Bib2Html: User Documentation

Eric Marchand

Lagadic project
http://www.irisa.fr/lagagic

December 24, 2008

Foreword:

This documentation aims to describe the Bib2html tool. As it is written by the developer, and not by users, it may not be convenient for the latter. Please feel free to comment and react on it: by doing so, you could help us to improve it.

General information

Bib2html is a BibTEX to HTML translator. It is written in C++.

The purpose of this software is to automatically produce html pages from BibTEX files, and to provide access to the BibTEX entries by several criteria: year of publication, category of publication, keywords, author name. Moreover cross-linking is generating between pages to provide an easy navigation through the pages without going back to the index.

Distribution

Current version is 216, it can be downloaded from:

        `http://www.irisa.fr/lagadic/soft/bib2html/download.html`

The distribution contains:

- Full source written in C++,

- Some icons,

- User manual in PDF

- An example directory with some BibTEX files and configuration files.

If you download Bib2html and find it useful for your work, I would appreciate your sending me a short email (to Eric.Marchand@irisa.fr) with your remark and let me known the url of your site (just to known if someone really uses this soft...). Such correspondence is mutually beneficial because I am curious to know how this software is being used, and your input can help guide future releases.

Condition of use

There are no restrictions on the use of Bib2html except that I take no liability for any problems that may arise from this use.

You can use it in your commercial projects. However, if you distribute the source, please make it clear which parts are mine and that they are available essentially for free over the Internet.

Please understand that there may still be bugs and errors. Use at your own risk. I take no responsibility for any errors or omissions in this package or for any misfortune that may befall you or others as a result of its use.

Please report bugs to me at `Eric.Marchand@irisa.fr`. When reporting a bug please tell me which C++ compiler you are using, and what version. Also give me details of your computer. And tell me which version of Bib2html you are using. If you made any changes to my code I cannot do anything. If the bug occurs while parsing a bibtex file send me the bibetx file.

Acknowledgment

This manual took advantage of the manual of bibtex2html written by Gregoire Malandain.

Warning

- Many english errors (should have been written in french...).

- All capabilities are not described.

Contents

1 Presentation

The purpose of this software is to automatically produce html pages from BibTEX files, and to provide access to the BibTEX entries by several criteria: year of publication, category of publication, keywords, author's names. Moreover cross-linking is generating between pages to provide an easy navigation through the pages without going back to the index.

2 Getting started (on unix/linux)

2.1 Installing the sources

I assume you get a package from the above http://www.irisa.fr/prive/Eric.Marchand/bib2htmlmentioned ftp site. It is named bib2html.code.v-XXX.extension where XXX denotes a version number. The extension can be either .zip or .tar.gz. Thus running

```
% unzip -d    bib2html.code.vXXX.zip
```

or

```
% tar -xvzf bib2html.code.vXXX.tar.gz
```

should create a directory named bib2html.code.vXXX containing the sources and some other files.

2.2 Compiling

Go into the bib2html.code.vXXX directory

```
% cd bib2html.code.vXXX
```

and compile with make

```
% configure
% make
```

2.3 Generating html pages from BibTEX files

According that you have installed (copied) Bib2html in some place listed in your PATH variable, or that you are using the full path toward Bib2html you can process your .bib files. While parsing its arguments, Bib2html considers that all arguments that do not match a command line option are input files. The following command will then create a bundle of html files from the BibTEX files given in the command line.

```
% bib2html file1.bib file2.bib ...
```

You may navigate through them using the index.html as starting point. Notice the -force option, that will create the needed subdirectories if they do not exist.

```
% bib2html file1.bib file2.bib ... -force
```

3 Description of the generated html pages

By default, html pages are generated in

- the current directory, namely '/tmp/', it only concerns the index file,

- and subdirectories: `Author/`, `Biblio/`, `Category/`, `Keyword/`, `Year/`;

Each directory contains a number of html files each one corresponding to a given bibliography.

3.1 Bibliography by categories

Categories have been defined as a group of several BibTEX entry types, they are by order :

Books and proceedings	@book, @proceedings
Thesis	@phdthesis
HDR	@phdthesis
Articles in journal	@article
Articles in a national journal	@article
book's chapters	@inbook
International Conference's articles	@inproceedings, @incollection
National conference's article	@inproceedings
Internal reports	@techreport
Miscellaneous	@misc
Popularization	@inproceedings, @misc, @article
Invited conference	@inproceedings, @misc

In some cases, additional fields may be required[1]

- HDR : type = {Habilitation à Diriger des Recherches}

- national journal : x-international-audience = {no}

- national conference : x-international-audience {no}

- popularization : x-popularization = {yes}

- invited conference : x-invited-conference ={yes}

Note that if you never use the fields `x-international-audience` in the `@inproceeding` entries, all the conference will be placed in the international conference category (it might be necessary to change its name in the configuration file, see section 4.9.3). Same remarks for national journals.

One html file is generated per category in the `Category/` subdirectory (default `/tmp/Category/`). In these files, entries are sorted by year of publication in decreasing order.

---

[1]These fields are compatible with the INRIA activity report RAWEB.

3.2 Bibliography by authors

Bib2html will generate one html file per indexed author in the `Author/` subdirectory (default `/tmp/Author/`). In these files, entries are sorted by categories. By default, all the authors are indexed but it is also possible to change this behavior by specifying only the authors that you want to be indexed (eg, only the current member of your research team, all the other will not be indexed): see the sections 4.2. If not specified, each generated file will be named after the author's name, namely the first name followed by a '.' and the last name, e.g. `Eric.Marchand.html` (see Figure 1 and section 4.2).
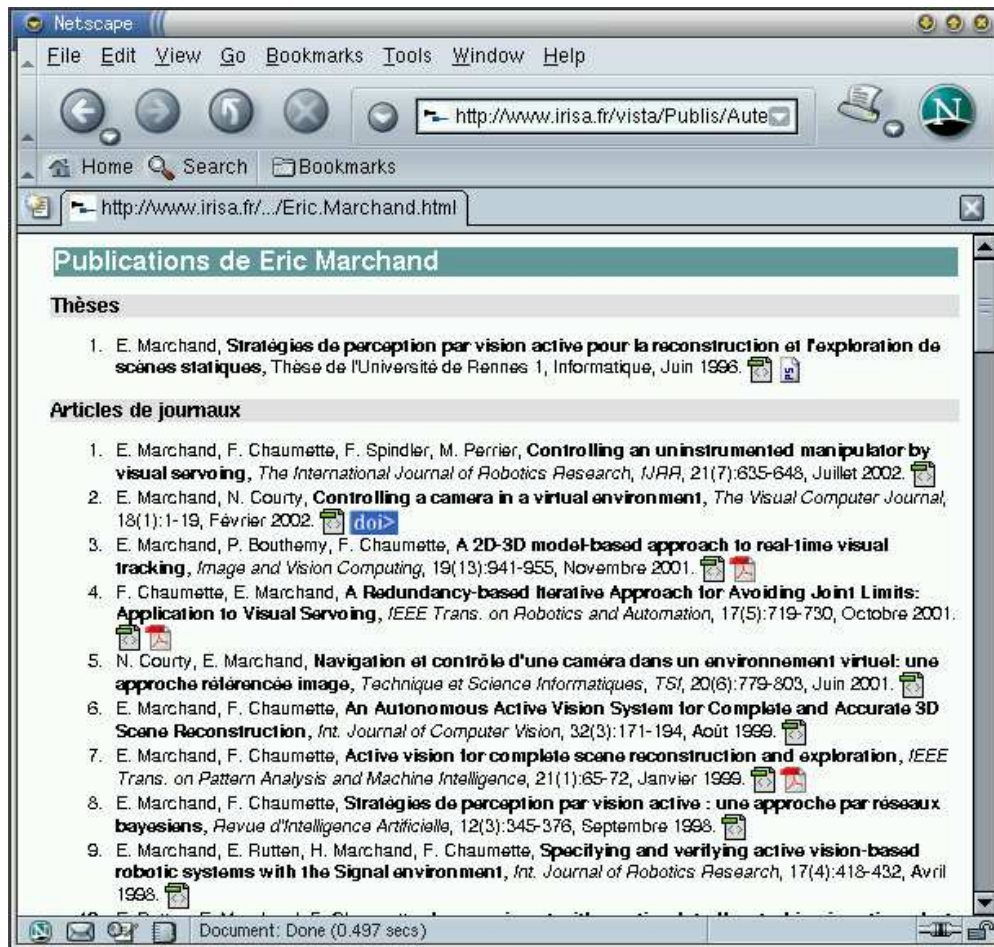

Figure 1: Bib2html will generate one html file per indexed author. In these files, entries are sorted by categories. By default, all the authors are indexed but it is possible to change this behavior by specifying only the authors to be indexed.

The way author's names should be write in the bibtex file is explained in appendix B extracted from [2]. Mainly if you do not want to have any problem you should write something like:

```
author = {Lamport, L. and {von Beethoven}, L. and Knuth, Donald E.}
```

Comparison of names   In large bibliographies, the same author name can be written in different ways. E.g. you may find either Jean-Sebastien Bach, J.S. Bach, J.-S. Bach, or J. Bach. Two names are considered

to be the same if the last names are exactly equal, and if the first initial of the first names are also the same. From above, it comes that Jean-Sebastien Bach, J.S. Bach, J.-S. Bach and J. Bach will be considered as designing the same person.

3.3 Bibliography by years

For each year found in the BibTEX files, a file is produced in the `Year/` subdirectory (default `/tmp/Year/`) and is named year.html where year is a four digit number corresponding to the year. In each file/page, the entries of the year are sorted with respect to the above described categories (see Figure 2).
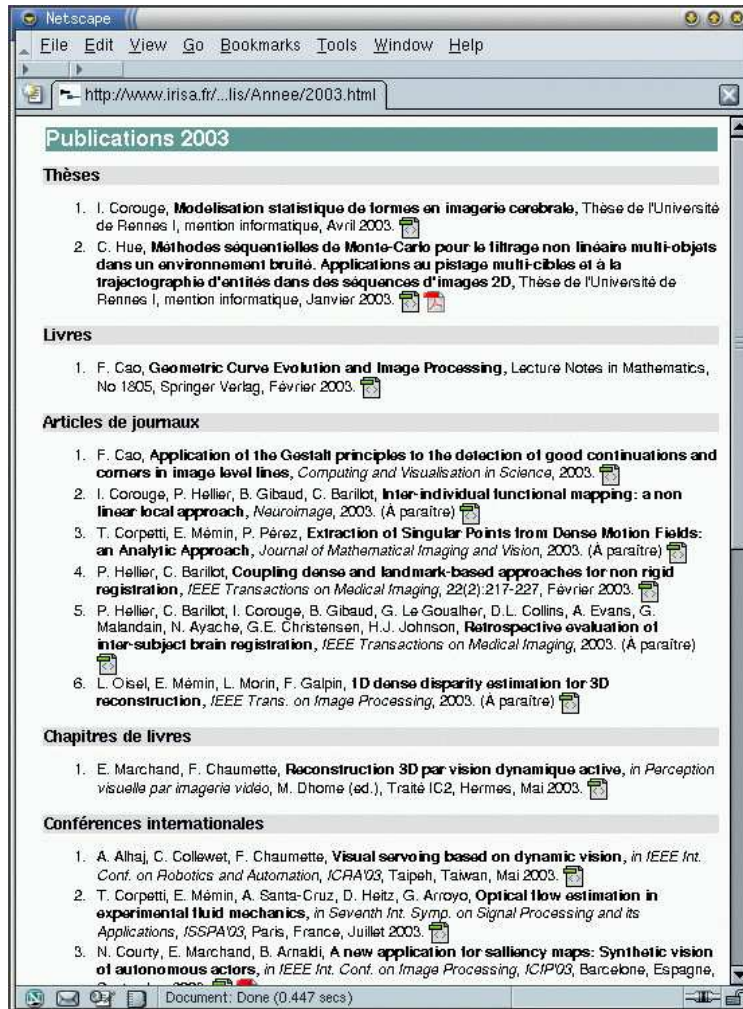
Figure 2: For each year found in the BibTEX files, a file is produced. In each file/page, the entries of the year are sorted with respect to the above described categories.

3.4 Bibliography by keywords

Bib2html will generate one html file per indexed keyword in the `Keyword/` subdirectory (default `/tmp/Keyword/`). In these files, entries are sorted by categories. By default, all the recognized keywords are indexed, but it is

8

possible to change this behavior by specifying either the keywords to be indexed (and all the other will not be indexed): see the section 4.3 for details. Each generated file will be named after the keyword.

3.5 Paper description

Along with the bibliography, Bib2html can provide a more complete description of each paper. It will create a specific page for each paper that contains the full paper description, download information, cross references, bibtex and endnote references, author contact if available, notes if available, abstract, etc. A file is produced in the `Biblio/` subdirectory (default `/tmp/Biblio/`). File is named after the entry key. Description is only in english.

Figure 4 shows the result of paper page produces from the bibtex entry presented on Figure 3 and that ilustrates all the possible outputs.

```
@Article{Marchand02b,
  author =        {Marchand, E. and Courty, N.},
  title =         {Controlling a camera in a virtual environment.},
  journal =       {The Visual Computer Journal},
  year =          {2002},
  volume =        {18},
  number =        {1},
  month =         {Fevrier},
  pages =         {1-19},
  abstract =      {This paper presents an original solution to the camera
                   control problem in a virtual ....
                   within this framework and presented in this paper.},
  doi =           {http://dx.doi.org/10.1007/s003710100122},
  pdf =           {http://www.irisa.fr/lagadic/pdf/2002_vc_marchand.pdf},
  crossref =      {Courty01b},
  x-misc =          {
   <li>French Version published in Technique et Science Informatiques, TSI
   <li> <a href="http://www.irisa.fr/vista/..../demo-animation.html">
        Demonstration
        </a>
                 }
}
```

Figure 3: The bibtex entry that allows to produce the paper page presented on Figure 4

This page (Figure 4) features the complete publication reference on the top of the page, the download possibilities (here a http://www.doi.org doi address to download the paper on the Elsevier www page (`doi`, `pdf` or `url` fields in the bibtex entry), a note about the paper (`x-misc` field), the abstract, some cross references, the possible contact (people for which an author page is created(see Section 4.2)), the bibtex entry, and the endnote (refer) reference (see Figure 5).

3.6 Index file

An index page, `index.html`, is created in the current directory. It contains links towards all the above mentioned files generated and is thus a good start to browse the bundle of files.

By default, all the authors (or editors if relevant) and keyword are indexed. This can be modified (see
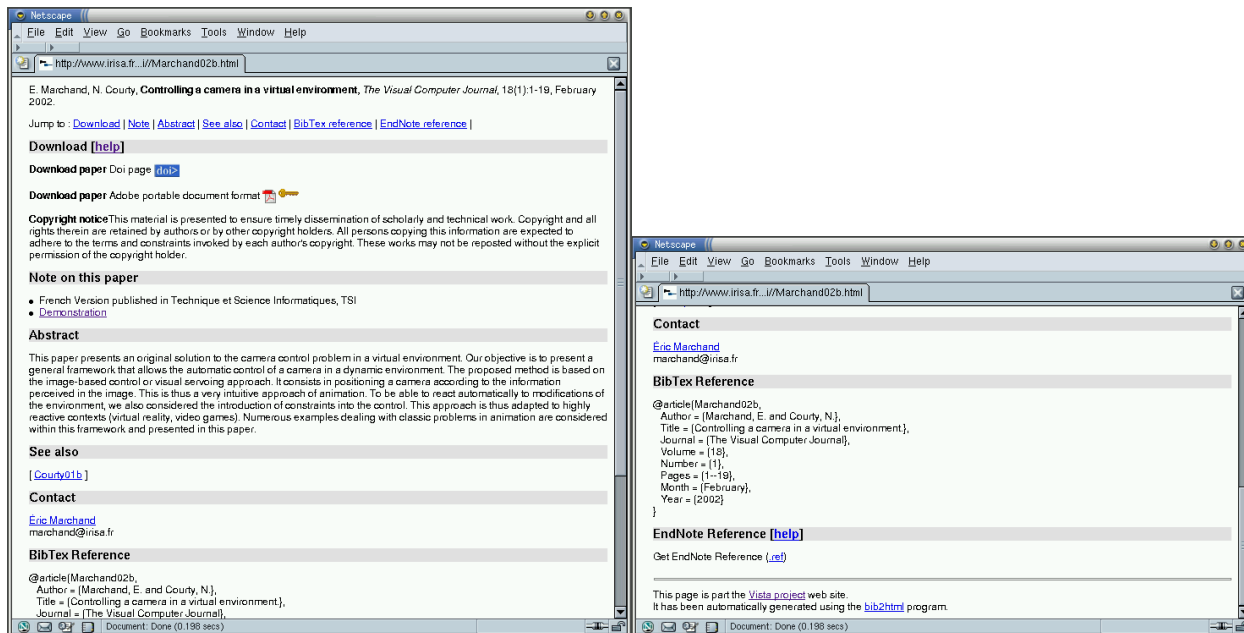
Figure 4: Bib2html produce specific page for each paper that contains the full paper description, download information, cross reference, bibtex and endnote references, author contact if available, notes if available, abstract, etc.

```
0 Journal Article
%F Marchand02b
%A Marchand, E.
%A Courty, N.
%T Controlling a camera in a virtual environment.
%J The Visual Computer Journal
%V 18
%N 1
%P 1-19
%X  This paper presents an original solution to the camera control problem in a virtual ... examples
%U http://dx.doi.org/10.1007/s003710100122
%8 February
%D 2002
```

Figure 5: The refer entry produced from bibtex entry on figure 3

Section 4.2 and 4.3). In the presented page (see Figure 6) all the authors and keywords found in the bibliography are not indexed.

Only categories with a positive number actual entries are added to the index file.

Figure 6: An index page is created. It contains links towards all the other generated and is thus a good start to browse the bundle of files.

3.7   Language

Each page can be generated in two languages (let's say english and french, but it can be whatever you want, eg German and Esperanto...). In these two pages the title are different and most of the page layout can be modified. Furthermore, in each entry month can be different (eg, Janvier vs January), address could be different (eg, Londres vs London), etc.

If such pages are required, they will be named by a specific name (eg, 1999-fra.html, 1999.english.html). Two index files are also generated.

See section 4.11 to set this option on/off.

4   Finer Control of the output

The default behavior of Bib2html should correspond to some consensual expectation of such a tool. However, one may want to adapt it to more specific need. The aim of this section is to explain how things can be changed.

4.1   Principle

There are a large number of variables that allows to parameterize the behavior of Bib2html. I choose to put those indications in some `@define ...` , `@author...`, `@keyword...`, etc. so that they can be added inside BibTex files (even if they may yields warnings). The file containing those indications, if

independent from the BibTex files to be processed, has simply to be added to the list of files to be processed. Thus, the syntax is always

```
@define{ key = the_value_I_want_to_set_for_key }
```

or

```
@define{ key.yes }
@define{ key.no }
```

when the choice is boolean.

The `@define` (or a similar commands) should be set within a configuration file that have to be passed to the command line. Assuming that the configuration file is `config.cnf`, the command to run Bib2html is:

```
% bib2html -config config.cnf file1.bib file2.bib ...
```

Let us note that the choice of the syntax also allows you to put it into your BibTEX file but it's better to create a Configure file.

4.2 Selecting authors

By default, all the authors (or editors if relevant) are indexed, and then one file will be generated per different name in the subdirectory `Author/`. Relevant editors are the ones of `@book`, `@proceedings`, or `@inbook` if there are no authors. However, one may want only to index some of them, and then all the others are considered as being non-indexed. It can be done by specifying the authors to be indexed in lines like:

```
@author{
 name = Marchand,
 givenname = Éric,
 email = marchand@irisa.fr,
 url = http://www.irisa.fr/prive/marchand,
 filename = E.M
}
```

or

```
@author{
 name = Marchand,
 givenname = Éric
}
```

if no email or url are available. These information will be useful to produce the contact information in each paper page (see section 3.5). If no specific filename is given the name of the output file (bibliography by author section 3.2) will be `GivenName.Name.html`, else it will be `filename.html`.

If authors are selected using the `@author` command, only these authors will appear in the index page and only the corresponding pages will be generated.

4.3 Selecting keywords

By default, all the keywords are indexed. However, one may want only to index some of them, and then all the others are considered as being non-indexed. It can be done by specifying the keywords to be indexed in lines like:

```
@keyword{key,
language1 = keyword_in_language_1,
language2 = keyword_in_language_2}
```

`language1` will defined the title of the page in language1, and `language2` its title in language2. This can be, for example:

```
@keyword{av,
language1 = Asservissement visuel,
language2 = Visual servoing}
```

If `language1` and `language2` are not defined the key, will the title in both languages. If keywords are selected using the `@keywords` command, only these keywords will appear in the index page.

4.4 Loading Bibtex File

If you do not want to use the command line to define all the BibTex files to be considered, you can specify their name in the configuration file as follows:

```
@BibFile{Lagadic01.bib, Lagadic02.bib, Lagadic03.bib}
```

4.5 Including more configuration file

You can specify other configuration files to be read within a configuration file as follow:

```
@ConfigFile{author.cnf,  month.cnf,  html.cnf}
```

It's like a #include in C.

4.6 Comparison of month

If you consider a large bibliography, people usually never write the month field in the same way (eg Jan, January, Janvier, 1, Jan., etc...). Though we defined in the code a large set of possibilities, you can add your personal setting as follow:

```
@Month{1, janvier, january, janvier, janvier, jan., janvier, jan}
@Month{2, fevrier, february, 2, f\'evrier, f\'{e}vrier, fev., février, feb, fev}
@Month{3, mars , march, 3, mars, mars, mar., mars, mar}
@Month{4, avril , april, 4, avril, avril, avr., avril, avr, apr }
@Month{5, mai , may, 5, mai, mai, mai, mai}
@Month{6, juin , june, 6, juin, juin, juin, juin, jun}
@Month{7, juillet , july, 7, juillet, juillet, juil., juillet, jul}
@Month{8, aout , august, 8, ao\^ut, ao\^{u}t, aout, août, aug, aou, AUG}
@Month{9, septembre, september, 9, septembre, septembre, sept., septembre, sep}
@Month{10, octobre , october, 10, octobre, octobre, oct,. octobre, oct}
@Month{11, novembre , november, 11, novembre, novembre, nov., novembre, nov}
@Month{12, decembre, december, 12, d\'ecembre, d\'{e}cembre, dec., décembre, dec}
```

In any case, a good choice it to name the months after their first three letters in english since these are standard macro in bibtex. For example:

```
@article{foo98a,
...
month = jun,
...
}
```

4.7 Files and directories names

The names of the subdirectories can be changed. The "directory.base" directory defines the location where the generated html or the sub directories will be created. It can be defined as follows (default):

```
@define{directory.base = /tmp/}
```

The sub directories are defined by:

```
@define{  directory.author = Author/ }
@define{  directory.keyword = Keyword/}
@define{  directory.year= Year/ }
@define{  directory.category= Category/ }
@define{  directory.publi= Biblio/ }
```

If you do not want to create these sub directories, set these variable to ".", eg:

```
@define{  directory.author = ./ }
```

The base http address defines the address where people could see the generated html pages on the net. It is given by the following definition (default):

```
@define{base.http = /tmp/}
```

Default is /tmp/ since it helps to see the results. However it should be something like:

```
@define{base.http = http://www.irisa.fr/lagadic/publi/}
```

It is necessary to create cross referencing and links between the created pages.
The index.filename command set the name of the index file.

```
@define{index.filename = index}
```

The index file will be index.html
You can modify the postfix of each html file. Default is a ".html" postfix.

```
@define{html.postfix = .htm}
```

If pages are generated for two languages, it is necessary to define a postfix for each page. Default is:

```
@define{language1.postfix = -eng}
@define{language2.postfix = -fra}
```

Only two languages are considered (see Section 4.11).

4.8   Selecting Years

By default, a file each year is created. However, one may want only to create some of them, and then all the others are considered as being non-created. It can be done by specifying the years to be indexed in lines like:

```
@year{ 1989, 1990}
```

If years are selected using the @year command, only these years will appear in the index page.

4.9   Changing page layout

You surely want to change the "look" of the generated pages. This is possible by setting the values of different keys within a set of @define command.

4.9.1   Page layout

A general description of the page is given by:

```
<**page.start**>
   <**file.css**>
   <meta name="GENERATOR" content="bib2html 2.00">
   <**page.header**>

   ... the html description of the page ...

   <**page.footer**>
<**page.end**>
```

4.9.2   page.start and page.end

page.start and page.end defines the beginning and closing of the html pages It can something like:

```
# page.start
#   tag to begin the html page
@define{ page.start =
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
<link rel="stylesheet" type="text/css" href="css/lagadic.css">
<link rel="stylesheet" type="text/css" href="css/lagadic-publi.css">
<html>
<body>
}

#  page.end
#   tag to end the html page
@define{ page.end =

This page is part the  <a href="http://www.irisa.fr/lagadic">Lagadic project</a> we
```

```
It has been automatically generated using the <a href="http://www.irisa.fr/lagadic/
```

```
</body>
 </html> }
```

Don't forget to include your css files in the `page.start`. For language issue you can define: `page.start.language1`, `page.start.language2`, `page.end.language1` and `page.end.language2`.

Warning Please do not remove or add this line from the page footer or page end:

```
 It has been automatically generated using the
 <a href="http://www.irisa.fr/lagadic/soft/bib2html/bib2html.html">
  bib2html
 </a>
 program.
```

4.9.3   Changing categories names

It is obviously possible to modify the category names. Default are defined (in french and english) as:

```
@define{ category.phd.language1 = Thesis }
@define{ category.journal.language1 =  Academic Journals }
@define{ category.book.language1 = Books }
@define{ category.bookchapter.language1 = Book Chapters }
@define{ category.conf.internationale.language1 = International Conferences }
@define{ category.conf.nationale.language1 = National Conferences }
@define{ category.conf.workshop.language1 = Workshops and Symposiums }
@define{ category.rr.language1 = Research Reports}
@define{ category.misc.language1 = Misc }

@define{ category.phd.language2 = Thèses }
@define{ category.journal.language2 = Articles de journaux }
@define{ category.book.language2 = Livres }
@define{ category.bookchapter.language2 = Chapitres de livres }
@define{ category.conf.internationale.language2 = Conférences internationales  }
@define{ category.conf.nationale.language2 =  Conférences nationales}
@define{ category.conf.workshop.language2 = Workshops et symposiums }
@define{ category.rr.language2 = Rapports de recherche  }
@define{ category.misc.language2 = Divers }
```

4.9.4   Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) is a simple mechanism for adding style (e.g. fonts, colors, spacing) to web documents. It is done by attaching styles to pages with the following statement

```
<link rel="stylesheet" type="text/css" href="mystyle.css">
```

inserted after `</title>` and before `</head>` in the `page.start`.

Please refer to the w3c www page for a description on how to use css file (`http://www.w3.org/Style/CSS/`).

Within the css file page style can defined:

It is thern possible to define the way author name are displayed. Default is:

```
.BibAuthor{}
```

If, for example, you want the author written boldface:

```
.BibAuthor{font-weight: bold}
```

For a journal:

```
.BibJournalTitle{ font-weight: bold ;}
.BibJournal{  font-style: italic}
```

defined the way the title of the paper and the journal name are displayed.

For a conference paper:

```
.BibInProceedingsTitle{ font-weight: bold ;  }
.BibInProceedingsBooktitle{ font-style: italic  }
```

defined the way the title of the paper and the conference name are displayed.

For a book:

```
.BibBookTitle{ font-weight: bold ; }
.BibPublisher{  font-style: italic}
```

defined the way the title of the book and the publisher name are displayed.

For other entries, other definition can be set:

```
.BibInBookTitle{ font-weight: bold ; }
.BibInBookBooktitle{ font-style: italic}

.BibPhDThesisTitle{ font-weight: bold ; }
.BibPhDThesisSchool{font-style: italic}

.BibMiscTitle{ font-weight: bold ; }
.BibMiscBookTitle{  font-style: italic}
.BibMiscHowpublished{ font-style: italic}

.BibTechReportTitle{ font-weight: bold ;}
.BibTechReportInstitution{  font-style: italic}
```

The style of the copyright notice can also be defined:

```
.BibCopyright{font-style: italic}
```

4.10   Other definitions

The list environment defines the way list of publication are displayed. Default are:

```
@define{ list.start = <ol> }
@define{ list.item = <li> }
@define{ list.end = </ol> }
```

which means that publication will be displayed as:

1. publi 1

2. publi 2

Copyright notice can be defined. It will be added at the end of the files.

```
@define{copyright.notice.language1 =
  This material is presented to ensure timely dissemination of scholarly and
  technical work. Copyright and all rights therein are retained by authors or
  by other copyright holders. All persons copying this information are expected
  to adhere to the terms and constraints invoked by each author's
  copyright. These works may not be reposted without the explicit permission of
  the copyright holder.
}
```

```
@define{copyright.language2 = Les documents contenus dans ces répertoires sont
  rendus disponibles par les auteurs qui y ont contribué en vue d'assurer la
  diffusion à temps de travaux savants et techniques sur une base
  non-commerciale. Les droits de copie et autres droits sont gardés par
  les auteurs et par les détenteurs du copyright, en dépit du fait qu'ils
  présentent ici leurs travaux sous forme électronique. Les personnes
  copiant ces informations doivent adhérer aux termes et contraintes
  couverts par le copyright de chaque auteur. Ces travaux ne peuvent pas
  être rendus disponibles ailleurs sans la permission explicite du
  détenteur du copyright.}
```

You can modify the title of the index page using

```
@define{title.index.language1 = Publications of the Lagadic project}
@define{title.index.language2 = Publications du projet Lagadic}
```

4.10.1   Icons and other files

You can add icon within the html page. These icons are download icons (pdf, ps and a key if access to the pdf or ps is restricted, a doi icon, and for the link toward the paper page an "abstract icon").

You can specify the icons location with:

```
# icons link
@define{ url.icon.key  = http://www.irisa.fr/lagadic/Icones/clef3.gif}
@define{ url.icon.pdf  = http://www.irisa.fr/lagadic/Icones/pdf.gif}
```

```
@define{ url.icon.ps  = http://www.irisa.fr/lagadic/Icones/ps.png}
@define{ url.icon.abstract  =
http://www.irisa.fr/lagadic/Icones/abstract_icon.gif}
@define{ url.icon.doi  = http://www.irisa.fr/lagadic/Icones/doi-logo-small.gif}
@define{ url.icon.html  = http://www.irisa.fr/lagadic/Icones/html.png}
```

Default are:

```
# icons link
@define{ url.icon.key  = }
@define{ url.icon.pdf  =  }
@define{ url.icon.ps  = }
@define{ url.icon.abstract  = }
@define{ url.icon.doi  = }
@define{ url.icon.html  = }
```

which means that no icons will be added. Comparison between a page generated with and without icons is presented on Figure 8. The icons included in the distribution are presented on Figure 7.

abstract icon

pdf icon

ps icon

doi icon

key icon

html icon

Figure 7: The icons included in the distribution.

Figure 8: Comparison between a page generated with and without icons.

You can also write a page to explain how to download the paper. Page location will be defined as:

```
@define{ url.download.help  = http://www.irisa.fr/lagadic/publi/download_help.html}
```

default is:

```
@define{ url.download.help  = }
```

An example of such an help page are given in the `icons` directory of this distribution (see Figure 9).
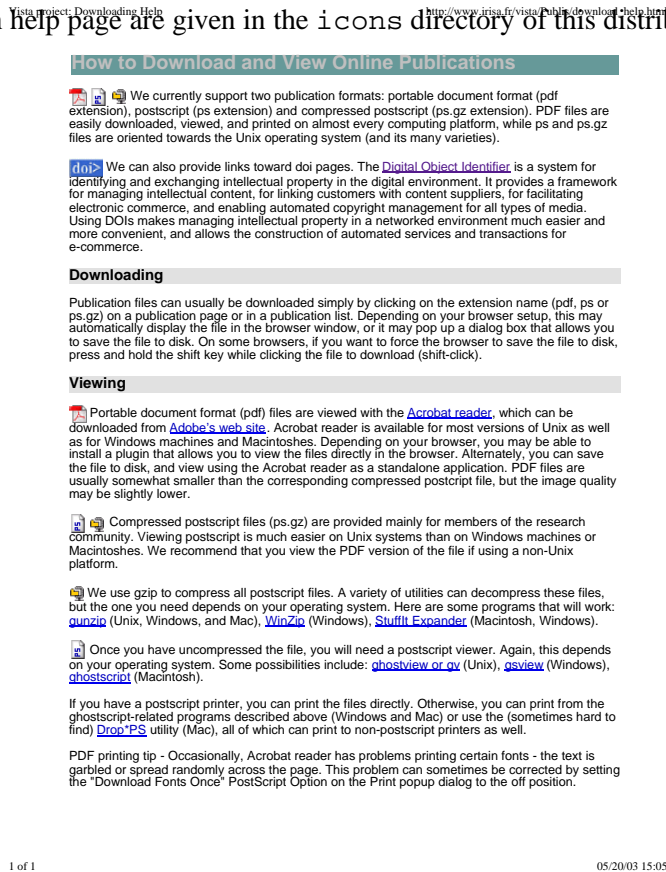

Figure 9: Example of Download help page.

4.11 Generating or not part of the bundle

It is also possible to generate only a sub-part of the bundle of files, by setting the values of different keys within @define (see section 5.1). The '1' value means the creation/generation of the files, while the '0' value means no creation.

- the bibliographies per author, described in section 3.2 and generated in the ./Author/ subdirectory, will not created if the author create pages key is set to false, i.e.

```
@define{ author.create.pages.no }
```

or

```
@define{ author.create.pages.yes }
```

- the bibliographies per keyword, described in section 3.4 and generated in the ./Keyword/ subdirectory, will not created if the keyword create pages key is set to false, i.e.

```
@define{ keyword.create.pages.no }
```

- the bibliographies per category, described in section 3.1 and generated in the ./Category/ subdirectory, will not created if the category create pages key is set to false, i.e.

```
@define{ category.create.pages.no }
```

- bibliographies per year, the ones of name year.html, described in section 3.3 and generated in the ./Year/ subdirectory, will not created if the year create pages key is set to false, i.e.

```
@define{ year.create.pages.no }
```

- paper description in section 3.5 and generated in the ./Bilbio/ subdirectory, will not created if the biblio create pages key is set to false, i.e.

```
@define{ biblio.create.pages.no }
```

- Pages in two languages can be generated. Default is:

```
@define{ french.no }
@define{ english.yes }
```

- Generate a category for popularization papers, to be used with the x-popularization = yes field in the entry (see section A.2).

```
@define{generate.popularization.pages.yes}
@define{generate.popularization.pages.no}
```

- Generate a category for invited papers, to be used with the x-invited-conference = yes field in the entry (see section A.2).

```
@define{generate.invited.pages.yes}
@define{generate.invited.pages.no}
```

- Generate a category for national journal papers, to be used with the x-international-audience = no field in the entry (see section A.2).

```
@define{generate.national.journal.pages.yes}
@define{generate.national.journal.pages.no}
```

- Generate a category for national conference papers, to be used with the x-international-audience = no field in the entry (see section A.2).

```
@define{generate.national.conferences.pages.yes}
@define{generate.national.conferences.pages.no}
```

5 Practical information

5.1 Examples of output

- I use the computational geometry bibliography, geom.bib, for testing. The version I got contains 13824 BibTEX items, there are 8016 different names among 27487 authors and 1851 different keywords among 8807 keywords. Due to the size of the generated bundle of files, I can not keep it as an output example. The BibTEX file can retrieved by anonymous ftp from ftp://ftp.cs.usask.ca/pub/geometry/ or use some web search interface

  - http://www.cs.ruu.nl/usr/cgi-bin/bib-search/geom
  - http://www-ma2.upc.es/ geomc/geombib/geombibe.html
  - http://cgm.cs.mcgill.ca/geombib.html
  - http://netlib.bell-labs.com/netlib/search.html

- `http://www.irisa.fr/lagadic/publi-eng.html` presents the bibliography of our research team Lagadic.

5.2 Other *TEX to HTML translators

Most of them have been found by searching on the web.

- BibTEX to html translators

  - http://membres.lycos.fr/sgalland/english/tools/bib2html/, set of script LaTeX and Perl, by Stéphane Galland.
  - http://www-sop.inria.fr/epidaure/personnel/Gregoire.Malandain/codes/bibtex2html.html, C, by Gergoire Malandain.
  - http://www.cs.cornell.edu/home/wkiri/bib2html/, flex and C, by Kiri Wagstaff.
  - http://www.ctan.org/tex-archive/biblio/bibtex/utils/bibtools/, by David Kotz.
  - http://www.research.att.com/mff/strudel/Bib2Html/README.html
  - http://www.lri.fr/filliatr/bibtex2html/, Objective Caml, by Jean- Christophe Filliatre.

- LATEX to html translators

  - http://pauillac.inria.fr/hevea/, Objective Caml, by Luc Maranget.
  - http://www.latex2html.org, LaTeX2HTML

5.3 Other useful URLs

- http://www.w3.org/ the world wide web consortium (short, W3C)

- http://www.w3schools.com/ tutorials on HTML, XHTML, CSS, etc.

A The entries

A.1 Entry types

Standard entries [1, 2]

`@article` An article from a journal or magazine. Required fields: author, title, journal, year. Optional fields: volume, number, pages, month, note.

`@book:` A book with an explicit publisher. Required fields: author or editor, title, publisher, year. Optional fields: volume or number, series, address, edition, month, note.

`@booklet:` A work that is printed and bound, but without a named publisher or sponsoring institution. Required field: title. Optional fields: author, howpublished, address, month, year, note.

`@conference:` The same as @inproceedings, included for Scribe compatibility.

`@inbook:` A part of a book, which may be a chapter (or section or whatever) and/or a range of pages. Required fields: author or editor, title, chapter and/or pages, publisher, year. Optional fields: volume or number, series, type, address, edition, month, note.

`@incollection:` A part of a book having its own title. Required fields: author, title, booktitle, publisher, year. Optional fields: editor, volume or number, series, type, chapter, pages, address, edition, month, note.

`@inproceedings:` An article in a conference proceedings. Required fields: author, title, booktitle, year. Optional fields: editor, volume or number, series, pages, address, month, organization, publisher, note.

`@manual:` Technical documentation. Required field: title. Optional fields: author, organization, address, edition, month, year, note.

`@mastersthesis:` A Master's thesis. Required fields: author, title, school, year. Optional fields: type, address, month, note. The default type is Master's thesis.

`@misc:` Use this type when nothing else fits. Required fields: none. Optional fields: author, title, howpublished, month, year, note.

`@phdthesis:` A PhD thesis. Required fields: author, title, school, year. Optional fields: type, address, month, note. It is the same as @mastersthesis, but with the default type set to PhD thesis.

`@proceedings:` The proceedings of a conference. Required fields: title, year. Optional fields: editor, volume or number, series, address, month, organization, publisher, note.

`@techreport:` A report published by a school or other institution, usually numbered within a series. Required fields: author, title, institution, year. Optional fields: type, number, address, month, note. The default type is Technical report.

`@unpublished:` A document having an author and title, but not formally published. Required fields: author, title, note. Optional fields: month, year.

In addition to these entries, I add the following

`@audiovisual`: Some audiovisual material. The same as `@Misc`.

`@film`: Some audiovisual material. The same as `@audiovisual`.

A.2 Fields

`address`: Usually the address of the publisher or other type of institution. For major publishing houses, van Leunen [3] recommends omitting the information entirely. For small publishers, on the other hand, you can help the readger by giving the complete address.

`annote`: An annotation. It is not used by the standard bibliography styles, but may be used by others that produce an annotated bibliography.

`author`: The name(s) of the author(s), in the format described in the LATEX book.

`booktitle`: Title of a book, part of which is being cited. See the LATEX book for how to type titles. For book entries, use the title field instead.

`chapter`: A chapter (or section or whatever) number.

`crossref`: The database key of the entry being cross referenced. (similar to `cites`, `sa`, `seealso`).

`edition`: The edition of a book (for example, Second). This should be an ordinal, and should have the first letter capitalized, as shown here; the standard styles convert to lower case when necessary. Not implemented.

`editor`: Name(s) of editor(s), typed as indicated in the LATEX book. If there is also an author field, then the editor field gives the editor of the book or collection in which the reference appears.

`howpublished`: How something strange (a `@Misc` for example) has been published.

`institution`: The sponsoring institution of a technical report.

`journal`: A journal name.

`key`: Used for alphabetizing, cross referencing, and creating a label when the author information is missing. This field should not be confused with the key that appears in the \cite command and at the beginning of the database entry. However, the default key IS the key that appears in the \cite command and at the beginning of the database entry. If a key field is specified it replaces the default one.

`month`: The month in which the work was published or, for an unpublished work, in which it was written. You should use the standard three-letter abbreviation, as described in Appendix B.1.3 of the LATEX book.

`note`: Any additional information that can help the reader.

`number`: The number of a journal, magazine, technical report, or of a work in a series. An issue of a journal or magazine is usually identified by its volume and number; the organization that issues a technical report usually gives it a number; and sometimes books are given numbers in a named series.

`organization`: The organization that sponsors a conference or that publishes a manual.

`pages`: One or more page numbers or range of numbers, such as 42–111 or 7,41, 73–97 or 43+ (the '+' in this last example indicates pages following that don't form a simple range). To make it easier to maintain Scribe compatible databases, the standard styles convert a single dash (as in 7–33) to the double dash used in TEX to denote number ranges (as in 7–33).

`publisher`: The publisher's name.

`school`: The name of the school where a thesis was written.

`series`: The name of a series or set of books. When citing an entire book, the the title field gives its title and an optional series field gives the name of a series or multi-volume set in which the book is published.

`title`: The work's title, typed as explained in the LATEX book.

`type`: The type of a technical report (for example, Research Note).

`volume`: The volume of a journal or multivolume book.

`year`: The year of publication or, for an unpublished work, the year it was written. Generally it should consist of four numerals, such as 1984.

Bib2html provides specific and non standard (optional) fields

`x-editorial-board =yes` for `inproceedings` entries ; yes if an editorial board exists for the conference, if no the entry will considered as `misc`

`x-proceedings =yes` for `inproceedings` entries ; yes if proceeding are published.

`x-popularization = yes` if yes create a new category for popularization article. Considered only if

        `@define{generate.popularization.pages.yes}`

has been set in the config file.

`x-invited-conference = yes` if yes create a new category for invited article. Considered only if

        `@define{generate.invited.pages.yes}`

has been set in the config file.

`x-international-audience =yes` if no create a new category for national article/conference paper according to the entry type (article/inproceedings). Considered only if

        `@define{generate.national.journal.pages.yes}`

and/or

$$\texttt{@define\{generate.national.conferences.pages.yes\}}$$

has been set in the config file.

We can provide a description of other field that may exist. The ones that are already recognized are:

`keyword` To specify keywords. Keywords are separated by commas (",") or semicolon (";")

`keywords` id.

`pdf` To specify the way to the pdf file, may be something like `ftp://www....`.

`postscript` To specify the way to the postscript file, may be something like `ftp://www....`.

`url` To specifiy some url, may be something like `http://www....`.

`abstract` The abstract of the article.

`comments` Some comments ...

`alt...` Always ignored.

`opt...` Always ignored.

`archive` Always ignored

`isbn` Always ignored

`issn` Always ignored

`location` Always ignored

`nickname` Always ignored

`precedes` Always ignored

`site` Always ignored

`succeeds` Always ignored

`update` Always ignored

B   Authors or editors name

This paragraph is mainly extracted and adapted from [2].

The text of an author or editor field represents a list of names. The bibliography file entry simply tells BibTeX what the name is. You should type an author's complete name and let the Bib2html style decide what to abbreviate.

Most names can be entered in the obvious way, either with or without a comma, as in the following examples.

```
"John Paul Jones" "Jones, John Paul" "Ludwig von Beethoven"
"von Beethoven, Ludwig"
```

Some people have multiple last names - for example, Per Brinch Hansen's last name is Brinch Hansen. His name should be typed with a comma:

```
"Brinch Hansen, Per"
```

To understand why, you must understand how BibTeX handles names (for what follows, a "name" corresponds to a person).

Each name consists of four parts: First, von, Last, and Jr; each part consists of a (possibly empty) list of name-tokens. For example the First part of Per Brinch Hansen's name has the single token "Per"; the Last part has two, "Brinch" and "Hansen"; and the von and Jr parts are empty. If you had typed

```
"Per Brinch Hansen"
```

instead, BibTeX would erroneously think "Brinch" were a First-part token, just as "Paul" is a First-part token in "John Paul Jones".

If you want BibTeX to consider something a single token, enclose it in braces. You should do this, for example, if a comma is part of a name:

```
"{Barnes and Noble, Inc.}" "{Barnes and} {Noble, Inc.}"
"{Barnes} {and} {Noble,} {Inc.}"
```

The braces surrounding the comma keep "Inc." from being interpreted as a First token; this name has only a Last part, with either one, two, or four tokens (there must be a space separating the tokens in the second and third forms). Probably the second form is slightly more meaningful, but don't lose sleep over this since only rarely will an institution be an author or editor.

So the two names

```
"von Beethoven, Ludwig" "{von Beethoven}, Ludwig"
```

are considered by BibTeX to be different names. In the first, "Beethoven" is the Last part and "von" is the von part; in the second, which in this case happens to be incorrect, the Last part has a single token and there's no von part. The bibliography style will probably print both the same, but it may alphabetize and label them differently.

"Juniors" pose a special problem. Most people having "Jr." in their name precede it with a comma. Such a name should be entered as follows:

```
"Ford, Jr., Henry"
```

However, a few people do not use a comma. They are handled by considering the "Jr." to be part of the last Last token:

```
"{Steele Jr.}, Guy L." "Guy L. {Steele Jr.}"
```

This name has no Jr part.

To summarize, you may type a name in one of three forms:

```
"First von Last" "von Last, First" "von Last, Jr, First"
```

You may almost always use the first form; you shouldn't if either there's a Jr part or the Last part has multiple tokens but there's no von part.

If there are multiple authors or editors, their names must be separated by the word "and", surrounded by spaces, not enclosed in braces:

```
AUTHOR = "Ralph Alpher and Bethe, Hans and George Gamow"
```

Since BibTeX interprets commas as separating parts of a name and "and" as separating names themselves, this example has three coauthors: Ralph Alpher, Hans Bethe, and George Gamow. If the word "and" appears as part of a name, it must be enclosed in braces, as in the example of "Barnes and Noble, Inc." given above. If you have too many names to list in a field, you can end the list with "and others"; the standard styles appropriately append an "et al."

BibTeX's rules are actually a bit more complicated than indicated here, but this description will suffice for most names.

References

[1] Leslie Lamport. LATEX: A Document Preparation System. Addison-Wesley, 1986.

[2] Leslie Lamport. LATEX: A Document Preparation System. Addison-Wesley, 1986. Appendix B : The bibliography database http://nwalsh.com/tex/texhelp/BibTeX.html

[3] Oren Patashnik. BibTEXing. Documentation for general BibTEX users, 8 February 1988.

[4] Mary-Claire van Leunen. A Handbook for Scholars. Knopf, 1979.