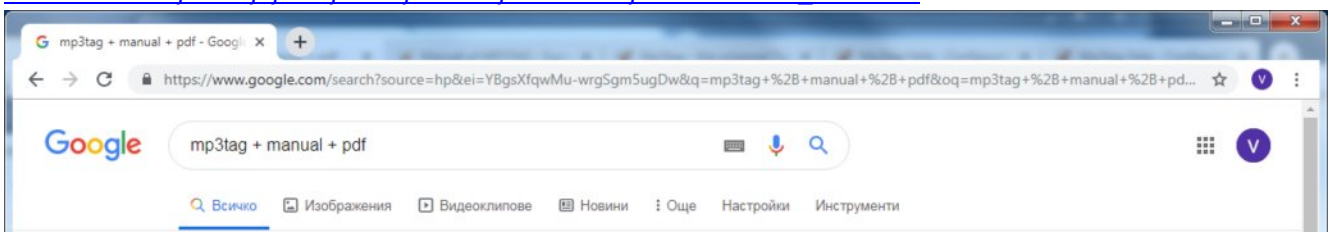
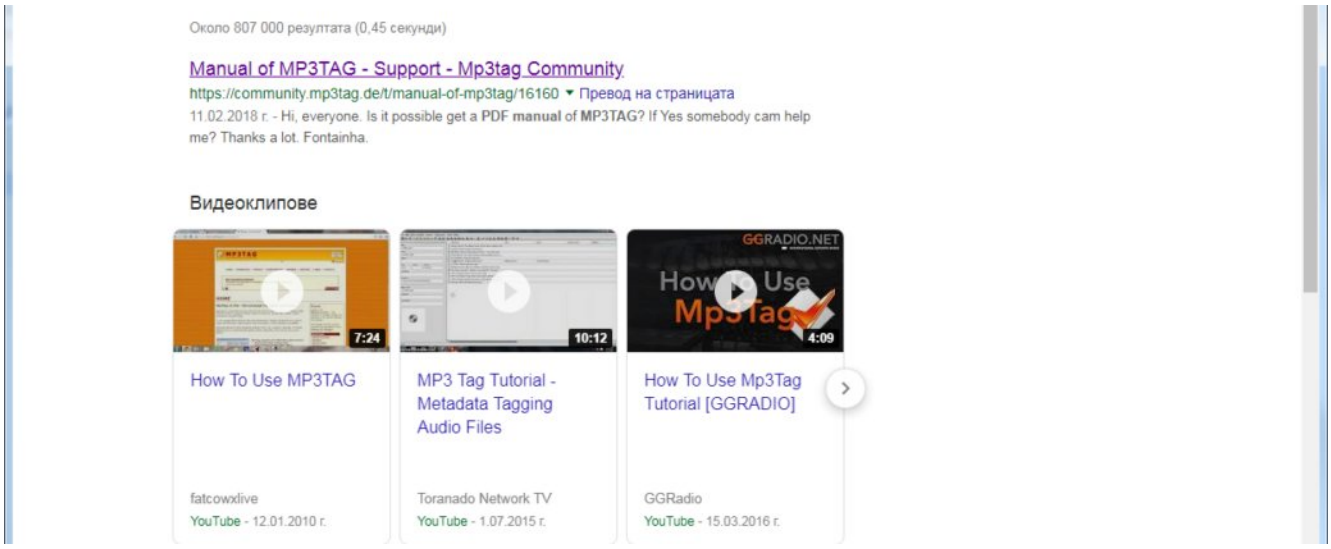
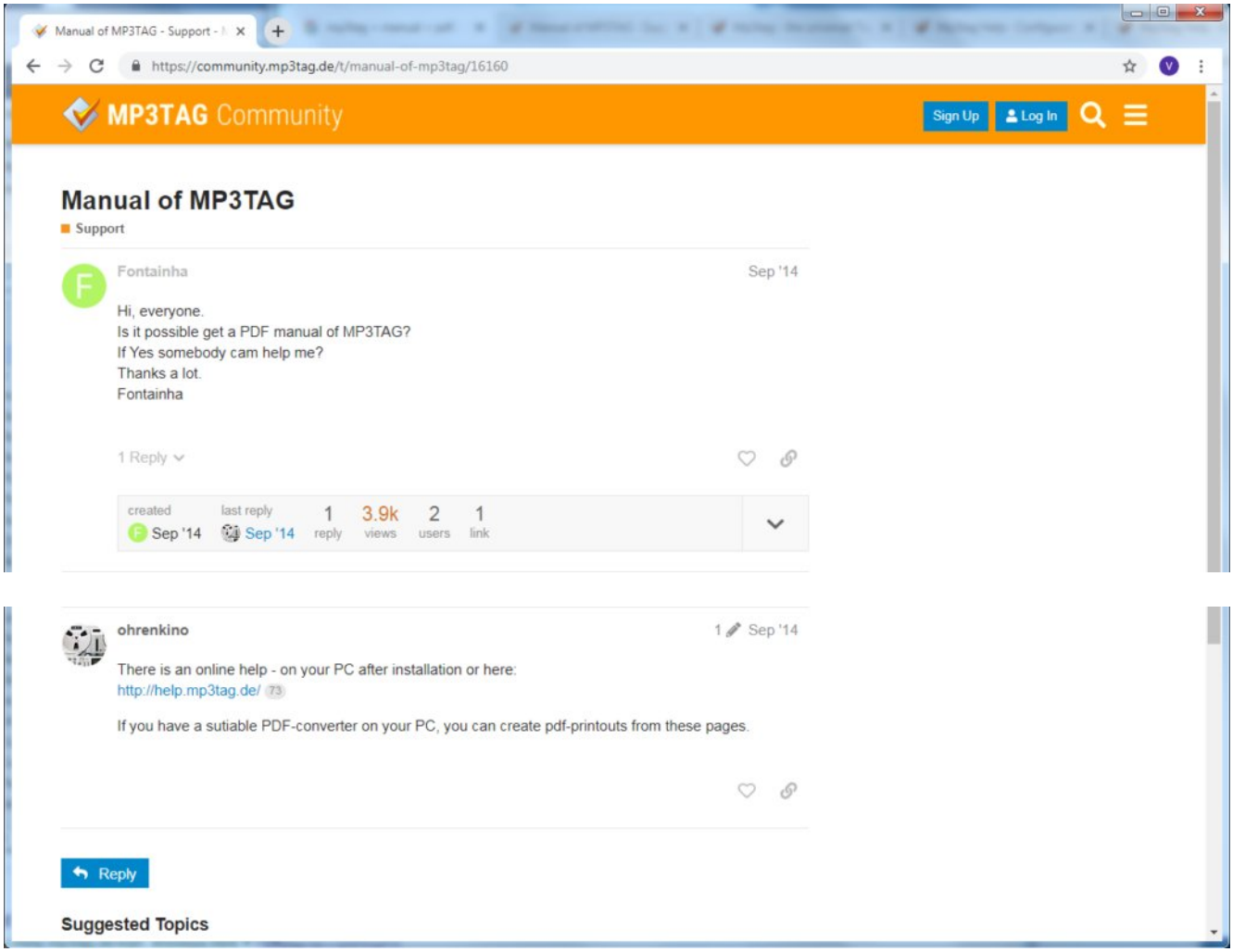
https://www.mp3tag.de/en/:

https://www.google.com/search?source=hp&ei=YBgsXfqwMu-wrgSgm5ugDw&q=mp3tag+%2B+manual+%2B+pdf&oq=mp3tag+%2B+manual+%2B+pdf&gs_l=psy-ab.3..0i22i30l2.1684.12401..12516...0.0..0.146.2250.13j9.....0....1..gws-wiz.....0..35i39j0i131j0j0i10j0i203j0i13i30j0i13i10i30j0i8i13i30.30u_CGVCI6A:





https://community.mp3tag.de/t/manual-of-mp3tag/16160:



http://help.mp3tag.de/:

MP3TAG - The Universal Converter. Mp3tag usage (mp3tag.de)

Contents

MP3TAG - UNIVERSAL TAG EDITOR AND MORE ...

Mp3tag is a powerful and yet easy-to-use tool to edit tags of several supported audio formats. It can rename files based on the tag information, replace characters or words from tags and filenames, import/export tag information, create playlists and more. The program supports online freedb database lookups for selected files, allowing you to automatically gather proper tag information for selected files or CDs.

About the Documentation

The intended purpose of the documentation is to explain the main features of Mp3tag and to highlight features that may not be obvious to first time users. The documentation is not exhaustive.
You should have a good understanding of what tags or metadata of audio files is and an idea of what you want to achieve by using Mp3tag.

Philosophy of Mp3tag

The main idea behind Mp3tag is that easy tasks should be easy to solve while solving advanced tasks should be still possible.

Mp3tag does this by providing a fairly simple interface that allows first-time users to quickly edit tags and file names. Beyond that, Mp3tag's Actions allow for performing arbitrary changes to tags in an automated way. Scripting fuctions and placeholders for tag fields are used in almost every place of the program. Also, Mp3tag doesn't restrict you to stick with a standard set of tag fields by allowing flexible tagging, i.e., user-defined field names for tags. While these tags are not supported by all audio programs, Mp3tag writes them in a way that conforms to the various specifications of the different tag formats.

Beyond that, Mp3tag's philosophy is also to *educate the user*. So to leverage Mp3tag's full potential, learning about Regular Expressions, creative ways to use the export feature and the unlimited possibilites offered by the built-in scripting function is much recommended.

Good places to start learning more about this program are:

- **First Steps**
- Convert Tag-information from and to filenames
- Overview of Actions to automate changes
- Export Tag-information

Technical requirements

- PC with Microsoft® Windows® XP SP3, Windows® 2003, Windows® Vista, Windows® 2008, Windows® 7, Windows® 8, Windows® 8.1 or Windows® 10
- Internet-Browser to view the Online-Help
- Connection to the internet to use the Tag Sources feature

First Steps

- Saving Tags to files
- Removing Tags from files
- Case conversion on filenames and tags
- Customizing the file view

Saving Tags to files

Step 1

First change to a directory containing supported audio-files by using the command File > Change directory...
The file view now contains all the files of this directory. If you also want to display the files of the subdirectories, please enable the option **Subfolders** in the change directory dialog.

Step 2

Now select the file to edit by using the mouse or the keyboard. The tag of the selected file will be displayed in the entry fields on the left of the main window.

If there are more file selected, the entry fields will show < keep > so that existing values are preserved.

Step 3

Please enter the tag-information of the current file to the entry fields.
If one field contains the text **< keep >**, Mp3tag will keep the current content of this tag-field when saving the new tag.
If one field contains the text **< remove >**, Mp3tag will remove the current content of this tag-field when saving the new tag.

Step 4

Save the new tag by using the command File > Save tag or use the disc button at Mp3tag's toolbar.

Note

Mp3tag always stores ID3v1- and ID3v2-Tags to MP3 files. You can change these settings via Options > Tags > Mpeg.

Removing Tags from files

Step 1

First change to a directory containing supported audio-files by using the command File > Change directory...
The file view now contains all the files of this directory. If you also want to display the files of the subdirectories, please enable the option **Subfolders** in the change directory dialog

Step 2

Now select the file(s) by using the mouse or the keyboard.

Step 3

Remove the Tags of the selected files by using the command File > Remove tag or use the delete button at Mp3tag's toolbar.

Note

Mp3tag always removes ID3v1- and ID3v2-tags from files. You can change these settings via
Options > Tags > Mpeg.

Case conversion on filenames and tags

Step 1

First change to a directory containing supported audio-files by using the command  File > Change directory...
The file view now contains all the files of this directory. If you also want to display the files of the subdirectories, please enable the option **Subfolders** in the change directory dialog.

Step 2

Now select the file(s) you want to edit by using the mouse or the keyboard.

Step 3

Run the command  Convert > Actions or just use the icon from the toolbar.

Step 4

Create a new action group by choosing the  **New** button and give this action group a meaningful name, e.g. "All uppercase".

Step 5

Create a new action by choosing the  **New** button and choose **Case conversion** as action type.

Step 6

Activate the check mark of the action group and run the action group via the OK button.

Customizing the file view

You can add and remove user-defined columns in the file view of Mp3tag.
Just display the column dialog via the command View > Columns... and follow the instructions on this page.

Important Changes

Release Notes

The release notes describe new features and notable changes. Please have a look at the Changelog for a complete list.

Mp3tag is Freeware

If you like using Mp3tag and want to help keeping the project alive, please consider donating to support the development of Mp3tag.

Mp3tag v2.96

Support for Opus with *.ogg file extension

So far, tagging Opus files required a *.opus file extension in Mp3tag. However, many applications are simply using Opus with the *.ogg file extension, which is perfectly valid. With this release, Mp3tag detects Opus for files with the *.ogg file extension — so no more renaming for you.

Improvements when copying cover art via clipboard

Mp3tag now also copies the original data next to an independent DIB format when copying cover art via the Windows clipboard. Previously, the independent DIB was re-encoded to a maximum quality JPEG, which was not the best option when the image was stored as, e.g., a PNG. Now, if you paste the cover to another file, the original format and data is used.

Further changes

I've also fixed many issues and performed many changes here and there. For a complete overview of all changes, please see the Changelog.

Mp3tag v2.95

Toggling of file-list columns

You can now toggle file list columns via the context menu on the column header. This way it's easy to enable and disable columns you only need in certain conditions. It's a long-requested feature and I'm already using it quite often.

Nested action menus

If you're one of the aficionados of Mp3tag's action feature, you can now create nested action menus by using the # character in the action group name.

Support for subtracks at Discogs

The Discogs tag source now supports subtracks. I think it's great and it's probably most welcomed by people who have lots of classical music to tag.

Further changes

I've also fixed many issues and performed many changes here and there. For a complete overview of all changes, please see the Changelog.

Mp3tag v2.94

Support for discnumbers at auto-numbering wizard.

The auto-numbering wizard can now, optionally, also assign discnumbers to the selected files. New discs start when the directory changes.

History for Replace and Regexp actions

The actions 'Replace' and 'Replace with regular expression' now have a history drop-down that remember the previous inputs.

Remember inputs for last used Replace or Regexp quick actions

The quick actions 'Replace' and 'Replace with regular expression' now remember the last used inputs.

UI improvements at Tag Sources dialog

The tag sources dialog now has a splitter between the lists for results and files, which allows for resizing the lists individually without resizing the whole dialog. Also, the selection from the files list to results list is now synchronized.

Option to disable showing release notes on update

If you prefer to not have these notes shown when you update to a new version, you can disable displaying release notes for new versions at "Options > Updates".

Further changes

I've also fixed many issues and performed many changes here and there. For a complete overview of all changes, please see the Changelog.

Mp3tag v2.93

Hotfixes regarding MP4 tagging

That was quick! v2.92 had some issues wrt. MP4 tagging and I've decided to pull a hotfix release to make sure that as many people as possible are using the fixed version.

Mp3tag v2.92

Tag-Panel field size *Smaller*

The Tag Panel continues to get love and it now has a new field size *Smaller* — which is smaller than *Small*. It's perfect for track- and disc numbers and can be nicely combined with another *Smaller* and one *Medium*-sized field to fill up one row and use the available space in the most efficient way.

Configurable height of multiline fields on Tag-Panel

Many users who are using multiline fields on the Tag Panel (e.g., for displaying UNSYNCEDLYRICS) were asking for a way to configure the height of those fields. It's now possible to decide on the number of rows that are occupied by a multiline field (one row is the combination of label and input of a normal-sized field).

Advanced configuration option to toggle listing chapters as separate files

This new option gives the ability to decide of chapters are listed as separate files or are simply not listed. It's especially helpful if you're into editing video files (MP4 or Matroska) and always wanted to edit the global tags of a file instead of the titles of the individual chapters. The option is located at "Options > Tags > Advanced" and is disabled by default.

Information fields for subsongs

Especially if the option to list chapter as separate files is disabled, it can be difficult to identify which files actually have chapters. I've added two new information fields to

identify those files, %_subsong_count% provides the number of subsongs in a file and %_subsong% contains the subsong index.

Information fields for video dimensions

I have the impression that more and more users are also handling their video library with Mp3tag. This new version also has two new information fields for video dimensions, namely %_video_width% and %_video_height% for Matroska files (and MP4 video as well).

Further changes

I've also fixed many issues and performed many changes here and there. For a complete overview of all changes, please see the Changelog.

Mp3tag v2.91

Copying of individual tag fields

A long-awaited feature I've implemented is the ability to copy individual fields via the extended tag dialog at Alt+T. Just select the relevant fields you want to copy, right-click and choose the appropriate entry from the popup menu.

Default values for Tag-Panel fields

Tag-Panel fields have their default values back. Just better :) You can now configure a default value that is set whenever you select multiple files directly via the field configuration at Options > Tag Panel. Double-click on the desired field and choose the default value.

Further changes

I've also fixed many issues and performed many changes here and there. For a complete overview of all changes, please see the Changelog.

Mp3tag v2.90a

Updated Code-Signing Certificate

I've updated the code-signing certificate that is used to digitally sign and verify the origin and integrity of Mp3tag's installer and executable.

Microsoft Store version

I've released Mp3tag to the Microsoft Store. It's still available for free and if you prefer getting your software from there you can give it a go.

Further changes

I've also fixed many issues with this release. For a complete overview of all changes, please see the Changelog.

Mp3tag v2.89

Improved Tag Panel

I've made the following improvements to the Tag Panel:

- many layout changes when displayed horizontally,
- all fields are now configurable (including default fields),
- added context menu to customize fields,
- added option to specify size when adding fields (normal, smaller, small, multiline),
- added dedicated separator field (_SEPARATOR) to allow for manual column breaks,
- added option to enable/disable fields, and
- added loading and saving of Tag Panel configurations.

Support for long file paths

Mp3tag now supports file paths longer than 260 characters, so also very wordy titles can be renamed properly.

Updated Discogs and MusicBrainz tag sources

Discogs changed their servers to require connections via TLS 1.2 (something very secure, if this sounds cryptic to you). I've added support for TLS 1.2 for Windows 7 and 8 (it's used by default starting from Windows 8.1). Unfortunately, TLS 1.2 is not available on Windows XP and Vista, so Discogs stops working on these systems.

MusicBrainz updated their server to a new search infrastructure and with that, some minor changes to the MusicBrainz tag sources were necessary.

Updated MusicBrainz identifier mapping

I've added various internal mappings so that fields for MusicBrainz identifiers are now mapped to the correct fields (e.g., MUSICBRAINZ_ALBUMID is mapped to 'MusicBrainz Album Id').

Option to keep moved files in file list

With Mp3tag v2.87, I've implemented the unpopular change that moved files are always kept in the file list and can be manually removed. I've now made this behavior optional so that you can decide depending on your preferences.

Further changes

For a complete overview of all changes, please see the Changelog.

Mp3tag v2.88

Going Deep

I've taken a deep dive in the bug reports section of the forums and fixed many, many long-lasting and not so old issues with this release.

Layout and other UI changes

I've reworked the layout of the Tag Panel in horizontal mode and changed Mp3tag to use a modern, default font in the main window. This one makes Mp3tag look more beautiful on modern Windows systems. This also implies, that any system-wide font changes are also visible in Mp3tag now.

Addressing memory above 2GB on 64 bit systems

This one is huge for people with huge libraries. Mp3tag can now allocate up to 4GB of your RAM. That's impressively much.

Displaying cover art dimensions in Mp3tag's file view

You can now use %_cover_width% and %_cover_height% for width and height of the first embedded cover art. It's also available in the filter, in the converters and everywhere else in Mp3tag.

Further changes

For a complete overview of all changes, please see the Changelog.

Mp3tag v2.87

Spring Maintenance

I've spent the last two months hunting bugs and issues and fixed many of them with the current release. Given the list of fixed issues, it feels like this is the most stable release of Mp3tag ever :)

MusicBrainz v2

The MusicBrainz v1 WEB service will be closed in a few months, so I've added a new official MusicBrainz Tag Source based on v2 of the MusicBrainz WEB service with this release.

Option to disable auto-filtering

The filter helper menu now has an option to enable/disable auto-applying the filter. This can be helpful if you're working with large libraries and find the auto-filtering to disturb your workflow.

LAME MP3 encoder version

I've rewritten the code to analyze MP3s with this version (it was sooooo old) and added detection of LAME encoder versions while I was at it. You can access it via the %_tool% information field, if this is something of interest to you.

Mapping of Matroska tag fields

I've reworked the tag fields mapping for Matroska and documented the process at the overview page for Tag field mappings. If you're tagging Matroska files, this is for you.

Further changes

It's quite a list for this version. You can get a complete overview via the history of changes.

Mp3tag v2.86

Library to permanently store metadata

Normally, Mp3tag reads metadata from files when loading files, directories or playlists and stores the data in memory. Then, when you close Mp3tag, the data once read from the files is discarded and must be read again the next time you're working with those files in Mp3tag.

This can become both time consuming and memory intensive, especially when dealing with large amounts of files. If you enable the Library in Mp3tag (see Options > Library), metadata is first read from the files and then stored in a database that resides in Mp3tag's configuration folder. On subsequent reads of the file, the

metadata is read directly from the database and updated iff the file's modification timestamp has been changed. In addition to that, cover art or other binary data is not kept in memory but is stored in the database, which can reduce the amount of memory that is consumed by Mp3tag significantly.

This is a quite extensive change and required to rewrite huge parts of Mp3tag base layers. It's my hope that it improves performance and serves as a basis for interesting future improvements.

Further changes

This version also fixes many issues — some reported recently, some were a little bit older. You can get a complete overview via the history of changes.

Mp3tag v2.85

Support for Matroska (MKA/MKV)

With the recent support for multi-track files, it was about time to add support for the Matroska container format. Similar to MP4, Matroska can contain both audio and video tracks. This means, that you now have another option to tag your video collection with Mp3tag. Mp3tag currently displays the default (or forced) audio track and all included chapters. I'm curious about your experience, so if you're going to use Matroska with Mp3tag please let me know what you think.

Extended list of default genres

I've extended the list of default genres to include all extensions made by Winamp (ID3v1 genres 148 to 191) which were included by several other programs. It now includes genres like Audiobook and IDM which I find quite nice. If you're using only a small set of genres, you can use 'Options > Genres' to add your own.

Further changes

This version also fixes many issues — some reported recently, some were a little bit older. You can get a complete overview via the history of changes.

Mp3tag v2.84

Support for Nero-style MP4 chapters

MP4 is a container format — which basically means that you can put loads of stuff into it. With Nero-style tags, it’s even possible to have one file consisting of multiple sub-tracks. The Nero chapter tags describe the position and the sub-track specific tags. With this new version, Mp3tag displays individual sub-tracks of such files and also allows for editing of those individual chapter tags. Tags fields with the same contents across all sub-tracks are used as global file tags when writing changes.

Cover-only Tag Sources to import album-art from Discogs

Many of you wrote me and wished for a cover-only alternative to the now discontinued Amazon Tag Sources. Here you go :) With this new version, Mp3tag comes with two cover-only Tag Sources to import cover art from Discogs.

Further changes

This version fixes many, many issues — some reported recently, some were a little bit older. You can get a complete overview of how busy I've been via the history of changes.

Mp3tag v2.83

Support for querying Discogs via Release ID

Sometimes, searching by album or artist name simply doesn't give the correct results. This version adds a new tag source to directly search by Discogs Release ID. This means, that you can directly jump to the correct result when importing tags from Discogs.com.

Import from Amazon no longer available

Unfortunately, Amazon has closed my associate account so that importing tags via Amazon is no longer possible. A highly recommended alternative tag source is Discogs, which offers high-quality metadata and cover art.

Support for removing Nero-style tags from MP4 files

Nero-style tags are one possible reason why tags cannot be saved to MP4 files. This version adds basic detection of those tags and reporting via %_tag% if present. Furthermore, you can now remove Nero-style chapter information from MP4 files via context-menu in the file list.

Support for restricting cover export to certain cover types

The Mp3tag action to export cover-art to files now has a option to restrict the covers to be exported to a certain cover type. This means, that you can now export, e.g., all the front covers of your files in one go.

Improvements at reading and writing cue sheets

It's now possible to read cue sheets that don't start with Track 1 and many other things. If you're into working with cue sheets, you'll probably like the changes in this release.

Further changes

Furthermore, this version fixes some issues which were reported during the last months as denoted in the history of changes.

Mp3tag v2.82

Spring Maintenance

This version is a spring-cleaning maintenance release and fixes issues which were reported during the past weeks.

Please see the history of changes for a complete list.

Mp3tag v2.81

Support for Portable Installation and Admin-Elevation

A long-requested feature of many users was a portable installation of Mp3tag, i.e., a installation that installs in one directory and also stores its configuration settings in this directory. This is now possible. And if you choose the standard installation, Mp3tag automatically elevates the process to the necessary user rights (after asking you, of course).

Support for iTunes Grouping Field

As mentioned in the release notes of Mp3tag v2.80, iTunes was writing the grouping information to its internal database only. With an updated version of iTunes, this information is now written to a new iTunes-specific ID3v2 field which is available in Mp3tag via the field name GROUPING.

Further changes

Furthermore, this version fixes some issues which were reported during the last months as denoted in the history of changes.

Mp3tag v2.80

Support for iTunes tag fields for classical music

In October, iTunes introduced new tag fields for better tagging of classical music. Those fields include work name, movement name, number and total count and are mapped in Mp3tag to these new field names:

- MOVEMENTNAME — Movement name
- MOVEMENT — Movement number
- MOVEMENTTOTAL — Movement count
- WORK — Work name
- SHOWMOVEMENT — Flag to trigger showing of classical fields instead of normal title in iTunes (1 = yes)

While the implementation of these fields in iTunes is not yet realized completely consistent, Mp3tag supports all fields for MP4 tags. For MP3 files, iTunes currently writes the work name to the CONTENTGROUP field (while what's added at "Grouping" is only stored in iTunes' internal database). The same applies to the flag SHOWMOVEMENT which is also stored in iTunes' internal database and not in the file itself. There are already several bug reports for that reported to Apple and I expect a consistent implementation with the next release of iTunes.

Further changes

Furthermore, this version fixes some issues which were reported during the last months as denoted in the history of changes.

Mp3tag v2.78

Support for AIF / AIFF audio format (*.aif, *.aifc, *.aiff)

The main feature of v2.78 is full support for the AIF / AIFF audio format. This includes support for ID3v2 tags (including user-defined fields and cover art) as dedicated RIFF chunks. The new version reads, writes and removes those tags and – as usual – also reports meaningful technical information about the file.

Support for iTunes MP4 owner information

This version adds support for the ITUNESOWNER field, which shows the owner of an MP4 file which has been bought via the iTunes Store. Besides ITUNESACCOUNT, this enables you to make sure that all files are from your own music library. Please note that removing those fields doesn't remove the (hidden) buyer information from the file.

Further changes

Furthermore, this version fixes some issues which were reported during the last months as denoted in the history of changes.

Mp3tag v2.76

Support for WAV audio format (*.wav)

The main feature of v2.76 is full support for the WAV audio format. This includes support for RIFF INFO tags (which are surprisingly limited) and writing ID3v2 tags (including user-defined fields and cover art) as dedicated RIFF chunks. The new version reads, writes and removes those tags and – as usual – also reports meaningful technical information about the file.

Further changes

Furthermore, this version fixes some issues which were reported during the last months as denoted in the history of changes.

Mp3tag v2.74

Support for HTTPS in WEB Sources Framework

Since more an more APIs are changing their access methods to require authentication or to be HTTPS-only, this version adds support for secure connections to WEB sources via HTTPS. This is a rather huge change since the complete HTTP connection part has been replaced in the course of this update.

Discogs access via HTTPS

As a prominent example of the changes described above, Discogs announced to be available via HTTPS only starting from March 14. The official Discogs WEB source of Mp3tag has been updated to use HTTPS with this update. In the background, I've already changed the Mp3tag Discogs image cache that is used to provide Discogs images to Mp3tag users to use HTTPS as well.

Support for Podcast Keywords

Mp3tag becomes more and more popular in the podcast community and already supports a variety of podcast-related tag fields (e.g., podcast category, podcast description, podcast ID and podcast URL). This version adds support for podcast keywords via the PODCASTKEYWORDS field for ID3v2 and MP4 tags.

Compatibility Improvements for ID3v2 Comments

When it comes to ID3v2 comments, iTunes and Windows Media Player seem not to like each other very much. While iTunes requires the ID3v2 comment language to be set to 'eng' (English), Windows Media Player and Windows Explorer need the comment language set to the language of the Windows version that is running. Otherwise, the comments are simply not shown in these applications. In the past Mp3tag offered a quite tedious via editing the language file to circumvent these compatibility issue. With this version, the ID3v2 comment language can be set via a configuration dialog which is available via 'Options > Tags > Advanced' and will preserved also when installing new versions of Mp3tag. So no more fiddling in the language files for you.

Further changes

Furthermore, this version fixes some issues which were reported during the last months as denoted in the history of changes.

Mp3tag v2.73

Automatically Select Files

In the past, many users have wished for an option to automatically select files that have been added to Mp3tag's file list. With this version, I've added a new configuration option to 'Options > General' that allows for automatically selecting files. When this option is enabled, files that are loaded when Mp3tag started or are added via drag'n'drop or are added via the right-click option in Windows Explorer are selected automatically.

Selecting All Files via Toolbar

Some users prefer working via keyboard shortcuts, others are effectively using the mouse. To select all files, the most efficient way was to use the Ctrl+A keyboard shortcut. This might change now, especially for the mouse users amongst you, because the current version now features a new toolbar button for selecting all files in Mp3tag's file list.

File Selection when Filtering Files

Till now, the file selection was not preserved when switching from a filter result to the complete file list via F3 or 'View > Filter...'. This had the significant drawback, that files which were identified using the filter and selected in the filtered file list, were not seen in context of the complete file list anymore once the filter has been reset. With this version, the file selection is preserved when disabling the filter.

New Bit-Depth Information Field

Some audio formats offer encoding in different bit depths. This information is now available using the %_bitspersample% information field for audio formats that support different bit depths.

Further changes

Furthermore, this version fixes some issues which were reported during the last months as denoted in the history of changes.

Mp3tag v2.72

Support for Windows 10

Mp3tag now officially supports Windows 10. It worked quite smoothly already when Windows 10 was released, but I adjusted minor things here and there. If you find any quirk or think you've found a bug please let me know.

Updated Code-Signing Certificate

To verify the integrity and the origin of the Mp3tag setup and Mp3tag program file, I've updated the code-signing certificate.

Mp3tag v2.71

Summer Maintenance

This version is a summer maintenance release and fixes issues which were reported during the past weeks.

Please see the history of changes for a complete list.

Mp3tag v2.70

DPI Awareness and High DPI Support

To be honest, when I first saw a screenshot of Mp3tag on a high-resolution 4K screen I had to laugh: the toolbar icons were ridiculously small and also other screen elements were drawn too small — while the app was still usable in itself. At that point I didn't knew yet, what it takes to make an app fully DPI aware.

Over the last month I worked on making Mp3tag more DPI aware and touched all the different code parts of drawing and moving user-interface elements. It was quite a ride and it took me to places in Mp3tag's code base which I hadn't visited for a long time.

Furthermore, I've redrawn the toolbar icons in 32x32 and 24x24 sizes. I'm not an icon designer so it took me some time to get it done. I did a lot of experiments (even with flat black and white icons) but ended up with a set which sticks very much to the original icons which you (— and I) are already familiar with. But some icons changed their face or got a new finishing and I hope you like it.

Further changes

Furthermore, this version fixes some issues which were reported during the last months as denoted in the history of changes.

Mp3tag v2.68

Discogs Image Caching — Reloaded

Discogs made a breaking API change on 2015-02-20 so that we had to update our Discogs image caching server and the way Mp3tag signs the image requests.

Further changes

Furthermore, this version fixes some issues which were reported during the last months as denoted in the history of changes.

Mp3tag v2.66

Winter Maintenance

This version is mainly a maintenance release which comes with many improvements and fixes various issues that have been reported over the last couple of weeks. Please see the history of changes for a complete list.

Mp3tag v2.65

Discogs Image Caching

Since Discogs implemented rate-limiting on images requests through their API, users of the Discogs WEB Source experienced quite often, that no images were provided by the WEB source—1000 request per day for all users of Mp3tag is simply not enough.

Over the last weeks, we've implemented a Discogs image cache on the Mp3tag servers which is now asked for images before the actual Discogs servers. This first reduces load on the Discogs servers, but also increases the chance that more and more images are served to you.

Mp3tag v2.64

Discogs API changed to JSON

Yet another Discogs-related change, you might be thinking—at least we did, when we read the corresponding announcement on their API development forums :-) This change removed the XML format from the API and changed the API format to JSON, which resulted in the official Discogs Tag Source not working anymore for you.

With this release, we have developed a new official Discogs Tag Source based on the new JSON format. This new format also is more robust w.r.t. changes in the API. If you notice any quirks, releases that cannot be tagged or other things, please let us know.

JSON-related functions for WEB Sources Framework

We've extended the WEB Sources Framework by JSON-related functions that enable the development of faster, more reliable, and easier to maintain Tag Sources. We're looking forward to any community-based development in this area!

Further changes

Furthermore, this version fixes some issues which were reported during the last months as denoted in the history of changes.

Mp3tag v2.62

Discogs API access changed to OAuth

Starting at August 15, 2014 Discogs restricts access to their API to authentication using OAuth. As a user of the Discogs Tag Source you need to have an account on the Discogs WEB site. If you don't have a user account yet, you can create a free account on the discogs WEB site via

https://www.discogs.com/users/create

When you use any of the Discogs Tag Sources from Mp3tag, Mp3tag will ask you to authorize access to your Discogs profile. You'll then have to authorize the Mp3tag application on your Discogs profile and enter the authorization code when Mp3tag is asking you to do so.

Further changes

Furthermore, this version fixes some issues which were reported during the last months as denoted in the history of changes.

Mp3tag v2.61

Summer Maintenance

With this version we're doing some summer maintenance and fix issues which were reported during the past weeks.
Please see the history of changes for a complete list.

Mp3tag v2.60

Update of Discogs Tag Source

Discogs updated their API which resulted in release information not being retrieved while using the discogs tag source. The access to the Discogs API has been fixed with this release.

Internal Restructuring

While this is not a feature that is visible to the user, it's worth mentioning that over the last weeks we did some major refactoring of Mp3tag's code base by changing to use standard container libraries and algorithms.

Further changes

Furthermore, this version fixes some issues which were reported during the last months as denoted in the history of changes.

Mp3tag v2.59

Support for Direct Stream Digital (DSF)

With this version Mp3tag supports Direct Stream Digital (DSF) files with ID3v2 tags.

Further changes

Furthermore, this version fixes some issues which were reported during the last months, improves some internal parts and makes use of the taskbar progress feature which was introduced with Windows 7.

Mp3tag v2.58

Improved performance when reading MP4 files

Mp3tag now reads MP4 files faster than before which should be significant on large collections (many files) or collections including MP4 video (large files).

Improved compatibility of YEAR in ID3v2.3

With Mp3tag v2.53 released almost one year ago we decided to combine the release year (ID3v2.3 frame TYER, displayed as YEAR) and release date (ID3v2.3 frame TDAT, displayed as DATE) in YEAR and display it as YYYY-MM-DD. The intention was to improve compatibility with other applications and the ID3v2 standard.
Since then we've received numerous support requests both on the Mp3tag Forums and via email from users who are unsatisfied with that change. Either because they use a different formatting of date values, use some value denoting a specific time frame (e.g., the 80s) or simply wanted to preserve values that were not fitting in the new date formatting scheme.
With Mp3tag v2.58 we revert this change to the previous behavior.

Mp3tag v2.55

Support for IETF Opus

Mp3tag now supports the IETF Opus codec. More information can be found on the official Website and on Wikipedia.

Mp3tag v2.54

Support for Windows 8

Mp3tag now officially supports Windows 8 operating system. If you find any quirk or think you've found a bug please let us know at support@mp3tag.de.

Mp3tag v2.53

Improved support for Drag & Drop of cover art

Mp3tag now uses a different method for drag & drop of cover art which does not require re-encoding the cover image again but instead uses the original image whenever possible. In addition, support for drag & drop of cover art from Google Chrome has been added.

Improved compatibility of YEAR in ID3v2.3

The release year (ID3v2.3 frame TYER, displayed as YEAR) and release date (ID3v2.3 frame TDAT, displayed as DATE) are now combined in YEAR and displayed as YYYY-MM-DD. This improves compatibility with other applications (e.g., foobar2000) and provides a consistent view with all revisions of the ID3v2 specification.

Mp3tag v2.52

Option to specify cover art directory

With this option located at 'Options > Directories' you can specify a default directory that is displayed when adding cover art. Besides fully-qualified path names, you can provide relative path names and use placeholders. If no default directory is specified or the directory cannot be opened, Mp3tag opens the directory of the first selected file iff it contains cover art or the current working directory in all other cases.

Mp3tag v2.50

Converter Tag — Tag

This converter formats tag fields by other fields content, which means that you can use it to copy tags from one field to another or to combine contents of multiple different tag fields in one tag field. As already known from the other converters, you can use placeholders for any tag field and scripting functions within the format string to format the content of the specified tag field.
Furthermore, the format string is evaluated while typing and you can verify the results using the built-in preview.

Unicode Playlists in UTF-8 format

You can now save playlists in Unicode format encoded as UTF-8 if you use *.m3u8 as file extension. This can be useful if file names contain special characters. Applications which support UTF-8 encoded playlists are, e.g., Apple iTunes and foobar2000.

Discogs API 2.0

Discogs.com released a completely redesigned API for accessing data from their Website based on the popular JSON format. Mp3tag's WEB source framework and the corresponding discogs WEB source have been updated to this new format.

Mp3tag v2.49

Support for unknown MP4 tags

Previous to this version, MP4 tags that were unknown to Mp3tag were deleted from the files, which was an issue for some users. With this version, Mp3tag keeps those unknown MP4 tags (also called atoms).

Support for wildcards at cover import from file

The actions **Import cover from file** and **Import text file** now support wildcards * (matches zero to many characters) and ? (matches exactly one character). This should ease cover import from randomly named files significantly.

Updated WEB Sources

The MusicBrainz and discogs WEB Sources are now both based on the corresponding WEB service API which makes them more robust to changes on the respective WEB sites.

Mp3tag v2.48

This version is mainly a maintenance release which updates the Amazon WEB Sources and improves Amazon query handling to decrease the amount of needed requests per query. As soon as enough users updated to this new release, the request limit enforced by Amazon should not be an issue anymore.

Mp3tag v2.47

Tag fields renamed

With this release some of the tag fields that are used for format strings, actions, columns, exports, and filters are renamed to have more accessible and intuitive names. This also means that all parts of Mp3tag where you have used the old names have to be updated. We know that this is an invasive change but think that it was a step necessary to keep it everything maintainable and understandable for new users of the program.
Please see the list below for details on the renamed tag fields:

- BAND to ALBUMARTIST
- ALBUMSORTORDER to ALBUMSORT
- ARTISTSORTORDER to ARTISTSORT
- BANDSORTORDER to ALBUMARTISTSORT
- COMPOSERSORTORDER to COMPOSERSORT
- TITLESORTORDER to TITLESORT
- TVSHOWSORTORDER to TVSHOWSORT
- ITUNESCOMPILATION to COMPILATION
- ITUNESPODCAST to PODCAST
- ITUNESPODCASTCATEGORY to PODCASTCATEGORY
- ITUNESPODCASTDESC to PODCASTDESC
- ITUNESPODCASTID to PODCASTID
- ITUNESPODCASTURL to PODCASTURL

Some of them might be familiar to you, others might be new. If you are wondering which other fields are available, please see Names of tag fields for a complete list.

Actions menu

The actions feature which was well hidden under the converter menu now has a dedicated top-level menu where all action groups are accessible via menu items. If you are not familiar with the actions in Mp3tag please have a look at the Actions help page. It is a powerful feature and worth looking at.

Amazon

Amazon has limited access to their WEB service to 2000 requests per hour. That means that you might get an occassional error 503 when using the Amazon tag source. However, the limit is reset each hour so you can try again later if it is not working for you.

Mp3tag v2.46

Explorer Shell Extension

The right-click menu extension is now realized by a dedicated Shell extension which improves opening files and directories via the context menu also for large amounts of files.

Improved user-defined field mappings

The user-defined mappings for field names introduced in Mp3tag v2.45 now have an own help page and were further improved with regard to mapping multiple field names to one user-defined field name.

Improved Tag Sources dialog

The Tag Sources dialog now also supports reordering of files via drag and drop allowing to quickly rearrange files according to the titles. Yes, we made it work with multiple files too :-) The sorting and selection state of the Tag Sources search results are now also remembered if you decide to go back and pick another result.

Portable settings

Previous versions of Mp3tag stored toolbar and panel positions in the Windows Registry. This changed with this release so that Mp3tag now stores all configuration data in its own configuration files.

Mp3tag v2.45

Improved Filter

The filter now supports filter expressions where it is now possible to combine filter criteria and use format strings for filtering. The filter now also comes with a history, providing easy access to your often-used filter expressions.

User-defined field mappings

Mp3tag now supports user-defined mappings for field names. This is particularly useful in case you need to make field names consistent across different types of tags. The feature is available via 'Options > Tags > Mapping' where you can create mappings for a specific tag type (APEv2, ID3v2, MP4, VorbisComment or WMA), provide a Source name (Mp3tag's internal name) and a target name (your preferred field name).

Multiline fields for Tag Panel

Are you adding lyrics, artist biographies or other text-intensive information to your songs? If so, you can now add multiline custom fields to the Tag Panel via Options > Tag Panel.

Mp3tag v2.43

Automatic Updates

This version comes with support for automatic checking for updates on program startup. You can decide whether you want to get notified on stable final releases only, or whether you want the freshest stuff from our compilers (Beta versions). If you don't want Mp3tag to check for updates you can disable it completely at 'Options > Updates'.

Horizontal Tag Panel

While it was always possible to drag the Tag Panel and dock it above or below Mp3tag's file list, the input fields are now arranged more effectively. It's also possible to disable the directory switcher from the Tag Panel.

New Translations

Mp3tag is now also translated to Danish, Hebrew, Vietnamese, and Ukranian language.

Mp3tag v2.34 - Support for cover art / Import from Amazon.com

Support for cover art

With the new release Mp3tag supports embedded cover art for files with ID3v2 tags, TTA- and WMA files. You can add cover art via the  extended tag dialog at View > Extended Tags... or Alt + T.

Import from Amazon.com

Besides the freedb support, Mp3tag now has the ability to get album information from Amazon.com. Just choose the small arrow from the right of the Icon  and select **Amazon.com**. You'll get a list of matching items and can choose the album to import the tags (and cover art).
The Amazon.com import uses Mp3tag's incredible WEB source framework. There are even more WEB sources available at the WEB Source Archive on the Mp3tag forums.

Please visit the Mp3tag forums if you have other questions regarding the new release.

Mp3tag v2.33 - Support for Unicode

Probably the most import change and new feature in this version is the support for Unicode through the whole program. Mp3tag writes ID3v2 tags in Unicode format (UTF-16) by default now, but there are still some programs, portables or car radios which have problems with ID3v2 tags stored in Unicode format (though this is specified within the ID3v2 standard). To use Mp3tag with this programs or devices, simply enable the option **Always write ISO-8859-1 tags instead of UTF-16** at **Options > Tags > Mpeg**. After that, just save the tags again by clicking  or by using the Ctrl + S hotkey.
If you have problems displaying tags with characters from other charsets (like Cyrillic) in Mp3tag, the tags are probably stored in a system codepage instead of Unicode or ISO-8859-1. To convert these tags from a specific codepage to Unicode,

simply create a new action Convert codepage, select the appropriate tag field(s) and choose the codepage from the drop-down list.

Please visit the Mp3tag forums if you have other questions regarding the new release.

Mp3tag v2.27 - New syntax for scripting functions

The last versions had major problems with the backslash and the double quotes characters within scripting functions so we had to reimplement the part which is responsible for evaluating the built in functions.
The names of the functions did not change - you don't have to learn new function names or placeholder names.

What has changed?

- There is no need to put parameters of function in double quotes anymore.
- There is no need to leave the $ character from nested function calls.
- There are (hopefully) no problems with backslashes and double quotes anymore.
- The scripting functions $meta and $join are obsolete now.

Are there any special characters with specific functionality/limitations?

- `[ ... ]` Displays contents of brackets only if at least one of fields referenced inside has been found.
- `'` Outputs raw text without parsing. This will output the contained string and ignore all reserved characters (%,$,[,]) in it.
- `$%[]`You have to put the single quote around these reserved characters if you want to use them unparsed.
- `'`You have to put the single quote before this reserved character if you want to output a singe quote.

Examples for conversion

- `$num("%track%", 2)` is now
  `$num(%track%,2)`
- `file://$replace("%_folderpath%","\","/")` is now
  `file://$replace(%_folderpath%,\,/)`
- `%_tag_read% [%_tag%]` is now
  `%_tag_read% '['%_tag%']'`

Please visit the Mp3tag forums if you have other questions regarding the new release.

Mp3tag usage. Overview of commands. Menus

Menu *File*

Save tag

This function saves the tag entered in the tag panel to the selected files. If you have selected **< keep >** at one of the fields on the tag panel, the value in the tag will not be changed.

You can enable/disable writing of specific tag versions at Options > Tags

Hotkey: Ctrl + S

Remove tag

This function removes the tags specified at Options > Tags from the selected files.

Hotkey: Ctrl + R

Read tag

This function reads the tag from the selected files.

Hotkey: Ctrl + T

Playlist (all files)

This function creates a playlist of all files listed in the file view. The filename of the playlist can be a format string and is defined at Options > Playlist

Note:
If you press the Shift-key when running this command, Mp3tag creates a playlist for every different directory.

Hotkey: Ctrl + P

Playlist (selected files)

This function creates a playlist of all selected files. The filename of the playlist can be a format string and is defined at Options > Playlist

Note:
If you press the Shift-key when running this command, Mp3tag creates a playlist for every different directory.

Export

This function exports file and tag information of the selected files to an export file based on the settings at Options > Export.

Note:
If you press the Shift-key when running this command, Mp3tag creates an export file for each different directory.

Hotkey: Ctrl + E

Change directory...

This function shows a folder tree to change the current working directory of Mp3tag. If you select the **Subfolders** option in this dialog, Mp3tag will also load all supported files from the subfolders of the selected folder.

Hotkey: Ctrl + D

Add directory...

This function shows a folder tree to add a directory to the file view of Mp3tag. If you select the **Subfolders** option in this dialog, Mp3tag will also load all supported files from the subfolders of the selected folder.

Hotkey: Ctrl + H

Load Playlist/Cuesheet...

This function loads the files referenced in the selected playlist or cuesheet to the file view.

Favorite directory

This function changes to current working directory to the specified favorite directory.

Hotkey: Ctrl + F

Save configuration...

This function saves all settings and configuration files to a single zip file for backup purposes.

Exit

Exit from Mp3tag.

Hotkey: Alt + F4

Menu *Edit*

Undo

Undo the last operation.

Hotkey: Ctrl + Z

Copy

Copies text from an edit field to the clipboard. If one or more audio files are selected, Mp3tag copies its tags to the clipboard. Please note that embedded cover art is not copied.

Hotkey: Ctrl + C

Cut

Cuts text from an edit field to the clipboard. If one or more audio files are selected, Mp3tag cuts its tags to the clipboard. Please note that embedded cover art is not copied.

Hotkey: Ctrl + X

Paste

Paste text from the clipboard to the current cursor position.
If one or more audio files are selected, and the clipboard contains some tag information, Mp3tag will paste the tags to the selected files.

Hotkey: Ctrl + V

Select all files

Select all files in the file view.

Hotkey: Ctrl + A

Invert selection

Inverts the current selection in the file view.

Hotkey: Ctrl + Shift + A

Unselect all

Unselects all files in the file view.

Hotkey: Ctrl + U

Select next file

This function selects the next file in the file view. Very helpful to walk through all files without leaving the keyboard.

Hotkey: Ctrl + N

Menu *View*

Refresh 

Rereads the current working directory and reloads the tag information from all files

Hotkey: F5

Sort by

Defines the sort criteria of the file view.

Tag Panel

Shows/hides the tag panel.

Hotkey: Ctrl + Q

Extended Tags... 

This menu item displays the extended tag dialog where you can edit extended and user-defined tag fields as well as editing embedded cover art.

Hotkey: Alt + T

Filter

Shows/hides the filter where you can filter the list of files using a search string or filter expressions.

Hotkey: F3

Customize columns

This function opens the customize columns dialog, which allows to enable and disable columns in the file view and change their arrangement.

Menu Convert

Tag - Filename 

This converter renames the selected files based on the tag and file information. It thereby uses a format string, which consists of predefined placeholders.

To show a preview of the conversion only, press the Shift key when running the converter.

Hotkey: Alt + 1

Filename - Tag

This function extracts tag information from the filename based on a format string.

To show a preview of the conversion only, press the Shift key when running the converter.

Hotkey: Alt + 2

Filename - Filename

This function renames the selected files based on the filenames of the files. You can define parts of the filenames and Mp3tag will rearrange them according to the format string.

To show a preview of the conversion only, press the Shift key when running the converter.

Hotkey: Alt + 3

Text file - Tag (Import)

This converter reads tag information from the text file and saves it to the selected files. The format of one line in the text file is defined by a format string.

Hotkey: Alt + 4

Tag - Tag

This converter formats tag fields by other fields content, i.e., this converter can also be used for copying the content of one tag field to another tag field.
You can use placeholders for any tag field and scripting functions as format string to format the content of the field specified.

Hotkey: Alt + 5

Menu Actions

Actions

This function runs user-defined action groups, which consist of one or more actions (like replace characters, case conversion, ...).

Furthermore, all user-defined action groups are listed for direct access in this menu.

Hotkey: Alt + 6

Menu Tag Sources

freedb ...

This function queries album information from the internet based on the Audio-CD in the first CD-ROM drive, based on the selected files, by specifying the freedb-ID or by a free text search and saves the information to the selected files.

Hotkey: Ctrl + I

freedb (local) ...

This function queries album from a local freedb database based on the Audio-CD in the first CD-ROM drive, or by specifying the freedb-ID and stores the information in the selected files.

Hotkey: Ctrl + L

Discogs, MusicBrainz, ...

Mp3tag has a very powerful WEB sources framework which allows you to get album information from various Websites. You can get the latest WEB sources from the WEB Sources Archive. To install a WEB source, just copy the file to your Mp3tag data folder (usually %appdata%\mp3tag\data\sources).

Configuration

You can find a description of the various settings at Options > Tag Sources.

Menu *Tools*

Restore all input fields

Clears all input fields.

Hotkey: Ctrl + B

Auto-numbering wizard

The Auto-numbering wizard adds track- and discnumbers to your files.

Hotkey: Ctrl + K

Options ...

Displays the option dialog.

Hotkey: Ctrl + O

Mp3tag usage. Working with tags and file names

Convert tags from/to filenames

- Tag - Filename
- Filename - Tag
- Filename - Filename
- Text file - Tag
- Tag - Tag

Mp3tag has a variety of possibilities to convert tags and file names.
Conversion means for example getting tag information from parts of the filename or renaming files based on the information stored in the tag of an audio file.

The base concept behind all converters in Mp3tag is the so called format string, which describes an abstract pattern of a filename or a line in a tag-list file.
A format string consists of any characters and predefined placeholders or scripting functions. Every placeholder begins with the percent sign (%), followed by a tag field name and ends with the percent sign.
For example, the placeholder '%album%' stands for the field *Album* in the tag at all converters.

This document gives a short introduction to the converters in Mp3tag. Please also try them, Mp3tag has an unlimited undo feature in case something is not like you would have expected it :-)

Tag - Filename 

Menu Convert > Tag - Filename

You can rename files based on the tag and file information with this converter.
The format string defines the format scheme of the new filename and all placeholders in this format string will be replaced with the information from the file when executing this converter.

Format string

The format string describes the filename scheme for the rename operation based on the tag and file information.

You can use the following placeholders:

%album%	Album
%artist%	Artist
%comment%	Comment
%genre%	Genre
%title%	Title
%track%	Track-Number
%_total%	Total number of tracks from xx/xx track-number field
%year%	Year
%fieldname%	Any other tag field not listed here with the name 'fieldname'. Please have a look at 'View > Extended Tags...' for an overview of all available tag fields in your file.

There are also some special fields, which give you some technical information about the file:

Technical Info

%_bitrate%	Bitrate in kbit/s
%_codec%	Codec
%_cover_mimetype%	Mimetype of first cover art in the tag of the file
%_cover_size%	Size of first cover art in the tag of the file in bytes
%_cover_type%	Cover type of first cover art in the tag of the file
%_covers%	Count of cover art in the tag of the file
%_id3v2_character_encoding%	Encoding of the ID3v2 text frames of the file.
%_length%	Length (formatted)
%_length_seconds%	Length (in seconds)
%_mode%	Mode
%_samplerate%	Sample rate
%_tag%	Available tag types in file
%_tag_read%	Displayed tag type
%_tag_size%	Size of all tags in bytes
%_tag_size_appended%	Size of all appended tags in bytes
%_tag_size_prepended%	Size of all prepended tags in bytes
%_tool%	Additional codec information (not available for all supported audio formats).
%_vbr%	Bitrate type or compression profile

File name / path info

%_directory%	Name of the parent directory
%_extension%	File extension
%_filename%	File name without extension
%_filename_ext%	File name with extension
%_folderpath%	Path without file name
%_parent_directory%	Name of the grandparent folder
%_path%	File name with path
%_volume%	Volume label
%_workingdir%	Name of current working directory
%_workingpath%	Path of current working directory

File properties

%_file_create_date%	Short creation date
%_file_create_datetime%	Long creation date
%_file_create_datetime_raw%	Long creation date (unix timestamp)
%_file_mod_date%	Short modification date
%_file_mod_datetime%	Long modification date
%_file_mod_datetime_raw%	Long modification date (unix timestamp)
%_file_size%	File size in human readable format
%_file_size_bytes%	File size in bytes
%_file_size_kb%	File size in kilobytes
%_file_size_mb%	File size in megabytes

System

%_app%	Mp3tag, e.g. Mp3tag v2.96
%_counter%	Counter (starting with 1)
%_date%	Short date
%_datetime%	Long date
%_total_files%	Number of selected files

Examples

%artist% - %album% - $num(%track%,2) - %title% renames to **Artist - Album - 01 - Title**
%artist%\%album% '['%year%']'\$num(%track%,2). %title% creates **Artist\Album [2010]\01. Title**

Notes

You can use the backslash to mark directories in the format string.
You can use the built-in Scripting functions.

Filename - Tag

Menu Convert > Filename - Tag

This converter takes parts from the filename and copies them to specified fields in the tag of the audio file. The format string defines the format of the filename and the placeholders are describing the way to which tag field the parts will be copied.

Format string

The format string describes the formatting scheme of the filename. The placeholders used mark the parts of the filename which are copied to the tag.
You can use the following placeholders in the format string:

`%album%`	Album
`%artist%`	Artist
`%comment%`	Comment
`%genre%`	Genre
`%title%`	Title
`%track%`	Track-Number
`%year%`	Year
`%dummy%`	this part will be ignored
`%fieldname%`	any field with name fieldname

Examples

Filename: (Artist - Album) 01 EXAMPLE Year.mp3
Format string: `(%artist% - %album%) %track% EXAMPLE %year%`

Notes

You can use the backslash to mark directories in the format string.

Filename - Filename

Menu Convert > Filename - Filename

This converter is a rename tool, which renames filenames based on the filename. You can define up to nine parts of the old filename and let Mp3tag organize the parts in a new way.

Old filename pattern

This format string splits the filename in several parts. You can use the following placeholders:

```
%1    first part
%2    second part
...
%9    last part
```

New filename pattern

Format string which is a combination of the parts defined in the format string above.

Examples

Old filename: (Artist - Album) 01 EXAMPLE 2004 (COMMENT).mp3
Old filename pattern: `(%1 - %2) %3 %4 %5 (%6)`
New filename pattern: `%5 (%1 - %2) Example '['%6']'`
New filename: 2004 (Artist - Album) Example [COMMENT].mp3

Notes

You can mark directories via the backslash within the format string. Please note that Mp3tag won't rename directories but will create new directories below the current working directory if you use backslashes within the new filename pattern.

You can use the built-in Scripting functions for the new filename (e.g. `$upper(%1)`).

Text file - Tag 

Menu Convert > Text file - Tag

This converter imports tag information from a text file. In this text file, each line corresponds to the tag information for one file. The format string describes the format of these lines.

Text files in Unicode format need a BOM (Byte Order Marker) to be correctly identified as Unicode files.

Filename

Filename of the text file. The tags will be imported from this file.

Format string

The format string describes the format of one line in the text file.
You can use the following placeholders in the format string:

`%album%`	Album
`%artist%`	Artist
`%comment%`	Comment
`%genre%`	Genre
`%title%`	Title
`%track%`	Track-Number
`%year%`	Year
`%field%`	Any other tag field from the list of supported tag fields
`%_filename_ext%`	Filename (current working directory is used as file path)
`%_path%`	Filename with full file path
`%dummy%`	this part will be ignored

Matching files from text file to file list in Mp3tag

`%_filename_ext%` or `%_path%` are used to ensure the correct file gets tagged.
Use `%_filename_ext%` if you have lots of files in one folder. Use `%_path%` if you have many folders each containing files. If you do not use either of these tags, mp3tag will simply work through the text file one line at a time, applying changes to each selected file in turn. It is possible that tags will be applied to the wrong files if the selected files and the text file are in a different order.

Example

Line in file: Artist / Title / Album / Year
Format string: `%artist% / %title% / %album% / %year%`

This example uses `space-slash-space` as separator. Depending on your import file, you'd use the corresponding character sequence, e.g., `space-dash-space -` or a semicolon `;`.

Tag - Tag

Menu Convert > Tag - Tag

This converter formats tag fields by other fields content, i.e., this converter can also be used for copying the content of one tag field to another tag field.
You can use placeholders for any tag field and scripting functions as format string to format the content of the field specified.

Mp3tag usage. More Features

Export tag and file information (html, csv, …)

Menu File > Export

With **Mp3tag** you can export file and tag information of your audio collection to files with user-defined file formats. You can either use one of the predefined export configurations to export file information to Excel via the CSV export configuration, let Mp3tag create a nicely looking HTML playlist with one of the html export configurations or you create an export configuration by yourself where you can freely decided on the file format and the layout.
An export configuration consists of a unique name, a filename for the exported file and the definition of the file format. The file format is defined by you, so it's possible to create, e.g., HTML, XML, CSV, batch file or LaTeX output.

Mp3tag features a simple template language which can be used to loop over artists, albums and arbitrary other tag fields. The tag and file informations are represented by placeholders, which are replaced when doing the export. You can find an overview of all valid placeholders and an example at *Configuration, Export*.

Scripting functions

Mp3tag features a set of built-in scripting functions which can be used at various places in the program. These functions offer advanced display and converting options and giving you the power to get the most out of Mp3tag.

I got the idea for implementing the scripting functions from my favourite audio player, foobar2000. I first used a different syntax, but now the syntax of both programs should be (almost) compatible.

The scripting functions are available in these parts of Mp3tag:

- File view
- Export
- Converter Tag - Filename
- Converter Filename - Filename
- Action, Format value
- Action, Replace with regular expressions
- Playlist

Characters with special functionality

`[...]` The contents of brackets are displayed only if at least one of the placeholders used inside the brackets has been found.

`'` Outputs raw text without parsing. This will output the contained string and ignore all reserved characters. If you want to output this character, please use `''`.

`[]$%` You have to put a single quote around these reserved characters if you want to use them unparsed.

`,()` These characters must only be escaped when they are inside a scripting function.

The following scripting functions are available:

- General
- String
- Boolean
- Arithmetic
- Variable operations
- Metadata

General functions

Note: Functions with variable parameter count (e.g., Replace) are limited to < 64 parameters.

Replace

Command: **$replace(string,from,to)** or **$replace(string,from1,to1,from2,to2,...)**
Example: `$replace(%artist% - %album% - %track% - %title%,_,-)`
This example replaces all underscores with dashes. You can provide additional pairs of from/to as parameters.

Remove invalid characters from file name/file paths

Command: **$validate(filename,to)**
Example: `$validate(%artist% - %album% - %track% - %title%,-)`
This example replaces all invalid characters (/?*"<>|:) with dashes.

Case conversion *Normal*

Command: **$caps(string,...)**
Example: `$caps(%artist% - %album% - %track% - %title%)`
This example converts the given string to normal case. The second parameter is optional and specifies additional characters that trigger upper case.
artist - ALBUM - 01 - tItle converts to ***Artist - Album - 01 - Title***

Case conversion *Normal* (without changing uppercase characters)

Command: **$caps2(string)**
Example: `$caps2(%artist% - %album% - %track% - %title%)`
This example converts the given string to normal case without changing uppercase characters. The second parameter is optional and specifies the characters that trigger upper case.
artist - ALBUM - 01 - tItle converts to ***Artist - ALBUM - 01 - TItle***

Case conversion *Sentence*

Command: **$caps3(string)**
Example: `$caps3(%artist% - %album% - %track% - %title%)`
This example converts the given string to first-letter upper case. The second parameter is optional and specifies additional characters that trigger upper case.
artist - ALBUM - 01 - tItle converts to ***Artist - album - 01 - title***

Case conversion *UPPER*

Command: **$upper(string)**
Example: `$upper(%artist% - %album% - %track% - %title%)`
This example converts the given string to upper case.
artist - ALBUM - 01 - tItle converts to ***ARTIST - ALBUM - 01 - TITLE***

Case conversion *lower*

Command: **$lower(string)**
Example: `$lower(%artist% - %album% - %track% - %title%)`
This example converts the given string to lower case.
artist - ALBUM - 01 - tItle converts to ***artist - album - 01 - title***

Pad decimal number with leading zeros

Command: **$num(x,y)**
Example: `$num(%track%,3)`
This example returns the track number with three digits.

Other functions

String functions

$ansi(x)	returns the string x converted to the system codepage. This is useful for renaming files to ensure compatibility with software which doesn't support Unicode.
$char(x)	returns the Unicode character for decimal code point x.
$cutLeft(x,n)	removes the first n characters of string x and returns the result.
$cutRight(x,n)	removes the last n characters of string x and returns the result.
$distance(x,y)	computes the edit distance between strings x and y.
$fmtNum(x)	formats number x with separator for thousands according to current locale settings.
$folderdepth(x)	returns the number of folders in path x.
$left(x,n)	returns the leftmost n characters of string x.
$len(x)	returns the length of string x.
$mid(x,i,n)	returns the first n characters of text x, starting at character i. If n is ommited, the character at position i is returned.
$ord(x)	returns the decimal Unicode code point of the first character of string x. Currently supports characters up to U+FFFF.
$repeat(x,n)	returns the string x n times.
$regexp(x,expr,repl)	replaces the pattern specified by the regular expression expr in the string x by repl. The fourth optional parameter enables ignore case (1) or disables the ignore case setting (0). Please note that you have to escape comma and other special characters in expr.
$reverse(x)	reverses the order of the characters in string x.
$right(x,n)	returns the rightmost n characters of text x.
$strcmp(x,y)	compares strings x and y case-sensitively.
$stricmp(x,y)	compares strings x and y case-insensitively.
$strchr(x,y)	finds the first occurrence of character y in string x.
$strrchr(x,y)	finds the last occurrence of character y in string x.
$strstr(x,y)	finds the first occurrence of string y in string x.
$trim(x)	trims all leading and trailing whitespace from the string x. The optional second parameter specifies the character to trim.
$trimLeft(x)	trims all leading whitespace from the string x. The optional second parameter specifies the character to trim.
$trimRight(x)	trims all trailing whitespace from the string x. The optional second parameter specifies the character to trim.

Boolean functions	
$and(x,y)	returns true if x and y are true.
$geql(x,y)	returns true if x is greater than or equal to y.
$eql(x,y)	returns true if x equals y. If x and y are strings, they're compare case-insensitively.
$grtr(x,y)	returns true if x is greater than y.
$if(x,y,z)	if x is true, y is returned, otherwise z.
$if2(x,y)	if x is true, x is returned, otherwise y.
$ifgreater(a,b,x,y)	if number a is greater than number b, x is returned, otherwise y.
$iflonger(a,b,x,y)	if string a is longer than number b, x is returned, otherwise y.
$isdigit(x)	returns true if character x is a decimal number.
$leql(x,y)	returns true if x is less than or equal to y.
$less(x,y)	returns true if x is less than y.
$neql(x,y)	returns true if x not equal to y.
$not(x)	returns true if x is false.
$odd(x)	returns true if x is odd.
$or(x,y)	returns true if x or y (or any additional argument given) is true.

Arithmetic functions (support up to 64 arguments)	
$add(x,y)	adds y to x.
$div(x,y)	divides x by y.
$mod(x,y)	returns the remainder of x divided by y.
$mul(x,y)	multiplies x by y.
$rand()	returns a pseudorandom number in the range 0 to 32767.
$sub(x,y)	subtracts y from x.

Variable operations	
$get(x)	returns the value of the user-defined variable x. Only available at Export.
$put(x,y)	sets the value of the user-defined variable x to y and returns y. Only available at Export.
$puts(x,y)	sets the value of the user-defined variable x to y and returns nothing. Only available at Export.
$getenv(x)	returns the value of the environment variable x. C.f., MP3TAGAPP for the app directory and MP3TAGAPPDATA for the configuration directory paths.

Metadata (including multiple tag fields)	
$list(x,y,z)	returns a list of all read tag fields and their values that stored in the file. The fields are prefixed by `x`, `y` is used as separator between field and value, and `z` is used as suffix after the tag field's value. Multiple values of a field are separated by ", ".
$meta(x)	returns all values of field `x` (e.g. `$meta(artist)`). Multiple values of a field are separated by ", ".
$meta(x,n)	returns the nth value of field `x` (where `n` starts at 0, e.g. `$meta(artist,1)`).
$meta_sep(x,sep)	returns all values of field `x` (e.g. `$meta_sep(artist,; )`). Multiple values of a field are separated by `sep`.

Filtering files and tags (a.k.a. Searching)

Introduction

The *Filter* allows for searching files that meet a certain criteria (e.g., the filename contains specific characters or a tag field has a specific content). By default it is located at the bottom of the file list and can be enabled and disabled via 'View > Filter' or the F3 keyboard shortcut.

To filter the file list simply start typing in the filter field. As a result, all files that contain the specified filter string in their tag or file name are listed in the file view.

Sometimes it is necessary to use more complex criteria to search for files (e.g., all files of a specific genre but not of that artist). To achieve this, Mp3tag has the notion of *filter expressions*. Filter expressions are simple expressions that consist of tag field names, filter keywords and text.

Notes

- If <field> or <string> contains spaces, it should be enclosed in double quotation marks, e.g., album IS "Moon Safari".
- If <field> in HAS/IS/GREATER/LESS/EQUAL/MATCHES contains $ or % it should be enclosed in double quotation marks and will be treated as a format string instead of a tag field name. This will decrease search speed on large libraries since the format strings need to be evaluated before filtering.

- It is not possible to filter by strings containing double quotations marks. This is a limitation of the feature.
- All operations are non-case sensitive. All filter keywords must be uppercase.

Examples of filter expressions

artist IS Air AND album IS Moon Safari

Artist is "Air" and album is "Moon Safari"

genre IS Rock AND artist HAS Stone

Genre is "Rock" and artist contains "Stone"

NOT genre IS Rock

Genre is not "Rock"

%_covers% IS ""

All files that do not contain cover art.

Description of the Filter Expression Language

<string>

Returns only files that contain every word in the string as a word or substring thereof. A word is a contiguous sequence of characters excluding the space character.

<field> HAS <string>

Returns only files that have all words from <string> in their tag field named <field>. You can use * to address all fields.

<field> IS <string>

Returns only files that where the content of at least one tag field named <field> is equal to <string>.

<expr1> AND <expr2>
<expr1> OR <expr2>

Combines the filter expressions <expr1> and <expr2> by logical *and* or *or*. Parentheses can be used to group combined expressions.

NOT <expr>

Inverts the result of <expr>.

<field> GREATER <number>
<field> LESS <number>
<field> EQUAL <number>

Returns all files where the content of tag field named <field> is greater, equal, or less that the integer value specified by <number>.

<field> ABSENT
<field> MISSING

Returns all files where tag field named <field> is absent.

<field> PRESENT

Returns all files where tag field named <field> is present.

<field> MATCHES <regexp>

Returns only files where the content of the tag field named <field> matches against the regular expression<regexp>. You can use * to address all fields.

More examples ...

%_tag% IS ""

All files without tags

%_tag% HAS "flac id3v2"

Files with flac and ID3v2 tags

%_extension% IS mp4

Files with mp4 extension

%_bitrate% GREATER 180

All files with a bitrate over 180

%_cover_size% GREATER 20480

All files with embedded covers over 20 kbyte

"$len(%bpm%)" GREATER 4

Check if a field is longer than 4 characters

"$if($eql(%artist%,%albumartist%),yes,no)" IS yes

Check if two fields have the same value

"$ifgreater($strstr(%_filename%,%title%),0,yes,no)" IS yes

Check if the title is part of the filename

track MATCHES \d\/\d

Track numbers that have number/number syntax

%_filename% MATCHES ^\d\d

Filenames that start with two digits

If you also use foobar2000 the filter language should look very familiar (because this is where we got the idea from).

Mp3tag usage. Importing tags from online Tag Sources (freedb, Discogs, …)

Menu Tag Sources

freedb ...

This function queries album information from the internet based on the Audio-CD in the first CD-ROM drive, based on the selected files, by specifying the freedb-ID or by a free text search and saves the information to the selected files.

Hotkey: Ctrl + I

freedb (local) ...

This function queries album from a local freedb database based on the Audio-CD in the first CD-ROM drive, or by specifying the freedb-ID and stores the information in the selected files.

Hotkey: Ctrl + L

Discogs, MusicBrainz, ...

Mp3tag has a very powerful WEB sources framework which allows you to get album information from various Websites. You can get the latest WEB sources from the WEB Sources Archive. To install a WEB source, just copy the file to your Mp3tag data folder (usually %appdata%\mp3tag\data\sources).

Configuration

You can find a description of the various settings at Options > Tag Sources.

Tag Sources. Confirming query results

Menu Tag Sources

This dialog is used for confirming and adjusting the tag information that is queried through freedb, a local freedb database or a WEB source.

Adjust tag information

You can adjust the information in the 'Tracks' and 'Addional information' lists by clicking on the item twice.

If there is artist and title information in the title field of the tag, you can split this field via the **Utils > Compilation**-button. Within this dialog, you can provide a format string, which describes the format of the title field via placeholders,
where `%title%` gets copied to the **title** field and `%artist%` gets copied to the **artist** field.

Example

Title: **Tom Waits / Underground**
Format string: `%artist% / %title%`

Assign track information to files

To adjust the assignment from tags to files, you can use the 'Move up' and 'Move down' buttons on the right side of the file list.

Automatically sort files to match the order of the tracks

The **Utils** button contains items to automatically suggest a sorting of your files according to the title or the length. If you are not satisfied with the result, just select the menu item again to revert to the initial order.

Select how album art should be imported

If the WEB source provides album art, you can enable saving of album art to the tag and/or the directory of the file via the **Utils** button. The menu items **Extract cover...** allows for saving the cover at a user-defined location.

freedb - different methods for getting album information

Menu Tag Sources > freedb...

With this dialog you can choose between different methods for getting album information from freedb:

1. **determine from inserted Audio-CD**
   You have to insert the Audio-CD in your first CD-ROM drive.
2. **determine from selected files**
   With this method, Mp3tag calculates the freedb-ID from the selected audio files. The files have to be in the same order as on the Audio-CD.
3. **enter**
   With this method, you can enter a freedb-ID by yourself. You can find a freedb-ID by using the search function at freedb.org.
4. **determine via WEB-search**
   You can search for the album by giving some information for artist or album. If you use a local freedb database, you'll have to create a local index at Options > Tag Sources > Local freedb first.

Tag Sources Development. WEB Sources Framework

Learn about the internals of Mp3tag's WEB Sources Framework and how other online sources can be incorporated to be queried by Mp3tag

Mp3tag features a internal WEB Sources Framework which is parametrised through WEB sources description files. Using these description files, you can import tag data from theoretically every WEB site which displays artist/album information via HTML (no JavaScript, ActiveX). You can find many examples at the WEB Sources Archive on the Mp3tag forums.

Definition of the file format used with WEB source description files

[Name]

Name of the WEB source, e.g. Discogs.com

[Based On]

Base URL of the WEB source, e.g. http://www.discogs.com

[IndexUrl]

Search URL (%s will be replaced by the search criteria entered by user), e.g. http://www.discogs.com/artist/%s

[AlbumUrl]

Result base URL (URL result from first search pass will be appended), e.g. http://www.discogs.com

[WordSeparator]

Character/string used instead of blanks within the search criteria entered by the user, e.g. %20

[IndexFormat]

Format string for splitting the output buffer from the first search pass into different fields. %_url% is needed, e.g. %_url%|%album%|%type%|%label%

[SearchBy]

Field(s) which are offered as search criteria by the WEB source, e.g. %artist%

[Encoding]

Encoding used for all urls. Can be utf-8, iso-8859-1, url, url-utf-8 or ansi (system codepage will be used).

[ParserScriptIndex]

This key contains a multi-line parser script (start with ...) which parses the search results page for different albums.

[ParserScriptAlbum]

This key contains a multi-line parser script (start with ...) which parses a WEB page found by the first search pass.

Tag Source Development. Parser Scripts

How to Start

First of all you need to find a way to identify the lines which contain the interesting bits of information, for example the year of a release. The Mp3tag parser sees output as lines and characters and you tell the parser how to move from the start to the places you want to display.
Mp3tag uses a pointer which is positioned at the beginning of the file and which can be moved with several commands. So how do we move this pointer to the year displayed on a Website?
We can either move it down N lines or we can tell it to move down until it finds the text "Year:". To do the first we would either use the command `MoveLine N` (where N is the number of lines) or `GotoLine N` -- this either means go down seven lines from where you are or go to the Nth line from the top of the text. To do the search, the command would be `FindLine "Year:"`, which - well - finds the next line from where you are that contains the text "Year:". In all cases the pointer would be moved to the first character of the target line. From there we could tell the pointer to move N steps to the right (`MoveChar N`) or to move to the Nth character in the line (`GotoChar N`) or to position the pointer after the text "Year:" (`FindInLine "Year:"`). All of these commands would result in the pointer being moved to the first digit of the number of hidden files. Please note the difference between
the `FindLine` and `FindInLine` commands: The earlier goes through lines from where you are to find a text and places the pointer at the beginning of the line, while the latter looks within the line where the pointer is and positions it after the found text.
Now that we are where we want, we need tell Mp3tag to store the data. To do this, a `Say` command is used, but we have to find a way to tell it what to say and what not. In this case we want to output the rest of the line and so we use
the `SayRest` command. An alternative would be the `SayUntil " "` command, which would output everything until a space character is found.

So, a script to display the year would look like this:

```
FindLine "Year:"
FindInLine "Year:"
SayNextNumber
```

You will notice that the example used the Find-Commands rather than the Move- or Goto-Commands. Whenever you have a chance to use a Find command, please do so, because WWW pages tend to be changed. Using a script that relies on Find-Commands is more likely to survive a change in the raw data than one that relies on absolute positions.

Debugging

Doing all this only theoretically can be a bit tricky and if you make an error counting lines or characters you might end up with quite unexpected results. To check what the parser is doing, you can add a `Debug "on" "debug.out"` command to the top of your script. This will give you an output file which will show you step by step what the parser is doing and why you end up with a given output.
Many times you will even want to start your script just with a Debug command, to see what data you actually get for parsing and before you build your script step by step.

List of Parser Commands

Command	Parameter(s)	Description
FindLine	S n	Find line with first or Nth occurrence of S (starting from the current position)
FindLineNoCase	S n	Find line with first or Nth occurrence of S (ignoring case and starting from the current position)
FindInLine	S n n	Find the next/Nth occurrence of S within the current line. If the 3rd parameter is set to 1, no error is produced if S is not found.
GotoChar	N	Skip to the Nth character in the current line
GotoLine	N	Go to Nth line (counting from top)
MoveChar	N	Move right/left N characters
MoveLine	N n	Move down/up N lines (starting from current position). If the second parameter is set to 1, possible errors are ignored.
Say	S	Send S to output
SayUntil	S	Send everything until S to output
SayUntilML	S	Send everything until S to output searching across multiple lines
SayRest		Send everything to the end of the current line to output
SayNChars	N	Send next N characters to output
SayNextNumber		Outputs the next numeric value from the input.
SayNextWord		Outputs the next word from the input.
SayOutput	S	Send the content of output S to the current output. The output CurrentUrl is always generated at runtime.
SayNewline		Outputs a carriage return, line feed (CR LF) sequence.
SayRegexp	S s s	Outputs all matches of the regular expression in the first parameter separated by the string in the second parameter. If the third parameter is provided, the match is only performed till the third parameter content is reached. If the third parameter content cannot be found, the line is ignored.
Set	S s	Sets the content of output S to the value s. Resets the content if s is omitted.

SkipChars	S	Skip any characters contained in S
If	S	Check for occurrence of S on current position.
IfNot	S	Check for absence of S on current position.
IfOutput	S	Check for content in output buffer named S.
IfNotOutput	S	Check for absence of content in output buffer named S.
Else		Else branch of an If operation.
Endif		End of an If/IfNot/IfOutput/IfNotOutput operation.
OutputTo	S	Sets the name of the output buffer of the Say commands to S.

Do ... While	S n	Execute the command surrounded by the two commands while S occurs on current position. The optional second parameter limits the execution of the loop to maximal n times.
		`Do`
		`MoveLine 1`
		`While "<td>"`
		Nesting of Do commands is not allowed.

Replace	S S	Replaces all occurrences of the first parameter by the second parameter.
RegexpReplace	S S	Replaces everything that is matched by the regular expression in the first parameter by the string in the second parameter.
JoinUntil	S	Joins the current line to the next occurrence of S.
JoinLines	N	Joins the current line with the next N lines.
KillTag	S s	Replaces tag S with s in current line (or blank if omitted).
Unspace		Removes leading and trailing spaces from the current line.
Debug	S s n	Debug output, S= "on" or "off", s is an optional file name. n is an optional maximum file size for the debug file in MB.
json	S s	Enables or disables JSON input mode. S is set to either "on" or "off". If in JSON input mode, the input is parsed as JSON data structure and can be accessed using the following JSON-related functions. The optional second parameter can be set to "current" to use the current, possibly transformed input as JSON

		input.
json_foreach	S	If in JSON input mode, starts iteration over an JSON array accessed by S. This scripting function emits the size of the accessed array on the input which can be then used inside the WEB source script. Iteration must be ended by json_foreach_end.
json_foreach_end		If in JSON input mode, ends iteration of the last iteration started by json_foreach.
json_select_object	S	If in JSON input mode, selects JSON object denoted by S and makes its fields available to json_select. If successful, the object name S is available at the current position.
json_unselect_object	S	If in JSON input mode, leaves current JSON object and returns to the previous one.
json_select	S	If in JSON input mode, selects JSON element denoted by S and emits its content to the input.
json_select_array	Sns	If in JSON input mode, selects the nth element of the JSON array denoted by S to the input. If n is -1, all elements are emitted delimited by s.
json_select_many	SSs	If in JSON input mode, selects all objects from the JSON array denoted by the first parameter and emits their corresponding elements denoted by the second parameter to the input, delimited by s.

N Required numeric parameter
S Required string parameter (in quotes)
n Optional numeric parameter
s Optional string parameter (in quotes)

Mp3tag usage. Customizing Mp3tag

Customizing the Tag Panel. Configuration > Tag Panel

Missing a specific tag field? Simply add it via Options > Tag Panel.

User-defined tag input fields

The Tag Panel is the area in Mp3tag's main window that shows the fields to enter metadata. Besides the standard fields, you can add user-defined tag fields to the Tag Panel via this configuration page.

If you're using the ampersand & character in the name of a tag field, the following character will be available keyboard shortcut in combination with the Alt key.

By clicking the checkbox that prepends each tag input field in the list, you can quickly enable and disable fields from the Tag Panel. This option is also available via the right-click menu of the Tag Panel.

The special field %_DIRECTORY% is used to add a directory picker to the Tag Panel that also shows a history of the last used folders.

The default value is used for the field in the Tag Panel when **multiple files** are selected. Please note that you have to save the tag in order to store this values in the tag.

Layout of the Tag Panel

Tag input fields can have four different sizes that steer the layout of the Tag Panel (Large, Medium, Small, Smaller and Multiline). Depending on the size, more fields are arranged in one line or span multiple lines. If the size of a tag input field is configured as Multiline, you can specify the height via the number of rows. One row is thereby the sum of the height of a label and the height of a normal tag input field.

You can drag the Tag Panel by its gripper on the top border and arrange it at the top, right, left and bottom of the main window. If it's docked as horizontal panel (top or bottom), the fields flow naturally to fill up the available space from top to bottom to the right. If you're using many user-defined fields and prefer a more narrow layout (but a vertical scroll-bar), you can use the special field _SEPARATOR to switch to manual column breaks.

Customizing the file list columns. Customize columns

Missing a specific column? Here's how you can add arbitray content to the file list.

Menu View > Columns

With this dialog you can add, remove and rearrange the columns of the file view.

Value

This field is required and is used for displaying content in the file view. You can use any placeholder from the list below.

Field

This field is optional. The placeholder of this field is used for editing the content of the field through the file view - if this field is empty, the column is write protected.
You can only use placeholders for tag fields (e.g. %artist%), %_filename_ext%, or %_filename% here.

Sort by

This field is optional. If this field is empty, Mp3tag uses the contents of **Value** for sorting. Otherwise the evaluated content of this field is used.

Numeric

This checkbox affects the sorting of the column and determines if the column contents are aligned left or right.

The following placeholders are available:

Metadata

%artist%	Artist
%album%	Album
%comment%	Comment
%genre%	Genre
%title%	Title
%track%	Track-Number
%year%	Year
%fieldname%	The content of any field name

Technical info

%_bitrate%	Bitrate in kbit/s
%_codec%	Codec
%_cover_mimetype%	Mimetype of first cover art in the tag of the file
%_cover_size%	Size of first cover art in the tag of the file in bytes
%_cover_type%	Cover type of first cover art in the tag of the file
%_covers%	Count of cover art in the tag of the file
%_id3v2_character_encoding%	Encoding of the ID3v2 text frames of the file.
%_length%	Length (formatted)
%_length_seconds%	Length (in seconds)
%_mode%	Mode
%_samplerate%	Sample rate
%_tag%	Available tag types in file
%_tag_read%	Displayed tag type
%_tag_size%	Size of all tags in bytes
%_tag_size_appended%	Size of all appended tags in bytes
%_tag_size_prepended%	Size of all prepended tags in bytes
%_tool%	Additional codec information (not available for all supported audio formats).

%_vbr%	Bitrate type or compression profile

File name / path info

%_directory%	Name of the parent directory
%_extension%	File extension
%_filename%	File name without extension
%_filename_ext%	File name with extension
%_folderpath%	Path without file name
%_path%	File name with path
%_parent_directory%	Name of the grandparent folder

File properties

%_file_create_date%	Short creation date
%_file_create_datetime%	Long creation date
%_file_create_datetime_raw%	Long creation date (unformatted)
%_file_mod_date%	Short modification date
%_file_mod_datetime%	Long modification date
%_file_mod_datetime_raw%	Long modification date (unformatted)
%_file_size%	File size in human readable format
%_file_size_bytes%	File size in bytes
%_file_size_kb%	File size in kilobytes
%_file_size_mb%	File size in megabytes

Names and mapping of ID3v2/WMA/MP4 tag fields. Tag field mappings

Tag fields in Mp3tag

The table below gives an overview of all ID3v2 / MP4 / WMA tag fields available with Mp3tag, their names in Mp3tag and their use in other programs.
You can edit these tag fields through the extended tag dialog, add custom fields to the Tag Panel via 'Options > Tag Panel' or customize the file list columns using these field names.

Please note that fields of VorbisComments (used with FLAC, OGG, SPX) and of APEv2 tags are not mapped internally by Mp3tag but are displayed with their actual name unless they have a user-defned mapping.

In case you need to change one of the mappings (e.g., for using consistent field names across formats), you can create user-defined field mappings.

ID3v2, MP4, Matroska and WMA tag fields

Name in Mp3tag	ID3v2.3	ID3v2.4	MP4	Matroska	WMA	iTunes	WMP	Winamp
ALBUM	TALB	"	©alb	T=50 TITLE	WM/AlbumTitle	Album	Album	Album
ALBUMSORT	TSOA	"	soal	T=50 SORT_WITH	WM/AlbumSortOrder	Album Sort Order		
ALBUMARTIST	TPE2	"	aART	T=50 ARTIST	WM/AlbumArtist	Album Artist	Album Artist	Album Artist
ALBUMARTISTSORT	TSO2	"	soaa	T=30		Album Artist Sort Order		
ARTIST	TPE1	"	©art	T=30 ARTIST	Author	Artist	Artist	Artist
ARTISTSORT	TSOP	"	soar	T=30	WM/ArtistSortOrder	Artist Sort Order		
BPM	TBPM	"	tmpo	T=30	WM/BeatsPerMinute	BPM	Beats per minute	
COMMENT	COMM	"	©cmt	T=30	Description	Comments	Comments	Comment
COMPILATION	TCMP	"	cpil	T=30		Part of a compilation		
COMPOSER	TCOM	"	©wrt	T=30	WM/Composer	Composer	Composer	Composer
COMPOSERSORT	TSOC	"	soco	T=30		Composer Sort Order		
CONDUCTOR	TPE3	"	©con	T=30	WM/Conductor		Conductor	
CONTENTGROUP	TIT1	"	©grp	T=30	WM/ContentGroupDescription	Grouping for MP4 / Work name for MP3	Music category description	
COPYRIGHT	TCOP	"	cprt	T=30	Copyright			Copyright
DESCRIPTION			desc					
DISCNUMBER	TPOS	"	disk	T=50 PART_NUMBER	WM/PartOfSet	Disc Number	Set	
ENCODEDBY	TENC	"		T=30 ENCODED_BY	WM/EncodedBy			Encoded by
ENCODERSETTINGS	TSSE	"		T=30	WM/EncodingSettings			
ENCODINGTIME		TDEN		T=30	WM/EncodingTime			
FILEOWNER	TOWN	"		T=30 PURCHASE_OWNER				
FILETYPE	TFLT	"		T=30				
GENRE	TCON	"	©gen | gnre	T=30	WM/Genre	Genre	Genre	Genre
GROUPING	GRP1	"		T=30		Grouping		

INITIALKEY	TKEY	"		T=30 INITIAL_KEY	WM/InitialKey		Key	
INVOLVEDPEOPLE	IPLS	TIPL		T=30				
ISRC	TSRC	"		T=30	WM/ISRC			
LANGUAGE	TLAN	"		T=30	WM/Language		Language	
LENGTH	TLEN	"		T=30				
LYRICIST	TEXT	"		T=30			Lyricist	
MEDIATYPE	TMED	"		T=50 ORIGINAL_MEDIA_TYPE				
MIXARTIST	TPE4	"		T=30 REMIXED_BY				
MOOD		TMOO		T=30	WM/Mood			
MOVEMENTNAME	MVNM	"	©mvn	T=20 TITLE		Movement Name		
MOVEMENT	MVIN	"	©mvi	T=20 PART_NUMBER		Movement Index		
MOVEMENTTOTAL	MVIN	"	©mvc	T=30		Movement Total		
MUSICIANCREDITS		TMCL		T=30				
NETRADIOOWNER	TRSO	"		T=30				
NETRADIOSTATION	TRSN	"		T=30				
ORIGALBUM	TOAL	"		T=30	WM/OriginalAlbumTitle			
ORIGARTIST	TOPE	"		T=30	WM/OriginalArtist		Original Artist	Orig. Artist
ORIGFILENAME	TOFN	"		T=30	WM/OriginalFilename			
ORIGLYRICIST	TOLY	"		T=30	WM/OriginalLyricist		Original lyricist	
ORIGYEAR	TORY	TDOR		T=30	WM/OriginalReleaseYear			
PODCAST	PCST	"	pcst	T=30				
PODCASTCATEGORY	TCAT	"	catg	T=30				
PODCASTDESC	TDES	"	ldes	T=30				
PODCASTID	TGID	"	egid	T=30				
PODCASTKEYWORDS	TKWD	"	keyw	T=30				
PODCASTURL	WFED	"	purl	T=30				
POPULARIMETER	POPM	"		T=30			Rating	
PUBLISHER	TPUB	"		T=30	WM/Publisher			
RATING MM	POPM	"		T=30			Rating	
RATING WMP	POPM	"		T=30	WM/SharedUserRating		Rating	

RELEASETIME	TDRL	"		T=50 DATE_RELEASED				
SETSUBTITLE		TSST		T=50 SUBTITLE				
SUBTITLE	TIT3	"		T=30	WM/SubTitle		Subtitle	
TAGGINGTIME		TDTG		T=30 DATE_TAGGED				
TITLE	TIT2	"	©nam	T=30	Title	Name	Title	Title
TITLESORT	TSOT	"	sonm	T=30 SORT_WITH	WM/TitleSortOrder	Title Sort Order		
TRACK	TRCK	"	trkn	T=30 PART_NUMBER	WM/TrackNumber	Track Number	Track number	Track #
UNSYNCEDLYRICS	USLT	"	©lyr	T=30 LYRICS	WM/Lyrics	Lyrics	Lyrics	
WWW	WXXX	"		T=30			Other WEB sites	URL
WWWARTIST	WOAR	"		T=30	WM/AuthorURL		Official artist WEB site	
WWWAUDIOFILE	WOAF	"		T=30	WM/AudioFileURL		Official audio file information	
WWWAUDIOSOURCE	WOAS	"		T=30	WM/AudioSourceURL		Official audio source	
WWWCOMMERCIALINFO	WCOM	"		T=30 PURCHASE_INFO	WM/PromotionURL			
WWWCOPYRIGHT	WCOP	"		T=30	CopyrightURL			
WWWPAYMENT	WPAY	"		T=30 PURCHASE_ITEM				
WWWPUBLISHER	WPUB	"		T=30				
WWWRADIOPAGE	WORS	"		T=30				
YEAR	TYER	TDRC	©day	T=50 DATE_RECORDED	WM/Year	Year		Year
Other fields	TXXX	"	*field name*	T=30 *field name*	*field name*			

Tag fields exclusive to MP4

Name in Mp3tag	MP4	iTunes
ITUNESACCOUNT	apID	
ITUNESADVISORY	rtng	
ITUNESCATALOGID	cnID	iTunes Store Catalog ID
ITUNESGAPLESS	pgap	Gapless album
ITUNESHDVIDEO	hdvd	Boolean flag for HD video (0/1)
ITUNESMEDIATYPE	stik	Media type
ITUNESOWNER	ownr	Owner
ITUNESPURCHASEDATE	purd	Purchase Date
SHOWMOVEMENT	shwm	Show Movement
TVEPISODE	tves	TV Episode
TVEPISODEID	tven	TV Episode ID
TVNETWORK	tvnn	
TVSEASON	tvsn	TV Season
TVSHOW	tvsh	Show
TVSHOWSORT	sosn	Sort Show
WORK	©wrk	Work Name

Annotations

COMPILATION

Syntax: Either enter the value 1 or delete the field
Note: Unofficial field that is only used by iTunes/iPod to mark albums as sampler.

INVOLVEDPEOPLE, MUSICIANCREDITS

Syntax: Role1:Person1;Role2:Person2; ...
Example: Singer:Robbie Williams;

ITUNESADVISORY

Syntax: 0, 1 or 2
Note: Content rating, use 0 = None, 1 = Explicit, 2 = Clean

ITUNESGAPLESS

Syntax: Set value to 1 or delete the field.
Note: Marks song as part of a gapless Album.

ITUNESMEDIATYPE

Syntax: Enter the media type
Possible values: Normal, Audiobook, Music Video, Movie, TV Show, Booklet

PODCAST

Syntax: Either enter the value 1 or delete the field
Note: Unofficial field that is only used by iTunes/iPod to mark tracks as podcasts.

POPULARIMETER

This description is for only ID3v2 tags. The playcounter is an optional value.
Syntax: Email|Rating|Playcounter
Example: Mp3tag|255|5
Note: The Rating is an integer between 0 (worst) and 255 (best).

RATING MM

Track rating from 1 = Bad to 5 = Very good like MediaMonkey does it. You can also enter stars * (and - for half stars) for rating the track.

RATING WMP

Track rating from 1 = Bad to 5 = Very good like Windows Media Player does it. You can also enter stars * for rating the track.

RATING WINAMP

Track rating from 1 = Bad to 5 = Very good like Winamp does it. You can also enter stars * for rating the track.

UNSYNCEDLYRICS

Syntax: xxx||Lyrics
Note: xxx = Language of the lyrics, abbreviated by 3 characters according to ISO-639-2.

Mp3tag usage. Configurations

General

Auto-completion at selection-boxes

This option enables/disables auto-completion at input fields.

Don't sort case sensitive

Mp3tag handles uppercase characters the same way as lowercase characters at sorting, when this option is enabled.

Automatically select next file when editing in file view

When you're editing a file directly in the file view using a delayed double click and have this option enabled, Mp3tag will jump to the next file after you've finished editing one file.

Display grid lines in file view

This option is a tri-state option. If it's enabled, the lines in the file view are displayed in alternate colors. If it's enabled but grayed out, the file view is drawn with real grid lines.

Use natural sorting

This option enables natural sorting (1, 2, 11, ...) within Mp3tag.

Select files automatically

This option enables the automatic selection of files that are added to Mp3tag's file view.

Messages

Options to enable/disable certain message dialogues that are displayed after operations in Mp3tag.

Directories

Subfolders

This option enables reading of subdirectories.

Favorite directory

You can enter a favorite directory here and quickly jump to this directory with the **File > Favorite directory**command.

Cover art directory

With this option you can specify a default directory that is displayed when adding cover art. Besides fully-qualified path names, you can provide relative path names and use placeholders. If no default directory is specified or the directory cannot be opened, Mp3tag opens the directory of the first selected file iff it contains cover art or the current working directory in all other cases.

Library. Configuration > Library

Enable Library

Normally, Mp3tag reads metadata from files when loading files, directories or playlists and stores the data in memory. Then, when you close Mp3tag, the data once read from the files is discarded and must be read again the next time you're working with those files in Mp3tag.

This can become both time consuming and memory intensive, especially when dealing with large amounts of files. If you enable the Library in Mp3tag, metadata is first read from the files and then stored in a database that resides in Mp3tag's configuration folder. On subsequent reads of the file, the metadata is read directly from the database and updated iff the file's modification timestamp has been changed. In addition to that, cover art or other binary data is stored solely in the database, which can reduce the amount of memory that is consumed by Mp3tag significantly.

Only include these directories

If you only want to keep metadata of certain directories in Mp3tag's Library, you can configure specific directories that are included in Mp3tag's Library. Files that reside outside of these directories are not permanently stored in Mp3tag's Library.

Cleanup Library

Mp3tag keeps track of files if the they're deleted or moved from inside of Mp3tag. If you delete or move files outside of Mp3tag, it's possible that orphaned entries are kept in the database. While this is not a problem per se, you can remove those entries with this option. It checks whether the referenced files are still on your hard drive and removes the entries if it cannot access the file anymore. Just to keep things clean and tidy :)

Tags. Configuration > Tags

Mpeg (MP3) > Mpeg

These options are only affecting audio formats which are using ID3v2 tags (currently MP3 and TTA).

Please have a look at Names and mapping of ID3v2 tag fields for a complete list of ID3v2 tag fields.

Read

The order below, is the order in which Mp3tag reads the different tags (that means that displaying APEv2 tags is prioritized over displaying ID3v1 tags).

ID3v1

if enabled, Mp3tag reads an existing ID3v1 tag.

ID3v2

if enabled, Mp3tag reads an existing ID3v2 tag.

APE

if enabled, Mp3tag reads an existing APEv1 or APEv2 tag.

Write

ID3v1

if enabled, Mp3tag writes an ID3v1 tag to the file.

APEv2

if enabled, Mp3tag writes an APEv2 tag to the file. If the file already has an APEv1 tag, Mp3tag will update this tag to an APEv2 tag. If the file has an ID3v1 tag, Mp3tag will update this tag too.

ID3v2

if enabled, Mp3tag writes an ID3v2 tag to the file.

ID3v2 only if ID3v1 too small

if enabled, Mp3tag writes only an ID3v2 tag, if the information in the tag input fields exceeds the limitations of the ID3v1 tag. If the file already has an ID3v2 tag, Mp3tag will update this tag. If the tag contains user definied genre, also an ID3v2 tag is written.

ID3v2.4 UTF-8

newest revision of the ID3v2 standard. Tags are written in Unicode as UTF-8 encoded string. If you have problems with displaying the tags in other programs, please aks their authors to support ID3v2.4 (or switch to another option).

ID3v2.3 UTF-16

if enabled, Mp3tag writes ID3v2.3 tags in Unicode encoded as UTF-16 LE.

ID3v2.3 ISO-8859-1

if enabled, Mp3tag doesn't use Unicode when writing ID3v2 tags. This is useful if you are using software or devices (like some car radios) which can't read tags stored in Unicode format.

Remove

ID3v1

if enabled, Mp3tag removes an existing ID3v1 tag.

ID3v2

if enabled, Mp3tag removes an existing ID3v2 tag.

APE

if enabled, Mp3tag removes an existing APEv1 or APEv2 tag.

APE, MPC, ... > APE, MPC, ...

Here you can do some settings for audio formats which are using APE-tags by default (like Musepack, Monkey's Audio, WavPack, ...)

Read

ID3v1 if enabled, Mp3tag reads an existing ID3v1 tag.
APE if enabled, Mp3tag reads an existing APEv1 or APEv2 tag.

If both, ID3v1 and APE, are enabled, Mp3tag first reads an ID3v1 tag and after that the APE tag.

Write

ID3v1 if enabled, Mp3tag writes an ID3v1 tag to the file.
APEv2 if enabled, Mp3tag writes an APEv2 tag to the file. If the file already has an APEv1 tag, Mp3tag will update this tag to an APEv2 tag. If the file has an ID3v1 tag, Mp3tag will update this tag too.

Remove

ID3v1 if enabled, Mp3tag removes an existing ID3v1 tag.
ID3v2 if enabled, Mp3tag removes an existing ID3v2 tag.
APE if enabled, Mp3tag removes an existing APEv1 or APEv2 tag.

Mapping > Mapping

User-defined field mappings

Mp3tag displays tag fields either directly with their corresponding name from the files (e.g., for files with APEv2 tags like Musepack or VorbisComments like OGG and FLAC) or uses a fixed mapping from format-specific names to human-readable names (e.g., for ID3v2 identifiers or MP4 atoms). For example, the ID3v2 field TPE1 is mapped to ARTIST in Mp3tag.

You can customize mappings from Mp3tag's internal field name to a user-defined field name in case you need to make field names consistent across different types of tags (e.g., YEAR in ID3v2 and DATE in FLAC) via 'Options > Tags > Mapping'. There you can create mappings for a specific tag type (APEv2, ID3v2, MP4, VorbisComment or WMA) and provide a Source name (Mp3tag's name) and a Target name (your preferred field name).
After reading tags from files, Mp3tag translates all source field names to their respective target (and vice-versa when writing tags).

Imagine you want to use one field name YEAR for the release year regardless of whether you are using MP3s (which already have YEAR as Mp3tag's name for release year) or FLACs (which have DATE). To create this unique field that can be used accross the different formats, you need to create a new mapping for VorbisComment (since FLAC files have VorbisComments) and provide DATE as Source (it is the name displayed in Mp3tag so far) and YEAR as Target (this is the field name you like to use). When you now reload the files, you will notice that instead of DATE the field is now displayed as YEAR and you can use this in format strings and everywhere else without differentiating between tag formats.

»But what happens when I save the tags?« you ask. Mp3tag then applies the mapping in the reverse direction. So any field that uses the name provided as Target gets translated to the corresponding Source field name. It's also possible to map multiple Source fields to one Target (e.g., DATE and RELEASE to YEAR). Please note that this implies that all Source fields are written on tag updates. Mapping one Source field to multiple Target fields is not supported.

Advanced > Advanced

ID3v2 comment language code (ISO-639-2)

Different programs require ID3v2 comments to be written with specific language code identifiers. Mp3tag's default is 'eng', but depending on the language of your Windows installation, you might want to use the ISO-639-2 of this language so that comments are displayed in Windows Explorer or Windows Media Player (e.g., 'deu' for a German Windows installation). However, if you use another language code than 'eng' iTunes won't read the comments of your ID3v2 tagged files anymore, since it requires them to be written with this particular language code identifier.

List chapters as separate files

Some file formats, e.g., MP4 or Matroska, can have chapters. If you enable this setting and reload the directory, Mp3tag lists chapters as separate files. It's especially helpful if you're into editing video files and want to edit the titles of the individual chapters (or the global tags of a file if disabled).

Especially if the option to list chapter as separate files is disabled, it can be difficult to identify which files actually have chapters. You can use the following information fields to identify those files: `%_subsong_count%` provides the number of subsongs in a file and `%_subsong%` contains the subsong index.

Tags. Tags

Save tags when using arrow-keys/single mouse click

This option enables automatic saving of entered data when the file loses the focus.

Preserve timestamps of modified files

Mp3tag will keep the modification date of the files, when writing or removing tags.

Restrict incoming files to

Mp3tag will read all supported files with the extensions given at **Restrict incoming files to**.

Save mode at extended tag dialog

You can specify whether Mp3tag should ask, ignore or always save changes made at the extended tag dialog.

Don't display first image from file directory as cover art

Normally, Mp3tag displays the first image from the files' directory as cover art in the Tag Panel if the file has no embedded cover art. You can restrict cover display to embedded cover art with this option.

Tags. Configuration > Tag Sources

Default file name for cover art

You can specify the default file name for saving cover art at the WEB sources dialog. Placeholders like %artist% can be used for the file name.

freedb Server

Name

The name of the freedb-configuration e.g. *FREEDB in USA*.
You can get the current list of freedb-servers, by pressing the **Current list of freedb-servers** button. This will restore all changes made to the mirror list from the freedb server.

Location

Location of the freedb server e.g. *San Jose (USA)*

Email-address

Your email-address for logging onto the freedb-server. Your email-address will only be used for handshaking with the freedb server!

Address

Domain name of the freedb server, e.g. *freedb.freedb.org*

Path

URL to the freedb server program, e.g. */~cddb/cddb.cgi*.

Port

Port of the freedb server.

freedb General

Copy disc-id to comment field

This option copies the disc-id to the comment field of the tag when tagging from freedb.

Copy disc-id to DISCID field

This option copies the disc-id to the DISCID field of the tag when tagging from freedb.

Local freedb

You can enter the path to your local freedb database here, if you want to use one. Please note, that accessing the database based on the audio files is not available at local databases. You'll need to create a **local index** if you like to use the text search option on local freedb databases. This will take a while...

Tags. Configuration > Tag Panel

User-defined tag input fields

The Tag Panel is the area in Mp3tag's main window that shows the fields to enter metadata. Besides the standard fields, you can add user-defined tag fields to the Tag Panel via this configuration page.

If you're using the ampersand & character in the name of a tag field, the following character will be available keyboard shortcut in combination with the Alt key.

By clicking the checkbox that prepends each tag input field in the list, you can quickly enable and disable fields from the Tag Panel. This option is also available via the right-click menu of the Tag Panel.

The special field `%_DIRECTORY%` is used to add a directory picker to the Tag Panel that also shows a history of the last used folders.

The default value is used for the field in the Tag Panel when **multiple files** are selected. Please note that you have to save the tag in order to store this values in the tag.

Layout of the Tag Panel

Tag input fields can have four different sizes that steer the layout of the Tag Panel (Large, Medium, Small, Smaller and Multiline). Depending on the size, more fields are arranged in one line or span multiple lines. If the size of a tag input field is configured as Multiline, you can specify the height via the number of rows. One row is thereby the sum of the height of a label and the height of a normal tag input field.

You can drag the Tag Panel by its gripper on the top border and arrange it at the top, right, left and bottom of the main window. If it's docked as horizontal panel (top or bottom), the fields flow naturally to fill up the available space from top to bottom to the right. If you're using many user-defined fields and prefer a more narrow layout

(but a vertical scroll-bar), you can use the special field `_SEPARATOR` to switch to manual column breaks.

Tags. Configuration > Genres

User defined Genres

Here you can add some extra user defined genres to the standard list of genres. These genres are not saved to an ID3v1 tag due to the limitations of the tag format.

Show only user defined genres

If this option is enabled, the genre dropdown list at the tag input mask only shows user defined genres after a restart of Mp3tag.

Tags. Configuration > Playlist

Write extended info

If this option is enabled, Mp3tag writes extended information to the playlist file according to the format string.
The audio player doesn't have to read the information from the tag of the audio file and displays the extended information from the playlist.

Entries relative to working directory

Enable this option, if you don't want full file paths in your playlist. This is enabled by default.

Create Playlist automatically

This option enables automatic creation of playlists when saving tags to files.

Filename of playlist

You can specify the filename (or a format string for the filename) of the playlist here. If you use '*.m3u8' as file extension, Mp3tag writes a UTF-8 encoded playlist (Unicode with support for special characters). You can use the following placeholders within the filename as well as the Mp3tag scripting funtions.

`%album%`	*Album*
`%artist%`	*Artist*
`%comment%`	*Comment*
`%year%`	*Year*
`%genre%`	*Genre*
`%_directory%`	*Directory name*
`%_folderpath%`	*Complete path of the folder*

Mp3tag uses the first selected file for replacing the placeholders at runtime.
Examples:

`%artist% - %album% (%year%).m3u` would be `artist - album (year).m3u`

`%_folderpath%%_directory%.m3u` resolves to `folderpath\directory.m3u`

Show confirm filename dialog

These options shows an input dialog for the playlists filename when creating the playlist. You can change the filename of the playlist within this dialog but Mp3tag will not evaluate placeholders entered in this dialog.

Don't create directories

Since you can also use relative and absolute file paths as playlist file names - e.g. %artist%\%album%.m3u - this option prevents Mp3tag from creating any directories when creating playlists.

Tags. Configuration > Actions

Mp3tag provides a variety of actions, which can be applied to filenames and tags. The actions are grouped together into named sets (action groups), which can be applied independently via Actions Alt + 6. If you do not want to create a reusable action group for applying one single action, the Actions (Quick) toolbar button or keyboard shortcut Shift + Alt + 6 can be used.

You can create, edit, duplicate, and delete action groups using the buttons on the right or the context menu. The button **Utils** allows for saving the current selection state to a file using the **Save selection...** menu item. These are then listed in the menu and can be later restored by clicking on the configuration name.
The **Save** menu item saves the current selection state directly (this is only necessary if you choose to not exit the dialog via the **OK** button).

Tags. Configuration > Export

Settings

The list shows the available export configurations. You can activate an export configuration by selecting it. This export configuration will be used when running the export command.

You can create a new export configuration by pressing the **New configuration** button.

You can edit the selected export configuration by pressing the **Edit configuration** button.

You can delete the selected export configuration by pressing the **Delete configuration** button.

Export file name

You can enter the filename of the export file here. If the filename contains some placeholders, Mp3tag will replace them with the specific data at runtime.

Directory of export configurations

Specify the directory of export configurations here.

Append data

Enable this option, if you want Mp3tag to append data at an existing export file.

One file per directory

If this option is enabled, Mp3tag creates one export file for each directory.

Show selection dialog

If this option is enabled, Mp3tag shows a dialog with the available export configurations at Export.

Don't create directories

Since you can also use relative and absolute file paths as export file names - e.g. %artist%\%album%.html - this option prevents Mp3tag from creating any directories during the export process.

Write BOM

If you enabled writing of UTF-8 or UTF-16 via the $filename function in the export configuration, this option enables writing of a Byte Order Marker which is required by some applications.

Keyboard shortcuts

F2	Rename export configuration
F4	Edit export configuration
Insert	New export configuration
Delete	Delete export configuration

Export Configurations

You can specify any text, any of the many placeholders below and instructions for loops in your export configuration. Scripting commands can also be used.

You can set the filename of the exported file with the function $filename(name[,enc]). The optional second parameter sets the encoding of the export file. Possible values are ansi, utf-8 and utf-16.

```
$filename(test.txt)
$filename(test.txt,ansi)
$filename(test.txt,utf-8)
$filename(test.txt,utf-16)
```

`$loop(%fieldname%)` starts a new loop. The loop data is sorted by the fieldname.
`$loopend()` ends a loop.

If you want to limit the loop output to a certain number of entries, simply use `$loop(%fieldname%,num)`. This is useful to eliminate duplicate records (num=1).

Placeholders for export configurations

Characters with special functionality

`[...]` The contents of brackets are displayed only if at least one of the placeholders used inside the brackets has been found.

`'` Outputs raw text without parsing. This will output the contained string and ignore all reserved characters. If you want to output this character please use `''`.

`[]$%` You have to put the single quote around these reserved characters if you want to use them unparsed.

Placeholders

You can use these placeholders within the export configurations and the filename of the export file.

Metadata

`%artist%`	Artist
`%album%`	Album
`%comment%`	Comment
`%genre%`	Genre
`%title%`	Title
`%track%`	Track-Number
`%_total%`	Total number of tracks from xx/xx track-number field
`%year%`	Year
`%fieldname%`	The content of any field name

Technical info

`%_bitrate%`	Bitrate in kbit/s
`%_bitspersample%`	Bits per sample (available only for some audio formats)
`%_codec%`	Codec
`%_cover_mimetype%`	Mimetype of first cover art in the tag of the file

%_cover_size%	Size of first cover art in the tag of the file in bytes
%_cover_type%	Cover type of first cover art in the tag of the file
%_covers%	Count of cover art in the tag of the file
%_id3v2_character_encoding%	Encoding of the ID3v2 text frames of the file.
%_length%	Length (formatted)
%_length_seconds%	Length (in seconds)
%_mode%	Mode
%_samplerate%	Sample rate
%_tag%	Available tag types in file
%_tag_read%	Displayed tag type
%_tag_size%	Size of all tags in bytes
%_tag_size_appended%	Size of all appended tags in bytes
%_tag_size_prepended%	Size of all prepended tags in bytes
%_tool%	Additional codec information (not available for all supported audio formats).
%_vbr%	Bitrate type or compression profile

File name / path info

%_directory%	Name of the parent directory
%_extension%	File extension
%_filename%	File name without extension
%_filename_ext%	File name with extension
%_filename_rel%	Filepath with extension, relative to Export output file if output and track files are on same drive, else absolute
%_folderpath%	Path without file name
%_folderpath_rel%	Folderpath relative to Export output file if output and track files are on same drive, else absolute
%_parent_directory%	Name of the grandparent folder

%_path%	File name with path
%_volume%	Volume label
%_workingdir%	Name of current working directory
%_workingpath%	Path of current working directory
%_playlist_filename%	File name of current playlist without extension
%_playlist_filename_ext%	File name of current playlist with extension
%_playlist_folderpath%	Folderpath of current playlist

File properties

%_file_create_date%	Short creation date
%_file_create_datetime%	Long creation date
%_file_create_datetime_raw%	Long creation date (unix timestamp)
%_file_mod_date%	Short modification date
%_file_mod_datetime%	Long modification date
%_file_mod_datetime_raw%	Long modification date (unix timestamp)
%_file_size%	File size in human readable format
%_file_size_bytes%	File size in bytes
%_file_size_kb%	File size in kilobytes
%_file_size_mb%	File size in megabytes
%_crc%	CRC32 value of file content
%_md5%	MD5-Hash of the file (slow!)
%_md5audio%	MD5-Hash of the audio part of the file (slow and only supported for ID3- and/or APE-only tagged files)

System

%_app%	Mp3tag, e.g. Mp3tag v2.96
%_counter%	Current counter value (ascending from 1) of the immediately containing loop
%_date%	Short date

%_datetime%	Long date
%_max_counter%	Maximum value of counter of the last closed loop.

Global placeholders (providing totals for the preceding $loop/$loopend)

%_total_files%	Total number of files
%_total_size%	Total file size (formatted output)
%_total_size_raw%	Total file size in bytes
%_total_time%	Total playing time (formatted output)
%_total_time2%	Total playing time (formatted output without days)
%_total_time_raw%	Total playing time in seconds

Tags. Configuration > Tools

With this dialog you can specify external applications to work directly from Mp3tag's main window via the context-menu of the file view. You can create up to 99 external applications here. The first ten tools can also be started by the keyboard shortcuts Ctrl + 0 to Ctrl + 9.

Name

Name of tool. This name will show up as menu item in the context-menu of the file view.

Path

Full file path to the external application.

Parameter

Optional command line parameters. You can use the following placeholders:

%_path%	Long filename
%_filename_ext%	Short filename
%_filename%	Short filename without extension
%_filename_rel%	Relative file path
%_directory%	Directory
%_folderpath%	Folderpath
%_folderpath_rel%	Relative folderpath

for all selected files

By enabling this option, Mp3tag will run the tool for every selected file.

Examples

Please read this topic from the Frequently Asked Questions for sample tool configurations.

Default tool

If you want Mp3tag to start a special program at double clicks, you can select one of the tools from the dropdown list.

Tags. Configuration > Network

General

Timeout

Timeout for connections to the internet in seconds.

Proxy

Use proxy

Enable this option, if your connection to the internet goes through a proxy server.

Proxy

Proxy address e.g. *localhost*.

Port

Port of proxy server, e.g. 8000.

Username

Username to authenticate on the server.

Password

Password to authenticate on the server.
The password will be stored encrypted.

Mp3tag usage. Other Topics

Tags. Overview of keyboard hotkeys > Keyboard shortcuts

This is an overview of most keyboard shortcuts you can use with Mp3tag:

General

Ctrl + A	Select all
Ctrl + Shift + A	Invert selection
Ctrl + B	Restore all input fields
Ctrl + C	Copy tag or selected text
Ctrl + Shift + C	Copy file
Ctrl + D	Change directory
Ctrl + Shift + D	Upper directory
Del	Remove file (without deleting it)
Ctrl + Del	Delete file (to Windows Recycle Bin if globally activated)
Ctrl + Shift + Del	Delete file (WARNING: the file will be removed from your hard-drive!)
Ctrl + E	Export
Ctrl + Shift + E	Export (with an export file for each different directory)
Ctrl + F	Favorite directory
Ctrl + H	Add directory
Ctrl + I	freedb ... / WEB source
Ctrl + Shift + I	freedb ... / WEB source (Quick)

Ctrl + K	Auto-numbering wizard
Ctrl + Shift + K	Auto-numbering wizard (with counter reset for each subfolder)
Ctrl + L	freedb (local) ...
Ctrl + N	Select next file
Ctrl + O	Options ...
Ctrl + P	Playlist (all files)
Ctrl + Shift + P	Playlist (all files) with one playlist for each subfolder
Ctrl + Q	Show/hide tag input mask
Ctrl + R	Remove tag
Ctrl + S	Save tag
Ctrl + T	Read tag
Ctrl + Shift + T	Play
Ctrl + U	Unselect all
Ctrl + V	Paste tag or text
Ctrl + Shift + V	Move file
Ctrl + X	Cut tag or selected text
Ctrl + Z	Undo
Ctrl + 1, 2, ..., 0	User-defined Tool 1 to 10.
F2	Rename file
F3	Filter
F4	Play selected file(s)
Alt + Right arrow	Forward in directory history

Alt + Left arrow	Backward in directory history
Alt + Return	File properties
Alt + T	Extended Tag dialog to edit extended and user defined tag fields as well as embedded cover art.
Alt + Drag 'n Drop	Change order of files in list of files.
Alt + Arrow down	Change order of files in list of files.
Alt + Arrow up	Change order of files in list of files.
Alt + Page down	Move selected files to end in list of files.
Alt + Page up	Move selected files to beginning in list of files.
Shift + Del	Remove format strings which are not longer needed from drop-down lists.

Convert

Alt + 1	Tag - Filename
Alt + 2	Filename - Tag
Alt + 3	Filename - Filename
Alt + 4	Text file - Tag
Alt + 5	Tag - Tag
Alt + 6	Actions
Alt + Shift + 6	Quick action
Alt + Shift + 1	Tag - Filename (Preview)
Alt + Shift + 2	Filename - Tag (Preview)
Alt + Shift + 3	Filename - Filename (Preview)

Useful helpers

Ctrl + Space	Displays format string helper menu (e.g., at converters or actions)
Ctrl + Tab	Quick forward navigation between sub-windows (Tag Panel, Filter, File View)
Ctrl + Shift + Tab	Quick backward navigation between sub-windows (Tag Panel, Filter, File View)
Ctrl + Num + +	Automatically size the file view columns to match the content

Tags. Command Line Interface > Command Line Interface

These parameters can be used when starting Mp3tag from the command line:

Mp3tag.exe /fp:"<full-qualified path to directory>"

Starts Mp3tag in the specified directory using the directory as working directory.

Mp3tag.exe /fn:"<full-qualified file name>"

Starts Mp3tag with the specified audio file using the directory of the file as working directory.

/add or holding the Ctrl key down

Use this in combination with /fp or /fn to add files to the file list. Please note, that files are not automatically sorted when added to the file list.

Tags. License information > License information

Mp3tag v2.96

Mp3tag v2.96

is designed for private use and commercial use without sale ("Freeware"), if the following rules are respected:

The author, Florian Heidenreich has no responsibility if errors occurs in direct or indirect relation with the software.

Mp3tag v2.96

cannot be used in a military domain or in a similar domain (Weapon creation, armament, etc.).

Mp3tag v2.96

can be distributed through non-commercial channels, if the delivery happens in the form of the self-extracting setup file available from http://www.mp3tag.de.

Mp3tag v2.96

can be distributed through commercial channels if the author, Florian Heidenreich, agrees this delivery and if the delivery happens in the form of the self-extracting setup file available from http://www.mp3tag.de.

Tags. Credits > Special thanks to …

Everyone who has ever sent a donation to support my work on Mp3tag.

This and your feedback means a lot!

all **Developers** who created these libraries used by Mp3tag:

- boost::regex,
- libflac,
- libmpcdec,
- libogg,
- libopus,
- libvorbis,

- libwavpack,
- libspeex,
- pfc,
- NSIS,
- zlib

freedb.org, Discogs, and MusicBrainz

for their great community-driven metadata services.

Members of the Mp3tag Community

who contribute their knowledge and understanding to the public and support others in using Mp3tag.

Peter Pawlowski

who has created foobar2000, is the developer of the 'pfc' library, inspired Mp3tag's scripting and filter languages and has helped me with many difficult problems.

dano

who contributed a lot to the community and gave a good and critical partner in discussing features, bugs and future development of Mp3tag. He also fluently speaks Mp3tag's WEB Sources language and created lots of WEB Sources available on the forums as well as the initial versions of the Discogs and MusicBrainz WEB sources which are included in Mp3tag.

Helmut Heidenreich

who has introduced me to the vast areas of programming.

Markus Schmidt

who has designed the logo of Mp3tag and who inspired me with his good software.

Martin Lang

who has helped me with his constructive feedback getting Mp3tag to version 1.00.

Translators

who translated Mp3tag into more than 40 languages.

and **all users** of Mp3tag (yes, that's **you** :)).

Tags. **Donations** > Donations

Mp3tag is Freeware — Please consider donating!

I like Mp3tag being free to use for everyone.

When I started working on Mp3tag around **18 years ago**, I loved the idea of creating software and giving back to the community. Over the years, this initial intention hasn't changed. In addition, I've got to know many people from all around the world who enjoy using Mp3tag and get value out of it. This gives me a ton of motivation.

After being in academia, co-founding a company and working there for a while, I went on to be an Indie Software Developer in autumn 2016.
I create my income from the software I'm creating. I love it! It allows for having long stretches of deep work, having my own schedule and focusing on a product.

So, I am asking you to download my software, try it, and if you like it and want to be one of Mp3tag's 1000 True Fans[1], please come back and make a donation. **Any amount is appreciated!**
It is absolutely voluntary — so you won't get less support or crippled versions if you don't.

Thank you!

Support via PayPal

To make a payment with PayPal, you can use your credit card or bank account without the need to open an account. There are no additional costs.

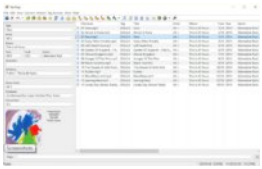
USD $ EUR € EUR € CAD $

You can also use the PayPal.Me link at **PayPal.Me/FlorianHeidenreich** to send money — it works with many currencies and is very easy to use.

Support via bank transfer

You can also show your support by IBAN/BIC bank transfer to:

Florian Heidenreich
Bank: Dresdner Volksbank Raiffeisenbank eG
IBAN: DE84850900003602201006
BIC: GENODEF1DRS

Tags. Mp3tag website > **MP3TAG** 

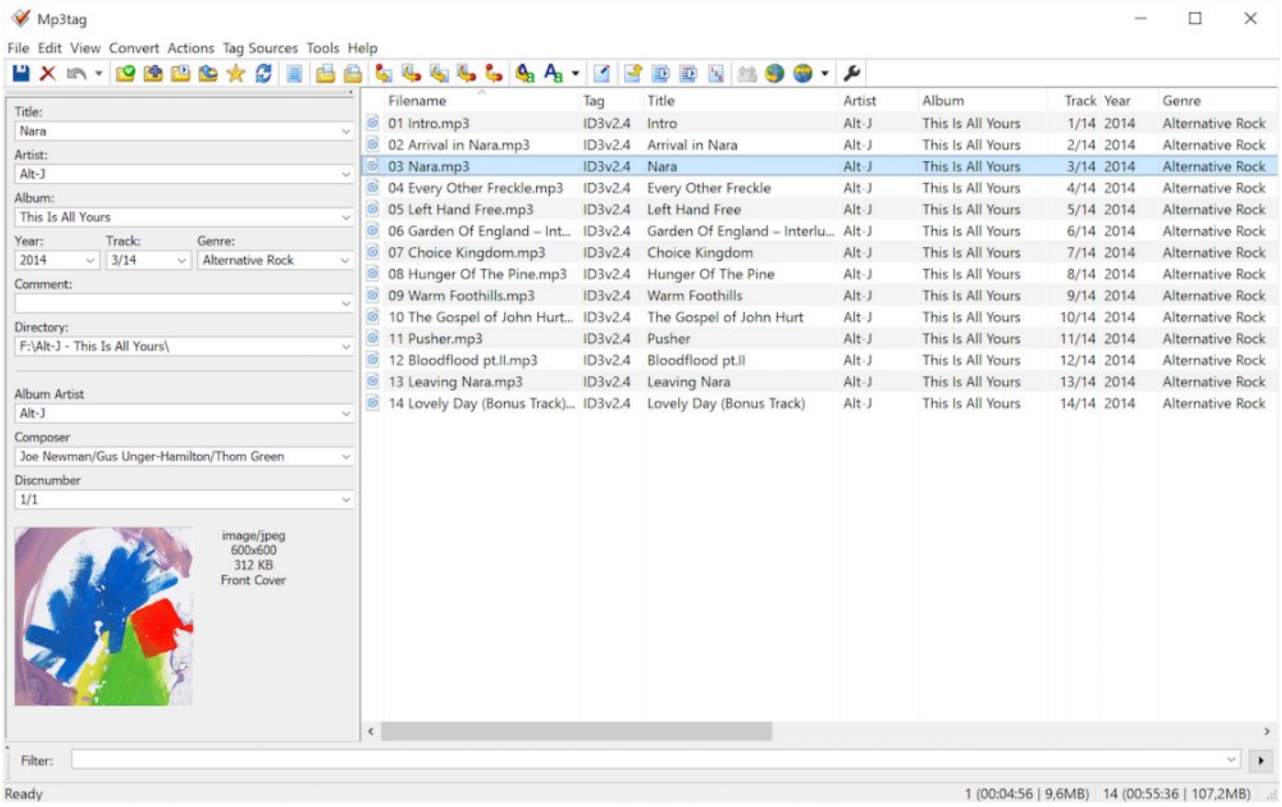
The universal tag editor and more ...

Mp3tag is a powerful and easy-to-use tool to edit metadata of audio files.

It supports batch tag-editing of ID3v1, ID3v2.3, ID3v2.4, iTunes MP4, WMA, Vorbis Comments and APE Tags for multiple files at once covering a variety of audio formats.

Furthermore, it supports online database lookups from, e.g., Discogs, MusicBrainz or freedb, allowing you to automatically gather proper tags and download cover art for your music library.

You can rename files based on the tag information, replace characters or words in tags and filenames, import/export tag information, create playlists and more.



Main features:

Batch Tag Editing

Write ID3v1.1, ID3v2.3, ID3v2.4, MP4, WMA, APEv2 Tags and Vorbis Comments to multiple files at once.

Support for Cover Art

Download and add album covers to your files and make your library even more shiny.

Import from Discogs, freedb, MusicBrainz

Discogs, freedb, MusicBrainz, and more.

Replace characters or words

Replace strings in tags and filenames (with support for Regular Expressions).

Create Playlists automatically

Create and manage playlists automatically while editing.

Rename files from tags

Rename files based on the tag information and import tags from filenames.

Export to HTML, RTF, CSV

Generate nice reports and lists of your collection based on user-defined templates.

Full Unicode Support

User-interface and tagging are fully Unicode compliant.

Besides these main features Mp3tag offers a variety of other functions and features ranging from batch export of embedded album covers, over support for iTunes-specific tags like media type or TV Show settings, to combining multiple actions into groups that can be applied with a single mouse click.

Supported Audio Formats

- Advanced Audio Coding (aac)
- Apple Lossless Audio Codec (alac)
- Audio Interchange File Format (aif / aifc / aiff)
- Direct Stream Digital Audio (dsf)
- Free Lossless Audio Codec (flac)
- Matroska (mka / mkv)
- Monkey's Audio (ape)
- Mpeg Layer 3 (mp3)
- MPEG-4 (mp4 / m4a / m4b / m4v / iTunes)
- Musepack (mpc)
- Ogg Vorbis (ogg)
- IETF Opus (opus)
- OptimFROG (ofr / ofs)
- Speex (spx)
- Tom's Audio Kompressor (tak)
- True Audio (tta)
- Windows Media Audio (wma)
- WavPack (wv)
- WAV (wav)