Mac YouTube-dl Video Downloader

Version 1.19.1 – 16 November 2021

MacYTDL – Mac YouTube-dl video downloader

Version 1.19.1 – 16 November 2021

What MacYTDL does

MacYTDL is a utility which downloads videos using the *[youtube-dl](#)* and *[YT-DLP](#)* video downloaders. *youtube-dl* and *YT-DLP* are able to download videos from a great many web sites. *YT–DLP* is the more capable tool and the default for users of macOS 10.15 and higher. *youtube-dl* is compatible with macOS versions 10.10 and higher.

Note: *youtube-dl* is no longer in active development. Also, *youtube-dl* cannot download videos from YouTube at full speed.

Features:

- Download individual and multiple videos, playlists and batches.

- Works with YouTube and many other sites including ITV and all Australian TV catchup sites.

- Offers choice of download file format from a range of video and audio formats.

- Optionally download and/or embed subtitles in chosen format (from six) and language.

- Optionally download a text description of the video.

- Optionally download or extract separate audio file in a range of formats.

- Can download many videos at one time i.e. in parallel, and multiple videos in succession in one process i.e. in series.

- Options to provide login credentials, proxy URL, cookies file and custom file name template (thanks to [tht7](#)).

- Download speed can be set for each download.

- Is localised – currently available in Italian (thanks to "[bovirus](#)"), Spanish (thanks to [michel-GH](#)) and German (thanks to Google Translate).

- Options to embed thumbnail images, subtitles and metadata in downloaded files.

- Facility to switch between *youtube-dl* and *YT-DLP*.

- An optional macOS Service which sends a URL in the correct text selection, the Clipboard or a web browser tab to MacYTDL.

- When download completed offers to play video in default player.

- External components downloaded and installed by MacYTDL, controlled by the user.

  - Can check for updates to MacYTDL and components.

  - *youtube-dl/YT-DLP* can be checked for updates automatically on startup.

- A separate log file is saved for each download enabling problem solving if a download fails.

- Has a built-in uninstaller which moves all components to Trash.

System requirements

MacYTDL will work on any Apple Mac running OS X 10.10 or later. *YT-DLP*, however, requires macOS 10.15 and higher. It can be used with earlier versions of macOS if two library files are sourced ([see below](#)).

MacYTDL is free

MacYTDL is a retirement project and will always be free.

How to install MacYTDL

Download and open the DMG file. Drag the MacYTDL applet to any location you prefer. Placing it in the *Applications* folder will make it available for all user accounts. It's best to delete old copies of MacYTDL – just move old copies to the Trash.

The applet is code signed and notarized so, you may see messages when you first run MacYTDL. The range of messages varies with the version of macOS being used. In macOS Mojave and Catalina this message is shown when MacYTDL is first run:
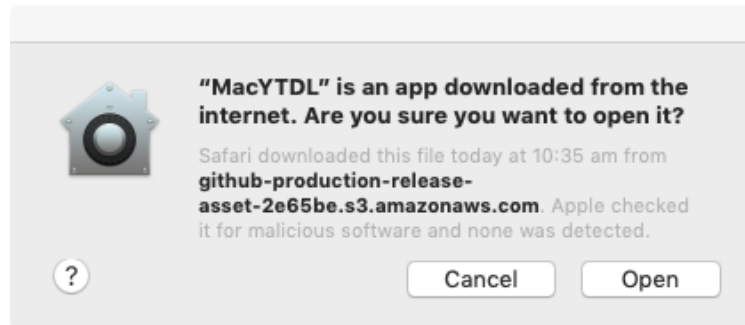
Click on "Open".

Detail on code signing and notarization is <u>available below</u>.

If you see this message instead …
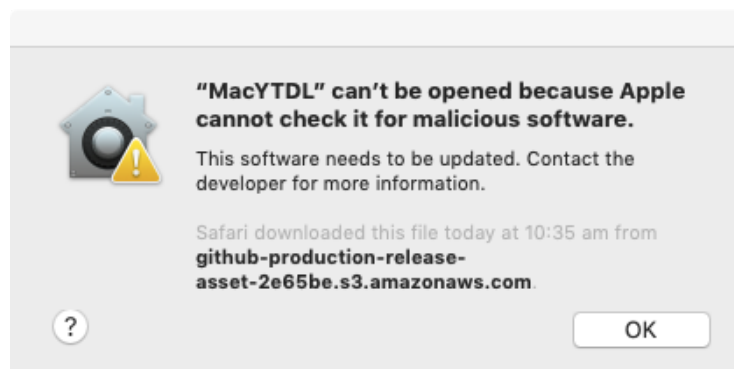
… you will need to change your Gatekeeper settings to allow applications from "identified developers". For information on Gatekeeper and how to change settings visit the Apple web site at: <u>https://support.apple.com/en-au/HT202491</u>.

Again depending on which version of macOS you are using, there may be a number of messages requesting permission for MacYTDL to control the Finder and System Events:
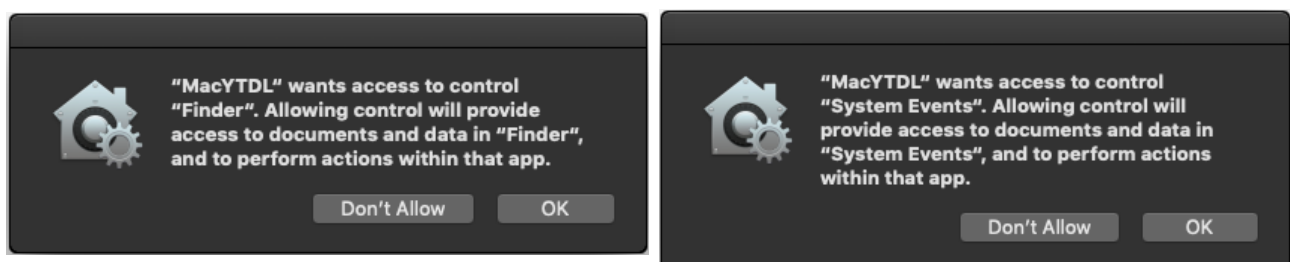
Alternatively, you may see these messages:
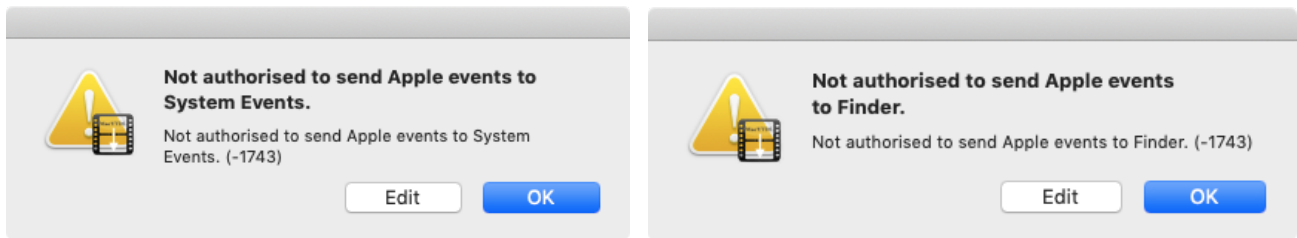
MacYTDL cannot function without those permissions, Click on "OK".

If you see the second kind of message or you click on "Don't Allow" on the first, you will need to give permission manually in System Preferences:

- go to *System Preferences* on your Mac;

- choose the "Security & Privacy" preference pane;

- choose the "Privacy" tab; and

- click on the "Automation" entry in the left hand sidebar.

If the two boxes under "MacYTDL" are not ticked:

- click on the lock symbol;

- enter an administrator user name and password; and

- place a tick in each box.

Your permissions should look like these:
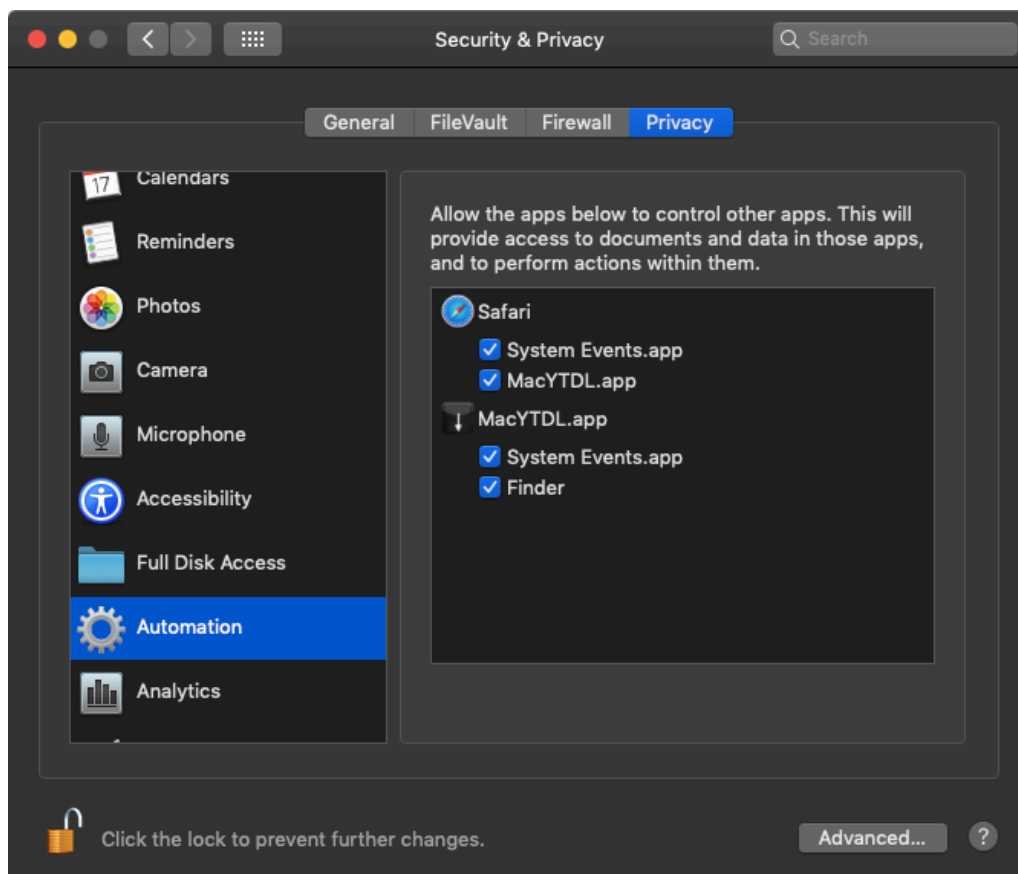
In the above example, there are two tick boxes under "Safari". They are required to use the [MacYTDL Service](#).

In macOS 10.14 and higher, the permission dialogs may be shown each time you open a new version of MacYTDL for the first time.

The first time you start MacYTDL you will also be advised that various components used by MacYTDL need to be installed:

- A new folder called "MacYTDL" will be created in your "~/Library/Preferences" folder. It will contain a preferences file for MacYTDL, a "simulate" file used to check URLs and get file names, log files produced by *youtube-dl/YT-DLP* and any batch file.

- If macOS 10.15 or higher is being used, *YT-DLP* will be downloaded and saved in the folder "/usr/local/bin/". For earlier macOS versions, *youtube-dl* will be downloaded. The folder "/usr/local/bin/" will be created if it doesn't exist.

- Two utilities, *FFmpeg* and *FFprobe*, will be downloaded and saved also in "/usr/local/bin/".

- A script library which provides code for MacYTDL's dialogs will be copied to "~/Library/Script Libraries". That folder will be created if it doesn't exist. The script library will be automatically updated as new versions are released.

- MacYTDL will offer to install a Service in your "~/Library/Services" folder. The Service called "Send-URL-To-MacYTDL.workflow" sends URLs to MacYTDL from other applications. It is not required for MacYTDL to function and can be installed later if preferred. If installed, the Service will be automatically updated as new versions are released.

  - [More detail is provided below](#).

- In Macs running macOS earlier than 10.15, MacYTDL will offer to install a small executable, AtomicParsley, in your "/usr/local/bin/" folder. Atomic Parsley is used by *youtube-dl* to embed thumbnail images in downloaded videos. It is not required by *YT-DLP* nor for MacYTDL to function and can be installed later if preferred.

  - [More detail is provided below](#).

Please note: downloading the various components takes time. It can look as if MacYTDL is not doing anything. Also, the component downloads cannot be cancelled. Please just wait.

Zip files will be saved into "/usr/local/bin/" during the process to install *FFmpeg* and *FFprobe*. Do not delete them. That will cause an error. The two Zip files will be automatically deleted when the install is finished.

On startup, MacYTDL will automatically look for and re-install any missing components.

When installation is finished, the MacYTDL Main dialog will be displayed.

Using MacYTDL

MacYTDL will try to download a video from any valid URL that is entered into the "Paste URL here" text box. URLs can be:

- copied from any app and pasted into the text box;

- clicked-and-dragged from any app into the text box;

- typed manually into the text box.

When a URL has been entered into the Main dialog, click on "Continue" or press the [Return] key to start the download.

This is the Main dialog:
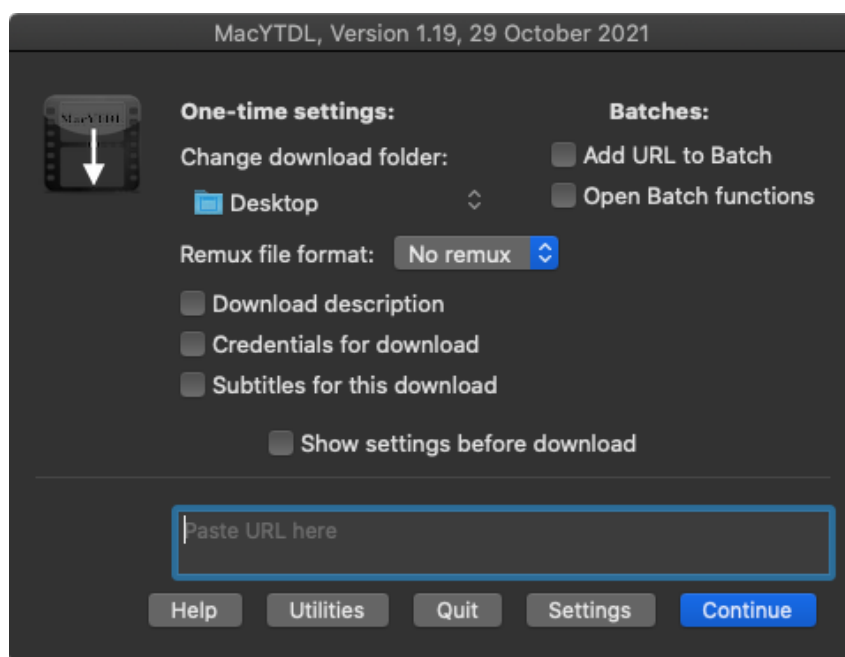
If you have installed the MacYTDL Service, instead of copying the URL you can choose:

[application menu eg. "Safari"] => Services => Send-URL-To-MacYTDL

The MacYTDL Service will check whether you have selected a URL in the current document, have copied a URL to the Clipboard or have opened a web page in a web browser, switch to MacYTDL and paste the URL into the text box.

Multiple URLs can be pasted or entered into the text box before downloading. They must be separated by a space character. MacYTDL will download each video in turn but in one process (and saving just one *youtube-dl/YT-DLP* response log file). Alternatively, URLs can be pasted and videos downloaded in separate processes. Current settings can be displayed before starting download.

After initiating the download, the Main dialog will disappear for a short while. MacYTDL will check the URL and if it's OK, will redisplay the Main dialog, start the download and display a "Monitor" dialog. Sometimes, the Monitor dialog is displayed before any downloaded data has been saved. In that case, the Monitor dialog reports "Progressing" but as the download continues, it will report on progress:
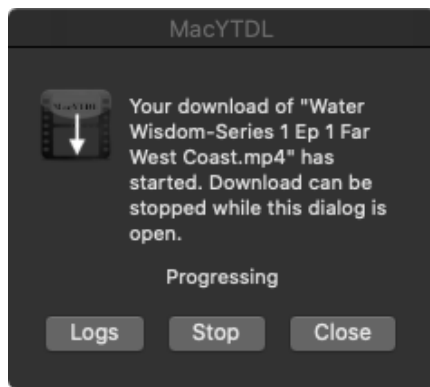
From the Monitor dialog, you can open the log folder, stop the download or close the dialog.

To cancel a download, click on the "Stop" button. The download will cease and any partially downloaded files moved to the Trash. This will also cancel downloading multiple videos that are in one progress (e.g. YouTube playlists). Completed downloads, subtitles files, description files and partially completed batch downloads are not deleted.

Note: MacYTDL cannot cancel a download after the Monitor dialog has been closed.

You can leave the Monitor dialog open and switch back to the Main dialog to download another video. A separate Monitor dialog will be shown for each download process. Each can be closed and the related downloads stopped separately in any order.

Monitor dialogs provide some detail on progress with the download. Depending on the site from which a video is being downloaded, progress is shown as a percentage of the total expected file size or the total downloaded up to that point. The time elapsed since download started is also shown. Some downloads involve a number of separate files e.g. playlists and separate video-audio merges. In those cases, the progress reported will reset when the next file download commences. Monitor dialogs are updated every 5 seconds.

Downloaded files are named as usefully as possible with series name and episode details for the Australian broadcasters ABC (iView), TEN (10Play), SBS (On Demand), Nine (9Now), Seven (7Plus) and for ITV.

MacYTDL issues a notification when each download process is finished:
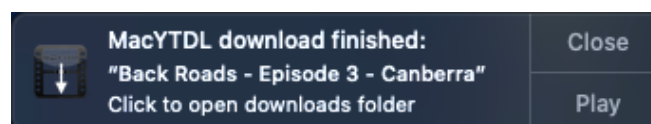
Clicking on the body of the notice will open the downloads folder. The "Play" button will open the downloaded video in the default video player. If the notification is for a playlist or other multiple download, the first video in the list will be played. The notification will automatically quit after 10 seconds.

The notification also advises if any errors were reported during the download whether or not the was download completed. There are two options: to open the log file of the download concerned; and to open the MacYTDL log folder.
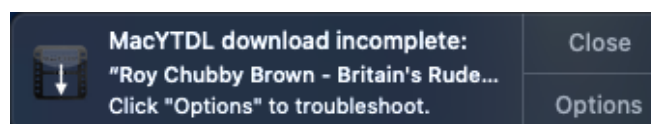
The "download incomplete" notification will automatically quit after 20 seconds.

In macOS 11 Big Sur and higher, you will need to hover your mouse over the right hand end of the notification to see the available options.

The form of notifications can be controlled in the "System Preferences". As with all macOS notifications, there is a choice of alerts, banners or no alert and MacYTDL notifications can be disallowed. MacYTDL notifications, however, do not play a sound.
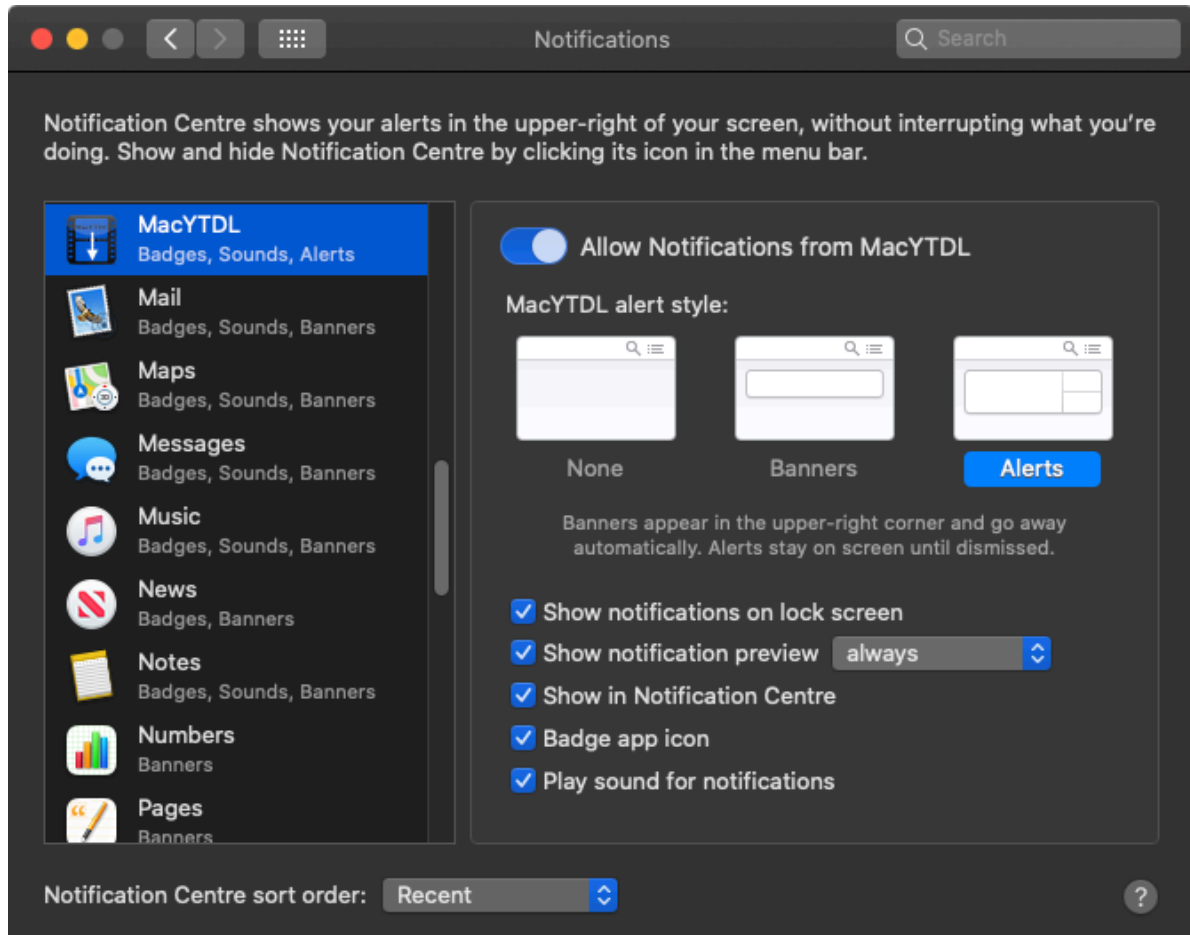
More on the Main dialog

The Main dialog should be reasonably self-explanatory. Paste or enter a URL into the text box and click on "Continue" or press the Return key.

The position of the Main dialog will be remembered if it is moved to another place on the Desktop.

Multiple URLs can be pasted into the box, separated by a space character. When "Continue" is selected, those videos will be downloaded sequentially in one process.

Click on "Settings", or press command-s, to change the MacYTDL settings. See below for details.

Click on "Utilities", or press command-u, to access a variety of functions. See below for details.

To show this help file click on "Help", or press command-shift-?

Click on "Quit" or press command-q to quit MacYTDL.

One-time settings

Five options can be set for the next download. These settings are not saved.

- The download folder can be set.

- The download container can be remuxed, that is, converted into another format. The choice of format is: mp4, mkv, webm, ogg, avi and flv. Remuxing is always done by *FFmpeg*. This can be handy when the desired container format is not available from the web site. The original file can be kept or automatically deleted according the the choice in Settings.

- A file containing a description of the video (if available) can be downloaded and saved with the video file. The file will be named "[the video name].txt".

- Username and password credentials can be added for the next download:
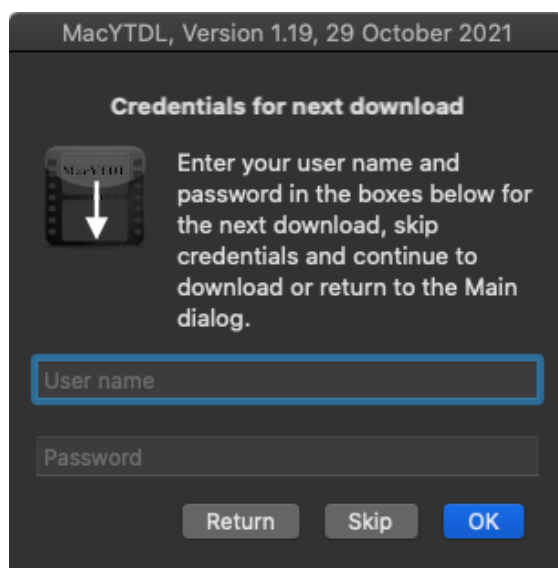
- A subtitles file can be downloaded (the subtitles format and language are taken from "Settings" or set to the default, "best").

The One-time settings return to the saved settings after the "Continue" button is clicked. Thus, the user may have set the usual download folder to "~/Movies/" but can use the runtime setting to

download the next video to somewhere else eg. "~/Desktop/". After that videos will be downloaded to "~/Movies/".

Batches

Instead of downloading immediately, URLs can be saved for downloading later. When the "Add URL to Batch file" checkbox is ticked and the "Continue" button is clicked, the URL is saved in a "batch" file. The batch file is stored in the MacYTDL preferences folder.

To access all batch functions, click on the "Open Batch functions" checkbox and click on "Continue". See below for more detail.

"Show settings before download"

If ticked, current download settings will be displayed with the options to quit, edit settings, cancel the download or download. The dialog also provides the option to change the saved "Show settings before download" setting.
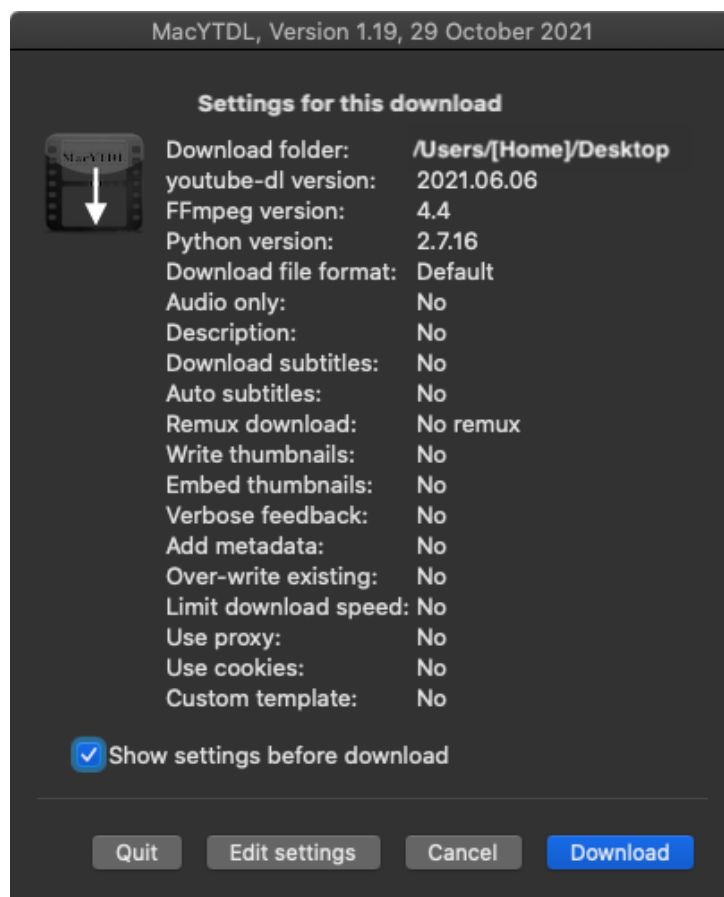
Settings

MacYTDL uses a number of options that are provided by *youtube-dl/YT-DLP* to control downloads. To access settings, click on the "Settings" button in the Main dialog or press command-s. The Settings dialog will be displayed:
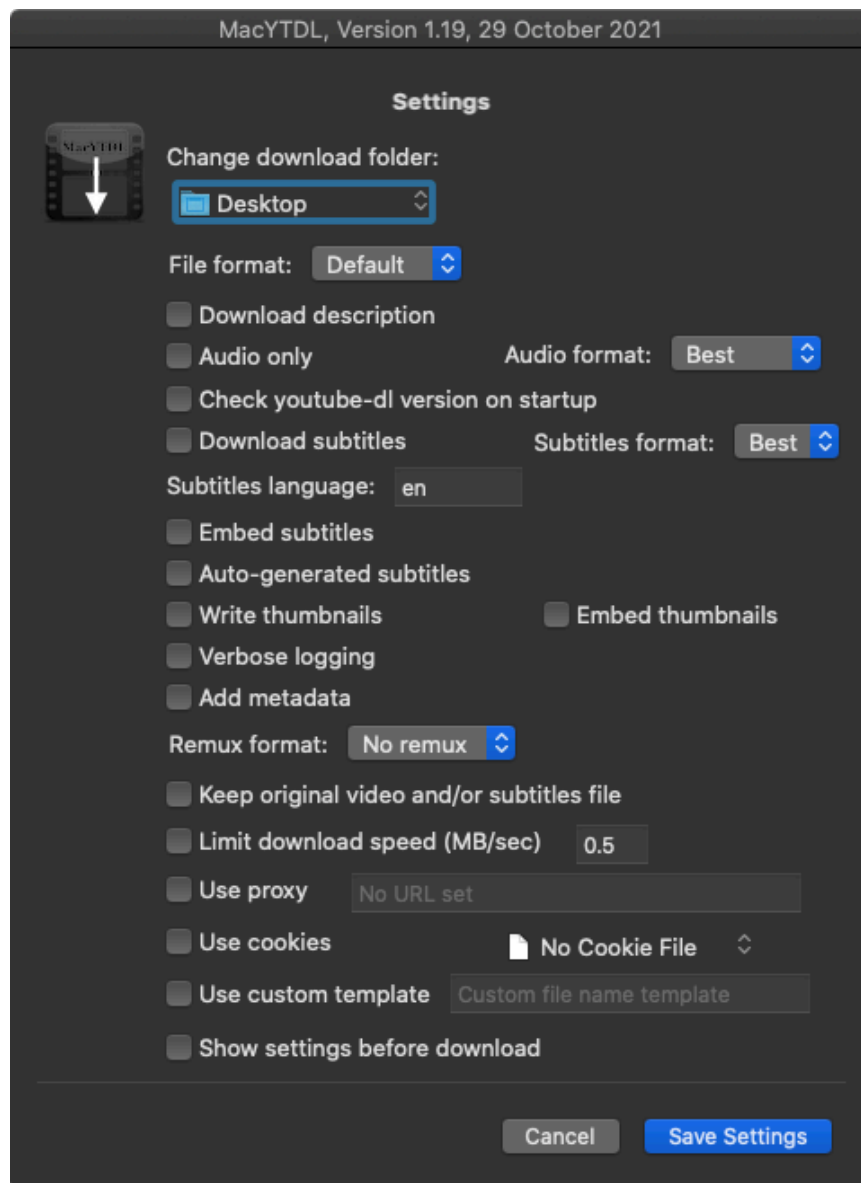
To save changes, click the "Save Settings" button or press the Return key. That will save your settings and take you back to the Main dialog.

"Change download folder:"

To change the folder in which videos are saved, click on the drop-down box and click on "Choose…". Choosing any of the other locations listed will not change the download folder.
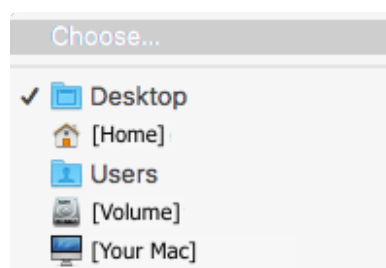
A choose folder dialog is shown enabling you to select a folder in which downloaded videos will be saved. To create a new folder, press command-shift-n or (macOS Catalina and later) click on the new folder icon in the dialog's toolbar.

Note: The download folder must be available for MacYTDL to be able to save videos into that folder. If the download folder is on an external volume, make sure that volume is mounted on your Desktop. Also, MacYTDL might not be able to save videos on an external volume which is password protected.

"File format"

By default, MacYTDL will try to source files from the given web site which can be merged to give the highest resolution video and best quality sound together. If the best quality available is a single file, that will be downloaded. This setting forces MacYTDL to download the best single file available in the chosen format. It can result in a lower quality, often smaller, file which often is not desirable. Thus, this setting can usually be left at "Default".

File format specifies a choice of container format. The choices are: Default, mp4, webm, ogg, 3gp and flv. MacYTDL checks whether the format chosen is available from the web site. If not, it asks whether to:

- ignore the preference and download the best format available;

- download the best format and remux to the preferred format;
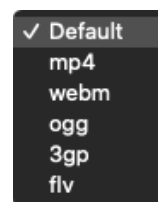
    Note: Remuxing to some formats will fail if the downloaded video does not contain the required codecs. Remuxing to "3gp" is not available and will revert to "mp4" instead;

- cancel the download.

"Download description"

Enables descriptive text provided by the video web site to be downloaded. The description is saved in a text file called "[the video name].txt". It is saved in the downloads folder.

"Audio only"

Requests a download of just the audio part of the video. Some web sites offer separate audio files for download. If a separate audio file is available, it will be downloaded in the best quality available. If a separate file is not available, the full video will be downloaded and the audio component extracted to a separate file with the best quality and the original file will then be deleted. If a remux format is set, the downloaded container will be remuxed to the desired format if it is compatible.

Note: Remuxing an audio-only download can be tricky and any incompatible combination will cause an error. Usually, there is no need to remux audio files as they are already in a compatible container.

"Audio format"

If "Audio only" is chosen, the resulting download will be in this format. Choices are best, aac, FLAC, mp3, m4a, Opus, Vorbis, wav.

Note: If a remux setting is also chosen, the resulting audio codec will be the most compatible with the selected container format. Thus, for example, if the container is
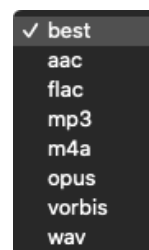
remuxed to "mp4", the audio will be stored in "aac" format even though mp3 was selected.

"Check youtube-dl/YT-DLP version on startup"

When this setting is on, MacYTDL will automatically check whether a new version of *youtube-dl/YT-DLP* is available every time MacYTDL is started. If so, the user is invited to update. This is useful as *YT-DLP* is updated regularly, sometimes every few days. Updates can improve downloading from some web sites, remove bugs from the *YT-DLP* core and add support for new sites. *youtube-dl* is no longer in active development and so this function will not find a more recent version.

"Download subtitles"

Enables subtitles file to be downloaded if they exist. Subtitles are saved in the chosen format and converted if the web site does not provide that format. If subtitles are not available, MacYTDL provides choice to cancel or continue the download without subtitles. If the video is being downloaded from YouTube, this setting will provide author-generated subtitles if they are available. MacYTDL will warn if author-generated subtitles are requested but only auto-generated are available and vice versa.
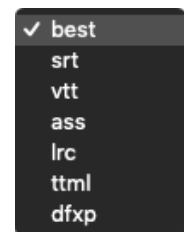
"Subtitles format:"

Enables a choice of subtitles format. Choices are: best, srt, vtt, ass, lrc, ttml and dfxp. If the chosen format is srt, ass, vtt or lrc but is not available, the best available will be automatically downloaded and converted to the desired format. If the chosen format is dfxp or ttml and that format is not available, MacYTDL will download the best available.

"Subtitles language"

Enables choice of subtitles language. To request a language, enter the relevant language code. A list of language codes is available on Wikipedia. Use the two-letter codes in the "639-1" column. Note, however, that some web sites adhere to the IETF standards which can include country or region codes e.g. "en-US" for English US. If a web site does not offer the language you request, MacYTDL presents a list of available languages.

"Embed subtitles"

Subtitles will be embedded in the video file instead of being saved in a separate file.

Note: The subtitles file will be downloaded, then deleted or retained after embedding is finished according to the "Keep original…." setting (see below). Embedding may remove custom formatting eg. text colour.

"Auto-generated subtitles"

The YouTube web site offers automatically generated subtitles (called "captions") for some videos. This setting will download those subtitles if they exist.

If both author and auto-generated subtitles are requested and both exist, author-generated subtitles will be downloaded (as they usually have a better translation into text). MacYTDL will warn if auto-generated subtitles are requested but only author-generated are available and vice versa.

"Write thumbnails"

If it exists, a separate thumbnail image file will be saved in the downloads folder with the video file.

"Embed thumbnails"

If it exists, a thumbnail image will be embedded in downloaded files. It's useful especially for music tracks (e.g. album cover art). For users on macOS 10.14.6 and earlier (who are using *youtube-dl*), the option is only available if the *AtomicParsley* tool has been installed.

"Verbose logging"

youtube-dl/YT-DLP provide feedback on progress and what happened during the download. The feedback is stored in a file called "youtube-dl_response-[the video name]-[date-time].txt" that is saved in this location: "~/Library/Preferences/MacYTDL".

If there is a problem with the download, that file might provide clues on the cause. With the verbose option, *youtube-dl/YT-DLP* provide more detail in the feedback. To open the folder containing the response files, use the "Open log folder" function in "Utilities".

"Add Metadata"

Any available metadata will be embedded in the downloaded file(s) in the correct variables.

"Remux format"

Video files are usually a combination of a video component and an audio component (i.e. codecs) within a container. Common video formats include: H.264, vp09, H.265 and DivX. Common audio formats include: aac, mp3, m4a, FLAC, Opus and WAV. Common container formats include: mp4, mkv, webm, avi and ogg. Some combinations of codecs and containers are more common than others e.g. audio within mp4 containers is usually in aac format. Also, some audio formats can exist without a container e.g. mp3. Other audio formats are stored within a simple container with the same name e.g. FLAC, m4a. Further, the file extension can vary with the kind of device on which the file is played e.g. mp4 audio with Apple's DRM protection are named ".m4p".

In MacYTDL, remuxing is the process of converting one video container format to another. An example would be to convert a webm file to mp4. MacYTDL implements remuxes with no change to the audio and video codecs, which are copied to the new container format. Thus, it is possible that the audio and video components might not be compatible with the container format. Detail on containers and compatible codecs are available on Wikipedia.

On occasion, the desired container format is not available from a web site. For example, many YouTube videos are in "webm" format which might not suit all situations. With the Remux option, MacYTDL will download the best quality available then convert the download into the chosen container format. The choice of container format is: mp4, mkv, webm, ogg, avi and flv. Remuxing is always done by *FFmpeg*. Remuxing is skipped if the downloaded file is already in the chosen format.
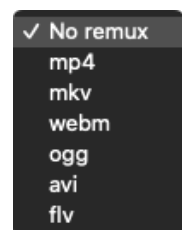
Note: It is often best to leave the file format and remuxing settings to "Default". That will ensure *youtube-dl/YT-DLP* downloads the best quality video and audio in the most compatible container.

Note: Remuxing to ogg or webm formats will often fail. Webm is a container for VP8/VP9/AV1 (video) and Vorbis/Opus (audio) formats and will fail in MacYTDL if one of those codecs is not in the dowloaded file. Similarly, ogg is a container for Speex, Vorbis, Opus, FLAC, Theora, Daala, Tarkin and Dirac formats and will fail in MacYTDL if one those codecs is not available. If there is an error, the MacYTDL response file should have information from *FFmpeg* on the reason.

"Keep original video and/or subtitles file"

If a remux format is set, this option enables the original file(s) to be retained instead of automatically deleted. Similarly, it will retain the original subtitles file if they have been embedded or converted.

"Limit download speed"

This sets the maximum download speed for the next download. It is useful if download bandwidth is limited and other apps have a greater need. The limit rate is in megabytes/second and must be a real positive number e.g. "0.5" will limit download to 0.5 megabytes/sec.

Note: If it detects features it does not support, *youtube-dl/YT-DLP* delegates downloading to FFmpeg (e.g. video delivered by HLS). In that case this setting has no effect.

Note: A different speed can be set for each download. This setting does not control downloads which are underway.

"Use proxy"

Some users need to supply a proxy URL in order to download a video. This setting passes the URL saved to *youtube-dl/YT-DLP* which uses that URL to request the video for download. The URL must start with "http:", "https" or "socks". A URL must be entered to use the proxy feature.

"Use cookies"

Features on some web sites are only available if user credentials are other details are supplied. Sometimes, the only way to provide those details is via a "cookies" file. This setting passes the location of the user's cookies file to *youtube-dl/YT-DLP* which uses the cookies to log into and/or provide any required data to the web sites concerned. A valid cookies file must be provided using the drop down path control and must contain the required data.

Note: A cookies file can be created using extensions in the Firefox and Chrome web browsers. The file must be formatted as a "Netscape HTTP Cookie File". It must be a text file with UTF-8 encoding.

"Use custom file name template"

Normally, MacYTDL gets the name of the video to be downloaded from the web site which serves the file. The custom file name template setting uses a name supplied by the user.

Beware: There is little error checking on the custom template. MacYTDL always checks that the last keyword in the template is the file extension. Otherwise, it passes the custom template setting on to *youtube-dl/YT-DLP*. It is thus quite easy to cause errors. Also, to avoid overwriting files with the same name, using the same name for successive files will probably result in a failed download.

A custom template can contain virtually anything. There are two main types of template – which can be combined.

The first kind of template uses keywords recognised by *youtube-dl/YT-DLP*. A list of keywords is available in the YT-DLP readme. There are many keywords. Not all keywords apply to all downloads. Some, for example, only apply to music tracks.

There is a strict format which must be used for keywords to be recognised. They must be provided according to the python string formatting operations. For example, all keyword entries must start with the '%' symbol, be enclosed by brackets and followed by formatting operations. The most

common operation flag is 's'. Thus the default template used by MacYTDL is "%(title)s.%(ext)s" – the video title followed by the file extension. With a few exceptions, any literal can be used between keywords and will be used in the file name. Note that some special characters cause errors in Windows and other systems.

A template with errors in formatting (e.g. a missing '%') will usually be treated as a literal which will be used for the file name. A template with errors in content (e.g. "%(None)s" a non-existent keyword), will cause the default placeholder to be added to the file name. The default placeholder is 'NA'.

The second type of template is a variation on the first which can control where files are saved. The '/' symbol in a template is used to indicate the path of the saved download. The '~' symbol indicates your "home" folder. Any missing folders will be automatically created. The standard path format on Windows systems can also be used. Some examples:

MyVideos/%(title)s.%(ext)s

 – will save downloads in the "MyVideos" folder inside your download folder with the file name and extension used by the serving web site – it will create the "MyVideos" folder if it doesn't exist.

~/Downloads/%(playlist)s/%(title)s.%(ext)s

 – will save downloads with the file name and extension used by the serving web site in a folder named with the playlist name inside the "Downloads" folder inside the home folder. It will create a folder named with the playlist name if required.

~/Videos/%(series)s - my favourite/%(season)s/%(episode)s.%(ext)s

– will save downloaded videos with the file name taken from the episode title and extension used by the serving web site inside a folder named with the title of the season inside a folder named with the title of the series followed by the text " - my favourite", inside the Videos folder in the user's home folder. Any missing folders will be automatically created.

/Music/%(title)s.%(ext)s

– The download will <u>fail with the error</u>: "unable to create directory [Errno 30] Read-only file system". That is because of the '/' character at the beginning of the template.

"Show download settings before starting download"

If ticked, current download settings will be displayed with the option to download, change settings, cancel the download or quit MacYTDL:
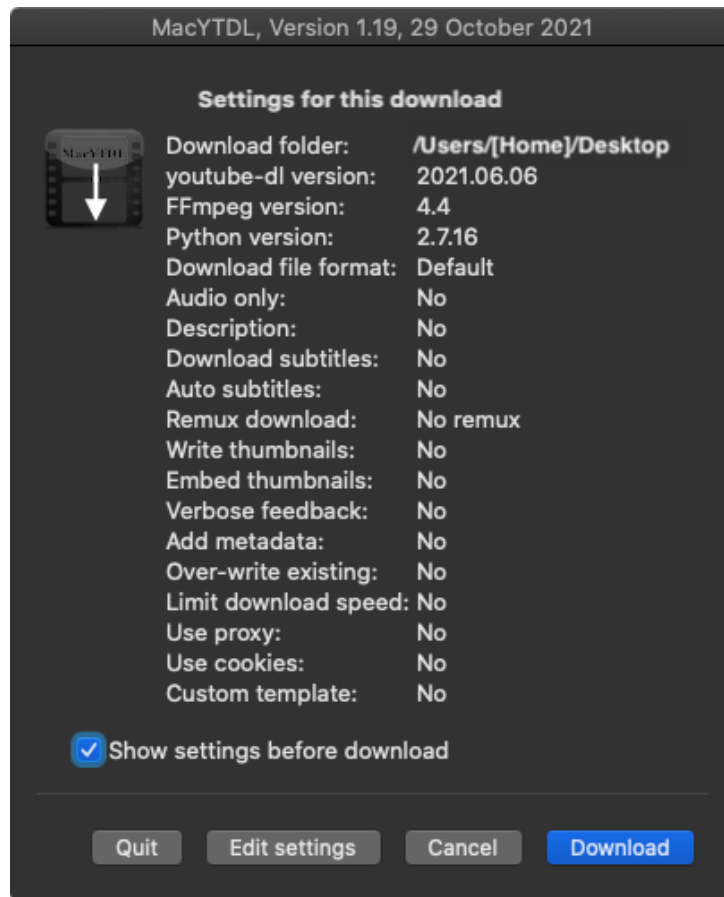
"Cancel" – command-c

Exit and return to the Main dialog without saving any changes. Keyboard shortcut is command-d.

"Save Settings" – Return

Save all changes to options and return to the Main dialog. Keyboard shortcut is Return.

Utilities

A number of additional utilities are available to help with using MacYTDL. Click on the "Utilities" button or press command-u in the Main dialog:
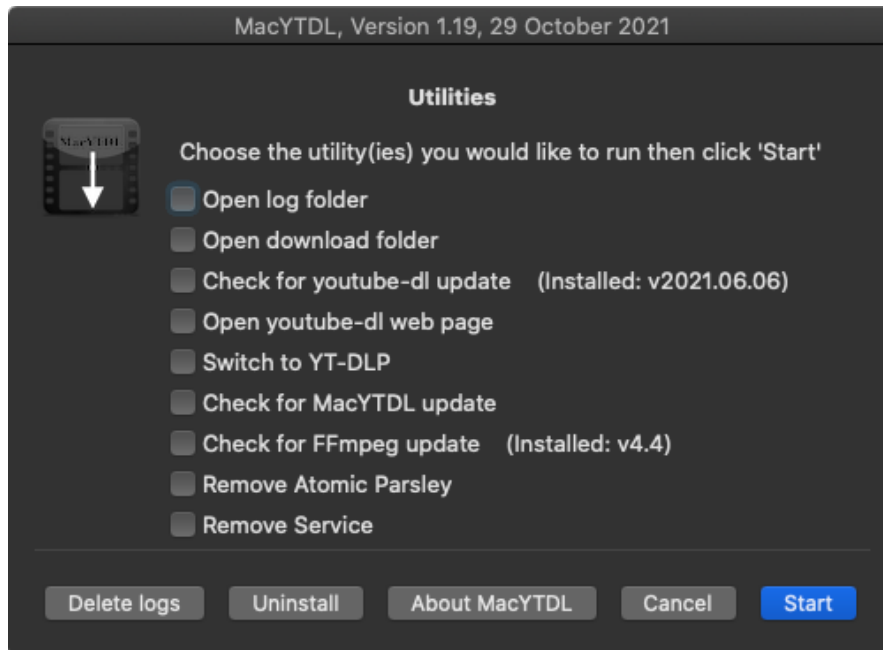
Any combination of the nine checkbox options can be selected at the same time. They will be performed in order when the "Start" button is clicked.

"Open log folder"

This will open the folder where all the *youtube-dl/YT-DLP* "response" or, log, files are stored (in "/Users/[your home]/Preferences/MacYTDL/"). A separate log file is saved for each download.

"Open download folder"

This will open the folder where downloaded videos are saved.

"Check for youtube-dl/YT-DLP update"

Checks whether a new version of *youtube-dl/YT-DLP* has been released. If so, offers to download and install.

Note: The *YT-DLP* developers issue updates regularly. So, it is worthwhile checking for updates regularly or setting the option to automatically check on start-up.

Note: *youtube-dl* is no longer in active development and so no updates will be found after version 2021.06.06.

"Open youtube-dl/YT-DLP web page"

Opens the *youtube-dl/YT-DLP* "Releases" web page in user's default web browser. The Releases page provides detail on the changes made to *youtube-dl/YT-DLP* in each release and the changes being prepared for the next release. The *YT-DLP* URL is: https://github.com/yt-dlp/yt-dlp/releases.

"Switch to youtube-dl/YT-DLP"

Switches MacYTDL between the *youtube-dl* and *YT-DLP* tools.

Note: *YT-DLP* requires macOS 10.15 or higher. *youtube-dl* requires macOS 10.10 and higher. It is possible to use *YT-DLP* on macOS 10.13 and 10.14. More detail is provided *below*.

"Check for MacYTDL update"

Checks whether a new version of MacYTDL has been released. If so, offers to download the new version. If downloaded, it will be saved into the MacYTDL downloads folder. Users can then install the new version where they prefer.

"Check for FFmpeg update"

Checks whether a new version of *FFmpeg* and *FFprobe* has been released. If so, offers to download and install the new version.

"[Install/Remove] Atomic Parsley"

AtomicParsley enables thumbnail images to be embedded in dowloaded files (e.g. album cover art). This function enables *AtomicParsley* to be installed if that was not done on the first startup. It enables *AtomicParsley* to be removed if it has been installed. It is used automatically by *youtube−dl*. *AtomicParsley* is not required by *YT-DLP*.

"[Install/Remove] Service"

This enables the MacYTDL Service to be installed if that was not done on the first startup. It enables the Service to be removed if it is installed. The MacYTDL Service takes a URL from the current text selection in any document, the Clipboard or the current web browser tab, switches to MacYTDL and pastes the URL in the MacYTDL URL text box.

"Delete logs" – command-d

Moves to Trash all the "response" log files.

"Uninstall" – command-shift-U

Moves <u>all</u> files related to MacYTDL to the Trash. Asks for confirmation before moving files. If you change your mind you can put back all those files individually or put back just MacYTDL and run it again. Fresh copies of all components will then be downloaded and installed.

"About MacYTDL" – command-a

Displays a dialog with information about MacYTDL.
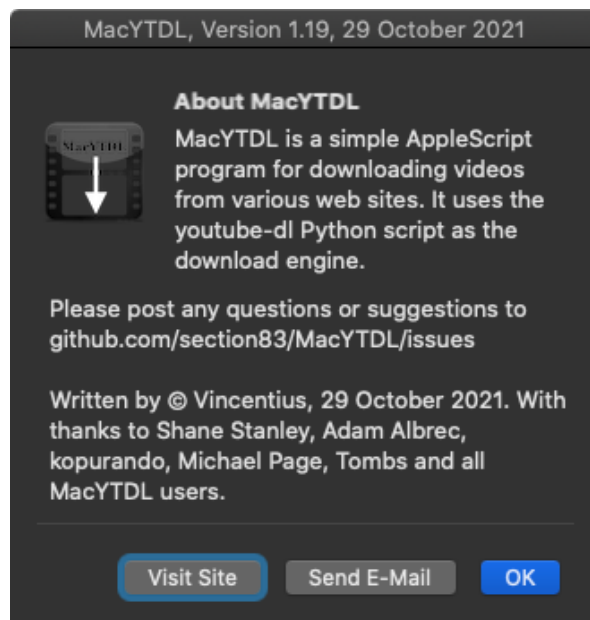
The "Visit Site" button opens the MacYTDL GitHub page in the user's default web browser. The GitHub page provides some detail on MacYTDL, a facility for logging issues or suggestions and links to downloads of the current and old versions. The URL is: https://github.com/section83/MacYTDL/. The second button opens a blank message in the user's default e-mail programme addressed to MacYTDL developer.

Keystroke shortcuts

A number of keystroke shorts make using MacYTDL a little quicker:

Main dialog

Return	Start download, Add URL to batch or Open batch functions as relevant
Command-u	Open Utilities dialog
Command-s	Open Settings dialog
Command-shift-/	Show this Help file
Command-q	Quit MacYTDL

Settings dialog

Command-c	Close and return to Main dialog without changing settings

Utilities dialog

Space bar	Place a tick in the selected check box.
Tab	Moves the selection between check boxes and function buttons.
Return	Start running the utilities checked
Command-d	Delete all log files
Command-shift-u	Uninstall MacYTDL
Command-a	Show the "About MacYTDL" dialog
Command-c	Close and return to Main dialog

About MacYTDL dialog

Return	Close and return to Main dialog
Command-v	Open the MacYTDL GitHub page and return to Main dialog
Command-e	Open a new message in user's default email app addressed to macytdl@gmail.com

Settings shown before download

Command-q	Quit MacYTDL
Command-e	Close and Open Settings dialog
Command-c	Cancel download and return to Main dialog
Command-d	Commence download

Credentials dialog

Return	Commence download (after entering credentials)
Command-r	Cancel download and return to Main dialog

Command-s   Skip credentials and commence download

Batch functions dialog

Command-r   Return to Main dialog

Command-e   Open the batch file for editing

Command-c   Clear the batch file i.e. remove all entries

Command-shift-u   Remove the last item in the batch file

Command-d   Command downloading all URLs in batch file

Monitor dialog

Command-l   Open logs folder

Command-s   Stop download being monitored

The MacYTDL Service – "Send-URL-To-MacYTDL"

MacYTDL includes an optional utility, the Send-URL-To-MacYTDL service. Services are small programs which provide extra functions in other applications. They are accessed using the "Services" item within the application and context menus.

Send-URL-To-MacYTDL works with any application that handles text. It makes downloading a video with MacYTDL a little quicker and easier. It takes any URL that has been copied or selected in any application and sends the URL to MacYTDL. If no URL has been copied or selected and a web browser is active, the Service will copy the URL of the current active tab and send that URL to MacYTDL. The web browsers in which it works are Safari, Opera, Firefox, Microsoft Edge, Chrome and Chromium.

The procedure to start with depends on where you find the video URL that you wish to download:

- If the URL is visible in text of any document in any application, select the text.

- If the URL is in a hyperlink in any document in any application, use the provided function to copy the link (e.g. in Safari, right-click the link text and choose "Copy Link").

- If the URL is in the address bar of a currently active web browser, just go to the next step.

Next, when you have copied or selected a URL or have that URL in your web browser:

- go to the application menu (eg. the menu with the heading "Safari");

- browse down to "Services";

- choose "Send-URL-To-MacYTDL".

Send-URL-To-MacYTDL will launch or switch to MacYTDL and set the URL of the video to be downloaded.

If you don't install Send-URL-To-MacYTDL when you first use MacYTDL, it can be installed later – see "Utilities" above.

Some people find using keystroke shortcuts to be quicker than using menus. To set a shortcut for the Send-URL-To-MacYTDL service:

- open System Preferences;

- click on "Keyboard";

- click on "Shortcuts";

- click on "Services" in the left hand pane;

- scroll down to "General";

- click in the tick box next to "Send-URL-To-MacYTDL"

- type the shortcut you wish to use.

On first using the Service, a message will probably be shown:
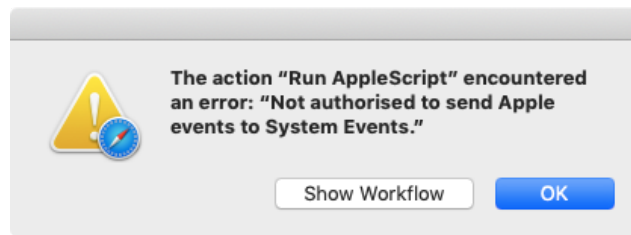
This means that the MacYTDL Service needs permission to use your web browser to send commands. In that case follow the procedure on above to give Safari permission to control "System Events".

Another message may be shown at the same time, if MacYTDL is running:
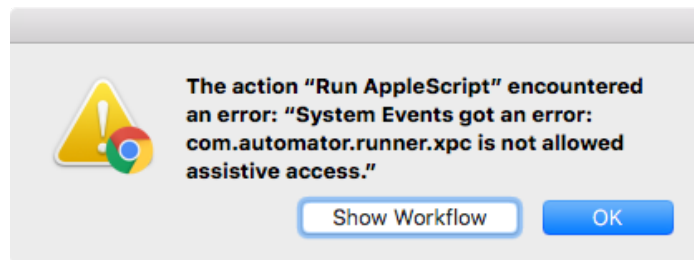
In that case, you will need to give permission for the MacYTDL Service to use your web browser to control your Mac:

- open "System Preferences";

- click on "Security & Privacy";

- click on "Accessibility";

- click on the padlock and enter an administrator account credentials when asked;

- if your web browser is not shown, click on the "+" symbol and navigate to your web browser application (usually in the "Applications" folder) and click on "Open";

- make sure there is a tick beside the web browser application;

- click on the padlock to prevent further changes.
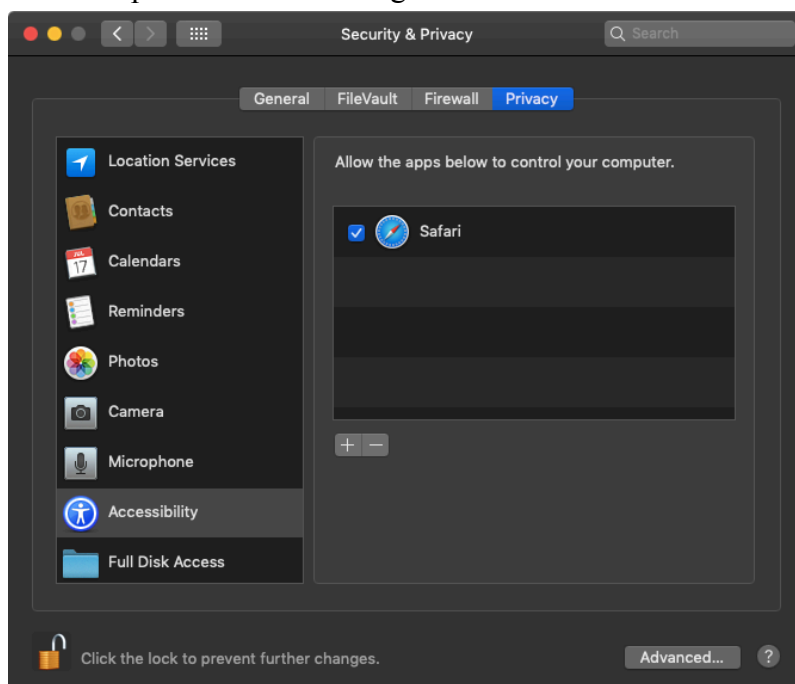
If MacYTDL is not running, you may also see this message:
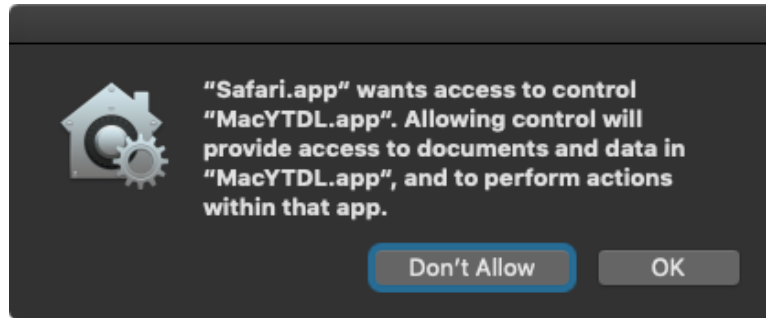
In that case, click on "OK" or follow the procedure on page 5 above to give Safari permission to control MacYTDL.

Using the default "config" file

youtube-dl and *YT-DLP* can be controlled using a "config" file. The config file can contain additional settings and options which are not provided by MacYTDL. MacYTDL automatically checks for the presence of a config file and passes its contents to *youtube*-dl/*YT-DLP*. It adds settings in the config file to those specified in the Settings dialog.

The config file must be a plain text file (e.g. created in TextEdit).

For *YT-DLP*, it may be stored in these locations:

- /etc/yt-dlp.conf [which will be used by all user accounts]

- Users/[Home]/.config/yt-dlp/config [which will be used by the "Home" user account]

- Users/[Home]/.config/yt-dlp.conf [which will be used by the "Home" user account]

- ~/yt-dlp.conf

- ~/yt-dlp.conf.txt

For *youtube-dl* it may be stored in these locations:

- /etc/youtube-dl.conf [which will be used by all user accounts]

- /Users/[Home]/.config/youtube-dl/config [which will be used by the "Home" user account]

The layout rules for the config file are very simple:

- Lines starting with a "#" are comments.

- Other lines are assumed to contain settings.

Some examples:

Mark videos watched (currently only supported for YouTube)
--mark-watched

Bypass geographic restriction via faking X-Forwarded-For HTTP header
--geo-bypass

Download playlist videos in reverse order
--playlist-reverse

A future version of MacYTDL will provide for:

- specifying different locations for the config file; and

- multiple config files.

More detail on the config file is available here:

https://github.com/yt-dlp/yt-dlp/blob/master/README.md#configuration.

https://github.com/ytdl-org/youtube-dl/blob/master/README.md#configuration

Using a ".netrc" file

MacYTDL respects the detail provided in a ".netrc" file if it is available and is activated. It enables separate credentials to be stored for individual web sites.

The file must be a UTF-8 plain text file and must be stored in the root of the user's home folder i.e.:

[Mac HD]/Users/[Home]/.netrc

Because it's name begins with a period, the netrc file is not normally visible in the *Finder*. To make it visible, press command-shift-.

The format of the credentials provided in the file is:

machine <extractor> login <login> password <password>

"<extractor>" is the name of the extractor used for the web site concerned. So, for example, credentials to log in to YouTube would be:

machine youtube login myaccount@gmail.com password my_youtube_password

A list of the extractors is available here:

https://github.com/yt-dlp/yt-dlp/tree/master/yt_dlp/extractor

To activate the netrc file, add "--netrc" to a default config file (see above).

How to uninstall MacYTDL

To uninstall MacYTDL, click on the "Utilities" button in the Main dialog and click on "Uninstall". There is an opportunity to reconsider uninstalling:
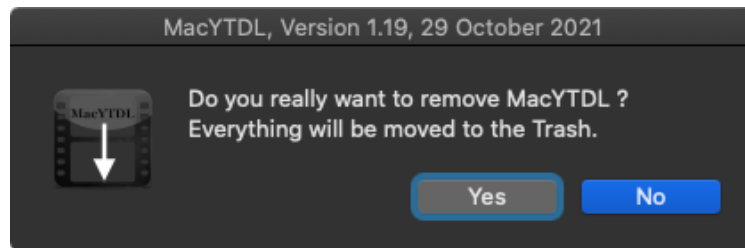
Confirm that you wish to remove MacYTDL and all MacYTDL related files will be moved to the Trash.

If you change your mind you can put back all those files individually or just put back MacYTDL and run it again. Fresh copies of all components will then be downloaded and installed.

YouTube URLs and playlists

There are a number of formats for YouTube URLs. The different formats lead to different results in MacYTDL. They can also result in the download of a very large number of files. MacYTDL pops a simple warning if a playlist contains more than 35 items.

This is a list of the common formats and what happens in MacYTDL.

Single video

> https://www.youtube.com/watch?v=zNpM6-5pFF4

> https://youtu.be/YlRY3KQpib4

A link to a single video or an abbreviated link to a single video. Individual videos usually have "watch?v=" in the URL or have no other identifying elements (as in the abbreviated form). MacYTDL will download the video.

A playlist

> https://www.youtube.com/playlist?list=PLA2-nXsHvm3LGeixJDJ29HT1saHhgLHHe

This is a YouTube playlist. Playlist URLs often contain "playlist?list=". The keyword "list" usually points to a particular playlist. Another indicator that this is a playlist URL is the "PL" at the beginning of the playlist ID.

MacYTDL will download all items in the list. It will warn if there are more than 35 items in the list and will advise if there were errors in the download.

Note: Playlist items are downloaded sequentially. It can take a long time to setup the download and then a long time to download the entire playlist.

Single playlist item

> https://www.youtube.com/watch?v=zNpM6-5pFF4&list=PLA2-nXsHvm3LGeixJDJ29HT1saHhgLHHe&index=1

This is a single playlist item. These URLs typically end with "&index=". MacYTDL will download just that item.

Note: To download the entire playlist, remove the "&index=[n]" part of the URL.

A playlist item

> https://www.youtube.com/watch?v=zNpM6-5pFF4&list=PLA2-nXsHvm3LGeixJDJ29HT1saHhgLHHe

> https://youtu.be/X0WOa_vDUFA?list=PLF76BA379BC905431

This is a playlist URL, or an abbreviated URL, that opens on a single item which usually starts playing automatically in a web browser. MacYTDL will download <u>all</u> items in the playlist. It will warn if there are more than 35 items in the list and will advise if there were errors in the download.

Live stream

> https://www.youtube.com/watch?v=W1ilCy6XrmI

Although it looks very much like a single video URL this is a live video stream URL. As yet, I have not found a reliable identifying feature in stream URLs. MacYTDL will download the stream until the "Stop" button is pressed on the Monitor.

The Monitor dialog can only be closed by stopping the stream download. This is to prevent download processes being left running with no facility to close them down.

Note: The recording will start from the moment MacYTDL starts to download the stream not from the beginning of the stream (which might have been some hours/days earlier).

Note: Some streams cannot be identified as such and so the Monitor dialog will show the usual "Logs" and "Close" buttons. To ensure such a stream download can be stopped avoid clicking on "Logs" or "Close".

An uploader playlist

> https://www.youtube.com/playlist?list=UUBABnxM4Ar9ten8Mdjj1j0Q

This is a playlist of videos from a particular uploader (i.e. a person or an organisation). Typically, the playlist ID starts with "UU". MacYTDL will download all items in the playlist. It will warn if there are more than 35 items in the list and will advise if there were errors in the download.

Beware: Some uploader playlists have a great many items. The example above has over 1,100 items. Such playlists take a <u>very long</u> time to prepare and then download.

A YouTube mix

> https://www.youtube.com/watch?v=NrLkTZrPZA4&list=RDEMtu8TSn01ATwUIqXnxfa6zQ&start_radio=1

> https://www.youtube.com/watch?v=8UVNT4wvIGY&list=RD8UVNT4wvIGY&start_radio=1

A YouTube mix is a playlist that has been curated by YouTube. Mixes are formed using the viewing history of each user (if known to YouTube). MacYTDL will download all items in the playlist. It will warn if there are more than 35 items in the list and will advise if there were errors in the download.

Beware: YouTube mixes can have a lot of items and so take a long time to prepare and then download.

Personal "Watch Later" playlist

> https://www.youtube.com/playlist?list=WL

A "Watch Later" playlist is very handy. Downloading, however, is not easy at present. I have not been able to get *youtube-dl* to download my "Watch Later" list by supplying my credentials directly. I have succeeded only by supplying my credentials in a "cookies" text file ([see above](#)). This will download all your YouTube "Watch Later" videos into your home folder. Remember to use ordinary double quotes, not the curly "smart" quotes you get in word processors. I have not been able to test the "Watch Later" downloads with *YT-DLP*. MacYTDL will warn if there are more than 35 items in the list and will advise if there were errors in the download.

Note: The cookies file must be encoded with UTF-8. Most cookies utilities will use that encoding.

Downloading videos from ABC iView (in Australia)

ABC iView in Australia serves most individual videos from what can be called "show" pages. Episodes are not available from their own web page. For example, all episodes for all seasons of the show *Back Roads* are available on the one "Back Roads" show page.

If the URL of a "show" page is pasted into MacYTDL and there is more than one episode, a separate dialog will be displayed listing all the episodes available. The desired videos can then be chosen for download. There is an "All episodes" checkbox which will download all episodes. This is an example iView download dialog:
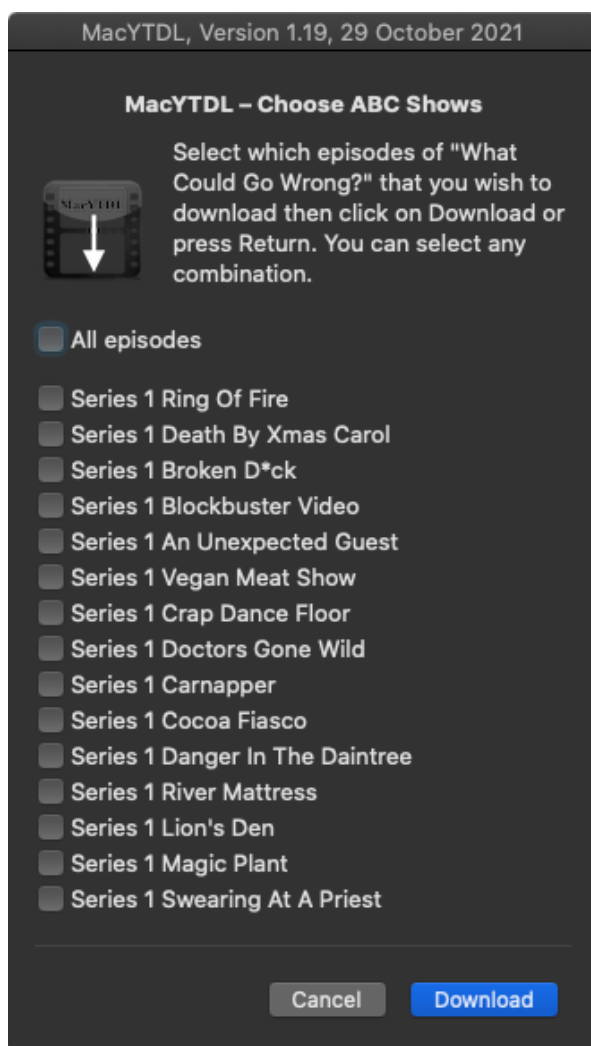
MacYTDL will also locate and list any "Extras" videos that are available for a show.

On iView the ABC has some shows for which there are a great many episodes ("Gardening Australia has 128). MacYTDL will try to show all the episodes in this dialog. On a Mac with low screen resolution (e.g. 1024x640), however, the dialog might be too large to show all episodes if there are more than 80. Hopefully a future version of MacYTDL will address that problem.

Notice that for some shows, episodes are shown in reverse order to match the order mostly used in iView.

At present MacYTDL can download from only one show page in each download process. However it can download:

- from any number of show pages if each download is requested separately;

- any number of episodes from any series of a single show; and

- any number of separate episodes from any show pages in one process if the URLs of each episode are supplied manually.

To download a number of individual episodes from more than one show page:

- navigate to the particular episode desired;

- click on the episode's thumbnail image to start the iView player;

- copy the URL from the address bar;

- switch to MacYTDL;

- paste the URL then add a space;

- return to iView, navigate to the new desired show episode and repeat the above steps.

If a show page has only one episode, MacYTDL will automatically commence to download that episode.

A separate destination folder can be specified in the "One-time settings" on the Main dialog. However, that must be done <u>before</u> selecting "Continue". A new download folder can also be requested by using a custom file name template ([see above](#)).

MacYTDL can also download and 'record' live streams from the ABC iView web site. It works in much the same way as for YouTube streams: MacYTDL identifies the download as an ABC live stream and shows the correct form of the Monitor. MacYTDL will download the iView stream until the "Stop" button is pressed on the Monitor.

Downloading videos from SBS On Demand (in Australia)

As with the iView web site, the SBS On Demand web site in Australia serves episodes of shows in a single "show page". Episodes usually are not available from their own web page. For example, all episodes of the show *Home Ground* are available on the one "Home Ground" page.

Note: MacYTDL can only show a list of episodes from the current season – which is the season that is first visible on a show page, usually but not always season 1.

If the URL of a "Show" page is pasted into MacYTDL and there is more than one episode, a separate dialog will be displayed listing all the episodes available. The desired videos can then be chosen for download. There is an "All episodes" checkbox which will download all episodes listed. This is an example On Demand download dialog:
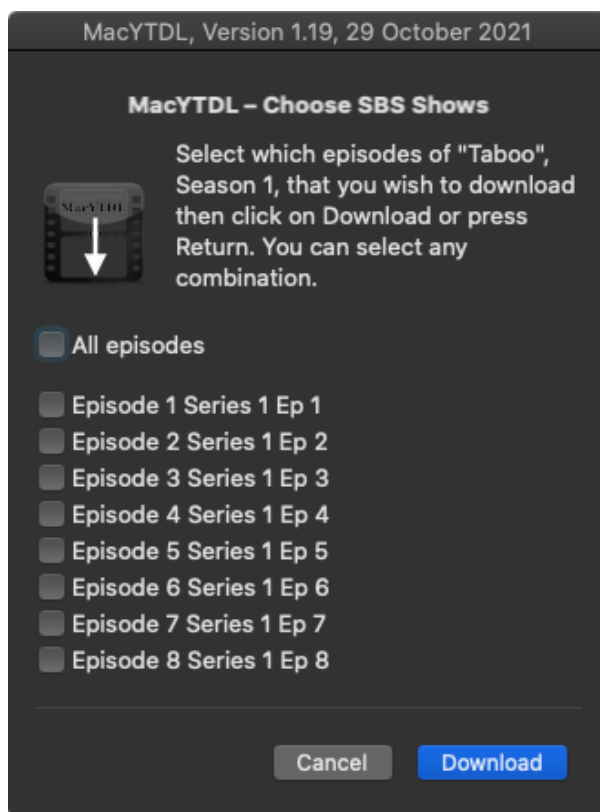
To download episodes in other seasons:

- click on the Season, to display a list of episodes for that season;

- right-click (or control-click) on the thumbnail picture of the desired episode;

- click on "Copy Link" – which will copy the episode's URL;

- switch to MacYTDL;

- paste the URL in the text box;

- click on Continue or press Enter/Return.

Note that for some shows, episodes are shown in reverse order to match the order mostly used in On Demand and there can be variation in the name of the show.

At present MacYTDL can download from only one show page in each download process. However it can download:

- from any number of show pages if each download is requested separately;

- any number of episodes from any series of a single show; and

- any number of separate episodes from any show pages in one process if the URLs of each episode are supplied manually.

If a show page has only one episode, MacYTDL will automatically commence to download that episode.

MacYTDL can download movies from SBS OnDemand "movie" pages. There is no need to manually copy the URL. For example: the movie "Coherence" can be downloaded from its web page:

https://www.sbs.com.au/ondemand/movie/coherence/1469404227931

A separate destination folder can be specified in the "One-time settings" on the Main dialog or in a custom file name template. However, that must be done before selecting "Continue". A new download folder can also be requested by using a custom file name template (see above).

MacYTDL can also download and 'record' live streams from the SBS On Demand web site. This feature is somewhat experimental as SBS do not indicate whether a URL is a live stream. Nonetheless, it works in much the same way as for YouTube streams: when possible, MacYTDL identifies the download as an SBS live stream and shows the correct form of the Monitor. MacYTDL will download the SBS stream until the "Stop" button is pressed on the Monitor.

Working with dual monitors

By default MacYTDL shows the Main dialog on the main screen of a dual monitor setup. If it is moved to the secondary monitor and closed, it should be displayed in that new location when reopened. However, results depend on the particular setup. The Monitor dialogs are <u>always</u> displayed in the top right hand corner of the main display.

Problems can arise if the screen layout is changed while MacYTDL is running. This is because the position of MacYTDL's dialogs is taken from the previously saved coordinates which become invalid if the layout is changed. It's best not to change the layout while MacYTDL is running. Making MacYTDL more friendly with dual monitor setups is still a work-in-progress.

Batch functions

There can be situations in which you want to download a number of videos at another time. For example, download speeds can be faster outside peak hour periods. The MacYTDL "Batch" functions enable URLs to be saved and downloaded at a later time. URLs are stored in a file called "BatchFile.txt" which is stored in:

/Users/[Home]/Library/Preferences/MacYTDL/

To add a URL to the batch file:

· in the Main dialog, click on the "Add URL to Batch" checkbox;

· paste the URL into the text box (if you did not use the MacYTDL Service); and

· click on the "Continue" button.

The URL will be checked to ensure it is valid then, added to the batch file and a message displayed accordingly. This involves checking with the video web site concerned and so can take time.

The "Add URL to Batch" checkbox will remain ticked until manually unticked or the user quits MacYTDL.

Any number of URLs can be added to the batch file in one step by pasting the URLs separated by spaces.

A separate dialog is provided to handle all other batch functions. To display the Batch functions dialog, in the Main dialog, click on the "Open Batch functions" checkbox then click on the "Continue" button. This dialog will be displayed:
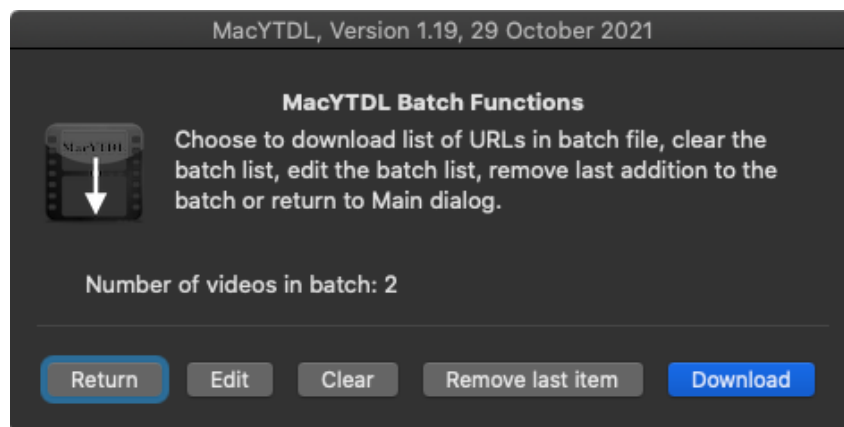
To download all the videos saved in the batch file, click on the "Download" button. The downloads will commence and continue, one at a time.

The Batch file is cleared automatically if the downloads are completed and no errors are reported.

Clicking on the "Edit" button will open the batch file in the user's default text editor. The batch file is a plain text file. There are restrictions on editing the file:

· the file must be saved as a plain text file – ie. not as rich text or word processing file;

· there can only be one URL per line – and no other text; and

· there must be a blank line at the end of the file – there can be no other blank lines.

The "Clear" button will remove all saved URLs from the batch file.

The "Remove last item" button will delete the most recently added URL from the batch. That can be useful if a URL was added by accident.

The "Return" button closes the Batch Functions dialog and returns to the Main dialog.

Note: Beware of including live stream URLs in a batch. That is because live streams continue recording until manually stopped with the Monitor dialog. No other downloads in the batch will proceed while a live stream is being downloaded.

There are a number of limitations with this batch function:

- Download settings apply to all videos in a batch – ie. settings are not saved with each URL. Thus, if a setting such as "Audio only" is set, then only the audio component of each video will be downloaded. It is usually safer to store URLs in the batch file that are all for video files or all audio files rather than mixing them up.

  − Consequently, also, all videos in a batch will be saved in the same downloads folder although this behaviour can be changed with the custom file name template setting (see above).

- Invalid text in the batch file may cause *youtube-dl/YT-DLP* to post an error. MacYTDL will ask *youtube-dl/YT-DLP* to ignore errors and continue downloading but might not advise of that error. In any event the error should be reported in the response file.

- Incomplete downloads are not automatically deleted by the Monitor dialog "Stop" function.

- MacYTDL does not check for duplicates in the batch file. *youtube-dl/YT-DLP* will download all files in the batch but will not save any duplicated files unless "%(autonumber)s" is added to the custom template setting.

- It is not possible to choose which videos in a batch to download. However, lists of URLs can be saved in separate files and copied and pasted into the batch file for downloading.

Overwriting existing videos ?

By default, MacYTDL will not overwrite an existing file. It always checks whether a video with the same name has been saved in the downloads folder. If so, a warning dialog is shown:
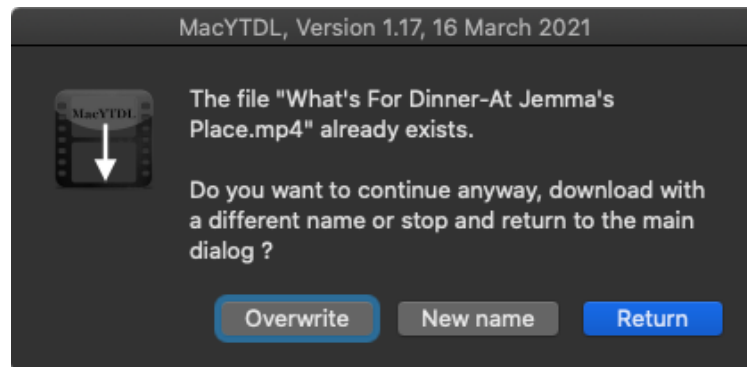
"Overwrite"

MacYTDL will move the existing file to Trash and commence downloading the new file.

"New name"

MacYTDL will commence downloading the new file with "-2" appended to the file name.

NB. At present, if there is more than one file to be downloaded, e.g. a playlist, ABC iView episodes, a batch or multiple video download, MacYTDL will add "-2" to the names of all files downloaded, not just those which already exist. That is because although *youtube-dl/YT-DLP* can download a number of files in one process, it apply only one output template for all the file names.

"Return"

Downloading is cancelled and the Main dialog is displayed.

Installing *YT-DLP* on macOS 10.13 and 10.14

YT-DLP has been built to run on macOS 10.15 and higher. It can, however, be used on macOS 10.13 and 10.14 if two macOS library files are installed. The two files are:

libcrypto.44.dylib

libssl.46.dylib

Those two library files are provided by macOS 10.15 and higher in the "/usr/lib/" folder. If they can be sourced, they should be copied to the "/usr/local/lib/" folder of a macOS 10.13 or 10.14 installation. *YT-DLP* will then function as it does on MacOS 10.15 and higher.

MacYTDL will check for the two library files if the user requests a switch from *youtube-dl* to *YT-DLP* on a Mac using macOS 10.14.6 or earlier.

With the two library files added, *YT-DLP* might also work on a version of macOS prior to 10.13. The MacYTDL developer has not tested that installation.

It is also possible to use the Unix *YT-DLP* executable by installing Python3.6 or higher and the required dependencies. More detail is provided here: https://github.com/yt-dlp/yt-dlp/blob/master/README.md#installation. The YT-DLP executable must be named "yt-dlp" to be used with MacYTDL.

What if the download doesn't work ? Troubleshooting

There are many reasons why a download might fail or not produce the desired result. MacYTDL does do a "simulated" test download and reports errors found (eg. because of invalid URLs or DRM limitations). But, that test cannot find all possible problems.

There is an option to attempt to download even when there is an error as well as some warnings. The download might well fail. If so, check the response file for possible reasons.

MacYTDL will show a warning that there was a problem if the download was not 100% complete or an error was reported by *youtube-dl/YT-DLP*. Also, if a download is interrupted, it is possible for *youtube-dl/YT-DLP* to continue the download from the point of interruption or to download a missing component. So, it is worthwhile trying the download again.

Note: An error with a video in a YouTube playlist will not prevent other videos in the list from downloading. If an error message is displayed, just click on "Download" to download the available videos.

A common problem is that a web site has changed and *youtube-dl/YT-DLP* is no longer able to find the correct file to download. The best solution is to check regularly for and install updates to *youtube−dl/YT-DLP*. Use the "Check for youtube-dl/YT-DLP update" function or turn on the "Check youtube-dl/YT-DLP version on startup" option.

Another common issue is AES-SAMPLE encryption or another DRM control. Many videos are encrypted with DRM protection and cannot be downloaded by *youtube-dl/YT-DLP*. That is a matter of principle. Sometimes, these issues are caught when MacYTDL checks the URL. In that case, what happened is noted in this file:

> /Users/[Home]/Preferences/MacYTDL/youtube-dl_simulate.txt

Note that the simulate file is overwritten by the next download.

If MacYTDL proceeds to download the video, *youtube-dl/YT-DLP* produces a detailed "response". If the download fails, that response might help to diagnose problems. MacYTDL saves the youtube−dl response in a separate file for each download. The file is saved here:

> /Users/[Home]/Preferences/MacYTDL/youtube-dl_response-[name of video].txt

The "verbose" option will provide more detail to help diagnose issues.

Information on how *youtube-dl/YT-DLP* functions is available at: https://github.com/ytdl-org/youtube-dl/blob/master/README.md and https://github.com/yt-dlp/yt-dlp/blob/master/README.md.

Limitations and known bugs

MacYTDL has some limitations and bugs:

- Can't download encrypted or DRM protected videos – this is a limitation of *youtube-dl/YT-DLP* imposed as a matter of principle.

- Does not use options to overcome geo-blocking.

- Does not provide access to some download *youtube-dl/YT-DLP* settings (eg. retries, dates, archives, filters). *YT-DLP* has many new features which MacYTDL cannot access at this time. The "config" file facility provides a way to access those features.

- Provides little control over the format of videos downloaded (eg. can't download and merge a video-only file with a video+audio file which enables higher quality audio with some downloads).

- The Send-URL-To-MacYTDL service cannot work when the web browser is in full screen mode. Therefore, it takes the browser out of full screen before switching to MacYTDL.

- It is not possible to download videos with the same name into the same folder unless the "% (autonumber)s" parameter is used in the custom file name template (see above).

- The second of duplicate downloads in a batch is not saved to disk (protection provided by *youtube-dl/YT-DLP*).

- Dialogs can appear in unpredictable locations when layout of dual screen setup is changed.

- The Monitor dialog and the download complete notification sometimes have the wrong extension on the file name shown (e.g. when a recode has been specified).

- The "Stop" function does not always delete incomplete downloads (including when using a custom file name template and batch downloads).

- A warning that the download already exists will be issued even though a recode to a different format has been specified.

- Using the Service causes MacYTDL to hang if the "Settings" or "Utilities" dialogs are active.

Future developments

MacYTDL is being actively developed. Functions which might be added in the future include:

- Option to choose items in playlists.

- Option to choose file format from a list of formats provided by the serving web site.

- Automatic downloads when Service is invoked.

- Facility to pause/continue downloads.

- More credentials options: video password, ".netrc" file and *Adobe Pass*.

- Option to add ad-hoc *youtube-dl/YT-DLP* command line parameters.

- Option to use *youtube-dl/YT-DLP*'s "archive" functions.

Why use a video downloader ?

There is some controversy around the practice of downloading videos especially from YouTube and TV catch-up sites. Copyright owners are concerned people will copy and distribute videos and so reduce the revenue from DVD and online sales. It is a real fear as downloaded videos can easily end up listed on Bit Torrent web sites such a Pirate Bay. That is why so many Bit Torrent sites have been blocked by court order in Australia and other countries. There is an active debate on the issue of such 'piracy' which would take too much space to cover here. But it's fair to conclude no one has won the debate as yet. Even reputable commentators provide advice on [how to bypass blocks](). Certainly, spreading copies of videos around the web via Bit Torrent is unethical. But, there are ethical reasons for downloading videos for personal viewing.

One reason for using a video downloader is particularly relevant to people in areas with poor internet access. When downloads are very slow or unreliable it is just not feasible to watch videos online using streaming services. With slow download speeds, it can take 2 hours or longer to watch a 1 hour video. If the video stream is interrupted, some sites require the viewer to start again. Downloading a video means it is possible to watch the full file without interruption.

Secondly, watching a downloaded video inside dedicated player software is much more flexible than watching a frame inside a web browser window. Controlling a video on TV catch-up web sites is quite horrible: cumbersome, flaky and slow, often with poor quality. Most have advertisements. Adverts are fine except that if there's a streaming problem and viewing is interrupted, one usually has to wade through all the advertisements again to continue watching.

Just controlling web browser video players is hard. One has to muck with the mouse or keyboard to pause; fast forwarding and rewinding is unreliable; full screen display often isn't; and play-back is sometimes interrupted or jerky even on a fast connection. Many people use a universal remote to control many functions on the Apple Mac they use to play videos eg. through a TV. A universal remote makes life much easier – it's really hard to use a keyboard and mouse on your lap in front of the TV ! But, universal remotes (and especially the Apple Remote) usually can't control a video playing in a web browser window.

In September 2020, on request of the Recording Industry Association of America (RIAA), the host GitHub disabled *youtube-dl*. In November 2020, *youtube-dl* was restored following a [letter]() sent to GitHub by the Electronic Frontiers Foundation. The letter explained why *youtube-dl* should remain available. It listed uses to which *youtube-dl* can be put:

> "It is used by journalists and human rights organizations to save eyewitness videos, by educators to save videos for classroom use, by YouTubers to save backup copies of their own uploaded videos, and by users worldwide to watch videos on hardware that can't run a standard web browser, or to watch videos in their full resolution over slow or unreliable Internet connections."

Given those reasons, using a video downloader can be ethical. But, I think it's best to adhere to a few principles:

- Do not provide copies of downloaded videos to anyone else – especially do not provide copies online to other people by any means.

- Watch videos then delete them – the result then is similar to what would happen if watching online or recording a broadcast with a PVR for playback later.

- Look for other ways to access videos (eg. buy DVDs, borrow DVDs from the local library, record from free-to-air broadcasts, record or download from Pay TV providers who sell that product).

- If a broadcast is missed look for a repeat or "encore" broadcast before downloading.

Disclaimer: Providing MacYTDL or this information is not to be taken to endorse any breaches of the law or of the terms and conditions imposed by owners of web sites and videos. <u>You should check for terms and conditions on a web site before downloading</u>.

MacYTDL components

MacYTDL includes a number of AppleScript components.

DialogToolkitMacYTDL.scptd

This is an AppleScript library that must be installed for MacYTDL to work. It is stored in the MacYTDL bundle resources folder. The installer copies it from there to the "~/Library/Script Libraries" folder for each user who runs MacYTDL. It is automatically updated.

It is a customised version of the *DialogToolkitPlus* script library created by [Shane Stanley, AppleScript Guru Extraordinaire](#).

Send-URL-To-MacYTDL.workflow

This is a service created in Automator. It enables users to invoke MacYTDL from within other applications. It is stored in the MacYTDL bundle resources folder. When requested by the user, the installer copies it from there to the "~/Library/Services/" folder for each user who runs MacYTDL. When running, the usual rotating wagon wheel icon will show in the "Status" menu bar. Each user's copy is automatically updated if it is installed.

Monitor.scpt

An AppleScript which handles downloads and shows progress. Having a separate script enables MacYTDL to download a number of videos at the same time. It is stored in MacYTDL bundle resources folder.

adviser.scpt

An AppleScript which tracks when a download has finished. If the download was 100% complete it issues a notification accordingly and provides the option of playing the video. If not 100% complete or if an error is reported, the notification advises that there may have been a problem and provides options to open the related log file or the log folder. It is stored in MacYTDL bundle resources folder.

Utilities.scpt

This file contains a number of AppleScript handlers which are called by the main script when required. Using a separate file for a number of handlers helps reduce crashes in my favourite AppleScript IDE. It is stored in the MacYTDL bundle.

External components

MacYTDL uses six tools to download and convert video files.

youtube-dl/YT-DLP

The *youtube-dl* developers' description is best:

> "youtube-dl is a command-line program to download videos from YouTube.com and a few more sites. It requires the Python interpreter (2.6, 2.7, or 3.2+), and it is not platform specific. It should work on your Unix box, on Windows or on macOS."

YT-DLP performs basically the same functions.

youtube-dl/YT-DLP can download videos from many sites around the world including some not on their supported sites list. More detail on *YT-DLP* is available at https://github.com/yt-dlp/yt-dlp.

youtube-dl can be downloaded separately at: https://github.com/ytdl-org/youtube-dl/releases.

YT-DLP can be downloaded separately at: https://github.com/yt-dlp/yt-dlp/releases.

Python scripting language

Python is a script-based programming language that works on most operating systems. More detail is available here: https://www.python.org. Every Apple Mac should have a workable copy of Python (usually version 2.7). However, if there are problems with Python, releases are available from: https://www.python.org/downloads/mac-osx/

Note: *YT-DLP* requires at least Python version 3.6. A Python runtime is included in the *YT-DLP* download used by MacYTDL. Thus, there is no need to install Python3 separately.

FFmpeg

FFmpeg, to quote from the developers, is:

> "A complete, cross-platform solution to record, convert and stream audio and video."

It is used by *youtube-dl/YT-DLP* for a number of functions including to convert downloaded videos.

FFmpeg can be downloaded separately at: https://evermeet.cx/pub/ffmpeg/

FFprobe

FFprobe is a stream analyser. It is used by *youtube-dl/YT-DLP* to collect details on the download video format etc.

FFprobe can be downloaded separately at: https://evermeet.cx/pub/ffprobe/

AtomicParsley

AtomicParsley is a small utility that embeds thumbnail images in audio files. It is the default tool required by *youtube-dl* for that function.

There is no generally available *AtomicParsley* binary for macOS. Therefore, I created a new build on my iMac running macOS 10.13.4 using HomeBrew. The build commands were:

```
sudo chown -R $(whoami) /usr/local/var/homebrew
brew install atomicparsley
```

The source code was for version 0.9.6, dated 25 July 2014, prepared by user *dinkypumpkin* (who also created *Get iPlayer Automator*).

That executable is stored in the MacYTDL bundle resources folder. When requested, the installer copies *AtomicParsley* to "/usr/local/bin". More detail on *AtomicParsley* is available at: atomicparsley.sourceforge.net.

Note: AtomicParsley is not required by *YT-DLP*.

Alerter

Alerter is a utility which issues custom macOS notifications. It is used in MacYTDL to notify when components are being downloaded and when a download has finished.

Because the current version of *Alerter* does not run on versions of macOS prior to macOS 10.15 (Catalina), I built a separate copy in *Xcode* using a duplicate of the *Alerter* source code. The *Alerter* source code was for version 4 released on 27 December 2020. It has been tested to work on macOS 10.13 (High Sierra) and 10.15. The executable is stored within the MacYTDL bundle and is signed with the MacYTDL developer's signature. More detail on *Alerter* is available at: https://github.com/vjeantet/alerter.

Any comments, suggestions or better code ?

Although MacYTDL has been in development for over 4 years, I need all the help available. If you use MacYTDL and have any comments or suggestions please make contact.

The *AppleScript* code in MacYTDL is strictly an amateur effort. If you have *AppleScript* skills and would like to contribute please go ahead. It is best to use the facilities of GitHub to document proposed changes e.g. in pull requests. Note, however, that I normally only publish code changes with each release and there will be changes that have not been published. So, please make contact before embarking on significant changes.

MacYTDL has been developed with a commitment to some guiding principles. Those principles are set out here: https://github.com/section83/MacYTDL/blob/master/Contributing.md.

Send an email to macytdl@gmail.com or post an issue on the MacYTDL GitHub page:

> https://github.com/section83/MacYTDL [logon required to post]

Software Licenses

The MacYTDL Software can be used for personal, non-commercial purposes and can be distributed, copied or uploaded without the author's permission. However, the license of the software must be mentioned when distributing the software for any other third party and a notification must be sent to the author (macytdl@gmail.com) when any of the MacYTDL source code is changed or used in other software. The Software may not be used for direct or indirect commercial (includes non-profit, or revenue-generating business) purposes. You use it entirely at your own risk.

AtomicParsley is protected by the GNU General Public License version 2.0 (GPLv2).

youtube-dl is in the public domain, which means you can freely use it, modify it or redistribute it.

YT-DLP is in the public domain. It is covered by the *Unlicense License*.

Dialog Toolkit Plus is copyright ©2014-19, Myriad Communications (sstanley@myriad-com.com.au). It may be freely distributed and copied. If you modify it, please rename the modified version to avoid clashes. You use it entirely at your own risk. Feedback is welcome.

FFmpeg is a trademark of Fabrice Bellard, originator of the *FFmpeg* project. *FFmpeg* is licensed under the GNU Lesser General Public License (LGPL) version 2.1 or later.

Alerter is protected by *the MIT License*.

Acknowledgements

Much is owed to Shane Stanley, for his many contributions solving problems with MacYTDL. Shane developed *Dialog Toolkit Plus* which provides the Main dialogs in MacYTDL.

Ideas for this GUI front-end came from many sources including:

* "Tombs" (https://forum.videohelp.com/members/235982-Tombs) an active contributor to Whirlpool and author of the *URLDown Dropper* utility for Windows (discontinued).

* Adam Albrec (game_creator@hotmail.com), author of *PPC Media Centre*.

- Michael Page (http://techion.com.au), author of the *Video Hoarder* automator script.

- "kopurando" (https://github.com/kopurando), author of the *Virga* downloader.

- "xplorr" (https://forum.videohelp.com/members/268051-xplorr), author of *TVDownloader*.

- Anonymous (https://cresstone.com/apps/youtubeDLFrontEnd/), for *youtubeDLFrontEnd*.

- Frank, Geoff, John, Santo, Trevor and Walter, fellow members of the ACT Apple Users Group.

- MacYTDL users 11lucasarr11, Adam, Alex, AirMarty, Andy, andyrb412, artcore-c, barney1903, BigJoe309, bovirus, CdrSpock, Dantha, darbid, defcon5at, EuronymousDeadOhlin, hunterbr3193, ItsMorePaul, Jack, janvdvelde, L-Kiewa, LeonardoMaracino, MalEbenSo, martinsstuff, meiwechner, michel-GH, Mike, MrJmpl3, pedrocadiz13, Peter, Rick, roest01, SwineBurglar, Ted, tht7, tigrr, Tobias, Tom, TomasCarlson, upekshapriya, zxzzz8.

Code signing and notarization

MacYTDL is code signed by me and notarized by Apple. These are separate functions which improve the security of macOS applications. They reduce the likelihood that an application has been interfered with after it was released by the developer.

Code signing is a process by which an application is "signed" with a certificate issued by Apple to the developer. macOS will not start an application that has been changed in any way after it is signed. Thus, there is some surety that the application has not be altered by another party eg. to add malicious code.

Notarization is an additional process in which a copy of the application is examined by Apple to ensure it is signed correctly and contains no malicious code. Apple's notarization process is automated. It is not a formal review by an Apple developer. However, it does provide some comfort that Apple have checked the application and found nothing malicious.