# biblatex-publist

Jürgen Spitzmüller*

Version 2.13, 2024/08/21

## Abstract

The `biblatex-publist` package provides a `biblatex` [1] style file for publication lists – a bibliography listing one's own publications. It employs `biblatex`'s *author-year* style by default (which can be changed), but provides extra features needed for publication lists, such as the omission or highlighting of the own name from/in author or editor data and specific numbering. The package requires at least version 3.8 of `biblatex` and `biber` (the respective version as required by `biblatex`).

## Contents

1 **Aim of the Package** 2

2 **Usage** 3
- 2.1 Standard Usage . . . 3
- 2.2 Additional Options . . . 4
- 2.3 Referring to Entries . . . 7
- 2.4 Numbers Games . . . 8
- 2.5 Statistics . . . 10
- 2.6 Handling Multiple Authors and/or Name Variants . . . 10
- 2.7 Truncation of Name Lists . . . 11
- 2.8 Indicating Author Roles . . . 11
- 2.9 Adding Specific Entry-Related Information . . . 12
  - 2.9.1 Journal Impact Factor . . . 13
  - 2.9.2 Number of Citations . . . 13
  - 2.9.3 Peer Review . . . 13
  - 2.9.4 OpenAccess Strategies . . . 14

3 **Customization** 14
- 3.1 Auxiliary Macros and Lengths . . . 14
- 3.2 Using a Different Base Style . . . 15
- 3.3 Clickable Titles . . . 16

4 **Localization** 16

5 **Further Extensions** 17
- 5.1 Review Bibliography Type . . . 17

---

[*] Please report issues via `https://github.com/jspitz/biblatex-publist`.

5.2 Partial Translations . . . . . . . . 18

**6 An Example** **18**

**7 Filtering** **18**

7.1 Filter out own work . . . . . . . . 18

7.2 Filter on publication or peer-review status . . . . . . . . 20

**8 Sorting** **20**

8.1 Sorting Publication Lists . . . . . . . . 20

8.2 Sorting Templates . . . . . . . . 21

**9 Revision Log** **22**

**10 Credits** **28**

# 1 Aim of the Package

The `biblatex-publist` package provides a `biblatex` bibliography style specifically for academic publication lists. Such lists, which are a central part of the academic CV, contain all or selected publications of a specific author, usually sorted by genre and year. Even though publication lists are actually nothing else than (specific) bibliographies, they diverge from those in some respects. Most notably, it is widespread practice to omit your own name in your publication list and only list your co-authors, if there are any, or to highlight your own name (e. g., with bold face letters). If you want to follow this practice, a normal bibliography style does not produce the desired result.

Given the fact that maintaining a publication list is a routine task in an academian's life, it is surprising how few specified solutions are at disposal to generate such lists (particularly from BibTEX data). For classic BibTEX, Nicolas Markey provides (off CTAN) a BibTEX style file dedicated to that task, *publist.bst*[1]. The `biblatex-publist` package started off as the attempt to emulate the features of *publist.bst* with `biblatex`'s means; it thus partly draws on its conceptual ideas. Meanwhile, however, it has significantly exceeded that initial goal and provides many more features than *publist.bst*, thereby accounting for my own needs and for requests that reached me.

The list of features includes:

- Omit or hide selected author names, with a specific indication of co-authors in the former case
- Indicate author roles
- Filter publications of specific authors from heterogeneous bibliography databases
- Date-centric sorting (as typical for publication lists)
- Number items in ascending or descending order, globally or sectional
- Highlight the year of publication
- Optionally indicate the year of publication (sets) in the margin
- Clickable titles, linking to the URL, DOI, or a web catalog

---

[1] See `http://www.lsv.fr/~markey/BibTeX/publist/publist.bst` for the style file, `http://www.lsv.fr/~markey/BibTeX/publist/doc.ps.gz` for documentation; see also [2].

- Indicate if/how a title has been peer-reviewed
- Mark OpenAccess publications
- Give bibliometric information (journal impact factor)
- List reviews of specific titles
- Adjustable base bibliography style

Many features have been suggested by users of the package. If you miss a specific feature, feel free to suggest it via `https://github.com/jspitz/biblatex-publist/issues`.

### Requirements and Caveats

Note that `biblatex-publist` relies on rather advanced features of `biblatex` for some of its own features. These are only available with the `biber` backend of `biblatex`. Hence `biblatex-publist` requires the use of `biber` as well – the `bibtex` backend won't work! Furthermore, the package employs some newer LaTeX 2ε and `latex3` features. So a quite recent LaTeX distribution is required.

Also note that the aim of `biblatex-publist` is to generate publication lists, so it is *not suitable* for normal bibliographies. Specifically, while basic citing works, more advanced forms of citing may break or not produce the expected result. The reason is that `biblatex-publist` needs to perform internal tweaks (e. g., to shortauthor, labelname and numbering) which might bite you with customized cite formats.

# 2 Usage

## 2.1 Standard Usage

The standard way of using the package is to load the style file via

```
\usepackage[style=publist]{biblatex}
\plauthorname[first name][von-part]{surname}
```

`\plauthorname`

The **`\plauthorname`** macro[2] (at least with the mandatory *surname* argument) needs to be given (at least[3]) once. It informs the style file which name(s) it should suppress or highlight in the author/editor list or which entries it should retrieve from a database. In other words, it defines the name(s) of the person(s) whose publications are listed.

By default, the author/editor name(s) will be omitted completely for all publications which are authored or edited only by the specified person(s), as in:

[1] **2012.** Some recent trends in gardening. In: *Gardening Practice* 56, pp. 34–86.

If there are co-authors/co-editors, your name(s) will be filtered out and the collaborators added in parentheses, as in:

---

[2] The macro was named `\omitname` until v. 1.4 of the `biblatex-publist` package. The old macro still works, but is marked as deprecated.

[3] See sec. 2.6 for the case of handling multiple authors and name variants.

[2] **1987** (with John Doe and Mary Hall). Are there new trends in gardening? In: *Gardening Practice* 24, pp. 10–15.

If `plauthorhandling=highlight` is used (see next section), the plauthor(s) will be printed in bold face instead, as in:

[2] Doe, John, **Myself, Me**, and Hall, Mary. **1987**. Are there new trends in gardening? In: *Gardening Practice* 24, pp. 10–15.

Note that `\plauthorname` expects the name constituents as they are recorded in the database (special characters are handled). The option `plauthorfirstinit` (see next section), however, allows you to pass only an initial character instead of a first name.

## 2.2 Additional Options

Currently, the following additional options are provided (next to the options provided by the `biblatex` package itself[4]):

`plauthorname=<surname>`

`plauthorfirstname=<first name>`

`plauthornameprefix=<von-part>`

This is an alternative to the `\plauthorname` macro described in sec. 2.1 to set the publication list author.[5] As opposed to the macro, however, the options do not provide for multiple authors and name variants (see sec. 2.6).

`plauthorhandling[=omit|highlight]` default: *omit*.

By default, the publist author (as defined with `\plauthorname`) is omitted from the author or editor list. If you use the option `plauthorhandling=highlight`, it is highlighted instead (set in bold face by default; see sec. 3.1 how to change that).

`annotations={comma-separated list}` A list of data annotations that should be handled in output. See sec. 2.8 for details.

`nameorder[=family-given|given-family]` default: *family-given*.

By default, the author and editor names with `plauthorhandling=highlight` are output in the order "Lastname, Given Names". To change the order to "Given Names Lastname", pass the option `nameorder=given-family` to `biblatex`.

`fixyear[=true|false]` default: *true*.

By default, the year (or pubstate, if no year is given) is positioned on a fixed slot (at the very beginning with `plauthorhandling=omit`, after the author list with

---

[4] Please refer to the `biblatex` manual [1] for those.

[5] The options were called `omitname`, `omitfirstname` and `omitnameprefix` until v. 1.4 of `biblatex-publist`. The old options still work, but are marked as deprecated.

**`plauthorhandling=highlight`**). If you want to have the year at the position determined by your base style instead, use **`fixyear=false`**. Note that this removes any highlighting of the year, independent of **`hlyear`**. Also note that `biblatex`'s `mergedate` option is not well supported by `biblatex-publist`.

**`hlyear[=true|false]`** default: *true*.[6]

By default, the year (or pubstate, if no year is given) is highlighted (printed in bold face). To prevent this, pass the option **`hlyear=false`** to `biblatex`. The form of highlighting can be customized as well (see sec. 3.1). Note that this has no effect if **`fixyear=false`**.

**`marginyear[=true|false]`** default: *false*.

With this option set to **`true`**, the publication year (or pubstate) will be printed in the margin once a new year starts. The option also has the effect that all marginpars are printed "reversed", i. e. on the left side in one-sided documents (via `\reversemarginpar`).

**`pubstateextra[=true|false]`** default: *false*.

If this is `true`, the extradate marker (a, b etc.) is also appended to pubstates if there are multiple indentical pubstates; e. g., *Forthcoming(a)*, *Forthcoming(b)*.

**`plsorting=<sorting scheme>`** default: *ydt*.

This option works like the **`sorting`** option you know from `biblatex`. It is provided since `biblatex-publist` sets its own sorting schemes (differently depending on **`plauthorhandling`**) and thus overrides any setting made via `biblatex`. You can overwrite those settings here yourself and set an own sorting scheme (see also sec. 8).

**`plnumbering[=global|local|global-descending|local-descending|none]`** default: *global*.[7]

By default, the publication list is numbered in continuous ascending order. If you divide your publication list into sections by means of `\refsections` (as documented in section 6), you will thus get a global numbering over all sections.

If you prefer the numbering to start from 1 at each section instead, set this option to **`local`**.

If you prefer descending rather than ascending enumeration, starting from the total number of entries back to 1 (either globally or within each section), use the **`*-descending`** variants.

If you do not want to have any numbers at all, set this option to **`none`**.

Note that if you use descending enumeration, `biblatex-publist` will have to do a full counting of all printed items first in order to calculate the numbering properly (considering any filtering you might have done). For this calculation,

---

[6] Named **`boldyear`** up to `biblatex-publist` 1.27. The old option is still supported.

[7] Up to `biblatex-publist` 1.27, numbering was controlled by two separate options, **`plnumbered`** and **`reversenumbering`**. These options (and their differently named values) are still supported, but you get a warning if you use them and are encouraged to use the new options.

a specific auxiliary file, *<filename>.bpx*, is used. The package will warn you if more `latex` runs are required. See sec. 2.4 for details.

**`plauthorfirstinit[=true|false]`** default: *false*.

If you set this option to **`true`**, you can (and are supposed to) pass only an initial character as first name value of **`\plauthorname`** (e. g., **`\plauthorname[J]{Doe}`** or **`\plauthorname[J][van]{Doe}`**). In consequence, `biblatex-publist` will consider all entries of the specified given name (and prefix, if specified) whose prename starts with the specified character. This allows you to deal with databases that record entries of your work with abbreviated and full first name (J. Doe and well as John Doe and John Robert Doe) *as well as* works of colleagues with the same surname (maybe your family members), which you will not want to mark as your own's. Of course, the solution does not help if there is an entry with same surname and same first name initial (such as Jane Doe).

**`citinfo[=true|false]`** default: *false*.

If this option is **`true`**, the number of citations, as specified via the `citations` field (see section 2.9.2), will be appended to the entries.

**`jifinfo[=true|false]`** default: *false*.

If this option is **`true`**, the journal impact factor, as specified via the `impactfactor` field (see section 2.9.1), will be appended to the entries.

**`oainfo[=simple|verbose|none]`** default: *none*.

If **`simple`** or **`verbose`** is selected, open-access information, as specified via the `openaccess` field (see section 2.9.4), will be appended to the entries. With **`verbose`**, different OpenAccess strategies ('gold', 'green') will be differentiated in the output.

**`prinfo[=true|false]`** default: *true*.

If this option is **`true`**, peer review information, as specified via the `peerreview` field (see section 2.9.3), will be appended to the entries.

The following option is available if hyperref is loaded:

**`linktitles=[all|doi|url|isbn|issn|none]`** default: *none*.

Turns the title (and subtitle, if available) into a clickable hyperlink to either the DOI, the URL, the ISBN, or the ISSN (the latter two via customizable search provider), if any of these is available.[8] The option **`all`** activates all these targets, **`none`** deactivates them. To enable several targets, multiple values might be passed to the option, embraced and separated by comma (e. g., **`linktitles={doi,isbn,issn}`**). If an entry provides several targets, the first found from the order DOI → URL → ISBN → ISSN is used, so the order passed to the option does not matter, except for **`all`** and **`none`**, which activate or deactivate all at

---

[8] Up to v. 1.27, `biblatex-publist` provided a range of boolean options for this task. These are still supported, but considered deprecated.

their position in the chain (and options that follow might re-enable them, as in **`linktitles={none,url}`**). See section 3.3 for details.

All options listed here should be passed directly to `biblatex` via `\usepackage` when loading `biblatex`, and they apply to the whole document.

`\ExecutePublistOptions`

For most options, it is possible to change them within the document via the macro **`\ExecutePublistOptions`**. It takes a comma-separated list of options as mandatory argument and can be used (repeatedly) in preamble or anywhere in the document body. The executed options apply as of the next bibliography (`\printbibliography`) that follows. Options that *cannot* be changed within a document that way are **`plnumbering`** and **`pubstateextra`**, as these options employ features that can only be set once per document. However, please refer to sec. 2.4 on how to address some frequently requested numbering changes. Also note that with the **`plauthor*`** options, as opposed to repeated **`\plauthorname`** calls, repeated setting of these name options respectively *overwrite*, and not extend, the previous name settings.

## 2.3 Referring to Entries

Even though publication lists usually do not include citations, there might be cases where you need to refer to specific entries within it. Especially with sectioned lists, it is arguably most advisable to refer to the items by their number, as otherwise your readers will have a hard time to find the item you are referring to. Of course, if you do not use global numbering, you should also specify the section the item is in, as in: "see [13] (sec. 4)".

Since `biblatex-publist` plays all sorts of tricks with the numbering (see sec. 2.4), the reference needs to take these into account. The normal citation commands will often produce inadequate or even wrong output particularly in numeric style. To account for these peculiarities, the package ships its own cite style which is used if you load `biblatex-publist` via **`style=publist`** (but not with **`bibstyle=publist`**!). The features described in what follows require this particular cite style.

`\citeitem`

If you want to refer to an item number, use **`\citeitem{<key(s)>}`** within your publication list.[9] This will print the item number in square brackets (like the `\cite` command in numeric citation styles), but as opposed to normal `\cite` it also considers all numbering tweaks you make via the `biblatex-publist` options. Note, however, that with refsections (see sec. 6) **`\citeitem`** only works within a given refsection because the references in a refsection are unknown outside of its scope.

`\citesecitem`

`\citesecitem*`

If you want to refer to an item number from outside a refsection, you can use **`\citesecitem{<key(s)>}`**. This will try to access the number of the given entry from a stored list that is generated by `biblatex-publist`. References of this kind only make sense, of course, with global numbering and if an entry only occurs once in the bibliography, since otherwise they would be too ambiguous. However, for locally numbered lists, there is the starred variant **`\citesecitem*{<key(s)>}`** which will append a reference to the section the item is in. By default, the form of this addition is "(sec. <no.>)", with "sec." being localized (see sec. 3.1 on how to customize this).

---

[9] As with the normal citation commands, the argument might also be a comma-separated list of keys.

All these commands have counterparts which can be used to refer to a range of entries:

`\citeitemrange`
- **`\citeitemrange{<first key>}{<last key>}`**

`\citesecitemrange`
- **`\citesecitemrange{<first key>}{<last key>}`**

`\citesecitemrange*`
- **`\citesecitemrange*{<first key>}{<last key>}`**

`\itemrangesep`
The range will by default be indicated by an en-dash ([12]–[17]), you can change this by redefining the macro **`\itemrangesep`**.

Note that due to the design of refsections, no hyperlinks can be generated for the item number with **`\citesecitem`** (or its starred and range counterparts) and the hyperref package being loaded; the section reference of the starred commands, however, is hyperlinked in this case. If you need hyperlinked item references, consider refsegments rather than refsections and the **`\citeitem`** command (although this will have other drawbacks).

Of course, both **`\citeitem`** and **`\citesecitem`** presuppose that you do not disable numbering, and **`\citesecitem*`** requires numbered section headers.

## 2.4 Numbers Games

Among the peculiarities of publication lists is the greater variety of numbering conventions, as compared with normal bibliographies in academic works. If a numeric citation style is used in the latter, the items are as a rule numbered in ascending order, and each entry has a unique number. The main function of the number here is to provide fast access from a reference to the referred entry, and a continuous ascending enumeration fits this purpose best.

In publication lists, this is only a secondary function of the number, as publication lists are not meant for citing. The main function here is, in fact, *metrical*. In our increasingly metricalized academic cultures, and against the backdrop of the "publish or perish" imperative scholars find themselves confronted with, the sheer number of publications a scholar has is, next to weighted references to them (e. g., the *h-index*), social capital. The numbering of items in publication lists is thus, if you will, a sort of account balance.

Now since publication genres (e. g., journal papers vs. chapters) are usually valued differently ('impact!'), many people prefer to have a more differentiated balance in the sense that the publication lists should swiftly tell the reader how many papers, monographs, chapters, etc., a person has published. Hence those people prefer to number their publications by section rather than continuously.

On the other hand, since publication lists are often ordered reversely chronological, many people also prefer to have a reversely ordered (descending) numbering, either globally over the whole list or, again, by section.

The `biblatex-publist` package accounts for these needs and provides, via the option **`plnumbering`**, five different numbering variants:

1. Global ascending numbers (that continue across sections) [*default*]
2. Global descending numbers [**`plnumbering=global-descending`**]

3. Local ascending numbers (that are reset to 1 in each section) [**`plnumbering=local`**]
4. Local descending numbers [**`plnumbering=local-descending`**]
5. No numbers at all [**`plnumbering=none`**]

Particularly reverse numbering is, technically, a bit of a challenge; `biblatex`, for reasons, does not provide this feature out of the box. In order to do descending numbering one has to know the number of entries in the publication list, or in individual sections, before the first number can get assigned. Since items can be filtered out, for instance via `biblatex`'s **`filter`** or **`keyword`** options, `biblatex` itself only knows this number after all entries have been processed for output. Yet since it only provides for ascending numbers, `biblatex` does not (need to) store this knowledge, and thus, it is not readily available to `biblatex-publist`.

Thus, with **`plnumbering=global-descending`** and **`plnumbering=local-descending`**, `biblatex-publist` will first count all printed entries, globally and per section, write the results in an auxiliary file (*<filename>.bpx*), and then do at least one additional pass in order to fix the numbering. This requires additional `latex` cycles, but you will be given a warning if the numbering is not yet fully resolved.

Now in addition to these mentioned variants, some people still need a more fine-grained control over the numbering. For instance, some scholars want to list both their publications and their conference papers (talks) in the same document, but separately numbered. Both publications and talks might themselves be divided into different sections (*monographs*, *papers*, ...; *keynotes*, *invited talks*, ...) which, however, should be numbered continuously within their category. So **`plnumbering=local`** alone does not produce the desired result.

`plnumgroup`

For this case, `biblatex-publist` provides two alternative and slightly different features: First, an environment **`\begin{plnumgroup}`**...**`\end{plnumgroup}`**. All sections you embrace by this in the context of **`plnumbering=local`** are numbered continuously, so resetting is suspended within this environment. This works both with ascending and descending enumeration.

`\setplnum`

In the ascending case only, the macro **`\setplnum`** is useful as a (more flexible) alternative. It allows you to set the first label number of the next bibliography section that follows to a specific (arbitrary) value. All following numbers ascend or descend from this new value. The adjustment applies to all following bibliography sections with continuous numbering, i. e., if a **`global`** numbering scheme is used. With **`local`** numbering, by contrast, following sections continue to be reset. In all cases, the macro can be used repeatedly for re-adjustments in forthcoming sections.

So in the use case mentioned above, inserting `\setplnum{1}` before the first section that lists conference papers, together with **`plnumbering=global`** (the default), will provide the desired result (provided that ascending numbering is wished for). The desired result will also be achieved by embracing all publication-related sections in a **`plnumgroup`** environment and and all conference-paper-related sections in another one, while using **`plnumbering=local`** or **`plnumbering=local-descending`**. This second approach, as opposed to the former, also works for descending enumeration.

`\shiftplnum`

The macro **`\shiftplnum`**, by contrast with **`\setplnum`**, provides a way to adjust the subsequent numbering in a *relative* way. It takes a positive or negative integer value that determines how much and in which direction the adjustment shall take place (e. g., `\shiftpblnum{2}` or `\shiftplnum{-1}`). This might be useful if you insert items in the list manually or in some very specific settings. Again, this macro can be used repeatedly and applies to all subsequent items as of the next section that follows. As opposed to **`\setplnum`**, the shifting also persists with **`local`** numbering schemes.

For the numbering output by `\citeitem`, analogous shifting can be done via the macro **`\shiftciteitem`**.

`\shiftciteitem`

## 2.5 Statistics

`\GetTotalCount`

The macro **`\GetTotalCount`** outputs the total number of entries in your publication list.

`\GetSectionCount`

If you want to output the number of publications per section (when employing `\refsection`), you can use the macro **`\GetSectionCount`** which takes one optional argument, the section number or a comma-separated list of section numbers. For instance,

```
\GetSectionCount[1]
```

outputs the number of items in the first section, whereas `\GetSectionCount[1,2]` outputs the sum of the numbers of items in the first and second section. If the optional argument is not given, the statistic for the current section is output (this requires the macro to be within a `\refsection`).

Note that both values might require additional LaTeX runs to get properly set. You will get a package warning if additional runs are needed.

## 2.6 Handling Multiple Authors and/or Name Variants

If multiple **`\plauthorname`** specifications have been entered (or a **`\plauthorname`** specification in addition to a specification via the options **`plauthorname`**, **`plauthorname`** and **`plauthornameprefix`**), all of them will be considered.

Thus it is possible to highlight multiple authors in the publication list (for instance to mark contributions of a research team):

```
\usepackage[style=publist,plauthorhandling=highlight]{biblatex}
\plauthorname[Cristiano]{Ronaldo}
\plauthorname[\'Angel][Di]{Mar\'ia}
\plauthorname{Neymar}
```

Multiple specifications can also be used to deal with name variants:

```
\plauthorname[Bill]{Gates}
\plauthorname[William]{Gates}
\plauthorname[William Henry]{Gates}
\plauthorname[William H.]{Gates}
```

Since the latter is also useful with **`plauthorhandling=omit`**, this mode also considers multiple specifications. By entering something such as the former, it is even possible to omit more than one and different authors from the entries (and `biblatex-publist` will take care of the change in the author separation this involves; think of final *and* vs. *comma*, which have to be adjusted accordingly if names are omitted). However, it does not strike me sensible to do so (in other words, if you need to deal with a *team* of authors, you should really consider to use **`plauthorhandling=highlight`**).

Note that multiple specifications also affect filtering (see sec. 7), i. e., the **`mine`** filter selects entries authored or edited by any and all specified persons.

## 2.7 Truncation of Name Lists

Truncation of name lists via the `maxnames` and `minnames` `biblatex` options is supported. However, it works a bit differently than normal truncation, since the publication list authors have to be taken care of specifically.

With **`plauthorhandling=omit`**, the `maxnames` value specifies how many co-authors are added in parenthesis (the omitted author name is not counted here). If the treshold is reached, *et al.* (or the corresponding localized string) is appended (and the list truncated to the `minnames` value, `1` by default). So you get something like:

> [3] **2020** (with John Doe et al.). What's up in gardening? In: *Gardening Practice* 44, pp. 1–7.

With **`plauthorhandling=highlight`**, `biblatex-publist` outputs all publist authors, even if the `maxnames` treshold has been reached. However, other authors (beyond `minnames`) are omitted. If they come before a publist author, this is indicated by [...], if authors follow after all publication list authors, *et al.* is appended, as in:

> [3] Doe, John, [...], **Myself, Me**, et al., **2020**. What's up in gardening? In: *Gardening Practice* 44, pp. 1–7.

`\plnameomission` The omission indicator, **`\plnameomission`**, can be redefined. The default definition is:

```
\newcommand*\plnameomission{\bibellipsis\addcomma\addspace}
```

## 2.8 Indicating Author Roles

If you want to indicate author roles (e. g., signify who is corresponding or main author of a paper), you need to add this information to the respective entries, but you do not want to clutter the author data. With `biblatex`, separating such extra information from the actual data is well supported via the data annotation feature, as in:

```
author    = {John Doe and Mary Hall and Me Myself},
author+an = {1=corresponding; 3=main},
```

where the number (key) refers to the respective name in the list, and the value to the annotation category which you can name as you wish. Multiple annotations to a single person are possible if you pass multiple comma-separated categories to one key, as in `1=corresponding,main` (please refer to the `biblatex` manual [1] for details).

To handle such annotations, however, the style file needs to know what to do with what kind of annotation. Otherwise annotations are just ignored.

For basic usage, `biblatex-publist` provides some support for such annotations. If you pass the annotation categories you have used in the `*.bib` file as a comma-separated list to the **`annotations`** option, `biblatex-publist` will go through these annotations and for each, it will print a superscripted symbol in front of the author family name. By default this uses a list similar[10] to the one you know from symbolic (`\thanks`) footnotes (in order of the passed category: *, ‡, §, ¶, ‖, †, **, ‡‡, or ††). Nine different annotations are supported this way.

With the example given above and **`annotations={main,correponding}`**, for instance, you would get

[4] ‡Doe, John, Hall, Mary, and ***Myself, Me*** …

Of course, each category always gets the same symbol within a publication list, so you can add a caption to explain those. For easy access to the symbols, you can use

`\GetAnnotSymbol`

**`\GetAnnotSymbol{category}`**. This prints the respective symbol in formatted way (superscripted with default settings). A starred version of the command prints it without such formatting.

If you want a different list of symbols, or need more, you can change it by means of

`\SetAnnotSymbols`

the macro **`\SetAnnotSymbols{comma-separated list of symbols/macros}`**. This will overwrite the default list.

You can also change the position of the symbol. The default definition is

```
\renewcommand*{\mkbibnamefamily}[1]{\bplannot #1}
```

with **`\bplannot`** being the annotation. You can move it after the name, or to another name part (e.g., `\mkbibnamegiven`).

To change the formatting, you can redefine

```
\newcommand*\mkbplannot[1]{\textsuperscript{#1}}
```

If you need a different markup (such as coloring or underlining) you need to set up your own annotation handling as documented in the `biblatex` manual.

With **`plauthorhandling=omit`**, the annotations of the omitted publist author(s) are appended to the entry (as an annotation and with the specific entry-related information discussed in the next section). If a `\bibstring` `annot:<category>` exists (predefined keys are `annot:main` and `annot:corresponding`), the respective localized string is output, otherwise the verbatim category.

So, again with the example given above, you would get:

[4] … (with John ‡Doe and Mary Hall) … [Main auth.].

## 2.9 Adding Specific Entry-Related Information

Publication lists usually contain some information which is not commonly given in normal bibliographies, information that is supposed to help ranking the author (bibliometrics; the peer review procedures publications have been subject to) or their publication strategy (e. g., OpenAcess). To allow for this, `biblatex-publist` provides some

---

[10] If have moved the single dagger (†) to a later place, since some people associate this with the ‘deceased’ indicator.

extra features which will be described in what follows. Please also refer to sec. 3.1 for some customization possibilities.

### 2.9.1 Journal Impact Factor

Although the meaningfulness of this measure has been heavily challenged, scholars in many fields are supposed to give the impact factor (citation ratio) of the journals they have published in. Thus, `biblatex-publist` provides a specific database field, `impactfactor`, in which you can store this value and maybe specify the metrics used.

If you publish in a journal repeatedly, it might make sense to store the journal name and impact factor in a specific `@xdata` subset (please refer to the `biblatex` documentation for details on such subsets).

If the option **`jifinfo`** is true, then `biblatex-publist` will append the respective information to the entries. A specific localizable bibkey is used for this purpose (see section 4).

### 2.9.2 Number of Citations

Another common quantification of 'impact' is the number of citations of a publication (by other publications). To track this, `biblatex-publist` provides a specific database field, `citations`.

If the option **`citinfo`** is true, then `biblatex-publist` will append the respective information to the entries. A specific localizable bibkey is used for this purpose (see section 4).

### 2.9.3 Peer Review

It is common, and sometimes mandatory, to add information to publication list entries indicating if and how listed work has been subject to peer review. To this end, `biblatex-publist` provides a specific database field, `peerreview`, in which you can use the following shorthands:

- `sb` (= single blind peer review)
- `db` (= double blind peer review)
- `op` (= open peer review)
- `co` (= collaborative peer review)
- `cc` (= cascading peer review)
- `tp` (= third-party peer review)
- `pp` (= post-publication peer review)
- `no` (= no peer review)

If the option **`prinfo`** is true, then `biblatex-publist` will append the respective information to the entries. A specific localizable bibkey is used for this purpose (see section 4). Instead of the shorthands, you can also enter arbitrary text to the `peerreview` field, which is then output as is (and not localized).

### 2.9.4 OpenAccess Strategies

Some people also want, or need, to indicate whether and which (some of) their publications have been published ‘OpenAccess’, that is, free of charges and other access barriers. To this end, `biblatex-publist` provides a specific database field, `openaccess`, in which you can use the following shorthands:

- `true` – Indicates that this publication has been published OpenAccess, no matter which strategy
- `gold` – Indicates that this publication has been published OpenAccess from the beginning (‘gold’ strategy)
- `green` – Indicates that a secondary version of this publication has been made openly accessible on a public document server (repository) (‘green’ strategy)

If you use the option **`oainfo=verbose`** or **`oainfo=simple`**, then `biblatex-publist` will append the respective information to the entries. In the former case, ‘gold’ and ‘green’ publications will be differentiated, in the latter, this differentiation is not done. Specific localizable bibkeys are used for this purpose (see section 4). Instead of the shorthands, you can also enter arbitrary text to the `openaccess` field, which is then output as is (and not localized).

# 3 Customization

## 3.1 Auxiliary Macros and Lengths

`\plmarginyear` The appearance of the *marginyear* is controlled by the **`\plmarginyear`** macro, which has the following default definition:

```
\providecommand*\plmarginyear[1]{%
  \raggedleft\small\textbf{#1}%
}
```

If you want to change the appearance, just redefine this macro via `\renewcommand*`.

`\plyearhl` The highlighting of the year within the entry, if **`hlyear=false`** has not been set, is controlled by the **`\plyearhl`** macro, which has the following default definition:

```
\providecommand*\plyearhl[1]{%
        \mkbibbold{#1}%
}
```

If you need another form of highlighting, redefine this macro via `\renewcommand*`.

`\plauthorhl` The highlighting of the publication list author, if **`plauthorhandling=highlight`** has been set, is controlled by the **`\plauthorhl`** macro, which has the following default definition:

```
\providecommand*\plauthorhl[1]{%
        \mkbibbold{#1}%
}
```

If you need another form of highlighting, redefine this macro via `\renewcommand*`.

`\bplopenoa`
`\bplcloseoa`

The embracing of authors with **`plauthorhandlung=omit`** (by default: in parentheses) is controlled by the two macros **`\bplopenoa`** and **`\bplcloseoa`** which are defined by default as:

```
\providecommand*{\bplopenoa}{\bibopenparen}
\providecommand*{\bplcloseoa}{\bibcloseparen}
```

You can remove the parentheses by redefining these macros with empty arguments, respectively, or add other delimiters instead. Note that if you use `\bibopenbracket` in `\bplopenoa`, you must also use `\bibclosebracket` in `\bplcloseoa` to balance the delimitation (as with `\bibopenparen` and `\bibcloseparen`).

`plextrainfo`

The embracing of extra information (brackets by default) can be customized by changing

```
\DeclareFieldFormat{plextrainfo}{\mkbibbrackets{#1}}
```

`\plextrainfosep`

The separator between extra information (peer review, OpenAccess) can be redefined. The default definition is:

```
\newcommand*\plextrainfosep{\addsemicolon\addspace}
```

`\secitemsecref`

The section reference with **`\citesecitem*`** is determined by **`\secitemsecref`**, which has the following default definition:

```
\newcommand*\secitemsecref[1]{%
        \mkbibparens{\bibstring{section}\addnbspace\ref{refsection:#1}}%
}
```

Redefine this macro if you need a different output.

`extralabelnumberwidth`

The indendation of the bibliographic entries (lines > 1) can be adjusted by setting the length **`extralabelnumberwidth`** via `\setlength` (default is `0pt`). This might be needed for long bibliographies (> 99 entries) in order to adjust to the extra space the item number needs.

## 3.2 Using a Different Base Style

By default, `biblatex-publist` loads `biblatex`'s *authoryear* style, and it has been written to work with that style. However, it is possible to try a diffent base style, if *authoryear* does not fit your needs.

In order to do so, enter the following *before* loading `biblatex`:

```
\newcommand*\publistbasestyle{<stylename>}
```

where <stylename> is the name of the biblatex bibliography style (*bbx*) you want to use, without the *bbx* entension (e. g., `\newcommand*\publistbasestyle{mla}`).

Note, though, that there is (and can be) no guarantee that `biblatex-publist` will work with all styles, although it has been successfully tested with several. Be prepared to bump into LaTeX errors and carefully check the output for correctness if you try a different base style.

Note, further, that the order of author's and editor's given and family names is hardcoded in `biblatex-publist` due to the complex omission/highlighting mechanism. This might differ from what you expect with specific base styles. To change the order, use the package option **`nameorder`** (see sec. 2.2).

The same applies to the position of the year, which is hardcoded to a specific position. This can be opted out by the package option **`fixyear=false`** (see sec. 2.2). With this option, however, you lose the possibility to highlight the year.

### 3.3 Clickable Titles

With the option **`linktitles`** (see sec. 2.2), titles and subtitles are turned into clickable hyperlinks if the hyperref package is loaded, and the respective data is there, i. e., if either the DOI field, the URL field, the ISBN field or the ISSN field is defined for the given entry (checked in this order if multiple of these targets are enabled or **`linktitles=all`** is used).

With URL and DOI, direct links are created. With ISBN or ISSN, a link to a search provider is created instead (worldcat by default). The search provider can be customized by redefining the following macros:

```
\newcommand*\plisbnlink[1]{https://www.worldcat.org/search?qt=worldcat_org_all&q=#1}
\newcommand*\plissnlink[1]{https://www.worldcat.org/search?qt=worldcat_org_all&q=#1}
```

where `#1` is a placeholder for the ISBN or ISSN, respectively.

Note that the **`linktitles`** option does not have any affect on whether URLs, DOIs, ISBNs and ISSNs are printed, so you might get redundant output. To control (e. g., omit) them, use the **`url`**, **`doi`** and **`isbn`** biblatex options.

## 4 Localization

Since the package draws on `biblatex`, it supports localization. The following additional localization keys (`bibstrings`) are added by the package, in verbose and abbreviated form (depending on your setting of the `biblatex` option **`abbreviate`**):

**annot:corresponding** the expression "corresponding author" (short form: "corr. auth.").

**annot:main** the expression "main author" (short form: "main auth.").

**citations** the expression "citations" (short form: "cit.").

**impactfactor** the expression "impact factor" (short form: "JIF").

**openaccess:true** the expression "OpenAccess" (short form: "OA").

**openaccess:gold** the phrase "OpenAccess (gold)" (short form: "OA (gold)").

**openaccess:green** the phrase "OpenAccess (green)" (short form: "OA (green)").

**parttranslationof** the phrase "partial translation of" (short form: "part. trans. of") for entries referring to partially translated work via `biblatex`'s 'related entries' feature (see sec. 5.2).

**peerreview:sb** the phrase “single blind peer review” (short form: “single blind peer rev.”).

**peerreview:db** the phrase “double blind peer review” (short form: “double blind peer rev.”).

**peerreview:op** the phrase “open peer review” (short form: “open peer rev”).

**peerreview:co** the phrase “collaborative peer review” (short form: “collab. peer rev.”).

**peerreview:cc** the phrase “cascading peer review” (short form: “casc. peer rev.”).

**peerreview:tp** the phrase “third-party peer review” (short form: “3rd-party peer rev.”).

**peerreview:pp** the phrase “post-publication peer review” (short form: “post-pub. peer rev.”).

**peerreview:no** the phrase “no peer review” (short form: “no peer rev.”).

**with** the preposition “with” that precedes the list of co-authors by default (i. e., with **`plauthorhandling=omit`**) (short and verbose forms identical).

Localization is provided in dedicated language definition files which follow the naming scheme, `<language>-publist.lbx`. The package currently ships such files for the following languages and their varieties:

- English (`american`, `australian`, `british`, `canadian`, `english`, `newzealand`, `UKenglish`, `USenglish`)
- French (`french`)
- German (`austrian`, `german`, `naustrian`, `ngerman`, `nswissgerman`, `swissgerman`)

If you want to add support for other languages, you can simply copy one of the existing lbx files, name it `<yourlanguage>-publist.lbx` and adapt the localization keys.

If you are confident enough, please send me your file so I can consider adding it to the package.

# 5 Further Extensions

The following extensions of standard `biblatex` features are provided.

## 5.1 Review Bibliography Type

Although a *review* entry type is provided by `biblatex`, this type is treated as an alias for *article*. The `biblatex-publist` package uses this entry type for a specific purpose: Foreign reviews of your own work. It therefore defines a new bibliography environment *reviews* with a specific look (particularly as far as the author names are concerned) and its own numbering; furthermore, it redefines the *review* bibliography driver. The purpose of this is that you can add other people’s reviews of your work to your publication list, while these titles are clearly marked and do not interfere with the overall numbering (see sec. 6 for an example).

### 5.2 Partial Translations

A new “related entry” type *parttranslationof* is provided. This is an addition to the *translationof* related entry type `biblatex` itself provides. Please refer to the `biblatex` manual [1] on what “related entries” are and how to use them.

## 6 An Example

Publication lists are usually categorized by genre (monographs, articles, book chapters, etc.). For this task, the use of `refsections` (see [1, sec 3.7.4] for details) is suggested. Other possibilities were not tested extensively and might fail (in particular as far as the numbering of the items is concerned).

The suggested procedure is to maintain separate bib files for each category, say *mymonographs.bib*, *myarticles.bib*, *myproceedings.bib*.[11] Then a typical file would look like example 1 (p. 19). If you want to add other people’s reviews of your work, add a section as shown in example 2.

Example 2: Adding foreign reviews

```
\subsubsection*{Reviews of my thesis}
\newrefsection[mythesis-reviews]
\renewcommand\bibfont{\small}
\nocite{*}
\printbibliography[heading=none,env=reviews]
```

Note that the `\printbibliography` option **`env=reviews`** is crucial if you want to use the specifics `biblatex-publist` defines for reviews (see sec. 5.1).

## 7 Filtering

### 7.1 Filter out own work

If you have a bibliographic database consisting not only of your own publications, you can extract yours with the bibliography filter **`mine`**, which has to be passed to `\printbibliography`, as in:

Example 3: Using a bibliography filter

```
\nocite{*}
\printbibliography[heading=none,filter=mine]
```

This will effectively print only publications which have been authored or edited by the person(s) specified as via **`\plauthorname`** (or the corresponding option).

You can also use other filter possibilities provided by `biblatex`, such as filtering by type or by keyword. So if you want to extract all of your articles from a larger database with entries of diverse type and authors, specify:

---

[11] But see sec. 7 for an alternative.

Example 1: Typical document

```
\documentclass{article}
\usepackage[T1]{fontenc}

\usepackage{csquotes}% not required, but recommended
\usepackage[style=publist]{biblatex}
\plauthorname[John]{Doe}

\addbibresource{mymonographs.bib}
\addbibresource{myarticles.bib}
\addbibresource{myproceedings.bib}

\begin{document}

\title{John Doe’s publications}
\date{\today}
\maketitle

\section{Monographs}
\newrefsection[mymonographs]
\nocite{*}
\printbibliography[heading=none]

\section{Proceedings}
\newrefsection[myproceedings]
\nocite{*}
\printbibliography[heading=none]

\section{Articles}
\newrefsection[myarticles]
\nocite{*}
\printbibliography[heading=none]

\end{document}
```

```
\printbibliography[heading=none,filter=mine,type=article]
```

Note that this method sometimes requires several reruns of `latex` to fix the numbering.

### 7.2 Filter on publication or peer-review status

Sometimes you might also want to omit some publications from the list which are not yet published or not yet accepted (e.g., because they are under blind review and you do not want to reveal your identity yet). To this end, `biblatex-publist` features some **`bibchecks`** (for the concept, please refer to [1, sec. 3.8.2]). Bibchecks can be activated by passing **`check=<bibcheck>`** to the optional argument of `\printbibliography` (see example 4).

The following bibchecks are available to this end:

- **`nosubmitted`**: omits all entries with pubstate **`submitted`**.
- **`noprepared`**: omits all entries with pubstate **`inpreparation`**.
- **`onlypublished`**: omits all unpublished entries (i.e., entries that have a pubstate) with the exception of **`prepublished`** entries.
- **`onlyaccepted`**: omits all unpublished entries (i.e., entries that have a pubstate) with the exception of **`prepublished`** and **`forthcoming`** (and thus accepted) entries.

The bibchecks below let you filter out publications depending on their peer-review setting. This is helpful if you want to differentiate peer-reviewed from non-peer-reviewed work in your publication list:

- **`onlypr`**: include only peer-reviewed entries (displays only titles which have the `peerreview` field set and do not have it set to `no`).
- **`nopr`**: exclude peer-reviewed entries (omits all entries which have the `peerreview` field set, except for those which have it set to `no`).

Several of these bibchecks might be used in a row, e.g.:

Example 4: Using bibchecks

```
\printbibliography[heading=none,filter=mine,check=nosubmitted,check=noprepared]
```

## 8 Sorting

### 8.1 Sorting Publication Lists

The sorting conventions of publication lists differ from those of normal bibliographies. Publication lists are usually not sorted by author name, the prime criterion of normal bibliographies, but rather chronologically (usually *descending* from the newest through the oldest publication). How to sub-sort within a year depends on the handling of author names. If you display all authors and only highlight your own (via **`plauthorhandling=highlight`**), it probably makes sense to sub-sort first by author name, and then by title. If you omit your own name and just mention your co-authors (the default), it makes more sense to sub-sort by title right away, without taking the author names into account.

To account for these needs, `biblatex-publist` adds some sorting options on top of those that come with `biblatex` itself.

## 8.2 Sorting Templates

The sorting of items is done via `biblatex`'s sorting mechanism, via so called *sorting templates* (please refer to the `biblatex` manual for details).

By default, `biblatex-publist` uses an own template, **`ydt`**, which sorts hierarchically by **y**ear (**d**escending) and **t**itle (alphabetically ascending), ignoring author names. This default is used since author name sorting does not make much sense at least in the default configuration, where the own name is omitted and the list of co-authors is presented in a particular way. If you use **`plauthorhandling=highlight`**, however, the default changes to **`ydnt`** (a template provided by `biblatex` itself) which sub-sorts by author names (alphabetically ascending) before sub-sorting by title.

In addition to this default template, `biblatex-publist` provides some sorting templates that account for the full date (rather than just the year). This is especially useful for sorting talks, since those usually do not only have a year, but a full date (day, month and year). The following templates, with and without author sorting, are provided:

- **`ddt`**: Sort by full **d**ate (**d**escending) and **t**itle (ascending).
- **`ddnt`**: Sort by full **d**ate (**d**escending), author **n**ame and **t**itle (both ascending).
- **`dt`**: Sort by full **d**ate and **t**itle (all ascending).
- **`dnt`**: Sort by full **d**ate, author **n**ame and **t**itle (all ascending).
- **`ydmdt`**: Sort by **y**ear (**d**escending), **m**onth, **d**ay and **t**itle (all ascending).
- **`ydmdnt`**: Sort by **y**ear (**d**escending), **m**onth, **d**ay, author **n**ame and **t**itle (all ascending).

In order to use any of these, or another sorting template provided by `biblatex` use `biblatex-publist`'s **`plsorting`** option, which can be passed globally as a `biblatex` option or locally by means of **`\ExecutePublistOptions`**. Alternatively, you can also use a `\newrefcontext` macro with the option `sorting=<template>`. So, to sort your talks in descending order by full date in your CV, you would use either

```
\usepackage[style=publist,plsorting=ddt]{biblatex}
```

or

```
\ExecutePublistOptions{plsorting=ddt}
\printbibliography[heading=none]
```

or

```
\newrefcontext[sorting=ddt]
\printbibliography[heading=none]
\endrefcontext
```

Note that if you use refsections, the third approach will only apply to the current refsection (`\ExecutePublistOptions` will apply to all subsequent refsections until further change).

## 9 Revision Log

**V. 2.13 (2024/08/21):**

- Add basic support for author annotations (see sec. 2.8).
- Add support for information about number of citations (see sec. 2.9.2).
- Add support for `\citesecitem` and `\citesecitem*` in descending order.
- Fix `\citeitem` in first refsection.

**V. 2.12 (2024/07/16):**

- Externalize localization to `*.lbx` files.
- Add `\citeitemrange` and `\citesecitemrange` commands (see sec. 2.3).
- Fix error with section count on first run.

**V. 2.11 (2024/07/02):**

- Add `\citesecitem` and `\citesecitem*` commands (see sec. 2.3).
- Support multiple sections in `\GetSectionCount`.

**V. 2.10 (2024/06/17):**

- Add bibchecks to show only/no peer-reviewed items (see sec. 7.2).
- Fix bibchecks to work with braced and non-braced values.

**V. 2.9 (2024/05/09):**

- Add bibchecks to exclude (specific kinds of) unpublished work (see sec. 7.2).

**V. 2.8 (2024/03/08):**

- Fix delimiter with plauthor second in a truncated author list.

**V. 2.7 (2023/10/07):**

- Fix ascending numbering without refsections.

**V. 2.6 (2023/07/01):**

- Fix marginyear output with **`fixyear=false`**.

**V. 2.5 (2023/06/08):**

- New macros **`\GetSectionCount[<int>]`** and **`\GetTotalCount`** to output statistics (see sec. 2.5).
- Fix issues with **`fixyear=false`**.

**V. 2.4 (2023/05/18):**

- New option **`fixyear`** to opt-out special year handling and positioning.
- Make delimiters of author list with **`plauthorhandling=omit`** configurable (see sec. 3.1).

**V. 2.3 (2023/03/18):**

- Change of sorting by means of **`\ExecutePublistOptions`** (via **`plsorting`** or **`plauthorhandling`**) now works for all subsequent refsections.

**V. 2.2 (2023/03/01):**

- Fix **`plsorting`** with custom sorting schemes.

**V. 2.1 (2022/12/03):**

- Honor `\mkbibname*`.
- Fix reverse numbering without refsections.

**V. 2.0 (2022/10/24):**

- Provide **`\ExecutePublistOptions`** to change (most) `biblatex-publist` options on the fly (see sec. 2.2).
- Properly support reverse (descending) numbering even if filtering is done.
- Add **`plnumgroup`** environment to temporarily suspend **`plnumbering=local`** (see sec. 2.4).
- Add new macro**`\setplnum`** (see sec. 2.4).
- Rename macro **`\shiftbplnum`** to **`\shiftplnum`** (old macro kept for compatibility).
- Fix macro **`\shiftplnum`** with ascending numbering.
- Introduce one option (**`plnumbering`**) that replaces the previous two separated options (**`plnumbered`** + **`reversenumbering`**). Old options are kept for compatibility.

- Introduce one multichoice option (**`linktitles`**) that replaces the previous options (**`linktitleall`**, **`linktitledoi`**, **`linktitleurl`**, **`linktitleisbn`**, **`linktitleissn`**). Old options are kept for compatibility.
- Rename option **`boldyear`** to **`hlyear`**. Old option kept for compatibility.
- Allow customization of the highlighting set with **`hlyear`**. See sec. 3.1.
- Introduce **`plsorting`** option.
- Add support for OpenAccess information (see sec. 2.9.4).
- Add support for journal impact factor (see sec. 2.9.1).

**V. 1.27 (2022-10-03):**

- Fix once more plauthor check with non-ASCII chars and macros.

**V. 1.26 (2022-01-05):**

- Provide means to add peer review information (see sec. 2.9.3).
- Add a data model (`*.dbx`) file.
- Fix **`\textcite`** output at least for the standard cases.
- Do not highlight/omit author/editor names in related entries.
- Rename `bpl:review:*` macros to `bpl:plain:*` and add `bpl:plain:editor`.

**V. 1.25 (2021-12-14):**

- More robustification with names consisting of non-ASCII chars.
- With **`pubstateextra`**, differentiate between pubstates.

**V. 1.24 (2021-12-11):**

- Robustify handling of names with non-ASCII chars.

**V. 1.23 (2021-09-01):**

- Fix omission of publist author after related field.

**V. 1.22 (2021-06-14):**

- Add option `pubstateextra`. See sec. 2.2.
- Use **`\revsdnamepunct`** rather than hardcoded comma.

**V. 1.21 (2020-09-21):**

- Add option `reversenumbering`. See sec. 2.2.
- Add **`\citeitem`** command. See sec. 2.1.
- Add **`\shiftbplnum`** and **`\shiftciteitem`** helper macros for manual adjustment of numbering. See sec. 2.4.
- Properly sort `prepublished` pubstate type.

**V. 1.20 (2020-09-15):**

- Do not output *(with <authors>)* if no publist author is among the authors.

**V. 1.19 (2020-08-21):**

- Fix parsing of names with initials.
- Fix output of *et al.* in `byeditor` lists.
- Do not omit names in related entries.

**V. 1.18 (2020-07-31):**

- Support name truncation via `maxnames`. See sec. 2.7.
- Fix **`filter=mine`** with author lists longer than `maxnames`.
- Fix double editor with `@periodical` type.
- Use `\editortypedelim`.

**V. 1.17 (2020-07-10):**

- Add options to get clickable titles. See sec. 3.3.

**V. 1.16 (2019-04-16):**

- Major code cleanup.

**V. 1.15 (2019-02-22):**

- Add support for omitting multiple authors. See sec. 2.6.
- Fix documentation issues.

**V. 1.14 (2019-02-21):**

- Add support for highlighting multiple authors. See sec. 2.6.
- Fix handling of non-ASCII names.
- Use `\DeclareStyleSourcemap` rather that `\DeclareSourcemap`.
- Update sorting documentation in the wake of `biblatex` changes.

**V. 1.13 (2018-11-30):**

- Introduce new sorting templates that ignore names. See sec. 8.2.
- **Change of output!** Use **`ydt`** template by default. See sec. 8.2.
- Assign extralabel independent of author group with **`plauthorhandling=omit`**.

**V. 1.12 (2018-11-25):**

- Switch name parsing toggles globally (fixes regression with `biblatex` 3.12).
- Account for omitted author when adding `\finalnamedelim`.
- Fix issue with initial dot in **`nameorder=family-given`**.

- Add option **`plauthorfirstinit`** that allows for specifying initials in first names of **`\plauthorname`**. See sec. 2.2.

**V. 1.11 (2018-09-01):**

- Fix **`marginyear=true`** with **`labeldateparts=false`**.
- Fix problem with empty parentheses in article with standard base style and with **`labeldateparts=false`**.

**V. 1.10 (2018-04-08):**

- Extend option **`plnumbered`** with **`plnumbered=reset`**. This allows to restart the numbering of the publication list items at **`\refsection`**s.
- Documentation improvements.

**V. 1.9 (2018-03-01):**

- New option **`plnumbered`** that allows to omit the numbering of the publication list items
- Documentation improvements.

**V. 1.8 (2017-11-14):**

- Adapt to `biblatex` 3.8. This version is now required.
- Rename some macros, using pseudo-namespaces:
  - `date:makedate` ⇒ `bpl:date:makedate`
  - `date:labelyear+extrayear` ⇒ `bpl:date:labeldate+extradate`
  - `marginyear` ⇒ `bpl:marginyear`
  - `rauthor` ⇒ `bpl:review:author`
  - `rauthor/label` ⇒ `bpl:review:author/label`
  - `year+labelyear` ⇒ `bpl:year+labelyear`

**V. 1.7 (2017-04-12):**

- Output marginyear before the author list. This prevents it from being vertically shifted in case of long author lists.

**V. 1.6 (2017-04-02):**

- New option **`nameorder`** that allows to change the ordering of author and editor name (**`given-family`** vs. **`family-given`** [= default]).
- Use proper name delimiters also for bookauthor.

**V. 1.5 (2017-02-28):**

- Fix extra *and* in name list with **`plauthorhandling=highlight`**.
- Whitespace fix with **`plauthorhandling=highlight`**.

- Use proper name delimiters.

**V. 1.4 (2017-02-12):**

- New option **`plauthorhandling`** that defines how the publist author is handled in the publication list (possible values: **`omit`** [= default], **`highlight`**).
- New command **`\plauthorhl`** that determines the aforementioned highlighting.
- Rename **`\omitname`** to **`\plauthorname`** (the old macro is still functional, but marked as deprecated).
- Rename **`omit*`** options to **`plauthor*`** (the old options are still functional, but marked as deprecated).
- Assure the margin text always starts uppercased (relevant for pubstates).
- Minor corrections to the manual.

**V. 1.3 (2016-08-06):**

- It is now possible to change the base style that is used by `biblatex-publist`. See sec. 3.2.
- Proper sorting of pubstates.
- Add possibility to increase the indentation of items (by means of the length **`extralabelnumberwidth`**). See sec. 3.1.
- Use `pagetracker=true` instead of `pagetracker=spread` by default (avoids warning, no change in functionality).

**V. 1.2 (2016-05-12):**

- Accomodate to the backwards-incompatible changes of `biblatex` 3.4 (`prefixnumber` ⇒ `labelprefix`, `\ifempty` ⇒ `\ifdefvoid`). This version of `biblatex` is now required.

**V. 1.1 (2016-03-09):**

- Adapt to the `\Declare*Name` changes of `biblatex` 3.3. Since `biblatex` 3.3 introduced backwards-incompatible changes that affect `biblatex-publist`, this version of `biblatex` is now required.

**V. 1.0 (2015-01-04):**

- Add portmanteau *.cbx file to allow loading `biblatex-publist` also via the `style` option (next to `bibstyle`).

**V. 0.9 (2014-03-13):**

- Fix problem with multi-token names.
- Support name prefix in **`\omitname`**.

- Support pubstate.

**V. 0.8 (2013-08-16):**

- Add custom sorting schemes **`ddnt`**, **`ydmdnt`** and **`dnt`** (see sec. 8).
- Revise the documentation.

**V. 0.7 (2013-07-25):**

- Support full dates.

**V. 0.6 (2013-07-21):**

- Fix numbering with recent `biblatex` versions.

**V. 0.5 (2013-05-03):**

- Fix numbering if `\printbibliography` is used multiple times within the same or without any `refsection`.

**V. 0.4 (2012-10-30):**

- More robust name parsing (especially for names with non-ASCII characters encoded with LaTeX macros). The code was kindly suggested by Enrico Gregorio.[12]
- Add **`\omitname`** command (see sec. 2.1).
- Support **`firstinits`** option.

**V. 0.3 (2012-10-23):**

- Bug fix: Add missing "and" if omitted name was last minus one.
- Bug fix: Fix output with "et al." if omitted name is first and *liststop* is 1.
- Set **`maxnames`** default to 4.
- Add filter possibility (see sec. 7).
- Add French localization.
- Some corrections to the manual.

**V. 0.2 (2012-10-21):** Initial release to CTAN.

## 10 Credits

Thanks go to Enrico Gregorio (egreg on *tex.stackexchange.com*) for helping me with correct name parsing (actually, the code the package uses is completely his), user gusbrs on *tex.stackexchange.com*, Marko Budišić, David Carlisle, Ulrike Fischer, Clea F. Rees, Yannick Kalff, Moritz Wemheuer and many other users for testing, bug reports and suggestions, Nicolas Markey for *publist.bst* and of course Philipp Lehman and the current `biblatex` team (Philipp Kime, Moritz Wemheuer, Audrey Boruvka and Joseph Wright) for `biblatex`.

---

[12] Cf. `http://tex.stackexchange.com/questions/79555/biblatex-bibliographyoption-with-braces`.

# References

[1] Lehman, Philipp (with Audrey Boruvka, Philip Kime and Joseph Wright): *The biblatex Package. Programmable Bibliographies and Citations*. March 3, 2016. `http://www.ctan.org/pkg/biblatex`.

[2] Markey, Nicolas: *Tame the BeaST. The B to X of BibTEX*. October 11, 2009. `http://www.ctan.org/pkg/tamethebeast`.