Bibliographies, LaTeX and friends

Guido Milanese

Abstract

This lecture deals with the treatment of bibliographies within a LaTeX framework and workflow. A comparison of BibTeX bibliography format with other widely used formats shows that BibTeX has several advantages. The classical bibtex programme is now obsolete, and biblatex + biber offer a highly customisable choice for bibliographies in any research area. GUI environments are also discussed, as well as possible future developments.

Sommario

Questa conversazione si occupa del trattamento delle bibliografie all'interno di un ambiente di lavoro basato su LaTeX. Una comparazione del formato bibliografico di BibTeX con altri formati usati ampiamente mostra che BibTeX offre parecchi vantaggi comparativi. Il classico programma bibtex è da ritenersi obsoleto, poiché biblatex + biber offrono un ambiente altamente flessibile per generare bibliografie in ogni area di ricerca. Vengono anche discussi agli ambienti grafici e le prospettive per il futuro.

1 BibTeX and other formats

Let us consider a working example. One of the most important online catalogues is "Library Hub Discover", a British online resource that has recently replaced Copac and SUNCAT, offering access to 117 catalogues for more than 40 million of bibliographic items (`https://discover.libraryhub.jisc.ac.uk`). "Library Hub Discover" (henceforth referred to as "LHB") gives the user the opportunity of exporting bibliographic items into several widely used formats. Let us compare the BibTeX file generated by LHB with the Endnote–Zotero file generated by the same online resource (actually it is a RIS card[1]: see fig. 1. At first glance, the two formats show the same amount of information (the abstract field is missing from RIS, but this is a problem of the filter used by "Library Hub Discover"); BibTeX, however, always generates a "label" for any item – this is not an option; this feature is very important in order to link items, which makes possible to biblatex to generate highly sophisticated bibliographic entries. The three formats referred to (RIS, Endnote/Zotero, and BibTeX) are widely

used on the Internet, and it is fairly easy to convert among them: BibTeX is known e.g. to Google Books and Google Scholar. From a practical point of view, the three formats are equally successful.
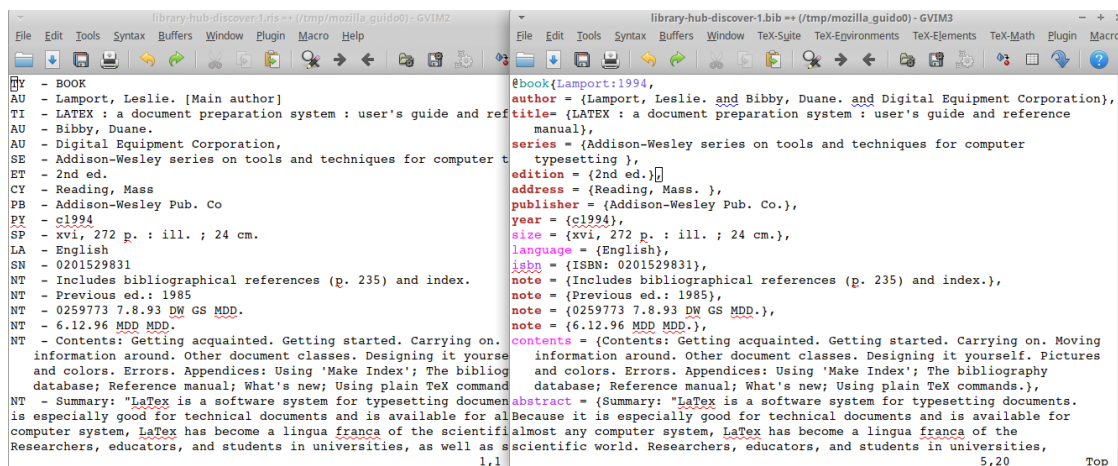
2 The BibTeX format

As an example of BibTeX, we can use a very basic card generated by LHB:

```
1 @book{Lamport:1994,
2 author = {Lamport, Leslie.},
3 title= {LATEX : a document preparation system},
4 edition = {2nd ed.},
5 address = {Reading, Mass. },
6 publisher = {Addison-Wesley},
7 year = {1994},
8 language = {English},
9 isbn = {ISBN: 0201529831},
10 isbn = {ISBN: 9780201529838},
11 note = {User's guide and reference manual.},
12 location = {University of Sussex Library},
13 }
```

There are some minor mistakes (see section 9); now, let us read the information provided. Line 1 is very important, providing a unique label for each entry; think of this line as the number plate of a car: you can have an archive of thousands of entries, but each one must have a unique label. Lines 2 to 10 feature the most important bibliographic information, such as author, title, and so on; lines 9 and 10 are not a useless duplicate, because the two ISBN code formats are listed, the former of 10 digits, and the latter of 13 digits. It is worth mentioning that some of the fields listed in this card are actually biblatex fields, unknown to the original BibTeX format: I refer to the "ISBN", "language" and "location" fields. Line number 11 is used by the LHB filter as a sort of black hole, because in this case the "note" field is used for what is really the subtitle of Lamport's book. The last line informs of the library location of this book. All the fields must be balanced: each piece of information must be enclosed in curly braces and followed by a comma, and the whole entry must be opened and closed by a curly braces pair. Uppercase and lowercase are immaterial: `AUTHOR` and `author` and `Author` are valid and can be mixed freely. Consistency is recommended for stylistic reasons, but it does not change the behaviour of programmes using this format.

There are other fields that can prove very useful to scholars, such as `abstract`, `keywords`, `annotation`... Any parser using BibTeX files can perform powerful researchers, combining the infor-

---

1. See `https://en.wikipedia.org/wiki/RIS_(file_format)`; `https://en.wikipedia.org/wiki/EndNote`; `https://en.wikipedia.org/wiki/Zotero`.

Figure 1: Endnote/Zotero and BibTEX formats

mation provided by fields – such as a query looking for of books concerning typesetting (`keywords` field) published between 2010 and 2018 (`year` field), and featuring the word «useless» in the `annotation` field. This research will extract all the books about typesetting, published between 2010 and 2018, regarded as useless by the owner of the database. There is of course great freedom in the use of "personal" fields.

3 Bibliographic data and output formats

BibTEX fields are in fact a particular format of tags. It is indeed quite simple to export a BibTEX file into another format called BibTEXML – a format designed in 2007 but not very successful[2]:

```
<bibtex:entry xmlns:bibtex="http://bibtexml.sf.net/"
    bibtex:id="Lamport:1994">
<bibtex:book>
<bibtex:author>Lamport, Leslie. </bibtex:author>
<bibtex:title>LATEX : a document preparation system
    </bibtex:title>
<bibtex:edition>2nd ed.</bibtex:edition>
<bibtex:address>Reading, Mass. </bibtex:address>
<bibtex:publisher>Addison-Wesley </bibtex:publisher>
<bibtex:year>1994</bibtex:year>
<bibtex:language>English</bibtex:language>
<bibtex:isbn>ISBN: 0201529831</bibtex:isbn>
<bibtex:isbn>ISBN: 9780201529838</bibtex:isbn>
<bibtex:note>User's guide and reference manual.
    </bibtex:note>
<bibtex:location>University of Sussex Library
    </bibtex:location>
</bibtex:book>
</bibtex:entry>
```

The information provided by this XML format is exactly the same featured by the internal tagging system of BibTEX – just heavier and more difficult to read; BibTEX is simply a tagging system, just a very light and simple one. The purpose of

---

2. See `https://sourceforge.net/projects/bibtexml/files/BibTeXConverter/`.

any tagging system, as XML which is familiar to anyone using a computer nowadays, is to separate data from representation: BibTEX is no exception: there are literally thousands of bibliographic formats required by publishers and journals, and it is vital to keep information on a book, article or whatever separated from the output needed for a particular publication. In this incredible variety of output formats, a very basic arrangement can be the following one:

1. the number reference system: each item referred to in a book or article is labelled with a number. At the end of the text, a list will show the complete bibliographic information. For example, a citation from page 32 of Lamport's book, provided it receives e.g. number 6 in the final list, will be «[6, 32]».

2. the author-year systems: from the BibTEX file, fields `author` and `year` will be extracted and citations will take this form: «Lamport 1994, 32» or «Lamport, 1994, 32» or «LAMPORT 1994, 32». . . Again, at the end of the book or article the reader finds the complete information about the titles referred to. If there are more than one title published by the same author in the same year, the year field is completed with a letter, such as, e.g., «Lamport 1994a, 32».

3. the author-title systems: from the BibTEX file, fields `author` and `title` will be extracted and citations will take this form: «Lamport, *LATEX*, 32», or a variety of this scheme. For this kind of system, it is advisable to add a `shorttitle` field, in order to obtain reasonably short references;

4. the varieties of the "verbose" systems, very popular in the humanities, particularly in Italy. Generally, the first citations is complete, given

in a footnote, and the following ones take the form of an author-title system. Normally with this kind of reference system a final bibliography is not required by journals, although it remains necessary for books.

4 BibTeX, **bibtex**, **biber** and **biblatex**

Since 1985, at the very beginning of LaTeX, the programme bibtex (with the same name of the bibliographic format) was designed by Lamport himself and Oren Patashnik in order to obtain from the output format agnostic bibliographic files provided by BibTeX the kind of output format required by the final user, i.e. by the journal or publishing house[3]. This programme never reached version 1.0 – the most recent version at CTAN (July 2019) is version 0.99d. bibtex is a good piece of software, but at the beginning of this century it was clear that it had serious troubles with Unicode and that it was not easily customisable. There were several attempts to improve the situation[4]: a very promising tool is (was) bibulous, a Python script with a simple, intuitive approach to develop new styles – but apparently the development ceased years ago[5]. The standard replacement for bibtex is now biber + biblatex[6]. The former is used to sort data and to deal with labels, while all the process of generating the desired output format is performed by biblatex. Advantages of the new pair of programmes in comparison with old bibtex are, from the point of view of the final user:

- no problems with Unicode – sorting can be adapted to the language of the document and also single entries can be made aware e.g. of the hyphenation required by the language of that particular entry;

- it is possible to build new styles, which is particularly important in countries, such as Italy, where each publisher wants to receive files formatted according to the particular styles of this of that particular journal or series;

- more flexible use of the `crossref` field;

- full support for related entries, e.g. "reprinted as..." or "translation of...";

- the new `XDATA` field, a sort of container for one or more fields, e.g. name of a publishing house and place of publication, or name of a series of books, publishing house, place of publication. This can be convenient to avoid useless repetition of information in a file.

- suppression/inclusion of fields is straightforward, even for some entry types.

No heavy tailoring of old BibTeX files is necessary to use the new features. However, to take full advantage from biber + biblatex some additional fields are advisable, and some old field names must be adapted. If we take the basic example downloaded from LHB (see sec. 1) we could adapt it as follows:

```
@book{Lamport:1994,
author = {Lamport, Leslie.},
title= {{LATEX} A document preparation system},
EDITION = {2},
LOCATION = {Reading, Mass. },
publisher = {Addison-Wesley},
year = {1994},
LANGID = {AMERICAN},
language = {english},
isbn = {ISBN: 0201529831},
isbn = {ISBN: 9780201529838},
subtitle = {User's guide and reference manual},
LIBRARY = {University of Sussex Library},
}
```

I have capitalised the fields that have to be adapted to use with biber + biblatex. The `edition` field must contain only a number, in order to format the field according to the needs of languages and styles, e.g.: «1994, 2$^{\mathrm{nd}}$ ed.», or «II edizione, 1994», or «1994²»; the new field `langid` is used by biblatex in order to format entries according to hyphenation and, if required, to all the rules of the language of that particular entry: for example the strings used to replace "translation" or "reprint" in languages other than American English. The old field `language` can still be used for the language of the work, but will not affect the rules for hyphenation etc. Since British English and American English have their local rules, it is advisable to use «american» or «british» for the `langid` field, avoiding the generic label «english». One possible ambiguity is the field `location`: traditional BibTeX used it for library location, while biblatex uses the same field name as an alias for `address`; the name of libraries holding the book is archived as `library`. In this example, the field `titleaddon` is used to describe the real hierarchy of Lamport's book title.

5 How to survive BibTeX

As it happens with any tagging system, BibTeX files are error-prone. Even if good text editors, such as (g)vim, do an excellent job checking syntax, it is too easy to forget to close a bracket or to add a comma at the end of a field. A good GUI interface makes for BibTeX makes the users' life much easier: see `https://en.wikipedia.org/wiki/Comparison_of_reference_management_software` for a comparison of the features offered by reference management programmes. Many among them have a Free Software licence, and the great majority of these programmes either use

---

3. See Lamport (1994, 69-72) and more recently Mittelbach *et al.* (2004, 683-812).
4. See Hufflen (2011).
5. See `https://pypi.org/project/bibulous`. The latest version is dated 2015.
6. A recent updated manual is Voss (2011); on biblatex, see Lehman's manual (Lehman *et al.*, 2018).

Figure 2: Jabref graphical user interface

BibTEX files or can import/export from this format – a choice is basically a matter of preferences and of personal taste. However, I think that some points may be taken into serious account:

- some very popular programmes are proprietary software. Those who prefer to use Free Software would probably avoid using for example Mendeley (`https://www.mendeley.com`);

- some programmes offer the user an online space to archive data. Those who prefer local storage should consider this point; see again Mendeley;

- does a programme use BibTEX to archive data, or just to import/export data? For example, Zotero can archive data on local storage, but using its own SQL format. Programmes using BibTEX directly have the great advantage of archiving data in a plain text file, that is not bound to a particular GUI programme and to its format;

- those users who work on several platforms, e.g. Windows at home and Linux at work, will probably prefer programmes being actively developed for different platforms. For example, pybibliographer (`https : / / pybliographer . org`) runs only under Linux, and this is an obvious problem for many users.

At the moment of writing, Jabref is probably the most complete GUI environment for using BibTEX files (`http://www.jabref.org/`). It offers a good support also for the new features provided by biblatex and links very effectively BibTEX entries to local or remote files, such as PDF files or other forms of archived information, and to texts written by users as their own comments. See fig. 2: the green square shows that a file containing notes by the user is available; the traditional PDF icon opens a local PDF; the third icon, with the form of a little globe, opens a link on the web. Even if these features are offered by other programmes, the advantages of Jabref are

1. to use a simple BibTEX file to archive information

2. to be platform-independent.

In other words, while a Zotero archive needs Zotero to work properly, the work done by Jabref can be edited and modified using any other GUI interface, such as e.g. pybibliographer, or any text editor, even Notepad; the file produced is a completely transparent, standard BibTEX archive.

Citing an entry from a BibTEX archive is made easy by all these GUI interfaces, either clicking on a particular item or hitting a combination of keys. To cite page 32 of Lamport's book, the LaTeX file will feature the required citation in the neutral form `\cite[32]{Lamport:1994}`. In the final output, the citation will be formatted, as previously described, according to the desired style.

6 Using BibTEX to build a scholarly archive

Since BibTEX entries are highly customizable, it is possible to use this format to build a complete environment to keep trace of one's scholarly work and make it available in the future. Let me use a real example taken from one of my databases:

```
@ARTICLE{Dalfen:Das-Gebet,
  author = {Dalfen, Joachim},
  journal = {Hermes},
  keywords = {fant, fhel, stoa},
  pages = {174-183},
  shorttitle = {{D}as {G}ebet des {K}leanthes},
  title = {{D}as {G}ebet des {K}leanthes an
      {Z}eus und das {S}chicksal},
  volume = {99},
  langid = {german},
  year = {1971},
  annotation = {2018-Cleante},
  file = {Dalfen\:Das-Gebet.md:Neworbis/
    Dalfen\:Das-Gebet.md:Markdown},
  issn = {00180777},
  pdf = {Dalfen:Das-Gebet.pdf},
  timestamp = {24.03.2018},
  url = {http://0-www.jstor.org.opac.
  unicatt.it/stable/4475677}
}
```

I am using the `annotation` field to archive a keyword showing for which publication I have used this particular article or book; I keep a list of these keywords in a separate file and in the future I will be able to know which publications I had been using for this particular research. The field `timestamp` is useful to know when I read and archived this work; the field `keywords` is obviously useful if and only if the user is consistent in naming keywords: if, for example, I would sometimes use "stoa" and

sometimes "stoicism", an attempt to extract all the entries concerning Stoicism would fail. This is one of the instances where consistency is crucial for a successful use of this format.

«Digital tools have yet to develop models for displaying and replicating the self-reflexive operations of bibliographical tools, which alone are operations for thinking and communicating – which is to say, for transforming storage into memory, and data into knowledge. We have to design and build digital environments for those purposes» (McGann, 2016, 363). I think that an imaginative use of BibTeX archives already offers a serious answer to these questions.

7   The **philosophy** package

A tremendous amount of bibliographic styles has been produced by journals and publishers: more than 9000 "Citation Style Language" (CSL) bibliographic styles are now available (`https://citationstyles.org/`). The CSL format is used by bibliography managers such as Zotero and Mendeley: unfortunately, BibTeX users are offered a good choice of styles, but not comparable to this incredible wealth. The Italian scholar Ivan Valbusa has developed a package aimed particularly at the needs of humanists, who every day struggle with the various and exasperating bibliographic requirements of publishers and journals. This package, called biblatex-philosophy (`https://ctan.org/pkg/biblatex-philosophy`), now at version 1.9.8a, features three basic styles, with a rich offer of options, whose combination and selection is likely to face, if not all, at least the large majority of the requirements of journals and publishers[7].

8   The **zblbuild** package

This utility offers a GUI aimed at helping the generic user building his own bibliographic format. A series of GUI widgets (question and checkboxes) select among the various possible formats and generate a biblatex call featuring all the necessary options. The programme is particularly tailored to work along with Valbusa's philosophy package[8]. See an example at fig. 3.

9   Other utilities

BibTeX is a complex system, and in more than 30 years of life, which is a geological era in this field, it has generated many utilities to maintain archives or to adapt them to new formats such as biblatex. Although a couple of dozens of these utilities are listed by CTAN, it is worth noticing

---

7. See Valbusa's manual on CTAN: Valbusa (2018), and – for a previous version – Valbusa (2010).

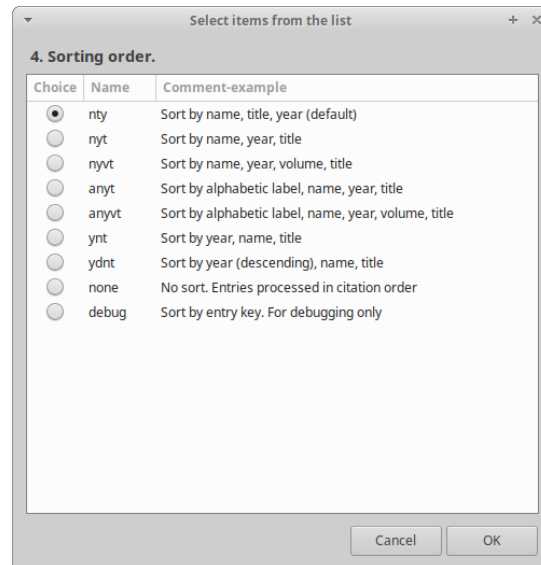8. See `https://ctan.org/pkg/zblbuild`. For a previous version see Milanese (2015).

FIGURE 3: Blbuild – a widget

that many among them are old, written even more than 25 years ago, and cannot be useful within a modern framework. The most complete I know among these utilities is called BibTool, developed and maintained by Gerd Neugebauer since 1995. This rich programme features a wide variety of utilities: sort an archive, extract items referred to in an article or book, rename fields – for example, the `location` field of legacy BibTeX files that is to be renamed to `library` if the archive is going to be used by biblatex. A real "must": the list of possible manipulations listed in the CTAN page are the following ones[9]:

- Pretty-printing data bases;

- Syntactic checks with error recovery;

- Semantic checks;

- Sorting and merging of data bases;

- Generation of uniform reference keys according to predefined rules or according to user specification;

- Selecting references used in one publication which are found by analysing an "aux" file;

- Controlled rewriting of fields utilising regular expressions to specify the rewriting rules;

- Macro (String) expansion to eliminate the need of extra string definitions;

- Collecting statistics about one or more databases.

---

9. See `https://www.ctan.org/pkg/bibtool`.

There are also many other little utilities that may prove worth trying. For example, frequently, BibTEX entries produced by automatic systems such as Google or Library Hub Discover contain little mistakes (spurious spaces, wrong labels) that should be manually corrected. The programme copac-clean (labelled from the old name of Library Hub Discover) improves the output produced by those systems, cleaning minor errors and adapting field names such as `location` and `language` to the needs of a present-day biblatex installation[10].

10   Using BibTEX without LATEX

BibTEX was designed to be used along with LATEX, but it can be effectively employed also within other environments and workflows because of its very simple tagged structure. The best example I know is offered by markdown, particularly in its pandoc variety. This format is very easy to read, needs no particular piece of software to edit files, and *via* the pandoc filter can generate a variety of output formats, including HTML, XML-TEI, ODT, DOCX, and LATEX. Direct generation of a PDF file is possible if a TEX system is installed[11].

The interesting point is that pandoc can make use of a standard BibTEX archive but compile a PDF file using a CSL style format (see section n. 7). If preferred, the user can compile the bibliography calling biber + biblatex or even legacy bibtex (which is clearly not advisable). For example, consider this minimal MD file:

```
---
title: A Test
author: John Tester
bibliography: 'test.bib'
---

This has been noticed by Kenneth Levy in a
seminal article [@Levy:ItalianNeophytes 34],
and in more recent literature.
```

Running pandoc as follows, the filter defaults to a generic author–year style (backslash for readability only):

```
pandoc --filter pandoc-citeproc\
bibliotest.md  -s -o bibliotest.pdf
```

The final output is displayed at fig. 4. The MD file is processed by pandoc, silently calling one of the varieties of LATEX compilers. Since no option is listed, pdflatex and the standard fonts are used, but the bibliography is produced calling the pandoc-citeproc filter, not biber + biblatex. Adding the name of a particular CSL style its features are used by the bibliographic filter, e.g.:

10. See `https://ctan.org/pkg/copac-clean`
11. Markdown – designed in 2004 by John Gruber and Aaron Swartz – is riding on a wave of popularity, also because pandoc makes the conversion to output formats a matter of a few keystrokes: LHB lists 20 manuals on Markdown, all published between 2013 and 2019. For pandoc see the online documentation (`https://pandoc.org/`).
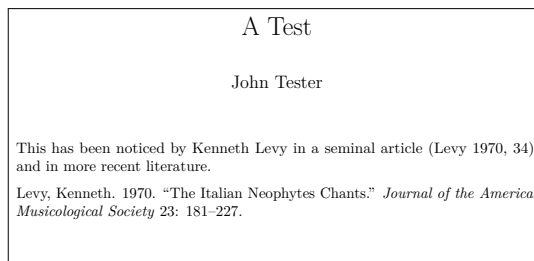

> A Test
>
> John Tester
>
> This has been noticed by Kenneth Levy in a seminal article (Levy 1970, 34), and in more recent literature.
>
> Levy, Kenneth. 1970. "The Italian Neophytes Chants." *Journal of the Americal Musicological Society* 23: 181–227.

FIGURE 4: Pandoc + LATEX: default CSL

> A Test
>
> John Tester
>
> This has been noticed by Kenneth Levy in a seminal article,[1] and in more recent literature.
>
> LEVY Kenneth, *The Italian Neophytes Chants*, in «Journal of the Americal Musicological Society» 23 (1970), 181–227.
>
> _____
> [1]K. LEVY, *The Italian Neophytes Chants*, in «Journal of the Americal Musicological Society» 23 (1970), 181–227: 34.
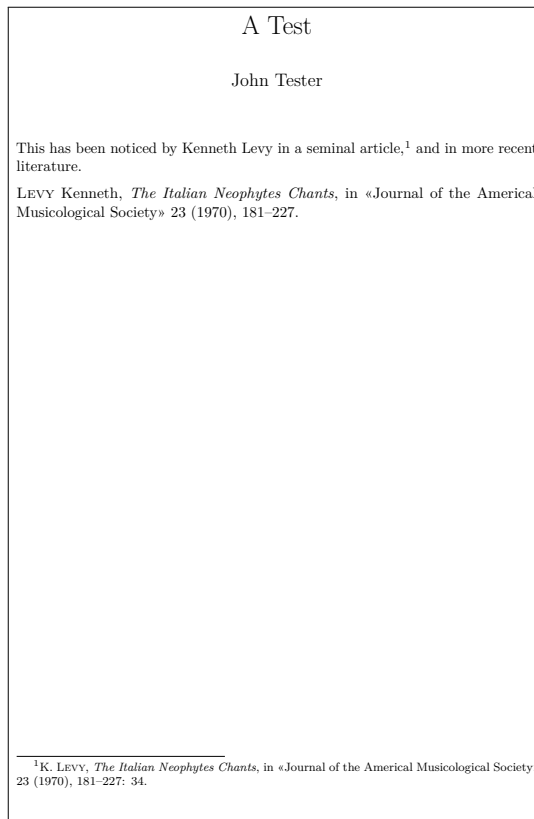
FIGURE 5: Pandoc + LATEX: verbose CSL

```
pandoc --filter pandoc-citeproc\
bibliotest.md --csl=universita-pontificia\
-salesiana-it.csl -s -o bibliotest.pdf
```

This is an Italan "verbose" style, and the filter instructs LATEX accordingly: see fig. 5. Pandoc can also use biber + biblatex: which one is the better? The pandoc-citeproc filter is attractive because it can make use of the huge collection of CSL styles; on the other hand, biber + biblatex offer a level of fine-tuning that is truly unparalleled by other systems.

11   The future of BibTEX

Again, the variety of styles is a challenge: the CSL repositories present the final user who is not a specialist in computing with +9000 styles; select the name of the needed journal or publishing house,

and that's all. The Zotero page features also a clever system offering the user a way to find a style strictly similar to the one required by his publisher, if not already listed. BiBTEX users can choose within a certain amount of choices (around 180–200), but this is ridiculous in comparison with the wealth offered by the Citation Style Language repositories. A good project would be a translation of all these styles in order for them to be usable with biblatex – or, even better, to make Citation Style Language directly digestible by biblatex. BiBTEX archives + CSL formatting, thanks to pandoc, is a good example for the future.

The reason of this relative paucity of styles lies in the historical roots of LATEX and of TEX itself, originally aimed to users in the field of mathematics and informatics. The present writer, who happens to be a humanist, believes that LATEX offers e.g. to linguists, historian, and philosophers, an ideal environment to maximise productivity without wasting time and energy in trivial matters – formatting texts and, in this case, formatting bibliography: «Worrying too much about formatting and not enough about content» is the mistake that people should stop making, according to Leslie Lamport himself (Lamport, 2000, 51). A more user-friendly approach is a *desideratum* badly needed in order to make LATEX + BiBTEX a real opportunity in comparison with the ubiquitous "Word + Zotero" approach. Markdown is very good for writing a paper, but the final output depends either on biber + biblatex or on pandoc-citeproc; therefore the problem of bibliography is not completely solved, as we saw above, because CSL filters do not offer the amount of details and features offered by biblatex, which on its turn has the problem of ready-to-use styles. A serious interest on the part of one or more universities (e.g. a financed research project) is needed, because the development of this complex family of software cannot rely only on the time and energy generously offered by individuals, skilled and competent as they may be.

References

Hufflen, Jean-Michael (2011). «A comparative study of methods for bibliographies». *TUGboat*, **32** (3), pp. 289–301. https://www.tug.org/TUGboat/tb32-3/tb102hufflen.pdf.

Lamport, Leslie (1994). *LATEX. A Document Preparation System. User's Guide and Reference Manual*. Addison–Wesley, Reading, Massachusetts.

— (2000). «How (La)TeX changed the face of Mathematics». *Mitteilungen der Deutschen Mathematiker-Vereinigung*, (1), pp. 49–51.

Lehman, Philipp, Philip Kime, Audrey Boruvka and Joseph Wright (2018). *The biblatex Package. Programmable Bibliographies and Citations. Version 3.12*. http://mirrors.ctan.org/macros/latex/contrib/biblatex/doc/biblatex.pdf.

McGann, Jerome (2016). *Marking Texts of Many Dimensions*, John Wiley & Sons, Ltd, Chichester, West Sussex, UK, capitolo 25, pp. 358–376. Blackwell companions to literature and culture 93.

Milanese, Guido (2015). «Zbl-build: a GUI interface for Biblatex». *ArsTEXnica*, **20**, pp. 31–34.

Mittelbach, Frank, Michel Goossens, Johannes Braams, David Carlisle, and Chris Rowley (2004). *The LATEX Companion. Second edition*. Addison–Wesley Publishing Company, Reading, Massachusetts.

Valbusa, Ivan (2010). «Creare stili bibliografici con biblatex: l'esperienza del pacchetto biblatex-philosophy». *ArsTEXnica*, **9**, pp. 39–50.

— (2018). *The biblatex-philosophy bundle v1.9.8a*. http://www.ctan.org/pkg/biblatex-philosophy.

Voss, Herbert (2011). *Bibliografien mit LATEX*. Lehmanns Media, Berlin. https://books.google.it/books?id=SgNACgAAQBAJ&pg=PA228&dq=biblatex&hl=it&sa=X&ved=0CCQQ6AEwAGoVChMI-dH5o8vJxwIVR1kUCh1uxwKa.

▷ Guido Milanese
UCSC – Milano; USI – Lugano
guido dot milanese at unicatt dot it