
























bibhtml produces a bibliography in HTML from one or several BibTeX files

- **bibhtml** supports regular and **user-defined** BibTeX types and fields. BibTeX types/fields are predefined, as well as **bibhtml**-specific and common **biblatex** fields. Other types/fields can be displayed in HTML files by changing **bibhtml Styles**.
- An **image** can be displayed on the left or right side of each BibTeX entry by means of the **bibhtml**-specific **image** field.
- Fields can **open URLs** or call **Javascript functions**. By default, *doi*, *hal*, *url*, *pdf*, *video*, *poster*, *slides*, *software*, *project* open URLs and *abstract*, *keywords*, *bibcite* show the content of these fields on demand. These fields can be displayed as **icons**.
- **bibhtml** can **merge several** BibTeX files, taking into account the BibTeX keys of their entries. This allows enriching entries by defining their fields in different files (eg. one file with regular fields, another with **bibhtml**-specific **fields** specifying images, keywords, non standard hyperlinks, etc.).
- Entries can be **selected** and **sorted** according to their field values or a combination of them. They can be sorted according to a special **rank** field that controls their order of appearance.
- Entries are **formatted** according to **bibhtml Styles** that are defined in BibTeX files (a generic style is used by default). HTML **rendering** can be easily customized by editing a **HTML/CSS template file**.
- **bibhtml** consists of a single **single C++11** program, no other tool is needed.

2017

- [167] **MarkPad: Augmenting Touchpads for Command Selection.**
B. Fruchard, E. Lecolinet, O. Chapuis.
In *CHI'17: Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, ACM (2017). 5630-5642.
    [keywords](#) [abstract](#) [link](#) [project](#) [software](#)
- [166] **CoReach: Cooperative Gestures for Data Manipulation on Wall-sized Displays.**
C. Liu, O. Chapuis, M. Beaudouin-Lafon, E. Lecolinet.
In *CHI'17: Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, ACM (2017). 6730-6741.
    [keywords](#) [abstract](#) [link](#)
- [165] **VersaPen: An Adaptable, Modular and Multimodal I/O Pen.**
M. Teyssier, G. Bailly, E. Lecolinet.
In *CHI'17 Extended Abstracts: ACM SIGCHI Conference on Human Factors in Computing Systems*, ACM (2017). 2155-2163.
    [keywords](#) [abstract](#) [link](#)
- [164] **VersaPen: Exploring Multimodal Interactions with a Programmable Modular Pen.**
M. Teyssier, G. Bailly, E. Lecolinet.
In *CHI'17 Extended Abstracts (demonstration): ACM SIGCHI Conference on Human Factors in Computing Systems*, ACM (2017). 377-380.
    [keywords](#) [abstract](#) [link](#)
- [163] **Revue et Perspectives du Toucher Social en IHM.**
M. Teyssier, G. Bailly, E. Lecolinet, C. Pelachaud.
In *IHM'17: Conférence francophone sur l'Interaction Homme Machine*, ACM (2017).
   [keywords](#)
- [162] **Démonstration de MarkPad : Augmentation du pavé tactile pour la sélection de commandes.**
B. Fruchard, E. Lecolinet, O. Chapuis.
In *IHM'17: 29ème conférence francophone sur l'Interaction Homme-Machine*, (2017). 2.
  [keywords](#)
- [161] **Interaction Techniques Exploiting Memory to Facilitate Command Activation.**
B. Fruchard, E. Lecolinet, O. Chapuis.
In *29ème conférence francophone sur l'Interaction Homme-Machine*, (2017). 5.
  [keywords](#)



Input

- One or several BibTeX files containing **regular**, **bibhtml** and **custom** entries or **styles**.
- A **HTML template** containing (or referring to) a **CSS** file (a default file is provided).
- Optionnaly, a directory containing **images** .

Output

- The bibliography in **HTML** or **BibTeX** format.

Batch and interactive mode

bibhtml starts in **interactive mode** if no argument is provided. Otherwise it is launched in **batch mode** and executes in sequence the arguments of the command line without prompting the user.

To launch **bibhtml** in **interactive mode** after executing the arguments of the command line, add **-inter** as an argument.

In interactive mode, the **-** sign is optional and **:** can be replaced by spaces.

Producing a HMTL file from BibText files

```
bibhtml refs.bib
bibhtml refs1.bib refs2.bib ref3.bib
bibhtml refs1.bib refs2.bib ref3.bib -aux:auxrefs.bib -ima:data/images
      -templ:mytemplate.html -out:mybib.html
```

An arbitrary number of **BibTeX** (.bib) files can be provided. They must be encoded in **UTF8**.

The BibTeX files are **merged**, taking into account the BibTeX **key** of their entries (a later definition of a field replaces its previous definition)..

This can serve to produce a common publication list for multiple authors or to **enrich** an existing BibTeX file by adding **additional fields** to its entries in another file. For instance, one can combine institutional BibTeX files containing only regular BibTeX fields with personal files containing customized or bibhtml fields for inserting **images**, **abstracts**, **keywords** and non standard **hyperlinks**.

The following arguments are optional:

-out [html] changes the name of **produced HMTL file**. *out.html* is used by default.

-ima specifies a **directory containing images**. The *current directory* is used by default. Images can be displayed on the side of entries. Hyperlinks can also appear as icons. See **-format**.

-templ changes the **HMTL template file**. *template.html* (provided in the distribution) is used by default. The produced HTML file is a copy of the template file with the bibliography inserted in it. This template file typically contains *Javascript* functions and a *cascaded style sheet (CSS)*.

-aux indicates an **auxiliary** BibTeX file. Entries of auxiliary files will appear only if they also exist in a "normal" BibTeX file. They serve to add **bibhtml** or **custom fields** to other entries. Multiple auxiliary files can be specified (each after **-aux:**).

Producing a BibTeX file from BibText files

Works in the same way except that **-outbib** must be used instead **-outhtml**. The image directory and the HTML template are not used in this case. This function can serve to create a new bibliography after merging files or sorting or selecting entries (see below).

Sorting entries

```
bibhtml refs.bib -sort:"type"
bibhtml refs.bib -sort:"-year"
bibhtml refs.bib -sort:"type & -year"
```

Any field can be used for sorting entries. Case does not matter. Examples:

- **-sort:"type"** : entries are **sorted** according to their *type*.
- **-sort:"-year"** : entries are sorted according to their *year* in **reverse order** (latest first).
- **-sort:"type & -year"** : entries are first sorted by *type* **then** by *year*. Note that the order matters (sorting by *year* **then** by *type* won't produce the same result).
- **-sort:"?"** : displays **help** about the **sort** command

Sorting entries with headers

- `bibhtml refs.bib -sort:"-year" -format:"headers=year"`

Not only entries are sorted by *year* but a header showing the *year* is printed before each group of entries with the same *year*. Any field can be used. See `-format` for more details.

Selecting entries

```
bibhtml refs.bib -select:"author=einstein"
bibhtml refs.bib -select:"author=einstein & year=1905"
bibhtml refs.bib -select:"author=einstein & year!=1905"
```

This command allows selecting a subset of the database. Any field can be used. Case does not matter. Examples:

- `-select:"author=einstein"` : the field *author* must **contain** *einstein*.
- `-select:"year!=1905"` : *year* must **not** contain *1905*.
- `-select:"author=einstein & year=1905"` : **both** criteria must be satisfied
- `-select:"*"` : selects **all** entries
- `-select:"?"` : displays **help** about the `select` command

`select` and `sort` commands can be executed in sequence as in the following example: entries containing *einstein* as an *author* are selected, then sorted in reverse order according to their *year*:

```
bibhtml refs.bib -select:"author=einstein" -sort:"-year"
```

`select` can be called multiple times in sequence in which case select clauses are *anded*. This feature is mostly useful when `bibhtml` is used interactively. To select again all the entries of the database, perform `-select:"*"` or `-all`.

bibhtml fields

The following fields have a special meaning:

- `image` contains the file name of an **image** that will be displayed on the side of the entry. Images must be located in the directory specified by `-ima`. Whether and how images are displayed depend on `-format`.
- `url`, `pdf`, `video`, `slides`, `poster`, `project`, `link` contain an hyperlink that will open the corresponding page when this field is clicked on the produced HTML page.
- `doi`, `hal` contain an ID that is transformed into an hyperlink providing information on this entry (respectively by prefixing this ID by `https://dx.doi.org/` or `https://hal.archives-ouvertes.fr/`).
- `abstract`, `keywords` allow displaying the **abstract** and the **keywords** of the entry **on demand** (clicking on the field will make the corresponding text to appear/disappear in the HTML page). The `abstract` and `keywords` fields contain the corresponding text.

Apart from images, all these fields can appear as text or as icons (or be dismissed) depending on the `Style` and on `-format` specifications. By default, they are displayed (for entries defining them) and displayed as text. When displayed as icons, the corresponding image files must be located in the directory specified by `-ima`.

Other hyperlink fields can be added by changing the `Style` (see this section).

Changing the HTML template

The produced HTML file is a copy of the template file with the bibliography inserted in it. The template file typically contains *Javascript* functions and a *cascaded style sheet (CSS)*. It can also refer to the CSS rather than including it.

`-templ` changes the **HMTL template file**. *template.html* (provided in the distribution) is used as a default. If the default template is used it must be in the same directory as the `bibhtml` program.

This template file must contain the following line, which will be replaced by the bibliography:

```
<!-- ###CONTENT - DO NOT REMOVE - CONTENT WILL BE ADDED HERE -->
```

The **CSS** (a default CSS is included in *template.html*) specifies how fields are rendered. This file can easily be customized. Example:

```
.image {height:70px; max-width:150px; float:right;}
.title {font-weight:bold;}
.booktitle {font-style: italic;}
.series {font-style: italic;}
.journal {font-style: italic;}
.volume {font-style: italic;}
.award {font-weight:bold; color:brown;}
.keywords-value {display:none; font-style: italic; color:brown;}
.abstract-value {display:none; color:brown;}
```

The *template.html* file also contain **Javascript** functions which allow sorting entries and showing/hiding the content of the `abstract` or `keywords` and `bibtex` fields.

Formatting options

```
bibhtml refs.bib -format:"number=yes & breaklines=yes & images=right"
```

- `breaklines = yes|no`
Insert **line breaks** (default = yes)
- `numbers = yes|no`
Show entry **numbers** (default = no)
- `images = no|left|right`
Show **images** on the left or right side (default = left). *no* disables images.
- `links = no|text|icons`
Show **HTML links** as text or icons (default = text). *no* disables all hyperlink fields.
- `headers = no|<field>`
Show **headers** (default = *year*). `<field>` is the name of the desired field (*year*, *type*, etc.)
- `style = default|<style>`
Change **style** (default = generic style). `<style>` is the name of an entry in a BibTeX file that defines a **Style** (see below).

Changing the BibTeX Style

A **Style** definition specifies, for each entry type, which fields are printed and in which order. **Styles** are defined as entries with the special **style** type in BibTeX files. The field names of a **style** entry are BibTeX types and their values are **style formats**:

- The **media** field, which is **mandatory**, provides a **default style format** for printing entries.
- The other fields specify how entries of the specified type are printed.

Only the **media** field need to be defined. It is used as a default for types that do have a specific format:

```
@style{mystyle,
  media = {%author.\n %title.\n %journal, %[In %booktitle], %eprint, %howpublished,
%volume, %number, %[editor (Eds.)], %school, %institution. %publisher %[(%year)]. %pages.
%pubstate.\n %award.\n %^doi<doi.png> %^url<url.png> %^hal<hal.png> %^pdf<pdf.png>
%^video<video.png> %^slides %^poster %^project %^link %?keywords %?abstract},

  inproceedings = {%author.\n %title.\n %[In %booktitle], %volume, %[editor (Eds.)],
%school, %institution. %publisher %[(%year)]. %pages. %pubstate.\n %award.\n
%^doi<doi.png> %^url<url.png> %^hal<hal.png> %^pdf<pdf.png> %^video<video.png>
%^slides %^poster %^project %^link %?keywords %?abstract},

  article = {%author.\n %title.\n %journal, %volume, %number, %[editor (Eds.)],
%school, %institution. %publisher %[(%year)]. %pages. %pubstate.\n %award.\n
%^doi<doi.png> %^url<url.png> %^hal<hal.png> %^pdf<pdf.png> %^video<video.png>
%^slides %^poster %^project %^link %?keywords %?abstract}
}
```

BibTeX files can contain as many **style** definitions as needed. The style that is actually used for printing the HTML file is specified by calling **-format** (see above). By default, a **generic style format** (which is hard-coded in the program) is used.

A **style format** is string that contains the fields that will be displayed for an entry of the corresponding type. If an entry does not contain a given field, it is not displayed.

The **generic style format** is defined as follows:

```
"%author.\n %title.\n %journal, %[In %booktitle], %eprint, %howpublished,
%volume, %number, %[editor (Eds.)], %school, %institution, %publisher
%[(%year)], %pages.\n %award.\n %^doi | %^url | %^pdf | %^video | %^slides
|^poster | %^project | %^link | %^link2 | %?keywords| %?abstract"
```

- **%field** (e.g. **%title**) means that the *value* of this field will be displayed if it is defined (and unempty). Punctuation marks and other characters between **%field** specifications are printed only if needed.
- **\n** means that a line break will be inserted. **-format:!br** disables this behavior.
- **%^pdf** indicates that the value of this field is an URL. The name of this field will appear as an hyperlink (i.e. [pdf](#)) in the HTML file. It can also be displayed as an icon by specifying an image file name as follows: **%^pdf<pdf-icon.png>**.
- **%?abstract** indicates that the value of this field contains text that will be displayed on demand. The name of this field will appear as an hyperlink (i.e. [abstract](#)). Clicking on the link will make the text to appear/disappear in the page. This hyperlink can also be displayed as an icon (see above).

- `%[In %booktitle], %[(%year)], %[editor (Eds.)]` are examples where a prefix and/or a suffix are needed. For instance, `In Whatever` will be printed in the HTML page if the value of `booktitle` is "Whatever" and `(1905)` if the value of `year` is "1905". Nothing will be printed if the value is undefined/empty.

Note that hyperlinks, hyperlink icons and images are displayed (or not displayed) depending on the `-format` option (see above). This makes it possible to print the HTML page differently according to these options without having to use different styles.

Calling the bibhtml program

The syntax for calling `bibhtml` is:

```
bibhtml file1.bib [file2.bib] ... -command1 -command2:argument ...
```

`bibhtml` enters **interactive mode** if no argument is provided or if the `-inter` command is present. The commands are the same as on the command line except that the `-` sign can be omitted and that `:` can be replaced by spaces.

Commands and abbreviations

Commands can be abridged: the part between `[]` is optional. The `-` sign can be omitted in interactive mode.

<code>-r[ead]:bib_file</code>	Read a BibTeX file (only needed in interactive mode)
<code>-aux:bib_file</code>	Read an auxiliary BibTeX file
<code>-t[empl]:html_file</code>	Read the HTML template (default: <code>template.html</code>)
<code>-ima:directory</code>	Set the directory containing images (default: <code>current directory</code>)
<code>-out[html]:html_file</code>	Set the name of the produced HTML file (default: <code>out.html</code>)
<code>-outbib:bib_file</code>	Set the name of the produced BibTeX file (default: <code>out.bib</code>)
<code>-f[ormat]:format</code>	Set printing format and style (<code>-format:"?"</code> for help)
<code>-a[ll]</code>	Select all entries
<code>-se[lect]:pattern</code>	Select entries (<code>-select:"?"</code> for help)
<code>-so[rt]:pattern</code>	Sort entries (<code>-sort:"?"</code> for help)
<code>-show[html]</code>	Show the produced HTML file (opens the default Web browser)
<code>-showbib</code>	Show the produced BibTeX file
<code>-st[ats]</code>	Show statistics
<code>-h[elp]</code>	Show available commands
<code>-i[nter]</code>	Interactive mode
<code>-v[erbose]</code>	Verbose mode
<code>-cl[ear]</code>	Clear the database (remove all entries).
<code>-ch[eck]:field</code>	Check validity of the links of this <i>field</i>
<code>-q[uit]</code>	Quit the application (useful in interactive mode)

Author/Contact

Eric Lecolinet – Télécom ParisTech

eric.lecolinet@telecom.paristech.fr

<http://www.telecom-paristech.fr/~elc>