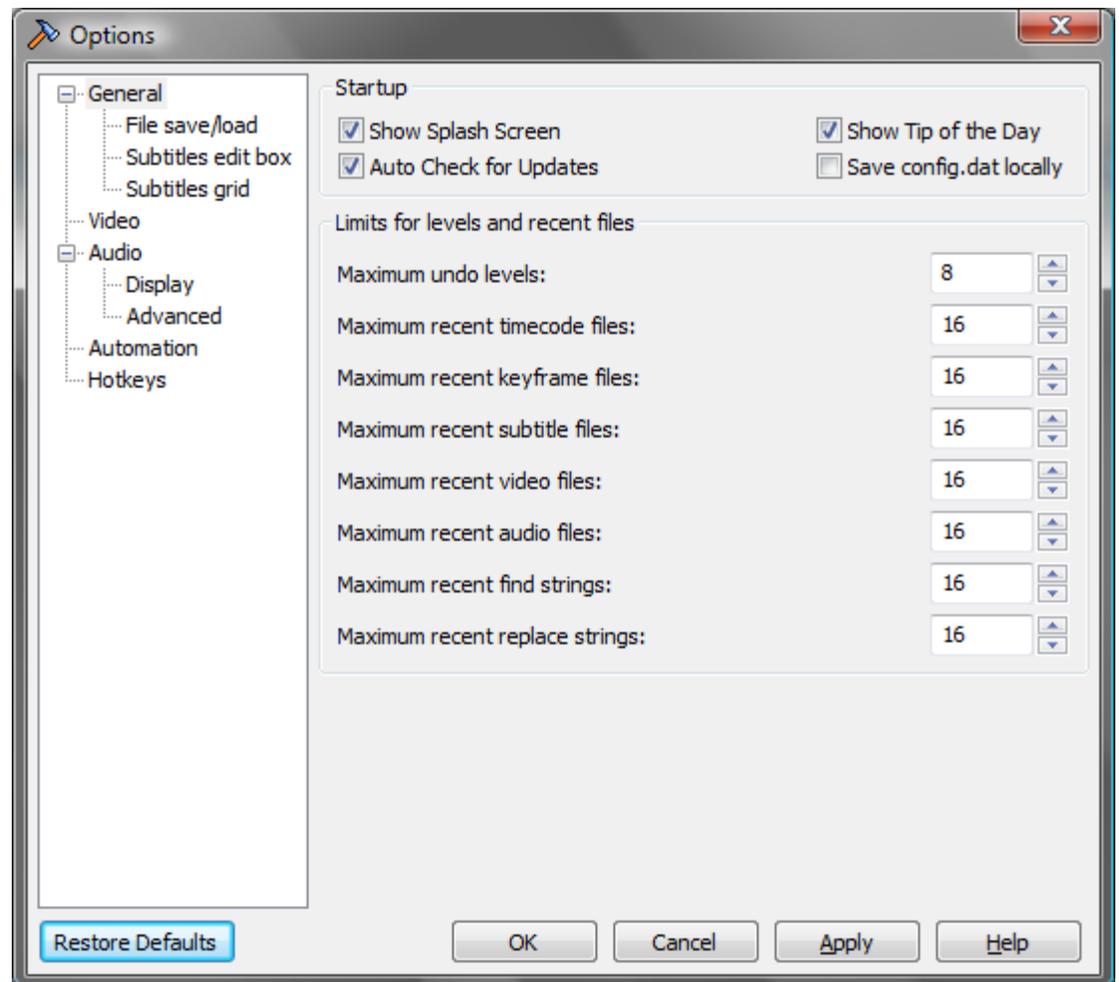
Aegisub is highly customizable and hence has a large amount of user-changeable options. These are available in the options dialog box, found in the View menu. This page is a reference of all the available options.

Aegisub stores all its configuration in a plain text file called *config.dat* which by default is stored in the `?user` directory. If you want to reset Aegisub to its default options without reinstalling the program, you can just delete *config.dat* and restart Aegisub.

General

Contents [hide]

- 1 General
 - 1.1 General -> File save/load
 - 1.2 General -> Subtitles edit box
 - 1.3 General -> Subtitles grid
- 2 Video
- 3 Audio
 - 3.1 Audio -> Display
 - 3.2 Audio -> Advanced
- 4 Automation
- 5 Hotkeys



- Show Splash Screen

If enabled, Aegisub will display its splash screen when starting up.

Navigation

- » Main Page

Introduction

- » What is Aegisub?
- » Highlights
- » Credits
- » Support Aegisub
- » FAQ
- » Tutorials

Working with Subtitles

- » Editing Subtitles
- » Exporting Subtitles
- » Applying Subtitles
- » Spell Checker
- » Translation Assistant
- » Paste Over
- » Select Lines

Typesetting

- » Introduction
- » Working with Video
- » Editing styles
- » Visual Typesetting
- » ASS Override Tags
- » Colour Picker
- » Styling Assistant
- » Resolution Resampler
- » Fonts Collector

Timing

- » Working with Audio
- » Shift times
- » Timing Post-Processor
- » Kanji Timer

Automation

- » Overview
- » Karaoke Templater
- » Lua Reference

Miscellaneous

- » Aegisub Options
- » Script Properties
- » Attachment Manager

- **Show Tip of the Day**

If enabled, Aegisub will display a (potentially) useful tip of the day after starting up.

- **Auto Check for Updates**

If enabled, Aegisub will periodically check whether there is a newer version available, and alert you if there is. Requires a working connection to the internet.

- **Save config.dat locally**

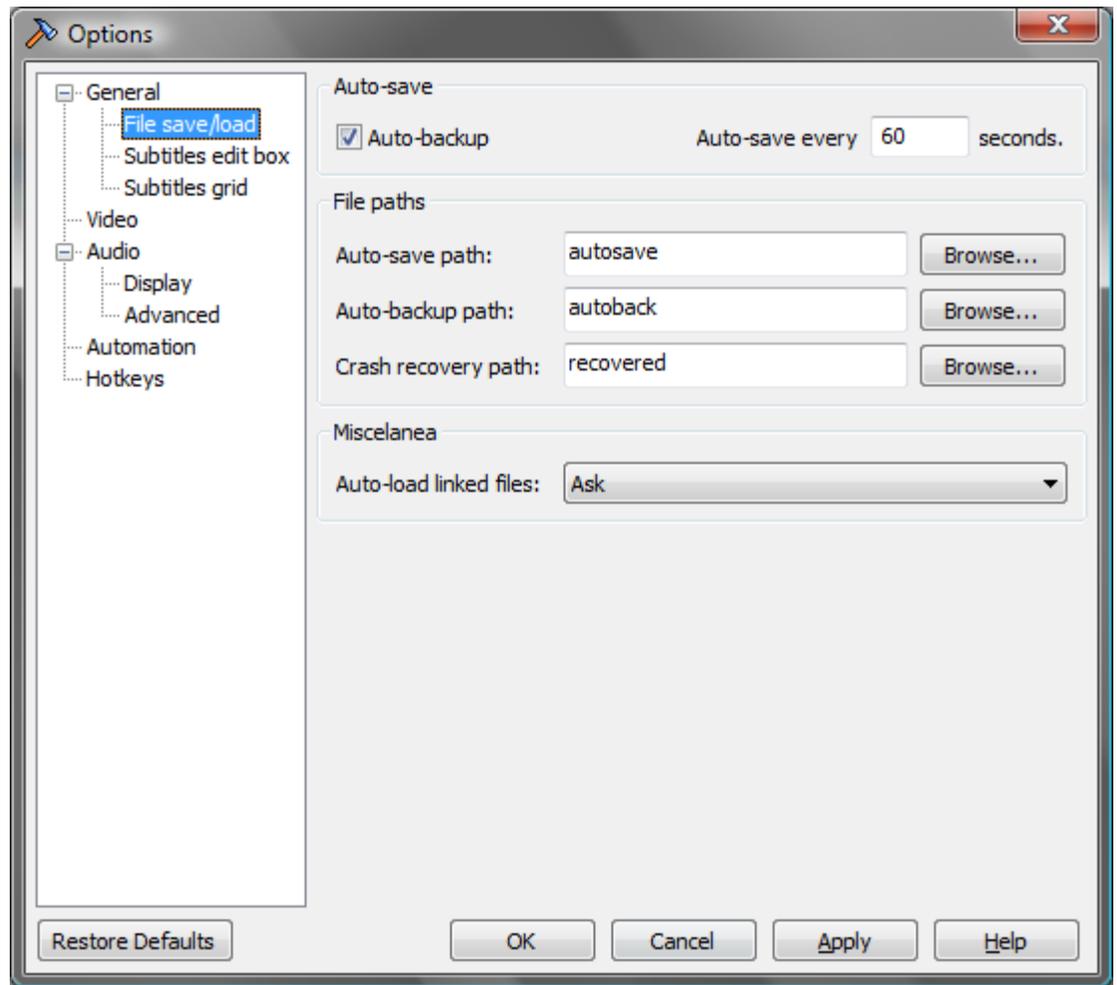
Normally, all configuration information for Aegisub is stored in a per-user basis in each system user's home directory. On Windows, this is `%APPDATA%\Aegisub\`, on Mac OS X it means `$HOME/Library/Application Support/Aegisub/` and on other POSIX-like systems (GNU/Linux, *BSD etc.) it's `$HOME/.aegisub/`. This is the default value of `?user`. However, by enabling this option, `?user` will instead refer to the same directory as `?data` (which is *not* a user-specific location), effectively making all configuration global to the installation of Aegisub.

Note: This option cannot be safely used if Aegisub is installed in a "usual" program directory on some operating systems unless you are the superuser, because it requires write permission to that directory or a nearby one. For example, normal users won't have write permission to the Program Files directory on Windows Vista, and most Linux distributions have similar restrictions for `/usr/share`.

- **Limits for levels and recent files**

These boxes all control how many files Aegisub will save in the "recently opened" submenus, and also how many undo levels will be saved. Change to your preference, but beware: the more undo levels you use, the more RAM Aegisub will consume. This is particularly noticeable with large scripts. Having lots of recently opened files stored has no real drawback except making config.dat bigger and the submenus harder to navigate, though.

General -> File save/load



- **Auto-backup**

If enabled, Aegisub will save a backup copy of each script you open, immediately on opening it. By default, it is saved to `?user/autoback/`, but this can be changed (see below).

- **Auto-save every X seconds**

Decides how often Aegisub should automatically save a copy of the script you are working on. Setting it to 0 disables autosaving. By default, the automatically saved copy is stored in `?user/autosave/` but this can be changed (see below).

- **Auto-save path**

Decides where to save autosaved copies of scripts you are working on. By default set to `autosave` in your Aegisub `?user` directory (see the [Aegisub path specifiers](#) page for details).

- **Auto-backup path**

Decides where to save automatic backup copies of scripts. By default set to `autoback` in your Aegisub `?user` directory.

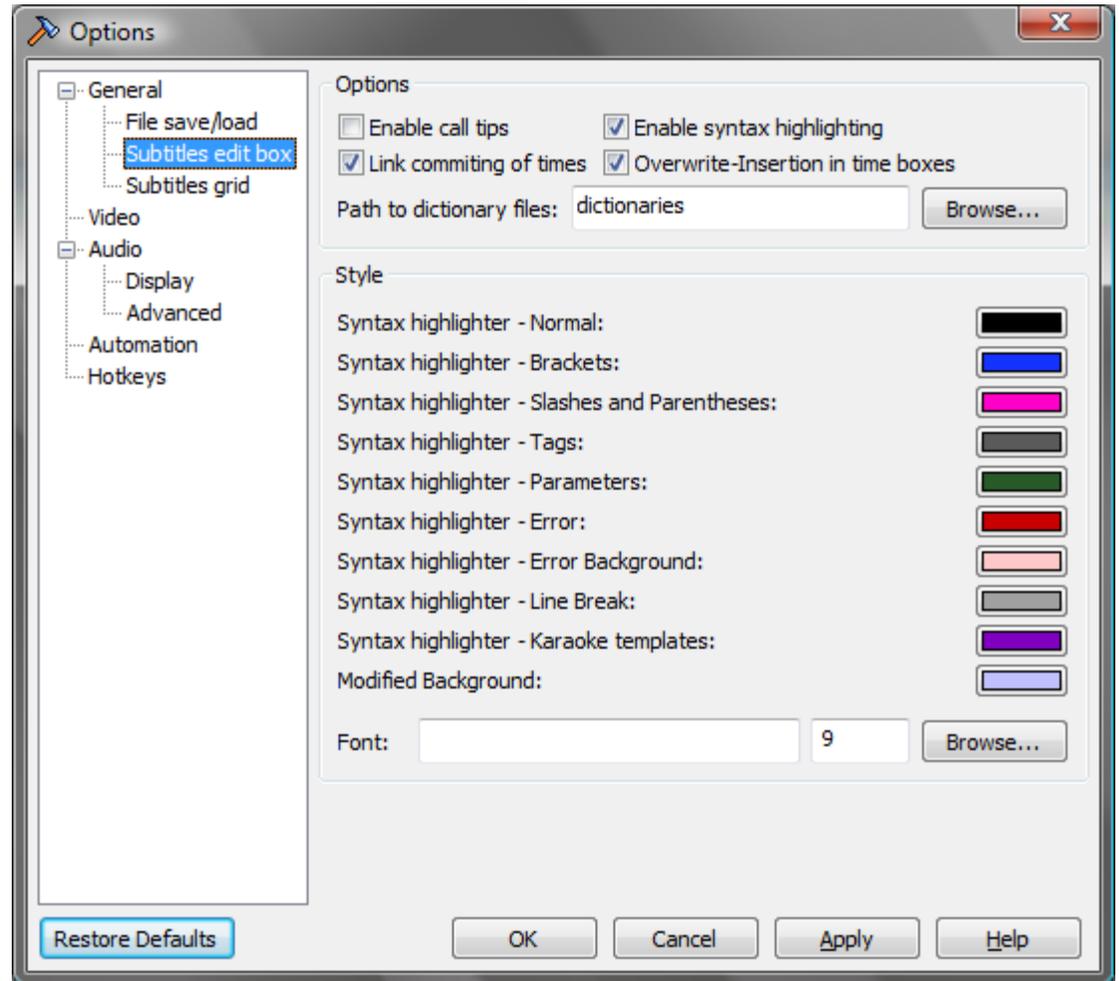
- **Crash recovery path**

Decides where Aegisub should save scripts recovered from crashes. By default set to `recovered` in your Aegisub `?user` directory.

- **Auto-load linked files**

Whenever you save a script, Aegisub also stores some information about what video, audio and timecodes files you had open while working on it, inside the script itself. This option decides what Aegisub does with these files that are "linked" to the script when opening it. If set to "ask", Aegisub will ask you if you want to load linked files. If set to "never", Aegisub will never load linked files, and likewise if set to "always", Aegisub will always try to load linked files (and report an error if the file(s) were not found).

General -> Subtitles edit box



- **Enable call tips**

When enabled, Aegisub will detect when you are writing an **override tag** and display a small box with a brief reference of the syntax of the tag in question until you close the tag. This is called a "call tip" and the feature may be familiar to users of various programming IDE's.

- **Link committing of times**

When enabled, Aegisub will commit both start and end times when you hit Enter with the mouse cursor in either time edit box. If disabled, they will be committed separately instead.

- **Enable syntax highlighting**

Enables or disables syntax highlighting of override tags.

- **Overwrite-Insertion in time boxes**

Controls the behavior of all time edit boxes in the program. By default, all time edit boxes in Aegisub behave like as if you had pressed the Insert button, so every digit you type overwrites what is already there, and you cannot erase numbers that are already there, you have to overwrite them. Unticking this box disables this behavior and makes the time edit boxes behave just like normal text edit boxes (almost).

- **Path to dictionary files**

Decides where Aegisub will look for dictionary files for its spellchecker and thesaurus. By default it looks in `?data/dictionaries`, but if you have your own dictionaries in the correct format somewhere else, feel free to point Aegisub there instead.

- **Syntax highlighter - Normal**
- **Syntax highlighter - Brackets**
- **Syntax highlighter - Slashes and parentheses**
- **Syntax highlighter - Tags**
- **Syntax highlighter - Parameters**
- **Syntax highlighter - Error**
- **Syntax highlighter - Error background**
- **Syntax highlighter - Line break**
- **Syntax highlighter - Karaoke templates**

All of these decide what colors the syntax highlighter should use for various parts of the text in the subtitles edit box. "Normal" is plain normal text.

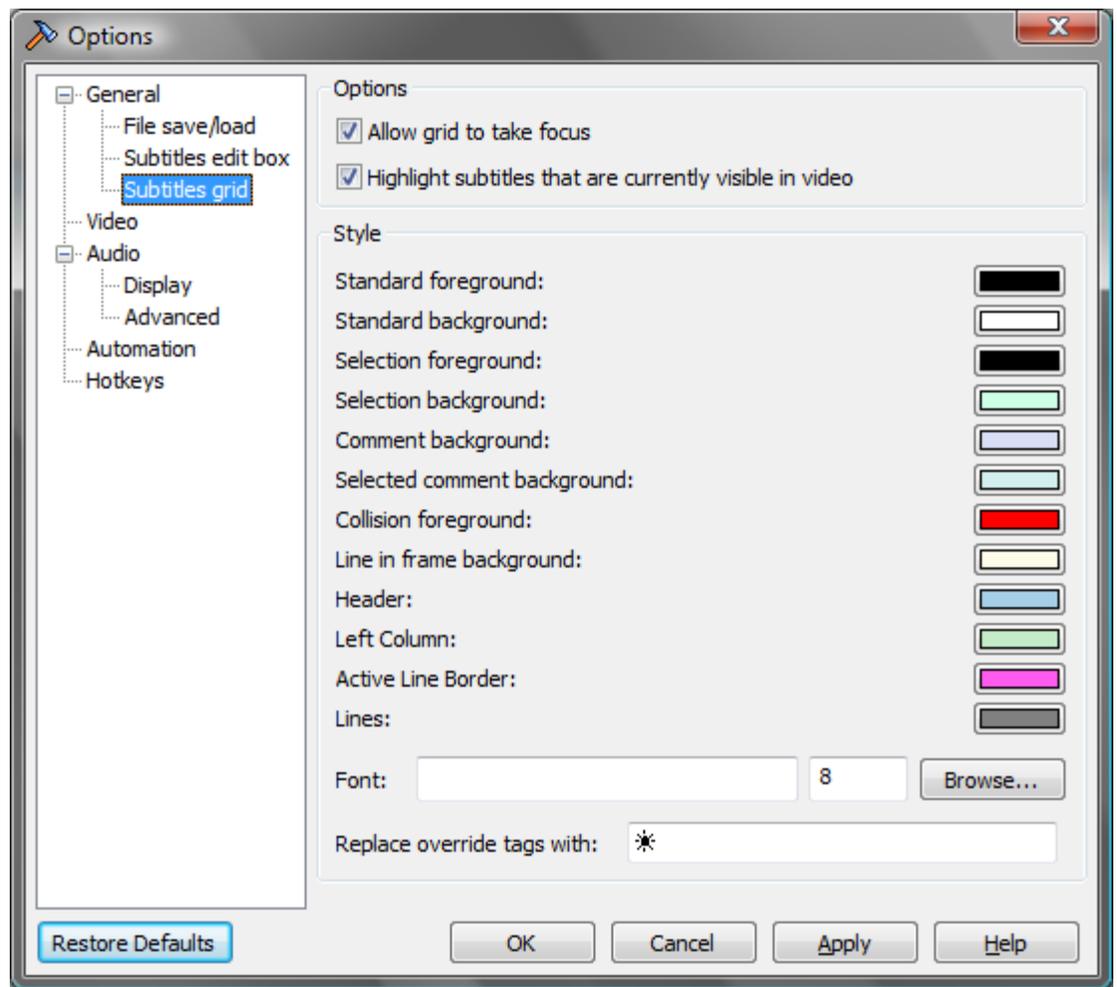
- **Modified background**

Decides what background color a changed but not yet committed box should have. Does not apply to the main subtitles edit box itself.

- **Font**

Decides the font and fontsize used for the subtitle edit box and other edit boxes.

General -> Subtitles grid



- **Allow grid to take focus**

When enabled, the subtitles grid acts as its own area of the program and it can have focus, just like the audio or the video can, and while it does you can use the arrow keys/mouse wheel to scroll around it etc. On the other hand, if you disable this option, the focus will stay where it was before whenever you click in the grid. This means you can't use keyboard shortcuts in the grid anymore, but on the other hand it means you can click in the grid to go to a line without losing the audio focus and so on. Use at your own discretion.

- **Highlight subtitles that are currently visible in the video**

When enabled, all subtitle lines that are currently visible in the video frame (or at least *should* be visible; Aegisub does not account for alpha and such trickery in this case, it only cares about the timing of the line) will be highlighted in the grid with a special background color (see the "Line in frame background" option below).

- **Standard foreground**
- **Standard background**

The normal color of lines in the grid. "Foreground" is the text color, and "Background" is obviously the background color.

- **Selection foreground**
- **Selection background**

The color of selected lines in the grid.

Comment background

- Selected comment background

The background color of commented-out lines and selected commented-out lines, respectively.

- Collision foreground

The text color of lines whose timings overlap.

- Line in frame background

The background color of lines currently visible in the video frame.

- Header
- Left column
- Active line border
- Lines

The color of the grid lines and fixed columns/headers.

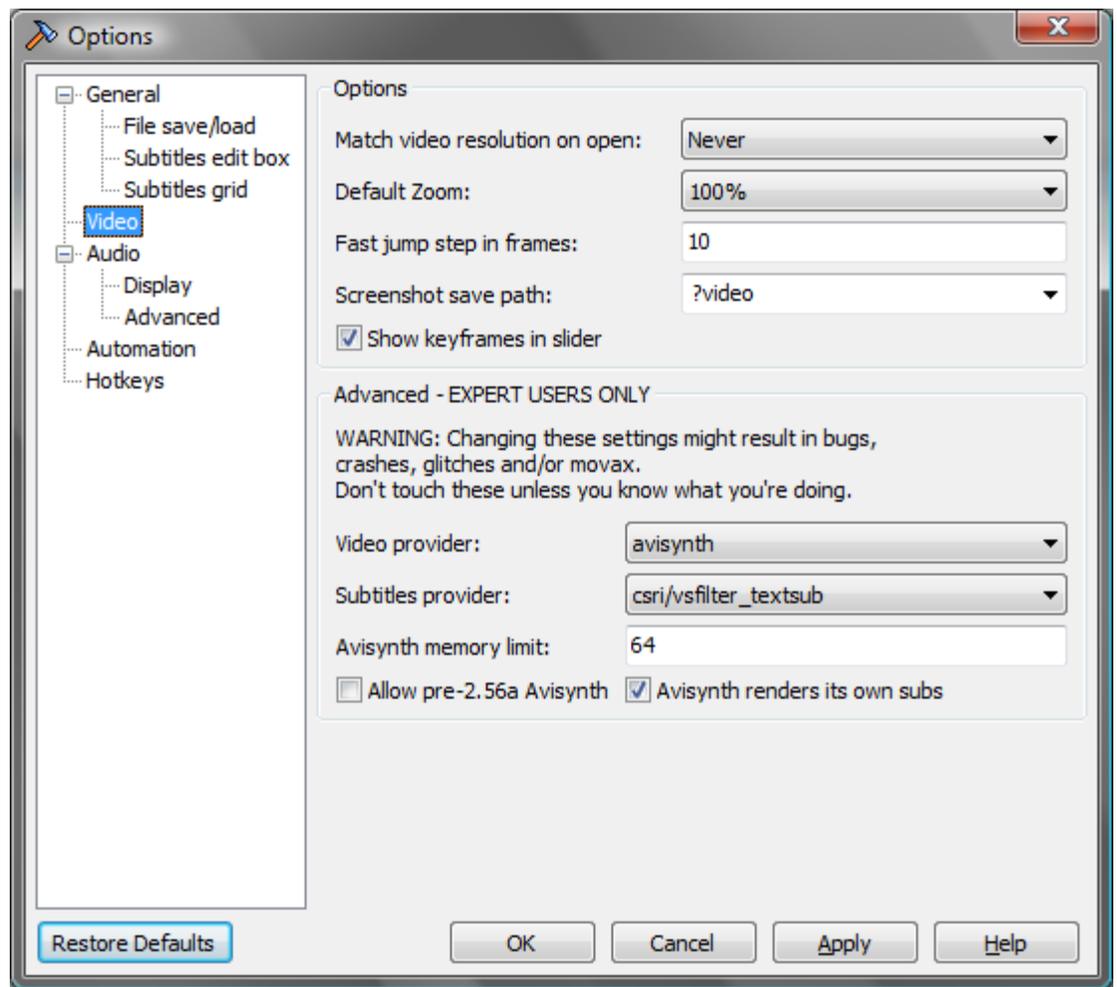
- Font

Decides the font and font size of all text in the grid.

- Replace override tags with:

The character that will be shown instead of override blocks if tag hiding is active.

Video



- **Match video resolution on open**

Controls what Aegisub will do about script resolution when you open a video. If set to "never", Aegisub will do nothing if the script resolution doesn't match the video resolution. If set to "ask", Aegisub will ask you if you want to change the script resolution to match the video resolution if they don't match. If set to "always", Aegisub will always change the script resolution to match the video resolution automatically. "always" is not recommended since it just changes script resolution without resampling anything, so it will most likely break existing typesetting.

- **Default zoom**

The default video zoom level. Useful if you have a very big or very small screen.

- **Fast jump step in frames**

Decides how big "jumps" Aegisub will make when you use the fast seek feature (Alt-rightarrow and Alt-leftarrow). Measured in frames.

- **Screenshot save path**

Decides where Aegisub should save screenshots. The default is `?video`, which means they are saved to wherever the video is, but you can change it to any path you like. [Aegisub path specifiers](#) are supported; another option directly available in the dropdown is `?script`, which is wherever the script is.

- **Show keyframes in slider**

When enabled, Aegisub will draw keyframe markers on the video seek slider. Note that this is not supported for all video formats and/or video providers.

- **Video provider**

Decides what method Aegisub should use to load video. What options you have available here depends on how your copy of Aegisub was compiled and what operating system you are running under. The following alternatives exist:

- *avisynth* (Windows only)

Uses [Avisynth](#) to load video. Versatile, supports loading almost all common formats as well as .d2v files (indexed DVD VOB's) if the correct plugin is supplied. Note that Aegisub can install its own avisynth.dll instead of using your system installation if so desired. Requires Video for Windows decoders for AVI files for best performance. Uses ffmpegsource() to load anything that isn't AVI (and tries that too for AVI's if no suitable VfW decoder is found).

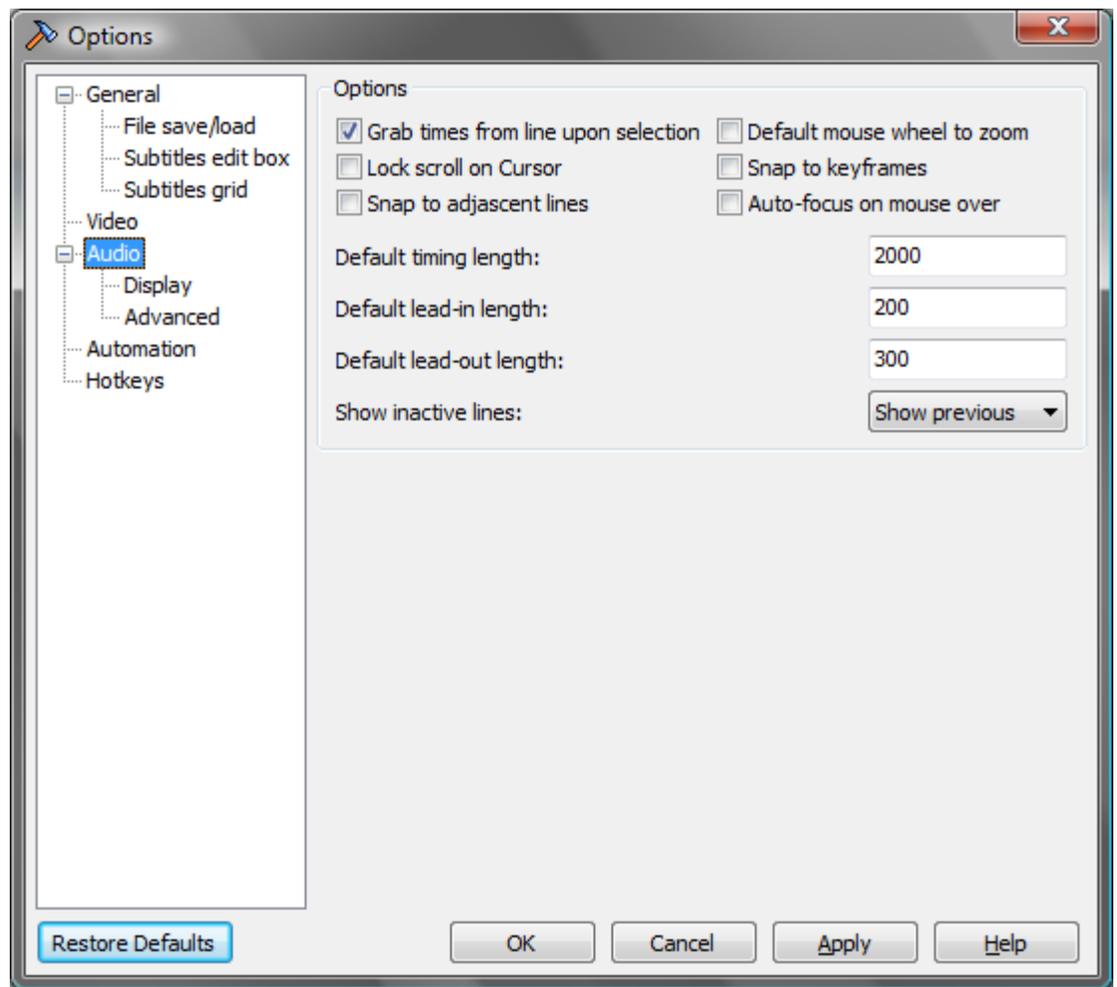
- *dshow* (Windows only)

Uses Microsoft's DirectShow framework to load video. Supports loading anything you can play in, say, Windows Media Player but is slower than Avisynth. May work with really odd formats that are not supported by ffmpegsource() but you do have a DirectShow decoder for.

- *ffmpeg*

Uses ffmpeg directly to load video. Supports most common formats. Technically there's nothing that stops you from using this on Windows, but most Windows builds (including the official Aegisub release builds for Windows) don't have it compiled in at all (since Avisynth uses ffmpegsource() anyway).

Audio



- **Grab times from line upon selection**

This option should really be named "Selecting lines changes audio selection", because that's what it does. When enabled, clicking a line in the grid or going to it using the next/previous line hotkeys will select the corresponding area in the audio waveform. When disabled, the selected area will always stay the same until you change it.

- **Lock scroll on cursor**

When enabled, the audio waveform view will automatically scroll to follow the playback cursor if it gets too close to either of the edges.

- **Snap to adjacent lines**

When enabled, Aegisub will snap the audio cursor to adjacent line start or end times if it is close enough.

- **Default mouse wheel to zoom**

When enabled, the mouse wheel will zoom the audio display horizontally. If disabled, it will scroll the audio display instead.

- **Snap to keyframes**

When enabled, the audio cursor will snap to keyframe lines (if you have any loaded, either from a video or from a keyframe file) if it is close enough.

- **Auto-focus on mouse over**

If enabled, moving the mouse cursor over the audio waveform will automatically

give it focus (as opposed to requiring a click).

- **Default timing length**

The default length of a new untimed line, in milliseconds.

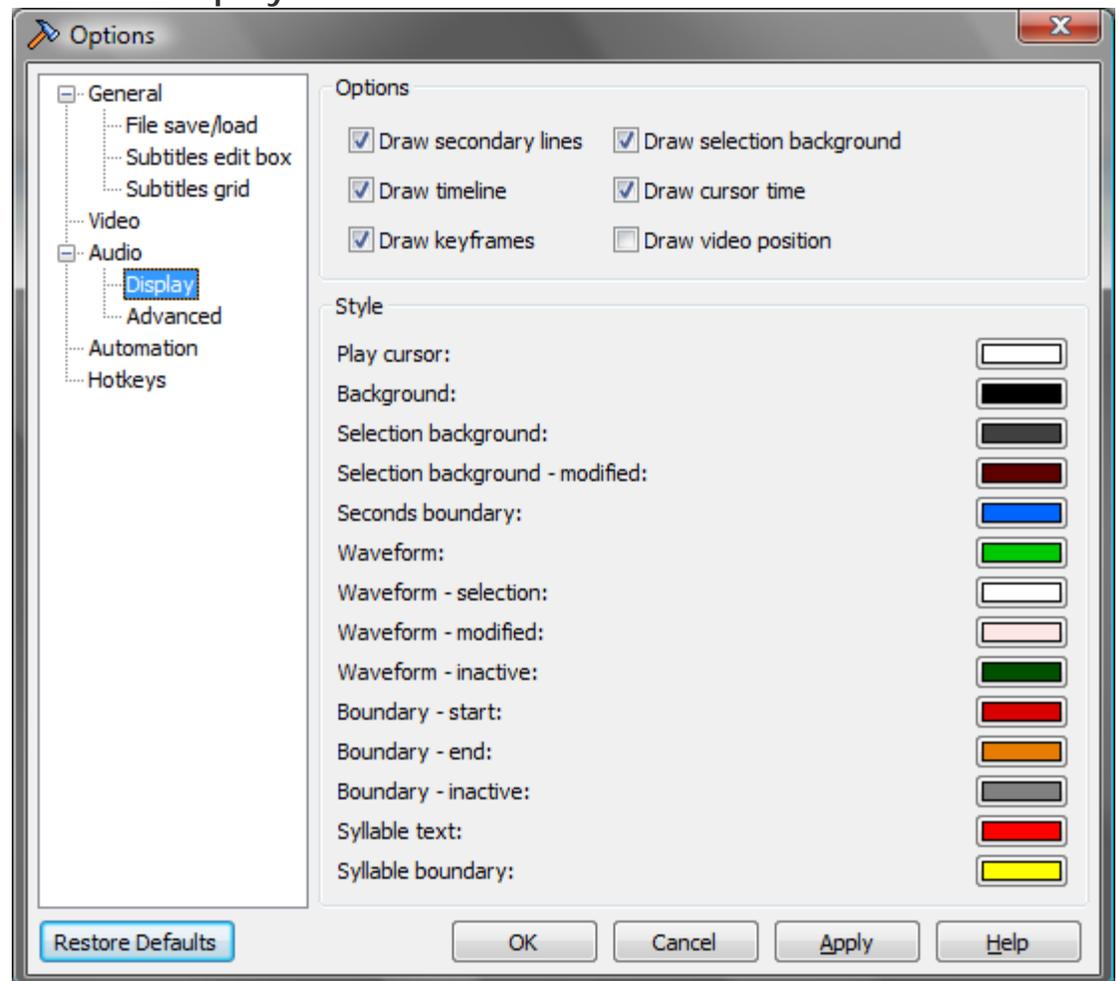
- **Default lead-in length**
- **Default lead-out length**

The duration added to the start of a line by the "add lead-in" and "add lead-out" functions. Also used in the [timing postprocessor](#).

- **Show inactive lines**

Controls how lines surrounding the currently selected line are displayed in the audio waveform. "Don't show" will only display the current line, "Show previous" will show the previous line (in the grid, *not* chronologically) in gray in addition to the current line, and "Show all" will show all lines in gray in addition to the current line.

Audio -> Display



- **Draw secondary lines**

If enabled, lines marking seconds will be drawn in the audio waveform.

- **Draw timeline**

If enabled, a timeline "ruler" with markers for each quarter second will be displayed below the audio waveform.

- **Draw keyframes**

If enabled, lines marking the positions of keyframes in the video will be drawn on the audio waveform.

- **Draw selection background**

If enabled, the selection will be drawn in an alternate color; if disabled only the line start and end markers will show where the selection is.

- **Draw cursor time**

If enabled, a timestamp showing the time since the start of the file will be drawn near the top of the audio waveform cursor.

- **Play cursor**

The color of the playback cursor.

- **Background**

The color of the audio waveform background.

- **Selection background**

The color of the audio waveform background when it is covered by the selection.

- **Selection background - modified**

The color of the audio waveform background when it is covered by the selection and the selection is detected as modified.

- **Seconds boundary**

The color of the second marker lines.

- **Waveform**
- **Waveform - selection**
- **Waveform - modified**
- **Waveform - inactive**

The color of the waveform itself in various situations (selected, detected as modified, and covered by an inactive line selection respectively).

- **Boundary - start**
- **Boundary - end**
- **Boundary - inactive**

The respective colors of the various line boundary markers.

- **Syllable text**

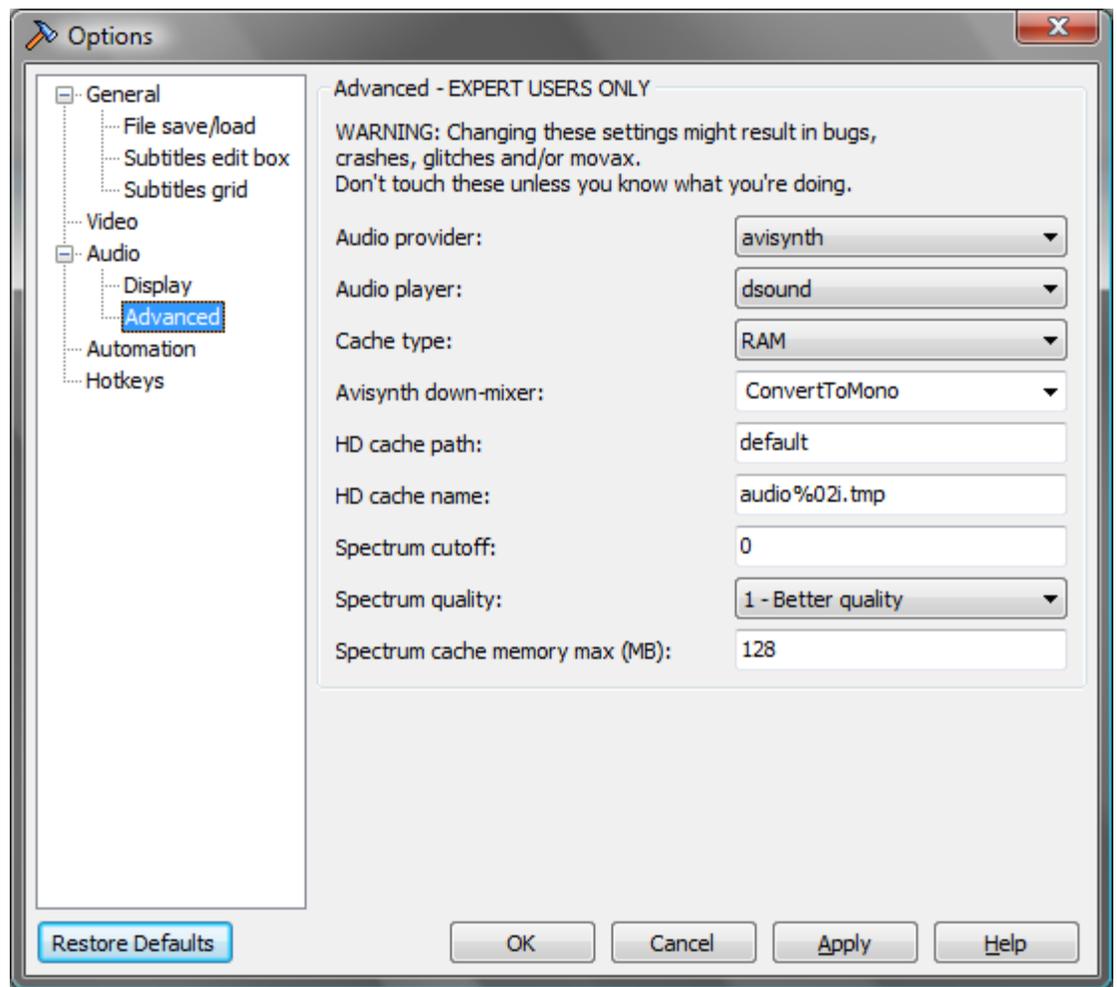
The color of a syllable's text in karaoke mode.

- **Syllable boundary**

The color of a syllable boundary line in karaoke mode.

It is not possible to change the color scheme of the spectrum display mode.

Audio -> Advanced



- **Audio provider**

What method to use for loading audio. Currently there are only two methods, which one is used depends on the platform.

- *avisynth* (Windows only)

Uses [Avisynth](#) to load audio. All file types will be loaded with `DirectShowSource()`, except for AVS files which will be opened with `Import()`.

- *lavc*

Uses `ffmpeg` directly to load audio. Supports most common formats. Technically there's nothing that stops you from using this on Windows, but most Windows builds (including the official Aegisub release builds for Windows) don't have it compiled in at all.

Regardless of this setting, the internal PCM WAV reader will always be tried first for WAV files.

- **Audio player**

What method to use for playing back audio. The options depend on the platform.

- *dsound* (Windows only)

Uses Microsoft DirectSound to play back audio. This is the best tested and most stable audio player.

- *alsa* (Linux only)

Uses the [Advanced Linux Sound Architecture](#) to play back audio. ALSA is the native sound architecture of Linux and is not available on any other systems. *It is possible to select the output device to use, but this is not exposed in the*

Options dialogue yet.

- *pulse* (Linux and other *NIX-like systems)

Plays sound back through a [PulseAudio](#) sound server.

- *portaudio*

Use the [PortAudio](#) API to play back sound. PortAudio has different playback implementations on different platforms. On most Unixes it uses Open Sound System (OSS) for output. PortAudio currently doesn't seem to work on Mac, and it is not included in Windows builds as the DirectSound player is more reliable.

- *openal* (Windows and Mac)

Uses the [OpenAL](#) API to play back audio. It is the only working audio player on Mac. It is not included in Windows builds due to the extra dependency it creates, and DirectSound is more reliable.

- **CacheType**

Use RAM unless you have very little of it, then use Hard Disk. The cache is not needed and not used when PCM WAV files are opened. If you disable caching, audio playback might become very unreliable.

- **Avisynth down-mixer** (Windows only)

Aegisub can only use mono (single-channel) audio. This option determines which Avisynth function to use to convert audio to mono. This option is only available in the Windows version of Aegisub.

- **HD cache path**

- **HD cache name**

These options determine where the hard disk audio cache will be located. Only used if cache is set to hard disk. You shouldn't need to change this unless you're low on disk space. For the name, the string expects a printf-style "%i" parameter, that will be replaced with a number. "%02i" is used by default, don't change that unless you know what you are doing.

- **Spectrum cutoff**

The index of the smallest frequency band the audio display will show in spectrum mode. There's usually no need to change this.

- **Spectrum quality**

Determines the quality of the audio spectrum display. Higher quality settings result in larger CPU and RAM use. Each consecutive setting uses a bit more CPU than the previous, and double the amount of RAM. For 48 kHz samplerate audio, one minute of audio uses this much memory at the different settings:

- 0 "regular" - 11 MB
- 1 "better" - 22 MB
- 2 "high" - 44 MB
- 3 "insane" - 88 MB

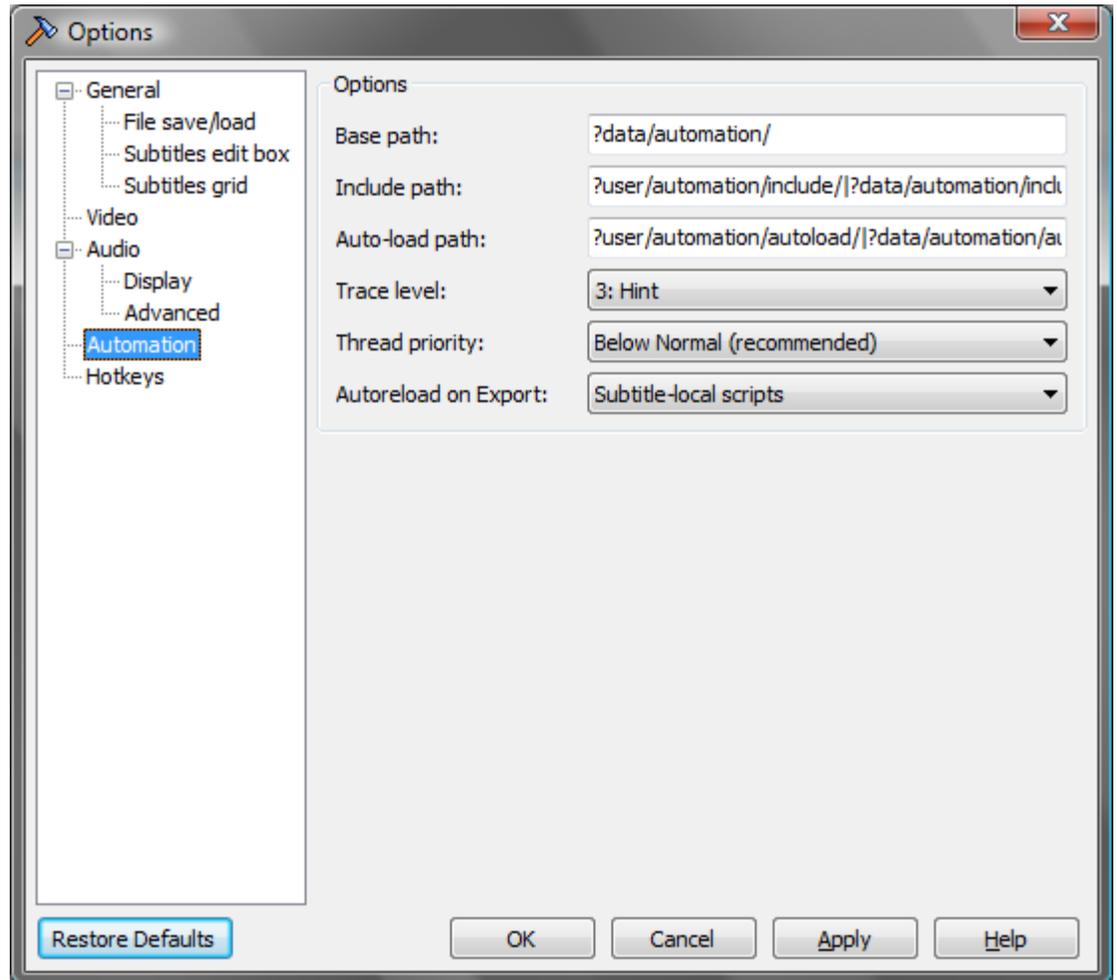
The amount of memory used does not depend on the number of channels (Aegisub always works in mono) or the bit-depth (the spectrum is always calculated in 32 bit floating point) of the audio.

- **Spectrum cache memory max**

The maximum amount of memory to use for audio spectrum caching. The results of

the calculations to display the audio spectrum are cached to make scrolling through the audio smoother. The amount of spectrum display that can be cached in an amount of memory depends on the quality setting above. The default cache size of 128 MB results a little less than 6 minutes of audio at 48 kHz in quality 1. If you set this smaller than 5 MB, the default of 128 MB will be used instead. You probably shouldn't set this to more than 1/4 of your amount of physical RAM installed.

Automation



- **Base path**

Not used by the Lua scripting engine. In the future this might be used by other scripting engines.

- **Include path**

List of directories where include files are searched for. Directories are separated with a pipe character, |.

- **Auto-load path**

List of directories that are searched for scripts on startup, which are then automatically loaded. Directories are separated with a pipe character, |.

- **Trace level**

When a script sends a message to the debug console it can also specify a trace level.

If the trace level of a message is lower than the value given here, the message is not logged. The names given to the levels are only suggestions, they don't have any effect on the execution of the script. (Eg. a "Fatal" level message will not cause the script to terminate.)

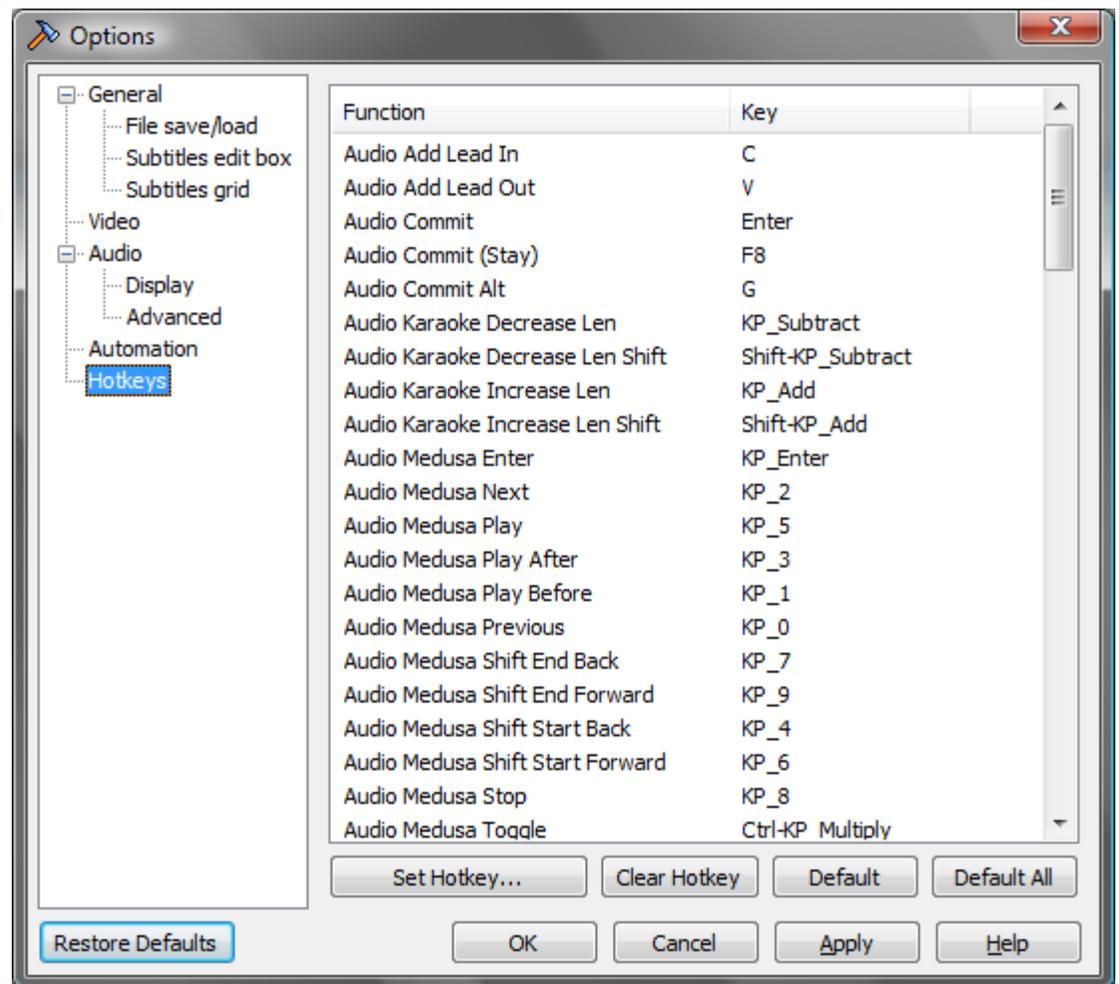
- **Thread priority**

Priority given to the script execution thread. If you're on a single-core/single-CPU system having this on lower than normal will make other programs more responsive while long-running scripts are active.

- **Autoreload on Export**

Automatically reloads the specified sets of scripts when the [Exporting](#) dialogue is opened. Note that if a script fails to reload, no errors are displayed, it just won't show up in the Export window then. In that case you will have to enter the [Automation Manager](#) window and determine the cause of the error.

Hotkeys

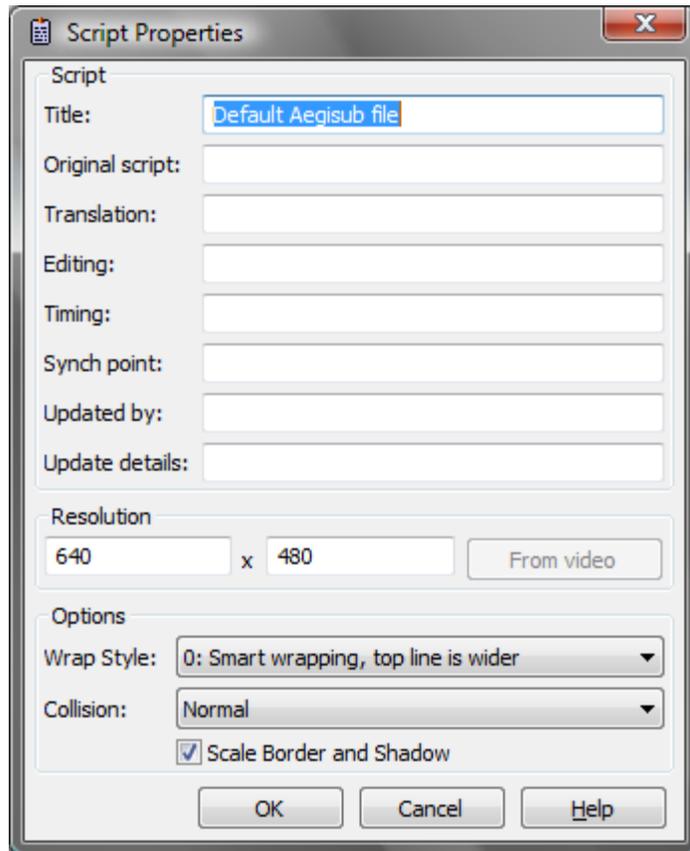


This page shows a list of all functions in Aegisub that can be bound to keyboard shortcuts, and what shortcut (if any) they are currently assigned to. Should be self explanatory.

Double click on a hotkey in the list to change it.

Properties

The **script properties** are some headers and other options that affect the entire script in various ways. They can be accessed from the *File menu -> Properties*.



The properties are:

- **Title**, **Original script**, **Translation**, **Editing**, **Timing**, **Synch point**, **Updated by** and **Update details** - These are for informational purposes only and do not affect rendering in any way.
- **Resolution** - The virtual resolution of the script. Unlike the informational headers, this **does** effect rendering in several ways. It's recommended to set this to the same as the video resolution, or an even multiple thereof. It affects:
 - All coordinates (for margins, `\pos`, `\move`, vector drawing etc) are given in script resolution pixels.
 - All font sizes are scaled according to the script resolution.
 - If *scale border and shadow* (see below) is enabled, all border and shadow values are given in script resolution pixels (if it is disabled they are in video resolution pixels instead).
 - Do however note that in a rather confusing manner, the script resolution does **not** affect the aspect ratio of any text, although it does affect the aspect ratio of vector drawings.
- **Wrap style** - Controls how the subtitle renderer will break lines that are too long to fit on one line. The modes are:
 - 0 - The default mode. "Smart" wrapping; if a line is too long to fit on a line by itself, breaks it into two roughly evenly long lines, but prefers the top line to be wider. `\N` (note capital N) can be used to insert a manual linebreak.
 - 1 - Inserts a linebreak when the line reaches the edges of the frame (minus margins); i.e. if it's just one word too long to fit on a line by itself, you get the last word all by itself on the bottom line. Almost never useful. As

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

with 0, `\N` can be used for manual linebreaks.

- 2 - No automatic linewrapping at all; if a line is too long to fit inside the video frame, it will just continue outside the frame. `\n` and `\N` can be used to insert manual linebreaks.
- 3 - Same as mode 0, but prefers the bottom line to be wider.
- **Collision** - Determines how overlapping lines should be handled. In "normal" mode, the first line will stay where it is and the second one drawn above it. In "reverse" mode, the first line will be moved upwards and the second drawn under it.
- **Scale border and shadow** - Controls how outline and shadow widths are drawn. If disabled, they are given in video resolution pixels (meaning that using the script with a higher resolution video will make all borders/shadows look thinner). If enabled, they are given in script resolution pixels instead (meaning that outlines and shadows will look the same no matter the resolution). The latter is usually what you want.

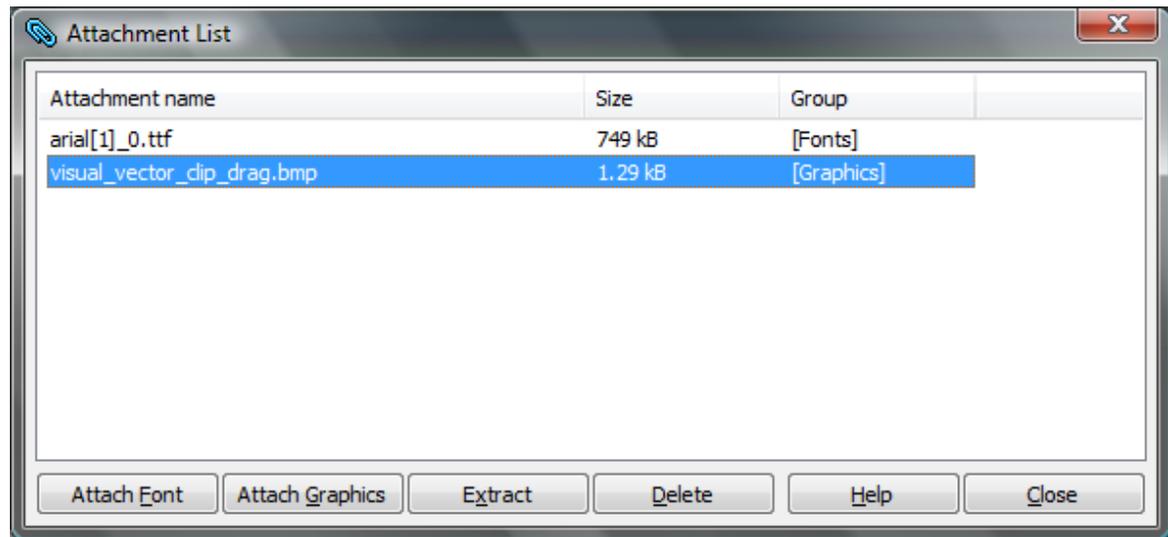
Attachment Manager

The attachment manager allows you to attach fonts and/or pictures to your script (by encoding them as text). This is especially useful for sharing fonts between everyone who is working on a script without having to send the fonts as separate files, for example.

Overview

Contents [hide]

- 1 Overview
- 2 Limitations and downsides
 - 2.1 Supported formats
 - 2.2 Compatibility issues
 - 2.3 Size



The entire dialogue is fairly self-explanatory. The two "Attach ..." buttons add attachments, "Extract" extracts existing attachments into separate files, and "Delete" deletes attachments from the subtitles file.

Limitations and downsides

Supported formats

The SSA format specification only allows certain filetypes to be attached. For fonts, only .ttf is allowed. For pictures, .bmp, .gif, .ico, .jpg and .wmf are allowed (note the absence of .png).

Compatibility issues

Many SSA/ASS editors ignore or strip attachments. The original SubStation Alpha (v4.08) will work fine, but only for real SSA files. Sabbu will complain about unrecognized fields, and strip the attachments if you save the file. Most other editors either ignore the attachments or crash when encountering them.

A notable exception is mkvmerge, which will convert the attached files to Matroska attachments on muxing. If you demux the script again, the attachments will be stripped from the script, but they're still there as MKV attachments.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Size

Unfortunately, storing binary data as text (in this case, a variant of UUEncoding) is not very efficient. The attached files will take considerably more space as script attachments than they do as separate files.

File:Main-window-overview.png

Navigation

- » Main Page

Introduction

- » What is Aegisub?
- » Highlights
- » Credits
- » Support Aegisub
- » FAQ
- » Tutorials

Working with Subtitles

- » Editing Subtitles
- » Exporting Subtitles
- » Applying Subtitles
- » Spell Checker
- » Translation Assistant
- » Paste Over
- » Select Lines

Typesetting

- » Introduction
- » Working with Video
- » Editing styles
- » Visual Typesetting
- » ASS Override Tags
- » Colour Picker
- » Styling Assistant
- » Resolution Resampler
- » Fonts Collector

Timing

- » Working with Audio
- » Shift times
- » Timing Post-Processor
- » Kanji Timer

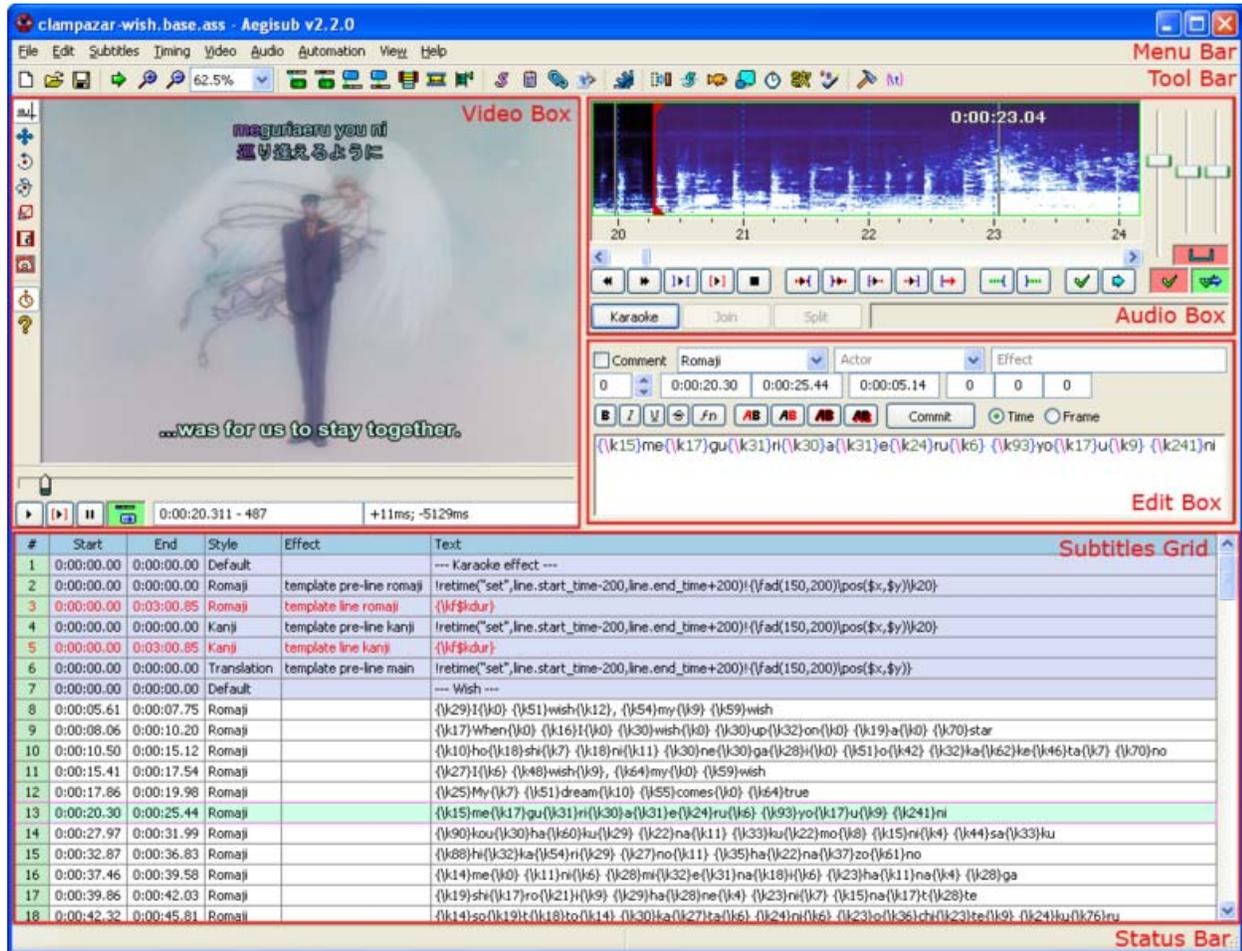
Automation

- » Overview
- » Karaoke Templater
- » Lua Reference

Miscellaneous

- » Aegisub Options
- » Script Properties
- » Attachment Manager

- File
- File history
- File links



Size of this preview: 779 × 600 pixels

Full resolution (1,018 × 784 pixels, file size: 314 KB, MIME type: image/png)

Overview of the Aegisub main window, with the main areas labelled.

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail

Dimensions User Comment

current contribs)

contribs) (Overview of the Aegisub main window, with the main areas labelled.)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Overview](#)

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

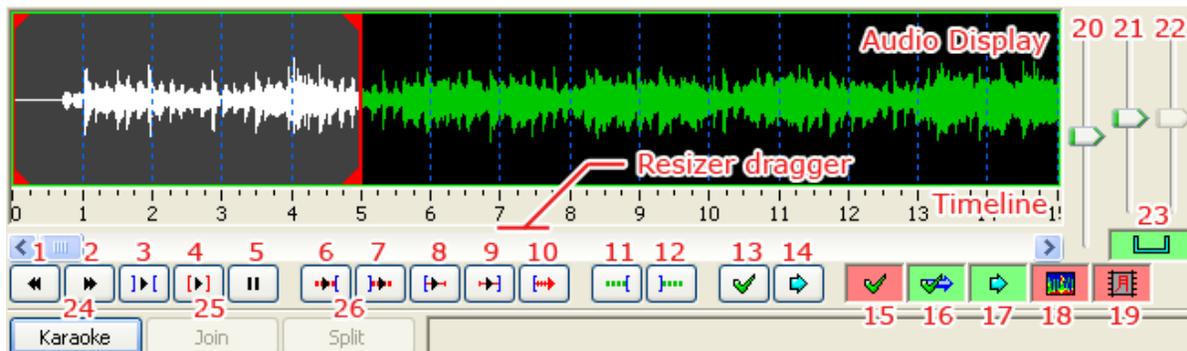
» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

File:Audio-box-waveform.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Audio-box-waveform.png](#) (654 × 193 pixels, file size: 26 KB, MIME type: image/png)

The audio box with waveform display.

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail

Dimensions User Comment

current contribs)

contribs) (The audio box with waveform display.)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Audio](#)

Options

(Redirected from Keyboard shortcuts)

Aegisub is highly customizable and hence has a large amount of user-changeable options. These are available in the options dialog box, found in the View menu. This page is a reference of all the available options.

Aegisub stores all its configuration in a plain text file called *config.dat* which by default is stored in the `?user` directory. If you want to reset Aegisub to its default options without reinstalling the program, you can just delete *config.dat* and restart Aegisub.

General

Contents [hide]

- 1 General
 - 1.1 General -> File save/load
 - 1.2 General -> Subtitles edit box
 - 1.3 General -> Subtitles grid
- 2 Video
- 3 Audio
 - 3.1 Audio -> Display
 - 3.2 Audio -> Advanced
- 4 Automation
- 5 Hotkeys

Navigation

- » Main Page

Introduction

- » What is Aegisub?
- » Highlights
- » Credits
- » Support Aegisub
- » FAQ
- » Tutorials

Working with Subtitles

- » Editing Subtitles
- » Exporting Subtitles
- » Applying Subtitles
- » Spell Checker
- » Translation Assistant
- » Paste Over
- » Select Lines

Typesetting

- » Introduction
- » Working with Video
- » Editing styles
- » Visual Typesetting
- » ASS Override Tags
- » Colour Picker
- » Styling Assistant
- » Resolution Resampler
- » Fonts Collector

Timing

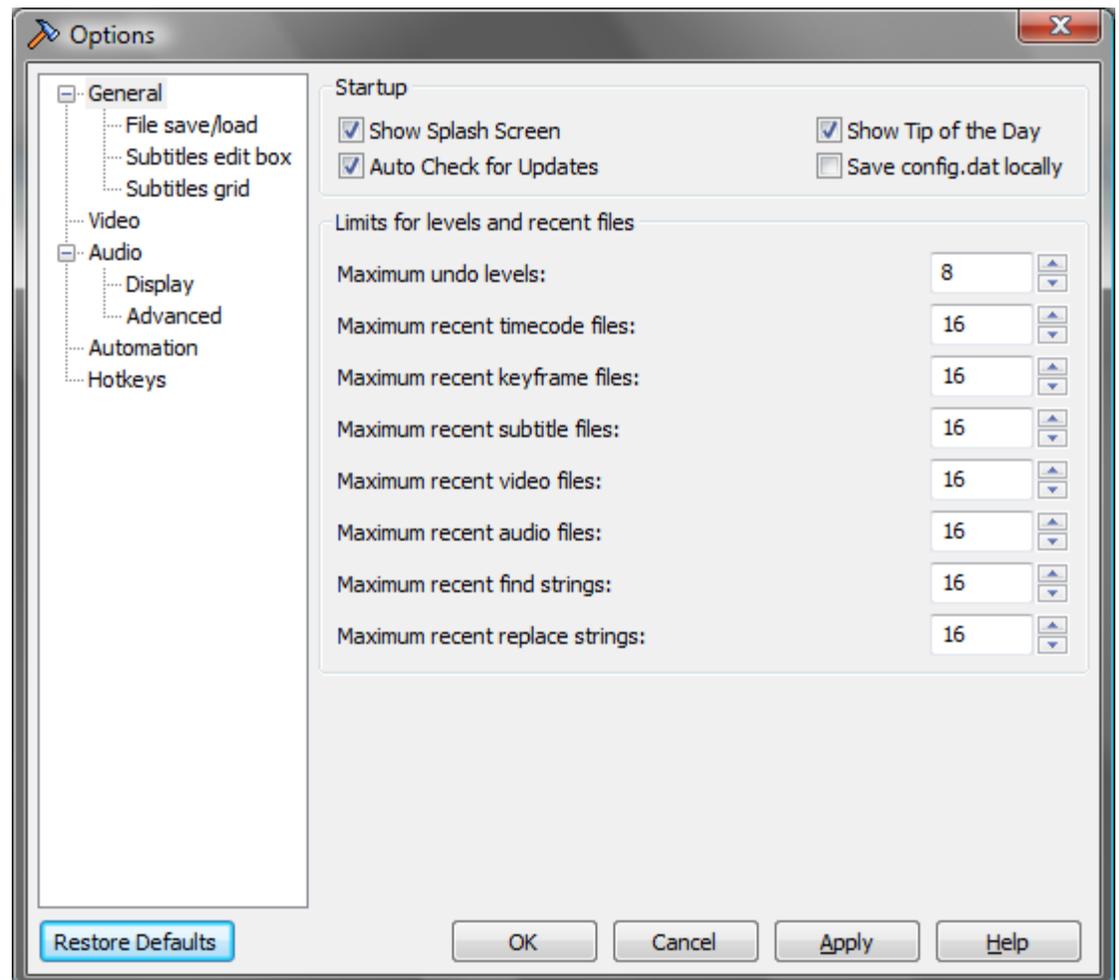
- » Working with Audio
- » Shift times
- » Timing Post-Processor
- » Kanji Timer

Automation

- » Overview
- » Karaoke Templater
- » Lua Reference

Miscellaneous

- » Aegisub Options
- » Script Properties
- » Attachment Manager



- Show Splash Screen

If enabled, Aegisub will display its splash screen when starting up.

- **Show Tip of the Day**

If enabled, Aegisub will display a (potentially) useful tip of the day after starting up.

- **Auto Check for Updates**

If enabled, Aegisub will periodically check whether there is a newer version available, and alert you if there is. Requires a working connection to the internet.

- **Save config.dat locally**

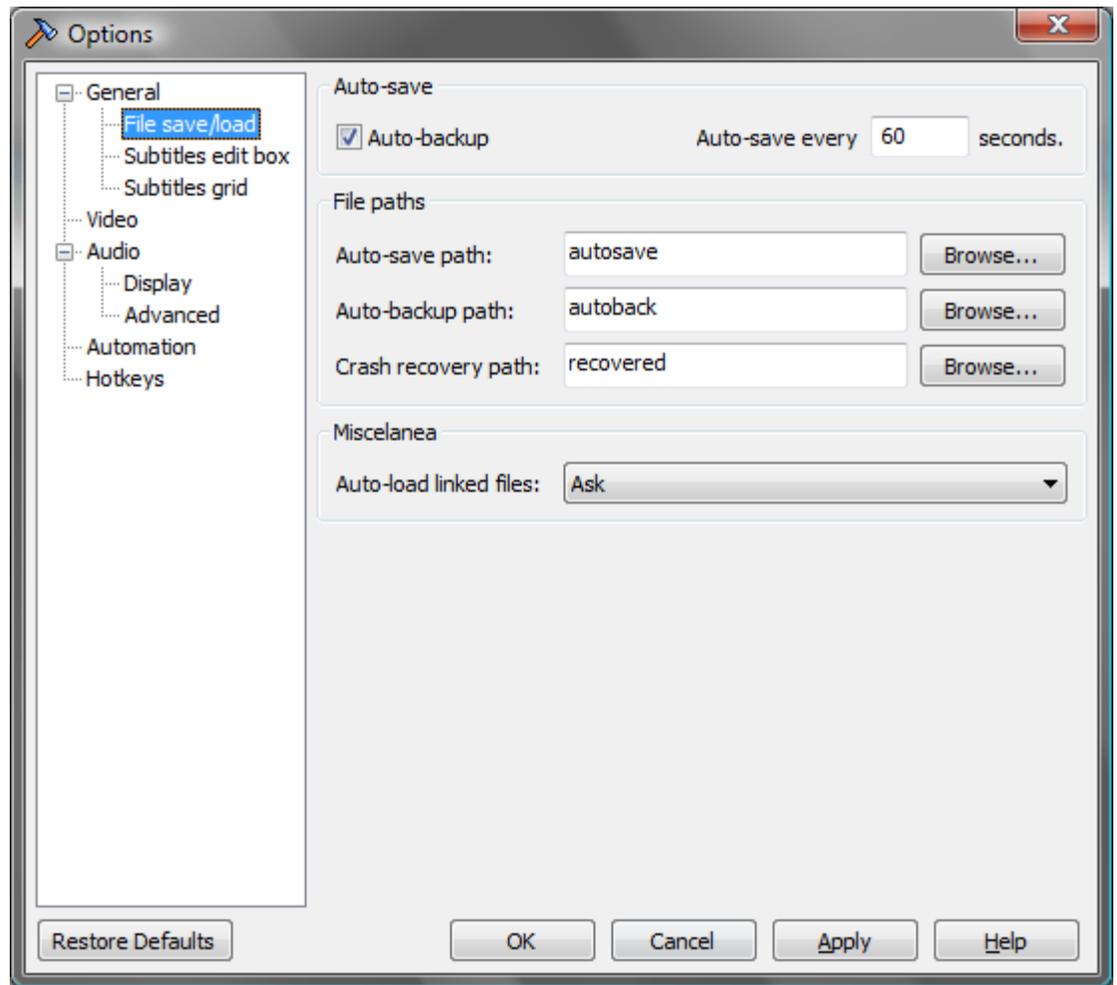
Normally, all configuration information for Aegisub is stored on a per-user basis in each system user's home directory. On Windows, this is `%APPDATA%\Aegisub\`, on Mac OS X it means `$HOME/Library/Application Support/Aegisub/` and on other POSIX-like systems (GNU/Linux, *BSD etc.) it's `$HOME/.aegisub/`. This is the default value of `?user`. However, by enabling this option, `?user` will instead refer to the same directory as `?data` (which is *not* a user-specific location), effectively making all configuration global to the installation of Aegisub.

Note: This option cannot be safely used if Aegisub is installed in a "usual" program directory on some operating systems unless you are the superuser, because it requires write permission to that directory or a nearby one. For example, normal users won't have write permission to the Program Files directory on Windows Vista, and most Linux distributions have similar restrictions for `/usr/share`.

- **Limits for levels and recent files**

These boxes all control how many files Aegisub will save in the "recently opened" submenus, and also how many undo levels will be saved. Change to your preference, but beware: the more undo levels you use, the more RAM Aegisub will consume. This is particularly noticeable with large scripts. Having lots of recently opened files stored has no real drawback except making `config.dat` bigger and the submenus harder to navigate, though.

General -> File save/load



- **Auto-backup**

If enabled, Aegisub will save a backup copy of each script you open, immediately on opening it. By default, it is saved to `?user/autoback/`, but this can be changed (see below).

- **Auto-save every X seconds**

Decides how often Aegisub should automatically save a copy of the script you are working on. Setting it to 0 disables autosaving. By default, the automatically saved copy is stored in `?user/autosave/` but this can be changed (see below).

- **Auto-save path**

Decides where to save autosaved copies of scripts you are working on. By default set to `autosave` in your Aegisub `?user` directory (see the [Aegisub path specifiers](#) page for details).

- **Auto-backup path**

Decides where to save automatic backup copies of scripts. By default set to `autoback` in your Aegisub `?user` directory.

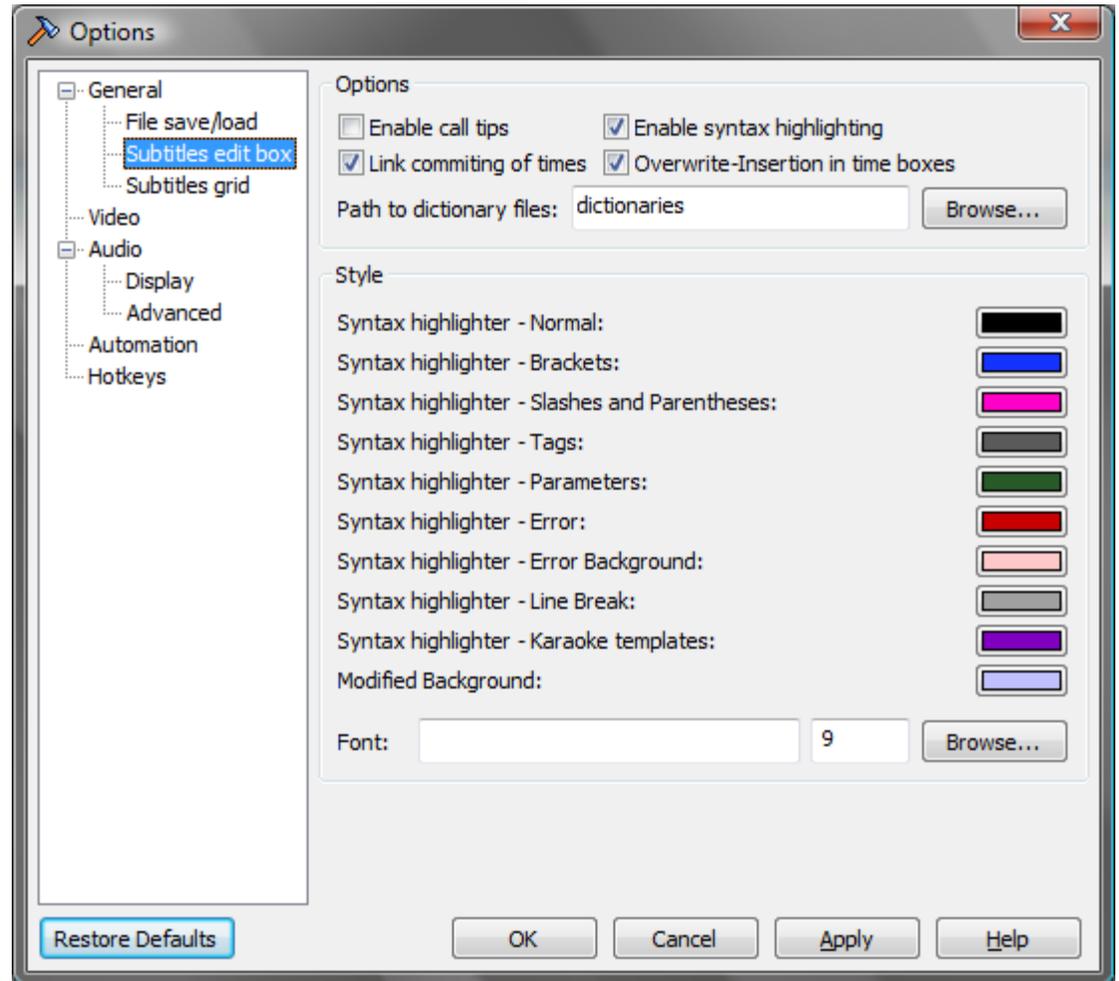
- **Crash recovery path**

Decides where Aegisub should save scripts recovered from crashes. By default set to `recovered` in your Aegisub `?user` diectory.

- **Auto-load linked files**

Whenever you save a script, Aegisub also stores some information about what video, audio and timecodes files you had open while working on it, inside the script itself. This option decides what Aegisub does with these files that are "linked" to the script when opening it. If set to "ask", Aegisub will ask you if you want to load linked files. If set to "never", Aegisub will never load linked files, and likewise if set to "always", Aegisub will always try to load linked files (and report an error if the file(s) were not found).

General -> Subtitles edit box



- **Enable call tips**

When enabled, Aegisub will detect when you are writing an **override tag** and display a small box with a brief reference of the syntax of the tag in question until you close the tag. This is called a "call tip" and the feature may be familiar to users of various programming IDE's.

- **Link committing of times**

When enabled, Aegisub will commit both start and end times when you hit Enter with the mouse cursor in either time edit box. If disabled, they will be committed separately instead.

- **Enable syntax highlighting**

Enables or disables syntax highlighting of override tags.

- **Overwrite-Insertion in time boxes**

Controls the behavior of all time edit boxes in the program. By default, all time edit boxes in Aegisub behave like as if you had pressed the Insert button, so every digit you type overwrites what is already there, and you cannot erase numbers that are already there, you have to overwrite them. Unticking this box disables this behavior and makes the time edit boxes behave just like normal text edit boxes (almost).

- **Path to dictionary files**

Decides where Aegisub will look for dictionary files for its spellchecker and thesaurus. By default it looks in `?data/dictionaries`, but if you have your own dictionaries in the correct format somewhere else, feel free to point Aegisub there instead.

- **Syntax highlighter - Normal**
- **Syntax highlighter - Brackets**
- **Syntax highlighter - Slashes and parentheses**
- **Syntax highlighter - Tags**
- **Syntax highlighter - Parameters**
- **Syntax highlighter - Error**
- **Syntax highlighter - Error background**
- **Syntax highlighter - Line break**
- **Syntax highlighter - Karaoke templates**

All of these decide what colors the syntax highlighter should use for various parts of the text in the subtitles edit box. "Normal" is plain normal text.

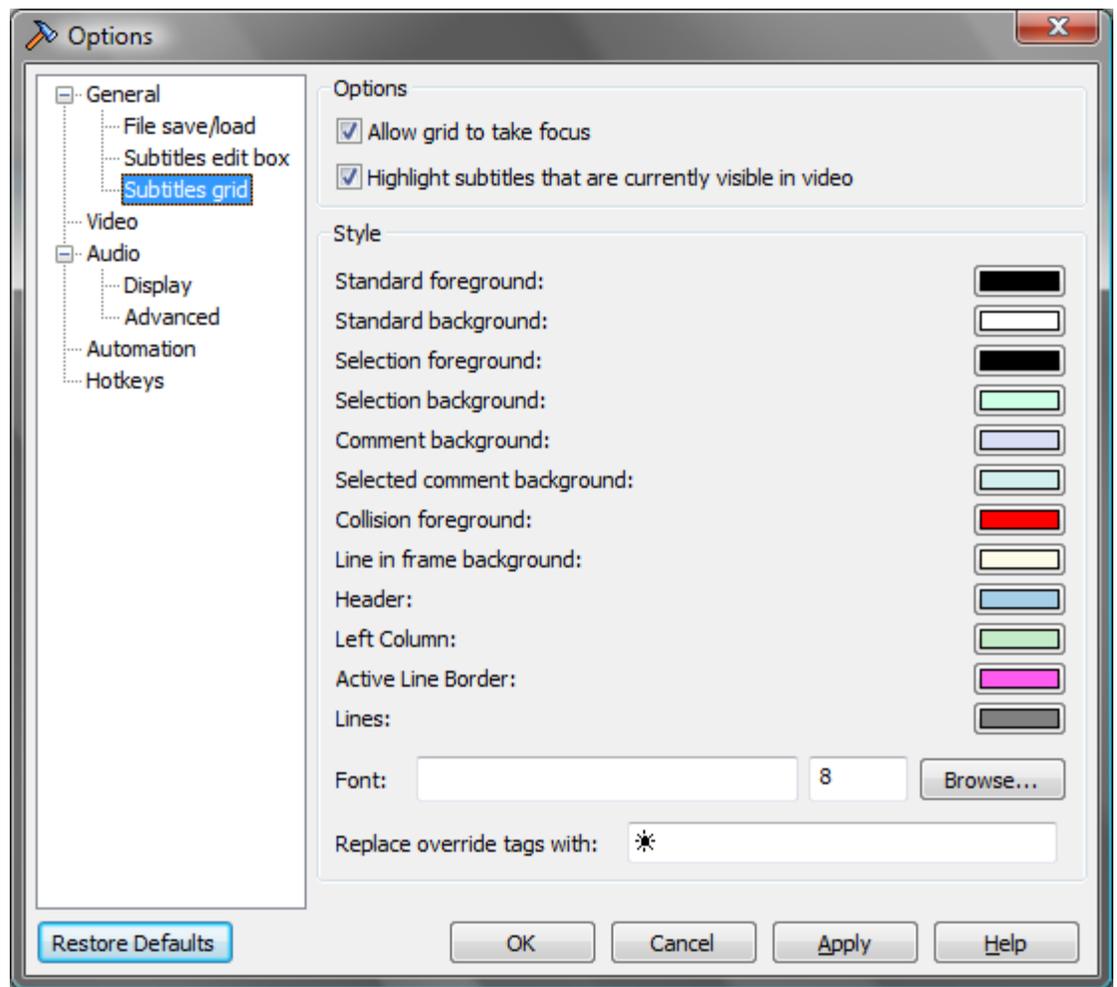
- **Modified background**

Decides what background color a changed but not yet committed box should have. Does not apply to the main subtitles edit box itself.

- **Font**

Decides the font and fontsize used for the subtitle edit box and other edit boxes.

General -> Subtitles grid



- **Allow grid to take focus**

When enabled, the subtitles grid acts as its own area of the program and it can have focus, just like the audio or the video can, and while it does you can use the arrow keys/mouse wheel to scroll around it etc. On the other hand, if you disable this option, the focus will stay where it was before whenever you click in the grid. This means you can't use keyboard shortcuts in the grid anymore, but on the other hand it means you can click in the grid to go to a line without losing the audio focus and so on. Use at your own discretion.

- **Highlight subtitles that are currently visible in the video**

When enabled, all subtitle lines that are currently visible in the video frame (or at least *should* be visible; Aegisub does not account for alpha and such trickery in this case, it only cares about the timing of the line) will be highlighted in the grid with a special background color (see the "Line in frame background" option below).

- **Standard foreground**
- **Standard background**

The normal color of lines in the grid. "Foreground" is the text color, and "Background" is obviously the background color.

- **Selection foreground**
- **Selection background**

The color of selected lines in the grid.

Comment background

- Selected comment background

The background color of commented-out lines and selected commented-out lines, respectively.

- Collision foreground

The text color of lines whose timings overlap.

- Line in frame background

The background color of lines currently visible in the video frame.

- Header
- Left column
- Active line border
- Lines

The color of the grid lines and fixed columns/headers.

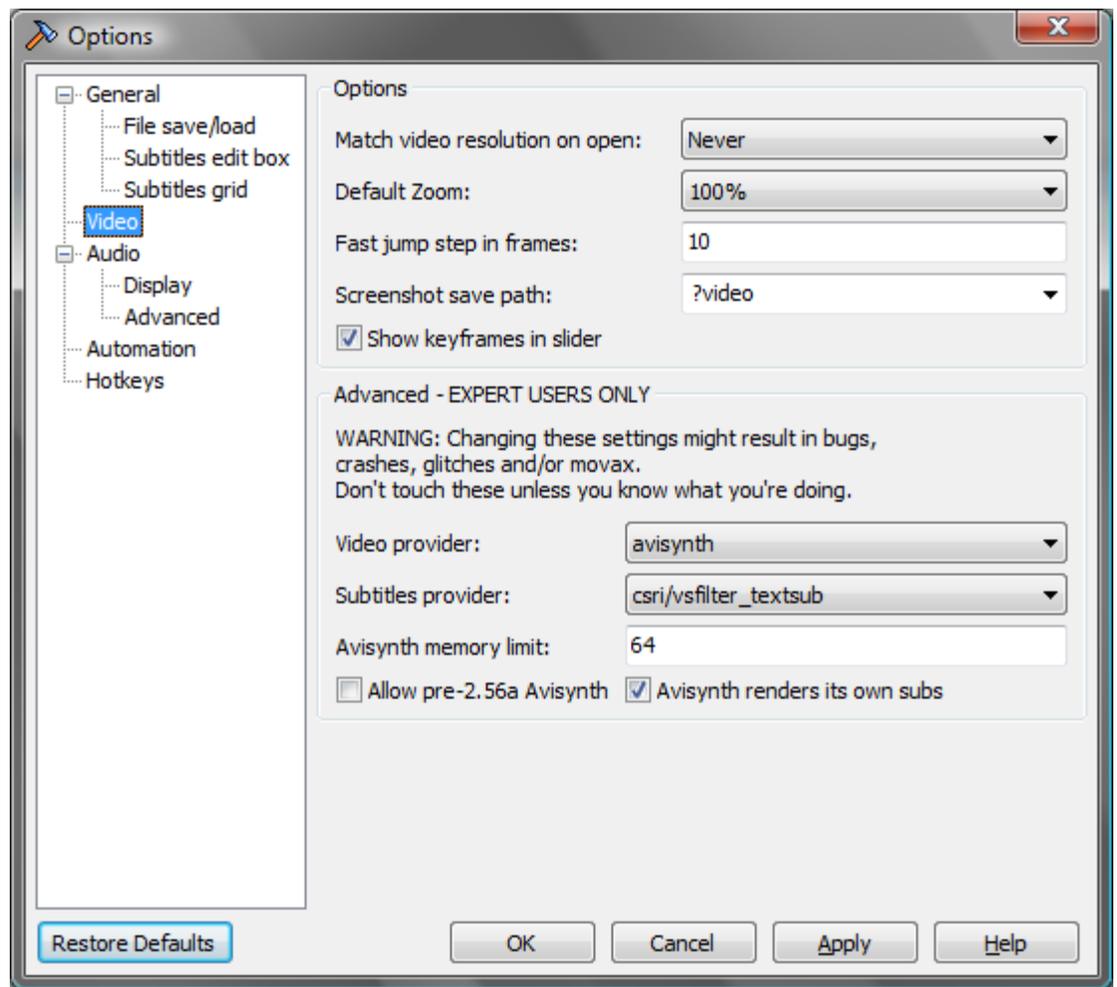
- Font

Decides the font and font size of all text in the grid.

- Replace override tags with:

The character that will be shown instead of override blocks if tag hiding is active.

Video



- **Match video resolution on open**

Controls what Aegisub will do about script resolution when you open a video. If set to "never", Aegisub will do nothing if the script resolution doesn't match the video resolution. If set to "ask", Aegisub will ask you if you want to change the script resolution to match the video resolution if they don't match. If set to "always", Aegisub will always change the script resolution to match the video resolution automatically. "always" is not recommended since it just changes script resolution without resampling anything, so it will most likely break existing typesetting.

- **Default zoom**

The default video zoom level. Useful if you have a very big or very small screen.

- **Fast jump step in frames**

Decides how big "jumps" Aegisub will make when you use the fast seek feature (Alt-rightarrow and Alt-leftarrow). Measured in frames.

- **Screenshot save path**

Decides where Aegisub should save screenshots. The default is `?video`, which means they are saved to wherever the video is, but you can change it to any path you like. [Aegisub path specifiers](#) are supported; another option directly available in the dropdown is `?script`, which is wherever the script is.

- **Show keyframes in slider**

When enabled, Aegisub will draw keyframe markers on the video seek slider. Note that this is not supported for all video formats and/or video providers.

- **Video provider**

Decides what method Aegisub should use to load video. What options you have available here depends on how your copy of Aegisub was compiled and what operating system you are running under. The following alternatives exist:

- *avisynth* (Windows only)

Uses [Avisynth](#) to load video. Versatile, supports loading almost all common formats as well as .d2v files (indexed DVD VOB's) if the correct plugin is supplied. Note that Aegisub can install its own avisynth.dll instead of using your system installation if so desired. Requires Video for Windows decoders for AVI files for best performance. Uses ffmpegsource() to load anything that isn't AVI (and tries that too for AVI's if no suitable VfW decoder is found).

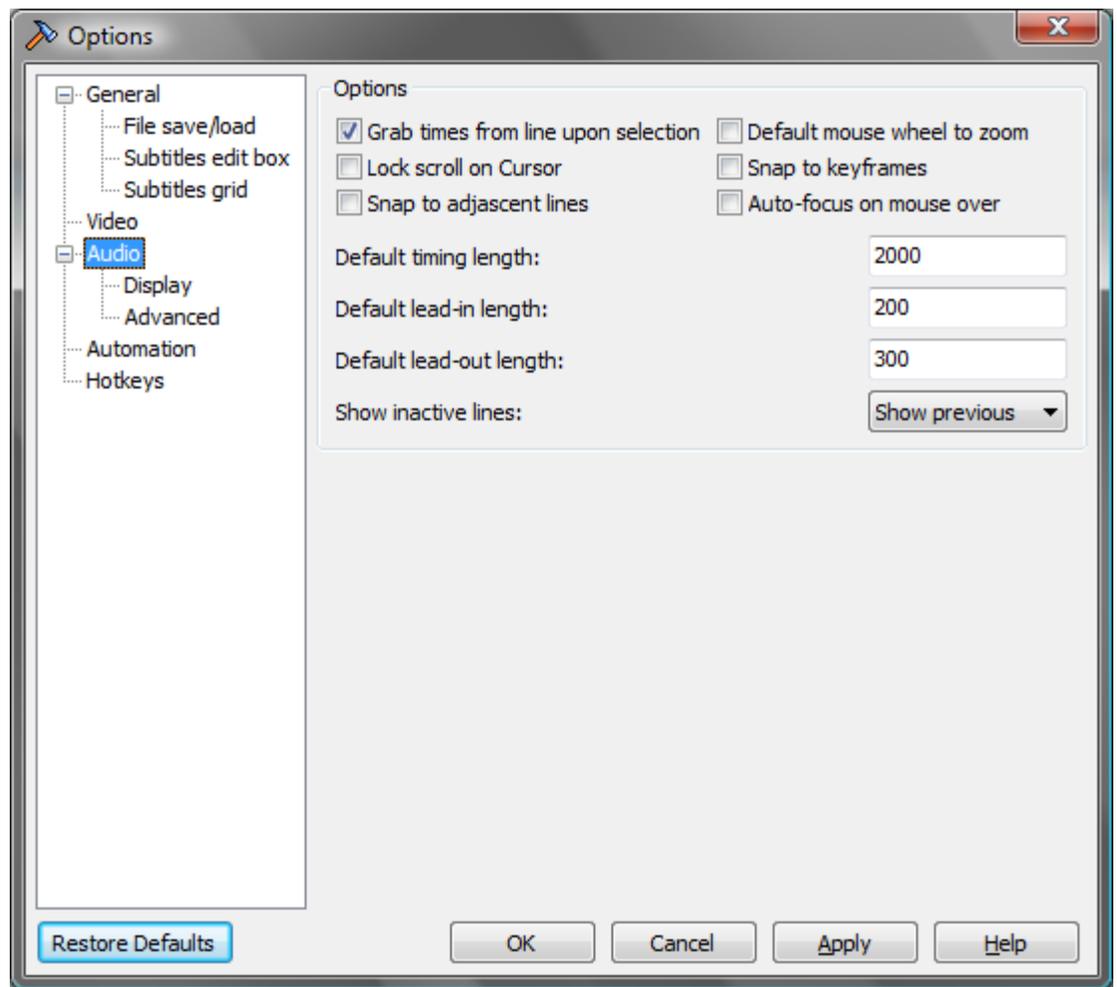
- *dshow* (Windows only)

Uses Microsoft's DirectShow framework to load video. Supports loading anything you can play in, say, Windows Media Player but is slower than Avisynth. May work with really odd formats that are not supported by ffmpegsource() but you do have a DirectShow decoder for.

- *ffmpeg*

Uses ffmpeg directly to load video. Supports most common formats. Technically there's nothing that stops you from using this on Windows, but most Windows builds (including the official Aegisub release builds for Windows) don't have it compiled in at all (since Avisynth uses ffmpegsource() anyway).

Audio



- **Grab times from line upon selection**

This option should really be named "Selecting lines changes audio selection", because that's what it does. When enabled, clicking a line in the grid or going to it using the next/previous line hotkeys will select the corresponding area in the audio waveform. When disabled, the selected area will always stay the same until you change it.

- **Lock scroll on cursor**

When enabled, the audio waveform view will automatically scroll to follow the playback cursor if it gets too close to either of the edges.

- **Snap to adjacent lines**

When enabled, Aegisub will snap the audio cursor to adjacent line start or end times if it is close enough.

- **Default mouse wheel to zoom**

When enabled, the mouse wheel will zoom the audio display horizontally. If disabled, it will scroll the audio display instead.

- **Snap to keyframes**

When enabled, the audio cursor will snap to keyframe lines (if you have any loaded, either from a video or from a keyframe file) if it is close enough.

- **Auto-focus on mouse over**

If enabled, moving the mouse cursor over the audio waveform will automatically

give it focus (as opposed to requiring a click).

- **Default timing length**

The default length of a new untimed line, in milliseconds.

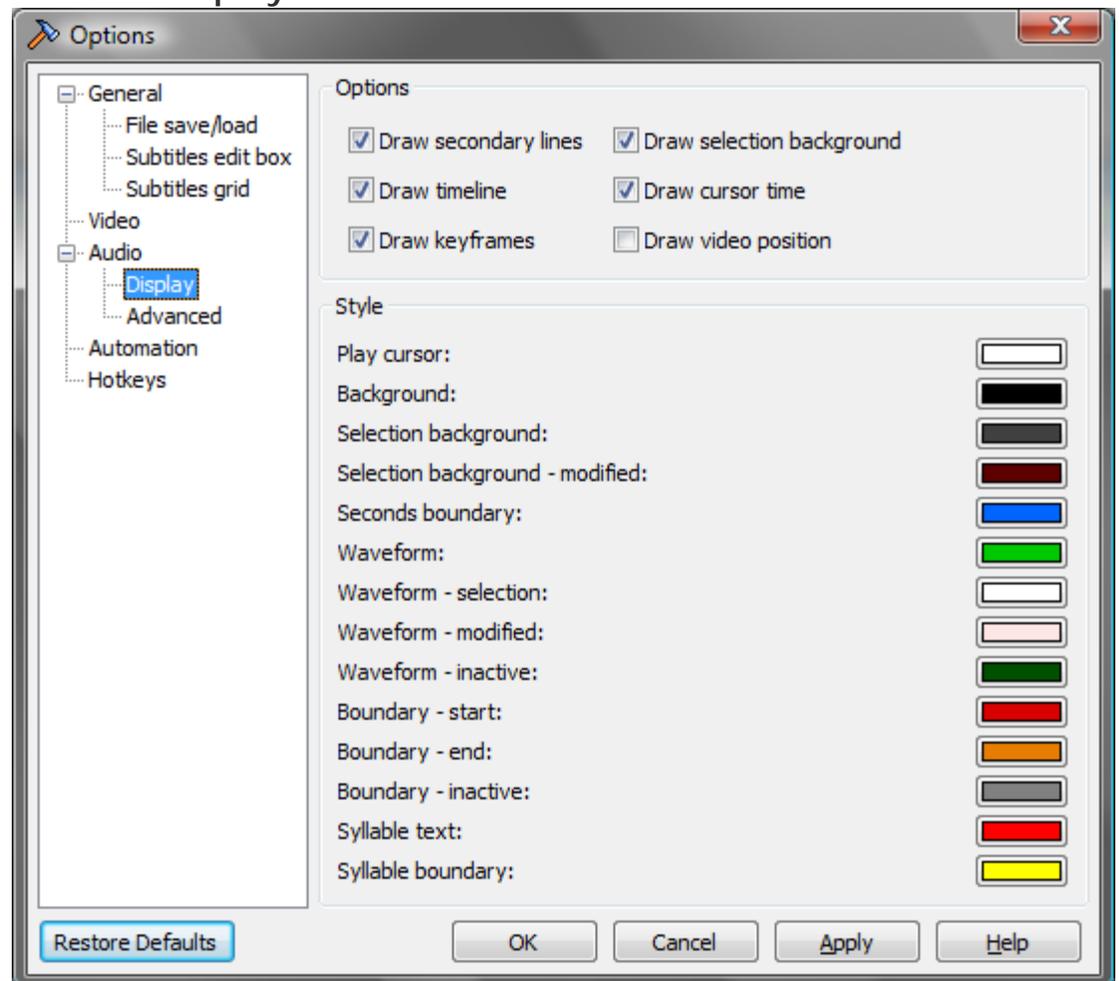
- **Default lead-in length**
- **Default lead-out length**

The duration added to the start of a line by the "add lead-in" and "add lead-out" functions. Also used in the [timing postprocessor](#).

- **Show inactive lines**

Controls how lines surrounding the currently selected line are displayed in the audio waveform. "Don't show" will only display the current line, "Show previous" will show the previous line (in the grid, *not* chronologically) in gray in addition to the current line, and "Show all" will show all lines in gray in addition to the current line.

Audio -> Display



- **Draw secondary lines**

If enabled, lines marking seconds will be drawn in the audio waveform.

- **Draw timeline**

If enabled, a timeline "ruler" with markers for each quarter second will be displayed below the audio waveform.

- **Draw keyframes**

If enabled, lines marking the positions of keyframes in the video will be drawn on the audio waveform.

- **Draw selection background**

If enabled, the selection will be drawn in an alternate color; if disabled only the line start and end markers will show where the selection is.

- **Draw cursor time**

If enabled, a timestamp showing the time since the start of the file will be drawn near the top of the audio waveform cursor.

- **Play cursor**

The color of the playback cursor.

- **Background**

The color of the audio waveform background.

- **Selection background**

The color of the audio waveform background when it is covered by the selection.

- **Selection background - modified**

The color of the audio waveform background when it is covered by the selection and the selection is detected as modified.

- **Seconds boundary**

The color of the second marker lines.

- **Waveform**
- **Waveform - selection**
- **Waveform - modified**
- **Waveform - inactive**

The color of the waveform itself in various situations (selected, detected as modified, and covered by an inactive line selection respectively).

- **Boundary - start**
- **Boundary - end**
- **Boundary - inactive**

The respective colors of the various line boundary markers.

- **Syllable text**

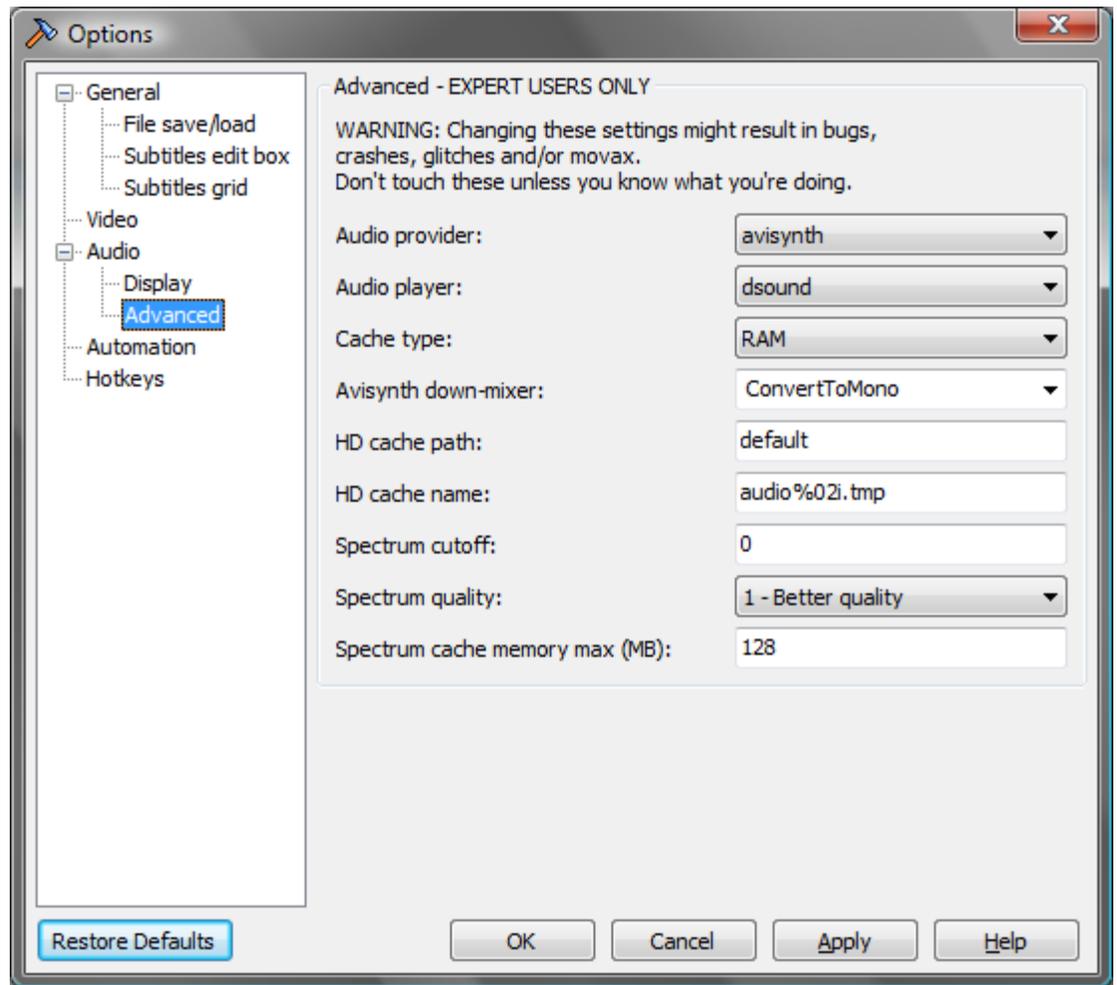
The color of a syllable's text in karaoke mode.

- **Syllable boundary**

The color of a syllable boundary line in karaoke mode.

It is not possible to change the color scheme of the spectrum display mode.

Audio -> Advanced



- **Audio provider**

What method to use for loading audio. Currently there are only two methods, which one is used depends on the platform.

- *avisynth* (Windows only)

Uses [Avisynth](#) to load audio. All file types will be loaded with `DirectShowSource()`, except for AVS files which will be opened with `Import()`.

- *lavc*

Uses `ffmpeg` directly to load audio. Supports most common formats. Technically there's nothing that stops you from using this on Windows, but most Windows builds (including the official Aegisub release builds for Windows) don't have it compiled in at all.

Regardless of this setting, the internal PCM WAV reader will always be tried first for WAV files.

- **Audio player**

What method to use for playing back audio. The options depend on the platform.

- *dsound* (Windows only)

Uses Microsoft DirectSound to play back audio. This is the best tested and most stable audio player.

- *alsa* (Linux only)

Uses the [Advanced Linux Sound Architecture](#) to play back audio. ALSA is the native sound architecture of Linux and is not available on any other systems. *It is possible to select the output device to use, but this is not exposed in the*

Options dialogue yet.

- *pulse* (Linux and other *NIX-like systems)

Plays sound back through a [PulseAudio](#) sound server.

- *portaudio*

Use the [PortAudio](#) API to play back sound. PortAudio has different playback implementations on different platforms. On most Unixes it uses Open Sound System (OSS) for output. PortAudio currently doesn't seem to work on Mac, and it is not included in Windows builds as the DirectSound player is more reliable.

- *openal* (Windows and Mac)

Uses the [OpenAL](#) API to play back audio. It is the only working audio player on Mac. It is not included in Windows builds due to the extra dependency it creates, and DirectSound is more reliable.

- **CacheType**

Use RAM unless you have very little of it, then use Hard Disk. The cache is not needed and not used when PCM WAV files are opened. If you disable caching, audio playback might become very unreliable.

- **Avisynth down-mixer** (Windows only)

Aegisub can only use mono (single-channel) audio. This option determines which Avisynth function to use to convert audio to mono. This option is only available in the Windows version of Aegisub.

- **HD cache path**

- **HD cache name**

These options determine where the hard disk audio cache will be located. Only used if cache is set to hard disk. You shouldn't need to change this unless you're low on disk space. For the name, the string expects a printf-style "%i" parameter, that will be replaced with a number. "%02i" is used by default, don't change that unless you know what you are doing.

- **Spectrum cutoff**

The index of the smallest frequency band the audio display will show in spectrum mode. There's usually no need to change this.

- **Spectrum quality**

Determines the quality of the audio spectrum display. Higher quality settings result in larger CPU and RAM use. Each consecutive setting uses a bit more CPU than the previous, and double the amount of RAM. For 48 kHz samplerate audio, one minute of audio uses this much memory at the different settings:

- 0 "regular" - 11 MB
- 1 "better" - 22 MB
- 2 "high" - 44 MB
- 3 "insane" - 88 MB

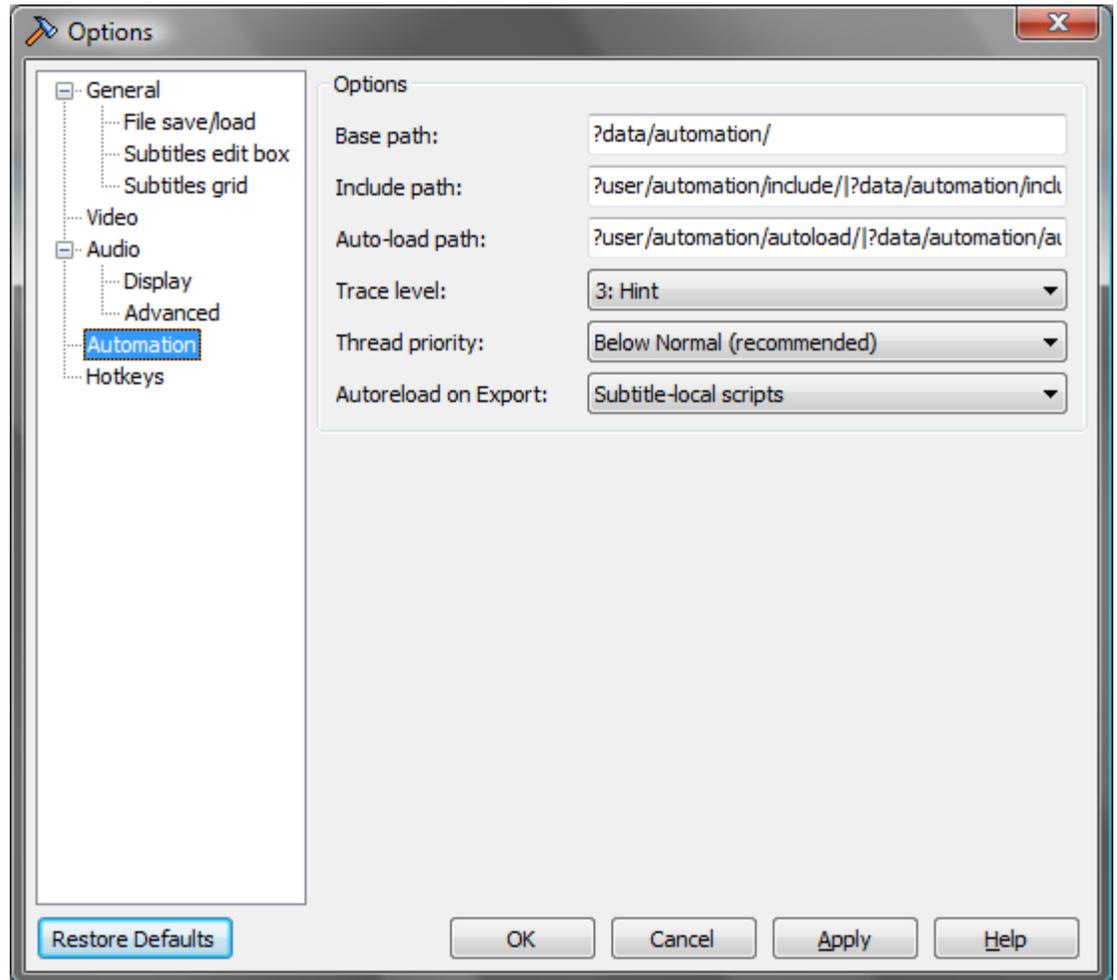
The amount of memory used does not depend on the number of channels (Aegisub always works in mono) or the bit-depth (the spectrum is always calculated in 32 bit floating point) of the audio.

- **Spectrum cache memory max**

The maximum amount of memory to use for audio spectrum caching. The results of

the calculations to display the audio spectrum are cached to make scrolling through the audio smoother. The amount of spectrum display that can be cached in an amount of memory depends on the quality setting above. The default cache size of 128 MB results a little less than 6 minutes of audio at 48 kHz in quality 1. If you set this smaller than 5 MB, the default of 128 MB will be used instead. You probably shouldn't set this to more than 1/4 of your amount of physical RAM installed.

Automation



- **Base path**
Not used by the Lua scripting engine. In the future this might be used by other scripting engines.
- **Include path**
List of directories where include files are searched for. Directories are separated with a pipe character, |.
- **Auto-load path**
List of directories that are searched for scripts on startup, which are then automatically loaded. Directories are separated with a pipe character, |.
- **Trace level**
When a script sends a message to the debug console it can also specify a trace level.

If the trace level of a message is lower than the value given here, the message is not logged. The names given to the levels are only suggestions, they don't have any effect on the execution of the script. (Eg. a "Fatal" level message will not cause the script to terminate.)

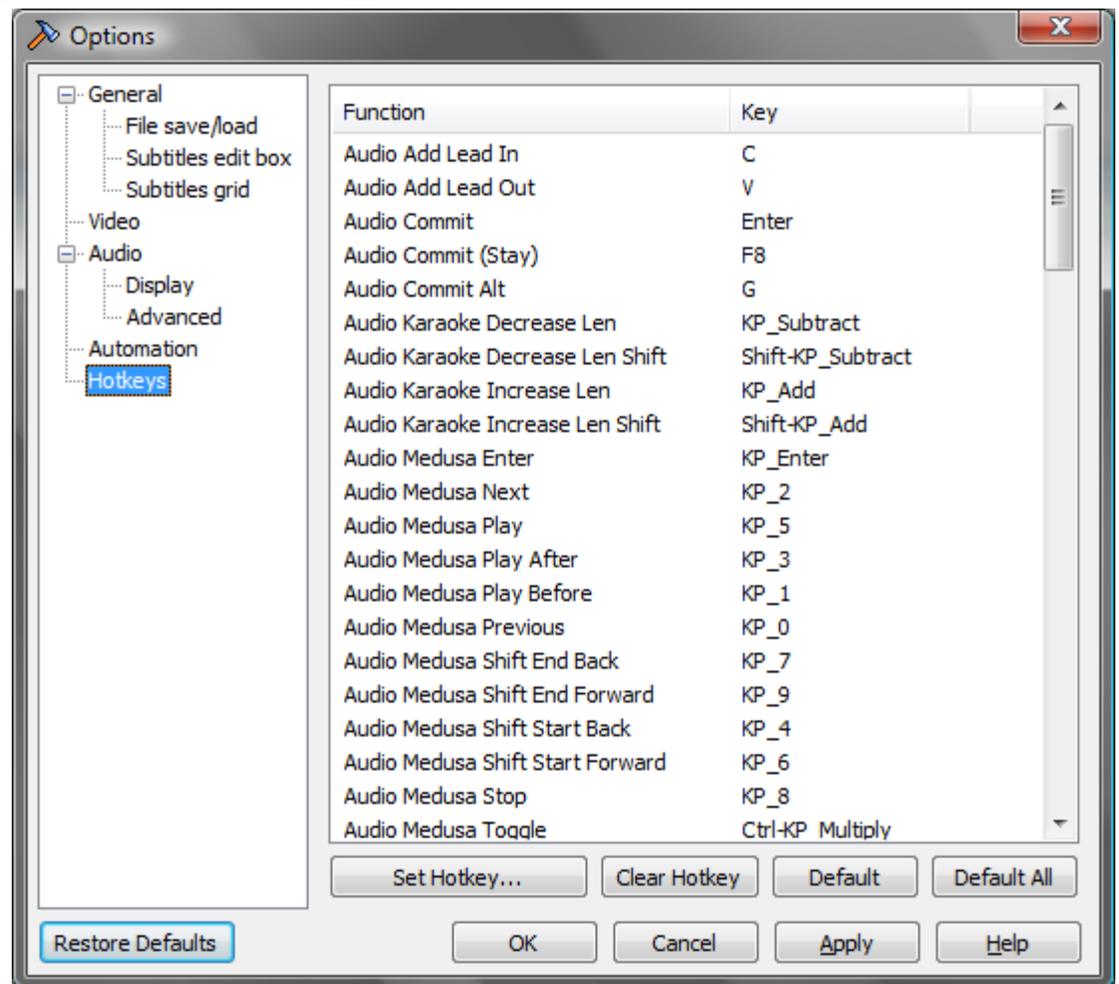
- **Thread priority**

Priority given to the script execution thread. If you're on a single-core/single-CPU system having this on lower than normal will make other programs more responsive while long-running scripts are active.

- **Autoreload on Export**

Automatically reloads the specified sets of scripts when the [Exporting](#) dialogue is opened. Note that if a script fails to reload, no errors are displayed, it just won't show up in the Export window then. In that case you will have to enter the [Automation Manager](#) window and determine the cause of the error.

Hotkeys



This page shows a list of all functions in Aegisub that can be bound to keyboard shortcuts, and what shortcut (if any) they are currently assigned to. Should be self explanatory.

Double click on a hotkey in the list to change it.

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

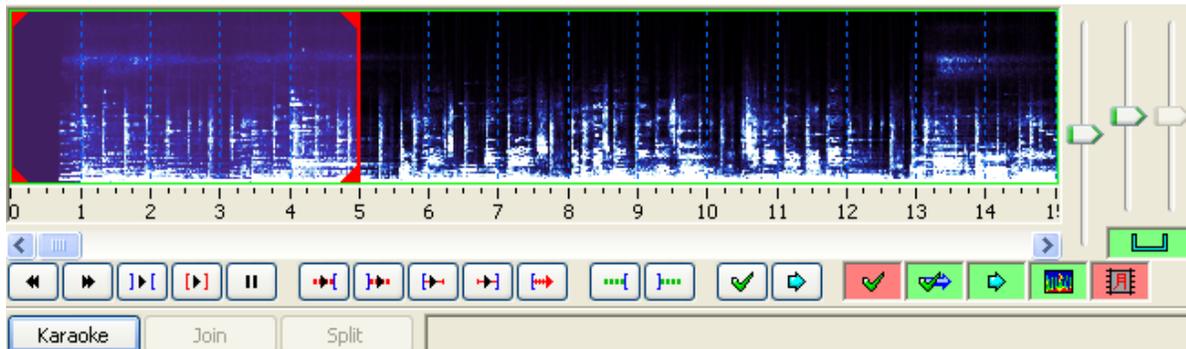
» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

File:Audio-box-spectrum.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Audio-box-spectrum.png](#) (654 × 193 pixels, file size: 76 KB, MIME type: image/png)

The audio box with spectrum display.

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail

Dimensions User Comment

current contribs (The audio box with spectrum display.)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Audio](#)

Category:Pages with Todo items

This category is for all pages with todo items. These should be resolved as quickly as possible, and none should go into a "release" version.

Use the `{{Todo|Text for todo item}}` template to add pages to this category.

Pages in category "Pages with Todo items"

The following 13 pages are in this category, out of 13 total.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

A

- [ASS Tags](#)
- [Audio](#)
- [Automation 4 Lua Configuration dialogues](#)

C

- [Credits](#)

E

- [Editing Subtitles](#)

F

- [Furigana karaoke](#)

K

- [Karaoke Templater](#)
- [Karaoke Templater Reference: Template execution rules and order](#)
- [Karaoke Templater Tutorial 1](#)

K cont.

- [Karaoke Templater Tutorial 2](#)
- [Karaoke Timing Tutorial](#)

T

- [Tutorials](#)
- [Typesetting](#)

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Karaoke Timing Tutorial

This tutorial will teach you how to load a song into Aegisub, how to enter the words for the song and how to add time-codes to the words to synchronise them with the song.

You don't need to have used Aegisub before to follow this tutorial.

Before we begin

There's a few things you need to have ready before starting:

- The song itself. This can for example be as an MP3 file or inside a video. Aegisub can read the sound from video files, you don't need to create a separate sound file if the song is on a video.
- The words for the song. It's easiest if you have them in a plain text file (.txt file) broken into verses and stanzas already.

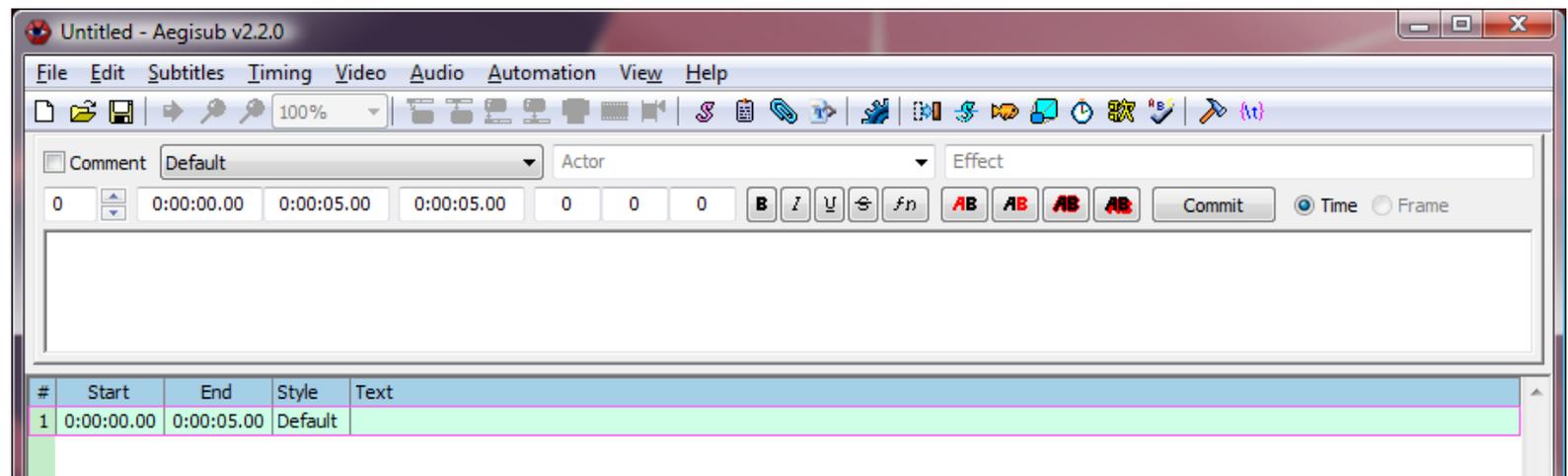
I am using an English song for demonstration here, but many of the more advanced functions in Aegisub are designed for use with songs in Japanese and other languages that often need transcription or transliteration into Latin script. I will show how to use those in a video tutorial.

Contents [\[hide\]](#)

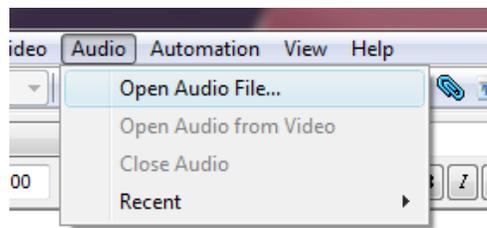
- 1 Before we begin
- 2 Loading the song
 - 2.1 Tips
- 3 Entering the words
- 4 Rough timing, the lines first
 - 4.1 Tips
- 5 Fine timing, words and then syllables
- 6 Styling
- 7 Wrapping up

Loading the song

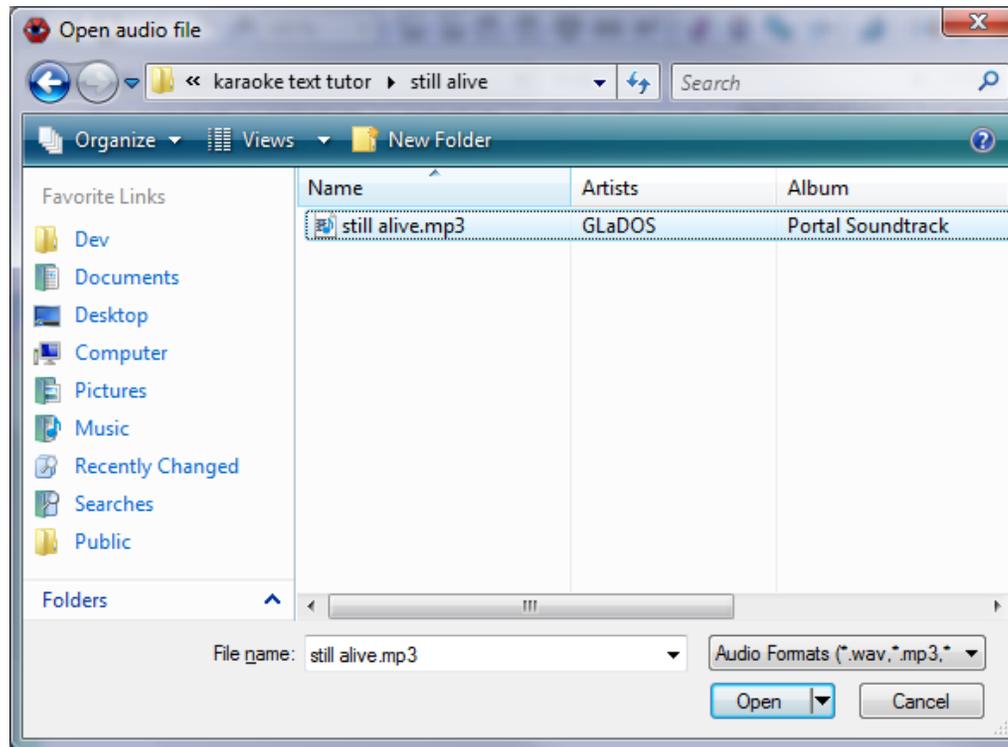
We'll start by creating a new file. You already have that if you have just started Aegisub.



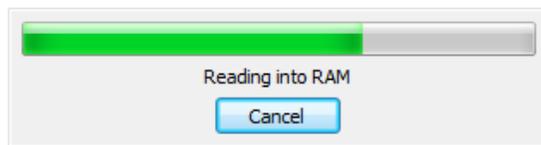
Now open your song. Select **Open Audio** from the **Audio** menu...



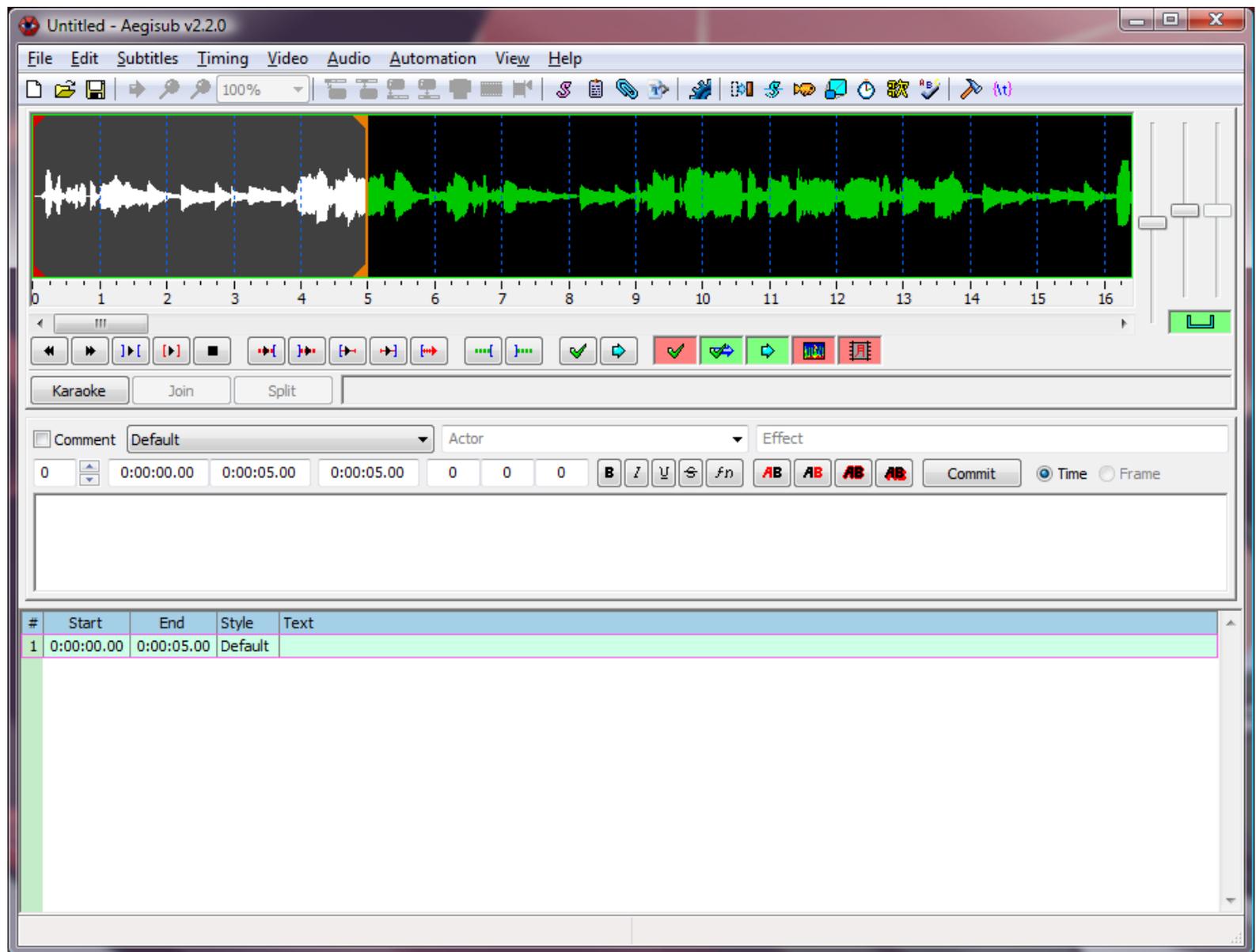
...then select your song file.



Aegisub will now spend a short while reading the audio file.



When it's done, you should have a waveform display (the audio display) at the top of the Aegisub window. If you've used Aegisub before things might look a little different, it might be easier to follow the rest of this tutorial if you set things to look like on this picture.



We'll look at how to use the audio display for timing in a moment, but first let's get the lyrics for the song loaded.

Tips

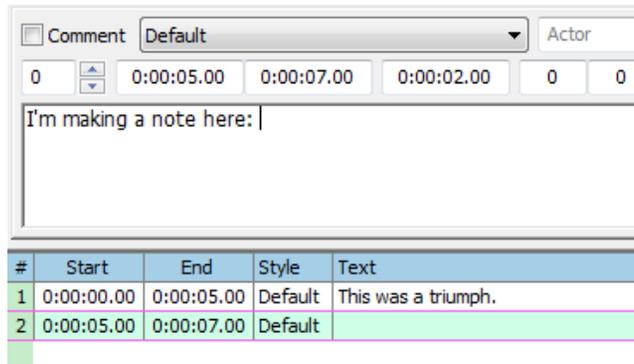
Loading audio directly from video files: You can select to choose from video files in the Open Audio file selector. This won't actually open the video in the file, it will just read the audio from the video file, the same as if it was just the audio in a separate file.

Instant loading of WAV files: If you have an uncompressed PCM WAV file, Aegisub can open it instantly without first loading it entirely into memory. This can be a great time-saver, but of course it requires a bit of extra disk space and probably also a bit of work beforehand to even create the WAV file. (Remember that this only works with *uncompressed PCM* files, things like ADPCM or MP3 in WAV files won't work and will

still trigger the pre-loading.)

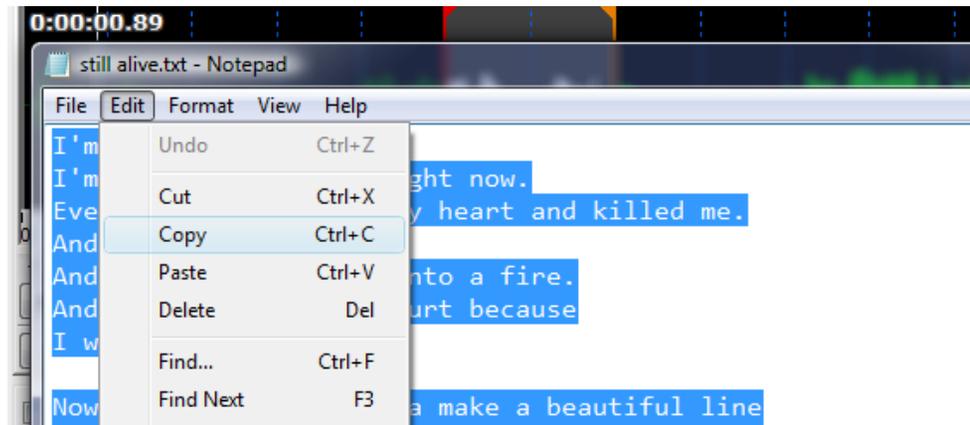
Entering the words

Now, to get the text in, we could just start typing it in...



But **don't do that!** You'll get much huger success if you have it all in a text file, copy it from there and paste it into Aegisub. (You can often also just copy-paste directly from your favourite lyrics website.)

I have the lyrics in a text file, so I open that, select the text and copy it to the clipboard.



Now things get a little complicated, but don't worry, it really isn't hard :-)

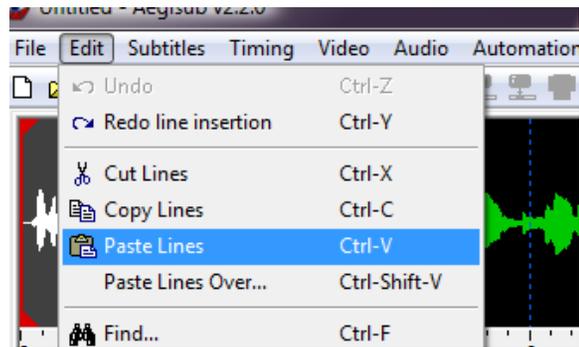
There are two different places you can paste into Aegisub: The subtitles grid and the subtitles edit box. When you paste into the subtitles grid, you create new lines in the subtitle file. When you paste into the subtitles edit box you change the currently selected subtitle line.

We want to make sure we paste into the subtitles grid, so click once inside the grid area (at the bottom of the window) to set the input focus to that.

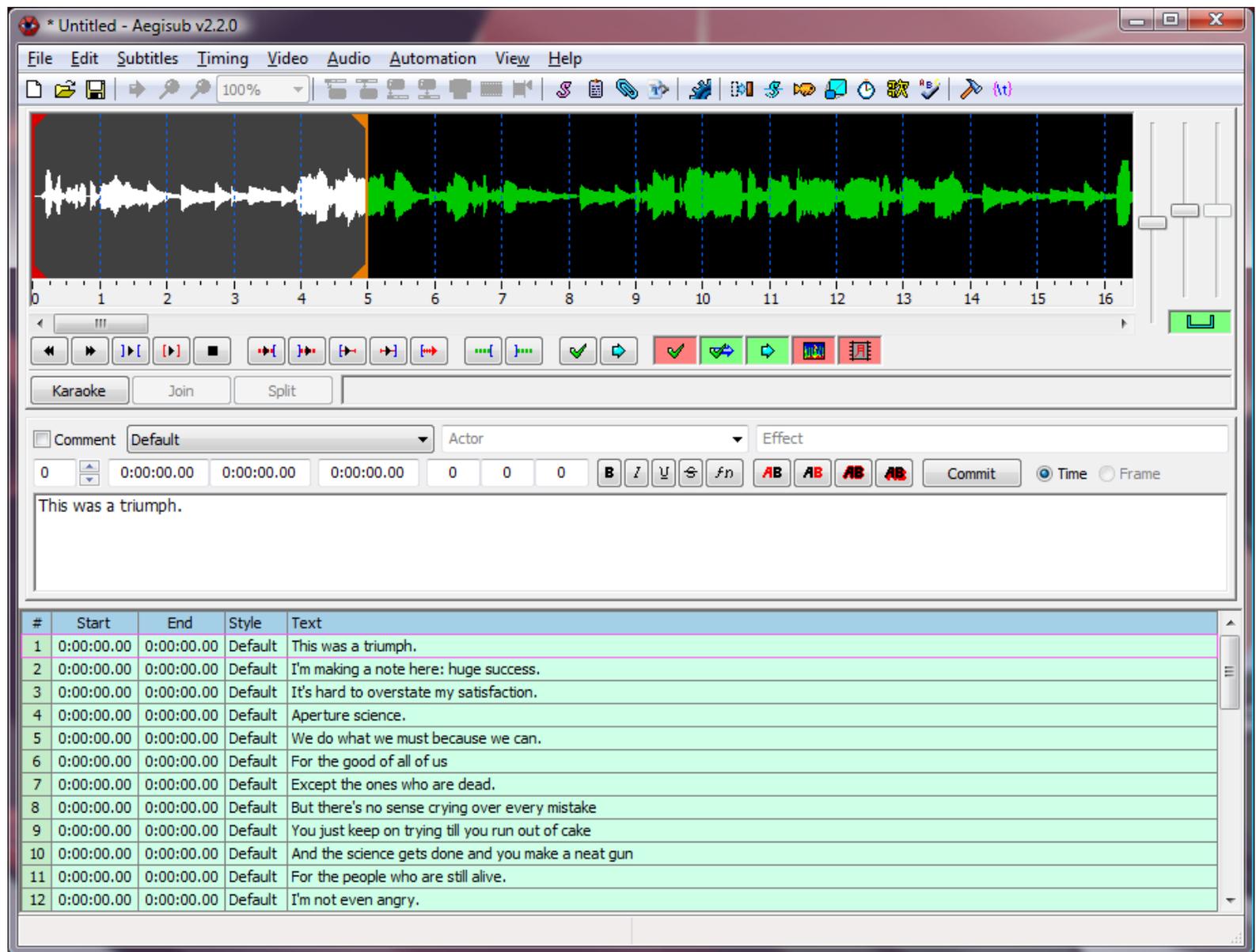
#	Start	End	Style	Text
1	0:00:00.00	0:00:05.00	Default	

Click!

And now we can paste in the lyrics.

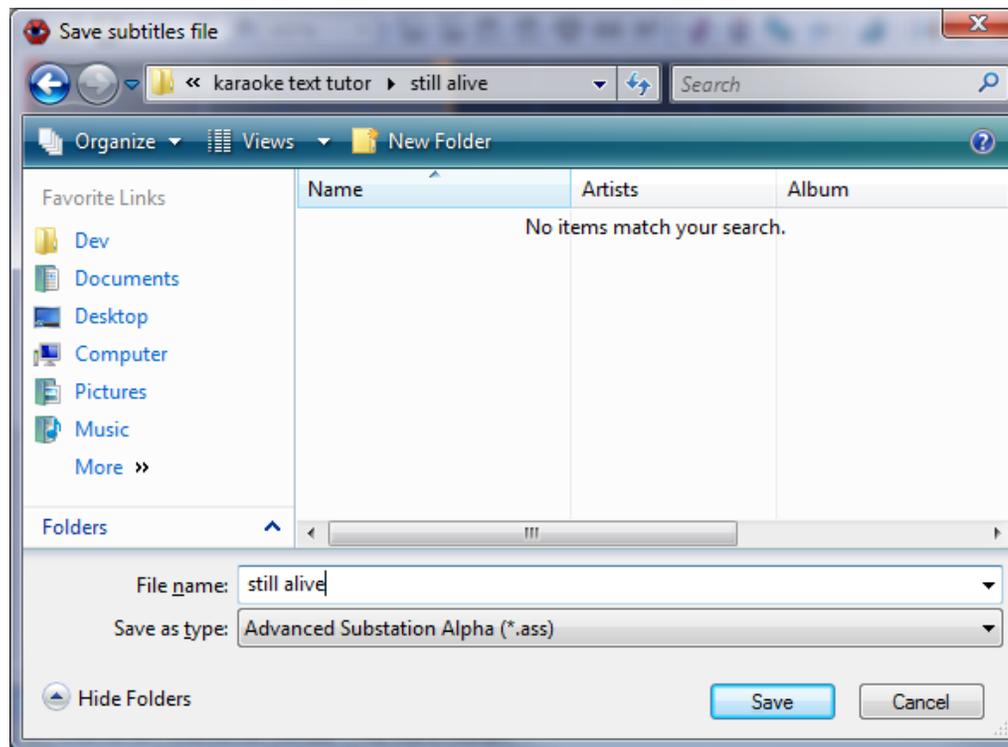


They should appear immediately as lines in the grid. Notice that they all have start and end times set to zero. This makes it easier when we're going to time each line of the lyrics to the song.



It might be a good idea to save your file now, just so you can easily save it later on without having to give it a name.

Remember that Aegisub automatically saves a copy of your file every minute, even if you haven't given it a name yet, so you rarely lose much work if something goes wrong.

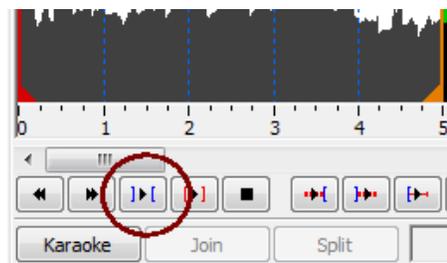


Now we're ready to time the individual lines in the lyrics.

Rough timing, the lines first

Before we start with timing, you should know that the way presented here is just one of many. There's several ways you can time to audio in Aegisub and this one might not be the best one for you. Try to also explore the program and see if you can find your own best way to do it. This is just the way I (jfs) usually do it.

First let's look at how to get around the audio display and play the audio. You might already have noticed that there's no less than 6 different "Play" buttons. Usually you'll just use one of them, though: The one with the blue outward-pointing brackets around. That one is Play Selection, and plays the part of the audio that's currently highlighted.



Try pressing the Play Selection button, you should hear the first 5 seconds of the song played. (Aegisub selects the first 5 seconds by default.)

Now try changing the selection: You can left-click and drag in the audio display to select the part you've dragged across. If you click and drag

on the left or right edge of the selection you can change just the start or end. Finally, you can make a single left-click (without dragging!) anywhere to set the selection start right to that point, and you can make a single right-click to set the selection end.

Let's time the first line. Find the start and end of the first line of the song you're working on and make sure the audio selection matches it exactly. Notice that at first, the selection was gray but as soon as you started changing it, it became red and the word "Modified" appeared. This means that you have changed the selection but not saved (committed) the new timing.

To commit the timing and store it back to the subtitle line, just press the Commit button, the green check mark.



When you commit, you will also be sent to the next line automatically, so you can immediately continue timing that.

Just continue timing like that until you have covered all the lines of the song: Find start and end of line, set the selection and then commit.

When you're done, save the file.

Tips

Timing from audio isn't hard at all, but here's some tips to make it even easier and also a lot faster!

Hotkeys: There's a number of keyboard shortcuts that can make audio timing much faster to work with.



The most important ones are:

- S - Play Selection: Play back the currently selected audio.
- A and F - Scroll Left and Right: Change the portion of the audio visible.
- G - Commit: Copy the start and end times of the current audio selection into the line selected in the subtitles grid and move to the next line.

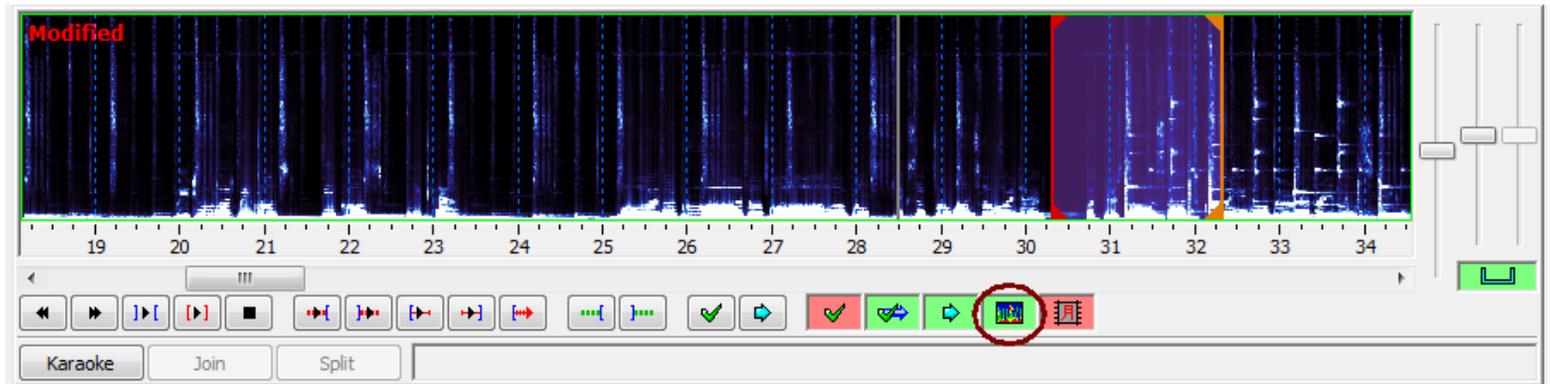
Play near start/end: There's four buttons (hotkeys Q, W, E and D) that play half a second just before or just after the start and end of the selection. You can use these to more accurately set the start and end to exactly where the singing starts/ends.

Change selection while playing: While audio is playing you can still change the selection. You won't see any difference if you change the

selection start, but if you change the selection end, playback will now end when it hits the new selection end. This way you can quickly stop playback by setting the end close to the playback cursor (the white line that moves while Aegisub is playing) or extend the playback to go even further.

For example, when looking for the start of the first line, you can just start playback with the initial 5 second selection and continue extending it until you find the line. Then, while it's still playing, you can set the right start time and then the end time. When you have the line approximately down like that, you can do an extra check by playing the entire selection again, or by using the Q/W/E/D keys to play the parts right around the start and end times.

Spectrum mode: Usually the audio display is in waveform mode, this is what I've shown on all the screenshots so far. But actually Aegisub has a much cooler way of showing the audio: Spectrum mode.



The spectrum mode takes more CPU and RAM than waveform mode, but it gives a better picture of the audio and with a bit of training you can learn to tell singing from music and even how different sounds look. For example, S sounds are very easy to recognise.

Zooming and scaling: You can use the slider bars to the far right of the audio display to zoom in and out on the audio and to change the volume.

Fine timing, words and then syllables

Todo:

Click Karaoke button.

Time words.

Click Split button. Place split markers. Click Accept Split button.

Time syllables.

Commit.

Repeat.

Styling

Todo: a bit about styles, how basic karaoke looks, and the \kf and \ko effects

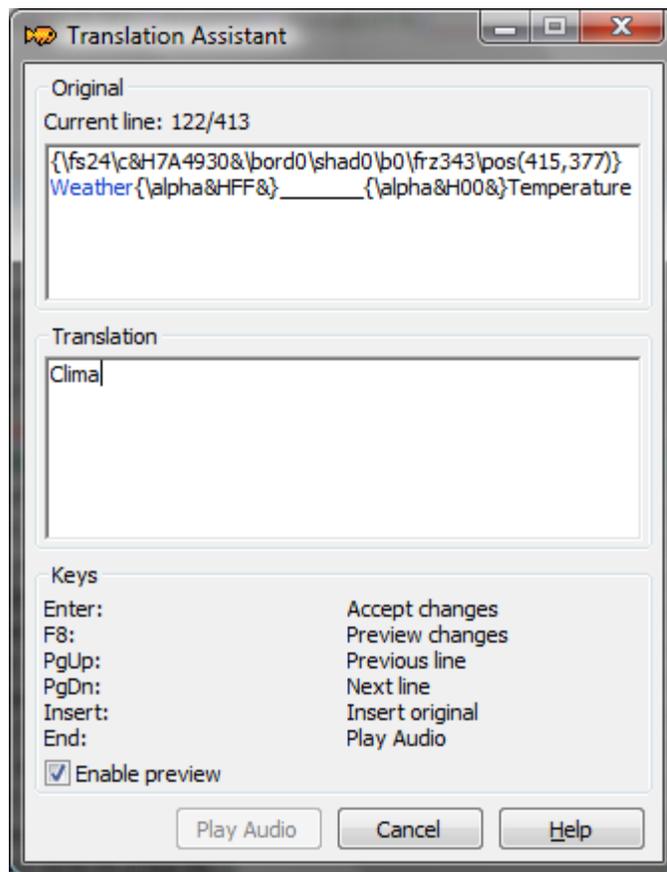
Wrapping up

Todo: mention the video tutorial again and point to other relevant topics

Category: [Pages with Todo items](#)

File:Translation Assistant.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Translation_Assistant.png](#) (336 × 437 pixels, file size: 28 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Translation Assistant](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

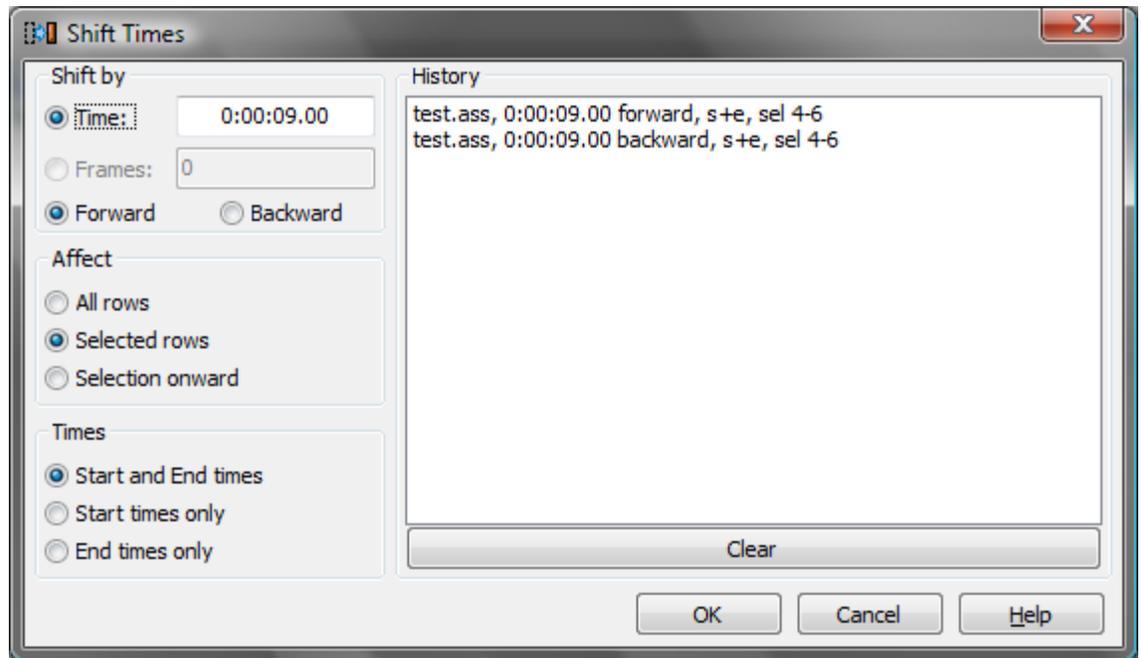
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Shift times.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Shift_times.png](#) (567 × 330 pixels, file size: 25 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Shift Times](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Glossary:Karaoke effect

Karaoke is (amateur) singing along to a song, usually a version of the song with the main vocals removed. When singing karaoke the lyrics to the song are usually presented on a TV screen and highlighted as they are to be sung.

In subtitling however, karaoke usually refers to those lyrics themselves. The **karaoke effect** refers to the way they are highlighted.

So, a karaoke effect is a visual effect synchronised with a song, used to help singing karaoke.

Reality has it that karaoke effects in the fan subtitling community today are used more for "eye candy" and showoff than the real purpose of karaoke subtitles, assisting the viewer sing along to the song.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

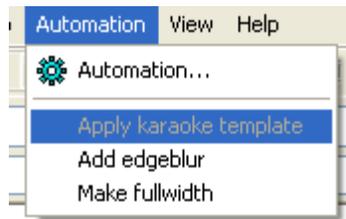
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Automation-menu-kara-templater-gray.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Automation-menu-kara-templater-gray.png](#) (173 × 113 pixels, file size: 1 KB, MIME type: image/png)

Automation menu pulled down, showing the Karaoke Templater item grayed out.

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time	Thumbnail	Dimensions	User	Comment
current contribs)	(Automation menu pulled down, showing the Karaoke Templater item grayed out.)			

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Karaoke Templater Tutorial 1](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Karaoke Templater Tutorial 2

In [the previous tutorial](#) we saw how to use the basic features of Karaoke Templater to make simple karaoke effects. We'll continue where we left off here, by expanding on the basics from last tutorial.

Todo: make and insert screenshots

Preparations

As before, you'll still need some timed karaoke and a video to preview the effects. I won't go into further details with that here.

Adding a fadeout

To recap, here's the effect from the end of last tutorial:

```
{\r\k$k\dur\t($start,$end,\lc&H00FF00&)\t($start,$mid,\fscy120)\t($mid,$end,\fscy100)}
```

We'll add a fadeout effect to this now, having each syllable fade out *after* it has been sung, not while. We'll have to do a little maths to make this: Start the fadeout at `$end` and have it continue until `$end+200`, ie. have it fade out for 200 milliseconds after the syllable.

Change the template to this:

```
{\r\k$k\dur\t($start,$end,\lc&H00FF00&)\t($start,$mid,\fscy120)\t($mid,$end,\fscy100)\t($end,!$end+200!,\alpha&HFF&)}
```

Then try applying templates again. You should see the old effect happen as usual, but this time afterwards, each syllable fades out.

The magic in this is the exclamation marks here: `!$end+200!`

When you have a pair of exclamation marks like that, everything in between them is treated as an *expression* (actually a very small Lua program, but don't worry about that yet.) Here we use an expression to take the end-time of the syllable and add 200 to it, getting a new number. The end result is that the `\t` fadeout effect lasts from `$end` and until 200 milliseconds later.

Tweaking the grow/shrink effect

Maybe you think the growing-shrinking effect looks a bit odd, switching right in the middle. It might look better if it was at the maximum height earlier, and used more time shrinking back to normal. Well, that can be changed:

```
{\r\k$k\dur\t($start,$end,\lc&H00FF00&)\t($start,!$start+$dur*0.3!,\fscy120)\t(!$start+$dur*0.3!,$end,\fscy100)}
```

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

With this, the growing-part will only take the first three-tenths of the syllable duration, and the shrinking the rest. We used one new variable here, $\$dur$. This is the duration of the syllable in milliseconds, just like $\$kdur$ is the duration in centiseconds. (We could actually just as well just have used $\$kdur$ here, and then multiplied by 3 instead of 0.3.)

Note that I removed the fadeout from here, it's just to make the line shorter and easier to read. You can add it back if you want.

Hopefully this tutorial has given you some more ideas of what you can do. In the next one we'll add another layer to the effect by using multiple templates!

Category: Pages with Todo items

File:Flag of Brazil.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Brazil.gif](#) (60 × 40 pixels, file size: 2 KB, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Denmark.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Denmark.gif](#) (60 × 40 pixels, file size: 933 B, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Sweden.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Sweden.gif](#) (60 × 40 pixels, file size: 1 KB, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) ↗ for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Canada.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Canada.gif](#) (60 × 40 pixels, file size: 836 B, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Germany.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Germany.gif](#) (60 × 40 pixels, file size: 754 B, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) ↗ for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of United-States.gif

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_United-States.gif](#) (60 × 40 pixels, file size: 2 KB, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Credits](#)

File:Flag of Italy.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Italy.gif](#) (60 × 40 pixels, file size: 611 B, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) ↗ for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Poland.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Poland.gif](#) (60 × 40 pixels, file size: 476 B, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Malaysia.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Malaysia.gif](#) (60 × 40 pixels, file size: 2 KB, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Russia.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Russia.gif](#) (60 × 40 pixels, file size: 787 B, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) ↗ for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Netherlands.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Netherlands.gif](#) (60 × 40 pixels, file size: 812 B, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Slovenia.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Slovenia.gif](#) (60 × 40 pixels, file size: 1 KB, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) ↗ for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flags of Japan.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flags_of_Japan.gif](#) (60 × 40 pixels, file size: 848 B, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of France.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_France.gif](#) (60 × 40 pixels, file size: 838 B, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Catalonia.png

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Catalonia.png](#) (125 × 83 pixels, file size: 387 B, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Credits](#)

File:Flag of Hungary.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Hungary.gif](#) (60 × 40 pixels, file size: 738 B, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) ↗ for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Finland.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Finland.gif](#) (60 × 40 pixels, file size: 1 KB, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Taiwan.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Taiwan.gif](#) (60 × 40 pixels, file size: 1 KB, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) ↗ for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Spain.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Spain.gif](#) (60 × 40 pixels, file size: 1 KB, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) ↗ for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Korea.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Korea.gif](#) (60 × 40 pixels, file size: 2 KB, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Soviet flag.gif

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

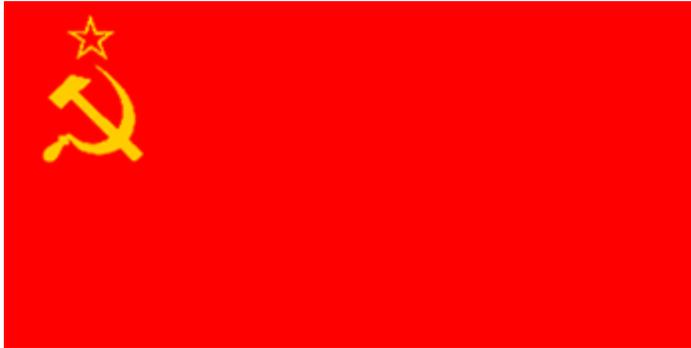
Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Soviet_flag.gif](#) (350 × 175 pixels, file size: 2 KB, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Credits](#)

File:Flag of Estonia.gif

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Estonia.gif](#) (60 × 40 pixels, file size: 498 B, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) ↗ for more information)

File links

The following page links to this file:

- [Credits](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Flag of Singapore.gif

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Flag_of_Singapore.gif](#) (60 × 40 pixels, file size: 787 B, MIME type: image/gif)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

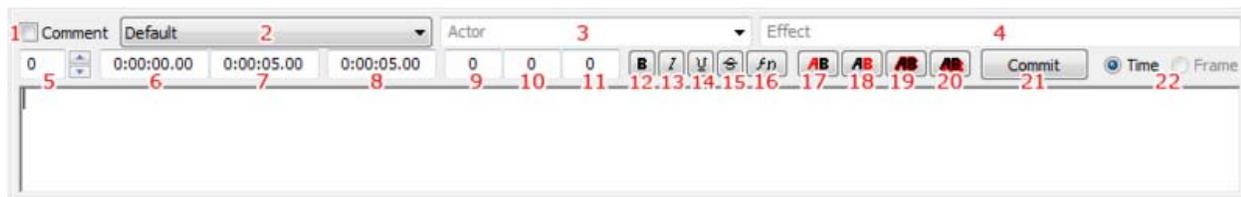
File links

The following page links to this file:

- [Credits](#)

File:Subs Edit Controls.png

- [File](#)
- [File history](#)
- [File links](#)



Size of this preview: 800 × 127 pixels

[Full resolution](#) (849 × 135 pixels, file size: 15 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Editing Subtitles](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Subs Edit Context.png

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

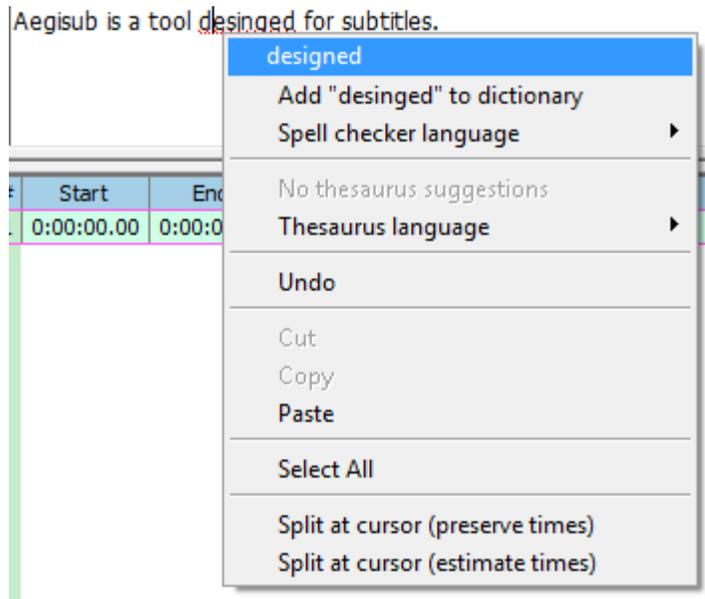
Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Subs_Edit_Context.png](#) (356 × 301 pixels, file size: 9 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Editing Subtitles](#)

File:Subs grid.png

- File
- File history
- File links

#	Start	End	Style	Text
13	0:01:18.41	0:01:21.87	Eva Op Roomaji	omoi de wo uragiru nara
14	0:01:22.13	0:01:25.63	Eva Op Roomaji	kono sora wo daite kagayaku
15	0:01:25.80	0:01:29.34	Eva Op Roomaji	shounen yo shinwa ni nare
16	0:00:00.00	0:00:00.00	Default	Kanji
17	0:00:01.03	0:00:06.77	Eva Op Kanji	残#酷#な天#使のようじ
18	0:00:07.01	0:00:14.48	Eva Op Kanji	少#年#よ 神#話#になれ
19	0:00:22.91	0:00:29.23	Eva Op Kanji	香#い風#がいま 胸#のドアを叩#いても
20	0:00:29.68	0:00:36.93	Eva Op Kanji	私##だけをただ見つめて 微笑#んでるあなた
21	0:00:37.70	0:00:44.23	Eva Op Kanji	そっとふれるものもとめることに夢中#で
22	0:00:44.60	0:00:51.36	Eva Op Kanji	運#命#さえまだ知らないたいけな瞳##
23	0:00:52.26	0:00:58.34	Eva Op Kanji	だけどいつか気付くでしょう その背中#には
24	0:00:59.75	0:01:06.26	Eva Op Kanji	運#か未来# めざすための 羽根#があること
25	0:01:07.14	0:01:10.70	Eva Op Kanji	残#酷#な天#使のテーゼ
26	0:01:10.94	0:01:14.43	Eva Op Kanji	窓#辺からやがて飛び立つ
27	0:01:14.65	0:01:18.07	Eva Op Kanji	ほとぼしる熱#いバトスで
28	0:01:18.41	0:01:21.87	Eva Op Kanji	思#い出を裏#切るなら
29	0:01:22.13	0:01:25.63	Eva Op Kanji	この宇宙を抱いて輝##く
30	0:01:25.80	0:01:29.34	Eva Op Kanji	少#年#よ 神#話#になれ
31	0:00:00.00	0:00:00.00	Default	Translation
32	0:00:01.03	0:00:06.77	Eva Op English	Just like the Cruel Angel,
33	0:00:07.01	0:00:14.48	Eva Op English	Young boy, rise as a legend!
34	0:00:22.91	0:00:29.23	Eva Op English	Even as the tender wind knocks at the door to your heart,
35	0:00:29.68	0:00:36.93	Eva Op English	You merely look at me and smile.

Size of this preview: 800 × 394 pixels

[Full resolution](#) (832 × 410 pixels, file size: 41 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

([Latest](#) | [Earliest](#)) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

([Latest](#) | [Earliest](#)) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Editing Subtitles](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Subs grid tags.png

- File
- File history
- File links

Navigation

- » Main Page

Introduction

- » What is Aegisub?
- » Highlights
- » Credits
- » Support Aegisub
- » FAQ
- » Tutorials

Working with Subtitles

- » Editing Subtitles
- » Exporting Subtitles
- » Applying Subtitles
- » Spell Checker
- » Translation Assistant
- » Paste Over
- » Select Lines

Typesetting

- » Introduction
- » Working with Video
- » Editing styles
- » Visual Typesetting
- » ASS Override Tags
- » Colour Picker
- » Styling Assistant
- » Resolution Resampler
- » Fonts Collector

Timing

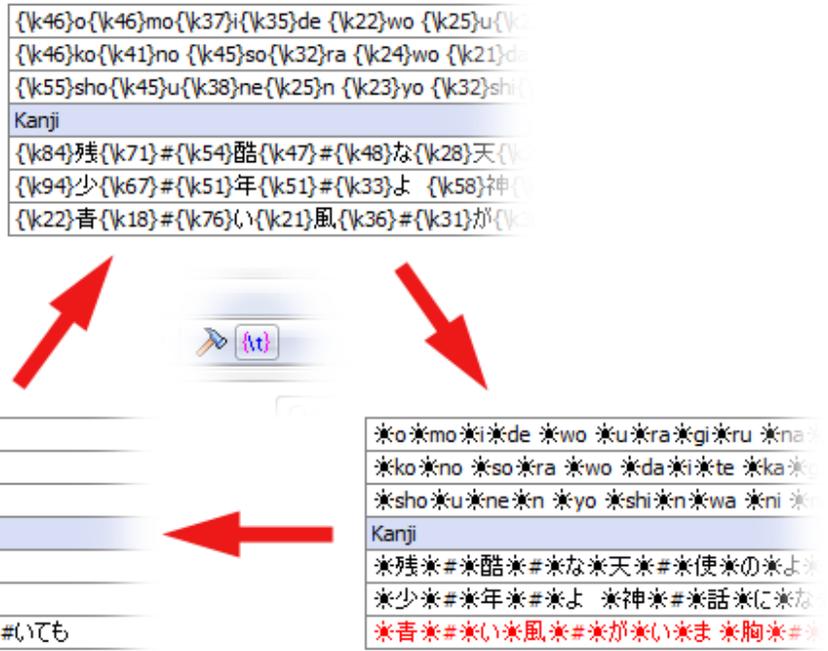
- » Working with Audio
- » Shift times
- » Timing Post-Processor
- » Kanji Timer

Automation

- » Overview
- » Karaoke Templater
- » Lua Reference

Miscellaneous

- » Aegisub Options
- » Script Properties
- » Attachment Manager



No higher resolution available.
[Subs_grid_tags.png](#) (600 × 350 pixels, file size: 32 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Editing Subtitles](#)

File:Subs grid menu.png

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

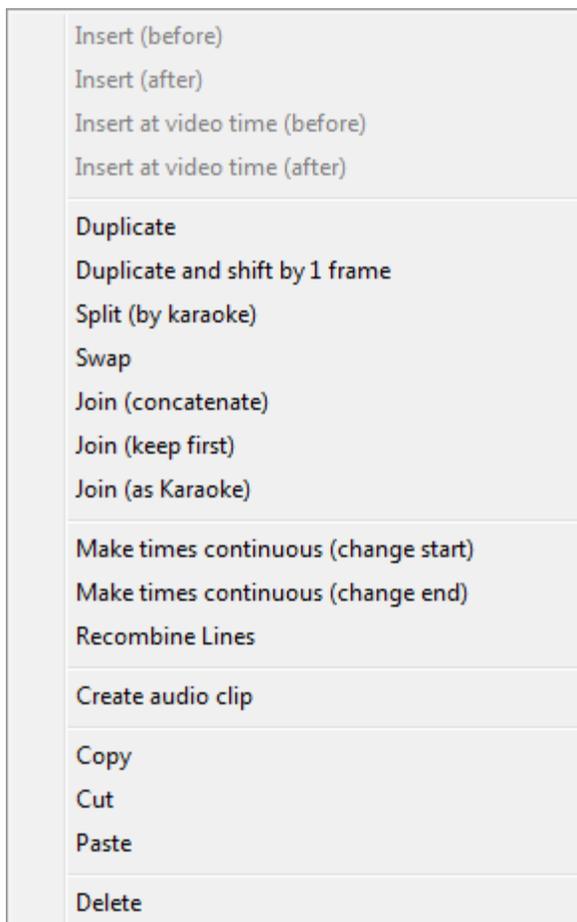
Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Subs_grid_menu.png](#) (289 × 463 pixels, file size: 8 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Editing Subtitles](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Recombine 01.png

- [File](#)
- [File history](#)
- [File links](#)

31	0:02:35.02	0:02:36.01	Style1	It's no use.
32	0:02:37.05	0:02:39.35	Style1	I shouldn't have come here after all.
33	0:02:43.22	0:02:43.66	Style1	To Shinji\NI'll be coming\nto get you,\nso wait for me. \NAttention here please!!
34	0:02:43.66	0:02:45.66	Style1	To Shinji\NI'll be coming\nto get you,\nso wait for me. \NAttention here please!! \NWell, I guess we won't be meeting here.
35	0:02:45.66	0:02:46.02	Style1	Well, I guess we won't be meeting here.
36	0:02:47.16	0:02:49.76	Style1	Can't be helped. I'll go to a shelter.
37	0:03:15.02	0:03:15.42	Style1	Estimated Path

No higher resolution available.

[Recombine_01.png](#) (795 × 119 pixels, file size: 14 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Editing Subtitles](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Recombine 02.png

- [File](#)
- [File history](#)
- [File links](#)

31	0:02:35.02	0:02:36.01	Style1	It's no use.
32	0:02:37.05	0:02:39.35	Style1	I shouldn't have come here after all.
33	0:02:43.22	0:02:45.66	Style1	To Shinji\N'I'll be coming\nto get you,\nso wait for me.\NAttention here please!!
34	0:02:43.66	0:02:46.02	Style1	Well, I guess we won't be meeting here.
35	0:02:47.16	0:02:49.76	Style1	Can't be helped. I'll go to a shelter.
36	0:03:15.02	0:03:15.42	Style1	Estimated Path

No higher resolution available.

[Recombine_02.png](#) (591 × 102 pixels, file size: 9 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#)  for more information)

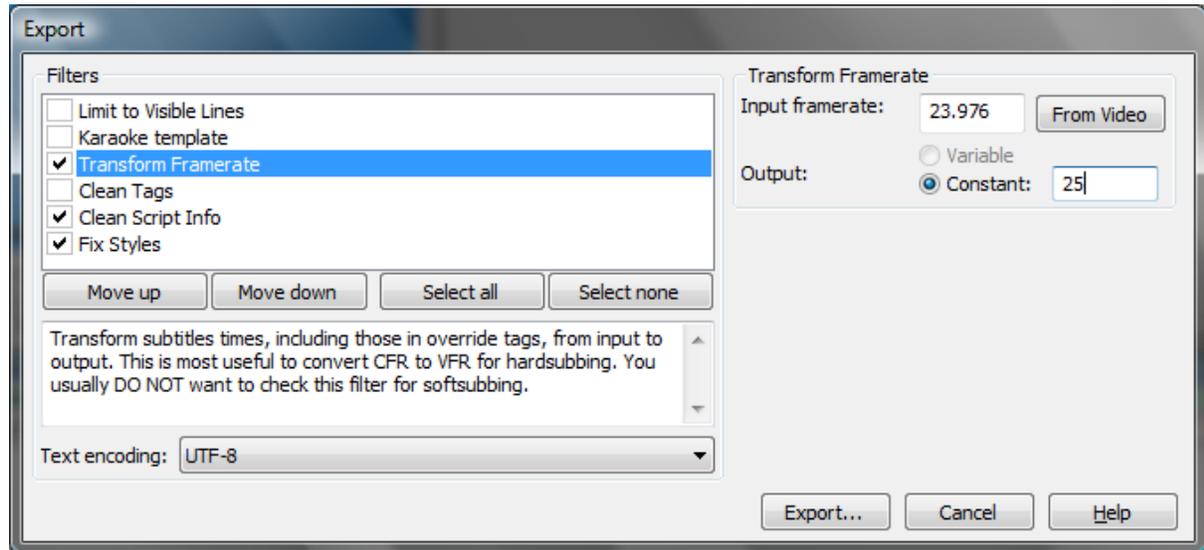
File links

The following page links to this file:

- [Editing Subtitles](#)

File:Export.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Export.png](#) (676 × 310 pixels, file size: 31 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Exporting](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

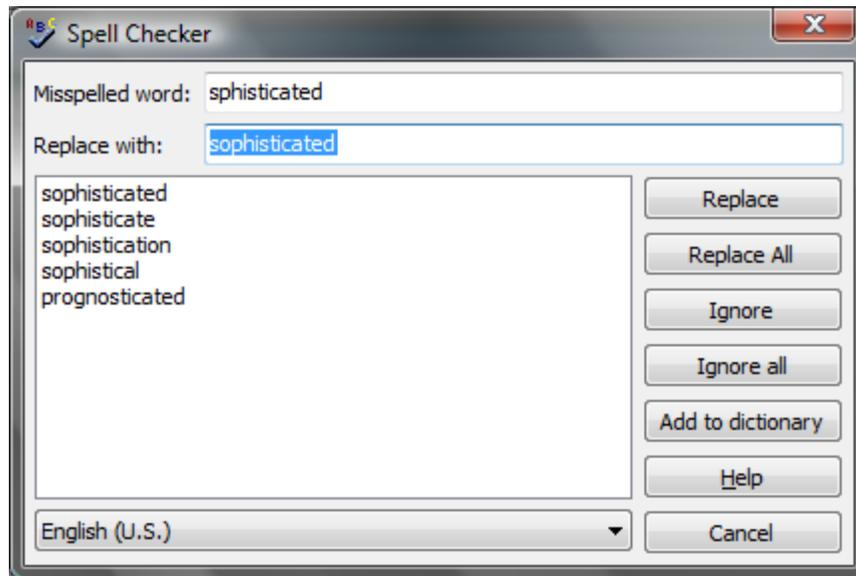
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Spellchecker.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Spellchecker.png](#) (432 × 289 pixels, file size: 25 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Spell Checker](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

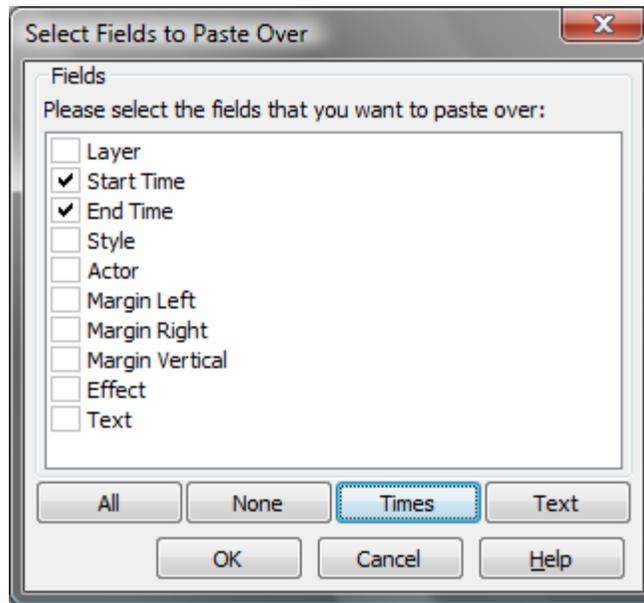
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Paste over.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Paste_over.png](#) (326 × 302 pixels, file size: 16 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Paste Over](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

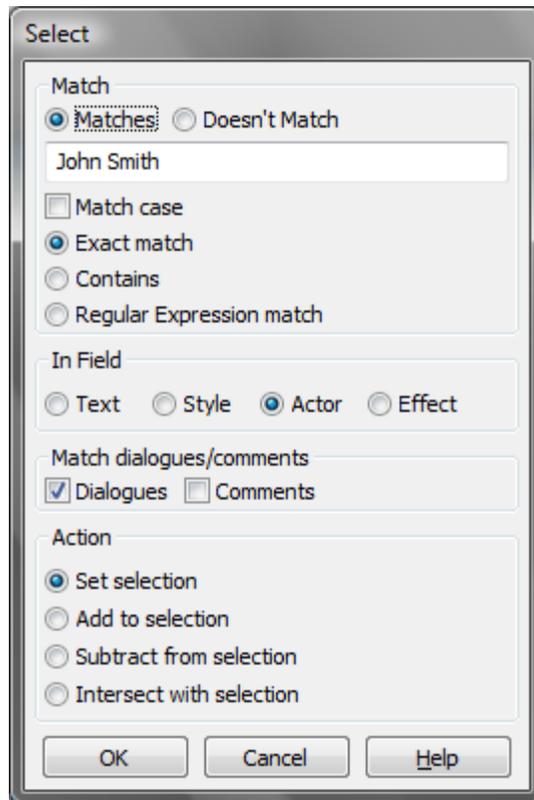
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Select lines.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Select_lines.png](#) (269 × 402 pixels, file size: 19 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Select Lines](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Aegisub path specifiers

Aegisub uses a simple system for denoting file locations. Most pathnames in Aegisub can be written starting with special variables, that each refer to specific locations.

- **?data** - The location where application data are stored. On Windows this is the installation directory (the location of the .exe). On Mac OS X this is inside the application bundle. On other POSIX-like systems this is `$prefix/share/aegisub/`.
- **?user** - The location for user data files, this is configuration files, automatic back-ups and some additional things. On Windows this is `%APPDATA%\Aegisub\`, on Mac OS X this is `$HOME/Library/Application Support/Aegisub/` and on other POSIX-like systems this is `$HOME/.aegisub/`. Enabling the "local configuration" option changes the value of this.
- **?temp** - The system temp directory. Audio cache and any required temporary subtitle files are stored here.
- **?script** - Only defined if a script file is open and saved somewhere, in which case it points to the location of the script.
- **?video** - Only defined if a video file is loaded. Points at the location of the video file. Do note that this is not a good place to save things with dummy video loaded.
- **?audio** - Only defined if an audio file is loaded. Points at the location of the audio file.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

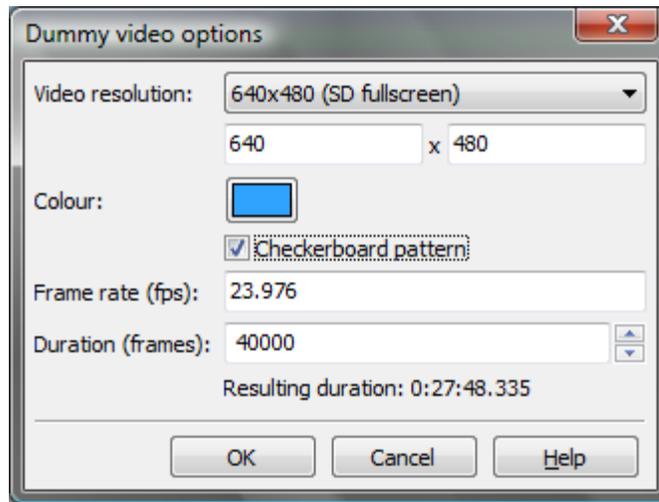
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Dummy video.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Dummy_video.png](#) (333 × 251 pixels, file size: 19 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) 🔗 for more information)

File links

The following page links to this file:

- [Video](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

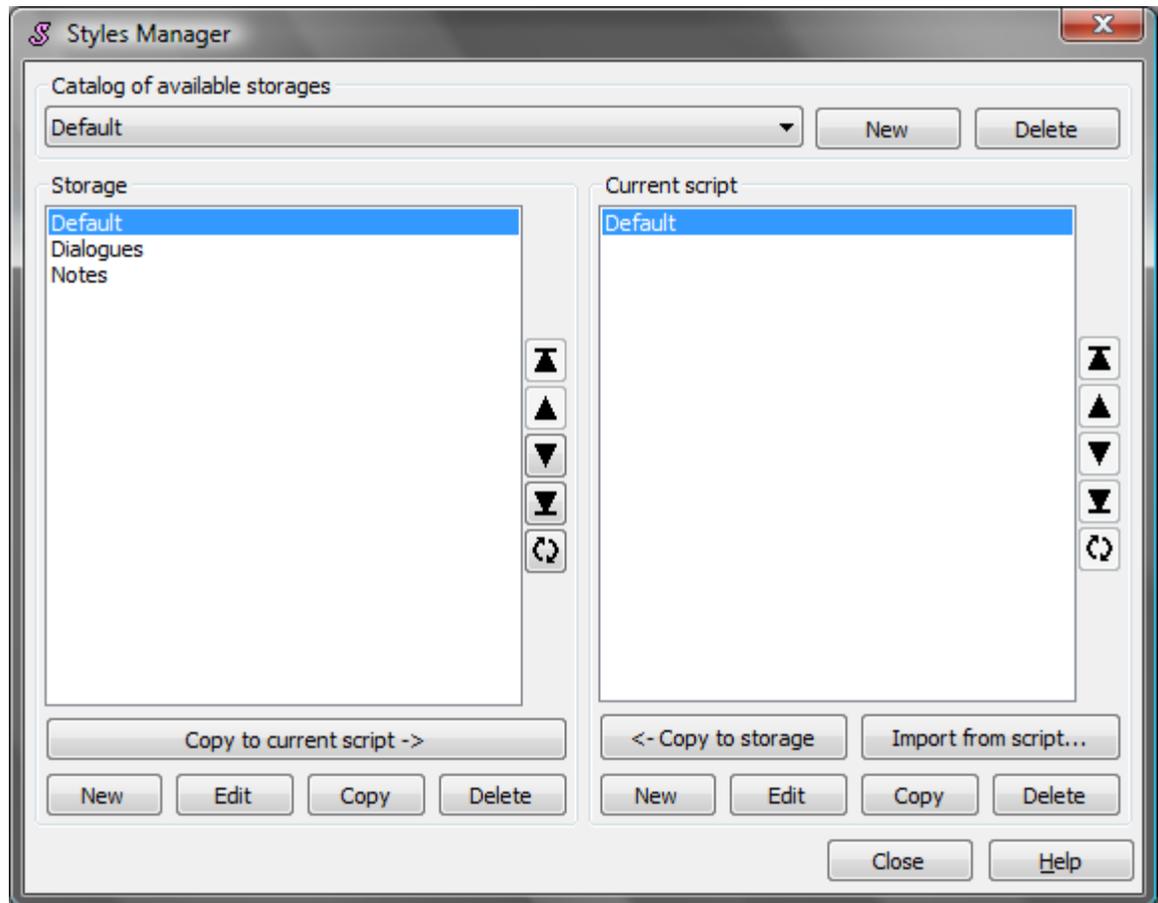
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Style manager.png

- File
- File history
- File links



No higher resolution available.

[Style_manager.png](#) (577 × 454 pixels, file size: 26 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Styles](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

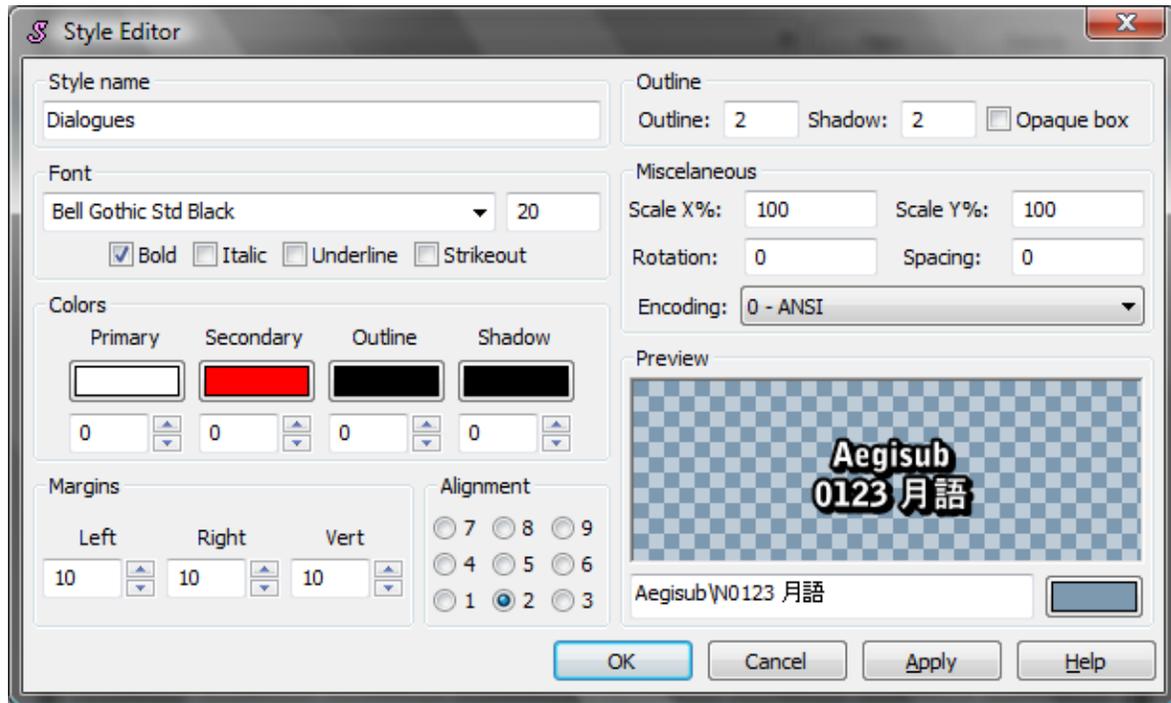
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Style editor.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Style_editor.png](#) (614 × 368 pixels, file size: 40 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Styles](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

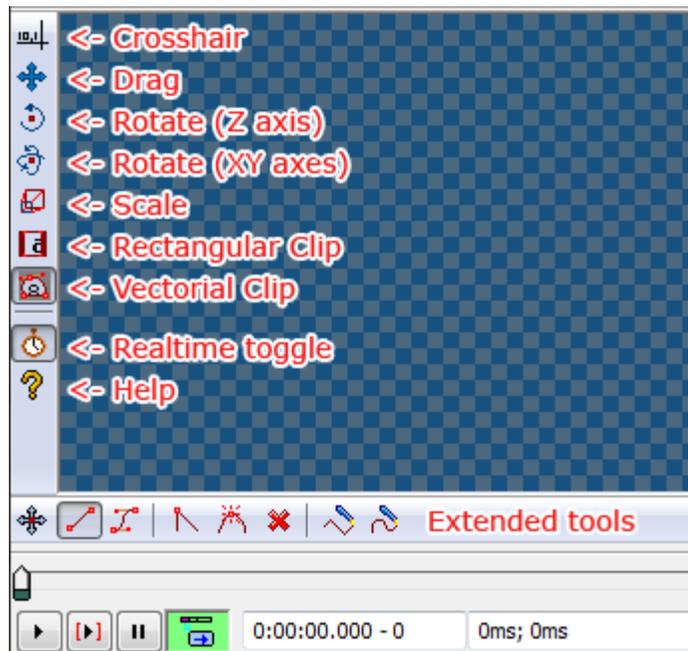
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Visual overview.png

- File
- File history
- File links



No higher resolution available.

[Visual_overview.png](#) (349 × 327 pixels, file size: 28 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Visual Typesetting](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

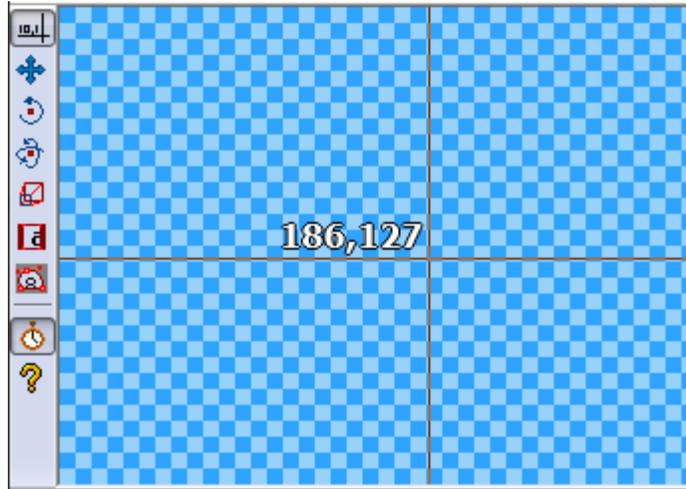
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Visual crosshair.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Visual_crosshair.png](#) (348 × 248 pixels, file size: 5 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Visual Typesetting](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

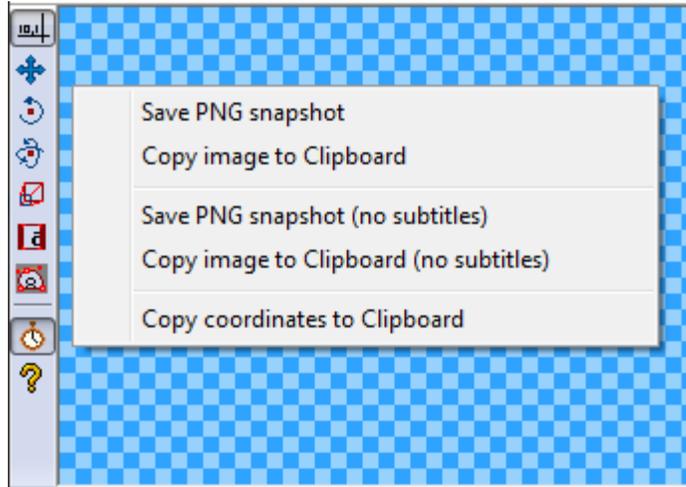
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Visual menu.png

- File
- File history
- File links



No higher resolution available.

Visual_menu.png (347 × 246 pixels, file size: 8 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Visual Typesetting](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

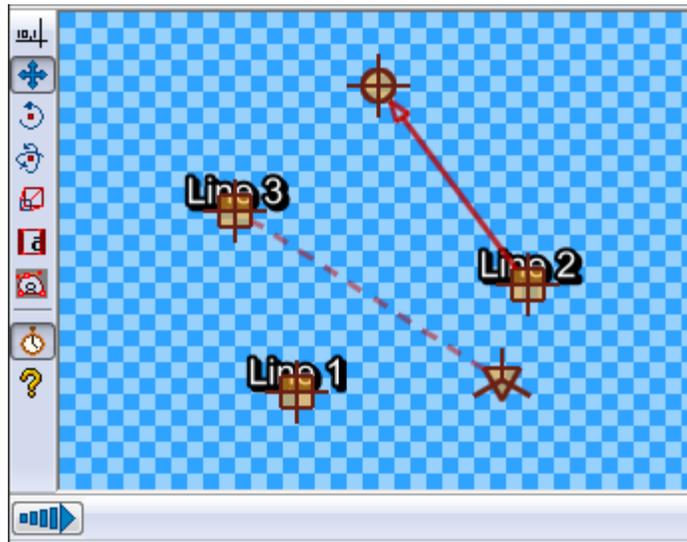
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Visual drag.png

- File
- File history
- File links



No higher resolution available.

[Visual_drag.png](#) (349 × 273 pixels, file size: 13 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Visual Typesetting](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

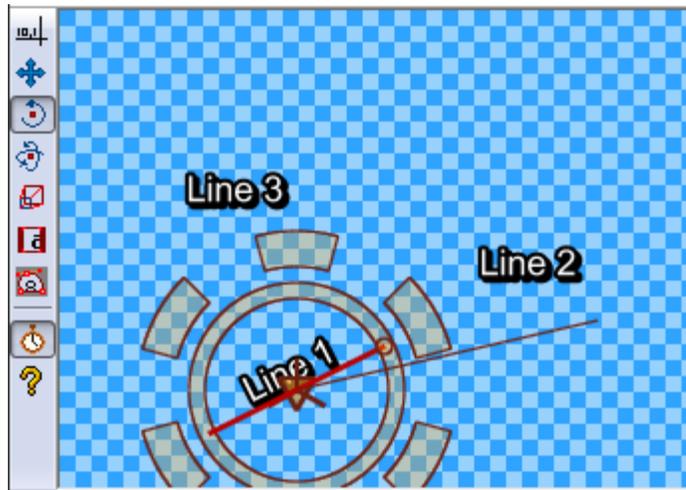
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Visual rotate 1.png

- File
- File history
- File links



No higher resolution available.

[Visual_rotate_1.png](#) (349 × 246 pixels, file size: 22 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Visual Typesetting](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

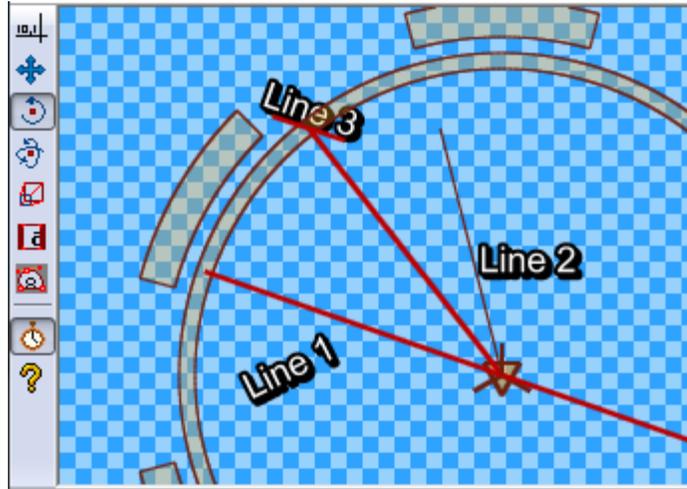
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Visual rotate 2.png

- File
- File history
- File links



No higher resolution available.

[Visual_rotate_2.png](#) (349 × 246 pixels, file size: 27 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Visual Typesetting](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

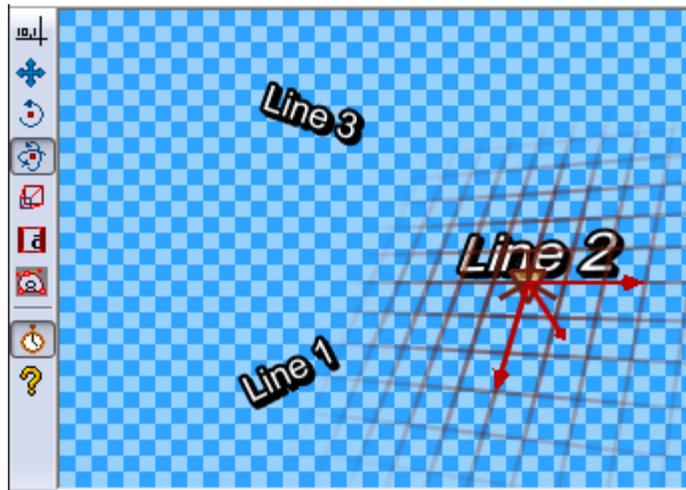
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Visual rotate xy.png

- File
- File history
- File links



No higher resolution available.

[Visual_rotate_xy.png](#) (348 × 247 pixels, file size: 37 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Visual Typesetting](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

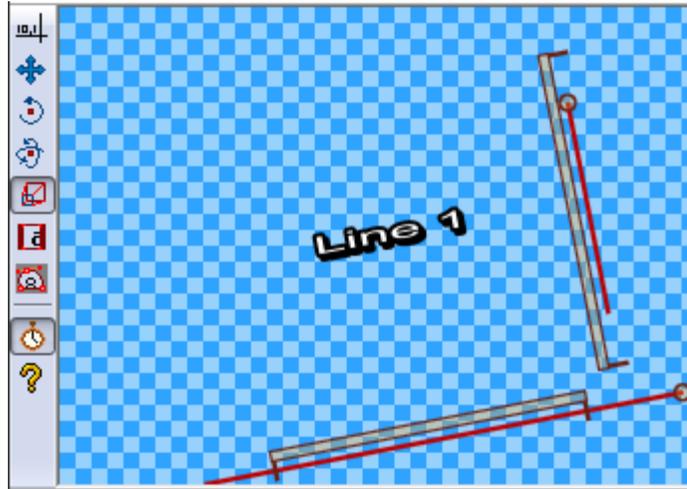
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Visual scale.png

- File
- File history
- File links



No higher resolution available.

[Visual_scale.png](#) (349 × 246 pixels, file size: 13 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Visual Typesetting](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Visual clip.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Visual_clip.png](#) (347 × 245 pixels, file size: 26 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Visual Typesetting](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

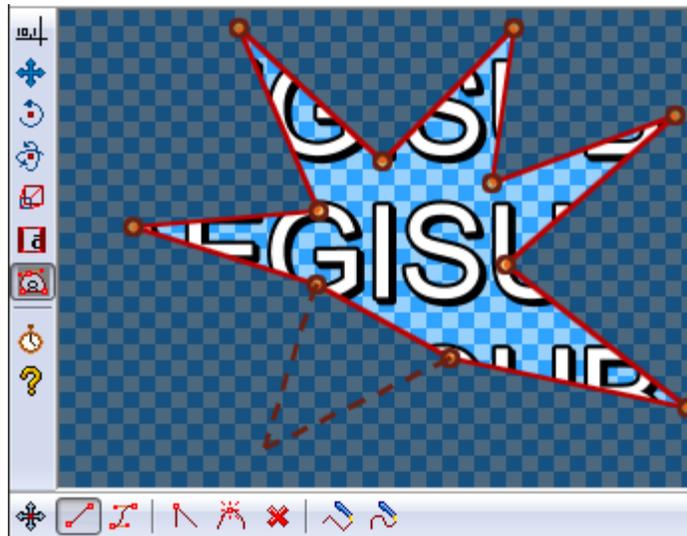
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Visual vector clip.png

- File
- File history
- File links



No higher resolution available.

[Visual_vector_clip.png](#) (349 × 270 pixels, file size: 29 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Visual Typesetting](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Visual vector toolbar.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Visual_vector_toolbar.png](#) (206 × 40 pixels, file size: 7 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) ↗ for more information)

File links

The following page links to this file:

- [Visual Typesetting](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Fr sample01 .jpg

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Fr_sample01.jpg](#) (352 × 240 pixels, file size: 6 KB, MIME type: image/jpeg)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [ASS Tags](#)

File:Fr sample02.jpg

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Fr_sample02.jpg](#) (352 × 240 pixels, file size: 6 KB, MIME type: image/jpeg)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [ASS Tags](#)

File:Fr sample03.jpg

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Fr_sample03.jpg](#) (352 × 240 pixels, file size: 6 KB, MIME type: image/jpeg)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [ASS Tags](#)

File:Pick-color-toolbar-buttons.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Pick-color-toolbar-buttons.png](#) (119 × 22 pixels, file size: 494 B, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [ASS Tags](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Pos sample01.jpg

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Pos_sample01.jpg](#) (480 × 360 pixels, file size: 28 KB, MIME type: image/jpeg)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [for more information](#))

File links

The following page links to this file:

- [ASS Tags](#)

File:Pos sample02.jpg

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Pos_sample02.jpg](#) (480 × 360 pixels, file size: 28 KB, MIME type: image/jpeg)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [ASS Tags](#)

File:Pos sample03.jpg

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Pos_sample03.jpg](#) (480 × 360 pixels, file size: 28 KB, MIME type: image/jpeg)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [for more information](#))

File links

The following page links to this file:

- [ASS Tags](#)

File:Clip sample01.jpg

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Clip_sample01.jpg](#) (528 × 360 pixels, file size: 23 KB, MIME type: image/jpeg)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

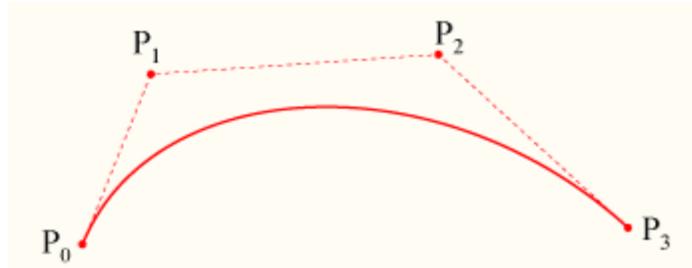
File links

The following page links to this file:

- [ASS Tags](#)

File:Bezier.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Bezier.png](#) (350 × 136 pixels, file size: 5 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [ASS Tags](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

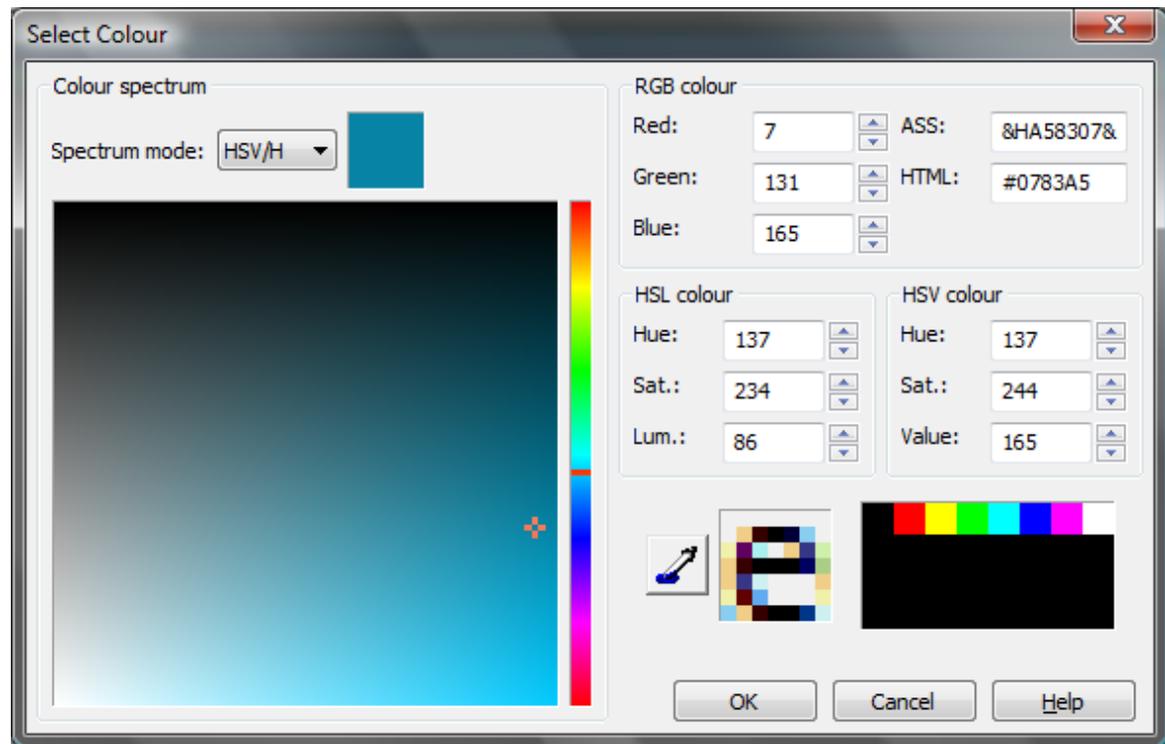
» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

File:Colour picker.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Colour_picker.png](#) (589 × 377 pixels, file size: 38 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

([Latest](#) | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

([Latest](#) | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [for more information](#))

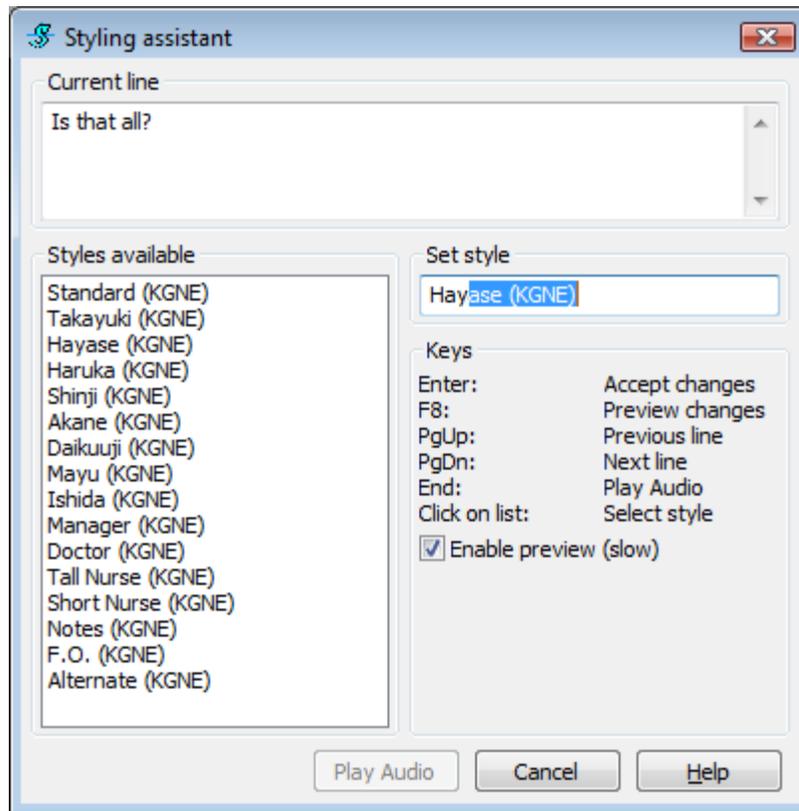
File links

The following page links to this file:

- [Colour Picker](#)

File:Styling assistant.png

- File
- File history
- File links



No higher resolution available.

[Styling_assistant.png](#) (403 × 407 pixels, file size: 18 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Styling Assistant](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Styling autocomplete.png

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

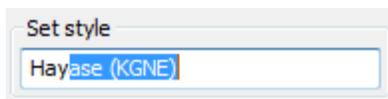
Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Styling_autocomplete.png](#) (195 × 49 pixels, file size: 2 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

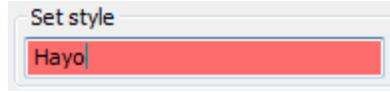
File links

The following page links to this file:

- [Styling Assistant](#)

File:Styling invalid.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Styling_invalid.png](#) (196 × 45 pixels, file size: 1 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Styling Assistant](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

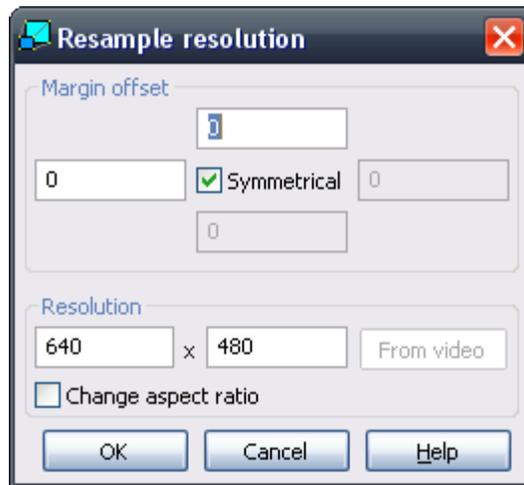
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Dialog resample.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Dialog_resample.png](#) (264 × 243 pixels, file size: 7 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Resolution Resampler](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Resample diagram.png

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Resample_diagram.png](#) (400 × 300 pixels, file size: 9 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

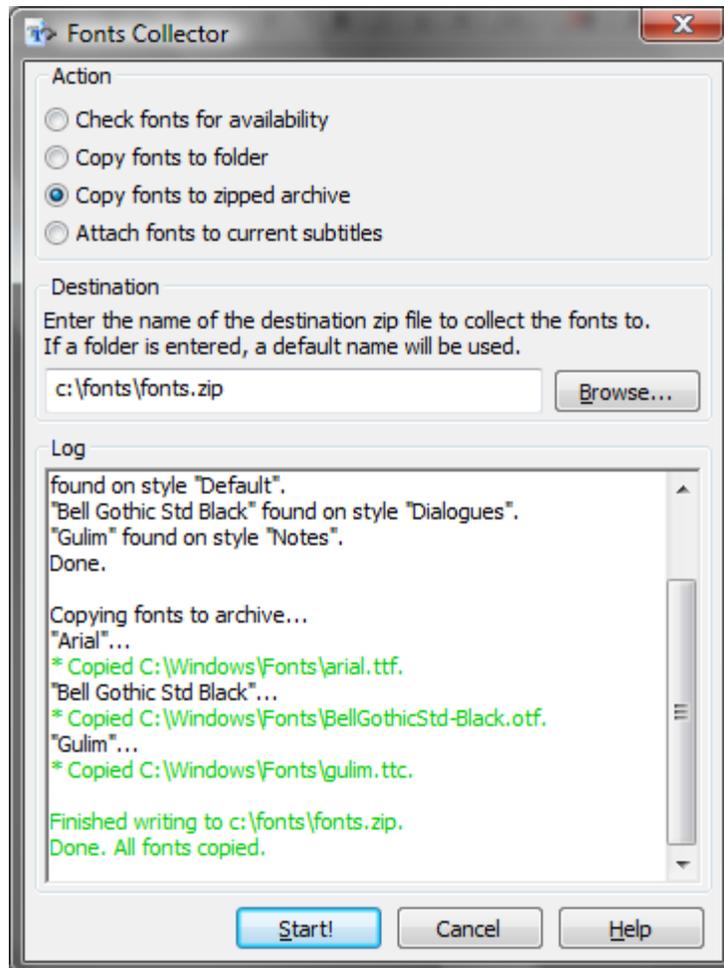
File links

The following page links to this file:

- [Resolution Resampler](#)

File:Fonts collector.png

- File
- File history
- File links



No higher resolution available.

Fonts_collector.png (366 × 488 pixels, file size: 30 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

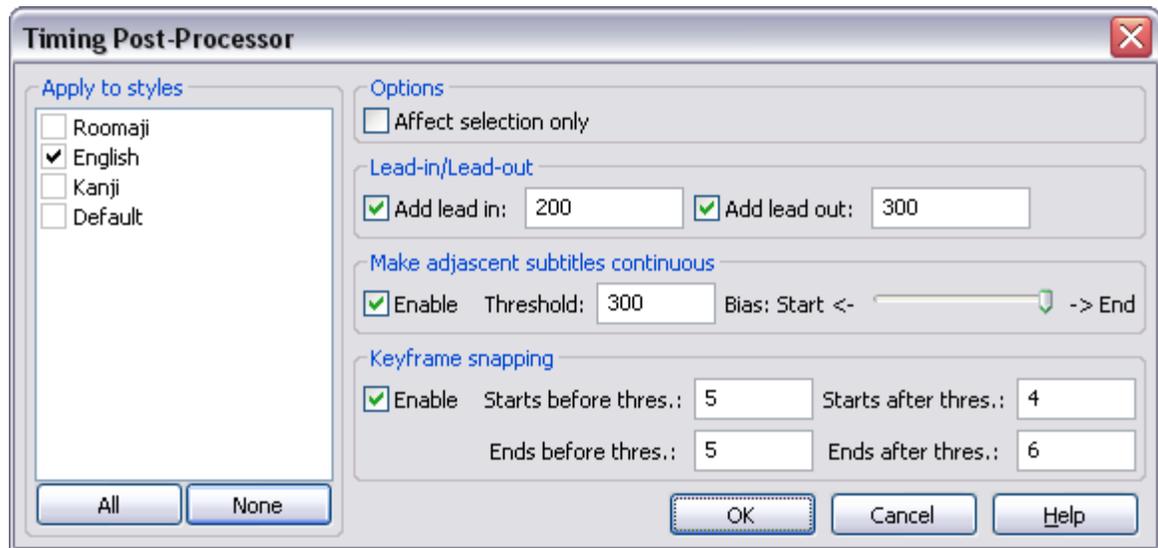
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Dialog timing processor.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Dialog_timing_processor.png](#) (579 × 275 pixels, file size: 11 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Timing Post-Processor](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

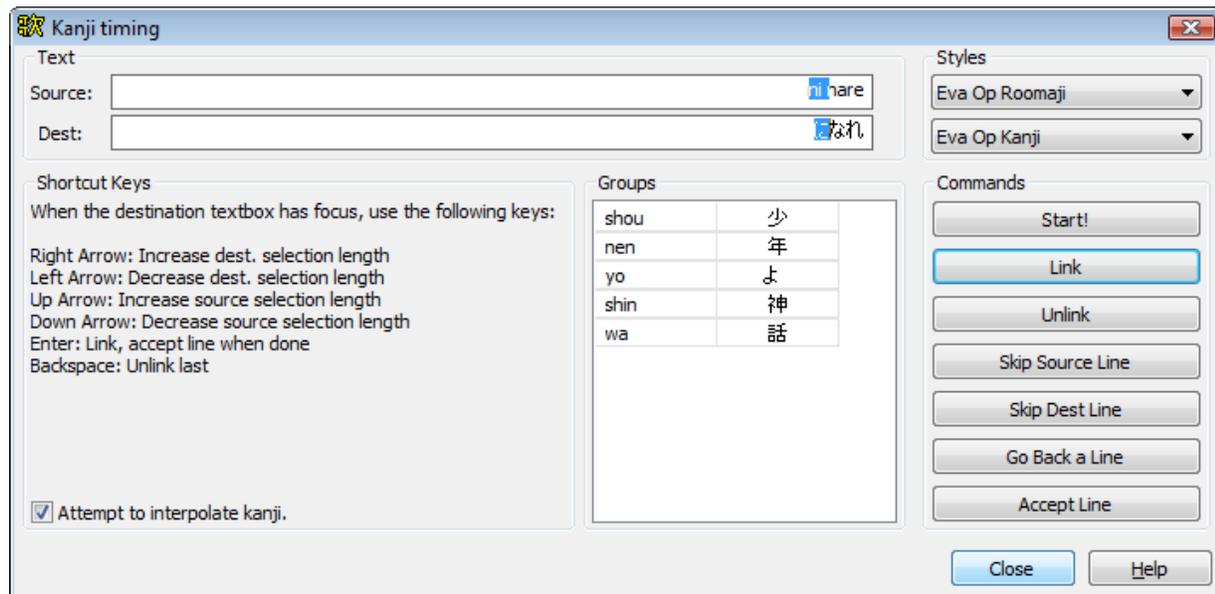
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Kanji timer.png

- File
- File history
- File links



No higher resolution available.

[Kanji_timer.png](#) (717 × 351 pixels, file size: 25 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Kanji Timer](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Sidebar:History of Automation

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

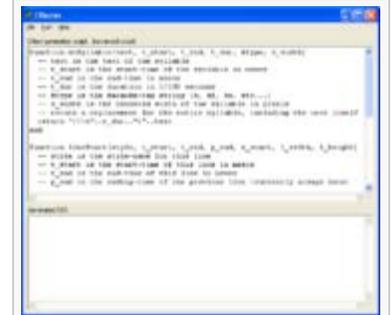
The Automation system in Aegisub has evolved from a very basic and hardly useful stand-alone system for creating basic karaoke effects into a very powerful extension mechanism. Here's a short account of its history, from the principal author of the systems.

Automation 1, really just Karaoke Effector

The *Karaoke Effector* program was originally created for making karaoke effects for a small translation project, that later died. It was written in Borland Delphi and used Lua 5.0 for scripting. This use of Lua is the primary reason I even call it "Automation 1" today. It basically allowed creating effects similar to what the *simple-k-replacer* script from Automation 3 does, but in a more complicated and not as usable way.

If you want to try this relic from the museum, you can download it here:

<http://www.animereactor.dk/aegisub/effector.rar> 



The main window of Effector.

Automation 2, the Python engine that never happened

Automation 2 was intended to be the scripting system in Aegisub, I drafted the specifications for it while Aegisub was still only in internal pre-alpha development. It was planned to use Python for scripting language and be quite flexible. It just proved to be an overall bad design (which on hindsight might have been a good thing) and it was never implemented. Instead I started looking into Lua again and drafting what became Automation 3.

I don't think there's much of any proof of the work on Automation 2 left now. The main thing to be said about it is that the current Automation 4 achieves everything Automation 2 was intended to be, and even more.

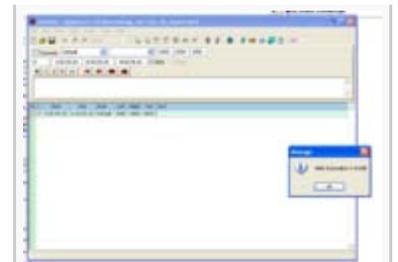
Automation 3, back to Lua and something usable

After the fiasco of Automation 2 and Python I went "browsing languages" again and ended up going back to Lua, and also went for a much less ambitious design. This proved to work and ended up as the Automation 3 system. Originally Automation 3 was also intended to be somewhat extensible and down the road support more than just the basic modification of subtitle lines, this is evident from the *kind="basic_ass"* statement required in all Automation 3 scripts. Unfortunately the overall architecture of Automation 3 in the end didn't allow it to be extended either way, and the first rough ideas for Automation 4 started forming.

In the end Automation 3 did prove very successful though and has done a great job.

Automation 4, unlimited feature works?

As the flaws of Automation 3 started showing through the design of Automation 4 begun. Lots of people had been crying for other languages, especially Perl and Python, so support for multiple scripting languages went into the core design. The development of Automation 4 was much on and off, standing still for months at a time. Originally it was planned for Aegisub 1.09, then got pushed to 1.10 and then finally to 1.11, which then grew into Aegisub 2 because of the load of new features and major redesigns in that version, Automation 4 one of them.



Automation 4 first "working", dated 2006-05-16.

In mid-May 2006 Automation 4/Lua was finally in a "working" state, if I am to trust the timestamp of *hello-auto4.png* but first now, more than a year later, it's really useful. A testament to my laziness.

- Niels Martin Hansen, July 2nd, 2007

Glossary:Macro

In computer programs, a **macro** is a (usually relatively small) snippet of program code that can easily be added to the program, to add a new function or such, intended to automate some aspect of using the program.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Using the included macros

Aegisub includes several macros. Here's a line-up of them.

Contents [\[hide\]](#)

- 1 Apply karaoke template
- 2 Make full-width
- 3 Automatic karaoke lead-in
- 4 Clean tags

Apply karaoke template

This is the macro incantation of Karaoke Templater. See the page on [Karaoke Templater](#) for how to use this.

This macro is only available when there is at least one template line in the subtitle file.

Make full-width

Converts all ASCII characters to Japanese "full-width" variations of them.

This can be useful if you need to typeset a sign translation vertically, such that the letters are "stacked".

This macro modifies all lines that are currently selected in the subtitles grid.

Example

Here's a typeset sign:

```
{\fn@DFPGothic-  
EB\fs26\shad0\fe128\bord3\3c&H25485A&\c&HDEEBF1&\pos(456,184)\frz-  
90}Sign text
```

Notice that it uses an "@-font", a variation that exists of every CJK font that has "fullwidth" characters rotated 90 degrees from the baseline. Fullwidth characters include not just these fullwidth variations of the Latin alphabet but also Japanese kana and kanji, hanzi, hanja and various punctuation characters.

Now after running this macro on the line:

```
{\fn@DFPGothic-  
EB\fs26\shad0\fe128\bord3\3c&H25485A&\c&HDEEBF1&\pos(456,184)\frz-90}Sig  
n text
```

This is what it looks before and after running the macro:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)



Automatic karaoke lead-in

Automatically join several karaoke-timed lines up timing-wise and add appropriate `\k` tags in front of them.

This macro is designed to help creating karaoke effects, most importantly creating transitions and lead-ins for lines. It's well suited for using when the karaoke is timed but before applying effects, such as karaoke templates.

This macro requires at least two lines to be selected and it only works sensibly if the start-time of each selected line is larger than the start-time of the selected line that comes before it. It changes the timing of the selected lines and adds `\k` tags at the start of them except the first.

Example

Here's two lines of "tightly" timed karaoke:

```
Dialogue:
0,0:00:44.46,0:00:46.28,Default,,0000,0000,0000,,{\k15}Ne{\k14}ver
{\k14}gon{\k13}na {\k37}give {\k40}you {\k49}up
Dialogue:
0,0:00:46.57,0:00:48.56,Default,,0000,0000,0000,,{\k13}Ne{\k13}ver
{\k13}gon{\k13}na {\k36}let {\k46}you {\k65}down
```

Both lines start exactly when the first word starts being sung, and they end exactly when the last word ends.

Now if the *Automatic karaoke lead-in* macro is run on these two lines, they are changed into this:

```
Dialogue:
0,0:00:44.46,0:00:46.28,Default,,0000,0000,0000,,{\k15}Ne{\k14}ver
{\k14}gon{\k13}na {\k37}give {\k40}you {\k49}up
Dialogue:
0,0:00:46.28,0:00:48.56,Default,,0000,0000,0000,,{\k29}{\k13}Ne{\k13}ve
r {\k13}gon{\k13}na {\k36}let {\k46}you {\k65}down
```

The start-time of the second line is changed so it matches the end-time of

the first line, and a `\k` tag is added to the start of the line, to make up for the shift otherwise created by this. This effectively creates an empty syllable that can be used as a "spacer" to create fade-in and fade-out effects.

The macro also shows this message:

```
Smallest inter-line duration: 290 milliseconds
```

This simply says that the smallest duration between two lines it found, was 290 milliseconds, or 0.29 seconds, so that's as much time you have to make fade-in, fade-out and other transition effects, if you want every syllable-highlight to be fully visible.

Clean tags

This macro does various cleaning up on the override tags in all selected lines.

- Combines adjacent override blocks (i.e. { ... }) except if both of the block contains `\k` tags each then they will be left as is
- Push any `\k` tags in override blocks to the front (e.g. from `{\frz90\k40}` to `{\k40\frz90}`). Special care will be taken for multiple `\k` tags within one block to preserve the ordering
- Move line-wide tags (i.e. tags whose effects affect the whole line -- `\a \an \org \pos \move \fade \fad`) to the beginning of the lines
- Remove all but first line-wide tags of the same class (note: `\pos` and `\move` are from the same class -- only first of them works in a line, therefore the script will find the first `\move` or `\pos` and retain which of the two comes first and remove others. The same are done for `\fad` and `\fade`)
- Remove spaces in comma-separated parameters (e.g. `\pos(200 , 200)` becomes `\pos(200,200)`)

This macro is also available as an export filter.

The main intended function of this macro is to make [karaskel.lua](#) split karaoke lines more sensibly into syllable structures, see the example.

This macro modifies all selected lines in the grid, re-writing all tag blocks in them.

Example

Original line:

```
{\r\frz90\k80}Test {\r\fry180\k60}me
```

Karaskel creates these syllable structures:

- 0 = `{\r\frz90}`
- 1 = `Test {\r\fry180}`
- 2 = `me`

After running *Clean Tags* on the line:

```
{\k80\r\frz90}Test {\k60\r\fry180}me
```

Now karaskel creates these syllable structures:

- 0 =
- 1 = `{\r\frz90}Test`
- 2 = `{\r\fry180}me`

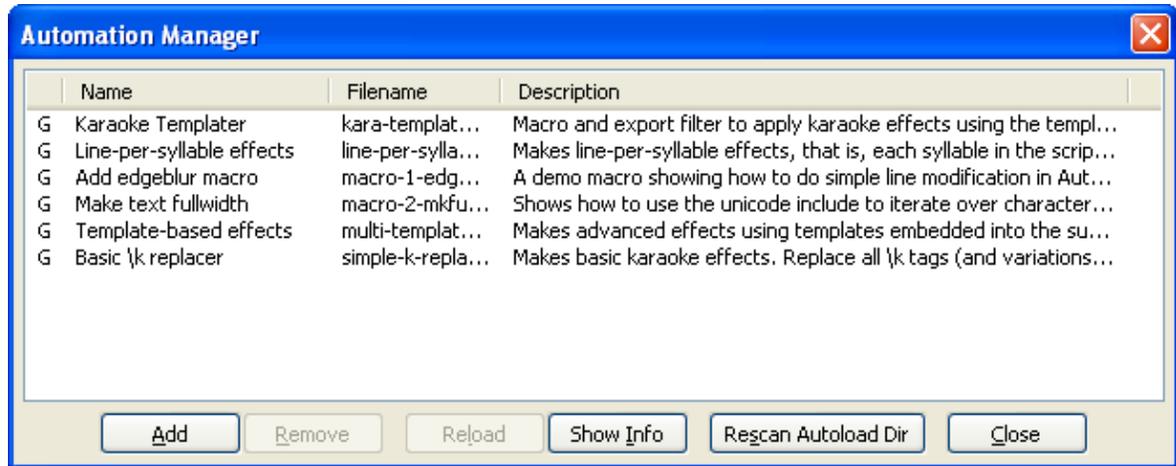
The cleaned up version is generally what you'd want since it places the override tags inside the syllables they affect.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Automation Manager

The Automation Manager window is used to view, load and unload [Automation](#) scripts.

The Automation Manager window is opened from the *Automation* menu or with the  toolbar button.



The script list

The main area of the window is a list of all *script files* loaded. It's important to remember that one script file can contain several [features](#), for example one script file can define two [macros](#) and one export filter.

Scripts can be loaded in one of two ways. In the screenshot above, all the scripts are *global* (autoloaded), this can be seen from the "G" in the leftmost column. Global scripts are automatically loaded along with Aegisub. They are put in one of the Automation autoload folders. You can not unload global scripts, you can however remove them from the autoload folder.

The other type of scripts are *local* scripts, these are scripts loaded by you. Each subtitle file has a list of local scripts stored in it, when you save your subtitle file the list of loaded scripts is stored along with it, and all the scripts are loaded when you then open that subtitle file again. Local scripts are shown with an "L" in the leftmost column.

Sometimes a script is shown in red in the list. This only happens if the script could not be loaded for some reason. The reason will usually be shown in the Description column. If the description of the error is too long to read, you can select the script and click the Show Info button to see it all. Scripts failing to load should only happen if you are writing your own scripts and manage to make a programming error.

The buttons

There are 6 buttons at the bottom of the Automation Manager window:

- The **Add** button is used to load a local script.
- The **Remove** button is used to unload a local script. It is only available when a local script is selected.
- The **Reload** button unloads and reloads the selected script file from disk. You can use this to reload scripts you're developing, but also see below for other ways to do this.
- The **Show Info** buttons shows detailed information on the selected script, as well as the entire Automation system.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- The **Rescan Autoload Dir** button scans the autoload folders to see if any scripts have been added or removed since Aegisub was started. All new scripts in the autoload folders are then loaded, scripts that have been removed are unloaded and all other global scripts are reloaded.
- The **Close** button closes the Automation Manager window.

Other ways to reload scripts

If you're developing scripts you might find yourself returning to the Automation Manager often to reload your script. There are also faster ways to reload scripts though:

- When the **Export** dialog is opened all local scripts are reloaded. You can change this in the Options dialog.
- Hold the **Ctrl** key and click the Automation toolbar button to rescan the autoload folders.
- Hold both the **Ctrl** and **Shift** keys and click the Automation toolbar button to reload all scripts, also rescanning the autoload folders.

The Automation Manager won't open when any of these methods are used, but you will get an error message window if a script failed to load.

For users of Aegisub 1.10 and earlier

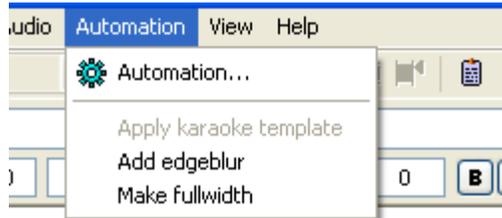
You may wonder where several of the functions that were previously in the Automation Manager have gone.

- The **Create** and **Edit** buttons were removed since they never worked fully as intended. You're encouraged to instead use your favourite file manager or editor to find and open the scripts you want to edit. Creating new scripts is also a bit easier since the formal requirements for Automation 4 scripts are much looser than for Automation 3 scripts. Another otherwise commonly used method to create new scripts is just to copy an existing script and edit that instead; the Create button wouldn't be much help there either way.
- The **Apply now** button was removed for two reasons. The first and most important is that it doesn't fit into the Automation 4 design at all. In Automation 3 every script file only did one thing so it was obvious what it meant to "apply" it. This isn't the case with Automation 4, one script file can have several functions (features), so which of them should be applied? The other, and rather minor reason, is that burying the button inside a dialog is just not a good place, you will have to cancel out of the Automation Manager window to undo the effect of the script if you decided you weren't satisfied, just to immediately return to it, reload the script and Apply again. The Automation 4 solution to this is to create a macro instead. It's available directly from the Automation menu and you can easily reload using one of the methods listed above.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Running Automation macros

Automation macros appear as options in the Automation menu.



The first option in the Automation menu, before the separator, always opens the [Automation Manager](#) window.

Below the separator is a list of all macros loaded. In the example above three macros are loaded. The first, [Karaoke Templater](#) is unavailable though, because the macro has determined it can't be run on the subtitles. In this case, because the subtitle file has no karaoke templates. Other macros might be unavailable for other reasons.

To run a macro, just select it from the Automation menu. Some macros might ask you for more information, and some work only on the selected subtitle lines. Always remember to read the documentation that came with macros you downloaded from the Internet.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Moving from Automation 3

If you're an avid user of Automation 3 (Lua) scripting from Aegisub 1.10 and earlier you will be happy to know that Aegisub 2 still supports Automation 3 scripts with no or only very small changes to them.

Things to look out for

Karaoke tables

The karaoke tables are no longer generated by Aegisub but in `karaskel`. If you're using `karaskel` already you shouldn't need to worry about this.

If you aren't using `karaskel`, here's how to get the karaoke tables:

First you will need the `karaskel-base`. Put this line as some of the first in your script:

```
include("karaskel-base.auto3")
```

You will then need to call a function in `process_lines`:

```
karaskel.parse_syllable_data(meta, styles, lines)
```

That line should be the very first thing you do in `process_lines`. It will modify `lines` so it also includes the karaoke data.

colorstring_to_rgb function

This should also not be a problem if you're using the include files as it has been recommended to do.

The `aegisub.colorstring_to_rgb` function was moved from being an internal function implemented in C++ to being in the `utils` include. If you're using `karaskel`, you also have `utils` included. Otherwise you can include it manually like this:

```
include("utils.auto3")
```

New script file name extension

As shown above, the `.lua` extension is no longer officially used for Automation 3 scripts. It should continue to work in most cases, however. Specifically, all the include files still also exist with `.lua` extension. These will detect whether you're using an Automation 3 or Automation 4 script and include the correct version of the include file for you.

It can in some cases be problematic to have a main script called `.lua` since those will first be loaded with the Automation 4 Lua 5.1 engine, which means that the Automation 4 includes will also be used at first. Only after the script has been loaded it's possible to test whether it's actually an Automation 3 script. If a `.lua` file is found to be an Automation 3 script after being loaded as an Automation 4 one, it's reloaded as an Automation 3 one instead.

Contents [hide]

- 1 Things to look out for
 - 1.1 Karaoke tables
 - 1.2 `colorstring_to_rgb` function
 - 1.3 New script file name extension
- 2 Good news
- 3 Running Automation 3 scripts
- 4 Why use Automation 4 instead?

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

If you find that this auto-detection fails to work for your script and it instead fails to load, or is just left loaded as a (non-functioning) Automation 4 script, manually rename the file to `.auto3`.

Good news

The `text_extents` function now uses the Automation 4 version of the calculation, which should be 100% VSFilter compatible in all cases. This means all weird spacing problems should be resolved. (There might still be some half-pixel-off errors due to rounding.)

Running Automation 3 scripts

With the Apply button gone from the [Automation Manager](#) you might be wondering how to run Automation 3 scripts.

You use them from the [Export dialog](#). (Automation 3 scripts have always first and foremost been export filters.) You do *not* have to go to the Automation Manager window every time to reload a script, nor do you have to use the old "self-include-reload" trick. All local Automation scripts are automatically reloaded whenever you open the Export dialog.

Why use Automation 4 instead?

Since Automation 3 still works you might be tempted to continue using it. Here are some reasons not to:

- Simpler script file structure. There are less required elements and you are somewhat more free in how to express things.
- Several array indexing inconsistencies have been corrected, everything is now indexed from one as the custom in Lua programming is. The exception to this rule is for karaoke tables, these have retained the "zero'th" syllable, everything before the first karaoke tag.
- The pre-calculations performed by [karaskel](#) provide more useful information and more advanced position. It's also more modular, allowing you to control what is pre-calculated and when.
- You get direct access to the `Script Info` and `Styles` sections of the subtitle file, allowing you to modify styles and header information, also adding and removing it.
- The ability to write macros. Macros are easier to access than the old semi-workaround that was the Apply button, and they have access to information about the lines selected in the subtitle grid.

But of course you shouldn't start rewriting all your existing scripts to Automation 4 unless you have a really good reason.

If you decide to write scripts for Automation 4 Lua you can read the [overview of changes from Automation 3](#) page for a jumpstart.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater	Declaring templates • Execution order • Modifiers • Inline-variables (\$-

reference:	variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Overview of changes from Automation 3

To put it short, the changes from Automation 3 to Automation 4 are huge, almost everything has been completely redesigned and rewritten. You should however find that many things still work the same, and the rest work better, hopefully. Automation 3 scripts can still run with little or no modifications, see the [moving from Automation 3](#) page for details on that.

Architectural changes

These are the changes in how the scripting engine(s) integrate with Aegisub, ie. how scripts are loaded and managed and not as much what happens when they are running.

Actually, what is called Automation 4 is only the architecture described here. When you use Automation 4 to run scripts written in Lua (and are not Automation 3 scripts) it's called Automation 4 Lua.

The most obvious change is the support for multiple scripting engines. Automation 3 was built around only supporting Lua, Automation 4 has been designed to support scripts written in all kinds of languages as long as a scripting engine has been implemented into it. Implementing a new scripting engine consists of implementing about 5-8 C++ classes and quite possibly a bunch more helpers. The basic implementation should be a few day's work and then an uncountable number of hours for documentation, standard includes and support. If you are interested in implementing support for a new scripting language please contact the Aegisub team.

Automation 4 now supports "plugging in" script functionality in several different places. Each different kind of functionality a script can support is called a "feature". Currently this really only means the export filters supported by Automation 3 and then macros. Export filters appear in the Export dialogue and are intended for large batch-operations. Macros appear in the Automation menu and are intended for smaller tasks. A third, still not fully drafted feature, is support for scripting file format reading and writing, to be able to import and export all the most exotic subtitle file formats in existence.

Macro features can retrieve the current selection in the subtitle grid, modify the subtitles in-line and change the selection upon return. They can also set one or more Undo points. Last but not least, they can show dialogue boxes to get more information from the user.

Finally, one script file can provide more than one feature, also more than one of each kind of supported feature. A script file can also provide no features at all, though that's not very useful.

The Lua scripting engine

The probably single biggest and most important change from Automation 3 to Automation 4 Lua is that the Lua version has been upgraded from 5.0 to 5.1. While there are a large number of changes between the two versions, the perhaps most important for writing Automation 4 Lua scripts is the modulo operator that has finally been introduced into the language. You can see more details at [the Lua homepage](#) .

The rest of the changes to Lua scripting interface can best be summed up as: Give as much control as possible to the script and don't do any unrequested data generation.

In more detail, these are the main points of the changes from Automation 3 to

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Automation 4 Lua. They are discussed in more detail below.

- No more a fixed entry point, script entry points are instead defined with `register_xxx` functions
- Direct manipulation of the subtitle file rather than manipulating a large Lua table
- Much more direct control over the entire execution process
- Various changes in the data structures used
- More complete Unicode support in include files
- More advanced layout calculations in karaskel include file

You no longer need to name your functions anything specific, rather you tell Aegisub what your entry point functions are called. What you do is register each feature you want, using the `aegisub.register_macro` and `aegisub.register_filter` functions. (Exception: If you use the [karaoke skeleton functions](#) you do need to follow some specific naming schemes and should not register any features.)

Instead of getting a huge table with all lines in the subtitle file, Automation 4 Lua provides your script with a "user data object" you can use to get, save and replace lines in the subtitle file with. (This object is usually called the *subs object*.) Using this subs object you can not only get the dialogue lines in the subtitle file, but also manipulate the style lines and the Script Info section. On the other hand, you are no longer given a *styles* or *meta* table directly, but the karaskel include has [a function](#) to extract that information.

As a result of the direct manipulation of the subtitle file you can now do, you also have much more control over what happens when during script execution. Automation 3 always split the script execution up in three phases: Generate input, run processing, read output. Using the subs object you can instead intermix those three phases freely, or just do as usual. This is especially convenient for macros, since they only need to touch the lines they actually modify, rather than needing to import the entire file and having it re-read by Aegisub when done. The skeletons provided by `karaskel.lua` can to an extent automate managing the execution process.

The data structures used have also been changed to better support the more complete subtitle file representation and to better reflect the philosophy of doing as little as possible before requested. For example, the karaoke data tables that were an inherent part of the line data structure in Automation 3 are no longer automatically generated, instead you use a separate function to have them generated. If you use karaskel, you will however not need to worry about that. A couple of new fields have also been introduced to support the changes in the ASS2 format and some preparations are made to support further evolutions of the ASS format, such as ASS3 and AS5. The actual changes are too many to list here.

A new [unicode.lua](#) include file has been introduced. This file provides a much more clean and usable interface for manipulating UTF-8 encoded text than the old `next_utf_char` and `utf_len` functions.

Last but not least, the text layout algorithms used in karaskel have been improved a lot and now, among other things, support sub-pixel positioned text, proper handling of inter-syllable spaces and an advanced furigana layout algorithm. The furigana layout algorithm also produces data structures so similar to syllable structures you can in effect use exactly the same code to produce effects for both main syllables and furigana.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua

Karaskel
concepts:

[Style tables](#) • [Dialogue line tables](#) • [Syllable tables](#) • [Inline effects](#) •
[Furigana](#)

Karaoke Templater Reference: Declaring template and code lines

This page describes how a template line or code line is declared.

- Template lines and code lines are always marked Comment.
- The first space-separated word in the Effect field determines whether a line is a template line, a code line, a timed karaoke line, a styled karaoke line or something otherwise undetermined.
- If the first word in the Effect field is `template`, the line is a template line.
- If the first word in the Effect field is `code`, the line is a code line.
- If the Effect field is exactly equal to `fx`, the line is a styled karaoke line. Styled karaoke lines are deleted during execution of Karaoke Templater.
- If the Effect field is `Karaoke`, `karaoke` or empty, the line is a timed karaoke line.
- If the Effect field contains anything else, the line is an undetermined type and is not touched by Karaoke Templater.

Template lines and code lines can have additional text after the `template` or `code` keywords. This text is parsed as a series of space-separated words and is called modifiers. See [Template modifiers](#) for more information on this.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Karaoke Templater Reference: Template execution rules and order

This page describes various technical details about how Karaoke Templater (*kara-templater*) works and will try to explain why various things work as they do and why some things can't and won't work.

Most of this is technical details you don't need to know to use *kara-templater*, but if you see some behaviour you don't understand this page might explain it.

Concepts

These are some terms and concepts used throughout the description. The names are close to or the same as those used in the actual script.

- **tenv** - The template environment, or **code execution environment**.
- **varctx** - The inline variable context, the storage for the actual values of the **inline variables**.
- **template** - The basic "execution unit" of *kara-templater*, a template is essentially a mini-program compiled and executed by *kara-templater*.
- **code template** - A template that runs a chunk of Lua code but doesn't produce output. (Declared with the *code* keyword.)
- **output template** - A template that produces output lines from some karaoke data input. (Declared with the *template* keyword.)
- **code line** - A line in the subtitle that defines a code template.
- **template line** - A line in the subtitle file that defines an output template, or part of one. (One *line* class output template can span multiple template lines.)
- **class** - A class is a kind of template. There's four basic classes, *once*, *line*, *syl* and *furi*, the first only available for code templates.
- **modifier** - Modifiers affect how and when templates are executed.
- **template text** or just **text** - The "text" part of a template, either the Lua code in a code template or the template code in output templates. *line* class output templates also have a *pre-line text*.

Startup

The first thing *kara-templater* does is simply use **karaskel** to collect some basic information on the subtitle file. It always passes `true` for *generate_furigana* in the `karaskel.collect_head` function, meaning that **furigana** styles are always generated, unless they already exist.

It then collects all template lines in the file.

Collecting, parsing and compiling templates

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Contents [\[hide\]](#)

- 1 Concepts
- 2 Startup
 - 2.1 Collecting, parsing and compiling templates
 - 2.2 Clean-up
 - 2.3 Initialising the tenv
- 3 Run once templates
- 4 Iterate through karaoke lines in file
 - 4.1 Matching a template against a line
- 5 Applying line class templates
- 6 Applying syl and furi class templates
- 7 Old mid-level description

Every line in the file is visited and checked for being a template line, ie. be a comment and have the first word in the Effect field be *code* or *template*.

The details aren't important here, but every modifier name found in the Effect field either sets a flag in the template or a value corresponding to the parameter given to the modifier.

When a named *line* class template lines is encountered, first it's checked if there is already a *line* class template with that name. If there isn't one, a new one is created with that name and initialised with the given modifiers. If there is already one with that name, *the text of the template line is appended to the current text of the template* and modifiers present in the new template line but not in the current template are added to the template. Modifiers cannot be removed from templates in this way or any other way. The text of *pre-line* template lines is added to the *pre-line text* of the template rather than the regular text.

The templates of different classes are each put in their own "bucket", so for example *line* and *syl* templates are not kept together.

Clean-up

After all templates have been collected etc., all old and no longer needed lines are deleted from the subtitle file. This mainly includes lines with *fx* in the Effect field, as those are assumed to have been generated in a previous run of *kara-templater*, so they should be replaced in this new run.

Initialising the *tenv*

The last thing done before starting actually running the templates is initialising the runtime environment for the templates. Basically, as much as possible before any templates are run, is put into *tenv*. See [Code execution environment](#) for more details on what's in there. (Basically everything but *line*, *orgline*, *syl* and *basesyl*.)

Run *once* templates

All templates in the *once* class are executed first. Nothing truly exciting happens here, the main thing that can happen is that some more things are added to *tenv*.

Iterate through karaoke lines in file

Every non-template line in the file is now run through and has all templates attempted applied in order.

- If a line is a comment and its Effect field doesn't contain *Karaoke* it is skipped immediately.
- If a line is not a comment and its Effect field contains anything else that *Karaoke* or nothing (is blank) it is skipped immediately.
- Kara-templater attempts to match all templates against all other lines.

Each line that hasn't been rejected by the above points is now run through all templates in three steps.

First, all *line* class templates are attempted matched against the line and then run on the line. See below for a definition of when a template matches a line.

Next, all syllables on the line are run through in order and for each, all *syl* class templates are attempted matched against the line and then run on the syllable.

Finally, all furigana syllables on the line are run through in order, for each every *furi* class template is attempted matched against the line and then run on the furigana syllable.

It is important to note that the syllables and furigana syllables looped through are the parsed-stored syllables, not multi-highlight virtual syllables, not per-character virtual syllables and not a combination.

Example

Assume there are three *syl* class templates: A, B and C.

- A is a regular template with neither *multi* nor *char* modifier.
- B has the *multi* modifier but not *char*.
- C has both the *char* and *multi* modifiers.

Now these templates are applied against a line with 2 syllables. This happens in order:

- Syllable 1 is picked.
 - Template A is matched against the line. It matches.
 - Template A is applied on syllable 1.
 - Template B is matched against the line. It matches.
 - Syllable 1 is split into multi-highlight pseudo-syllables 1.1 and 1.2
 - Template B is applied on pseudo-syllable 1.1.
 - Template B is applied on pseudo-syllable 1.2.
 - Template C is matched against the line. It matches.
 - Syllable 1 is split into per-character pseudo-syllables 1.a and 1.b
 - Syllable 1.a and 1.b are further split into per-character pseudo-syllables 1.a1, 1.a2, 1.b1 and 1.b2.
 - Template C is applied on pseudo-syllable 1.a1.
 - Template C is applied on pseudo-syllable 1.a2.
 - Template C is applied on pseudo-syllable 1.b1.
 - Template C is applied on pseudo-syllable 1.b2.
- Syllable 2 is picked.
 - Processing proceeds similar to syllable 1.

Also see later down for more details on multi-highlight and per-character pseudo-syllables.

If any template matches at any time during the three steps above the (original) line is marked as "timed karaoke" and is then made into a comment with `karaoke` in the Effect field.

Matching a template against a line

Templates are always matched against a line, not against a syllable or otherwise.

- If the template has the *fxgroup* modifier and the *fxgroup* named is disabled, the template never matches anything.
- If the template has the *all* modifier it always matches any line.
- If the template has the same *Style* as a line, it matches the line.
- Otherwise the template does not match the line.

Applying *line* class templates

Todo: write this

Applying *syl* and *furi* class templates

Todo: write this

Old mid-level description

Main kara-templater process:

1. Collect header
 1. Find all header information, primarily PlayResX and PlayResY
 2. Find all styles
 3. Generate furigana styles for styles missing them
2. Collect templates and delete existing "fx" lines
3. Initialise tenv
 1. Add "string", "math" and "_G" references
 2. Add "tenv" self-reference
 3. Add "retime" function
 4. Add empty "fxgroup" table
4. Run every "code once" template
5. For every pre-existing dialogue line in subtitle file:
 - a. If Effect field start with "code" or "template":
 1. Skip line
 - b. Else:
 1. If Effect field is not empty and not "karaoke":
 - a. Skip line
 2. If Effect field is empty and line is a Comment:
 - a. Skip line
 3. Preprocess line with karaskel
 4. Initialise varctx
 5. Reset tenv
 1. Set "orgline" to input line
 2. Set "line", "syl" and "basesyl" to nil
6. For every "line" template:

If template matches line style or template is "all":
Repeat this "template.loops" number of times:

 1. Set "tenv.j" to loop counter
 2. a. If template is a code line:
 1. Set "tenv.line" to input line
 2. Run code
 - b. Else:
 1. Produce output line as copy of input line
 2. Set "tenv.line" to output line
 3. Initialise output line Layer to template Layer
 4. Initialise output line Text to empty
 5. If template has pre-line:
 1. Run pre-line template
 2. Append result to output Text
 6. a. If template has regular line:

```

For every syllable in input line:
1. Set "tenv.syl" to syllable
2. Update varctx for syllable
3. Run line template
4. Append result to output Text
5. If "notext" is not set:
  a. If "keptags" is set:
    1. Append "syl.text" to output Text
  b. Else:
    1. Append "syl.text_stripped" to output Text
  b. Else:
    a. If "keptags" is set:
      1. Append "syl.text" to output Text
    b. Else:
      1. Append "syl.text_stripped" to output Text
7. Set Effect field of output line to "fx"
8. Append output line to subtitle file
7. For every main syllable in line:
  For every "syl" template:
  If template matches line style or template is "all":
  If template is not in a disabled fxgroup:
  1. Set "tenv.syl" to syllable
  2. Update varctx for syllable
  3. If syllable inlinefx does not match template inlinefx:
    1. Skip syllable
  4. If template has "noblack" set and syllable is blank:
    1. Skip syllable
  5. If template is "char":
    1. Create "charsyl" as copy of syllable
    2. Set "tenv.basesyl" to current "tenv.syl"
    3. Set "tenv.syl" to "charsyl"
    4. For every Unicode character in syllable:
      1. Calculate virtual syllable characteristics for "charsyl"
      2. Update varctx for "charsyl"
      3. Continue syllable processing for the virtual syllable (from
5.b.7.6.)
6. If template is "multi":
  1. Create "hlsyl" as copy of syllable
  2. Unless "tenv.basesyl" already exists, set it to "hlsyl"
  3. Set "tenv.syl" to "hlsyl"
  4. For every highlight on syllable:
    1. Calculate virtual syllable characteristics for "hlsyl"
    2. Update varctx for "hlsyl"
    3. Continue syllable processing for the virtual syllable (from
5.b.7.7.)
7. a. If template is a code line:
  1. Set "tenv.line" to input line
  2. Run code
  b. Else:
  Repeat this "template.loops" number of times:
  1. Set "tenv.j" to loop counter
  2. Create output line
  3. Set output line Style to virtual syllable style
  4. Set output line Layer to template layer
  5. Set "tenv.line" to output line
  6. Run template
  7. Set output line Text to result
  8. a. If "keptags" is set:
    1. Append "syl.text" to output line Text
  b. If "notext" is not set:
    1. Append "syl.text_stripped" to output line Text

```

- c. Otherwise nothing is appended
- 9. Set output line Effect to "fx"
- 10. Append output line to subtitle file
- 8. For every furigana part in line:
 - Same process as for main syllables (5.b.7.)
- 9. If any non-code templates were applied to the line:
 - 1. Set input line to Comment
 - 2. Set input line Effect field to "karaoke"
 - 3. Store modified input line back to subtitle file

Running a code line:

- 1. Compile line text to a Lua function
- 2. If compilation failed, report error
- 3. Set compiled function's environment to tenv
- 4. Repeat this "template.loops" number of times:
 - 1. Set "tenv.j" to loop counter
 - 2. Run compiled function
 - 3. If an error occurred, report it

Running a single template:

- 1. Set result text to template
- 2. If there is a varctx:
 - For every match of "\$([a-zA-Z_]+)" in result text:
 - 1. Lowercase the captured name
 - 2. a. If the captured name is a field in varctx:
 - 1. Replace match in result text with value from varctx
 - b. Else:
 - 1. Report warning
 - 2. Keep match as-is in result text
- 3. For every match of "!(.~)!" in result text:
 - 1. Append "result " to captured code
 - 2. Compile captured code to a Lua function
 - 3. If compilation failed, report error
 - 4. Set compiled function's environment to tenv
 - 5. Run compiled function
 - a. If compiled function produced an error:
 - 1. Report error
 - 2. Leave match in result text
 - b. Else:
 - 1. Replace match with result of running the function

Todo: Turn this into something more reasonable?

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Category: Pages with Todo items

Karaoke Templater Reference: Template modifiers

Template lines and code lines can take a number of modifiers.

This is a space-separated list of words in the Effect field following the `template` or `code` keyword.

While modifiers can be combined to some extent, not all are compatible, and not all work on both code lines and for templates.

There is a special set of modifiers that declare the class of the template line or code line.

Class declaring modifiers

Both template lines and code lines can be produced without having a class modifier. Having one is still recommended for clarity, however.

A template line without a class modifier is implicitly given the `syl` modifier.

A code line without a class modifier is implicitly given the `once` modifier.

once

This class modifier is only valid for code lines.

Code lines with the `once` modifier are run exactly once during Karaoke Templater execution, and are always run before any other code lines or templates. They are run in the order they are declared.

"code once" lines are primarily intended to declare functions for use in templates.

Example

```
Comment: 0,0:00:00.00,0:00:05.00,Default,,0000,0000,0000,code  
once,function setlayer(newlayer) line.layer = newlayer; return ""; end
```

This example declares a new function that changes the Layer field in the output line.

line *[name]*

This class modifier is valid for both code lines and template lines.

When used on template lines it takes an optional parameter naming the line template the template line participates in. The template name must not match any template modifier names.

Anonymous line templates (with no template name given) can not have pre-line template

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Contents [\[hide\]](#)

- 1 [Class declaring modifiers](#)
 - 1.1 [once](#)
 - 1.2 [line \[name\]](#)
 - 1.3 [pre-line \[name\]](#)
 - 1.4 [syl](#)
 - 1.5 [furi](#)
 - 1.6 [syl furi](#)
- 2 [Other modifiers](#)
 - 2.1 [all](#)
 - 2.2 [char](#)
 - 2.3 [fx name](#)
 - 2.4 [fxgroup name](#)
 - 2.5 [keep tags](#)
 - 2.6 [multi](#)
 - 2.7 [noblank](#)
 - 2.8 [notext](#)
 - 2.9 [repeat n, loop n](#)

text.

Code lines can not be named, they must be anonymous.

Named line template lines append to the template text in the order they appear. The appending of template text happens at template parse time, not at execution time.

Example

```
Comment: 0,0:00:00.00,0:00:05.00,Default,,0000,0000,0000,code_  
line,fxgroup.funky = line.actor == "funky"
```

This code line is run once per input line. It enables/disables an effect group named "funky" depending on the Actor field of the input line.

Example

```
Comment: 0,0:00:00.00,0:00:05.00,Default,,0000,0000,0000,template_  
line,{\r\t($start,$end,\bord0)}
```

This template line declares an anonymous line template. The effect produced will transform the border of each syllable to zero during the syllable's duration.

Example

```
Comment: 0,0:00:00.00,0:00:05.00,Default,,0000,0000,0000,template_line_  
jumper,{\r\t($start,$mid,\frz-0.1)\t($mid,$end,\frz0)}
```

This template line appends to a line template named "jumper" or creates it if it doesn't exist. Together with the pre-line template example given below, this will produce a "jumping" effect for the syllables.

pre-line [*name*]

This class modifier is only valid for template lines.

The `pre-line` modifier takes an optional parameter naming the line template the template line participates in. The template name must not match any template modifier names.

Anonymous line templates with only pre-line text leave the original input line text alone and just prepends the template text to the line.

Named pre-line template lines append to the pre-line template text in the order they appear. The appending of template text happens at template parse time, not at execution time.

Example

```
Comment: 0,0:00:00.00,0:00:05.00,Default,,0000,0000,0000,template_pre-  
line,{\bel}
```

This template line declares an anonymous line template, that will prepend `{\bel}` to all matching lines.

Example

```
Comment: 0,0:00:00.00,0:00:05.00,Default,,0000,0000,0000,template_pre-  
line_jumper,{\org(-10000,$y)}
```

This template line appends to the pre-line template text of a line template named "jumper", or creates it if it doesn't exist. Together with the line template example given above, this will produce a "jumping" effect for the syllables.

syl

This class modifier is valid for both code lines and template lines.

Syl templates can not be named.

Example

```
Comment: 0,0:00:00.00,0:00:05.00,Default,,0000,0000,0000,template  
syl,{\pos($x,$y)}
```

This template line declares a syl template that simply positions the syllable text.

furi

This class modifier is valid for both code lines and template lines.

Furi templates can not be named.

Example

```
Comment: 0,0:00:00.00,0:00:05.00,Default,,0000,0000,0000,template  
furi,{\pos($x,$y)}
```

This template line declares a furi template that simply positions the syllable text. It's not needed to do anything further to get correct furigana formatting.

syl furi

It's possible to combine the `syl` and `furi` class modifiers. This results in two identical templates being generated from the template line, one syl template and one furi template.

This is the only possible combination of class modifiers, they are otherwise exclusive.

Other modifiers

all

Apply template to all styles, not just the one of the template line.

Applicable for both code lines and templates, and for all classes.

Example

```
Comment: 0,0:00:00.00,0:00:05.00,Default,,0000,0000,0000,template syl
```

```
all, {\pos($x,$y)}
```

This template will be applied to every single syllable in the entire subtitle file, regardless of the style of the line they are on.

char

Make the template work per-character instead of per-syllable. This changes application order semantics in a significant way, see [Template execution and order](#) for details.

While this will work on code lines, it is generally not useful, see the discussion on execution order.

Example

```
Comment: 0,0:00:00.00,0:00:05.00,Default,,0000,0000,0000,template_syl  
char, {\pos($x,$y)}  
Comment: 1,0:00:00.00,0:00:05.00,Default,,0000,0000,0000,template_syl  
char, {\pos($x,$y)\bord0}
```

Every single character on the line will be positioned separately. For each syllable, each template will apply for all characters in one go, and not be applied interleaved.

For example, if there are two syllables, "ab" and "cd", and the above two templates are applied to them, the result will be 8 lines with the following text, in this order:

```
{\pos($x,$y)}a  
{\pos($x,$y)}b  
{\pos($x,$y)\bord0}a  
{\pos($x,$y)\bord0}b  
{\pos($x,$y)}c  
{\pos($x,$y)}d  
{\pos($x,$y)\bord0}c  
{\pos($x,$y)\bord0}d
```

fx name

Make template only apply to syllables that have the named [inline-fx](#). Specifying an inline-fx name is required; the name may also overlap with template modifier names though this is not recommended.

Example

```
Comment: 0,0:00:00.00,0:00:05.00,Default,,0000,0000,0000,template_syl fx  
drop, {\move($x,$y,$x,!$y+30!,$start,$end)}
```

With this template, all syllables that have the inline-fx "drop" will get an additional line produced, where the syllables moves down 30 pixels during its duration.

All other template lines that don't have *fx* specified will still be applied as usual to those syllables as well.

fxgroup *name*

Declare template to be in the named effect group. Specifying an effect group name is required; the name may also overlap with template modifier names and Lua reserved words, though this is not recommended.

Example

There is an example of *fxgroup* on the [Code execution environment](#) page.

keep tags

Specify that the original tags must be kept in the syllable after application.

This has no effect when combined with `char` or `multi`.

Example

```
template line keep tags:
{\r\t($start,!$start+1!,\frx40)\t(!$start+1!,$end,\frx0)}
karaoke: {\k21}hi{\k10}gu{\k23}ra{\k22}shi {\k38}ga
{\k37\lc&H0000FF&}na{\k37}ku
```

The syllables "tip" back over a bit during highlight. One of them ("na") is coloured differently by putting an override tag in the timed karaoke line, but the following syllables don't get it because of the customary `\r` at the start of the template.

The *notags* modifier ensures that the special colour of the special syllable gets carried over to the output.

multi

Make the template apply per-highlight in [multi-highlight](#) timed karaoke. This changes application order semantics in a significant way, see [Template execution and order](#) for details.

While this will work on code lines, it is generally not useful, see the discussion on execution order.

Example

```
template syl multi:
{\an5\pos($scenter,$smiddle)\la&HFF&\t($start,$end,\bord5\3a&HFF&)}
karaoke: {\k33}風{\k36}#\k89}の{\k46}花{\k28}#\k57}よ
```

The timed karaoke line uses basic multi-highlight markup, the `#` syllables, to create multi-highlight syllables. Such, the 風 (ka-ze) and 花 (ha-na) kanji each get stored as a single syllable that gets two highlights each, and the `#` characters aren't displayed at all in the applied effect. (They will still display if you try to play the timed karaoke line without applying any templates.)

The template uses the *multi* modifier to signal that it wants to use multi-highlights instead of just one highlight/application per displayed syllable. The effect is a kind of simple "exploding border", but it explodes twice on both the 風 and 花 kanji. If the *multi* modifier wasn't there, it would only explode once on each.

noblank

Specify that the template will not be applied to syllables considered "blank". A syllable is considered blank if its tag-stripped text consists only of a combination of ASCII whitespace characters and ideographic fullwidth space characters, or is completely empty. A syllable is also considered empty if its duration is zero.

See the *notext* modifier below for an example.

notext

Specify that the original text will not be appended to the output line.

This is intended for use primarily with templates that output drawing tags and similar.

Not applicable for code lines.

Example

```
code once: sword_shape = "m 0 0 1 5 -5 1 5 -30 1 10 -30 1 10 -32 1 2
-32 1 2 -40 1 -2 -40 1 -2 -32 1 -10 -32 1 -10 -30 1 -5 -30 1 -5 -5 "
template syl notext noblank: {\an5\move($scenter,!$smiddle-
30!, $scenter, $smiddle, !$start-20!, $start)\p2}!sword_shape!
```

The first code line defines a vector drawing shape for convenience, so it doesn't clutter up the actual template lines later on. The drawing is of a small simple sword pointing downwards. The effect itself is these small swords dropping down onto the syllables, by a move.

The template uses the *notext* modifier to avoid getting the original syllable text shown, because it's being replaced with a vector drawing here. Also the *noblank* modifier is used to avoid producing anything for "invisible" syllables, eg. we don't want a sword dropping down on a lone timed space, that just looks dumb.

repeat *n*, loop *n*

Specify that the template will be applied the given number of times. Specifying the number of loops is required. The number of loops must be a constant integer number, it can not be a variable or otherwise calculated dynamically.

`repeat` and `loop` are synonymous.

Note that the execution order of looped line templates and looped `syl/furi` templates is different. See [Template execution and order](#) for details.

Example

```
template syl loop 4: {\move($x,$y,!$x+math.random(-
30,30)!, !$y+math.random(-
30,30)!, $start, $end)\alpha&Hc0&\t($start, $end, \alpha&HFF&)}
```

The *loop* modifier is used to create 4 copies of the syllable for each time this template is run. Each of those move in a random direction, up to 30 pixels away in X and Y direction. They also fade out.

The starting alpha for each copy, `&Hc0` is chosen as $256 - (256 / 4)$, 4 being the number of loops made. This way, the opacity for each copy adds

up to exactly 256. (Technically it should be 255, but that can't be achieved with an even number of loops.)

Also see the examples on the [Code execution environment](#) page for more advanced usage.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Karaoke Templater Reference: Inline variables

This page describes the **inline variables** also known as **dollar variables** available in Karaoke Templater.

How to use inline variables

All inline variables start with a dollar-sign. They only work in template lines, not in code lines. You can, however, use them in code blocks on template lines.

Here is an example of how a template text using inline variables could look:

```
{\pos($x,$y)\t($start,$end,\bord0)}
```

The highlighted parts are the inline variables in the template.

When a template is applied, the first thing that happens is that all inline variables are found and replaced with their value. Eg. in the above example `$x` and `$y` are replaced with the X and Y coordinates of the syllable the template is being applied to, and `$start` and `$end` are replaced by the start and end times of the syllable.

Case does not matter for inline variables, `$start`, `$START` and `$StArT` all work and give the same result.

Limitations

Inline variables are not "intelligent", they do the same no matter where you place them or use them, it doesn't "know" what tag it's being used with. Not every variable can be used with success in all places, and the meaning of some are affected by usage of eg. the `retime` function. In these cases, inline variables may not be appropriate and you need to use code blocks.

Because inline variables have their values determined as the very first thing when a template is applied you can't affect their values in any way.

Using inline variables is an easy way to get started with an effect, but for many advanced effects they might not be the best choice.

All positioning and sizing inline variables (such as `$y`, `$right` and `$width` are rounded to the nearest whole pixel, unlike the values in the internal data structures which you can get in code blocks, they have sub-pixel precision.

Line and syllable variables

The inline variables exist in both "line" and "syllable" variants. The "line" variants contain information about the entire line being processed, the "syllable" variants contain information about the current syllable being processed.

There are also "automatic" variants of most of the variables, there are either the line or the syllable variant depending on what kind of template they are used in. In pre-line templates the automatic inline variables refer to the line variants, and everywhere else

Contents [hide]

- 1 How to use inline variables
 - 1.1 Limitations
- 2 Line and syllable variables
- 3 The variables

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

they refer to the syllable variants.

The variables

The line variables that also exist as automatic variants all start with a lowercase L ("ell") letter, the syllable variants start with the letter S.

Line variants	
layer	line layer
lstart, lend, ldur, lmid	line start time, end time, duration and midway, all absolute times in milliseconds
style	name of the line style
actor	name of the line actor
margin_l, margin_r	effective left and right margin (line if nonzero, otherwise style)
margin_v, margin_t, margin_b	effective vertical, top and bottom margin, vertical and top is same
syln	number of syllables on line
li	line index (first physical line in file is 1)
lleft, lcenter, lright	line left, horizontal center and right edges, taking margins and alignment into account, rounded to an integer value
ltop, lmiddle, lbottom	line top, vertical middle and bottom edges, taking margins and alignment into account, rounded
lx, ly	line x and y position suitable for a \pos command when alignment is not overridden
lwidth, lheight	line width and height in pixels, this is rounded and might not match exactly with the positioning variables
Syllable variants	
sstart, send, smid	syllable start, end and midway times relative to start of line, suitable for putting into \t and \move
sdur, skdur	syllable duration in milliseconds and centiseconds
si	syllable index from start of line
sleft, scenter, sright	absolute left, horizontal center and right edges for syllable from left edge of screen, suitable directly for \pos and \move
sbottom, smiddle, stop	absolute bottom, vertical middle and top edges for syllable from top edge of screen, suitable directly for \pos and \move, adjusted for furigana positioning if needed
sx, sy	syllable absolute x and y position in default alignment, suitable for using directly in \pos and \move
swidth, sheight	syllable width and height in pixels this is rounded and might not match exactly with the positioning variables
Automatic variants	
start, end, mid	start and and midway time for line/syllable; absolute for lines and relative for syllables

dur, kdur	duration in milliseconds and centiseconds of line/syllable
i	line or syllable index
left, center, right	left, center and right edges of line/syllable, absolute from left screen edge
top, middle, bottom	top middle and bottom edges of line/syllable, absolute from top screen edge
x, y	x and y position of line/syllable when using default alignment
width, height	width and height of line/syllable in pixels, this is rounded and might not match exactly with the positioning variables

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Karaoke Templater Reference: Code lines and blocks

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Code lines and blocks in Karaoke Templater allows you to create advanced effects by incorporating small snippets of Lua code. This can range from simple mathematical expressions adding two numbers to complex functions that for example could generate various shapes in cycling colours.

Both code lines and code blocks are run in a separate semi-closed execution environment, meaning they are mostly undisturbed by the primary Lua environment the Karaoke Templater script itself runs in. For an overview of what variables are available in the code line/block execution environment see: [Code execution environment](#).

Contents [\[hide\]](#)

- 1 Code lines
 - 1.1 Classes of code lines
- 2 Code blocks
 - 2.1 Hints for using code blocks

Code lines

A code line is a special kind of template line. Instead of using the `template` keyword in the Effect field it uses the `code` keyword. A code line contains only Lua code and does not by itself produce any lines in the resulting file.

The two primary uses of code lines are:

- Defining/updating variables for use later in templates
- Defining functions for use later in templates

For example, if you need a random number, but also need to use it twice in one template, you can use a code line to first generate the number and store it to a variable, then use that variable in your template line.

Another example could be defining a function that produces a random colour.

Classes of code lines

Like there's multiple classes of template lines there's also multiple classes of code lines. Some of them are the same, and some only exist for one or the other.

You specify the class of the code line in the Effect field after the `code` keyword. The possible classes are:

once

Code lines in the `once` class are run exactly one time, before any templates are applied. This is usually the best place to define functions and general tables of values you need to look up later.

This is the default class, if you don't specify a class for a code line it's automatically in the `once` class.

line

Code lines in the `line` class are run when a new line is encountered. They are run once per line. They are run interspersed with `line/pre-line` templates in the order they appear. (There are no "pre-line" code lines.)

syl

Code lines in the `syl` class are run when a new syllable is encountered. They run once per syllable. They are run interspersed with `syl` templates.

furi

Code lines in the `furi` class are run when a new furigana syllable is encountered. They run once per furigana syllable. They are run interspersed with `furi` templates.

You *cannot* have templates with `char` or `multi` modifiers run per-character/per-highlight interspersed with code lines. This is a limitation of the execution model. This may or may not change in later versions of Karaoke Templater.

Code blocks

A code block is a block of Lua code within a template line. Code blocks are used to insert more complex things than can be expressed with [inline variables](#).

Code blocks are required to be single Lua expressions, since a `return` statement is automatically prepended to the code. This means you (among other things) can't do assignments or use `if` statements within code blocks, you must use a code line if you want to do any of those things. (There is a way to do basic conditionals in code blocks though, see below.)

You create a code block by surrounding the code by exclamation marks, like this:

```
{\t($start,!syl.start_time+20!,\bord0)}
```

It is possible to use inline variables within code blocks. They are expanded before the code block is parsed, so to the Lua interpreter the inline variables look like regular constants.

Hints for using code blocks

Most simple mathematical expressions work just like you'd expect them to. Operator precedence rules are those of regular arithmetic.

A code block should always return a string or numeric value, if it returns a boolean, a table or something else it might cause a warning and the resulting line containing the wrong output.

To create simple conditionals within code blocks you can use the `and` and `or` operators to chain values and conditions. For example:

```
{\k!(syl.duration>100) and "f" or ""!$kdur}
```

If the syllable duration is longer than 100 ms the first sub-expression is true, and the code block returns `"f"`, otherwise the entire `and` expression is false, and the right-hand argument of the `or` expression is returned.

In Lua, `and` binds stronger than `or` meaning that `and` expressions are evaluated first. In the above expression the effective grouping is like this: `((syl.duration > 100) and "f") or ""`

Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Karaoke Templater Reference: Code execution environment

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

The Lua code in code blocks and on code lines is run in a separate global environment such that it won't accidentally disturb the main script function.

You can store your own data in this environment for use later, for example pre-compute some values on code-lines and later insert them using code blocks, but it also contains several pre-defined variables and functions designed to make it easier writing effect templates.

It's important to understand that the contents of code execution environment and the [inline-variables](#) (\$-variables) are not related. You cannot change an inline-variable by changing something in the code execution environment nor can you add new ones. However, you can create and re-define the contents of the code execution environment.

Line and syllable information

The code execution environment contains a few variables pointing to the current line and syllable structure being processed, as well as some more supporting tables. These are just references to the structures produced by [karaskel](#) and are not modified in any way.

You should treat these as read-only, except `line`. If you change the other ones, the kara-templater script might start misbehaving.

- **line** - The line currently being produced, changing this will affect the resulting line in the file. See the [reference for dialogue line tables](#).
- **orgline** - The original line, this is the source line the current syllable is located on.
- **syl** - The current syllable structure. If the current template is a *furi* template, it's the current furigana syllable. If the current template has one or both of the *char* or *multi* modifiers, this is a pseudo-syllable structure, a copy of the original syllable structure with several values changed to look like the current part of the syllable being processed. Also see the [reference for syllable tables](#).
- **basesyl** - Usually the same as `syl`, except when the template has the *char* or *multi* modifier, then this is the original syllable. (If `syl == basesyl` is true, then the current template is neither *char* nor *multi*.)
- **meta** - Contains various metadata about the script, namely the contents of the *Script Info* section. Most importantly, it has the `res_x` and `res_y` fields describing the script resolution.

All of these variables are reset to `nil` whenever processing starts for a new line, except `meta`. They are then set to the relevant value whenever processing hits a new stage. This means that, for example *pre-line* templates only has `line` and `orgline` set and both `syl` and `basesyl` are `nil`. In *code once* templates, all of the variables except `meta` are `nil`.

Standard libraries and related things

Both the [string](#) and [math](#) Lua standard libraries are imported into the execution environment, as they are generally useful.

You can also access the main execution environment of the kara-templater script itself using the `_G` (underscore capital-G) variable and through that access the rest of the Lua standard library. For example, `_G.table.sort` refers to the regular `table.sort` function. See the [Lua 5.1 manual](#) for details on the available libraries.

You can also access the Automation 4 Lua standard libraries through the `_G` variable: [karaskel.lua](#), [unicode.lua](#) and [utils.lua](#).

There is also the self-reference `tenv` variable, this refers to the code execution environment itself. This means that `tenv.tenv == tenv` is true.

Contents [hide]

- 1 Line and syllable information
- 2 Standard libraries and related things
- 3 Utility functions
 - 3.1 retime
 - 3.2 relay
 - 3.3 restyle
 - 3.4 maxloop
 - 3.5 loopctl
- 4 Template execution data
 - 4.1 Looping templates
 - 4.2 Conditional templates with `fxgroup`

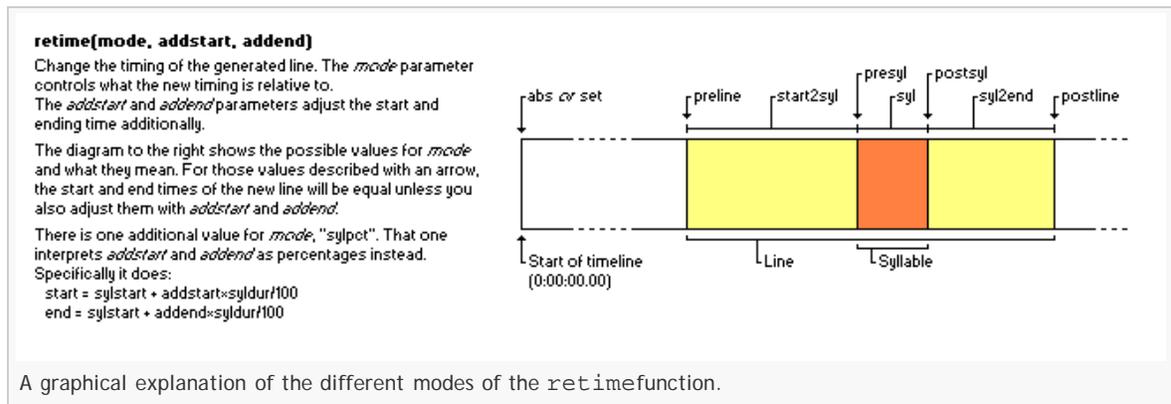
Utility functions

These functions help do more complex modifications of the output line (the `line` variable) and are unavoidable when creating complex effects.

Currently there is just one, but it is possible to define your own functions in code lines.

retime

Synopsis: `retime(mode, startadjust, endadjust)`



This function is usually used once in a template in a code block by itself. It adjusts the start and end time of the output line in various ways.

The *mode* parameter determines how the start and end times of the line are changed, it must be a string with one of the following values. Because it must be a string, the name of the mode must be enclosed in quotation marks!

The *startadjust* and *endadjust* parameters slightly change meaning based on the mode, but generally is a number of milliseconds added to the "base" time controlled by the mode.

Possible *modes*:

- **abs** or **set** - Both *startadjust* and *endadjust* are used as absolute time values to set the start and end time of the line directly.
- **preline** - Intended to make effects that happen before the actual line start. Both start and end time of the line are set to the start time of the line, then *startadjust* is added to the start time and *endadjust* added to the end time. Usually *startadjust* should be negative here and *endadjust* be zero.
- **line** - Use the regular line timings and just add *startadjust* to the start time and *endadjust* to the end time.
- **start2syl** - Intended to make the look of the syllable from the start of the line until it is highlighted. The start time of the line is kept and the end time is set to the start time of the syllable. Use *startadjust* and *endadjust* to offset the times.
- **presyl** - Similar to *preline* but for the syllable timing instead.
- **syl** - From start of syllable to end of syllable.
- **postsyl** - Similar to *presyl* but the base timing is the syllable end time instead of start time. You will usually want to use a zero *addstart* and positive *addend* here.
- **syl2end** - The time from the end of the syllable to the end of line, similar to *start2syl*.
- **postline** - Similar to *postsyl* but for the line timing instead.

There is also a special *mode*:

- **sylpct** - Both of *startadjust* and *endadjust* are treated as percentage values from 0 to 100 and are used to set the line timing to cover that part of the syllable's time.

Be careful with the `retime` function on *line* templates, if you use it directly on a *line* template it will probably not do what you want. You should only use it on *pre-line*, *syl* and *furi* templates. You should also only use it once in each template.

The `retime` function always returns the empty string (" ") which will cause it to output nothing when

used in code blocks, but still evaluate to true if used in boolean expressions.

Example

```
template syl: !retime("preline", -1000,
0)!{\pos($scenter,$smiddle)\an5\fscx0\fscy0\t(\fscx100\fscy100)}
```

This creates a kind of "pop-in" effect for the syllables that lasts 1 second (1000 milliseconds) before the actual line timing. The two important things to code: Quotation marks around "preline" and that the start offset is negative, -1000, because the start time needs to be moved backwards.

Example

```
template syl: !retime("syl", 0, 0)!{\pos($x,$y)\t(\fscx360)}
```

Makes the syllable spin around itself during its highlight. Unless you also have `syl` templates retimed to `start2syl` and `syl2end` the syllable will only be visible during its highlight. Note how retiming a syllable line to just the syllable time makes it unneeded to put start and end times in the `\t` tag, they default to the duration of the entire line and here the duration of the line is the duration of the syllable.

Example

```
template syl: !retime("sylvct", 0, 50)!{\move($x,$y,$x,!$y-10!)}
template syl: !retime("sylvct", 50, 100)!{\move($x,!$y-10!,$x,$y)}
```

These two templates together makes the syllable move 10 pixels upwards during the first half of its highlight and back down during the last half. Using `retime` is an easy way to get multiple `\move` tags to affect the same syllable; there can only be one `\move` tag on one line, but if you split the line into many "chained" times you can create an effect of the same syllable moving in several directions.

relayer

Synopsis: `relayer(newlayer)`

Change the Layer field of the generated line to *newlayer*.

Note: If you want a template to always generate lines with a static layer number, you do not need to use this function. You can just set the Layer field on the template line, it will transfer to the generated lines.

Example

```
template syl: !relayer(syl.i*5+20)!
```

Each syllable generated from the line gets a progressively higher layer number. The first syllable gets in layer 25, the second in layer 30 and so on, each syllable getting a layer 5 larger than the previous.

restyle

Synopsis: `restyle(newstyle)`

Change the Style field on the generated line to *newstyle*.

Be careful that this does not update the sizing and positioning information. If you want to use sizing or positioning information such as `$x`, `$lwidth`, `line.middle` and `syl.right` you must change to a style that uses the same font name, font size, boldness, italics, font encoding, X and Y scaling, character spacing, alignment and margins. If you change to a style where any of those properties are different, the positioning and sizing information will be invalid.

No example because the function has limited use.

maxloop

Synopsis: `maxloop(newmax)`

Dynamically control the number of times a template will be looped.

Be careful that using this function incorrectly might result in a template that loops forever, until Aegisub runs out of memory or you cancel the template application.

You do not need to use the loop modifier on templates to use this function.

Example

```
template syl: !maxloop(syl.width + 2*line.styleref.outline!){\clip(!line.left+syl.left-
line.styleref.outline+j-1!,0,!line.left+syl.left-
line.styleref.outline+j!,!meta.res_y!)\an5\move(!line.left+syl.center!,!line.middle!,!li
ne.left+syl.center!,!line.middle+math.random(-20,20)!,$start,$end)\shad0}
```

Cut each syllable into a number of slivers, the actual number depends on the size of the syllable. Each sliver moves randomly on highlight.

Example

```
template syl: !maxloop(j+1)!
```

Makes an infinite loop. It continually sets `j` one higher, making the loop never complete.

loopctl

Synopsis: `loopctl(newj, newmaxj)`

Control both loop variables. This function has questionable utility.

`newj` sets the new value of `tenv.j` and `newmaxj` sets the new value of `tenv.maxj`.

No example because the function has limited use.

Template execution data

These variables either give some further information on the status of the executing template or modify the rules for template execution in some way. They generally work together with specific template modifiers.

Looping templates

When a template with the `loop` or `repeat` modifier is running, two new variables are introduced in the code execution environment, `j` and `maxj`.

- `maxj` is the number of loops, ie. simply the parameter given to the `loop` modifier.
- `j` is the loop iteration counter, it starts at 1 in the first iteration and `maxj` in the last.

If you change `j` or `maxj` while a template is executing, you can affect the number of iterations the loop makes. The `maxloop` function is convenient for making dynamic loops.

Example

```
template syl loop 5:
{\an5\pos($scenter,$smiddle)\la&HFF&\3a&Hcc&\t($start,$end,\fscx!100+j*10!\fscy!100+j*1
0!\3a&HFF&)}
```

The syllable fill is hidden so only the border is visible, then several copies of the that border-only line is made through the loop, and made to "explode" to different, growing sizes using the `j` variable.

This example assumes that the style definition has shadow disabled but it does have a

border.

Example

```
template syl loop 20:  
{\move($x,$y,!$x+15*math.cos(math.pi*2*j/maxj)!,!$y+15*math.sin(math.pi*2*j/maxj)!, $start,$end)\t($start,$end,\alpha&HFF&)}
```

Here looping is used to [calculate several points on a circle](#) with radius 15 and make the syllables move out to those. Just by changing the number of loops in the Effect field you can make a more detailed circle because $j/\max j$ is used to calculate how large a portion of the total number of loops have been completed.

Conditional templates with fxgroup

The *fxgroup* modifier uses a special table **fxgroup** in the code execution environment to control whether a template will be executed or not.

The parameter given to the *fxgroup* modifier names a key (always a string) in the *fxgroup* table in the execution environment, and when a template assigned to an *fxgroup* is about to be executed, the value for that key in the *fxgroup* table is looked up. If the value is true or the key doesn't exist, the template is executed, if it's false the template is skipped.

While you can technically use any text string for *fxgroup* names, because they're used in Lua code it's best to avoid ones that overlap with Lua reserved words such as `end`, `break`, `return` and several more.

Example

```
code syl: fxgroup.long = (syl.duration > 200)  
template syl noblank: all here:  
template syl fxgroup.long: is long:  
karaoke: {\k10}huh? {\k40}wee~~
```

It's important to understand the template execution order to understand this example. For each input syllable (ie. "huh?" and "wee~~") all the templates and code lines are run in the order they appear.

This means that for "huh?", first the code line is run. It determines that the duration of that syllable is less than 200 ms and thus sets *fxgroup.long* to false. The first template has no *fxgroup*, so it's applied to the syllable then, outputting a line "all here: huh?", but the second template has *fxgroup* "long". This *fxgroup* was disabled for that syllable by the code line, so that template is not run at all.

For "wee~~", the code line determines that its duration is longer than 200 ms, so the "long" *fxgroup* is enabled. Then the first template outputs its line, "all here: wee~~", and when the second template is to run, its *fxgroup* is enabled now so it's also run, outputting "is long: wee~~".

Neither of the two templates will output anything for the zero'th syllable. The first template, because it has the "noblank" modifier, and the second because the zero'th syllable's duration is too short for the *fxgroup* to be enabled.

Automation 4

Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Automation 4 Lua Registration

Registration covers presenting your [Automation 4 Lua](#) script to Aegisub, providing information about it and registering what *features* it provides.

Features explained

One of the primary concepts in Automation 4 is the *feature*. A feature is something a script makes available for Aegisub to call back in response to a user action.

A feature is not a plain callback. Rather, it's usually a set of several callback functions as well as some information on how they should be presented to the user in the GUI.

One feature is the **macro**. A macro is presented as an item in the Automation menu. A macro has a name (the title show in the menu), a description (the text shown on the status bar when hovering over the menu item), a processing function (the function called when the user selects the menu item) and an optional validation function (determines whether the macro can even do any work in the current state.)

Another feature is the **export filter**. The export filter is presented in the [Export](#) dialogue and can be applied during an export operation. Export filters also have a name, description, processing function and then an optional configuration panel provider. The configuration panel provider is a function that returns a configuration dialogue definition structure which will be displayed in the Export dialogue when the export filter is enabled. The settings filled into the configuration panel are passed to the processing function when it is run.

Script information globals

A script can set a few global variables to provide metadata about the script to Aegisub. The information given with these variables are displayed in the [Automation Manager](#) dialogue and the Script Info dialogue.

- **script_name** (string) - Name of the script, this should be short.
- **script_description** (string) - Description of the purpose of the script. Shouldn't be too long either.
- **script_version** (string or number) - Version numer/name of the script. This is freeform, no specific meaning is assigned to this.
- **script_author** (string) - Author credits for the script.

All of these are optional, a script does not have to provide any of these. If no script name is given, the file name is used instead for display purposes.

Registration functions

Contents [hide]

- 1 Features explained
- 2 Script information globals
- 3 Registration functions
 - 3.1 aegisub.register_macro
 - 3.2 aegisub.register_filter
- 4 Feature callback functions
 - 4.1 Macro processing function
 - 4.2 Macro validation function
 - 4.3 Export filter processing function
 - 4.4 Export filter configuration panel provider

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

The registration functions are the functions provided by Automation 4 Lua you can call to make a feature available to Aegisub. You will usually call these in the top level, at the very bottom of your script.

aegisub.register_macro

Synopsis: `aegisub.register_macro(name, description, processing_function, validation_function)`

Register a macro feature.

- **name** (string) - The name displayed on the Automation menu. This should be very short, try three words or less, and should be in command tense.
- **description** (string) - The description displayed on the status bar when the user hovers the mouse over the menu item. This should be a concise description of what the macro does. Try to keep it at most 60 characters.
- **processing_function** (function) - The function that is called when the user selects the menu item. This must be a function with the [macro processing function API](#).
- **validation_function** (function, optional) - This function is called to determine whether the menu item should be available to the user or not. (Grayed out or not.) If no validation function is provided the macro is always available. This function must follow the [macro validation function API](#).

aegisub.register_filter

Synopsis: `aegisub.register_filter(name, description, priority, processing_function, configuration_panel_provider)`

Register an export filter feature.

- **name** (string) - The name displayed in the export filters list. The name should be rather short.
- **description** (string) - The description displayed in the description box when the user highlights the export filter in the Export dialogue.
- **priority** (number) - Determines the initial ordering of export filter application. Filters with higher priority are applied earlier than filters with lower priority. The user can change the filter application order in the Export dialogue. Priorities of the Aegisub built in export filters:
 - Transform Framerate = 1000 (karaoke effects should have higher priority than this)
 - Clean Script Info = 0 (your script might depend on the information cleaned by this)
 - Fix Styles = -5000 (should almost always run last)
- **processing_function** (function) - The function that is called when the user initiates the export operation. This must be a function with the [export filter processing function API](#).
- **configuration_panel_provider** (function, optional) - A function that provides a configuration panel for the export filter. If this function is not provided the export filter will not have a configuration panel. This function must follow the [export filter configuration panel provider API](#).

Feature callback functions

These are the callback functions you provide to the registration functions.

Macro processing function

Signature: `process_macro(subtitles, selected_lines, active_line)`

Macro processing functions passed to `aegisub.register_macro` must have this signature. The name `process_macro` is a placeholder for your own function name.

Important, undo points: If you make any modifications to the subtitles data you must always set an undo point when finished, otherwise the undo system will become inconsistent and the user will find that the Undo function behaves wrong.

- `subtitles` (user data) - The `subtitles object` you use to manipulate the subtitles with.
- `selected_lines` (table) - An array with indexes of the selected lines. The values in this table are line indexes in the `subtitles` object at its initial state. Only `dialogue` class lines can ever be selected.
- `active_line` (number) - The line that is currently available for editing in the subtitle editing area. This is an index into the `subtitles` object.

Return value: The macro processing function can optionally return a new `selected_lines` table, containing indexes of the lines that must be selected after the macro has returned.

Macro validation function

Signature: `validate_macro(subtitles, selected_lines, active_line)`

Macro validation functions passed to `aegisub.register_macro` must have this signature. The name `validate_macro` is a placeholder for your own function name.

Important, execution time: Validation functions should always run very fast. Do as little work as possible inside this function, because it is run every time the user pulls open the Automation menu, and every millisecond you spend in `validate_macro` is one millisecond delay in opening the menu. Consider that the user might have very large files open. Don't block the UI.

- `subtitles` (user data) - The `subtitles object` for the current subtitle file, this is *read-only*, you cannot modify the subtitles in the validation function. Attempting to modify the subtitles will cause a run-time error.
- `selected_lines` (table) - An array with indexes of the selected lines. The values in this table are line indexes in the `subtitles` object at its initial state. Only `dialogue` class lines can ever be selected.
- `active_line` (number) - The line that is currently available for editing in the subtitle editing area. This is an index into the `subtitles` object.

Return value: Boolean, `true` if the macro can run given the current state of `subtitles`, `selected_lines` and `active_line`, `false` if it can not.

Export filter processing function

Signature: `process_filter(subtitles, settings)`

Export filter processing functions passed to `aegisub.register_filter` must have this signature. The name `process_filter` is a placeholder for your own function name.

You do not have to worry about undo issues with export filters. You always work on a copy of the subtitle file.

- `subtitles` (user data) - The `subtitles object` you use to manipulate the subtitles

with. This is a copy of the open subtitles file, modifying this subtitles object does not modify the open file and will only affect the exported file.

- **settings** (table) - Configuration settings entered into the configuration panel or an empty table if there is no configuration panel. See the page on [configuration dialogues](#) for more information on the format of this table.

Return value: Nothing.

Export filter configuration panel provider

Signature: `get_filter_configuration_panel(subtitles, old_settings)`

Export filter configuration panel providers passed to `aegisub.register_filter` must have this signature. The name `get_filter_configuration_panel` is a placeholder for your own function name.

Important, execution time: This function is called automatically when the user opens the Export dialogue, and Aegisub blocks until it returns with a configuration panel. Consider that the user might have a very large file open, and that every millisecond spent creating your configuration dialogue is one more millisecond the user has to wait for the Export dialogue to open. Don't block the UI.

- **subtitles** (user data) - The [subtitles object](#) for the current subtitle file, this is **read-only**, you cannot modify the subtitles in the filter configuration provider. Attempting to modify the subtitles will cause a run-time error.
- **old_settings** (table) - Previous configuration settings entered into the configuration panel, if any. When an Automation 4 export filter is run, any configuration settings are automatically stored to the original file. If any stored settings exist for this filter, they are passed as *old_settings* so you can use them as a base for filling in defaults.

Return value: A configuration dialogue table, see the page on [configuration dialogues](#) for more information on the format of this table.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Category: [Lua Reference](#)

Automation 4 Lua Subtitle file interface

This page describes the subtitle file interface use in Automation 4 Lua scripts to manipulate subtitle files.

There is one special object (the `subtitles` user data object) which has a number of functions, and a number of table formats defined.

The subtitles object

Most (currently all) Automation 4 Lua feature functions are passed a **subtitles object** when called. This object is used to obtain data from and manipulate the subtitles the feature is being applied on.

A subtitles object can have two special properties depending on the context it is created for:

- **Read-only** - Some feature functions must not be allowed to modify the subtitle file at all. This includes for example [macro validation functions](#) and [export filter configuration panel providers](#), because this would be outside user expectations.
- **Allow undo points** - Only some feature functions allow undo points to be set. Currently this is only [macro processing functions](#). Due to the current (2.1/2.2) implementation of the Undo system in Aegisub, attempting to set undo points in any other context can cause data corruption, and thus is disallowed.

To allow the most flexibility, the subtitles object represents a complete ASS format file, line by line, including all meta-lines such as section headers, blank lines and semicolon comments.

The subtitles object supports the following operations:

- Retrieve number of lines
- Read line
- Append line (to end of file)
- Insert line (at arbitrary position in file)
- Replace line
- Delete line
- Create undo point

These operations are described in detail below. In all operation synopses and examples, `subtitles` is used for name of the subtitles object being operated on.

Retrieve number of lines

Synopsis:

- `num_lines = #subtitles`
- `num_lines = subtitles.n`

Contents [\[hide\]](#)

- 1 The subtitles object
 - 1.1 Retrieve number of lines
 - 1.2 Read line
 - 1.3 Append line
 - 1.4 Insert line
 - 1.5 Replace line
 - 1.6 Delete line
 - 1.7 Creating an undo point
- 2 Line data tables
 - 2.1 clear class
 - 2.2 comment class
 - 2.3 head class
 - 2.4 info class
 - 2.5 format class
 - 2.6 style class
 - 2.7 dialogue class
 - 2.8 unknown class

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

This operation retrieves the total number of lines in the subtitle file currently. This number only changes by using the other operations on the subtitle object, it cannot change spontaneously during the execution of a script.

- `num_lines` (number) - Number of lines in subtitle file.

The first syntax is the preferred one, per normal Lua 5.1 coding style.

Read line

Synopsis: `line = subtitles[i]`

This retrieves the indexed line and creates a new table object with data about it.

- `line` (table) - Table with data about the retrieved line. Will be `nil` if the line indexed is out of range.
- `i` (number) - Index into the subtitles file of the line number to retrieve. This is one-based, the first line in the file has index 1.

Aegisub internally stores the subtitle file as a linked list, which means that random access is slow, but sequential access is fast. While Automation 4 Lua presents the subtitles as if it was an array, it internally maintains a cursor used to optimise for sequential access. It is faster to access a line with an index close to the one you last accessed than to access one further away. It is always fast to access lines near the beginning or end of the file.

Append line

Synopsis:

- `subtitles[0] = line`
- `subtitles.append(line)`
- `subtitles.append(line1, line2, ...)`

Append one or more lines to the end of the subtitles file. In the first syntax, it is the number 0 (zero) used for index. (Setting index 0 causes an append operation.)

The third syntax supports appending multiple lines with one single operation.

- `line` (table) - The line object table to append to the subtitles file.

The latter function-call syntax is preferred for readability. The table index setting syntax is slightly faster.

Appending a line always runs in constant time, it does not move the cursor otherwise used to optimise sequential access.

Insert line

Synopsis:

- `subtitles[-i] = line`
- `subtitles.insert(i, line)`
- `subtitles.insert(i, line1, line2, ...)`

Inserts one or more lines into the subtitles file before the numbered line. In the first syntax, you supply a negative index, eg. to insert a line before line 5 you supply index -5 (negative 5.)

Inserting lines will move lines after the inserted lines to move down in index, such that old indexes will no longer be valid.

- **i** (number) - Index to insert before.
- **line** (table) - The line object table to insert into the subtitles file.

The latter function-call syntax is preferred for readability. The table index setting syntax is slightly faster.

Inserting lines uses the list cursor and will move it.

Replace line

Synopsis: `subtitles[i] = line`

Delete the indexed line and insert the given line in its stead.

- **i** (number) - The line index to replace at.
- **line** (table) - The line object table to replace with.

Replacing lines uses the list cursor and will move it.

Delete line

Synopsis:

- `subtitles[i] = nil`
- `subtitles.delete(i)`
- `subtitles.delete(i1, i2, ...)`
- `subtitles.deleterange(first, last)`

Remove one or more from the subtitles file. All lines after the deleted line(s) will move up to fill the deleted indexes, so old indexes will no longer be valid.

The third syntax supports deleting multiple indexed lines in one call. The indexes given must all be correct for the subtitle file's state before any lines are deleted.

The fourth syntax deletes a range of lines, both indexed lines inclusive.

- **i** (number) - Index of the line to delete.
- **first, last** (numbers) - Indices of the first and last lines of the range to delete.

Deleting lines uses the list cursor and will move it.

Creating an undo point

Synopsis: `aegisub.set_undo_point(description)`

You must always set an undo point at the end of running a macro, otherwise the Undo and Redo functions stop working correctly. You can set multiple undo points while running a single macro, but be aware that it might confuse the user. (You can try to see what happens if you don't set an undo point after changing the subtitle file, just don't release scripts without this to the public!)

Only available in macro processing functions, and doesn't make sense anywhere else either.

- **description** (string) - Text to appear in the Edit menu for the Undo and Redo items to describe the action that can be undone.

The behaviour of the undo system will change in the next major release of Aegisub, but that's far away. The exact way it will manifest in Automation 4 scripts can't be known yet.

This is not really a function in the subtitles object, but it is still closely tied to it.

Line data tables

When you read lines from the subtitle file object they will always be one of a few classes of lines, and when you write lines back to the subtitle file they must also follow the format of one of those classes.

The line data objects are regular Lua tables with some specific fields defined.

Here's a list of the different classes of lines:

- **clear** - blank line, there's nothing (or only whitespace) on the line
- **comment** - semicolon-comment, a line starting with a semicolon, usually ignored
- **head** - a section heading in the file, such as `[Script Info]`
- **info** - a key/value pair in the Script Info section of the file
- **format** - a Format line in the file, usually ignored
- **style** - a regular style definition line
- **dialogue** - a dialogue line, which may be a comment or not; these are the lines you see in the grid in Aegisub
- **unknown** - an unknown kind of line

There's three fields that always exist in all line data tables:

- **class** (string) - The name of the class of line this is, see the list above.
- **raw** (string) - The raw text of the line, from first to last character on the physical line.
- **section** (string) - Which section of the file the line belongs to. If the line is placed before the first section heading, this field is `nil`.

clear class

This class doesn't define any additional fields.

comment class

This class defines one additional field:

- **text** (string) - The text that follows the semicolon. Basically the same as the `raw` field with the first character chopped off.

head class

The `head` class doesn't define any additional fields, but the `section` field contains the name of the new section started.

info class

This class defines two additional fields:

- **key** (string) - The part of the line before the first colon, with leading and trailing spaces removed.
- **value** (string) - Everything after the first colon on the line, also with leading and trailing spaces removed.

format class

This class defines one additional field:

- **fields** (array table) - An array table of strings, the field names listed in the order they appear on the format line.

style class

This class defines a large number of additional fields. It's usually processed by the *karaskel* and modified a bit by that. See the *karaskel.lua* section on [style tables](#) for more information about this class.

dialogue class

This class defines a large number of additional fields. It's usually processed by the *karaskel* and has many calculated fields added by that. See the *karaskel.lua* section on [dialogue line tables](#) for more information on this class.

unknown class

No additional fields are defined by this class, due to its nature. This might be things like files embedded into the subtitles. You shouldn't try to work with these lines unless you really know what you're doing, deleting, modifying and inserting **unknown** lines has undefined consequences. (That means, even if it works today it might not work tomorrow or in the next version of Aegisub.)

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Category: Lua Reference

Automation 4 Lua Progress reporting

These functions are used to report various status and progress back to the GUI while a script is running.

Progress reporting

A progress dialogue box is always shown when an Automation 4 Lua script is running. You can use these functions to control what is displayed in it.

`aegisub.progress.set`

Synopsis: `aegisub.progress.set(percent)`

Set the position of the percent-done bar in the progress window.

- `percent` (number) - Number from 0 to 100.

`aegisub.progress.task`

Synopsis: `aegisub.progress.task(msg, ...)`

Set the "task" text in the progress window, this is the small text below the progress bar showing what the script is currently doing.

- `msg` (string) - A format string specifying the message, see the Lua standard string library `string.format` function for details on format strings.
- `...` - Parameters to the format string.

`aegisub.progress.title`

Synopsis: `aegisub.progress.title(title, ...)`

Set the title of the progress window, this is the large text displayed above the progress bar. This text should usually not changing while the script is running. By default this is set to the name of the feature running, eg. the name of the macro (menu item text) if it's a macro.

- `title` (string) - A format string specifying the title, see the Lua standard string library `string.format` function for details on format strings.
- `...` - Parameters to the format string.

`aegisub.progress.is_cancelled`

Synopsis: `cancelled = aegisub.progress.is_cancelled()`

Tells whether the user has clicked on the Cancel button.

You should call this function regularly during long operations, and stop the execution of your script as quickly as possible. If you're a macro you should either roll back all changes you have made if possible, or [set an undo point](#) so the user can undo the

Contents [hide]

- 1 Progress reporting
 - 1.1 `aegisub.progress.set`
 - 1.2 `aegisub.progress.task`
 - 1.3 `aegisub.progress.title`
 - 1.4 `aegisub.progress.is_cancelled`
- 2 Debug output
 - 2.1 `aegisub.debug.out`

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

changes made by the script.

- `cancelled` (bool) - `false` if the user has not clicked the Cancel button, `true` if the user has clicked Cancel. If `is_cancelled` returns `true` all subsequent calls to it in the current execution will also return `true`.

Debug output

The primary support for script debugging in Automation 4 Lua is through sending debug messages to the message log integrated in the progress window.

If a script shows a debug or other message, the progress window stays open after the script has finished running, until the user clicks the Close button. Please consider whether it's really that important that the user sees your messages, blocking other input to the program to display something that might be irrelevant to the user can create a bad experience.

`aegisub.debug.out`

Synopsis:

- `aegisub.debug.out(msg, ...)`
- `aegisub.debug.out(level, msg, ...)`
- `aegisub.log(msg, ...)`
- `aegisub.log(level, msg, ...)`

The two names are synonymous, you can use either name depending on your preference.

Sends a message to the message log, optionally with a specific severity level. The user can control in Aegisub's options the highest level messages that will be shown.

- `level` (number) - Severity level of the message. This parameter is optional, if you leave it out (by entirely skipping it) the message will always be shown.
- `msg` (string) - A format string specifying the message, see the Lua standard string library `string.format` function for details on format strings.
- `...` - Parameters to the format string.

The following severity levels are suggested:

- 0 "fatal" - Something really bad happened and the script can't continue. Level 0 messages are always shown. Note that Aegisub won't terminate your script because you display a level 0 message, you will still have to do that yourself.
- 1 "error" - A real error occurred so the user should expect something to have gone wrong even though you tried to recover. A fatal error might happen later.
- 2 "warning" - It looks like something is wrong and the user ought to know because it might mean something needs to be fixed.
- 3 "hint" - A tip or otherwise on how the user can improve things, or hints that something might cause a warning or error later on.
- 4 "debug" - Information meant to help fix errors in the script, such as dumps of variable contents.
- 5 "trace" - Extremely verbose information about what the script is doing, literally a message for each single step done with lots of variable dumps.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater	Declaring templates • Execution order • Modifiers • Inline-variables (\$-

reference:	variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Category: [Lua Reference](#)

Automation 4 Lua Configuration dialogues

Todo: Wikify this... but save it for last because nobody really care too much about it, it's still buggy and all the docs really are here anyway.

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

```
Automation 4 Configuration Dialog interface
```

```
This file describes the functions and data structures used for the  
Configuration Dialog functionality in Automation 4.
```

```
---
```

```
Dialog Control table format
```

```
A Dialog Control table describes a single control in a configuration dialog,  
which can display information to the user and allow them to change it.
```

```
There are a number of different classes of controls, and the keys a Dialog  
Control table must contain depends on the control class.
```

```
Common keys for all control classes:
```

```
class (string)  
    Defines which class this control has. Must be one of:  
    "label",  
    "edit", "intedit", "floatedit", "textbox",  
    "dropdown",  
    "checkbox",  
    "color", "coloralpha", "alpha"
```

```
x (number)
```

```
y (number)
```

```
width (number)
```

```
height (number)
```

```
Determines the position and size of the control in the dialog. These values  
are used to create a grid containing the controls. They should all be  
integer. The top left corner is x,y=0,0.
```

```
If any of width and height are set to zero or less, it will be set to one  
instead.
```

```
Keys defined for all classes except "label":
```

```
hint (string)
```

```
A string displayed to the user as tooltip, when hovering over the control.
```

```
name (string)
```

```
A name that uniquely identifies the control. This is recommended to be a  
string easily used as an identifier in Lua, since it will be used to access  
the value input into the control.
```

```
Keys defined only for "label" and "checkbox" classes:
```

```
label (string)
```

The text displayed to the user on the control.

Key defined only for the "edit" and "textbox" classes:

text (string)

The contents of the control when the dialog is first displayed.
This can contain newlines if the control is of the "textbox" class.

Keys defined only for the "intedit" and "floatedit" classes:

value (number)

The value in the control when the dialog is first displayed. For the "intedit" class, if this is a non-integer number it will be truncated.

min (number or nil)

max (number or nil)

If one of these are nil, the other must also be nil. (Ie. undefined.)
If both are present, the control gets a spin button, the user can click to update the value of the control. The user won't be able to close the dialog if the value is outside the range between "min" and "max".

Key defined only for the "floatedit" class:

step (number or nil)

If nil/undefined, the value 1 (one) is used instead.
Specifies the size of change when the spin buttons are clicked.

Keys defined only for the "dropdown" class:

items (table)

This is an Array Table containing only strings. They are used for the options displayed to the user in the dropdown box.
All strings in the array table should be unique. (There is not way to distinguish non-unique strings from each other.)

value (string)

Determines which item is selected when the dialog id first displayed. If this is not one of the items specified, no item is selected. This is case-sensitive.

Key defined only for the "checkbox" class:

value (boolean)

Determines whether the checkbox is checked or not when the dialog is first displayed.

Keys defined only for the "color", "coloralpha" and "alpha" classes:

value (string)

A color value in VB or HTML hexadecimal format.
For the "color" class, this should be a 3 byte value, ie. "#RRGGBB".
For the "coloralpha" class, this should be a 4 byte value, ie. "#RRGGBBAA".
For the "alpha" class, this should be a one-byte value, ie. "#AA".

```

---
Dialog Definition table format

The Dialog Definition table is simply an Array Table of Dialog Control tables.

Note, however, that while the visual ordering of the controls are decided
entirely by the "x", "y", "width" and "height" of the controls, the
"tab order" of the controls is decided by their ordering in the Dialog
Definition table.

---

Dialog Result table format

A Dialog Result table contains the user input from a configuration dialog.

The control "name" properties are used as keys in this table.

The type of the value for each entry in the table depends on the class of the
control. The control classes map to types in the following manner:

"label"
  None. Since the user cannot change a label, they do not produce any value.

"edit", "textbox"
  String. The text input in the box. This can contain newlines in the case of
  a "textbox" class control.

"intedit", "floatedit"
  Number. The number input into the control, guaranteed to be within the
  constraints set by the class (integer or float) and the min/max properties.

"dropdown"
  String. The case-exact text of the selected item.

"checkbox",
  Boolean. The checked-state of the checkbox.

"color", "coloralpha", "alpha"
  String. A VB colorstring following the same scheme as for setting the
  "value" property.

---

Display Configuration Dialog function

This function displays a configuration dialog to the user and waits for it to
close. It then returns whether the user accepted or cancelled the dialog, and
what values were input.

function aegisub.dialog.display(dialog, buttons)

@dialog (table)
  A Dialog Definition table containing the controls to be in the dialog.

@buttons (table)
  Optional. This is an Array Table of strings defining the buttons that appear
  in the dialog. If this is left out, empty or is otherwise not a table, the
  standard Ok and Cancel buttons appear.
  The strings in this Array Table are used as labels on the buttons, and for

```

identifying them in the return values of the function.

Returns: Two values.

1. Boolean or string.

If no custom buttons were specified, this is a boolean telling whether Ok (true) or Cancel (false) were clicked in the dialog.

If custom buttons were specified, this is the text on the button clicked by the user.

Even if custom buttons were specified, this can still be boolean false if the user closes the dialog without pressing any button.

2. Table.

The Dialog Result table corresponding to the values the user input in the dialog.

Automation 4

Overview:

[Automation Manager](#) • [Running macros](#) • [Using export filters](#) • [Standard macros](#) • [Changes from Automation 3](#) • [Moving from Automation 3](#)

[Karaoke Templater](#) reference:

[Declaring templates](#) • [Execution order](#) • [Modifiers](#) • [Inline-variables \(\\$-variables\)](#) • [Code lines and blocks](#) • [Execution environment](#)

[Lua](#) reference:

[Registration](#) • [Subtitles object](#) • [Progress reporting](#) • [Config dialogues](#) • [Misc. APIs](#) • [karaskel.lua](#) • [utils.lua](#) • [unicode.lua](#) • [cleantags.lua](#)

[Karaskel](#) concepts:

[Style tables](#) • [Dialogue line tables](#) • [Syllable tables](#) • [Inline effects](#) • [Furigana](#)

Categories: [Pages with Todo items](#) | [Lua Reference](#)

Automation 4 Lua Miscellaneous APIs

This page documents **miscellaneous API's useful for working with subtitles**, these can't be clearly placed into any of the other main categories and there's too few of each kind to warrant a separate category.

aegisub.text_extents

Synopsis: `width, height, descent, ext_lead = aegisub.text_extents(style, text)`

Obtain system font metrics and determine the rendered size in pixels of the given **text** when using the **style**.

- **width** (number) - Width in pixels of text. This may be non-integer.
- **height** (number) - Height in pixels of the text. This may be non-integer.
- **descent** (number) - Length of descenders in the font. This may be non-integer.
- **ext_lead** (number) - External leading for the font. This may be non-integer.
- **style** (table) - A [style table](#) as defined by the subtitle interface. The font name, size, weight, style, spacing and encoding is used to determine the size of the text.
- **text** (string) - The text the extents should be determined for. This should not contain linebreaks (`\n` or `\r\n`) nor should it contain formatting codes of any kind. Formatting codes are not attempted interpreted and will be taken as verbatim text.

You should only feed plain text strings without line breaks into this function, it cannot handle any kind of formatting codes or text layout. Rather, it is intended as a helper to create text layouts by determining rendered sizes of bits and pieces of a longer text, which can then be layouted by the script.

Getting information on the video

Automation 4 Lua offers two functions designed to be able to work with frame-based timing without having to consider whether the video source is VFR or CFR.

The primary purpose of these functions is to be able to generate per-frame effects, ie. get the timestamps of a number of sequential frames and calculate coordinates, sizes etc. for an object for each of those frames.

One thing to remember when using these functions is that, considering a one-dimensional time line, a time stamp is a point on the time line, while a video frame spans a range of the time line, from its beginning time to its ending time. The ending time of a frame is the beginning time of the next. The beginning time of a frame is included in the range while the ending time is excluded from the range.

aegisub.frame_from_ms

Synopsis: `frame = aegisub.frame_from_ms(ms)`

Use loaded frame rate data to convert an absolute time given in **milliseconds** into a **frame number**.

Contents [hide]

- 1 aegisub.text_extents
- 2 Getting information on the video
 - 2.1 aegisub.frame_from_ms
 - 2.2 aegisub.ms_from_frame
 - 2.3 aegisub.video_size

Navigation

» [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- **frame** (number) - Frame number corresponding to the time in ms.
- **ms** (number) - Absolute time from the beginning of the video, for which to determine the frame number.

If the time is in the middle of the frame it is "rounded down" to the frame number that contains the given time.

aegisub.ms_from_frame

Synopsis: `ms = aegisub.ms_from_frame(frame)`

Use loaded frame rate data to convert a **frame** number of the video into an absolute time in **milliseconds**.

- **ms** (number) - First integer millisecond time stamp to lie within the frame.
- **frame** (number) - Frame to obtain the beginning time of.

Because beginning times of frames can have better precision than one millisecond this function rounds up and returns the first whole millisecond that is guaranteed to be within the frame.

aegisub.video_size

Synopsis: `xres, yres, ar, artype = aegisub.video_size()`

Get information about the resolution and aspect-ratio of the loaded video, if any.

Returns `nil` if there is no video loaded.

- **xres** and **yres** (numbers) - Coded width and height of the video in pixels.
- **ar** (number) - Custom display aspect ratio override.
- **artype** (number) - See below.

There are 5 values that *artype* can take:

- If *artype* is 0, the video has square pixels, ie. PAR is 1.00 or DAR is *xres/yres*.
- If *artype* is 1, the video is 4:3, ie. DAR is 1.33.
- If *artype* is 2, the video is 16:9, ie. DAR is 1.78
- If *artype* is 3, the video is 2.35 format, ie. DAR is 2.35.
- If *artype* is 4, the DAR is whatever the *ar* return value contains.

You cannot count on *ar* always having the correct value when *artype* is different from 4.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Category: [Lua Reference](#)

Automation 4 karaskel.lua

The Automation 4 `karaskel.lua` include file contains several functions intended to help the development of karaoke effects with Automation 4 Lua. It also defines several new data structures, and extensions to those defined by Automation 4 Lua itself.

`karaskel.lua` itself includes `utils.lua` and `unicode.lua` so you do not need to include those yourself when using `karaskel.lua`.

Using `karaskel.lua` is strongly recommended when creating karaoke effects, and it can also be useful for other tasks as it contains several text layouting functions.

Functions

`karaskel.collect_head`

Synopsis: `meta, styles =`

```
karaskel.collect_head(subtitles, generate_furigana)
```

Reads the subtitle file to collect all header information and style definitions, and optionally also generates new styles for furigana layouts.

- `subtitles` is the Subtitle File object defined by Automation 4 Lua.
- `generate_furigana` is a boolean, if it is true a style for [furigana layout](#) is generated for each style that does not have one already. Generation of furigana styles will never overwrite existing styles, create double style definitions or create meaningless furigana styles for other furigana styles.

Calling `collect_head` is usually one of the first things you do in your processing function.

The returned `meta` table contains a map of all `Name: Value` pairs in the `[Script Info]` section. It also always contains `meta.res_x` and `meta.res_y` calculated from the `PlayResX` and `PlayResY` fields, following VSFILTER conventions for default values when one or both of the fields are missing.

The returned `styles` table contains a map of all defined styles, it also include any generated furigana layout styles. The style structures stored in this table have one added field, `style.margin_v` which is an alias for `style.margin_t`, for convenience. `styles` can be indexed by style names (case sensitive, names not mangled) and by numbers. `styles.n` is the number of styles stored, and `styles[1]` is the first style defined.

Contents [hide]

1 Functions

- 1.1 `karaskel.collect_head`
- 1.2 `karaskel.preproc_line`
- 1.3 `karaskel.preproc_line_text`
- 1.4 `karaskel.preproc_line_size`
- 1.5 `karaskel.preproc_line_pos`
- 1.6 `karaskel.do_basic_layout`
- 1.7 `karaskel.do_furigana_layout`

2 Karaoke skeletons

- 2.1 Effect Library
- 2.2 Classic Advanced

3 Data structures

- 3.1 Styles array
- 3.2 Style table
- 3.3 Dialogue line table
- 3.4 Karaoke and furigana syllable tables
 - 3.4.1 Highlight table

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

karaskel.preproc_line

Synopsis: `karaskel.preproc_line(subtitles, meta, styles, line)`

Calculate sizing, positioning and various other information for a single subtitle line. This function calls `karaskel.preproc_line_text`, `karaskel.preproc_line_size` and `karaskel.preproc_line_pos` in order.

Note that the interface for the three functions used to do the work is not frozen, it might change, though it probably won't.

The interface for this function is frozen and will not change. It is recommended to call this function to pre-process a line.

This function does not return a value, but rather modifies the `line` table, see below for more information.

karaskel.preproc_line_text

Synopsis: `karaskel.preproc_line_text(meta, styles, line)`

Preprocess the text of a single line. `meta` and `styles` are the tables returned by [karaskel.collect_head](#).

This function does not return a value, but rather modifies the `line` table. The following fields are added:

- `line.text_stripped` - Line text with all override tags and vector drawings removed.
- `line.duration` - Duration of the line in milliseconds
- `line.kara` and `line.furi` - Extended [karaoke and furigana tables](#), without sizing and position data.

This function does not calculate any text sizing or positioning information. (In fact it doesn't use the `meta` or `styles` arguments at all.)

karaskel.preproc_line_size

Synopsis: `karaskel.preproc_line_size(meta, styles, line)`

Calculate sizing data for a line and all karaoke syllables and furigana parts. Also adds a reference to the line style.

This function does not return a value, but rather modifies the `line` table. The following fields are added:

- `line.styleref` - A reference to the Style table representing this line's selected style.
- `line.furistyle` - A reference to the Style table representing this line's furigana layout style. If there is no style with the right name, this field is `false` instead.
- `line.width`, `line.height`, `line.descent` and `line.extlead` - Sizing information for the stripped line text, as returned by [aegisub.text_extents](#).

Also, this function modifies the `line.kara` and `line.furi` tables, adding sizing information.

No position information is calculated here.

If the line table does not seem to have been processed with `karaskel.preproc_line_text` yet, this will be done automatically.

`karaskel.preproc_line_pos`

Synopsis: `karaskel.preproc_line_pos(meta, styles, line)`

Calculate line, karaoke and furigana position information.

This function invokes `karaskel.do_basic_layout` when no furigana style is available, and `karaskel.do_furigana_layout` when a furigana style is defined for the line. The furigana layout algorithm might change the calculated width of the line.

This function does not return a value, but rather modifies the `line` table. The following fields are added:

- `line.margin_v` - A convenience alias for `line.margin_t`.
- `line.eff_margin_l`, `line.eff_margin_r`, `line.eff_margin_t`, `line.eff_margin_b` and `line.eff_margin_v` - Effective margin values for the line. If the corresponding margin override for the line is non-zero, that value is used, otherwise the value defined in the style is used.
- `line.halign` - One of "left", "center" or "right", the horizontal alignment of the line, derived from `line.styleref.align`.
- `line.valign` - One of "top", "middle" or "bottom", the vertical alignment of the line, derived from `line.styleref.align`.
- `line.left` - The left edge X coordinate for the line, assuming its given alignment, effective margins and no collision detection
- `line.center` - The line centre X coordinate, assuming its given alignment, effective margins and no collision detection
- `line.right` - The right edge X coordinate for the line, assuming its given alignment, effective margins and no collision detection
- `line.top` - The top edge Y coordinate for the line, assuming its given alignment, effective margins and no collision detection
- `line.middle` - The line vertical centre Y coordinate, assuming its given alignment, effective margins and no collision detection `line.vcenter` is an alias for this.
- `line.bottom` - The bottom edge Y coordinate for the line, assuming its given alignment, effective margins and no collision detection.
- `line.x` and `line.y` - X and Y coordinates for the line, suitable for using in a `\pos` override tag to get the line's original position.

Furthermore, the `line.kara` and `line.furi` tables are modified by the layout function called, adding positioning information.

See the part on [data structures](#) later on this page for more details on the various fields that are added.

If no line sizing information is found, `karaskel.preproc_line_size` will be invoked, which might in turn also invoke `karaskel.preproc_line_text`.

`karaskel.do_basic_layout`

This function is not intended to be called directly, but is rather called as a helper function for `karaskel.preproc_line_pos`.

It runs a very simple layout algorithm for the `line.kara` table, which simply calculates the positions of the syllables when placed in one straight line with no additional spacing in between. Positioning information is added to each karaoke syllable.

The `line.furi` table is not touched.

karaskel.do_furigana_layout

This function is not intended to be called directly, but is rather called as a helper function for `karaskel.preproc_line_pos`.

It runs an advanced text layout algorithm to position karaoke syllables and furigana neatly, avoiding unwanted overlapping. People interested in the actual algorithm used should read the function source code, it should be well enough commented.

This function adds positioning information to both the `line.kara` and `line.furi` tables. It might also change the `line.width` field as the line base text is expanded to make room for furigana.

Karaoke skeletons

A karaoke skeleton is a framework for building karaoke effects in. It usually works by writing a couple of functions yourself for handling the actual effect work, and these are then called at various times. The actual details of what functions you need to write depends on the actual karaoke skeleton.

Effect Library

Main function: `karaskel.use_fx_library_furi(use_furigana, add_macro)`

Call the `karaskel.use_fx_library_furi` function to install the Effect Library skeleton for this script file. The `script_name` and `script_description` globals are used to name the export filter produced. If `use_furigana` is true, furigana styles are created and added as needed. If `add_macro` is true, a macro is registered in addition to the export filter.

The basic premise of the Effect Library skeleton is that each timed karaoke line has a word in its Effect field that describes what effect to apply to that line. This makes Effect Library a good choice if you want to use several different effects in a single karaoke.

When Effect Library is invoked, it calls a function named `fx_effect` for each Dialogue line in the subtitle file. For example, if the Effect field of a dialogue line is `"jump"`, the function named `fx_jump` is called. For lines with empty Effect field, the function `fx_none` is called.

If an fx function does not exist, the original line is left in the subtitle file. Otherwise, whether the original line is left depends on the return value of the fx function, a true return value means the original line is kept, a false value means it is made into a Comment line.

Signature of fx functions: `keep = fx_effect(subtitles, meta, styles, line, fxdata)`

`fxdata` is the contents of the Effect field after the initial word defining the effect to be used. All output of an fx function should be appended to the subtitle file represented by `subtitles`.

Simplified main function: `karaskel.use_fx_library(add_macro)`

Identical to the `_furi` variant above, except that the `use_furigana` parameter is removed; it is assumed to be false.

Classic Advanced

Main function: `karaskel.use_classic_adv(use_furigana, add_macro)`

Call the `karaskel.use_classic_adv` function to install the Classic Advanced skeleton for this script file. The `script_name` and `script_description` globals are used to name the export filter produced. If `use_furigana` is true, furigana styles are created and added as needed, and furigana processing is enabled. If `add_macro` is true, a macro is registered in addition to the export filter.

This skeleton is created in the image of the Automation 3 `karaskel-adv` skeleton, but it is *not* compatible with it. (You cannot use a `karaskel-adv` script with Classic Advanced without rewriting parts of your script.) The basic premise is that a function is called once for each syllable, this is the `do_syllable` function. Optionally, you can have a function called for each line, using the `do_line` function.

Classic Advanced uses a slightly different model than the usual Automation 4 Lua one, here all subtitle lines are collected first before any further processing is done. They also have `line.prev` and `line.next` fields added, to allow linked list style access. To add lines to the output, you must still add lines to the `subs` object though. Before processing starts, all original lines are *deleted* from the `subs` object.

Signature of syllable function: `do_syllable(subs, meta, styles, lines, line, syl)`

The syllable function *must* be named `do_syllable`. If furigana processing is enabled, you can also define a function called `do_furigana` with the same signature, to process furigana syllables. Furigana still follows the Automation 4 model here.

Signature of line function: `do_line(subs, meta, styles, lines, line, default_do_line)`

Defining a line function is optional, and is often not required. The line function *must* be named `do_line` if it exists. The `default_do_line` parameter is the function that would be called if `do_line` didn't exist, you can call it to run the default line processing along with your own processing.

Data structures

`karaskel.lua` defines and extends several data structures. Some of the changes are already listed above under the individual functions.

Styles array

The `styles` array is produced by the `karaskel.collect_head` function and should be passed to most other `karaskel.lua` functions. It contains a list of all styles in the subtitle file, and can be accessed in two ways.

`styles.n` is a number telling the number of styles in the array. `styles[1]` is the first defined style and `styles[styles.n]` is the last defined style.

The `styles` array can also be indexed by style names, such that `styles[style.name] == style`. The names are not mangled and the indexing is case sensitive.

Be aware that modifying the `styles` will never update the subtitles file, and conversely updating the styles in the subtitle file will not automatically be reflected in `styles` either.

Style table

This is a slight extension of the basic `style` class subtitle line structure.

One field is added:

- `style.margin_v` is a convenience alias for `style.margin_t`.

Full list of fields:

- `style.class == "style"`
- `style.raw` - The raw line text.
- `style.section == "[V4+ Styles]"`
- `style.name` - Name of the style.
- `style.fontname` - Name of the font face used by the style.
- `style.fontsize` - Font size for the style.
- `style.color1`, `style.color2`, `style.color3` and `style.color4` - The four colours used by the style, in regular order. Use `extract_color` and `family` to manipulate these.
- `style.bold` - `true/false` to specify bold/non-bold font face. Can also be a number to specify font weight, but this is not well supported and should be avoided.
- `style.italic` - Boolean, whether an italic/oblique version of the font face is used or not.
- `style.underline` and `style.strikeout` - Boolean, whether to apply these two decorations to the text.
- `style.scale_x` and `style.scale_y` - Scaling in X and Y direction, 100 is neutral.
- `style.spacing` - Additional spacing in pixels between individual characters in text.
- `style.angle` - Z axis rotation for the text.
- `style.borderstyle` - 1 (one) for regular outlined text, 3 for opaque box behind subtitles.
- `style.outline` - Width of the extended outline around the text.
- `style.shadow` - Distance to the shadow behind the text.
- `style.align` - Numpad-style alignment for the text on screen.
- `style.margin_l`, `style.margin_r`, `style.margin_t` and `style.margin_b` - Margins for the style. `style.margin_v` is an alias for top margin.
- `style.encoding` - Windows font encoding ID for the style.
- `style.relative_to` - Currently unsupported.
- `style.vertical` - Unsupported, tentative AS5 feature.

Dialogue line table

A large number of new fields have been added to the dialogue line class.

Basic fields:

- `line.class == "dialogue"`, also for comment lines
- `line.raw` - The raw line text.
- `line.section` - Usually `"[Events]"`.
- `line.comment` - Boolean, true if the line is a Comment line rather than Dialogue.
- `line.layer` - Layer of the line.

- `line.start_time`, `line.end_time` - Start and end times of the line in milliseconds.
- `line.style` - Name of the style used for the line.
- `line.actor` - Actor field for the line.
- `line.margin_l`, `line.margin_r`, `line.margin_t` and `line.margin_b` - Margin overrides for the line, a zero value means use margin from style.
- `line.effect` - Effect field of the line.
- `line.userdata` - Unused, tentative AS5 field.
- `line.text` - Dialogue text.

Basic added fields, by `karaskel.preproc_line_text`:

- `line.text_stripped` - Line text with all override tags and vector drawings removed.
- `line.duration` - Duration of the line in milliseconds
- `line.kara` and `line.furi` - Array tables of extended karaoke and furigana tables, respectively. They do not contain sizing and positioning data from the beginning.

Added fields for sizing, by `karaskel.preproc_line_size`:

- `line.styleref` - A reference to the Style table representing this line's selected style.
- `line.furistyle` - A reference to the Style table representing this line's furigana layout style. If there is no style with the right name, this field is `false` instead.
- `line.width`, `line.height`, `line.descent` and `line.extlead` - Sizing information for the stripped line text, as returned by `aegisub.text_extents`. `line.width` may also be modified by `karaskel.preproc_line_pos`.

Added fields for positioning, by `karaskel.preproc_line_pos`:

- `line.margin_v` - A convenience alias for `line.margin_t`.
- `line.eff_margin_l`, `line.eff_margin_r`, `line.eff_margin_t`, `line.eff_margin_b` and `line.eff_margin_v` - Effective margin values for the line. If the corresponding margin override for the line is non-zero, that value is used, otherwise the value defined in the style is used.
- `line.halign` - One of `"left"`, `"center"` or `"right"`, the horizontal alignment of the line, derived from `line.styleref.align`.
- `line.valign` - One of `"top"`, `"middle"` or `"bottom"`, the vertical alignment of the line, derived from `line.styleref.align`.
- `line.left` - The left edge X coordinate for the line, assuming its given alignment, effective margins and no collision detection
- `line.center` - The line centre X coordinate, assuming its given alignment, effective margins and no collision detection
- `line.right` - The right edge X coordinate for the line, assuming its given alignment, effective margins and no collision detection
- `line.top` - The top edge Y coordinate for the line, assuming its given alignment, effective margins and no collision detection
- `line.middle` - The line vertical centre Y coordinate, assuming its given alignment, effective margins and no collision detection `line.vcenter` is an alias for this.
- `line.bottom` - The bottom edge Y coordinate for the line, assuming its given alignment, effective margins and no collision detection.
- `line.x` and `line.y` - X and Y coordinates for the line, suitable for using in a `\pos` override tag to get the line's original position.

Added fields for linked list access, only available when using the Classic Advanced skeleton:

- `line.prev`, `line.next` - Access the dialogue line before and after this one.

These might be `nil` on the first/last dialogue lines. Blank lines, style lines, header lines etc. are *not* included in this linked list.

Karaoke and furigana syllable tables

Tables for regular karaoke syllables and furigana parts are identical in (almost) every aspect, and can usually be processed by the same code without problems. There are a few points to take note of, they will be marked. Everywhere it says `syl` here, you can replace that with `furi` unless otherwise noted.

Basic fields, defined by `aegisub.parse_karaoke_data`:

- `syl.duration` - syllable duration in milliseconds (divide by 10 to get a number suitable for `\k` tags.)
- `syl.start_time`, `syl.end_time` - Start and end time of the syllable in milliseconds, relative to the start time of the line.
- `syl.tag` - The name of the tag defining this syllable, without backslash. It will usually be one of `k`, `K`, `kf` or `ko`. Note that `kt` is not handled. Furigana parts have the same tag as the original syllable defining them.
- `syl.text` - Text including tags of the syllable. Same as stripped text for furigana.
- `syl.text_stripped` - Text of the syllable with all tags removed. For main syllables, this also has furigana and multi-highlight parts removed. This is the text you will usually want to use.

Additions by `karaskel.preproc_line_text`:

- `syl.kdur` - Syllable duration in centiseconds, suitable for use in `\k` tags.
- `syl.line` - Back reference to the line table containing this syllable.
- `syl.inline_fx` - Name of the *inline-fx* for this syllable.
- `syl.i` - Index number of this syllable.
- `syl.prespacer`, `syl.postspacer` - Space characters at the start/end of the syllable. Always blank for furigana. These are spaces included in `syl.text_stripped`. You will usually never need this.
- `syl.text_spacestripped` - Syllable text stripped for tags and trimmed of spaces at the start and end. This, `syl.prespacer` and `syl.postspacer` together can produce the same as `syl.text_stripped`. You will usually never need this.
- `syl.isfuri` - `true` if the table is a furigana table, `false` if it is not. If you use a single function to process both regular and furigana syllables, you can use this to do differentiated processing still.
- `syl.highlights` - Array table of multi-highlight data for the syllable. For furigana, there is always exactly one highlight defined. See below for format of highlight tables.

Additions by `karaskel.preproc_line_size`:

- `syl.style` - Reference to the style used to calculate sizing for this syllable. This will be the main line style for regular syllables and the furigana style for furigana. You should always set the style of the generated lines to this one.
- `syl.width`, `syl.height` - Width and height of `syl.text_spacestripped`, as returned by `aegisub.text_extents`.
- `syl.prespacerwidth`, `syl.postspacerwidth` - Width of `syl.prespacer` and `syl.postspacer` respectively. You will usually not need these. Always zero for furigana.

Additions by `karaskel.preproc_line_pos`:

- `syl.left`, `syl.center`, `syl.right` - Respectively left, center and right aligned positions of the syllable/furigana, for use with different alignments. The

positions are *relative to the left edge of the line*, meaning you will need to add a value for line positioning to use these values to position syllables on screen. There is no guarantee that `syl.right` for one syllable is equal to `syl.left` for the next syllable.

Example

```
line.left + syl.center
```

Calculates the default X position of a syllable, suitable for use with `\an2`, `\an5` or `\an8` alignment.

Highlight table

A highlight table defines one highlight of a multi-highlight timed syllable.

Highlight tables are entirely defined by `karaskel.preproc_line_text`, and contain the following fields:

- `hl.start_time`, `hl.end_time` - Start and end time of the highlight, in milliseconds, relative to the start of the line.
- `hl.duration` - Duration of the highlight in milliseconds.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Category: [Lua Reference](#)

Automation 4 utils.lua

The Automation 4 Lua include file `utils.lua` contains various support functions to aid writing Lua scripts. There is no general theme for the file.

Table functions

Duplicating tables in various ways is a common task. `utils.lua` provides some functions to handle the most common cases.

table.copy

Synopsis: `newtable = table.copy(oldtable)`

Makes a shallow copy of the table passed as parameter. Shallow here means that it does not dive into contained tables and copy those as well, so if `oldtable.st` refers to a table, `newtable.st` will refer to the same table, such that modifying `newtable.st` will also be reflected in `oldtable.st` and vice versa.

copy_line

Alias for `table.copy`. Included to help transitioning from Automation 3.

table.copy_deep

Synopsis: `newtable = table.copy_deep(oldtable)`

Makes a deep copy of the table passed as parameter. While this function attempts resolving circular references and not do infinite recursion on them, it might not work in all cases. You will rarely need to use this function; if you think you need to do a deep copy, consider your task an extra time.

Warning: This function has not been heavily tested.

Colour functions

It is often needed to do various transformations on colour data. Several functions for this is included.

ass_color

Synopsis: `colorstring = ass_color(r, g, b)`

Makes an ASS colour string in the form `&HBBGGRR` from the given `r`, `g` and `b` arguments.

Contents [\[hide\]](#)

- 1 Table functions
 - 1.1 table.copy
 - 1.2 copy_line
 - 1.3 table.copy_deep
- 2 Colour functions
 - 2.1 ass_color
 - 2.2 ass_alpha
 - 2.3 ass_style_color
 - 2.4 extract_color
 - 2.5 alpha_from_style
 - 2.6 color_from_style
 - 2.7 HSV_to_RGB
- 3 String functions
 - 3.1 string.trim
 - 3.2 string.headtail
 - 3.3 string.words
- 4 Numeric functions
 - 4.1 clamp
 - 4.2 interpolate
 - 4.3 interpolate_color
 - 4.4 interpolate_alpha

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Warning: The arguments are not checked for range, values outside the 0..255 range will produce garbage output.

ass_alpha

Synopsis: `alphastring = ass_alpha(a)`

Makes an ASS alpha string in the form `&HAA&` from the given `a` argument.

Does not check input range.

ass_style_color

Synopsis: `colorstring = ass_style_color(r, g, b, a)`

Makes an ASS colour string suitable for use in Style definitions, ie. in format `&HAABBGRR.`

Does not check input range.

extract_color

Synopsis: `r, g, b, a = extract_color(colorstring)`

Extracts colour components from a colour string. Several formats of colour strings are recognised:

- Style definition: `&HAABBGRR`
- Inline override: `&HBBGRR&`
- Alpha override: `&HAA&`
- HTML with alpha: `#RRGGBBAA`

Note that this function always returns four numbers when passed a valid colour string. Unused values (depends on the format of the colour string) are assigned 0 (zero.) If an unrecognised colour string is passed, `nil` is returned.

Example: After executing `r, g, b, a = extract_color("&H7F&")`, the `r` `g` `b` variables will be 0 and the `a` variable will be 127.

alpha_from_style

Synopsis: `alphastring = alpha_from_style(coloralphastring)`

Returns the alpha part of a colour string, as a an alpha override string, ie. `&HAA&` format. This function is a composition of `extract_color` and `ass_alpha`.

color_from_style

Synopsis: `colorstring = color_from_style(coloralphastring)`

Returns the colour part of a colour string, as a colour override string, ie. `&HBBGRR&` format. This function is a composition of `extract_color` and `ass_color`.

HSV_to_RGB

Synopsis: `r, g, b = HSV_to_RGB(h, s, v)`

Transforms a colour given in Hue, Saturation, Value space into Red, Green, Blue space.

`h` is given in degrees, the nominal range is 0..359; values outside this range will be translated into it. Input range of `s` and `v` are 0..1, these are not range checked. Output range of `r`, `g` and `b` are 0..255.

String functions

The Lua standard `string` library is sometimes not powerful enough. A few additional helper functions are provided. See also [Automation 4 unicode.lua](#).

`string.trim`

Synopsis: `outstring = string.trim(instring)`

Removes all space characters at the start and end of the input string, and returns the transformed string.

Warning: This function is not UTF-8 safe. It uses the Lua regex `%s` class to match spaces, which in some legacy encodings will result in it also matching some prefix bytes in UTF-8 encoded text.

`string.headtail`

Synopsis: `head, tail = string.headtail(instring)`

Splits a string by first space-sequence into a "head" and a "tail", similar to the handling of linked lists in several functional languages.

If `instring` does not contain any space characters it returns `instring, ""`.

`string.words`

Synopsis: `for word in string.words(instring) do ... end`

Returns an iterator function for use in a `for` loop, to loop over all the words in the string using `string.headtail` semantics.

Numeric functions

Functions to handle various operations on numbers.

`clamp`

Synopsis: `outval = clamp(ival, min, max)`

Clamps `ival` to be in range `min..max`.

interpolate

Synopsis: `outval = interpolate(t, a, b)`

Interpolates between a and b. t is the time variable in range 0..1, values outside this range are clamped.

interpolate_color

Synopsis: `outcolor = interpolate_color(t, color1, color2)`

Interpolate between color1 and color2 with t as time variable in range 0..1. color1, color2 and outcolor are colour strings, outcolour will be in colour override format.

interpolate_alpha

Synopsis: `outalpha = interpolate_alpha(t, alpha1, alpha2)`

Similar to `interpolate_color`, but interpolates alpha values instead. Also works on colour strings, and will return an alpha override string.

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment
Lua reference:	Registration • Subtitles object • Progress reporting • Config dialogues • Misc. APIs • karaskel.lua • utils.lua • unicode.lua • cleantags.lua
Karaskel concepts:	Style tables • Dialogue line tables • Syllable tables • Inline effects • Furigana

Category: [Lua Reference](#)

Automation 4 unicode.lua

The `unicode.lua` include for Automation 4 Lua contains various helper functions for working with UTF-8 encoded text.

Contents [hide]

- 1 [unicode.charwidth](#)
- 2 [unicode.chars](#)
- 3 [unicode.len](#)
- 4 [unicode.codepoint](#)

unicode.charwidth

Synopsis: `width = unicode.charwidth(instring, index)`

Returns the number of bytes occupied by the UTF-8 encoded character starting at position `index` in `instring`. The character pointed to is assumed to be a prefix byte.

The `index` parameter is optional, it defaults to 1 (one) when left out, meaning the width of the first character in `instring` will be returned.

unicode.chars

Synopsis: `for char = unicode.chars(instring) do ... end`

Returns an iterator function for looping over all characters in the given UTF-8 encoded string. For each iteration of the loop, `char` will contain a string representing the next character in the string.

unicode.len

Synopsis: `length = unicode.len(instring)`

Determine the length of the given UTF-8 encoded string.

Be aware that this function does not run in constant time, but in linear time ($O(N)$) proportional to the number of Unicode characters in `instring`.

unicode.codepoint

Synopsis: `val = unicode.codepoint(instring)`

Calculate the Unicode codepoint for the first character in `instring`.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Automation 4

Overview: [Automation Manager](#) • [Running macros](#) • [Using export filters](#) • [Standard macros](#) • [Changes from Automation 3](#) • [Moving from Automation 3](#)

Karaoke Templater reference: [Declaring templates](#) • [Execution order](#) • [Modifiers](#) • [Inline-variables \(\\$-variables\)](#) • [Code lines and blocks](#) • [Execution environment](#)

Lua reference: [Registration](#) • [Subtitles object](#) • [Progress reporting](#) • [Config dialogues](#) • [Misc. APIs](#) • [karaskel.lua](#) • [utils.lua](#) • [unicode.lua](#) • [cleantags.lua](#)

Karaskel [Style tables](#) • [Dialogue line tables](#) • [Syllable tables](#) • [Inline effects](#) •

concepts:

Furigana

Category: [Lua Reference](#)

Karaoke inline-fx

Karaoke inline-fx (inline effects) is a way of marking up [timed karaoke](#) to assign different effects to different parts of a line.

By itself, inline-fx markup doesn't do anything, it only has an effect when a [karaoke effect script](#) that understands it is applied to the timed karaoke.

The markup

Inline-fx tags are (otherwise invalid) ASS override tags of the form `\-effectname`, where *effectname* is the name of the inline-fx defined.

An inline-fx tag affects the syllable it is placed in and every following syllable, until the next syllable with an inline-fx tag in it.

At the start of each line the inline-fx is reset to nothing.

Example

Here is a timed karaoke line with inline-fx markup:

```
{\k40}zu{\k20}t{\k42}to {\k32}\-paint}e{\k17}ga{\k45}i{\k32}te{\k26}ta  
{\k24}\-cloud}yu{\k55}me
```

These syllables get inline-fx assigned like this:

Syllable Inline-fx

zu	(blank)
t	(blank)
to	(blank)
e	paint
ga	paint
i	paint
te	paint
ta	paint
yu	cloud
me	cloud

Usage in Karaoke Templater

If you use [Karaoke Templater](#) to create effects, you can use the *fx* modifier on templates to make that template affect only syllables with a specific inline-fx. It isn't possible (directly) to match only syllables with blank inline-fx.

Example

With the sample timed karaoke from above, you could have the following templates:

```
template syl: {base effect applied for all syllables}
```

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

```
template syl fx_paint: {overlay effect applied only to the 'paint'  
syllables}  
template syl fx_cloud: {overlay effect applied only to the 'cloud'  
syllables}
```

The idea here is to have a base effect and then some of the syllables get some more effects on top of that.

Example

It is possible to match only syllables with blank inline-fx in kara-templater by using an *fxgroup* that enables or disables basing on inline-fx. You can also use *fxgroups* to have templates that run for multiple inline-fx.

```
code_syl: fxgroup.blankfx = (syl.inline_fx == "")  
template syl fxgroup_blankfx: {effect only applied on blank inline-fx  
syllables}
```

The important thing is that the code line runs per syllable and runs before any per-syllable templates that must use it.

Usage in Lua scripts

The inline-fx tags are parsed by `karaskel.preproc_line_text` so they will only work if you have applied at least that much karaskel pre-processing on your subtitle lines.

The inline-fx for a syllable is then available as `syl.inline_fx`, you can compare that to a string to conditionally apply effects.

Example

In some code that runs per-syllable in your script:

```
if syl.inline_fx == "" then  
    apply_base_effect(subs, meta, line, syl)  
elseif syl.inline_fx == "paint" then  
    apply_paint_effect(subs, meta, line, syl)  
elseif syl.inline_fx == "cloud" then  
    apply_cloud_effect(subs, meta, line, syl)  
end
```

Simply compare the inline-fx name to the various possibilities and run the right effect code.

Example

At top-level of your script:

```
effects = {}  
effects[""] = function(subs, meta, line, syl)  
    -- base effect code here  
end  
effects.paint = function(subs, meta, line, syl)  
    -- paint effect code here  
end
```

```
effects.cloud = function(subs, meta, line, syl)
  -- cloud effect code here
end
```

Then later, in some per-syllable processing code:

```
effects[syl.inline_fx](subs, meta, line, syl)
```

First, a table is created and filled with functions for applying the different effects. The keys used for the table are the names of the possible inline-fx. When the effect has to be applied, the right function is looked up in the effect table and then called.

Automation 4

Overview:

[Automation Manager](#) • [Running macros](#) • [Using export filters](#) • [Standard macros](#) • [Changes from Automation 3](#) • [Moving from Automation 3](#)

[Karaoke Templater](#) reference:

[Declaring templates](#) • [Execution order](#) • [Modifiers](#) • [Inline-variables \(\\$-variables\)](#) • [Code lines and blocks](#) • [Execution environment](#)

[Lua](#) reference:

[Registration](#) • [Subtitles object](#) • [Progress reporting](#) • [Config dialogues](#) • [Misc. APIs](#) • [karaskel.lua](#) • [utils.lua](#) • [unicode.lua](#) • [cleantags.lua](#)

[Karaskel](#) concepts:

[Style tables](#) • [Dialogue line tables](#) • [Syllable tables](#) • [Inline effects](#) • [Furigana](#)

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

Furigana karaoke

Furigana (in Aegisub often shortened to *furi*) refers to little phonetic guide characters written along the main text in Japanese, specifically using the hiragana phonetic alphabet to describe how the ideographic kanji characters should be pronounced. Putting smaller text next to a main line of text is in general referred to as *ruby text* [↗](#), but since the implementation discussed here is designed specifically with Japanese furigana in mind, the ruby text is also referred to as furigana everywhere.

None of the subtitle formats Aegisub supports, natively support ruby text or furigana, however the [Automation 4 karaskel.lua](#) standard include implements an algorithm that can create basic furigana layouts by calculating the position of every individual character.

This page describes the syntax the Automation 4 karaskel.lua script understands for furigana text, and how to use the layout information it calculates to actually create positioned characters.

[Karaoke Templater](#) also implements support for furigana using the karaskel.lua algorithm and syntax.

It's important to note that the syntax is designed for karaoke, and revolves around karaoke timed text. It isn't suited for typesetting regular text (eg. dialogue lines) with general purpose ruby text, a more elaborate syntax and more complex layout engine would be required for that.

Multi-highlight syntax

A prerequisite for an integral part of the furigana syntax is the multi-highlight syntax.

If you make the text of a syllable a number sign (#, ASCII 35, Unicode U+0023) that syllable will "join" with the previous one: The number sign is removed and the timing of the two syllables are added together, producing just one syllable. You can have multiple number sign syllables in a row, adding up multiple timings in that way.

The timings of the individual number sign syllables are still stored in the [highlight table](#) of the generated syllable structure, but the main timing (`start_time` and `end_time`) of the syllable structure reflects only the added-together timings of the number sign syllables.

Contents [\[hide\]](#)

- 1 [Multi-highlight syntax](#)
- 2 [Basic furigana](#)
- 3 [Controlling the layout](#)
- 4 [Summary](#)
- 5 [Usage in Karaoke Templater](#)
- 6 [Usage in Lua scripts](#)



Example

This line shows how multi-highlight syntax is used to mark up kanji and groups of kanji that cover multiple syllables:

```
{\k5} 明日{\k10}#\k5#\k10}ま{\k7}た{\k10}会{\k4}う{\k6}時{\k14}#
```

It generates the following syllable structures:

Text	Syllable duration	Highlight durations
明日	20	5
		10
		5
ま	10	10

た	7	7	
会	10	10	
う	4	4	
時	20	6	
		14	

Basic furigana

To add furigana to a syllable, you add a pipe character (|, ASCII 124, Unicode U+007C) after the main syllable text, and then add the furigana text after the pipe. You can also add furigana to repeat-syllables (number sign syllables for multi-highlight) to have the furigana for a single main syllable span multiple furigana syllables.

When multiple consecutive syllables all have furigana, the furigana for all of those syllables are collected together and centered above the string of main syllables they belong to. If the string of furigana is wider than the main text the furigana is left-aligned with the main text. You can control this behaviour with special control characters, see below.

Example

Adding furigana to the example above:

```
{\k5}明
日|あ{\k10}#|し{\k5}#|た{\k10}ま{\k7}た{\k10}会|あ{\k4}う{\k6}時|と{\k14}#|き
```

The following syllables, highlights and furigana are produced:

Text	Syllable duration	Highlight/furigana durations	Furigana
明日	20	5	あ
		10	し
		5	た
ま	10	10	
た	7	7	
会	10	10	あ
う	4	4	
時	20	6	と
		14	き

Controlling the layout

Often the layout produced with the plain furigana syntax isn't exactly what you want, or maybe even plain misleading. Because of this, there's two special characters that can be used to control how the furigana are laid out.

Both of these two special characters are placed before the first character of the furigana of a syllable, ie. right after the pipe character.

First is the exclamation mark (!, ASCII 33, Unicode U+0021) which marks a "sequence break". This acts as a kind of invisible divider that prevents the furigana in this syllable to merge with that of the previous syllable. You will usually use this when you have two adjacent kanji words that both have furigana, but the furigana for them need to be separate. In that case, put the exclamation mark as the first character in the furigana for the first syllable of the second word.

The other special character is the less-than sign (<, ASCII 60, Unicode U+003C) which marks a "sequence break with float-left". It has the same sequence break semantics as the exclamation

mark, but also changes the overflow behaviour. When the furigana sequence starts with a less-than sign marked furigana syllable is wider than the main text it applies to, it will always center above the main text, even if it means it has to extend over the left edge of it.

In all cases, if two furigana sequences extend beyond their main text such that they would overlap, the main text is moved such that the furigana won't overlap.

Example

Here is the same (rather contrived) sample text shown without layout control and with each of the two layout control characters:

It is very hard to tell the difference between the two first and the difference is indeed only a few pixels, however it is there. In the first sample, the た extends a bit over the left edge of 魂 and above 国 while it exactly left-aligns with 魂 in the second. In the second, ちゅうごく is also centered above 中国 while it isn't in the first.

Summary

Char	ASCII	Unicode	Where	Meaning
#	35	U+0023 U+FF03	Instead of main text	Extend previous syllable with another highlight
	124	U+007C U+FF5C	Between main text and furigana	Separate main text and furigana text of a syllable
!	33	U+0021 U+FF01	First character of furigana	Sequence break; prevent joining furigana for this syllable with furigana from previous syllable
<	60	U+003C U+FF1C	First character of furigana	Sequence break with float-left; prevent joining furigana for this syllable with furigana from previous syllable, but allow furigana to extend left of main text

Note that every special character can in fact be represented by two different Unicode codepoints. The first is the regular character, corresponding to the ASCII character, while the second (high) codepoint is the *full width* version of the character. Often when using an IME (Input Method Editor) to edit Japanese text it is easier to input text in full width mode than switching the IME off to enter a single or two regular ASCII characters and switch it on again. Therefore both the half width (ASCII) and full width versions of the characters are accepted.

Usage in Karaoke Templater

Furigana: [The *furi* template class](#)

Todo: elaborate

Example

The examples used earlier on this page are all generated using this kara-templater snippet:

```

Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template
syl,{\pos(!line.left+syl.center!,!line.middle!)\an5\k!syl.start_time/10!\k$kdur}
}
Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template
furi,{\pos(!line.left+syl.center!,!line.middle-
line.height!)\an5\k!syl.start_time/10!\k$kdur}
Comment:
0,0:00:00.00,0:00:02.00,Default,,0000,0000,0000,karaoke,{\k15}二|ふ{\k15}#|た{\k10}
人|リ{\k15}だ{\k57}け{\k5}の{\k6}地|ほ{\k5}球|し{\k8}で
Comment:
0,0:00:02.00,0:00:04.00,Default,,0000,0000,0000,karaoke,{\k10}中|ちゆ{\k10}#|う{\k10}
国|ぢ{\k10}#|<{\k10}魂|<た{\k10}#|ま{\k10}#|し{\k10}#|い
Comment:
0,0:00:04.00,0:00:06.00,Default,,0000,0000,0000,karaoke,{\k10}中|ちゆ{\k10}#|う{\k10}
国|ぢ{\k10}#|<{\k10}魂|!た{\k10}#|ま{\k10}#|し{\k10}#|い
Comment:
0,0:00:06.00,0:00:08.00,Default,,0000,0000,0000,karaoke,{\k10}中|ちゆ{\k10}#|う{\k10}
国|ぢ{\k10}#|<{\k10}魂|た{\k10}#|ま{\k10}#|し{\k10}#|い

```

The font used in MS PMincho 30 pt with the furigana being 15 pt.

Usage in Lua scripts

It's all in [karaskel](#).

Furigana layout is automatically invoked by `karaskel.preproc_line_pos` if a furigana style exists for a line main style. The furigana style for a main style is a style with the same name, except `-furigana` appended to the name. Eg. the furigana style of `Default` is `Default-furigana`.

Karaskel can generate automatic furigana styles if the `generate_furigana` argument (second) to the `karaskel.collect_head` function is `true`. Automatic furigana styles are identical to the main style they're based on, except the font size is halved.

Furigana syllables are stored in `line.furi` and follows the same format as regular syllables. You have to remember setting the style of the lines you generate to the furigana style.

Multi-highlights are always processed even when furigana layout isn't done. Multi-highlight data are stored in `syl.highlights`.

Todo: more details

Automation 4	
Overview:	Automation Manager • Running macros • Using export filters • Standard macros • Changes from Automation 3 • Moving from Automation 3
Karaoke Templater reference:	Declaring templates • Execution order • Modifiers • Inline-variables (\$-variables) • Code lines and blocks • Execution environment

Lua reference:

[Registration](#) • [Subtitles object](#) • [Progress reporting](#) • [Config dialogues](#) • [Misc. APIs](#) • [karaskel.lua](#) • [utils.lua](#) • [unicode.lua](#) • [cleantags.lua](#)

Karaskel concepts:

[Style tables](#) • [Dialogue line tables](#) • [Syllable tables](#) • [Inline effects](#) • [Furigana](#)

Category: [Pages with Todo items](#)

Category:Lua Reference

This category collects all articles documenting the Automation 4 Lua scripting interface in Aegisub.

See the [Lua Reference](#) page for a general overview of the system.

Pages in category "Lua Reference"

The following 9 pages are in this category, out of 9 total.

C

- [Automation 4 Lua Configuration dialogues](#)

R

- [Lua Reference](#)
- [Automation 4 Lua Registration](#)

u

- [Automation 4 unicode.lua](#)
- [Automation 4 utils.lua](#)

M

- [Automation 4 Lua Miscellaneous APIs](#)

S

- [Automation 4 Lua Subtitle file interface](#)

P

- [Automation 4 Lua Progress reporting](#)

k

- [Automation 4 karaskel.lua](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

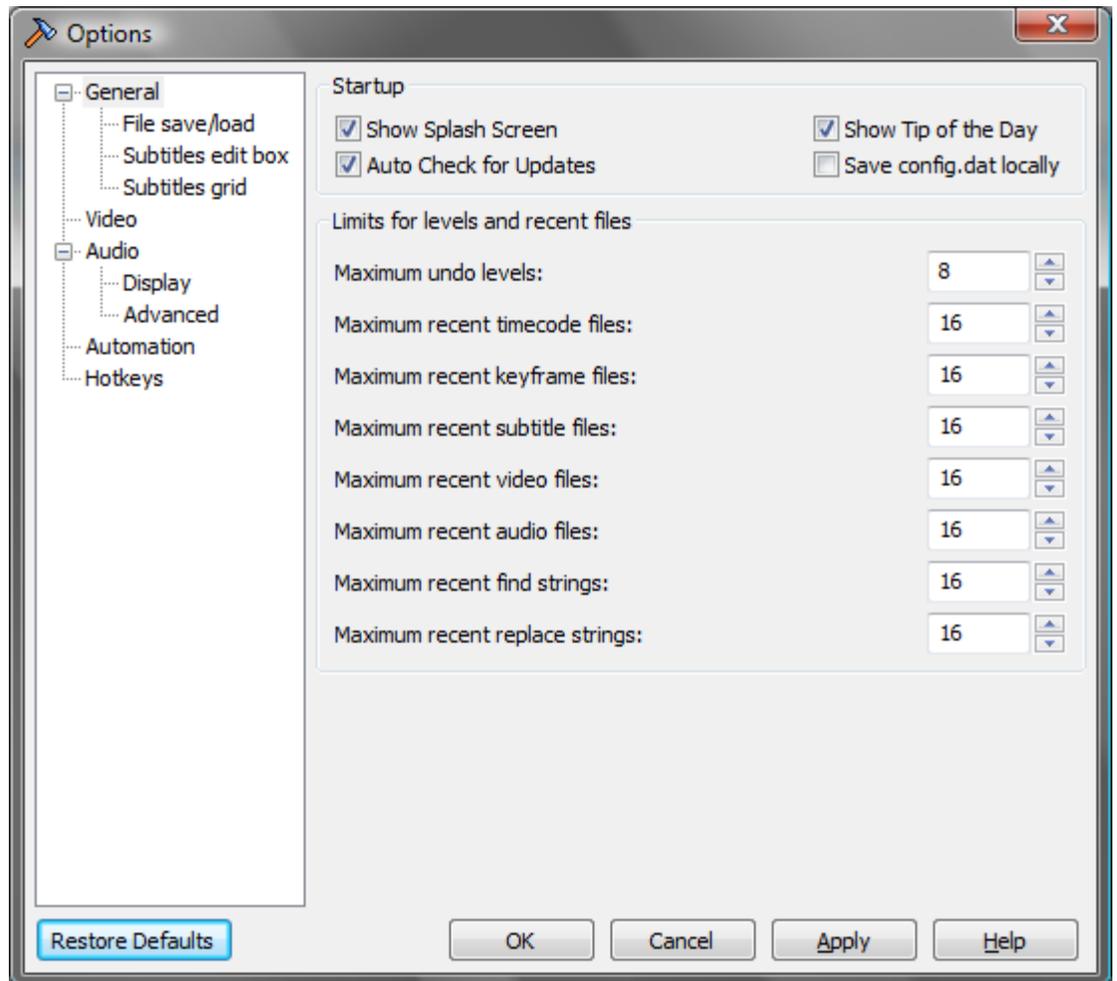
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Options general.png

- File
- File history
- File links



No higher resolution available.

[Options_general.png](#) (554 × 494 pixels, file size: 35 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [🔗](#) for more information)

File links

The following page links to this file:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

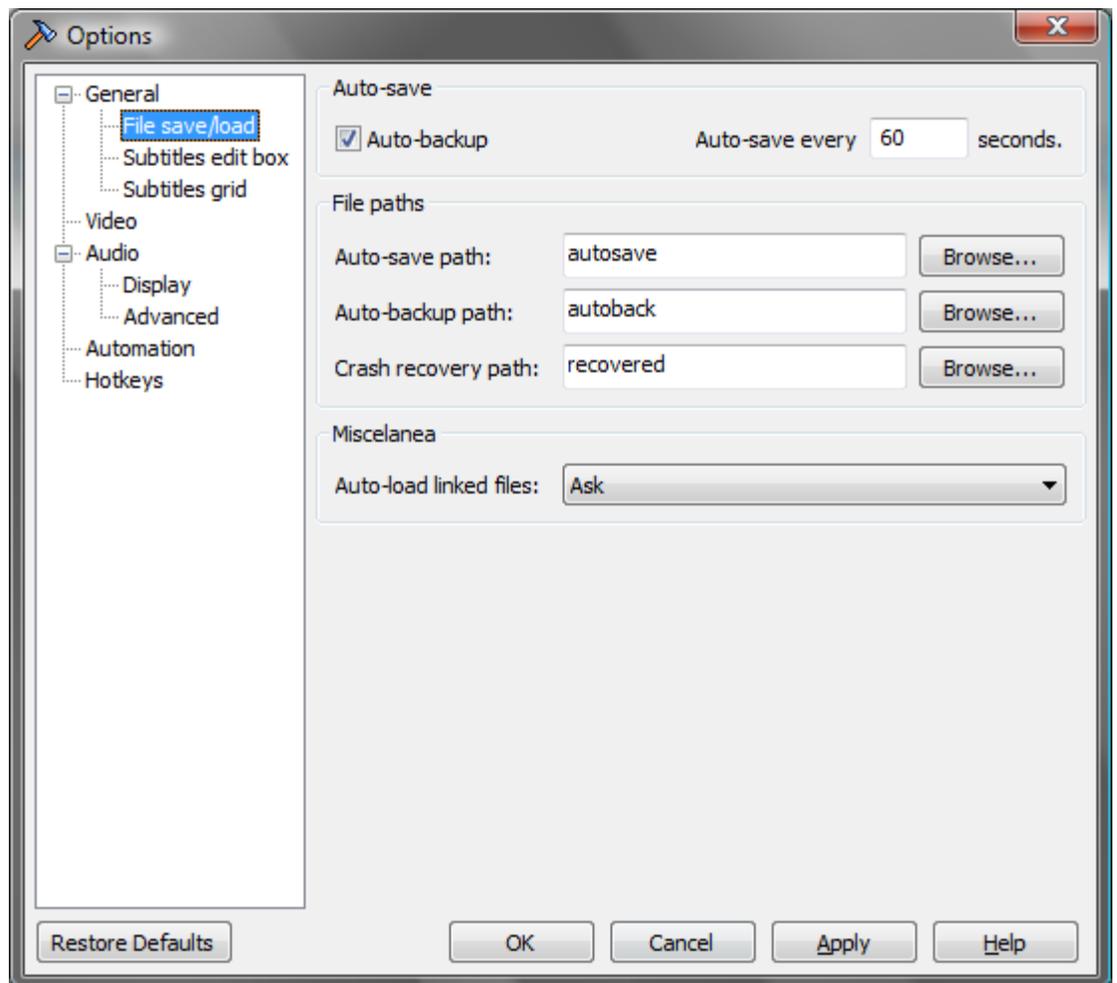
Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- Options

File:Options file.png

- File
- File history
- File links



No higher resolution available.

[Options_file.png](#) (554 × 494 pixels, file size: 30 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

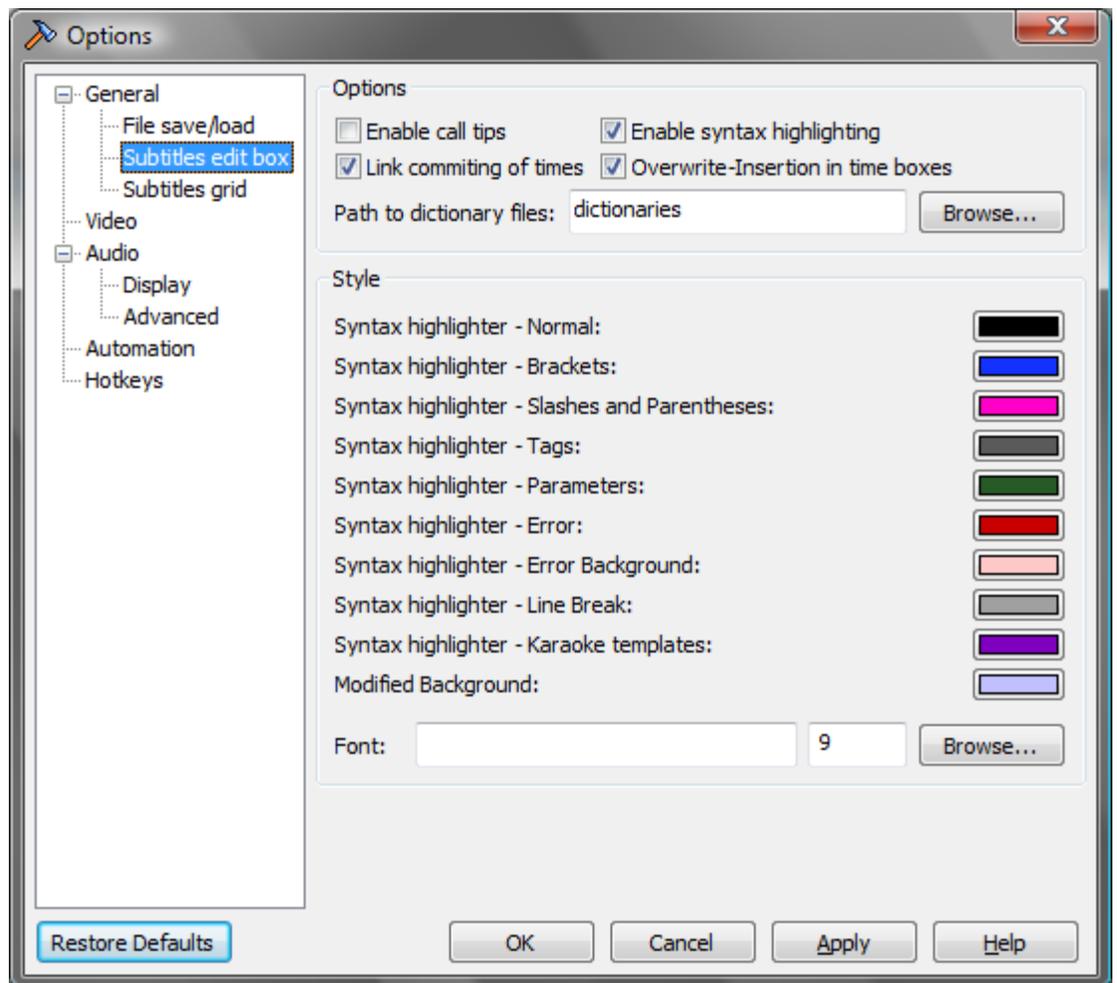
Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- Options

File:Options editbox.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Options_editbox.png](#) (554 × 494 pixels, file size: 39 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

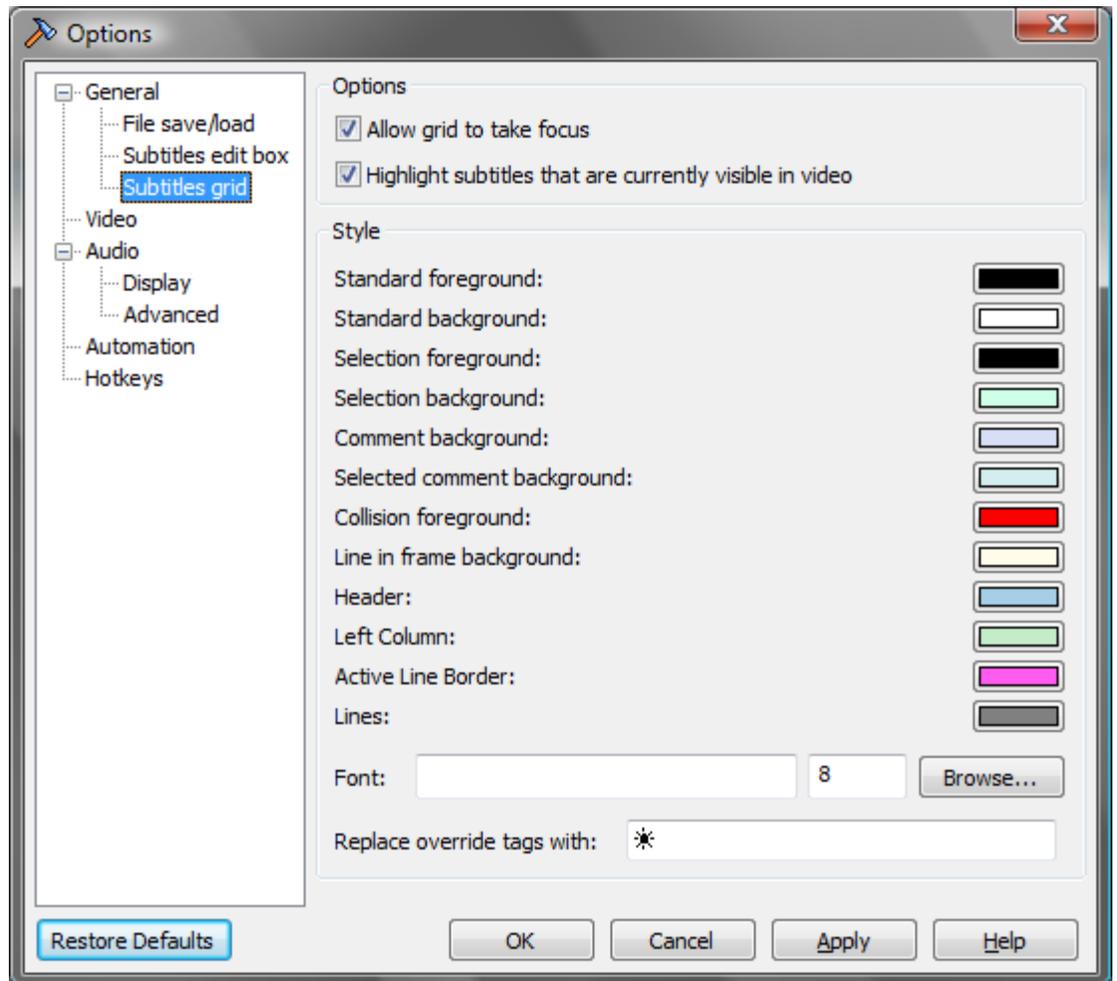
Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- Options

File:Options grid.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Options_grid.png](#) (554 × 494 pixels, file size: 36 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

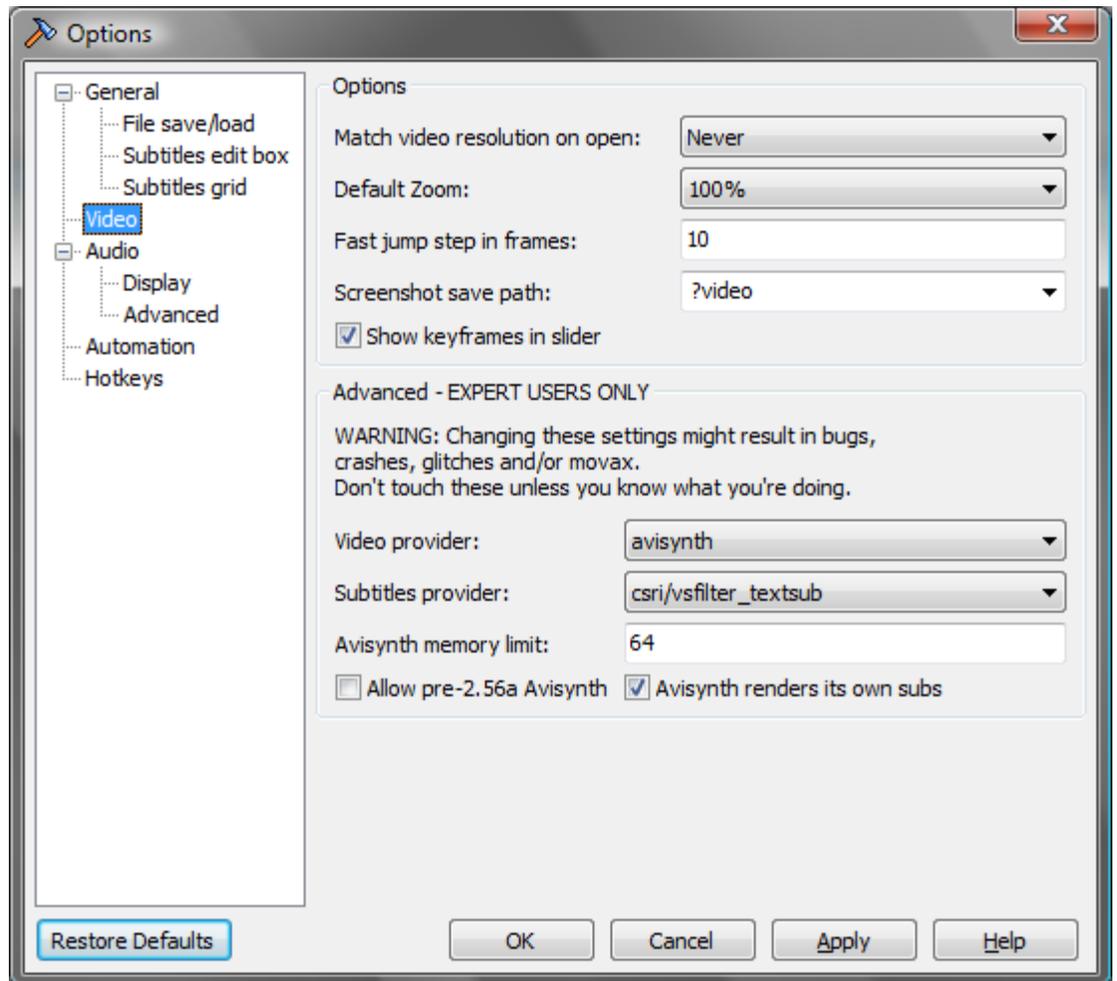
Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- Options

File:Options video.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Options_video.png](#) (554 × 494 pixels, file size: 37 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

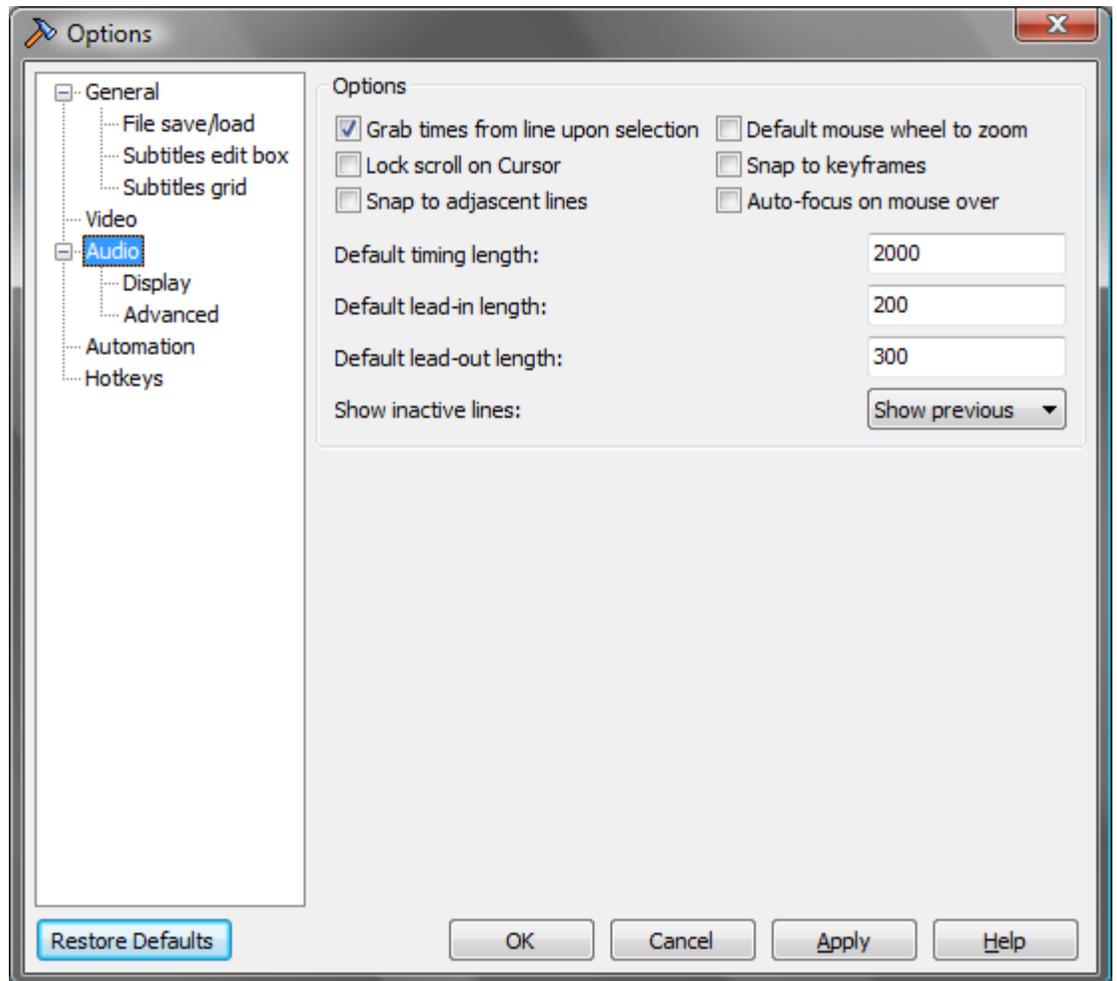
Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- Options

File:Options audio.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Options_audio.png](#) (554 × 494 pixels, file size: 28 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

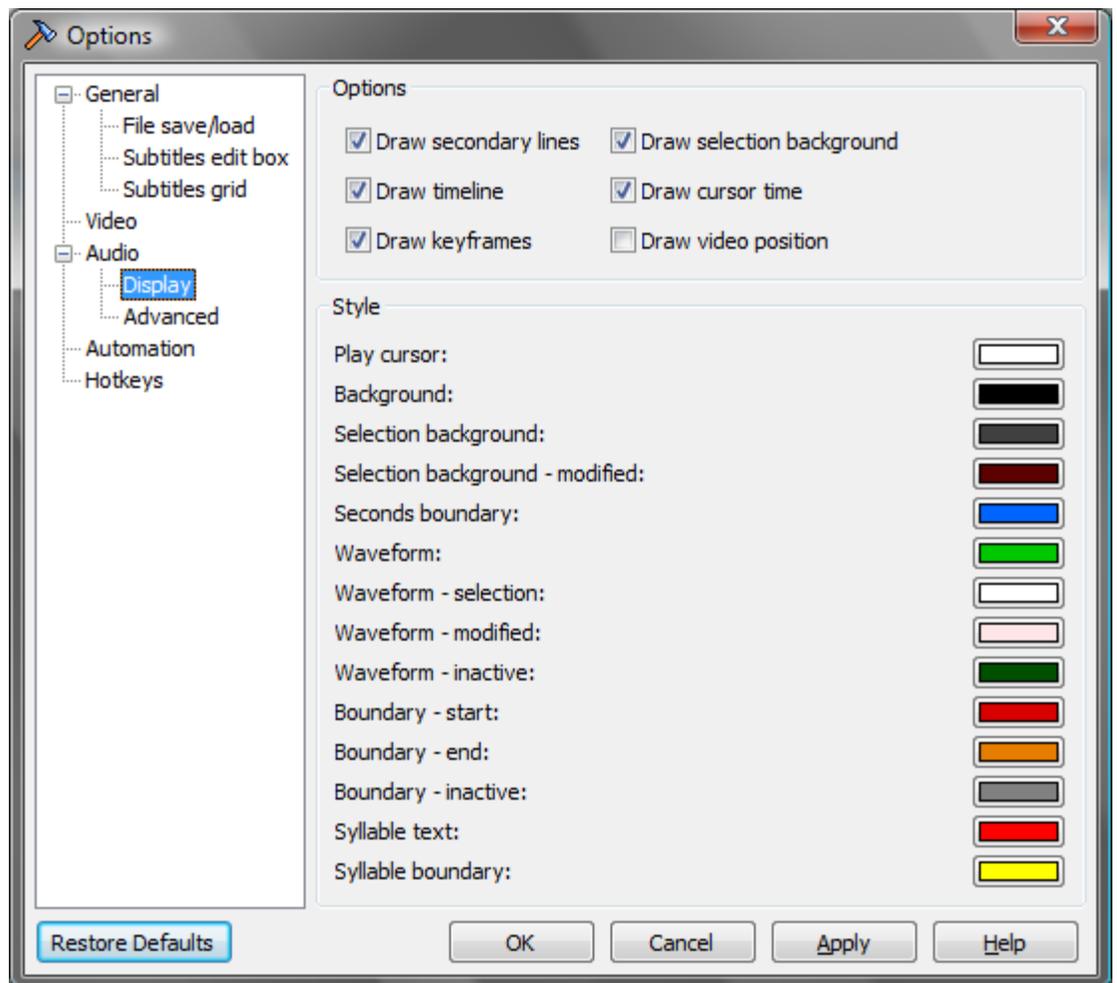
Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- Options

File:Options audio display.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Options_audio_display.png](#) (554 × 494 pixels, file size: 37 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [for more information](#))

File links

The following page links to this file:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

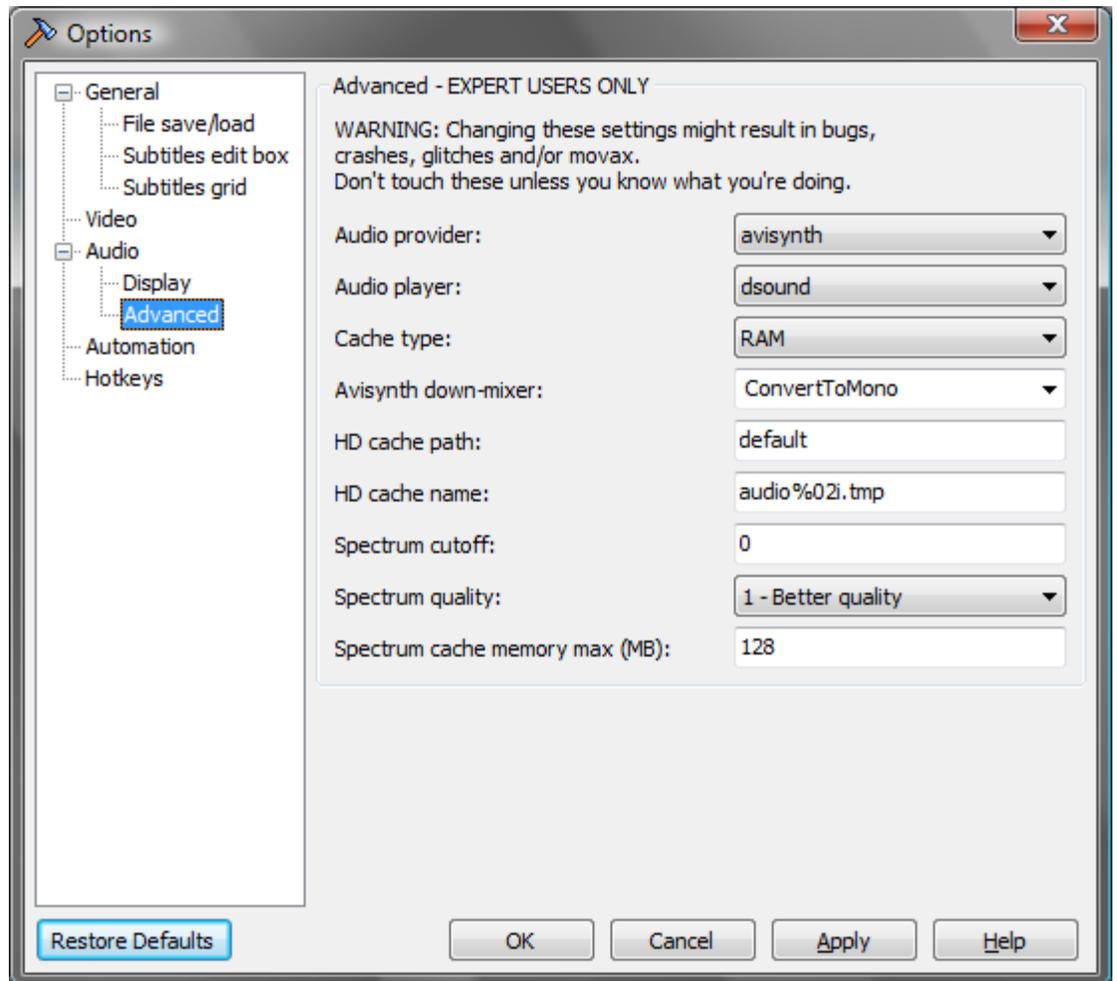
Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- Options

File:Options audio advanced.png

- File
- File history
- File links



No higher resolution available.

[Options_audio_advanced.png](#) (554 × 494 pixels, file size: 36 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

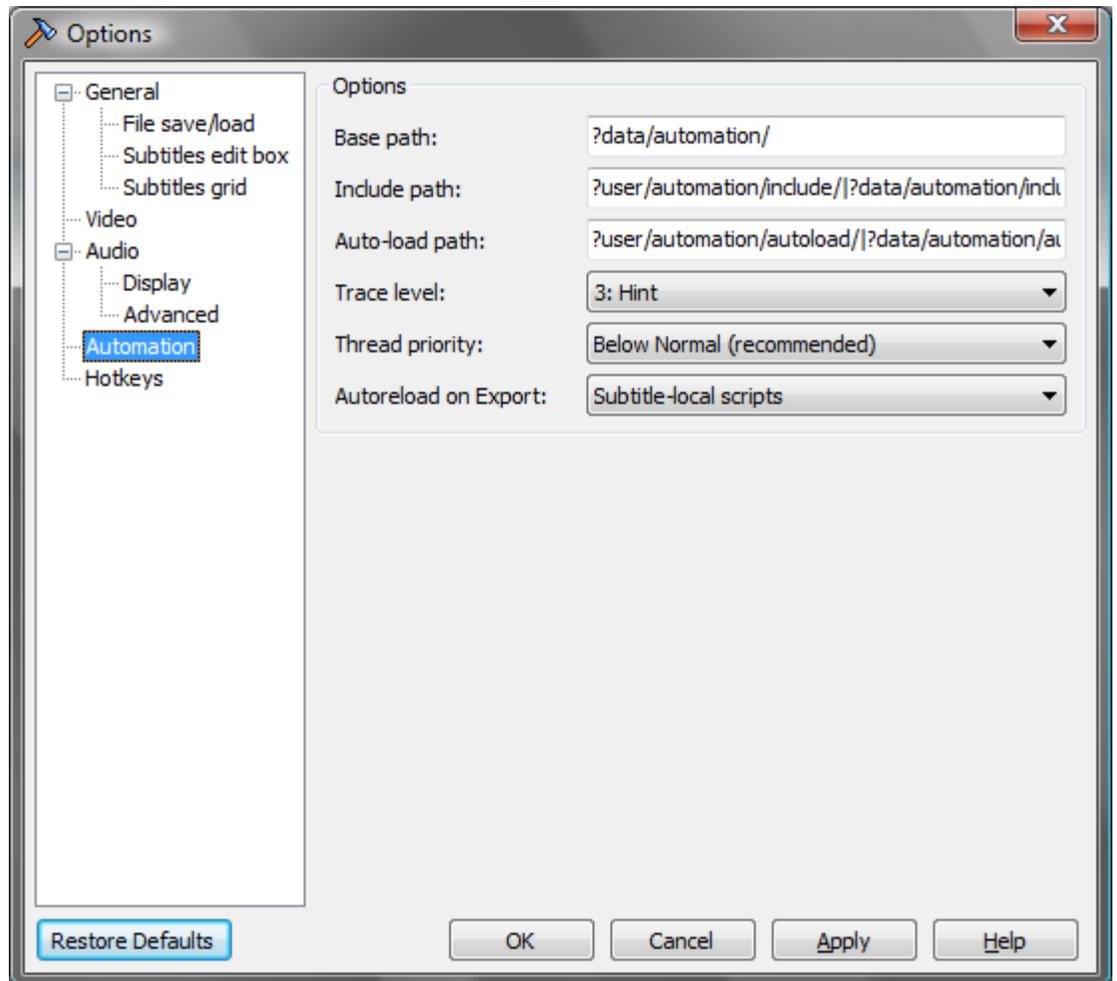
Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- Options

File:Options automation.png

- File
- File history
- File links



No higher resolution available.

[Options_automation.png](#) (554 × 494 pixels, file size: 30 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

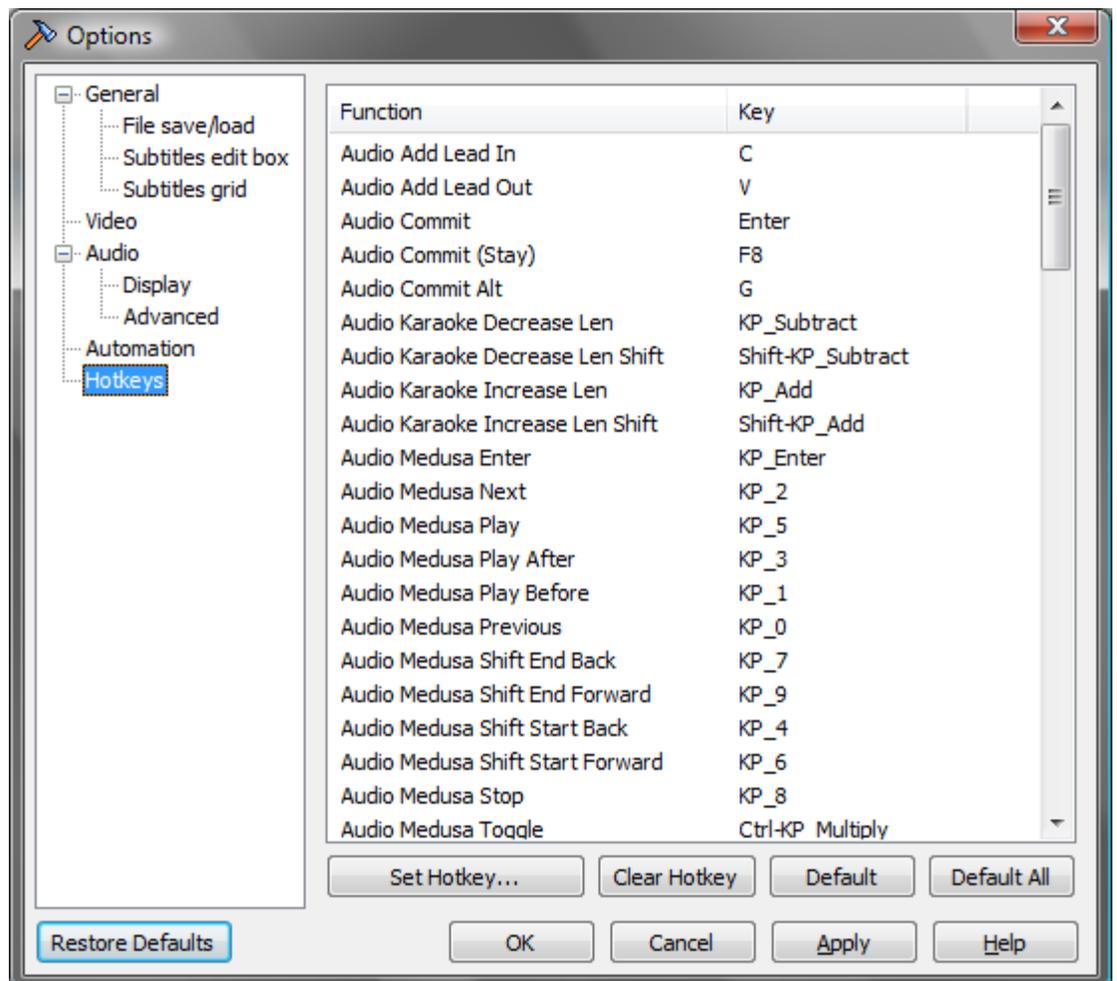
Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- Options

File:Options hotkeys.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Options_hotkeys.png](#) (554 × 494 pixels, file size: 32 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

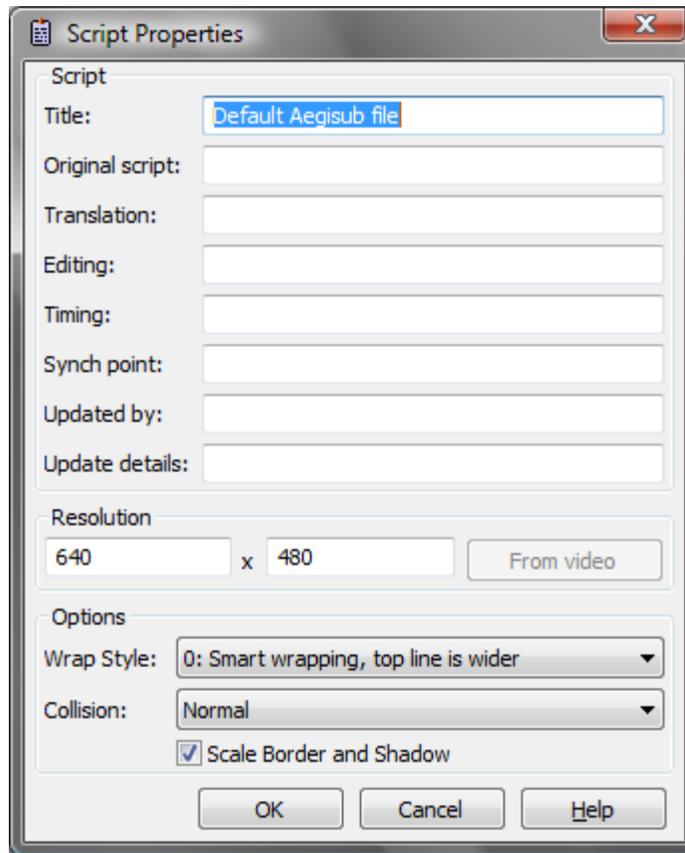
Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- Options

File:Properties.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Properties.png](#) (347 × 428 pixels, file size: 22 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Properties](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

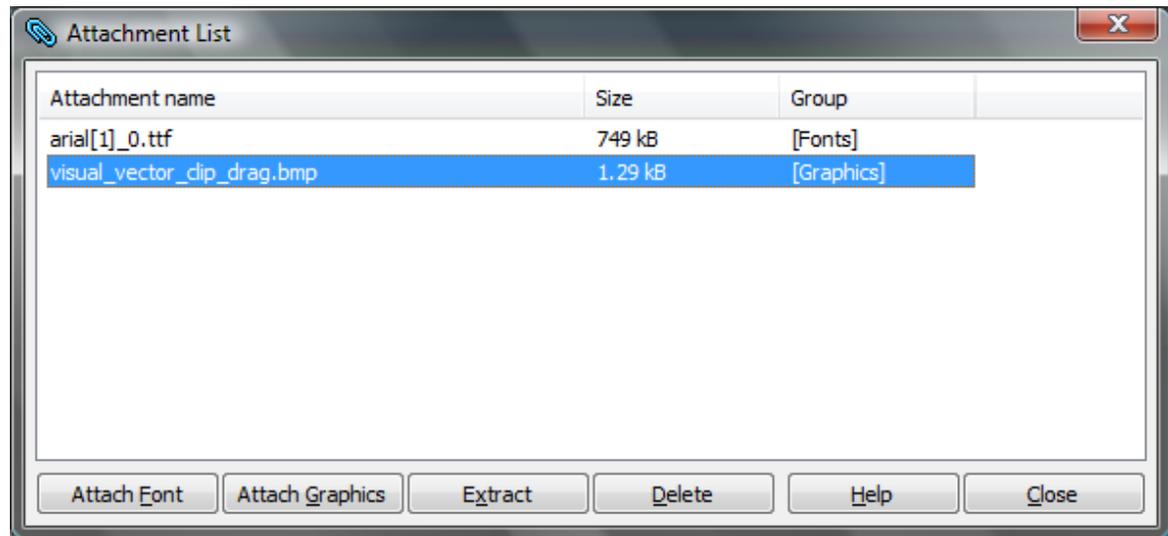
» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

File:Attachment list.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Attachment_list.png](#) (596 × 274 pixels, file size: 23 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Attachment Manager](#)

clampazar-wish.base.ass - Aegisub v2.2.0

File Edit Subtitles Timing Video Audio Automation View Help

62.5%

Menu Bar
Tool Bar

Video Box

meguriaeru you ni
巡り逢えるように

...was for us to stay together.

Audio Box

0:00:23.04

Karaoke Join Split

Edit Box

Comment Romaji Actor Effect

0 0:00:20.30 0:00:25.44 0:00:05.14 0 0 0

B I U ↺ ↻ AB AB AB AB Commit Time Frame

{\k15}me{\k17}gu{\k31}ri{\k30}a{\k31}e{\k24}ru{\k6} {\k93}yo{\k17}u{\k9} {\k241}ni

0:00:20.311 - 487 +11ms; -5129ms

Subtitles Grid

#	Start	End	Style	Effect	Text
1	0:00:00.00	0:00:00.00	Default		--- Karaoke effect ---
2	0:00:00.00	0:00:00.00	Romaji	template pre-line romaji	!retime("set",line.start_time-200,line.end_time+200)!{\fad(150,200)}pos(\$x,\$y){k20}
3	0:00:00.00	0:03:00.85	Romaji	template line romaji	{\kf\$kdur}
4	0:00:00.00	0:00:00.00	Kanji	template pre-line kanji	!retime("set",line.start_time-200,line.end_time+200)!{\fad(150,200)}pos(\$x,\$y){k20}
5	0:00:00.00	0:03:00.85	Kanji	template line kanji	{\kf\$kdur}
6	0:00:00.00	0:00:00.00	Translation	template pre-line main	!retime("set",line.start_time-200,line.end_time+200)!{\fad(150,200)}pos(\$x,\$y)}
7	0:00:00.00	0:00:00.00	Default		--- Wish ---
8	0:00:05.61	0:00:07.75	Romaji		{\k29}I{\k0} {\k51}wish{\k12}, {\k54}my{\k9} {\k59}wish
9	0:00:08.06	0:00:10.20	Romaji		{\k17}When{\k0} {\k16}I{\k0} {\k30}wish{\k0} {\k30}up{\k32}on{\k0} {\k19}a{\k0} {\k70}star
10	0:00:10.50	0:00:15.12	Romaji		{\k10}ho{\k18}shi{\k7} {\k18}ni{\k11} {\k30}ne{\k30}ga{\k28}i{\k0} {\k51}o{\k42} {\k32}ka{\k62}ke{\k46}ta{\k7} {\k70}no
11	0:00:15.41	0:00:17.54	Romaji		{\k27}I{\k6} {\k48}wish{\k9}, {\k64}my{\k0} {\k59}wish
12	0:00:17.86	0:00:19.98	Romaji		{\k25}My{\k7} {\k51}dream{\k10} {\k55}comes{\k0} {\k64}true
13	0:00:20.30	0:00:25.44	Romaji		{\k15}me{\k17}gu{\k31}ri{\k30}a{\k31}e{\k24}ru{\k6} {\k93}yo{\k17}u{\k9} {\k241}ni
14	0:00:27.97	0:00:31.99	Romaji		{\k90}kou{\k30}ha{\k60}ku{\k29} {\k22}na{\k11} {\k33}ku{\k22}mo{\k8} {\k15}ni{\k4} {\k44}sa{\k33}ku
15	0:00:32.87	0:00:36.83	Romaji		{\k88}hi{\k32}ka{\k54}ri{\k29} {\k27}no{\k11} {\k35}ha{\k22}na{\k37}zo{\k61}no
16	0:00:37.46	0:00:39.58	Romaji		{\k14}me{\k0} {\k11}ni{\k6} {\k28}mi{\k32}e{\k31}na{\k18}i{\k6} {\k23}ha{\k11}na{\k4} {\k28}ga
17	0:00:39.86	0:00:42.03	Romaji		{\k19}shi{\k17}ro{\k21}i{\k9} {\k29}ha{\k28}ne{\k4} {\k23}ni{\k7} {\k15}na{\k17}t{\k28}te
18	0:00:42.32	0:00:45.81	Romaji		{\k14}so{\k19}t{\k18}to{\k14} {\k30}ka{\k27}ta{\k6} {\k24}ni{\k6} {\k23}o{\k36}chi{\k23}te{\k9} {\k24}ku{\k76}ru

Status Bar

The image shows a screenshot of an audio editing software interface. The main window is divided into two sections: "Audio Display" at the top and "Timeline" at the bottom. The "Audio Display" section shows a waveform with a white segment on the left and a green segment on the right. The "Timeline" section shows a ruler with markers from 0 to 13. Below the timeline is a control panel with various buttons and sliders. Red numbers 1 through 26 are placed over different parts of the interface to highlight specific features. A red arrow labeled "Resizer dragger" points to a red L-shaped handle on the timeline at approximately 7.5 seconds. On the right side, there are three vertical sliders labeled 20, 21, and 22, and a green button labeled 23. The control panel includes buttons for "Karaoke", "Join", and "Split", as well as various playback and editing icons labeled 1 through 19.

1 2 3 4 5 6 7 8 9 10 11 12 13 Timeline

Audio Display

Resizer dragger

20 21 22

23

24 25 26

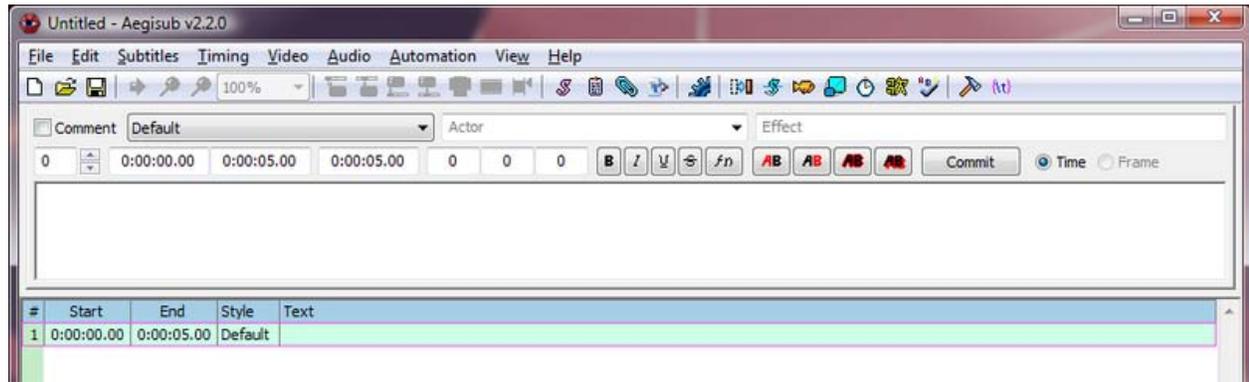
15 16 17 18 19

Karaoke Join Split

The image shows a software interface for a karaoke application. At the top, there is a video player window displaying a night cityscape with blue light trails. Below the video player is a horizontal timeline with numerical markers from 0 to 15. Underneath the timeline is a row of control buttons including play, pause, stop, and various seek functions. To the right of the video player are three vertical sliders and a green L-shaped button. At the bottom of the interface are three buttons labeled "Karaoke", "Join", and "Split".

File:Karatiming-1.png

- File
- File history
- File links



Size of this preview: 800 × 243 pixels

Full resolution (920 × 279 pixels, file size: 36 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

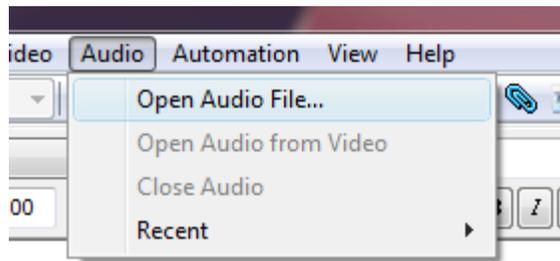
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Karatiming-2.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Karatiming-2.png](#) (282 × 147 pixels, file size: 9 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) ↗ for more information)

File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

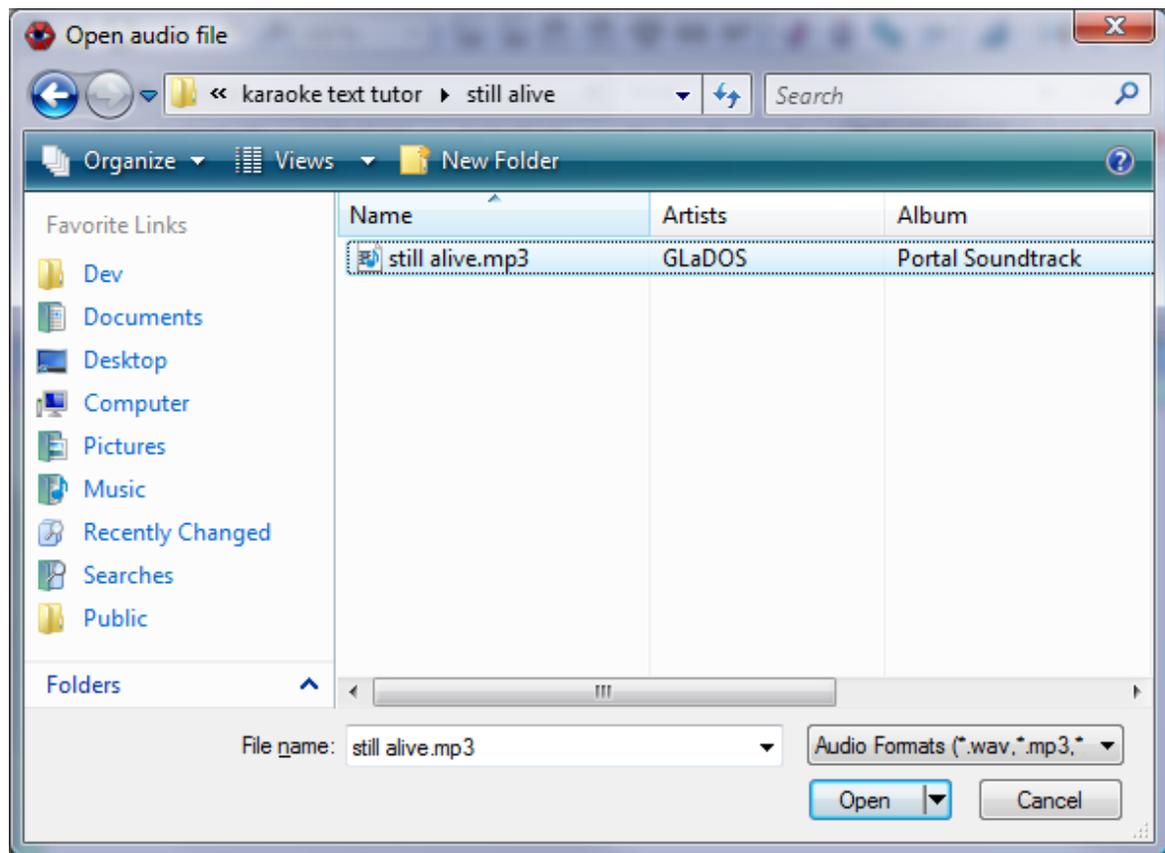
» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

File:Karatiming-3.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Karatiming-3.png](#) (593 × 434 pixels, file size: 65 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

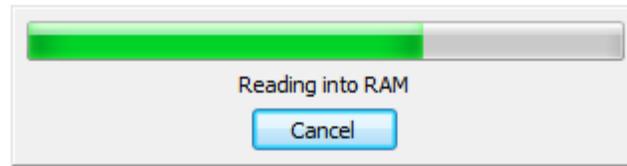
File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

File:Karatiming-4.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Karatiming-4.png](#) (331 × 98 pixels, file size: 3 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

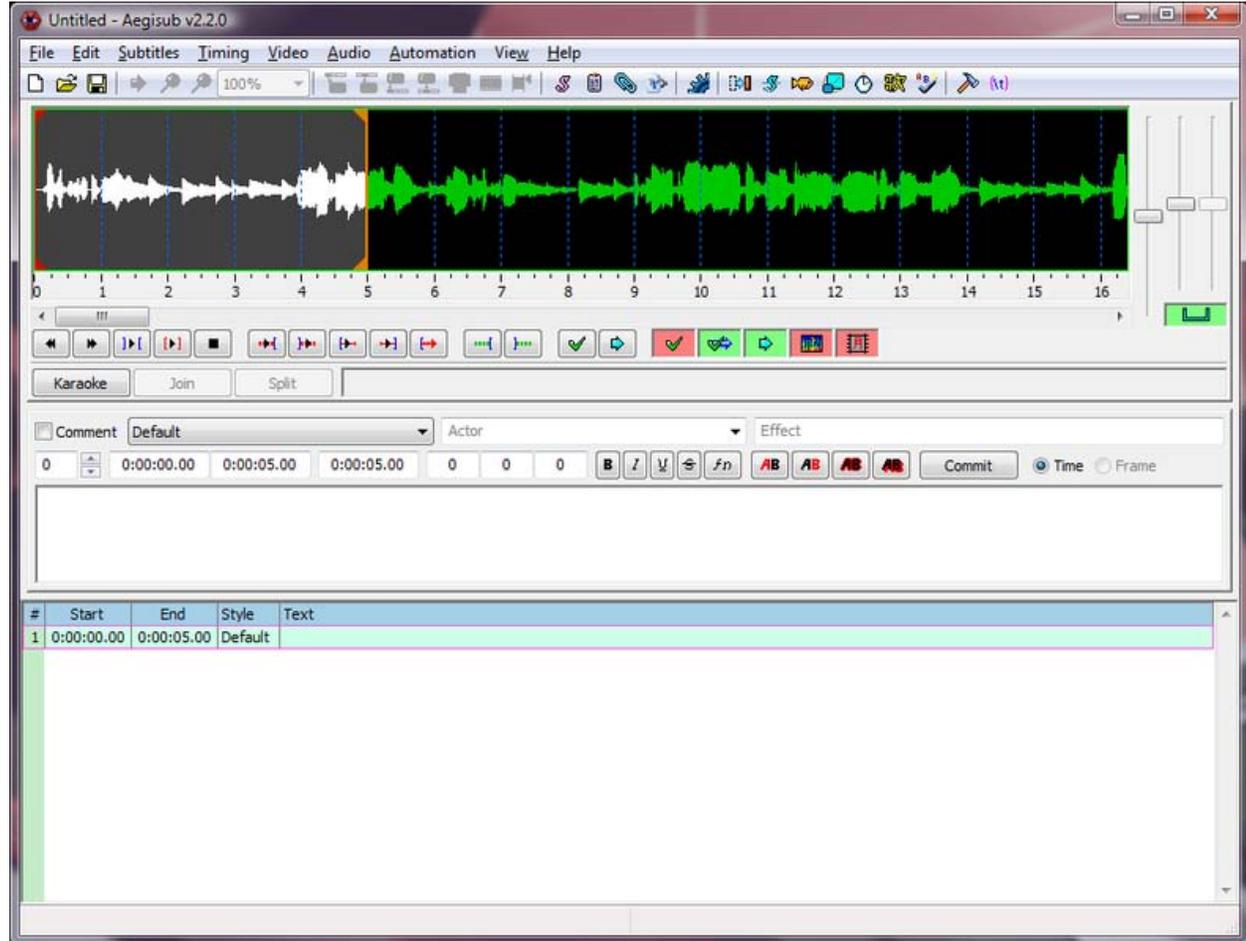
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Karatiming-5.png

- [File](#)
- [File history](#)
- [File links](#)



Size of this preview: 788 × 600 pixels
[Full resolution](#) (920 × 700 pixels, file size: 58 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

([Latest](#) | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

([Latest](#) | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

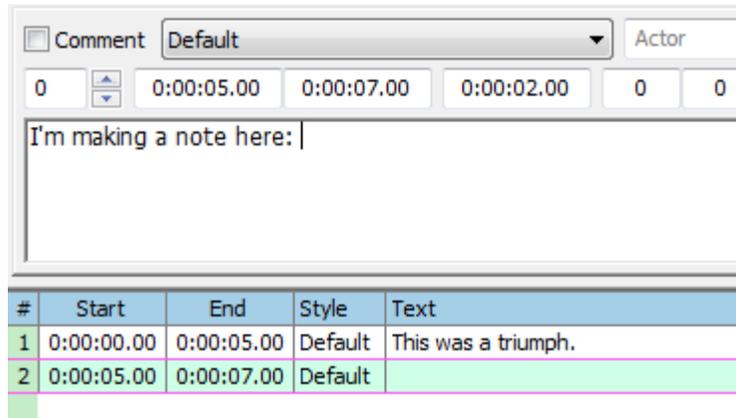
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Karatiming-6.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Karatiming-6.png](#) (372 × 211 pixels, file size: 7 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

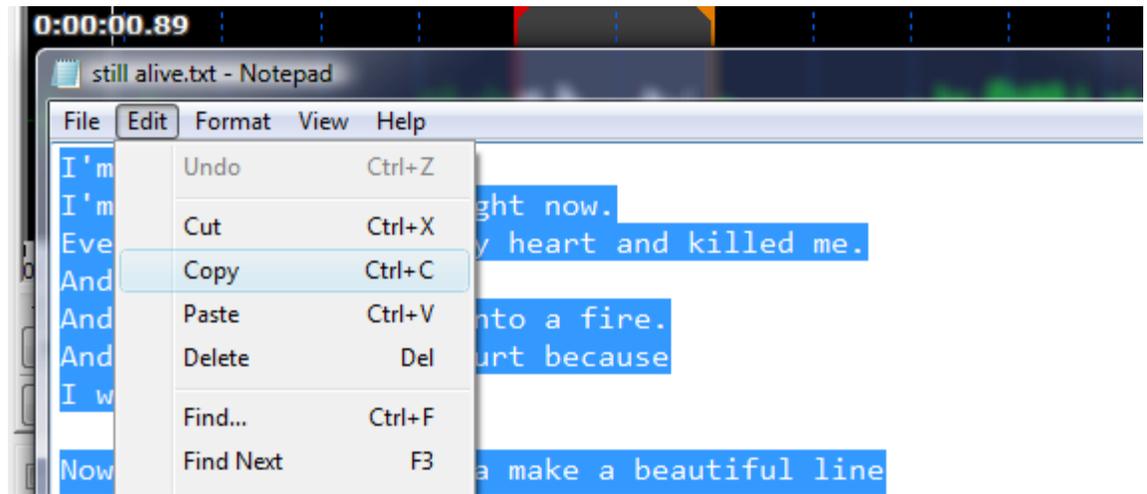
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Karatiming-7.png

- File
- File history
- File links



No higher resolution available.

[Karatiming-7.png](#) (570 × 246 pixels, file size: 28 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

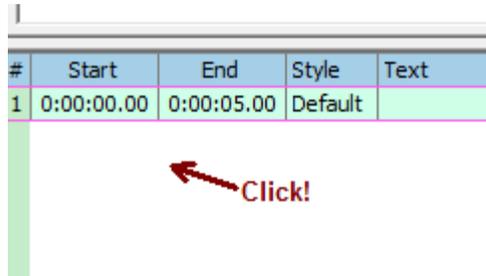
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Karatiming-8.png

- [File](#)
- [File history](#)
- [File links](#)



#	Start	End	Style	Text
1	0:00:00.00	0:00:05.00	Default	

No higher resolution available.

[Karatiming-8.png](#) (244 × 138 pixels, file size: 3 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [for more information](#))

File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

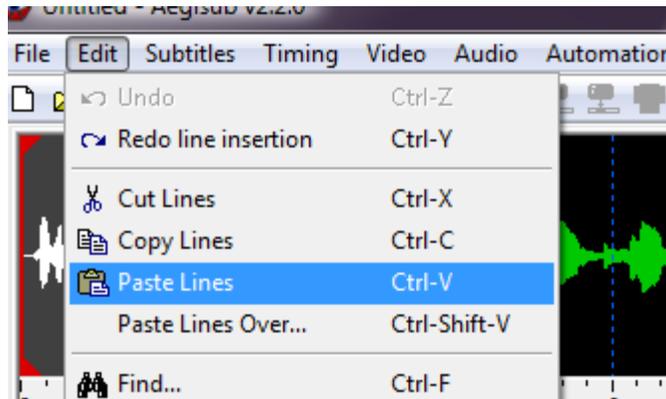
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Karatiming-9.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Karatiming-9.png](#) (332 × 198 pixels, file size: 12 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

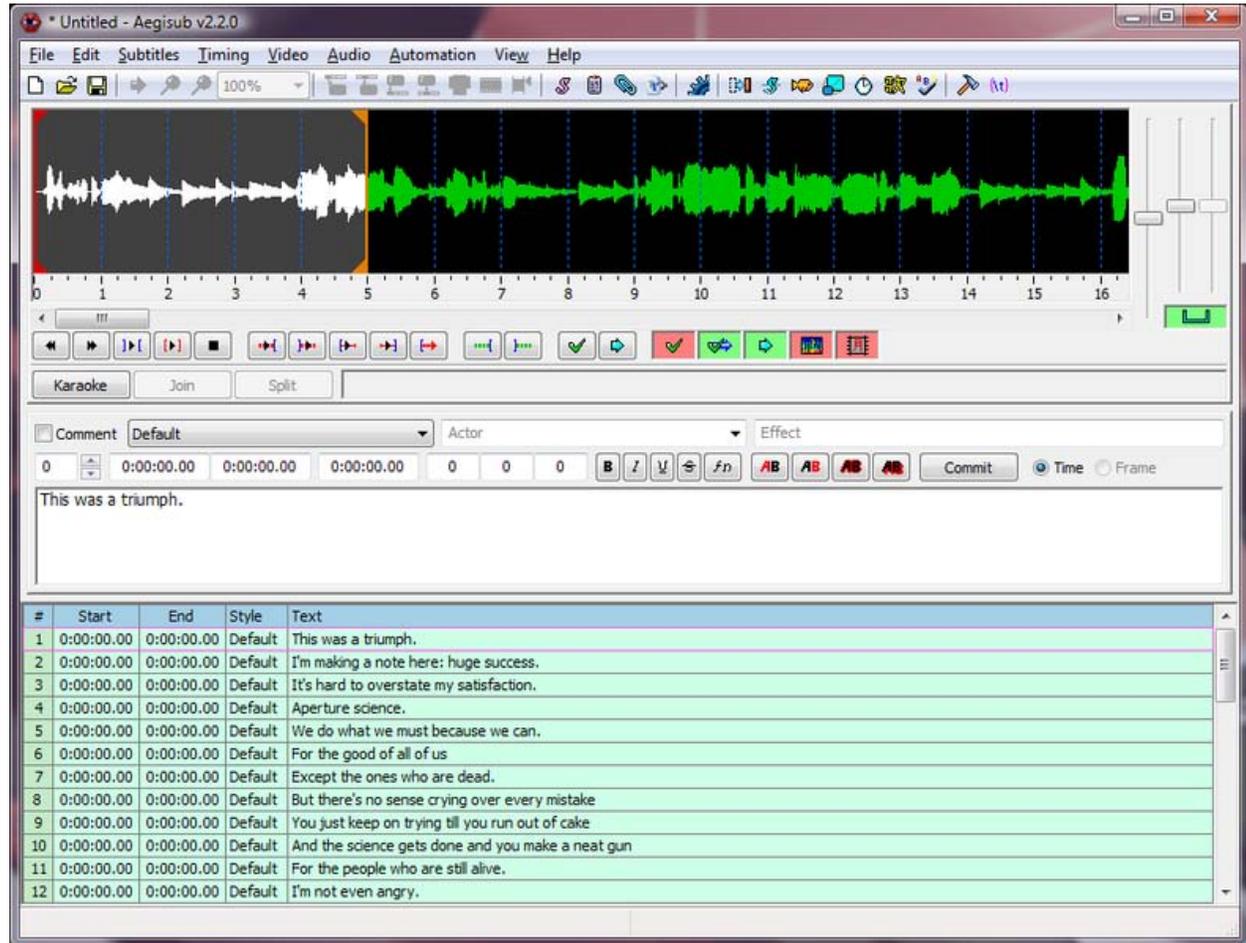
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Karatiming-10.png

- [File](#)
- [File history](#)
- [File links](#)



Size of this preview: 788 × 600 pixels
[Full resolution](#) (920 × 700 pixels, file size: 76 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

([Latest](#) | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

([Latest](#) | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

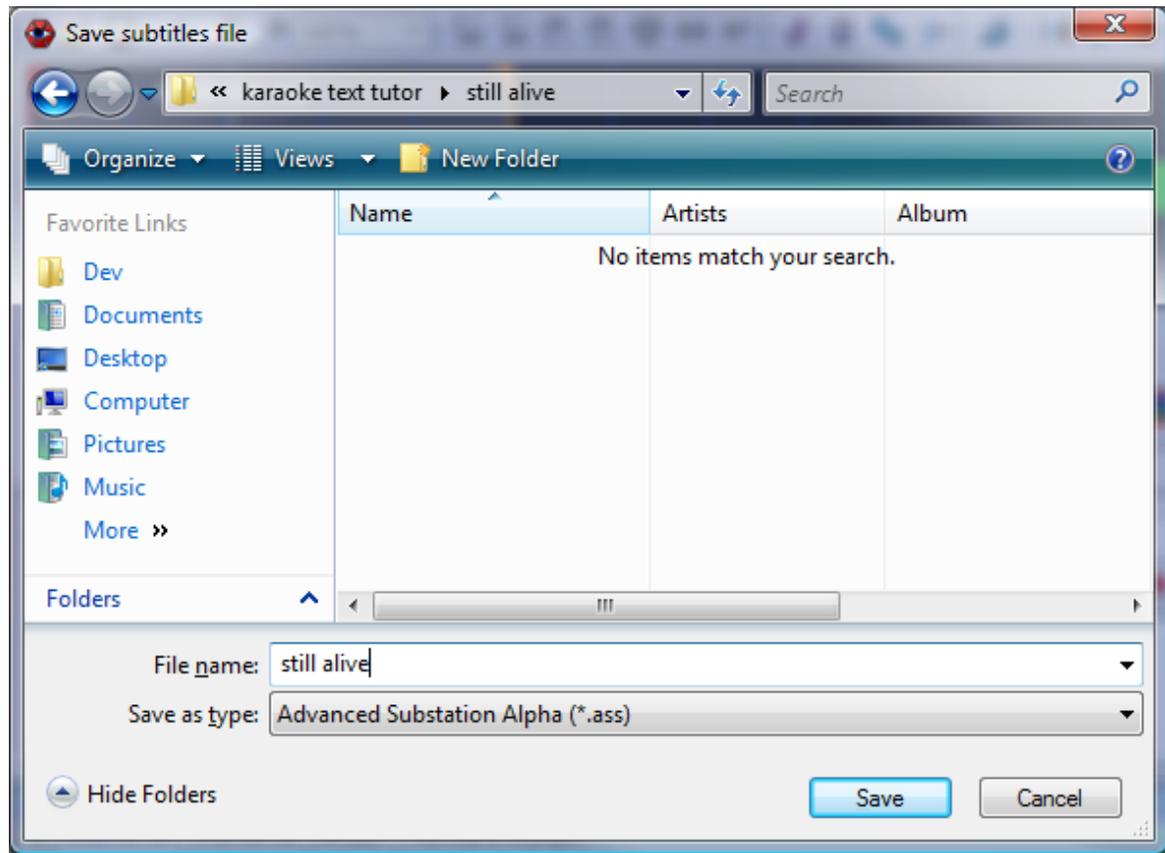
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Karatiming-11.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Karatiming-11.png](#) (593 × 434 pixels, file size: 66 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

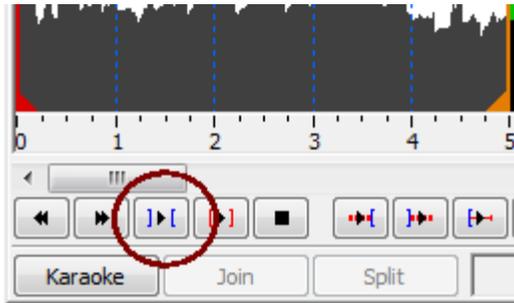
File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

File:Karatiming-12.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Karatiming-12.png](#) (260 × 152 pixels, file size: 5 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Karatiming-13.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Karatiming-13.png](#) (461 × 226 pixels, file size: 9 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time	Thumbnail	Dimensions	User	Comment
-----------	-----------	------------	------	---------

current	contribs			
---------	----------	--	--	--

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Karatiming-14.png

- [File](#)
- [File history](#)
- [File links](#)



Size of this preview: 800 × 81 pixels

[Full resolution](#) (825 × 84 pixels, file size: 8 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [for more information](#))

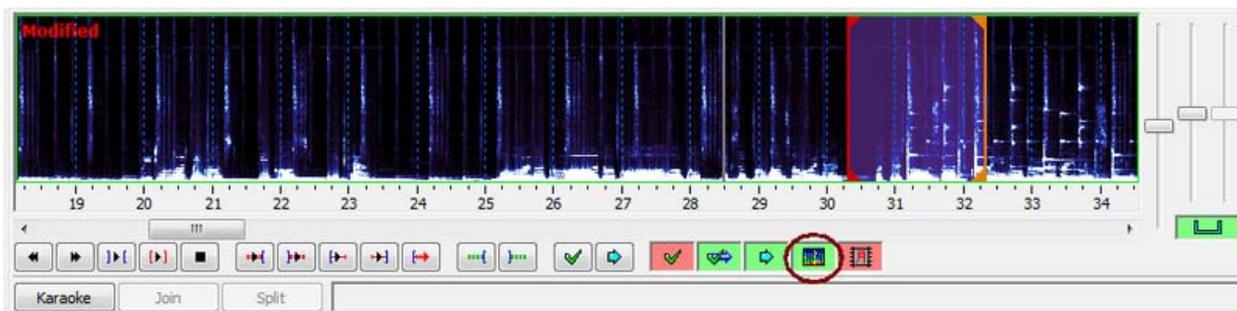
File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

File:Karatiming-15.png

- [File](#)
- [File history](#)
- [File links](#)



Size of this preview: 800 × 200 pixels
[Full resolution](#) (900 × 225 pixels, file size: 121 KB, MIME type: image/png)

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail Dimensions User Comment

current contribs)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Karaoke Timing Tutorial](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

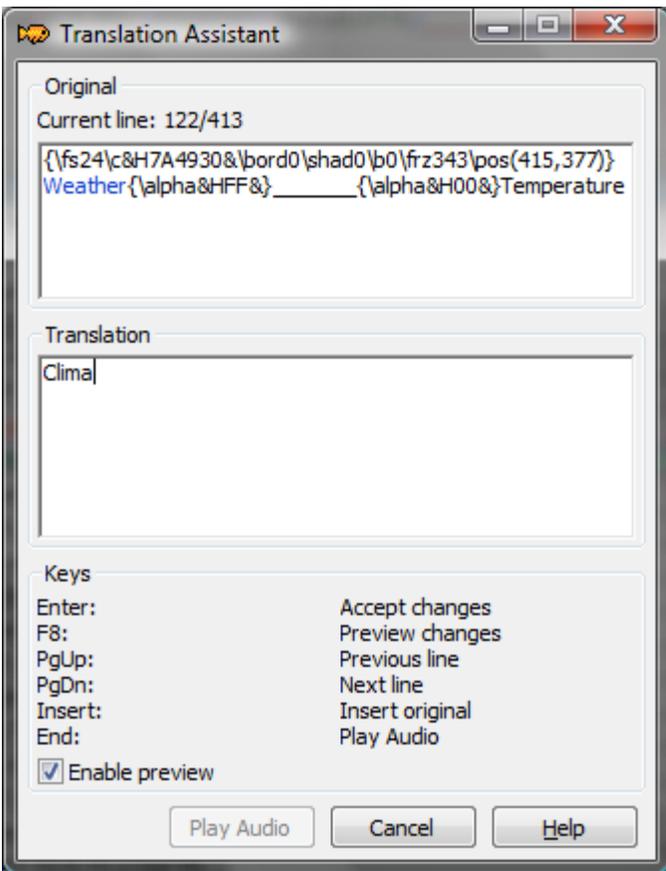
- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

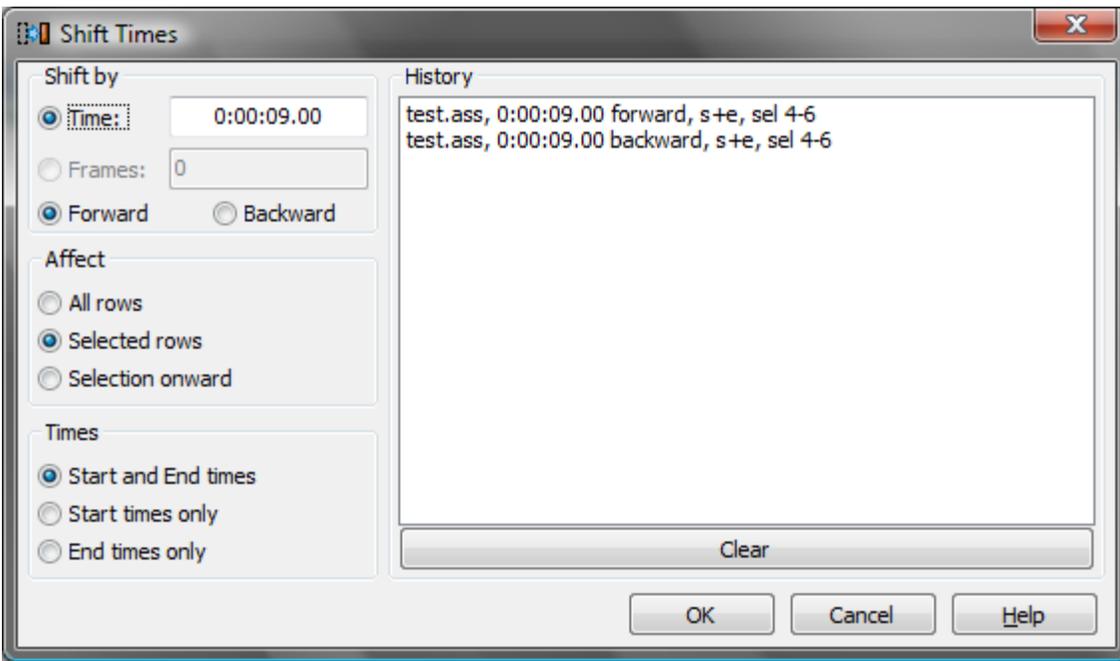
Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)





Automation View Help



Automation...

Apply karaoke template

Add edgeblur

Make fullwidth



1 Comment Default 2 Actor 3 Effect 4

0 0:00:00.00 0:00:05.00 0:00:05.00 0 0 0 B I U ↺ fn AB AB AB AB Commit Time Frame

5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22

The image shows a software interface for video editing or animation. At the top, there is a toolbar with several sections. On the left, there is a 'Comment' field with a dropdown menu set to 'Default'. Next to it is an 'Actor' dropdown menu set to '3', and an 'Effect' dropdown menu set to '4'. Below these are three timecode fields: '0:00:00.00', '0:00:05.00', and '0:00:05.00'. To the right of these are three numerical input fields, each containing '0'. Further right are icons for Bold (B), Italic (I), Underline (U), a circular arrow (refresh), and a function key (fn). Next are four red 'AB' markers. A 'Commit' button is located to the right of these markers. At the far right of the toolbar are radio buttons for 'Time' (selected) and 'Frame'. Below the toolbar is a large, empty rectangular workspace.

Aegisub is a tool designed for subtitles.

The screenshot shows a context menu in Aegisub. The menu is open over the word "desinged" in a subtitle. The menu items are:

- desinged (highlighted in blue)
- Add "desinged" to dictionary
- Spell checker language ▶
- No thesaurus suggestions
- Thesaurus language ▶
- Undo
- Cut
- Copy
- Paste
- Select All
- Split at cursor (preserve times)
- Split at cursor (estimate times)

In the background, a subtitle table is visible with columns for "Start" and "End". The first row shows "0:00:00.00" and "0:00:00.00".

#	Start	End	Style	Text
13	0:01:18.41	0:01:21.87	Eva Op Roomaji	omoi de wo uragiru nara
14	0:01:22.13	0:01:25.63	Eva Op Roomaji	kono sora wo daite kagayaku
15	0:01:25.80	0:01:29.34	Eva Op Roomaji	shounen yo shinwa ni nare
16	0:00:00.00	0:00:00.00	Default	Kanji
17	0:00:01.03	0:00:06.77	Eva Op Kanji	残#酷#な天#使のように
18	0:00:07.01	0:00:14.48	Eva Op Kanji	少#年#よ 神#話になれ
19	0:00:22.91	0:00:29.23	Eva Op Kanji	香#い風,#がいま 胸#のドアを叩#いても
20	0:00:29.68	0:00:36.93	Eva Op Kanji	私##だけをただ見つめて 微笑#んでるあなた
21	0:00:37.70	0:00:44.23	Eva Op Kanji	そっとふれるもの もとめることに夢中#で
22	0:00:44.60	0:00:51.36	Eva Op Kanji	運#命#さえまだ知らない いたいけな瞳##
23	0:00:52.26	0:00:58.34	Eva Op Kanji	だけどいつか気付くでしょう その背中#には
24	0:00:59.75	0:01:06.26	Eva Op Kanji	運#か未来# めざすための 羽根があること
25	0:01:07.14	0:01:10.70	Eva Op Kanji	残#酷#な天#使のテーゼ
26	0:01:10.94	0:01:14.43	Eva Op Kanji	窓#辺からやがて飛び立つ
27	0:01:14.65	0:01:18.07	Eva Op Kanji	ほとばしる熱#いワトスで
28	0:01:18.41	0:01:21.87	Eva Op Kanji	思#い出を裏#切るなら
29	0:01:22.13	0:01:25.63	Eva Op Kanji	この宇宙を抱いて輝##く
30	0:01:25.80	0:01:29.34	Eva Op Kanji	少#年#よ 神#話になれ
31	0:00:00.00	0:00:00.00	Default	Translation
32	0:00:01.03	0:00:06.77	Eva Op English	Just like the Cruel Angel,
33	0:00:07.01	0:00:14.48	Eva Op English	Young boy, rise as a legend!
34	0:00:22.91	0:00:29.23	Eva Op English	Even as the tender wind knocks at the door to your heart,
35	0:00:29.68	0:00:36.93	Eva Op English	You merely look at me and smile.

{\k46}o{\k46}mo{\k37}i{\k35}de {\k22}wo {\k25}u{\k25}
{\k46}ko{\k41}no {\k45}so{\k32}ra {\k24}wo {\k21}da
{\k55}sho{\k45}u{\k38}ne{\k25}n {\k23}yo {\k32}shi
Kanji
{\k84}残{\k71}#{\k54}酷{\k47}#{\k48}な{\k28}天{\k28}
{\k94}少{\k67}#{\k51}年{\k51}#{\k33}よ {\k58}神{\k58}
{\k22}青{\k18}#{\k76}い{\k21}風{\k36}#{\k31}が{\k31}



omoide wo uragiru nara
kono sora wo daite kagayaku
shounen yo shinwa ni nare
Kanji
残#酷#な天#使のように
少#年#よ 神#話になれ
青#い風#がいま 胸#のドアを叩#いても

*o*mo*i*de *wo *u*ra*gi*ru *na*
*ko*no *so*ra *wo *da*te *ka*
*sho*u*ne*n *yo *shi*n*wa *ni *
Kanji
残#*酷*#*な*天*#*使*の*よ*
少#*年*#*よ *神*#*話*に*な*
青#*い*風*#*が*いま *胸*#*

Insert (before)

Insert (after)

Insert at video time (before)

Insert at video time (after)

Duplicate

Duplicate and shift by 1 frame

Split (by karaoke)

Swap

Join (concatenate)

Join (keep first)

Join (as Karaoke)

Make times continuous (change start)

Make times continuous (change end)

Recombine Lines

Create audio clip

Copy

Cut

Paste

Delete

31	0:02:35.02	0:02:36.01	Style 1	It's no use.
32	0:02:37.05	0:02:39.35	Style 1	I shouldn't have come here after all.
33	0:02:43.22	0:02:43.66	Style 1	To Shinji\I'll be coming\nto get you,\nso wait for me. \NAttention here please!!
34	0:02:43.66	0:02:45.66	Style 1	To Shinji\I'll be coming\nto get you,\nso wait for me. \NAttention here please!!\NWWell, I guess we won't be meeting here.
35	0:02:45.66	0:02:46.02	Style 1	Well, I guess we won't be meeting here.
36	0:02:47.16	0:02:49.76	Style 1	Can't be helped. I'll go to a shelter.
37	0:03:15.02	0:03:15.42	Style 1	Estimated Path

31	0:02:35.02	0:02:36.01	Style 1	It's no use.
32	0:02:37.05	0:02:39.35	Style 1	I shouldn't have come here after all.
33	0:02:43.22	0:02:45.66	Style 1	To Shinji I'll be coming into get you, so wait for me. Attention here please!!
34	0:02:43.66	0:02:46.02	Style 1	Well, I guess we won't be meeting here.
35	0:02:47.16	0:02:49.76	Style 1	Can't be helped. I'll go to a shelter.
36	0:03:15.02	0:03:15.42	Style 1	Estimated Path

Export

Filters

- Limit to Visible Lines
- Karaoke template
- Transform Framerate
- Clean Tags
- Clean Script Info
- Fix Styles

Move up

Move down

Select all

Select none

Transform subtitles times, including those in override tags, from input to output. This is most useful to convert CFR to VFR for hardsubbing. You usually DO NOT want to check this filter for softsubbing.

Text encoding: UTF-8

Transform Framerate

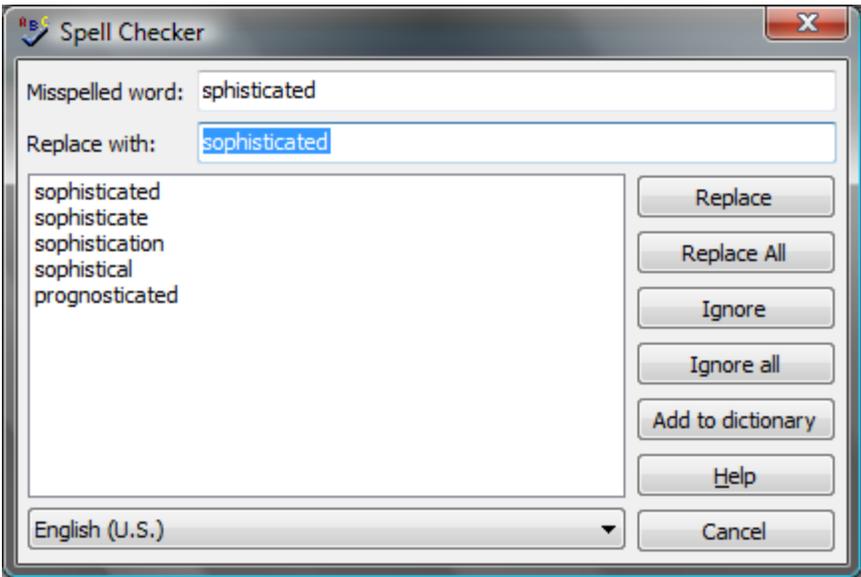
Input framerate: 23.976 From Video

Output: Variable Constant: 25

Export...

Cancel

Help



Select Fields to Paste Over

Fields

Please select the fields that you want to paste over:

- Layer
- Start Time
- End Time
- Style
- Actor
- Margin Left
- Margin Right
- Margin Vertical
- Effect
- Text

All None Times Text

OK Cancel Help

Select

Match

- Matches Doesn't Match

John Smith

- Match case
 Exact match
 Contains
 Regular Expression match

In Field

- Text Style Actor Effect

Match dialogues/comments

- Dialogues Comments

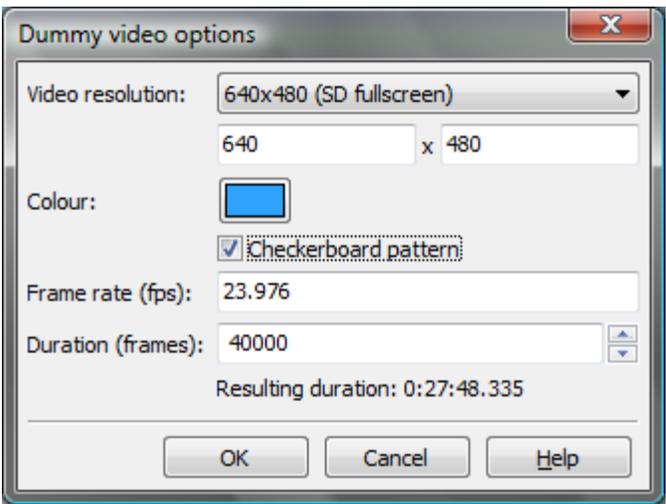
Action

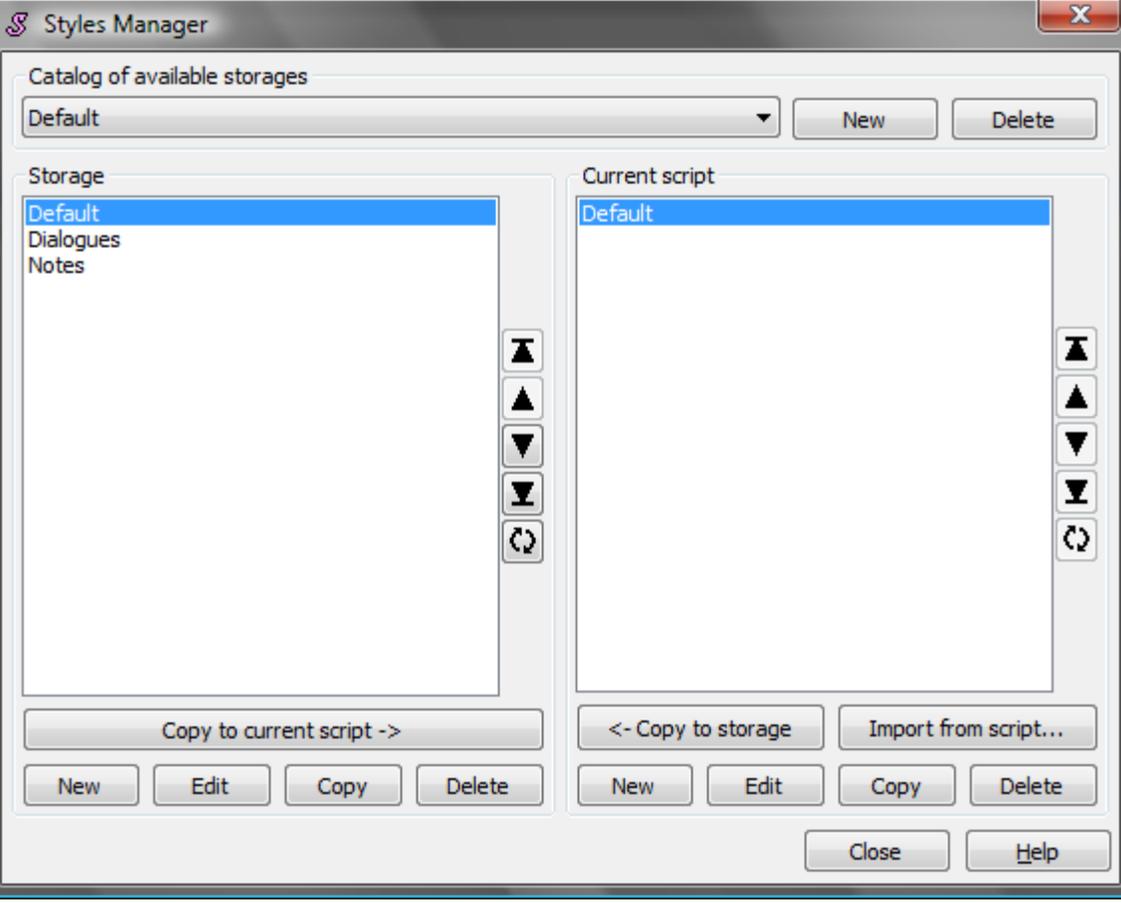
- Set selection
 Add to selection
 Subtract from selection
 Intersect with selection

OK

Cancel

Help





Style Editor

Style name

Dialogues

Font

Bell Gothic Std Black 20

Bold Italic Underline Strikeout

Colors

Primary Secondary Outline Shadow



0 0 0 0

Margins

Left Right Vert

10 10 10

Alignment

7 8 9
 4 5 6
 1 2 3

Outline

Outline: 2 Shadow: 2 Opaque box

Miscellaneous

Scale X%: 100 Scale Y%: 100

Rotation: 0 Spacing: 0

Encoding: 0 - ANSI

Preview



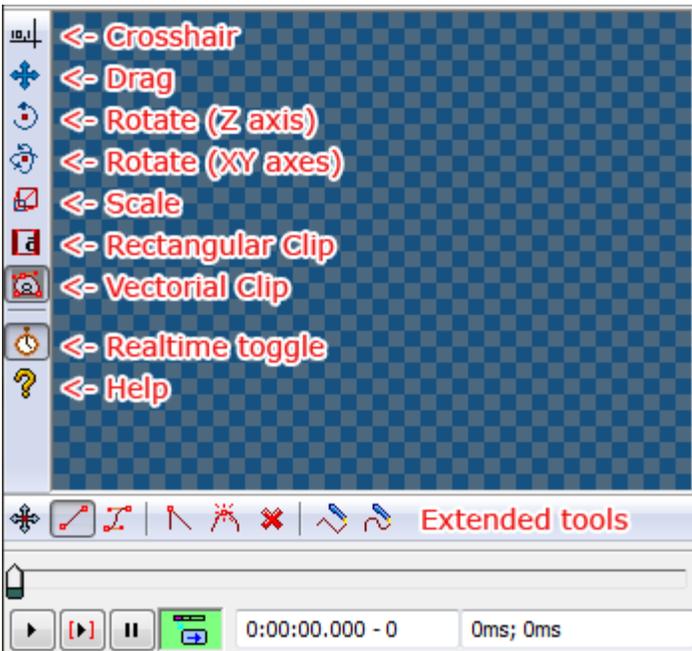
Aegisub\N0123 月語

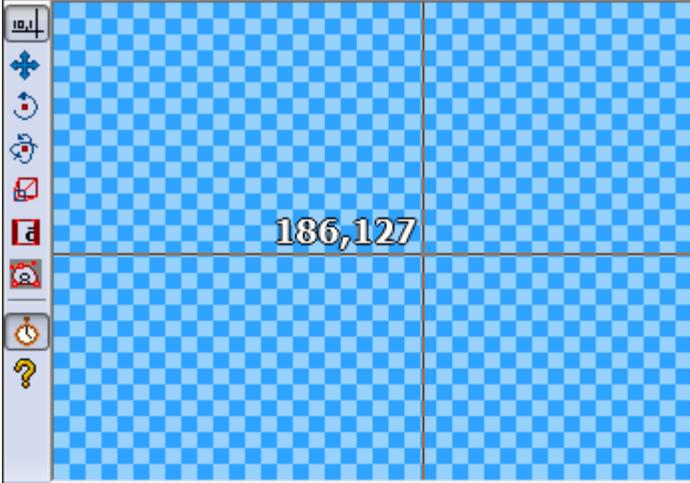
OK

Cancel

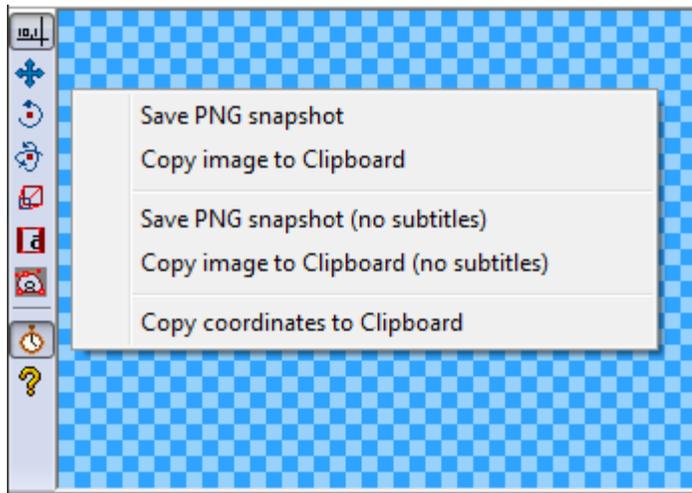
Apply

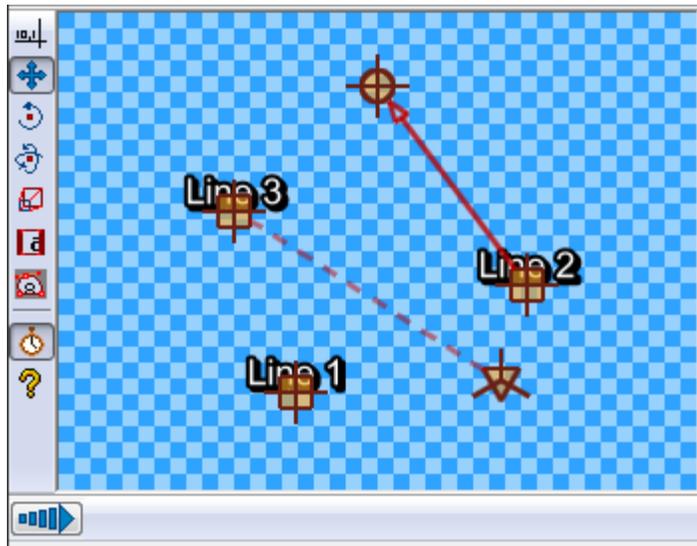
Help

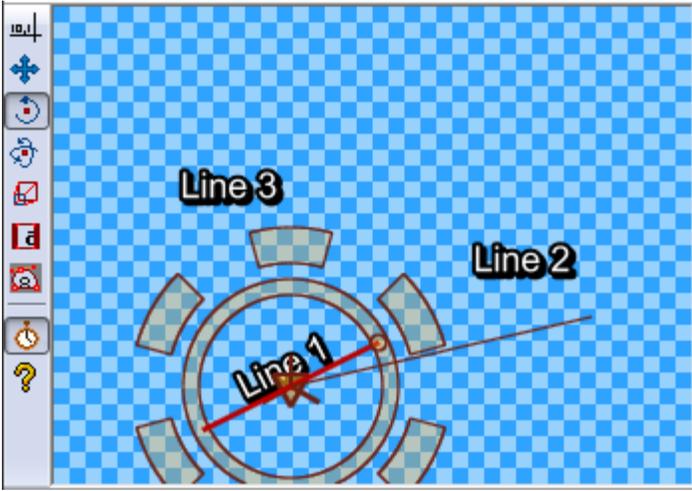


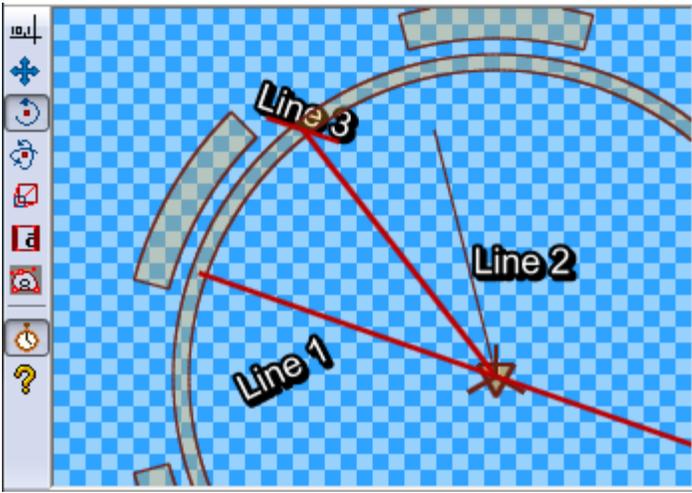


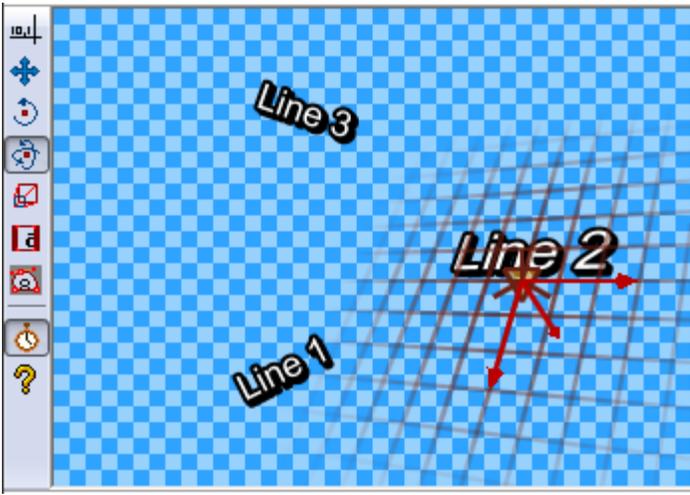
186,127

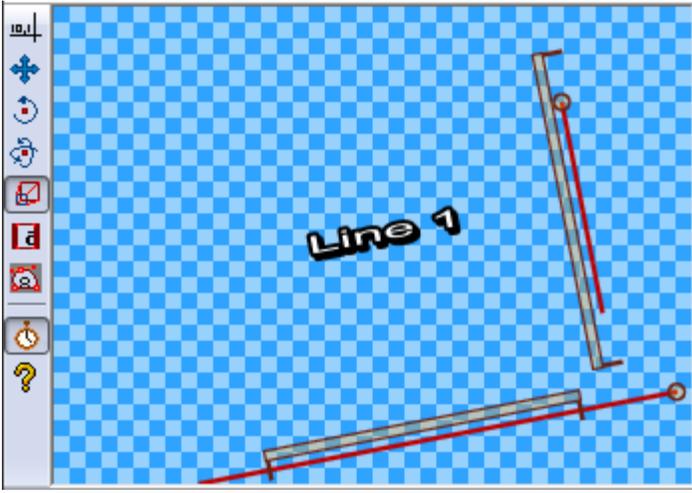




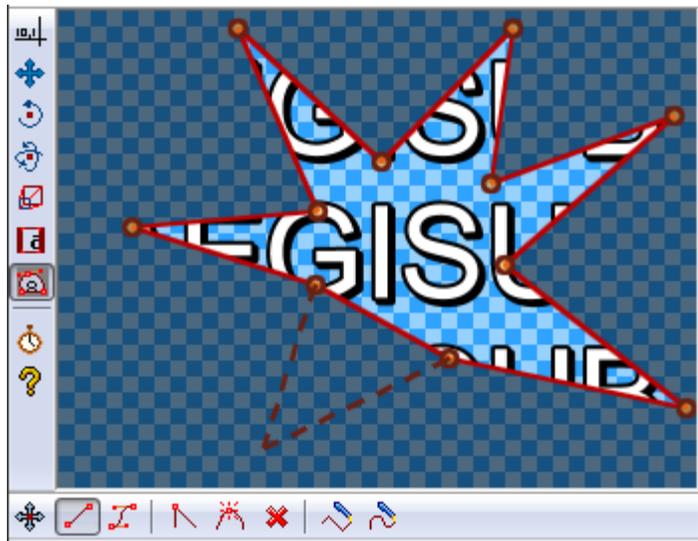














lfz45

Vrx60

Vfry60

AB AB AB AB

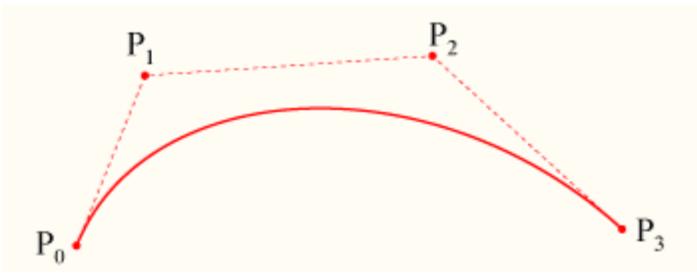






`\pos(320,240) with \an3 sample`





Select Colour



Colour spectrum

Spectrum mode: HSV/H



RGB colour

Red: 7

ASS: &HA58307&

Green: 131

HTML: #0783A5

Blue: 165

HSL colour

Hue: 137

Sat.: 234

Lum.: 86

HSV colour

Hue: 137

Sat.: 244

Value: 165



OK

Cancel

Help

Styling assistant

Current line

Is that all?

Styles available

- Standard (KGNE)
- Takayuki (KGNE)
- Hayase (KGNE)
- Haruka (KGNE)
- Shinji (KGNE)
- Akane (KGNE)
- Daikuuji (KGNE)
- Mayu (KGNE)
- Ishida (KGNE)
- Manager (KGNE)
- Doctor (KGNE)
- Tall Nurse (KGNE)
- Short Nurse (KGNE)
- Notes (KGNE)
- F.O. (KGNE)
- Alternate (KGNE)

Set style

Hayase (KGNE)

Keys

Enter:	Accept changes
F8:	Preview changes
PgUp:	Previous line
PgDn:	Next line
End:	Play Audio
Click on list:	Select style

Enable preview (slow)

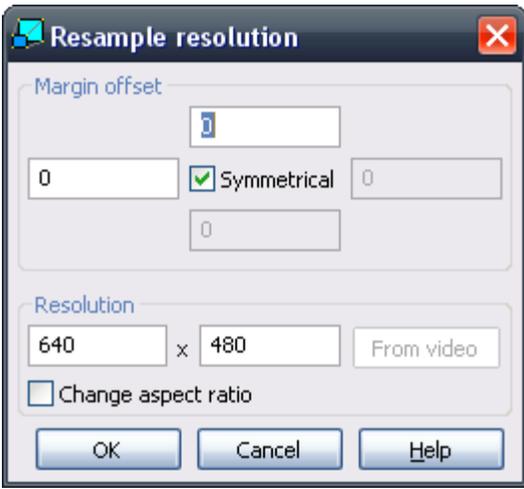
Play Audio Cancel Help

Set style

Hayase (KGNE)

Set style

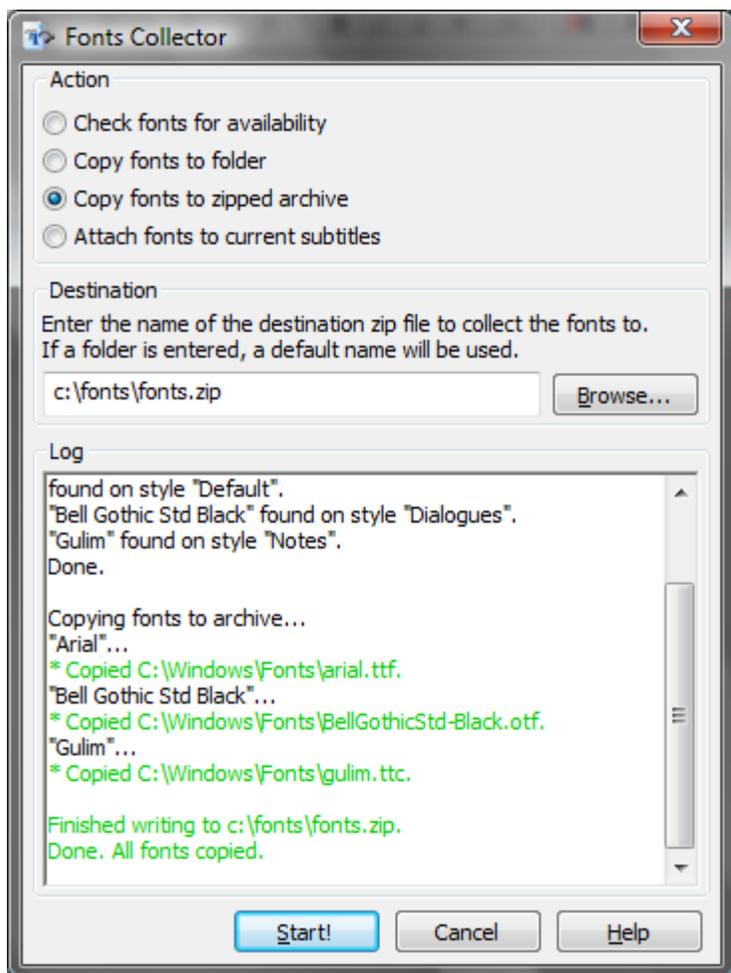
Hayo



16:9 ($\text{OrigW} * 16/9$) x OrigH



Left margin offset



Timing Post-Processor



Apply to styles

- Roomaji
- English
- Kanji
- Default

All None

Options

Affect selection only

Lead-in/Lead-out

Add lead in: 200 Add lead out: 300

Make adjacent subtitles continuous

Enable Threshold: 300 Bias: Start <- -> End

Keyframe snapping

Enable Starts before thres.: 5 Starts after thres.: 4
Ends before thres.: 5 Ends after thres.: 6

OK Cancel Help

 Kanji timing ✕

Text

Source: ni hare

Dest: なれ

Shortcut Keys

When the destination textbox has focus, use the following keys:

- Right Arrow: Increase dest. selection length
- Left Arrow: Decrease dest. selection length
- Up Arrow: Increase source selection length
- Down Arrow: Decrease source selection length
- Enter: Link, accept line when done
- Backspace: Unlink last

Attempt to interpolate kanji.

Groups

shou	少
nen	年
yo	よ
shin	神
wa	話

Styles

Eva Op Roomaji ▼

Eva Op Kanji ▼

Commands

Start!

Link

Unlink

Skip Source Line

Skip Dest Line

Go Back a Line

Accept Line

Close Help

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

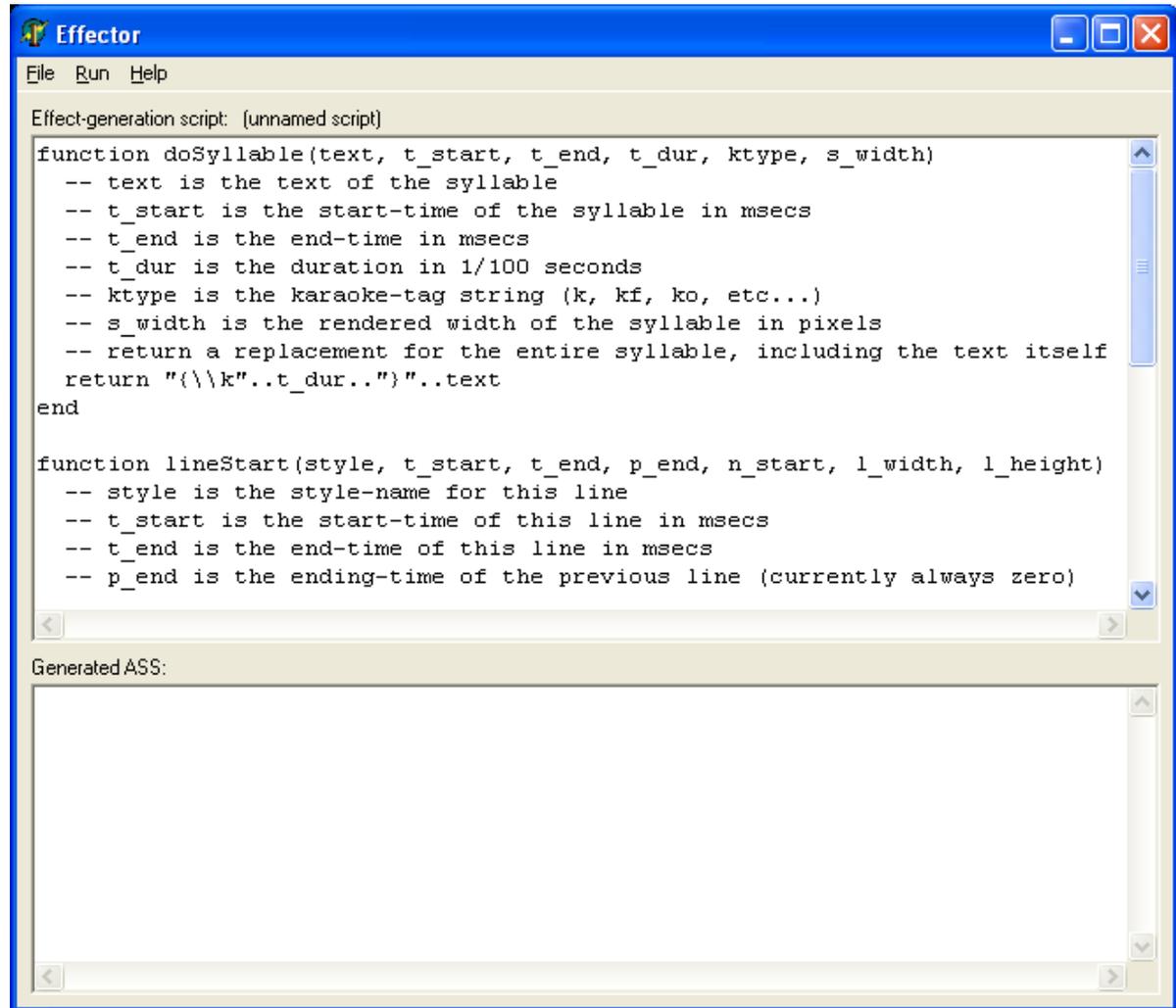
» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

File:Effector screenshot.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Effector_screenshot.png](#) (664 × 574 pixels, file size: 17 KB, MIME type: image/png)

Screenshot of the old Karaoke Effector program, the precursor of Aegisub Automation.

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail

Dimensions User Comment

current contribs) (Screenshot of the old Karaoke Effector program, the precursor of Aegisub Automation.)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

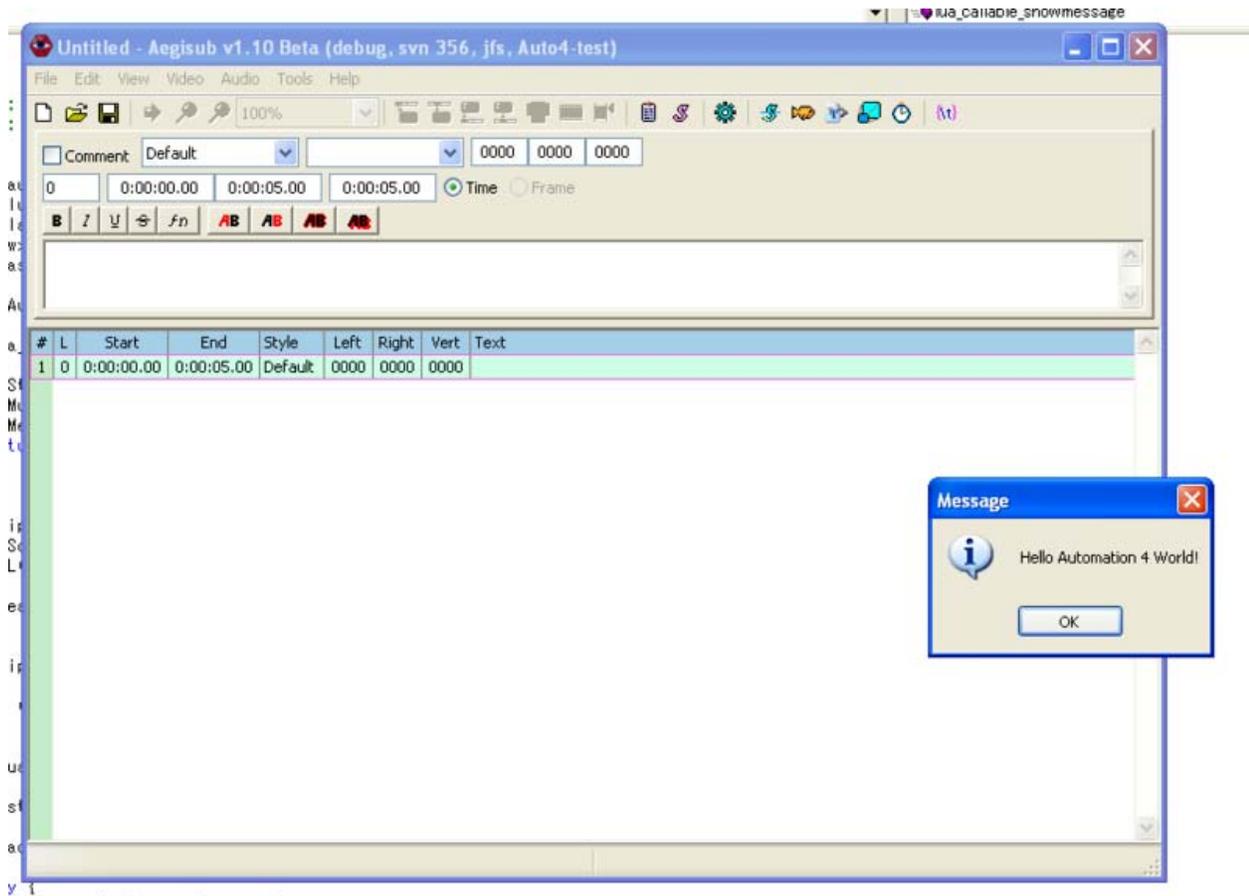
File links

The following page links to this file:

- [Sidebar:History of Automation](#)

File:Hello-auto4.png

- File
- File history
- File links



Size of this preview: 800 × 574 pixels
Full resolution (869 × 623 pixels, file size: 30 KB, MIME type: image/png)

The first screenshot of Automation 4 "working". Originally dated 2006-05-16.

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail

Dimensions User Comment

current contribs) (The first screenshot of Automation 4 "working". Originally dated 2006-05-16.)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Sidebar:History of Automation](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:StackedSign1.png

- File
- File history
- File links



No higher resolution available.

StackedSign1.png (100 × 200 pixels, file size: 2 KB, MIME type: image/png)

Rendered with the ASS code:

```
{\fn@DFPGothic-EB\fs26\shad0\fe128\bord3\3c&H25485A&\c&HDEEBF1&\pos(456,184)\frz-90}Sign text
```

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time **Thumbnail**

Dimensions **User** **Comment**

	(Rendered with the ASS code: {\fn@DFPGothic-current contribs) EB\fs26\shad0\fe128\bord3\3c&H25485A&\c&HDEEBF1&\pos(456,184)\frz-90}Sign text)			
--	---	--	--	--

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Using the included macros](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:StackedSign2.png

- File
- File history
- File links



No higher resolution available.

StackedSign2.png (100 × 250 pixels, file size: 2 KB, MIME type: image/png)

Rendered with the ASS code:

```
{\fn@DFPGothic-EB\fs26\shad0\fe128\bord3\3c&H25485A&\c&HDEEBF1&\pos(456,184)\frz-90}Sign text
```

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail

Dimensions User Comment

(Rendered with the ASS code: {\fn@DFPGothic-EB\fs26\shad0\fe128\bord3\3c&H25485A&\c&HDEEBF1&\pos(456,184)\frz-90}Sign text)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Using the included macros](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Automation-toolbar-icon.png

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Automation-toolbar-icon.png](#) (16 × 15 pixels, file size: 235 B, MIME type: image/png)

The Automation toolbar icon.

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail

Dimensions User Comment

current contribs) (The Automation toolbar icon.)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#)  for more information)

File links

The following page links to this file:

- [Automation Manager](#)

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

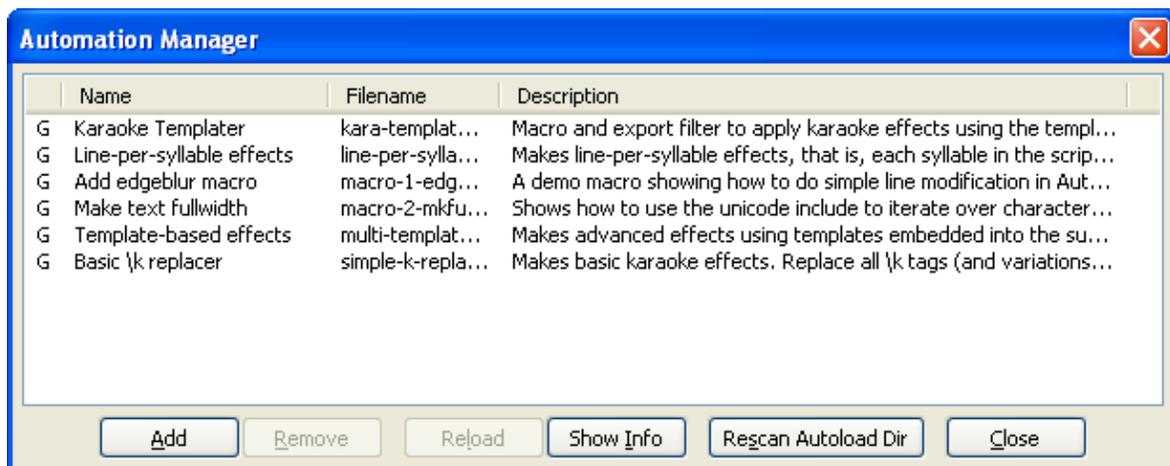
» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

File:Auto4-manager.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Auto4-manager.png](#) (616 × 245 pixels, file size: 11 KB, MIME type: image/png)

The Automation Manager window in Automation 4, with several autoload-scripts loaded.

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time Thumbnail

Dimensions User Comment

(The Automation Manager window in current contribs) [Automation 4, with several autoload-scripts loaded.](#)

(Latest | Earliest) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Automation Manager](#)

Glossary:Automation script feature

In Automation 4 a **feature** is a function a script makes available to Aegisub. Currently two kinds of features are specified: Export filters and macros.

If a script defines two macros and one export filter, it is said to have defined three features in total.

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

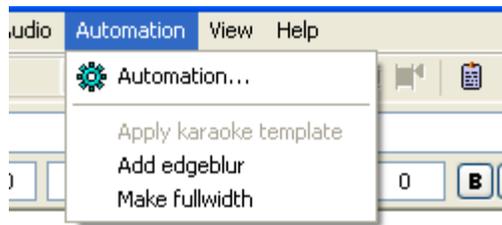
- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Auto4-menu.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Auto4-menu.png](#) (252 × 127 pixels, file size: 2 KB, MIME type: image/png)

The Automation menu showing three macros. The first is unavailable because the [Karaoke Templater](#) script has determined that there are no karaoke templates in the subtitle file.

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time	Thumbnail	Dimensions	User	Comment
current contribs)				(The Automation menu showing three macros. The first is unavailable because the Karaoke Templater script has determined that there are no karaoke templates in the subtitle file.)

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [↗](#) for more information)

File links

The following page links to this file:

- [Running Automation macros](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

File:Auto4-kara-templater-ptime-explanation.png

- [File](#)
- [File history](#)
- [File links](#)

ptime(mode, addstart, addend)

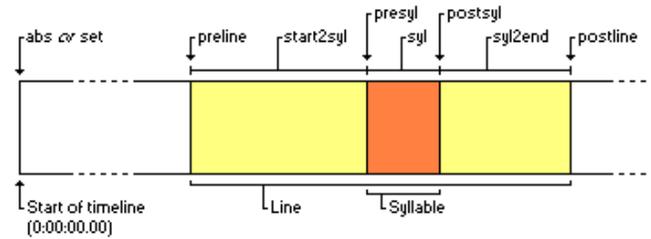
Change the timing of the generated line. The *mode* parameter controls what the new timing is relative to.

The *addstart* and *addend* parameters adjust the start and ending time additionally.

The diagram to the right shows the possible values for *mode* and what they mean. For those values described with an arrow, the start and end times of the new line will be equal unless you also adjust them with *addstart* and *addend*.

There is one additional value for *mode*, "sylpot". That one interprets *addstart* and *addend* as percentages instead. Specifically it does:

```
start = sylstart + addstart*sylldur/100  
end = sylstart + addend*sylldur/100
```



No higher resolution available.

[Auto4-kara-templater-ptime-explanation.png](#) (654 × 198 pixels, file size: 11 KB, MIME type: image/png)

A graphical explanation of the different modes of [the ptime function](#) in Karaoke Templater.

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time Thumbnail

Dimensions User Comment

current contribs) (A graphical explanation of the different modes of [the ptime function](#) in Karaoke Templater.)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Karaoke Templater Reference: Code execution environment](#)

File:Furigana-demo-1.png

- File
- File history
- File links



No higher resolution available.

[Furigana-demo-1.png](#) (256 × 72 pixels, file size: 9 KB, MIME type: image/png)

Rendered with [kara-templater](#):

```
Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template
syl,{\pos(!line.left+syl.center!,!line.middle!)\an5\k!syl.start_time/10!\k$kdur}
Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template furi,{\pos(!line.left+syl.center!,!line.middle-
line.height!)\an5\k!syl.start_time/10!\k$kdur}
Comment:
0,0:00:00.00,0:00:02.00,Default,,0000,0000,0000,karaoke,{\k15}二|ふ{\k15}#|た{\k10}人|リ{\k15}だ{\k57}け{\k5}の{\k6}地|ほ
{\k5}球|し{\k8}で
```

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time	Thumbnail	Dimensions	User	Comment
current			contribs	(Rendered with kara-templater : <pre>Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template syl,{\pos(!line.left+syl.center!,!line.middle!)\an5\k!syl.start_time/10!\k\$kdur} Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template furi,{\)

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- [Edit this file using an external application](#) (See the [setup instructions](#) [for more information](#))

File links

The following page links to this file:

- [Furigana karaoke](#)

Navigation

- » [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Furigana-demo-4.png

Navigation

- » Main Page

Introduction

- » What is Aegisub?
- » Highlights
- » Credits
- » Support Aegisub
- » FAQ
- » Tutorials

Working with Subtitles

- » Editing Subtitles
- » Exporting Subtitles
- » Applying Subtitles
- » Spell Checker
- » Translation Assistant
- » Paste Over
- » Select Lines

Typesetting

- » Introduction
- » Working with Video
- » Editing styles
- » Visual Typesetting
- » ASS Override Tags
- » Colour Picker
- » Styling Assistant
- » Resolution Resampler
- » Fonts Collector

Timing

- » Working with Audio
- » Shift times
- » Timing Post-Processor
- » Kanji Timer

Automation

- » Overview
- » Karaoke Templater
- » Lua Reference

Miscellaneous

- » Aegisub Options
- » Script Properties
- » Attachment Manager

- File
- File history
- File links



No higher resolution available.

Furigana-demo-4.png (256 × 72 pixels, file size: 6 KB, MIME type: image/png)

Rendered with kara-templater:

```

Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template
syl,{\pos(!line.left+syl.center!,!line.middle!)\an5\k!syl.start_time/10!\k$kdur}
Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template furi,{\pos(!line.left+syl.center!,!line.middle-
line.height!)\an5\k!syl.start_time/10!\k$kdur}
Comment:
0,0:00:06.00,0:00:08.00,Default,,0000,0000,0000,karaoke,{\k10}中|ちゆ{\k10}#|う{\k10}国|ご{\k10}#|<{\k10}魂|た{\k10}#|ま
{\k10}#|し{\k10}#|い

```

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time	Thumbnail	Dimensions	User	Comment
current	contribs			(Rendered with kara-templater: <pre>Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template syl,{\pos(!line.left+syl.center!,!line.middle!)\an5\k!syl.start_time/10!\k\$kdur} Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template furi,{\

(Latest | Earliest) View (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

- Edit this file using an external application (See the [setup instructions](#) for more information)

File links

The following page links to this file:

- [Furigana karaoke](#)

Navigation

» [Main Page](#)

Introduction

» [What is Aegisub?](#)
» [Highlights](#)
» [Credits](#)
» [Support Aegisub](#)
» [FAQ](#)
» [Tutorials](#)

Working with Subtitles

» [Editing Subtitles](#)
» [Exporting Subtitles](#)
» [Applying Subtitles](#)
» [Spell Checker](#)
» [Translation Assistant](#)
» [Paste Over](#)
» [Select Lines](#)

Typesetting

» [Introduction](#)
» [Working with Video](#)
» [Editing styles](#)
» [Visual Typesetting](#)
» [ASS Override Tags](#)
» [Colour Picker](#)
» [Styling Assistant](#)
» [Resolution Resampler](#)
» [Fonts Collector](#)

Timing

» [Working with Audio](#)
» [Shift times](#)
» [Timing Post-Processor](#)
» [Kanji Timer](#)

Automation

» [Overview](#)
» [Karaoke Templater](#)
» [Lua Reference](#)

Miscellaneous

» [Aegisub Options](#)
» [Script Properties](#)
» [Attachment Manager](#)

File:Furigana-demo-3.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Furigana-demo-3.png](#) (256 × 72 pixels, file size: 6 KB, MIME type: image/png)

Rendered with [kara-templater](#):

```
Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template
syl,{\pos(!line.left+syl.center!,!line.middle!)\an5\k!syl.start_time/10!\k$kdur}
Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template furi,{\pos(!line.left+syl.center!,!line.middle-
line.height!)\an5\k!syl.start_time/10!\k$kdur}
Comment:
0,0:00:04.00,0:00:06.00,Default,,0000,0000,0000,karaoke,{\k10}中|ちゆ{\k10}#|う{\k10}国|ご{\k10}#|<{\k10}魂|!た{\k10}#|
ま{\k10}#|し{\k10}#|い
```

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

Date/Time	Thumbnail	Dimensions	User	Comment
current	contribs			(Rendered with kara-templater : <pre>Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template syl,{\pos(!line.left+syl.center!,!line.middle!)\an5\k!syl.start_time/10!\k\$kdur} Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template furi,{\

(Latest | [Earliest](#)) [View](#) ([newer 50](#)) ([older 50](#)) ([20](#) | [50](#) | [100](#) | [250](#) | [500](#))

- [Edit this file using an external application](#) (See the [setup instructions](#) [for more information](#))

File links

The following page links to this file:

- [Furigana karaoke](#)

Navigation

» [Main Page](#)

Introduction

- » [What is Aegisub?](#)
- » [Highlights](#)
- » [Credits](#)
- » [Support Aegisub](#)
- » [FAQ](#)
- » [Tutorials](#)

Working with Subtitles

- » [Editing Subtitles](#)
- » [Exporting Subtitles](#)
- » [Applying Subtitles](#)
- » [Spell Checker](#)
- » [Translation Assistant](#)
- » [Paste Over](#)
- » [Select Lines](#)

Typesetting

- » [Introduction](#)
- » [Working with Video](#)
- » [Editing styles](#)
- » [Visual Typesetting](#)
- » [ASS Override Tags](#)
- » [Colour Picker](#)
- » [Styling Assistant](#)
- » [Resolution Resampler](#)
- » [Fonts Collector](#)

Timing

- » [Working with Audio](#)
- » [Shift times](#)
- » [Timing Post-Processor](#)
- » [Kanji Timer](#)

Automation

- » [Overview](#)
- » [Karaoke Templater](#)
- » [Lua Reference](#)

Miscellaneous

- » [Aegisub Options](#)
- » [Script Properties](#)
- » [Attachment Manager](#)

File:Furigana-demo-2.png

- [File](#)
- [File history](#)
- [File links](#)



No higher resolution available.

[Furigana-demo-2.png](#) (256 × 72 pixels, file size: 6 KB, MIME type: image/png)

Rendered with kara-templater:

```
Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template
syl,{\pos(!line.left+syl.center!,!line.middle!)\an5\k!syl.start_time/10!\k$kdur}
Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template furi,{\pos(!line.left+syl.center!,!line.middle-
line.height!)\an5\k!syl.start_time/10!\k$kdur}
Comment:
0,0:00:02.00,0:00:04.00,Default,,0000,0000,0000,karaoke,{\k10}中|ちゆ{\k10}#|う{\k10}国|ご{\k10}#|<{\k10}魂|<た{\k10}#|
ま{\k10}#|し{\k10}#|い
```

File history

Click on a date/time to view the file as it appeared at that time.

(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

Date/Time	Thumbnail	Dimensions	User	Comment
current	contribs			(Rendered with kara-templater: <pre>Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template syl,{\pos(!line.left+syl.center!,!line.middle!)\an5\k!syl.start_time/10!\k\$kdur} Comment: 0,0:00:00.00,0:00:00.00,Default,,0000,0000,0000,template furi,{\

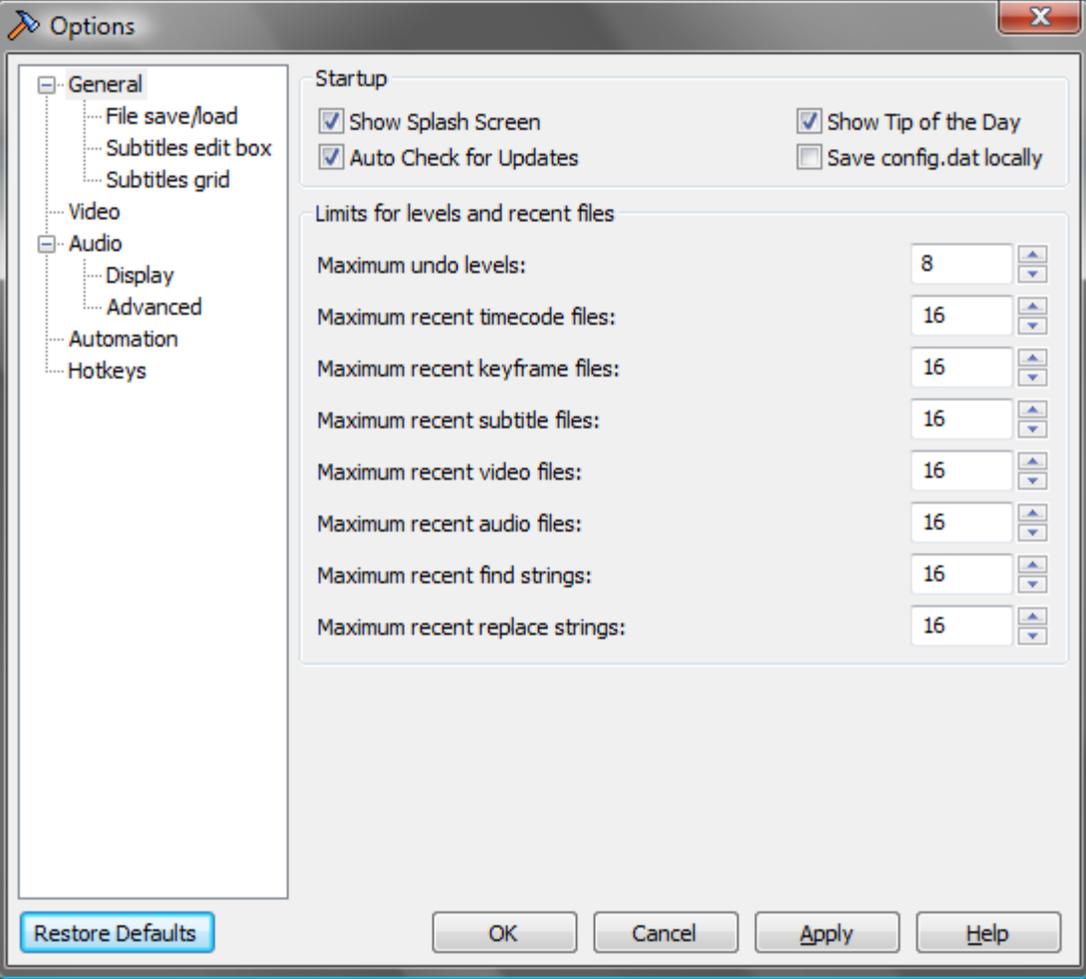
(Latest | Earliest) [View](#) (newer 50) (older 50) (20 | 50 | 100 | 250 | 500)

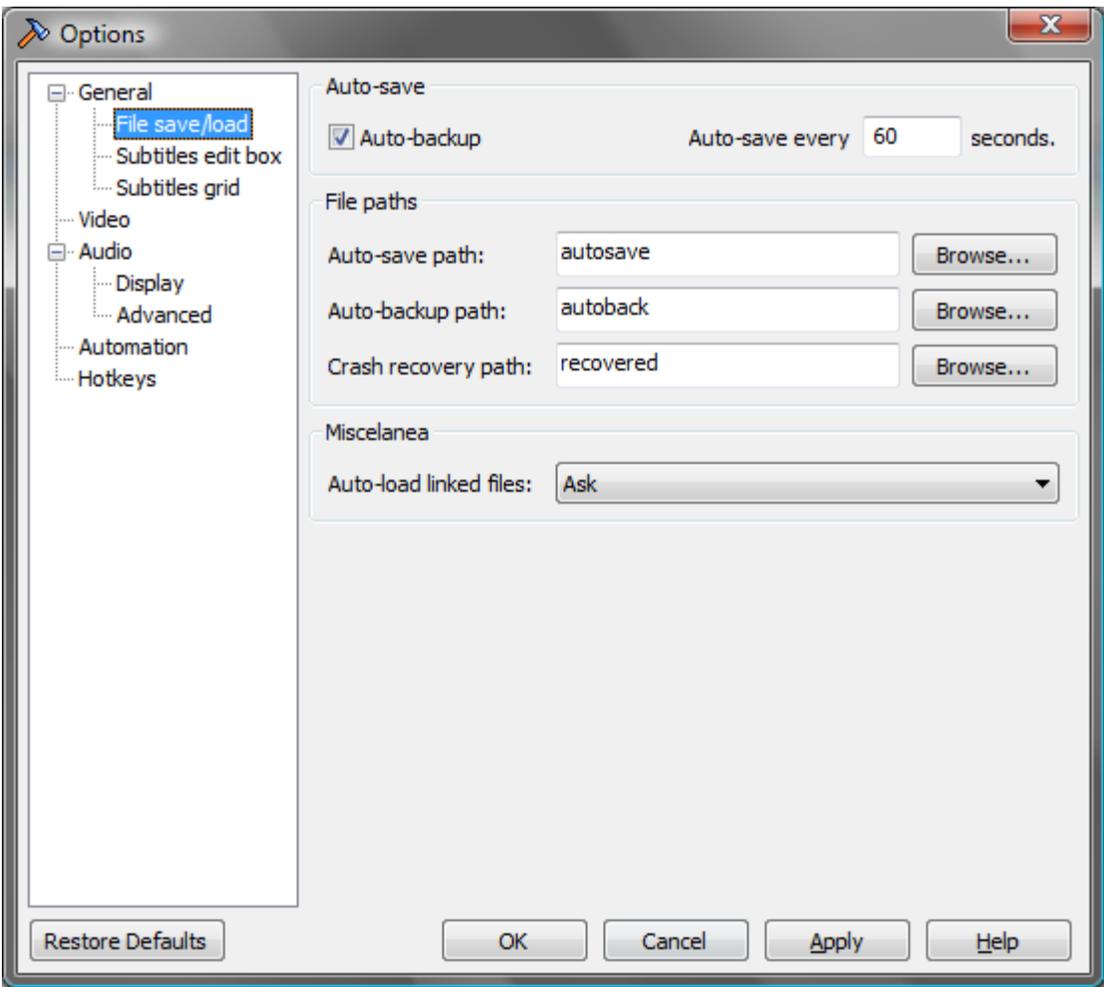
- [Edit this file using an external application](#) (See the [setup instructions](#) [for more information](#))

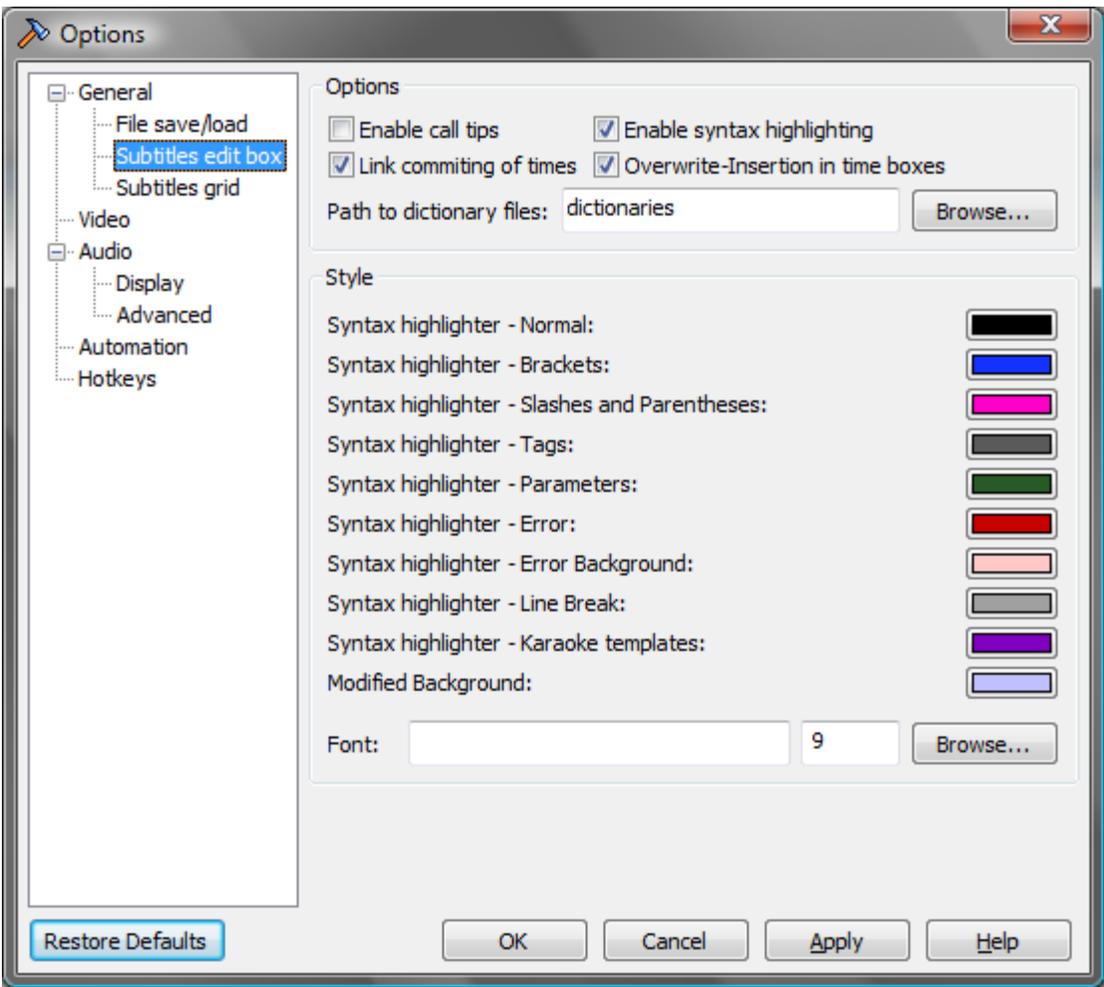
File links

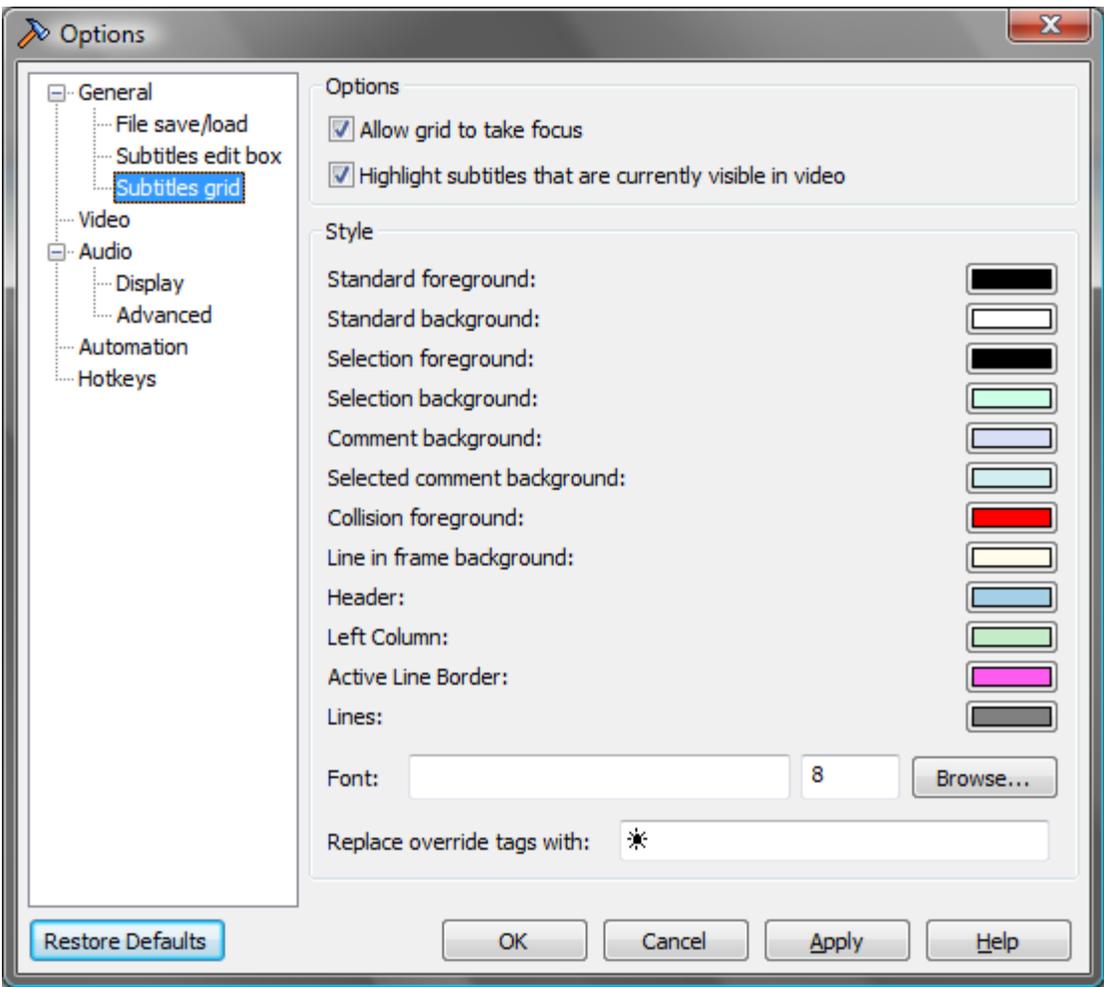
The following page links to this file:

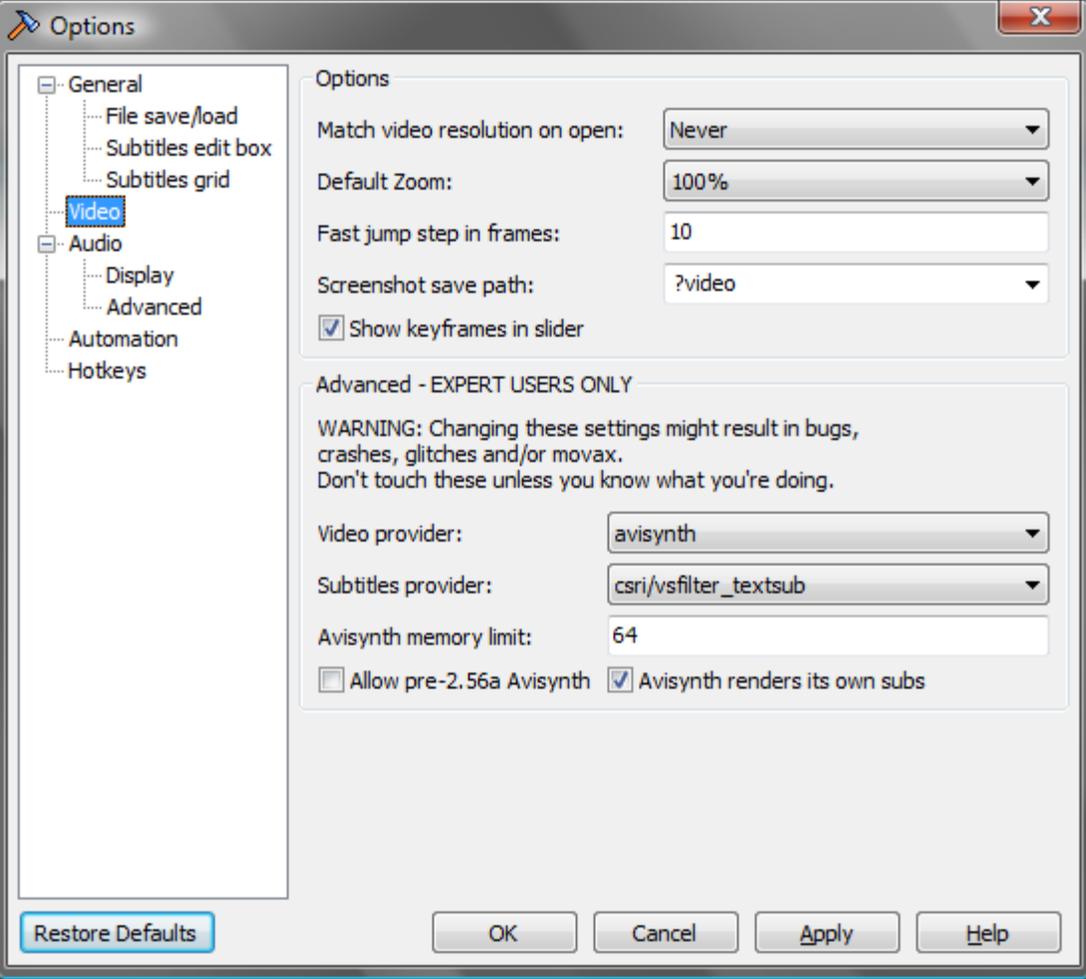
- [Furigana karaoke](#)

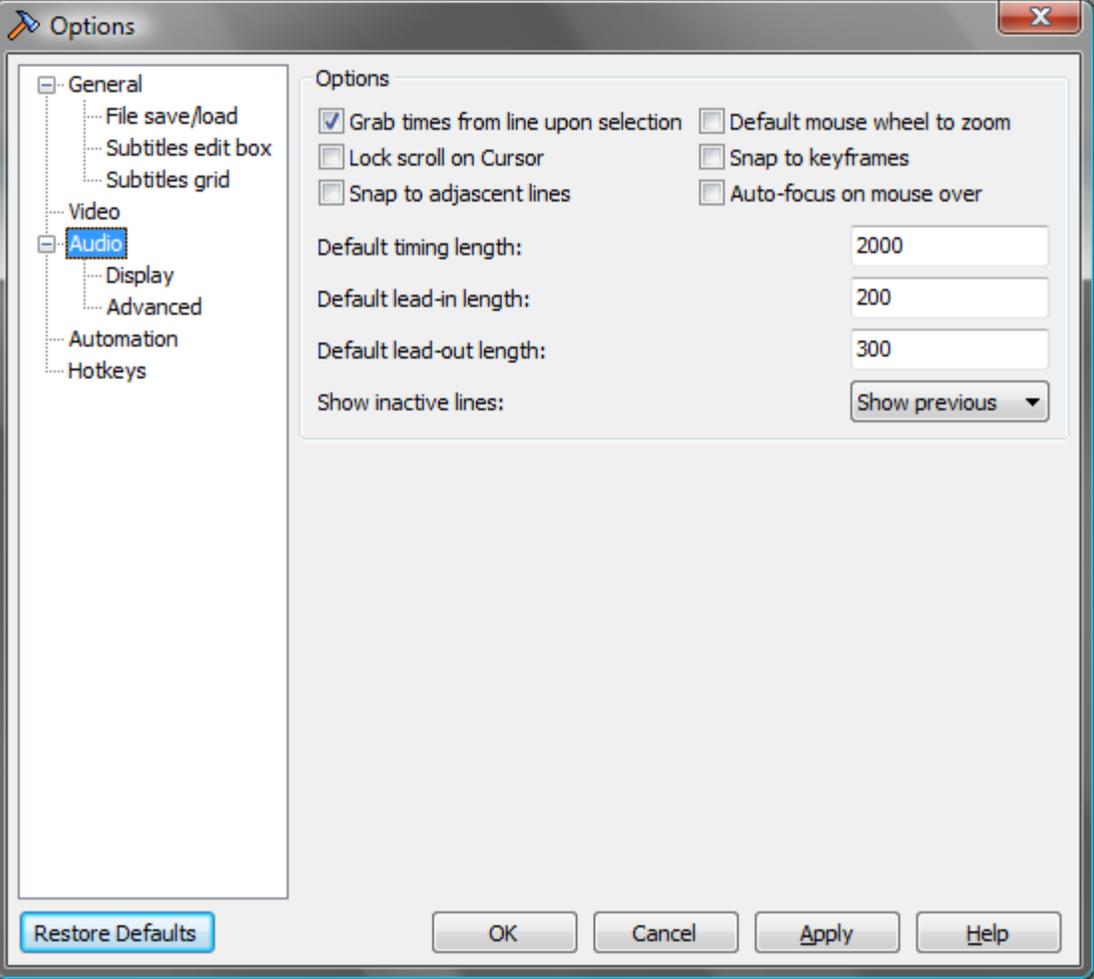


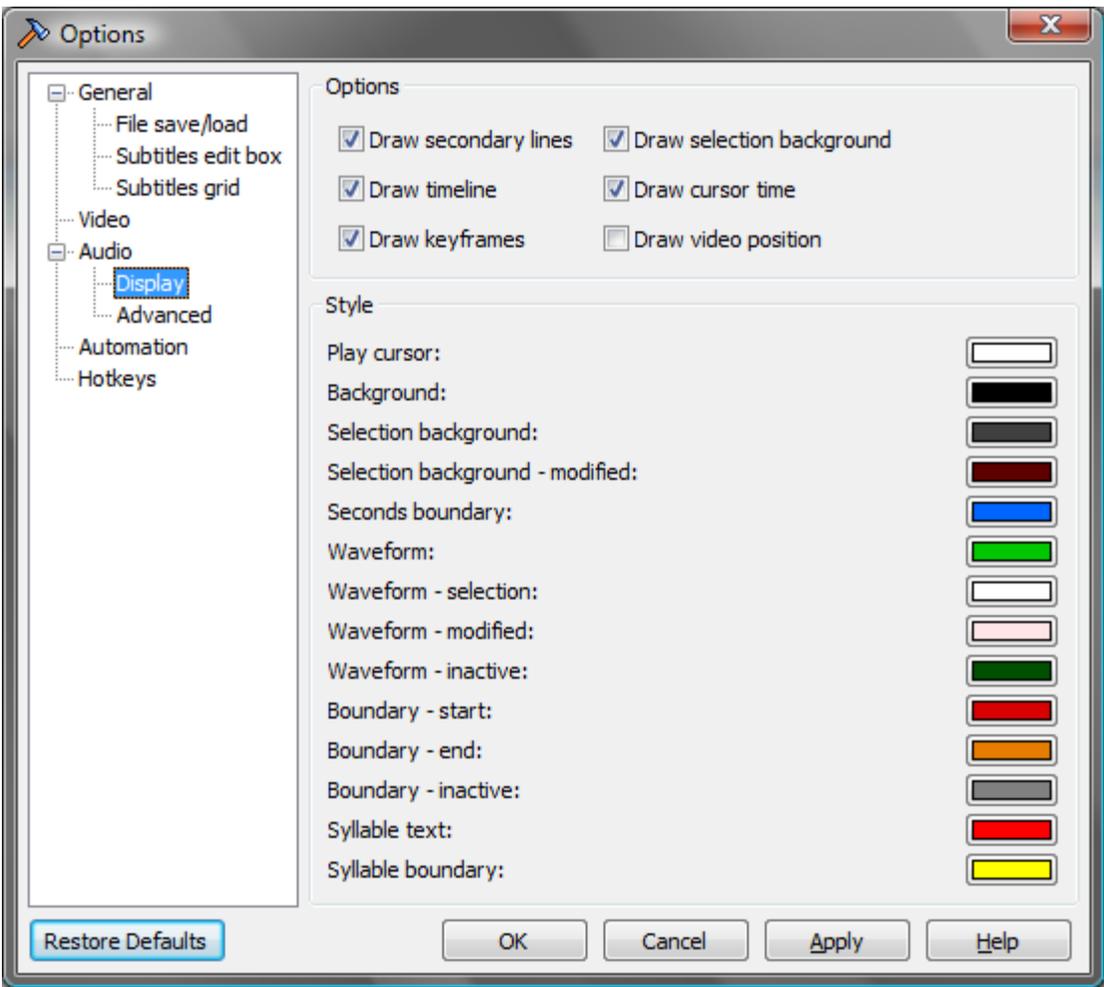


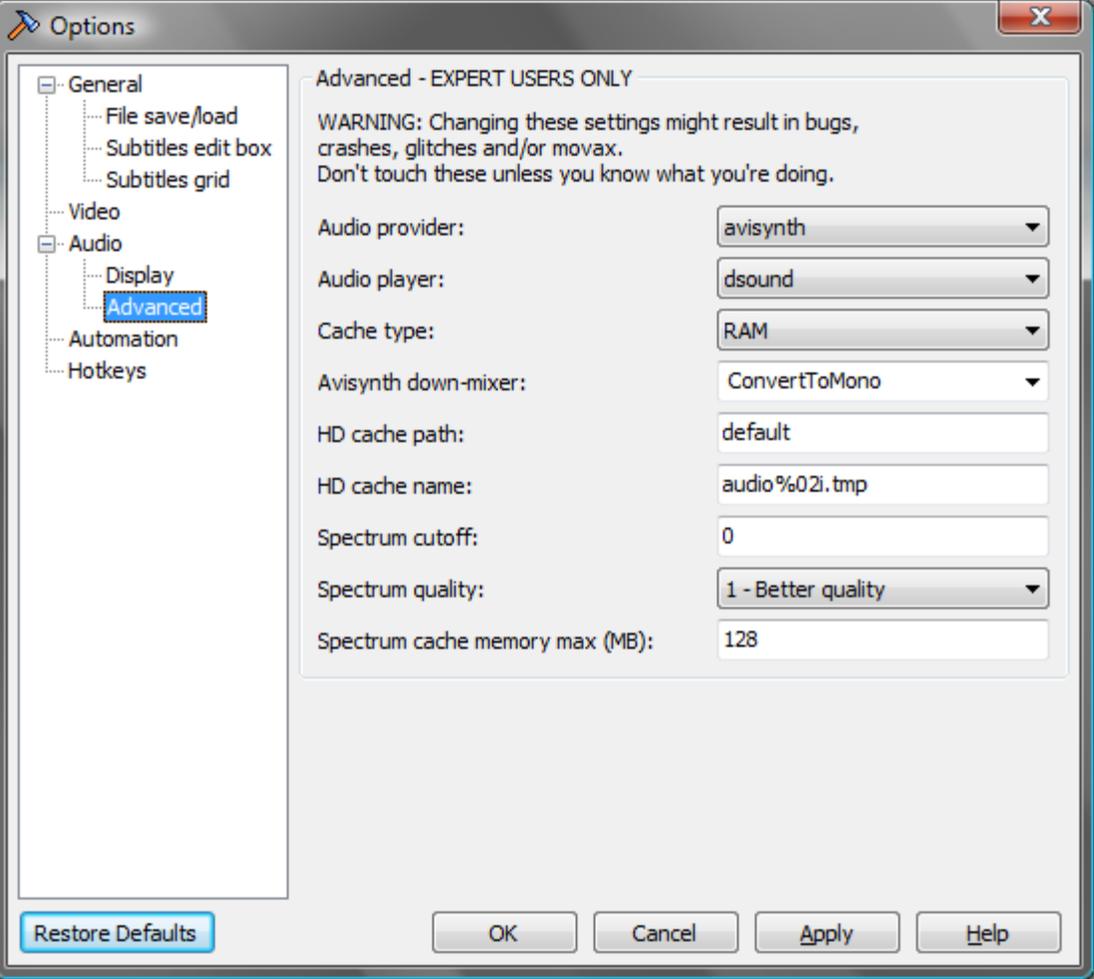


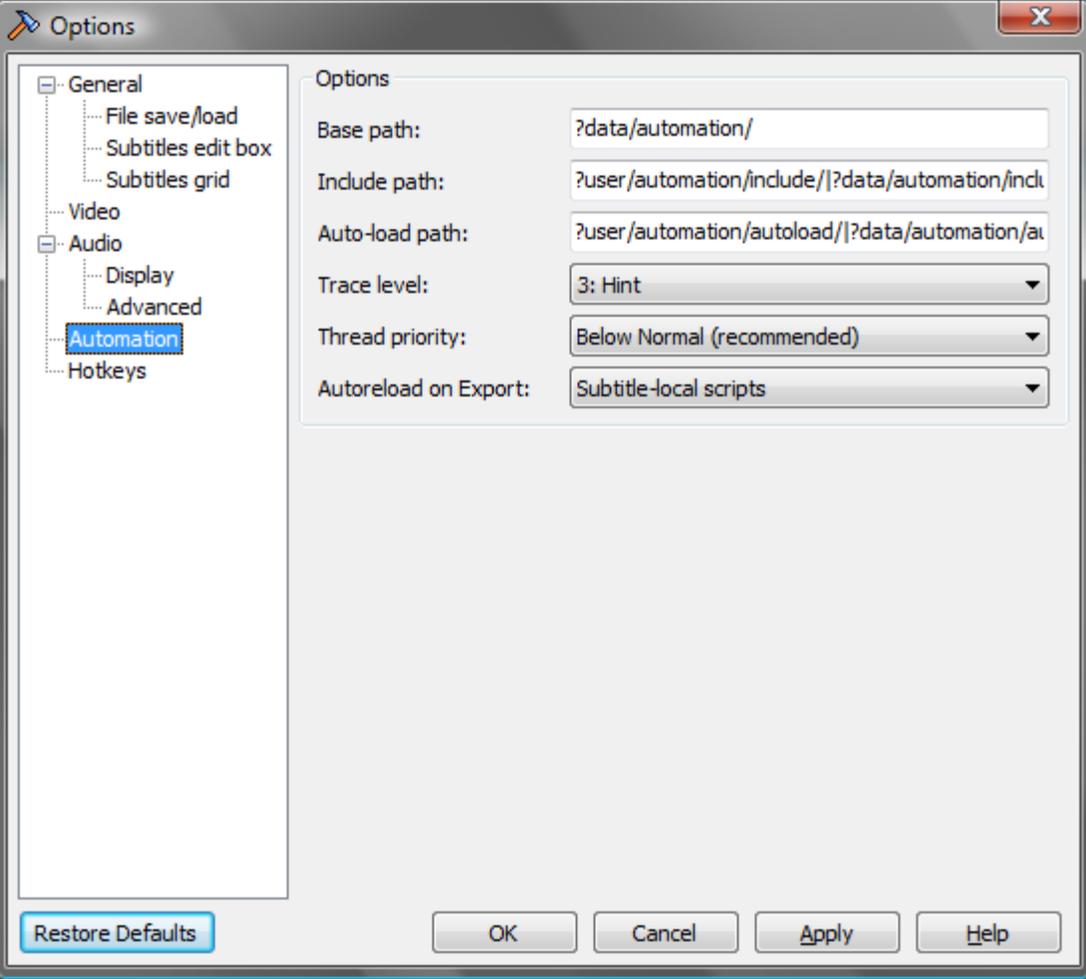


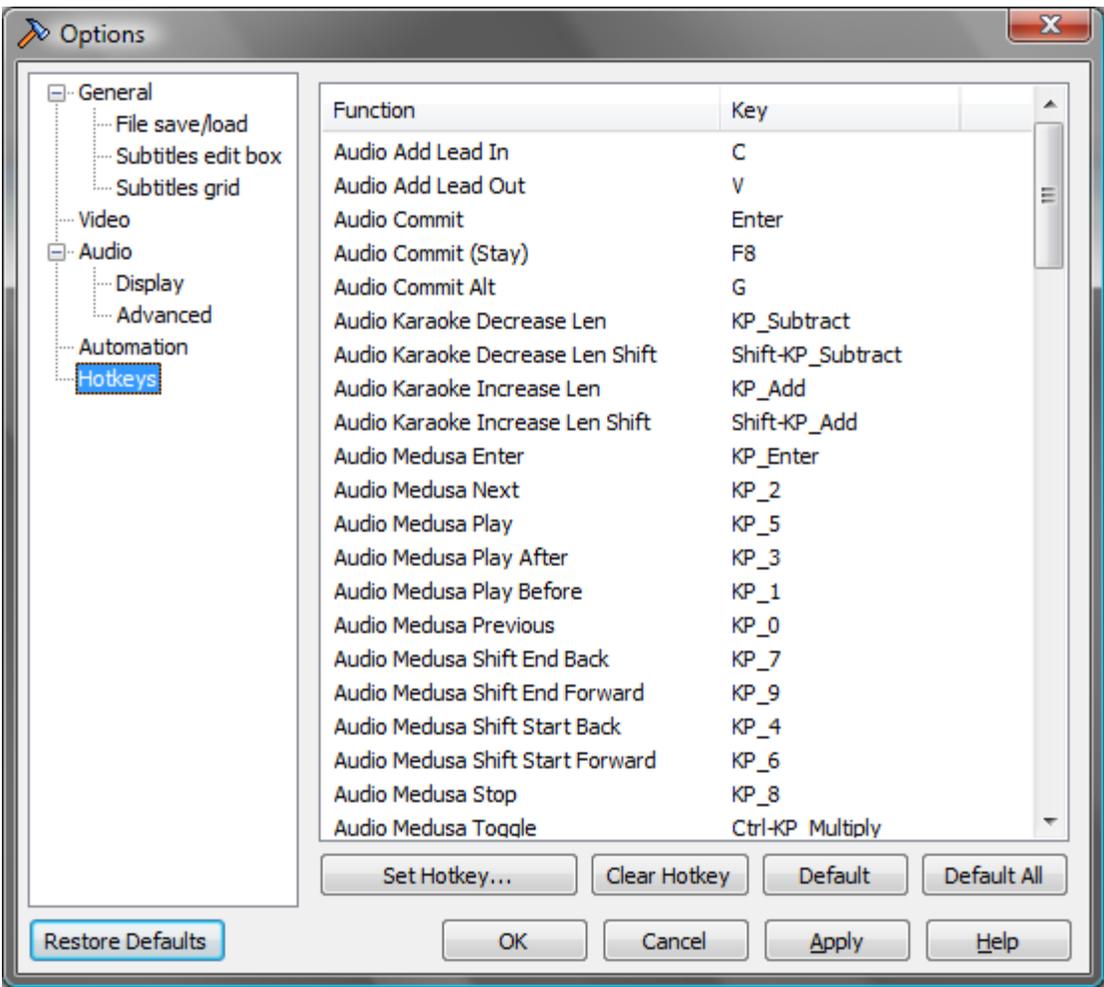












Script Properties

Script

Title:

Original script:

Translation:

Editing:

Timing:

Synch point:

Updated by:

Update details:

Resolution

x

Options

Wrap Style:

Collision:

Scale Border and Shadow

Attachment List



Attachment name	Size	Group
arial[1]_0.ttf	749 kB	[Fonts]
visual_vector_dip_drag.bmp	1.29 kB	[Graphics]

- Attach Font
- Attach Graphics
- Extract
- Delete
- Help
- Close