Biblet

A portable \BIBTEX\ bibliography style for generating highly customizable XHTML

Tristan Miller
German Research Center for Artificial Intelligence
Erwin-Schrödinger-Straße 57
67663 Kaiserslautern

tristan.miller@dfki.de
Introduction
Problem

How can I put a list of my publications on my website?
Possible solutions

- manually compose HTML
  - fine-tuned control of formatting
  - difficult to change bibliography style
  - must maintain separate \texttt{B\TeX} and HTML files
Possible solutions

- manually compose HTML
  - fine-tuned control of formatting
  - difficult to change bibliography style
  - must maintain separate \texttt{BibTeX} and HTML files

- convert \texttt{\LaTeX} to HTML with, \textit{e.g.}, \texttt{latex2html}
  - easy to change bibliography style
  - poor control of formatting
  - poor handling of special characters
Possible solutions

- manually compose HTML
  - fine-tuned control of formatting
  - difficult to change bibliography style
  - must maintain separate \LaTeX{} and HTML files

- convert \LaTeX{} to HTML with, e.g., \texttt{latex2html}
  - easy to change bibliography style
  - poor control of formatting
  - poor handling of special characters

- convert \BibTeX{} to HTML with, e.g., \texttt{bibtex2html} or \texttt{bib2html}
  - difficult to change bibliography style
  - poor control of formatting
  - poor handling of special characters
Introducing Biblet

biblet [from *Bible* + -et, diminutive]: a small library

- Biblet is a set of \texttt{B\LaTeX} bibliography styles (\texttt{bst} files) which generate (X)HTML directly from \texttt{B\LaTeX} databases
- highly portable: requires only \texttt{B\LaTeX}
- appearance is highly customizable: all formatting controlled with CSS
- special text characters converted to Unicode or HTML entities whenever possible
- customizable graphical hyperlinks to PostScript, PDF, DVI, and HTML versions
Development status

- Biblet is currently in active development
Development status

- Biblet is currently in active development
- not yet stable, but very much usable!
Development status

- Biblet is currently in active development
- not yet stable, but very much usable!
- in this talk:
  - how Biblet was developed
  - how to use Biblet
  - examples of beautiful Biblet bibliographies
Using B\LaTeX

How is it done?

CSS formatting examples
Basic \LaTeX\ workflow

LaTeX \rightarrow \ aux \rightarrow \ BibTeX \rightarrow \ bbl \rightarrow \ LaTeX

\rightarrow \ dvi \rightarrow \ dvips

\rightarrow \ ps

.c2005 Tristan Miller Biblet — June 17, 2005 - p. 8/31
\documentclass{article}

\begin{document}

I wrote \cite{miller2002why}.

\bibliographystyle{plain}
\bibliography{foo}

\end{document}
@Article{miller2002why,
  author = {Tristan Miller},
  title = {Why {I} Will Never Have A Girlfriend},
  journal = {The Annals of Improbable Research},
  year = {2002},
  volume = {8},
  number = {3},
  pages = {13-17},
}
\relax
\citation{miller2002why}
\bibstyle{plain}
\bibdata{foo}
\begin{thebibliography}{1}

\bibitem{miller2002why}
Tristan Miller.
\newblock Why {I} will never have a girlfriend.

\end{thebibliography}
Introduction

Using BibTeX

Basic BibTeX workflow
- foo.tex
- foo.bib
- foo.aux
- foo.bbl

foo.dvi

Eureka moment

Using Biblet

How is it done?

CSS formatting examples

References

**Eureka moment**

\textit{B\LaTeX\xspace uses a \texttt{.bst} style to convert \texttt{foo.bib} into the completely different-looking \texttt{foo.bbl} file in \texttt{\LaTeX} syntax:}

\begin{verbatim}
\begin{thebibliography}{1}

\bibitem{miller2002why}
Tristan Miller.
\newblock Why {I} will never have a girlfriend.

\end{thebibliography}
\end{verbatim}
Therefore, why not make a .bst style to convert foo.bib into a foo.bbl file in HTML syntax?

```html
<html>
  <head><title>My publications</title></head>
  <body>
    <div class="bibitem">
      Tristan Miller.
      Why I will never have a girlfriend.
      <em>The Annals of Improbable Research</em>,
    </div>
  </body>
</html>
```
Using Biblet

Sprucing things up

How is it done?

CSS formatting examples
Biblet workflow

Introduction

Using BibTeX

Using Biblet

Biblet workflow
- .bib
- .bst
- .aux
- BIBTEX
- .bbl
- rename
- .html
- .css
- web browser

How is it done?

CSS formatting examples

foo.aux
foo.bbl
foo.html
foo.css
foo.html with foo.css
Rather than creating a .tex file, the user creates a .aux file directly.

The user types a \texttt{\citation} command for each bibliography item, or \texttt{\citation{*}} to include all items.

For \texttt{\bibstyle}, the user specifies one of the Biblet .bst files.

\begin{verbatim}
\relax
\citation{miller2002why}
\bibstyle{blplain}
\bibdata{foo}
\end{verbatim}

The user runs B\TeX\ on the .aux file as usual.

The output .bbl is actually an HTML document; it can be renamed and opened in any web browser.
<!DOCTYPE html PUBLIC '−//W3C//DTD XHTML 1.0 Strict//EN'
'http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd'>
<html xmlns='http://www.w3.org/1999/xhtml' xml:lang='en' lang='en'>
<head><title>My publications</title></head>
<body>
<h1>My publications</h1>
<div class='bib-bibliography'>
<h2 class='bib-year' id='year-2002'>2002</h2>
<ul>
<li class='bib-bibitem' id='cite-miller2002why'>
<div class='bib-article'>
<p>
<span class='bib-author'>Tristan Miller.</span>
<span class='bib-title'>Why I will never have a girlfriend.</span>
</p>
</div>
</li>
</ul>
</div>
</body>
</html>
Using BibTeX

 foo.html

Introduction

Using Biblet

foo.html

Sprucing things up

foo.css

foo.html with foo.css

How is it done?

CSS formatting examples

My publications

2002

Sprucing things up

- Because virtually every element of a bibliography item is encapsulated in an HTML element, it is easy to apply formatting styles with CSS.
- A `.bst` style can also output arbitrary HTML at the beginning or end of a bibliography, or even inbetween entries.
- The only problem: there is no way to pass arguments to BibTeX, so any formatting that cannot be done in CSS must be done by editing the `.bst` file or resulting `.html` file.
.bib-article {
  background-color: #ffaaaa;
}

.bib-author {
  font-weight: bold;
}

.bib-article em {
  font-style: normal;
  text-decoration: underline;
}
foo.html with foo.css

My publications

2002

How is it done?
Overview

I started by going through Patashnik’s `plain.bst` and replacing any outputted `\LaTeX` markup with HTML markup.
I started by going through Patashnik’s `plain.bst` and replacing any outputted \LaTeX{} markup with HTML markup.

Added in even more HTML markup so that appearance could be customized with CSS. (Lots of work!)
Overview

- I started by going through Patashnik’s `plain.bst` and replacing any outputted \LaTeX\ markup with HTML markup.
- Added in even more HTML markup so that appearance could be customized with CSS. (Lots of work!)
- Added in support for some custom \BibTeX\ fields (ps, pdf, url) useful for hyperlinks.
Overview

- I started by going through Patashnik’s `plain.bst` and replacing any outputted \LaTeX{} markup with HTML markup.
- Added in even more HTML markup so that appearance could be customized with CSS. (Lots of work!)
- Added in support for some custom \BibTeX{} fields (ps, pdf, url) useful for hyperlinks.
- Added some custom sorting routines typical of author publication lists (sort by type, by year, etc.)
Overview

- I started by going through Patashnik’s `plain.bst` and replacing any outputted \LaTeX\ markup with HTML markup.
- Added in even more HTML markup so that appearance could be customized with CSS. (Lots of work!)
- Added in support for some custom \BibTeX\ fields (`ps`, `pdf`, `url`) useful for hyperlinks.
- Added some custom sorting routines typical of author publication lists (sort by type, by year, etc.)
- Added routines to convert \LaTeX\ characters to Unicode or HTML entities.
Mapping \LaTeX\ to HTML/Unicode
Sample entry from `ent.xml`

```xml
<char pos="127">
  <entity name="para" set="iso-8879-num">
    <desc>=pilcrow (paragraph sign)</desc>
  </entity>
  <entity name="para" set="html4-lat1">
    <desc>pilcrow sign = paragraph sign</desc>
  </entity>
  <unicode value="00B6">
    <desc>PILCROW SIGN</desc>
  </unicode>
  <latex>
    <seq>\P</seq>
    <seq req="textcomp">\textparagraph</seq>
    <seq req="textcomp">\textpilcrow</seq>
  </latex>
  <plain value="B6" set="iso-8859-1" glyph="¶"/>
</char>
```
"\textfractionsolidus" "\frasl;" find.replace
"\textparagraph" "\para;" find.replace
"\texttrademark" "\trade;" find.replace
"\quotedblbase" "\ldquor;" find.replace
"\textpilcrow" "\para;" find.replace
"\textrecipe" "\#x211E;" find.replace
"\checkmark" "\#x2713;" find.replace
"\copyright" "\copy;" find.replace
"\'{A}" "\aa{acute;}" find.replace
"\'{C}" "\#{x0106;}" find.replace
"\'{E}" "\ae{acute;}" find.replace
"---" "\mdash;" find.replace
"\TH" "\thorn;" find.replace
"\aa" "\ae{ring;}" find.replace
"\&" "\amp;" find.replace
"\P" "\para;" find.replace
"~" "\nbsp;" find.replace
trans.bst

Some problems with trans.bst as output:

- It includes some glyphs (e.g., , , f) we'd rather not convert to HTML/Unicode.
- It is missing some other glyphs we would indeed like to convert (e.g., the TEX logo, \slash).
- We end up with hundreds of lines of bst code, but BiBTeX can handle only 100 tokens per function.
- BiBTeX has no built-in string search-and-replace function.
Some problems with `trans.bst` as output:

- It includes some glyphs (e.g., fi, ffi) we’d rather not convert to HTML/Unicode.
Some problems with `trans.bst` as output:

- It includes some glyphs (e.g., fi, ffi) we’d rather not convert to HTML/Unicode.
- It is missing some other glyphs we would indeed like to convert (e.g., the \TeX logo, \texttt{\textbackslash slash}).
Some problems with `trans.bst` as output:

- It includes some glyphs (e.g., fi, ffi) we’d rather not convert to HTML/Unicode.
- It is missing some other glyphs we would indeed like to convert (e.g., the \TeX logo, \textbackslash{slash}).
- We end up with hundreds of lines of `bst` code, but Bib\TeX\ can handle only 100 tokens per function.
trans.bst

Some problems with trans.bst as output:

- It includes some glyphs (e.g., fi, ffi) we’d rather not convert to HTML/Unicode.
- It is missing some other glyphs we would indeed like to convert (e.g., the \TeX logo, \slash).
- We end up with hundreds of lines of bst code, but \TeX can handle only 100 tokens per function.
- \TeX has no built-in string search-and-replace function.
Writing `split_trans`

```bash
i=1
func=1
words=0
lines=$(wc -l <$tmpfile)

echo "FUNCTION\{latex2html.$func}"
echo "{"
while [ $i -le $lines ]
  do
    newwords=$(sed -n "$i"p $tmpfile | wc -w)
    if [ $((words + newwords)) -ge 100 ]
      then
        words=0
        let func=func+1
        echo -e "}\n\nFUNCTION\{latex2html.$func}\n{"
      fi
    sed -n "$i"p $tmpfile
    let i=i+1
    let words=words+newwords
  done
echo -e "}\n"
```
Writing `find.replace`

```plaintext
STRINGS{replace find text}
INTEGERS {find_length}
FUNCTION {find.replace}
{ 'replace :=
   'find :=
   'text :=
   find string.length 'find_length := ""
   { text empty$ not }      
   { text #1 find_length substring$ find =}
   { replace *}
   text #1 find_length + global.max$ substring$ 'text :=
   }
   { text #1 #1 substring$ *}
   text #2 global.max$ substring$ 'text :=
   }
if$
   }
   while$
   }
```
CSS formatting examples