

ZotFile

Advanced PDF management for Zotero

Download

[View the Project on GitHub](#)



49

Content

[Features](#) • [Install Zotfile](#) • [Renaming Rules](#) • [User-Defined Wildcards](#) • [Extract PDF Annotations](#) • [Hidden Options](#) • [Reporting a Bug](#) • [Changelog](#)



FEATURES

Zotfile is a Zotero plugin to manage your attachments: automatically rename, move, and attach PDFs (or other files) to Zotero items, sync PDFs from your Zotero library to your (mobile) PDF reader (e.g. an iPad, Android tablet, etc.) and extract annotations from PDF files.

★ Attach New Files to Zotero Items

ZotFile can rename and add the most recently modified file from the Firefox download or a user specified folder as a new attachment to the currently selected Zotero item. It renames the file using metadata from the selected Zotero item (user configurable), and stores the file as a Zotero attachment to this item (or alternatively, moves it to a custom location).

★ (Batch) Rename and Move Attachments based on Zotero Metadata

The user can also select any number of Zotero items and automatically rename and move all attachments of these items according to the user defined rules using

metadata of the respective zotero item (batch processing).

★ Sync PDFs with your iPad or Android tablet

To read and annotate PDF attachments on your mobile device, zotfile can sync PDFs from your Zotero library to your (mobile) PDF reader (e.g. an iPad, Android tablet, etc.). Zotfile sends files to a location on your PC or Mac that syncs with your PDF reader App (PDF Expert, iAnnotate, GoodReader etc.), allows you to configure custom subfolders for easy access, and even extracts the annotations and highlighted text to Zotero notes when you get the files back from your tablet. Instructions are below.

★ Extract Annotations from PDF Files

After highlighting and annotating pdfs on your tablet (or with the PDF reader application on your computer), ZotFile can automatically extract the highlighted text and note annotations from the pdf. The extracted text is saved in a Zotero note. Thanks to Joe Devietti, this feature is now available on all platforms based on the pdf.js library.

INSTALL ZOTFILE

To start using zotfile, make sure that Zotero is installed and follow these simple steps:

1. Install ZotFile

For *Zotero Firefox*, click on the the download link above and follow the instructions on the Mozilla Add-Ons page. For *Zotero Standalone*, first download the extension file (follow the download link above, click on 'Continue to Download'. In Firefox, right-click 'Add to Firefox' and select 'Save Link As'. In other browsers, click on 'Download Now' (it's greyed out) and then 'download anyway'). Now start Zotero Standalone, and go to 'Tools->Add-ons->Tools for all Add-ons (the small,drop-down wheel in the top right corner)->Install Add-on From File' and pick the downloaded .xpi file.

2. Change the Source Folder for Attaching new Files

To attach new files to Zotero items, zotfile looks for the most recently modified (e.g. just downloaded) file in a user specified folder. For Zotero Firefox, this option is set to the Firefox download folder by default. For Zotero Standalone, this option has to be changed on the 'General Settings' tab in the preference window (Zotero Actions -> ZotFile Preferences). The source folder can be set to any location but I generally recommend setting it to your browser's download folder such as ~/Downloads on the mac for most browsers.

3. Changing other Options (optional)

ZotFile offers many other options that can be changed by the user. Most of them are located in the zotfile preference window under Zotero Actions -> ZotFile Preferences.

Syncing pdf attachments with your ipad or android tablet

To read and annotate PDF attachments on your mobile device, zotfile can sync PDFs from your Zotero library to your (mobile) PDF reader (e.g. an iPad, Android tablet, etc.). For this purpose, Zotfile sends files to a location on your PC or Mac that syncs with your PDF reader App (PDF Expert, iAnnotate, GoodReader etc.), and gets them back when you have finished reading them. To get started, first set up a folder on your PC or Mac that syncs with your tablet reader application. Files that are copied to this folder should automatically appear in your PDF reader application. One possibility is Dropbox, which is free for up to 2GB of space and works with most PDF reader apps. More detailed instructions as well as alternative options should be available on the website of your PDF reader App. Second, open the 'Tablet Settings' tab in the zotfile preference window and enable the option 'Use ZotFile to send and get files from tablet'. Third, change the zotfile location for files on the tablet to the folder that syncs with your pdf reader app ('Base Folder' on the 'Tablet Settings' tab). Forth (optional), set up subfolders that make it easy to sort your files in the tablet folder so that you can easily find them on your tablet. You can now start sending pdfs (or other files) to your tablet. Simply right-click on a zotero item and select 'Send to Tablet' under 'Manage Attachments'. When you are done reading and annotating your pdf, just get the file back from the tablet by

clicking on 'Get from Tablet' under 'Manage Attachments'. ZotFile will automatically remove the file from your tablet folder and extract the annotations from the pdf file to a zotero note. Alternatively, you can use the 'Scan Tablet Files' function in Zotero's Action Menu to automatically check whether any files in the tablet folder have changed. ZotFile then asks the user how to proceed for each modified file.

RENAMING RULES

ZotFile renames files based on bibliographic information from the currently selected Zotero item. You can change the renaming rules in the zotfile preference window under renaming rules (Zotero Actions -> ZotFile Preferences). The option 'Renaming Format' allows you to create custom renaming rules using wildcards, which are replaced by metadata from the selected Zotero item. Zotfile also supports optional and exclusive wild-cards. Optional wild-cards mean that `{-%y}` only includes the separator - in the filename if `%y` is defined. Exclusive wild-cards such as `%s|%j` will generate the entry for `%s` if that exists and the entry for `%j` otherwise. Other characters between the wildcards and the bar are ignored (`%s | stuff %j | - more %p` is equivalent to `%s|%j|%p`). Some examples are below.

Wildcards

- `%a` last names of authors (not editors etc) or inventors. The maximum number of authors are changed under 'Additional Settings'.
- `%I` author initials.
- `%F` author's last name with first letter of first name (e.g. EinsteinA).
- `%A` first letter of author (useful for subfolders)
- `%d,%D, %L, %l` wildcards for editors, same as for authors.
- `%y` year (extracted from Date field)
- `%t` title. Usually truncated after : . ? The maximal length of the remaining part of the title can be changed.
- `%T` item type (localized)
- `%j` name of the journal
- `%p` name of the publisher
- `%w` name of the journal, publisher, meeting name for presentation etc.
- `%s` journal abbreviation

- %v journal volume
- %e journal issue
- %f pages
- %c collection path (only for sub-folders, not file names). When item is in multiple collections, user can choose between the different collections.
- %n patent number (patent items only)
- %i assignee (patent items only)

Examples

Abbott, Andrew, and Alexandra Hrycak (1990): Measuring Resemblance in Sequence Data: An Optimal Matching Analysis of Musicians' Careers. American Journal of Sociology 96:144-185.

- {%a-}{-%y-}{-%j (%s)} - Abbott-1990-American Journal of Sociology (AJS)
(if either “%j” or “%s” is empty, “Abbott-1990”)
- {%a-}{-%y-}{%s|%j} - “Abbott-1990-AJS” or “Abbott-1990-American Journal of Sociology” if “%s” is empty
- {%a_}{%y_}{%t}: Abbott_Hrycak_1990_Measuring Resemblance in Sequence Data
- %a-%y %t: Abbott_Hrycak-1990 Measuring Resemblance in Sequence Data
- {%w_}{%y_}{%a}: American Journal of Sociology_1990_Abbott_Hrycak

With ‘Maximum number of authors’ set to 1 and ‘Add suffix ...’ set to ‘et al’

- %a_%y_%t: Abbott et al_1990_Measuring Resemblance in Sequence Data

With ‘Maximum number of authors’ set to 1, ‘Add suffix ...’ disabled, and ‘Maximum length of title’ set to 10

- %a_%y_%t: Abbott_1990_Measuring

USER-DEFINED WILDCARDS

All wildcards are now defined in the hidden preference `zotfile.wildcards.default`

and can be changed by the user. But I strongly suggest that you do not change this preference. Instead, there is a second hidden preference `zotfile.wildcards.user`, which allows you to add and overwrite wildcards (hidden preference can be changed in `about:config`). This is a preference for advanced users without any error checking. Be careful what you do and validate your JSON using [this site](#)! By default, `zotfile.wildcards.user` is set to `{}` so that no user wildcards are defined. Below is an example JSON that defines wildcards for `%1`, `%2`, `%3`, `%4`, and `%5` illustrating all the possibilities:

1. String with the name of Zotero field (`%1`)
2. JSON with item type specific field names (`%2`)

Always include a `default` value. Otherwise this is not going to work. A list of all item types is available [here](#).

3. JSON with `field` element and transformations based on regular expressions (`%3` and `%4`)

ZotFile uses the specified `field` as an input string and then applies the transformations specified in `operations`. The value of `field` can either be the name of a Zotero field (see 1) or a javascript object with item type specific field names (see 2). `operations` is an array of javascript objects and supports three types of transformations that are identified by the `function` element:

- `exec`: Search for matches using regular expressions (`%3`). Zotfile uses the `exec()` function based on the regular expression defined in `regex`, and returns the element specified in `group` so that `0` returns the matched text and higher values the corresponding capturing parenthesis. If the match fails, this operation returns the input data.

Required parameters: `regex`

Optional parameters: `group` (default `0`), `flags` (default `"g"`)

- `replace`: Replaces matches of a pattern using regular expressions (`%4`). Zotfile uses the `replace()` function with the regular expression `regex` and replacement string `replacement`. The replacement string can include `$n`

for the *n*th parenthesized sub-match string and other special replacement patterns (see [replace\(\) documentation](#)). The wildcard %4, for example, takes the date when an item was added (format 2012-02-18 02:31:37) and returns the reformatted date as 20120218.

Required parameters: **regex**, **replacement**

Optional parameters: **flags** (default "g")

- **toLowerCase** etc: Simple string functions that do not require any additional arguments(%5). Currently supported are **toLowerCase**, **toUpperCase**, and **trim**.

flags is an optional parameter for both searching and replacing and corresponds to [flags for regular expressions](#) in javascript (default: "g").

4. Finally, the wildcard %5 illustrates a complex combination of operations (completely arbitrary).

Example for user-defined wildcards

```
{
  "1": "publicationTitle",
  "2": {
    "default": "publicationTitle",
    "book": "publisher",
    "bookSection": "publisher"
  },
  "3": {
    "field": "title",
    "operations": [{
      "function": "exec",
      "regex": "([\\w , -]{1,50})[:\\.?!]?",
      "group": 1
    }]
  },
  "4": {
    "field": "dateAdded",
    "operations": [{
      "function": "replace",
      "regex": "(\\d{4})-(\\d{2})-(\\d{2})(.*)",
      "replacement": "$1$2$3",
    }]
  }
}
```

```

        "flags": "g"
    }
  },
  "5": {
    "default": {
      "field": "title",
      "operations": [
        {
          "function": "replace",
          "regex": "\\d",
          "replacement": ""
        },
        {
          "function": "exec",
          "regex": "([\\w , -]{1,10})[:\\.?!]?",
          "group": 1
        },
        {
          "function": "toUpperCase"
        },
        {
          "function": "trim"
        }
      ]
    },
    "journalArticle": "publicationTitle"
  }
}

```

Item types and field names

A complete list of Zotero fields is available [here](#) (`dateModified` and `dateAdded` seem to be missing from that list) and all the item types are [here](#). The fields for each item type are listed on [this page](#). Zotfile defines a number of *additional fields* that can be used as part of wildcards: `itemType` is the language specific item type, `formattedTitle` is the title formatted using the options defined in the zotfile preferences, `author` is the author string formatted using the zotfile preferences, `authorLastF` is the author string formatted as LastnameF, and `authorInitials` are the initial of the authors.

Formatting rules

There are a couple of formatting rules for the user-defined wildcards:

- Wildcards can only be one character

- Use double escape characters in regular expression so that a `\d` becomes `\\d`
- Always use `"` never `'`

Most importantly, [validate your json](#). Check out `zotfile.wildcards.default` for more examples (see below). Finally, the JSON has to be reformatted to one line that can be pasted into the preference field in `about:config`. Here is the example from above:

```
{
  "1": "publicationTitle",
  "2": {
    "default": "publicationTitle",
    "book": "publisher",
    "bookSection": "publisher"
  },
  "3": {
    "field": "title",
    "operations": [
      {
        "function": "exec",
        "regex": "([\\w , -]{1,50})[:\\.?!]?",
        "group": 1
      }
    ]
  },
  "4": {
    "field": "dateAdded",
    "operations": [
      {
        "function": "replace",
        "regex": "(\\d{4})-(\\d{2})-(\\d{2})(.*)",
        "replacement": "$1$2$3",
        "flags": "g"
      }
    ]
  },
  "5": {
    "default": {
      "field": "title",
      "operations": [
        {
          "function": "replace",
          "regex": "\\d",
          "replacement": ""
        },
        {
          "function": "exec",
          "regex": "([\\w , -]{1,10})[:\\.?!]?",
          "group": 1
        },
        {
          "function": "toUpperCase"
        },
        {
          "function": "trim"
        }
      ]
    },
    "journalArticle": "publicationTitle"
  }
}
```

You can use a javascript console such as Firefox's Scratchpad to test whether the JSON is properly formatted:

```
wildcard = '{"1": "publicationTitle"}';
JSON.parse(wildcard)
```

Default setting for `zotfile.wildcards.default`

The information in this file might not be up to date but you can look at the default wildcards and learn something about user-defined wildcards [here](#). The minified version in one line is [here](#) so that you can copy it to `zotfile.wildcards.default` if you screw up.

EXTRACT PDF ANNOTATIONS

Zotfile can extract annotations and highlighted text from many PDF files. The extracted annotations are saved in Zotero notes and you can go back to the annotation in the pdf by clicking on the link after the extracted text. PDFs are a very complex format and the extraction will never work for all files. If you can not copy & paste meaningful text from the file in your pdf viewer (open your pdf viewer (not the

browser plugin), select text, copy and paste it somewhere), zotfile won't be able to extract the highlighted text either. If you can, there is a chance that future versions of zotfile will solve the problem. When you have a pdf file that does not work or with clear spacing problems, feel free to share the file on the [zotfile thread](#) in the Zotero forum or upload it to the [zotfile Zotero group](#) in the **PUT FILES HERE** folder. But only share files for which you can copy and paste text! Otherwise you are wasting everyone's time.

On Mac OS, you can also use poppler to extract pdf annotations (ZotFile Preferences -> Advanced Settings). Currently, pdf.js is more reliable and should be the default in most cases. The poppler-based tool, however, is faster and might handle certain pdf standards that are not yet supported by pdf.js.

Goto Annotation in PDF

Zotfile adds a link to extracted annotations that allows you to open the pdf file at the page with the annotation. Just click on the link after some extracted text and your pdf should open on the correct location. On Windows, zotfile detects the default pdf viewer and opens the pdf on the correct page (not tested for Windows 8). Adobe Reader, Foxit and PDF-XChange all work (other might as well but are untested). Adobe Reader, however, does not jump to the correct page when the file is already open (Foxit and PDF-XChange do). If zotfile is unable to detect the default viewer or you want to force zotfile to use a different viewer, simply change the hidden option `zotfile.pdfExtraction.openPdfWin` to the desired path (e.g. `C:\Program Files\Adobe\Reader 11.0\Reader\AcroRd32.exe`). On Mac OS, zotfile works with Preview (the default) and Skim. Preview, however, does not support scripting very well and has certain limitations. Don't press any keys while the pdf is opening, for example. [Skim](#), an alternative pdf viewer for Mac OS, works much better! Just set the hidden option `zotfile.pdfExtraction.openPdfMac_skim` to `true` and zotfile will open the pdf in Skim. For Linux, zotfile first tries `okular` and then `evince` but you can also set the `zotfile.pdfExtraction.openPdfLinux` option to change the default behavior. One example would be `/usr/bin/okular -p`, which tell zotfile the path and the argument for the page number.

HIDDEN OPTIONS

Zotfile has a number of hidden options that allow you to further configure zotfile. You can access the hidden options through `about:config`. In Zotero Firefox, type

`about:config` in the url bar. In Zotero Standalone, go to Actions->Preferences->Advanced->Open `about:config`.

Search for `extensions.zotfile` to see a list of the hidden zotfile options. Here is a list of the options that can be changed by the user (I strongly discourage to change any of the other options):

- `.allFiles`

By default, zotfile's 'Attach New File' function attaches the most recently modified file from the user defined folder. With this option set to true, zotfile attaches all files in the user defined folder to the currently selected zotero item. (Note: I haven't tested this for a while but it should still work)

- `.disable_renaming`

Disable any renaming of files - just moves them to the specified location.

- `.tablet.mode`

In *background mode* (mode=1, default), zotfile leaves zotero attachments at their current location and moves a copy of the file to the tablet folder (set in the zotfile preference window) when they are send to the tablet. Getting the file back from the tablet replaces the zotero attachment file and removes it from the tablet folder. This mode is recommended when you sync attachment files in your zotero library across multiple computers or when you index your attachments.

The *foreground mode* (mode=2) moves the attachment file to the tablet folder and links to this location from zotero. In this mode there is always only one copy of the file. You can not, however, sync linked attachments to the zotero server.

- `.confirmRepush`

By default, zotfile asks the user whether an attachment should be send to the tablet that is already on the tablet, which can be useful to move it to a different subfolder. This user confirmation can be disabled with this option.

- `.tablet.tagParentPush`, `.tablet.tagParentPush_tag`, `.tablet.tagParentPull`,
`.tablet.tagParentPull_tag`

These options allow the user to tag the parent item when sending (push) or getting back (pull) attachments to or from the tablet.

- `.pdfExtraction.NoteHtmlTagStart`, `.pdfExtraction.NoteHtmlTagEnd`,
`.pdfExtraction.HighlightHtmlTagStart`, `.pdfExtraction.HighlightHtmlTagEnd`,
`.pdfExtraction.UnderlineHtmlTagStart`, `.pdfExtraction.UnderlineHtmlTagEnd`

These options allow the user to fine-tune the formatting of the extracted PDF annotations in the zotero note. They define the opening and closing html tag for different types of annotations. The default settings format highlighted text from the pdf normally, note text in italics (`<i>` for start and `</i>` for end), and underline underlined text (`<u>` for start and `</u>` for end). The end options for note, highlight and underline have to be the closing tag for the corresponding start option.

- `.pdfExtraction.NoteRemoveHyphens`

By default, zotfile removes hyphens from extracted text. Setting `NoteRemoveHyphens` to false, disables this option.

- `.UsePDFJSandPoppler`

With this option, zotfile extracts PDF annotations twice using both pdf.js and poppler. This option only works on Mac OS when the poppler based extraction script is installed.

- `info_window_duration`, `info_window_duration_clickable`

Duration (in milliseconds) for which the info windows show up.

- `pdfExtraction.replacements`

Custom, regular expression-based replacements in extracted annotations. This can be useful because some pdfs contain 'broken' characters. For example, `[{"regex": " ?\u00f0", "replacement": " ("}]` replaces the unicode character ð with

(to fix a problem in pdfs from a certain publisher. In this case, ð is a problem with the pdf and not with zotfile's extraction. The hidden option can be used to fix it.

REPORTING A BUG

You can report bugs on the [Zotfile thread](#) in the Zotero forum. Please provide information about your system (Windows, Mac OS, Linux etc) as well as your Zotfile, Zotero and Firefox version. Also make sure that you can reproduce the bug and describe the steps as closely as possible. In addition, any information from the Error Console are very helpful. You can check the Error Console in Zotero for Firefox by going to 'Tools->Web Dev.->Error Console' in Firefox and look for any zotfile related stuff after the bug occurred. For zotfile bugs, the 'Source File' should be something like `chrome://zotfile/content/...` (most likely zotfile.js). You can also clear the console, execute the actions that caused the problem and then check again. If I ask you to provide a Report ID, follow the instructions [here](#).

CHANGELOG

Changes in 4.0 and 4.1

- Goto annotation in pdf

This is not really a new feature but with two recent changes in Zotero (see [this](#) and [this](#) pull request), it became much more useful! Simply click on the link that is part of your extracted annotations, and zotfile will open the pdf on the page with the annotation. The feature now works on Windows as well (thanks to aurimasv) and I have added support for Skim on Mac. Check out the [documentation](#) for some more details.

- Improved extraction of annotation

This version includes four improvements for the extraction of annotations. First, the new version greatly improves the detection of correct spaces between words. Second, the extraction is now based on the most recent pdf.js version ([here](#) is my fork with the modified version of [pdf.js](#) used in zotfile). With this update, zotfile should work with more pdfs. Second, the extraction is now about 40-60% faster (depending on the pdf) thanks to some

improvements in the extraction code. Third, the extraction now runs in the background so that Zotero is not blocked while annotations are extracted.

- Get Table of Contents from PDF

Similar to [Mendeley](#), ZotFile can now get the table of contents from PDF and save it with links to the correct page in attachment notes. Simply click on the the desired section and zotfile opens the pdf on the correct page. Zotfile automatically extracts the table of content for all newly added pdf attachments (disable with `zotfile.pdfOutline.getToc` setting) or you can manually extract the toc using 'Manage Attachments->Get Table of Contents' (remove menu item with `zotfile.pdfOutline.menuItem` setting). This feature only works for pdfs that have an embedded table of content. Unfortunately, many don't.

- Support for sub-folders based on collection path (via wildcard %c) (thanks to [simpzan](#))
- Tablet status is now shown in right-pane with other attachment information

The right pane now includes a row with the current tablet status such as **No** for files that are not on the tablet or **[Basefolder]** for files that are in the tablet base-folder. Click on this information to change the tablet status and open or reveal the file on the tablet (very convenient because double-clicking on the attachment opens the imported zotero attachment and not the file on the tablet).

- Extract Annotation Color (thanks to [RAG2ko](#))

Zotfile now extracts the color of annotations. The color can be used in two ways: (a) To format extracted annotations: Just add some css styling to the hidden options for formatting. For example, change `formatAnnotationHighlight` to `<p style="background-color:%(color);">"%(content)" (%(cite))</p>`. The wildcard `%(color)` is replaced by the annotation color (you can also use `%(color_category)` for the color category—reddish is red etc). This styling has the same effect but looks a little better: `<p>"%(content)" (%(cite))</p>` (b) To create separate notes for

color categories: The hidden option `pdfExtraction.colorNotes` allows you to create a separate note for each color category (yellow, red, green etc annotations).

- Improved formatting of extracted annotations with hidden options (`formatNoteTitle`, `formatAnnotationNote`, `formatAnnotationHighlight`, `formatAnnotationUnderline`)

WARNING: I removed the old settings. You will have to change the new settings if you want to customize the format of the note with extracted annotations.

- Improved default PDF reader detection on Windows (thanks to aurimasv)
- Hidden option `pdfExtraction.replacements` for custom, regular expression-based replacements in extracted annotations. This can be useful because some pdfs contain 'broken' characters. For example, `[{"regex": "?\\u00f0", "replacement": " ("}]` replaces the unicode character `ð` with `(` to fix a problem in pdfs from a certain publisher. In this case, `ð` is a problem with the pdf and not with zotfile's extraction. The hidden option can be used to fix it.
- Information for attachments on the tablet is now hidden in the attachment note
- Improved information windows
- Improved renaming function (avoids re-indexing of linked attachments)
- New wildcards for editors (`%d`, `%D`, `%L`, `%l`)
- Fix bug with unnecessary suffix after multiple renames of same file
- Further improvements for the extraction of annotations in 4.1
- Truncate title after '!

Changes in 3.1

- New zotfile webpage at www.zotfile.com (please update links)
- User-defined wildcards
- watch folder now adds an attachment and retrieves metadata if no file is selected
(change message, change version)
- fix pdf.js issue with some pdf annotation (see [this](#) discussion)

- New hidden preferences for duration of info windows (`info_window_duration` and `info_window_duration_clickable`)
- The `%w` wildcard now maps to the correct field for most item types
- The `%u` wildcard as redundant (use `%y` instead)
- Choose the number of authors to display when truncating authors during renaming (thanks to bwiernik)
- fix problem that small info window does not disappear

Changes in 3.0.3

- fix for issue with showing website on every restart
- fix for issue when adding new attachment

Other small fixes mainly for Italian translation, restriction of saved searches and renaming based on collections.

Changes in 3.0

- Improved extraction of annotation

The extraction of pdf annotations using pdf.js works much better now! Zotfile uses a modified version of the updated [pdf.js](#) library ([here](#) is the fork). The new version supports more pdf standards, detects spaces more precisely, sorts annotations in the correct order, and future updates to new versions of pdf.js are relatively easy. There are still pdfs that won't work though! Some pdf standards are not yet supported and if you can not copy & paste text from the pdf file using your pdf viewer (e.g. Preview), it's unlikely that zotfile can help.

- Goto annotation in pdf

The extracted annotations now include a link that opens the pdf file on the corresponding page. For the extracted annotation "**This is my text**" (**zotfile 2013: 4**), **zotfile 2013: 4** is a link that opens the pdf on the page with the annotation. Currently, this feature only works from reports (right-click on item and select **generate report**) but future version of Zotero might be able to open the links directly from the note (see discussion [here](#) and [here](#))

- Tablet feature: Restrict saved search for tablet files to sub-folders

You can now right-click on the two saved searches for tablet files and restrict them to one of your custom sub-folders. This is very helpful to quickly see the files that are in a specific sub-folder.

- Tablet feature: Support of colored tags in Zotero 4

Zotfile now tags the parent item when an attachment is send to the tablet so that you can easily see which items are on the tablet. Simply assign colors to the two tablet tags (`_tablet` and `_tablet_modified`). But DO NOT manually add the tag to items or attachments (also not using the keys for colored tags).

- Italian localization (thanks to Roberto Caviglia)
- Remove empty sub-folders when getting files from tablet
- Fix alert window (headline was missing in Zotero 4)
- Fix automatic renaming option "Only ask if..."
- Fix for zotfile item menu on Zotero as a tab

Changes in 2.3.4

- compatible with Zotero 4.0
- Annotation extraction compatible with FF 20

Changes in 2.3.1

- adding attachments from watched folder now works when child item is selected
- language improvements for French and German
- bug fix for subfolders based on wildcards

Changes in 2.3

- enhanced renaming rules (thanks to [Midnighter](#))

1) Optional wild-cards: `{-%y}` only includes - if `%y` is defined.

2) Exclusive wild-cards: `%s|%j` journal abbr. or if not defined full journal name.

(for examples see below)

- additional wild-cards for author formating (author initials %I and lastnameF %F), pages (%f) and short title (%h)
- watch source folder for new files

Whenever the focus changes to the item list in Zotero, Zotfile checks for new files in the source folder. If a new file was added to the folder, zotfile uses a clickable, non-disruptive window to ask the user whether s/he wants to attach that file to the currently selected Zotero item.

- revised auto rename with additional options

Four options: Never, Always ask, Only ask if item has other atts, Always rename. The 'asking' uses a clickable, non-disruptive window that appears in the bottom right corner - same as previous info window but clickable.

- revised notifications and error handling
- Zotfile translation to German (thanks to [wuffi](#)) and French (thanks to [gracile-fr](#))
- Allow periods as delimiter in filenames (thanks to [jjatria](#))
- new option: "lower case" filenames (thanks to [jjatria](#))
- new option: disable renaming so that attachments are only moved (hidden: `.disable_renaming`)
- new option: set opening and closing quotation mark for extracted annotations (hidden: `.openingQuotationMarks`, `.closingQuotationMarks`)
- new option: remove periods from filenames (hidden: `removePeriods`)
- bug fix: download of poppler tool was broken
- bug fix: preview of renaming rules for Unix and Windows
- bug fix: sending to and getting from tablet deleted note content

Examples for enhanced renaming rules

%j - journal; %s - journal abbreviation

`{%a}{-%y}{-%j (%s)}` - author-2001-Proceedings of the National Academy of Sciences (PNAS)

(if either %j or %s is empty, author-2001)

`{%a-}{%y-}{%s|%j}` - author-2001-PNAS or author-2001-Proceedings... if %s is empty

For full description, see [renaming rules](#).

Changes in 2.2.3

- bug fix: check whether selected attachments are valid (no top-level item, no web attachments and attachment exists)
- bug fix: editing custom folder created error if user had maximum number of custom folders

Changes in 2.2.2

- Add option to change delimiter between multiple authors (thanks to [gracile-fr](#))

Changes in 2.2

- New feature: automatic renaming of attachment files (thanks to [Robin Wilson](#))
- Bug fix: zotfile produced error when trying to move open files on Windows (thanks to Dominik)

Changes in 2.1

- Important: the tag for tablet files was changed from '_READ' to '_tablet'
- New saved search for modified files on tablet (updates automatically, replaces 'Scan Tablet Files' function, which has been removed)
- Zotfile menu items only appear for bibliographic items and attachments (not for notes)
- Bug fix: allow the extraction of annotations in group libraries
- Other bug fixes

Changes in 2.0

- Sync Zotero Attachments with your iPad or Android tablet
- Extract Annotations from PDF Files (thanks to Joe Devietti)
- redesigned preference pane
- many more features and bug fixes

