Bibliographies in LaTeX

You can keep your bibliographic references in a file called `BIBLIO.bib`; this file is to be processed by the program `bibtex`.

The text references in your paper are made with the `\cite` command:

```
\cite{KEY}
```

Bibliographies in LaTeX

You cause the actual generation of the bibliography with:

```
\bibliographystyle{STYLE}
\bibliography{BIBLIO}
```

Creating your bibliography database

Each entry in the database contains predefined information, some general and some specific to various types of publications.

These fields include author, title, journal, volume, number, pages, date, institution, publisher, url.

Creating your bibliography database

The general form of each of the entries in a *.bib file is:

```
@entry_type{key,
    field_name = ‘‘text’’,
    field_name = ‘‘text’’,
     ...
    field_name = ‘‘text’’
}
```

Examples

```
@book{Crandal:2001:PNCP,
  author = "Richard Crandall and Carl Pomerance",
  title = "Prime Numbers: A Computational Perspective",
  year = "2001",
  address = "New York",
  publisher = "Springer-Verlag",
  ISBN = "0-387-94777-9"
}
```

Examples

```
@article{Cipra:1996:SLLN,
  author = "Barry Cipra",
  title = "The Secret Life of Large Numbers",
  year = "1996",
  journal = "What's Happening in the Mathematical Sciences",
  volume = "3",
  address = "Providence Rhode Island",
  publisher = "American Mathematical Society",
  pages = "90-99",
  ISBN = "0-8128-0355-7"
}
```

Bibliography styles

There are four `\bibliographystyles` recognized:

- `plain` – entries are ordered alphabetically and markers are a number inside square brackets

Bibliography styles

- `unsrt` – entries are ordered by appearance of citation inside the paper
- `alpha` – same as plain but markers are an abbreviation of the author's name and year

Bibliography styles

- `abbrv` – same as plain but bibliographic listing abbreviates first names, months, and journal names

The order of events

In order to have your bibliography compiled into your paper, you run the following sequence of programs:

```
pdflatex BASENAME
bibtex BASENAME
pdflatex BASENAME
```

The order of events

While you can specify suffixes with pdflatex/latex, bibtex is not some accommodating and it is easier to just specify the basename. This is also true inside of your document: at the `\bibliography` command, don't put the `.bib`.

Viewing output

You have a number of choices for viewing various output:

- `dvi` files – you can use `xdvi` or `evince`.
- `ps` files – you can use `gv`, `ghostview`, or `evince`.
- `pdf` files – you can use `xpdf` or `evince`.

Conversions

As mentioned earlier, there are a number of conversions that you might want to do with your LaTeX output:

- `dvips / dvi2ps` – converts a DVI file to PostScript® (PS).
- `ps2pdf` – converts a PostScript file to Portable Document Format (PDF).

Conversions

- `dvipdf` – converts a DVI file to PDF.
- `pdftops` – converts a PDF file to PS.

Conversions

- `pdftotxt` – converts a PDF file to text.

Diagrams with dot files

The `graphviz` package allows you to use an ordinary text file to automatically create graph visualizations.

As you can see from the examples displayed, it can make some very neat visualizations. You can find more information at `http://www.graphviz.org`.

The dot language

Here's the dot code for the graph in my sendmail paper:

```
// Uses graphviz package from http://www.graphviz.org

digraph MailSplit
{
        "Outside Mailer"  [shape = parallelogram];
        "Incoming Mailer"  [shape = parallelogram];
        "Outgoing Mailer"  [shape = parallelogram];
        "Outside Mailer" -> "Incoming Mailer"
           [label =
             "An Email Message With\n Multiple Recipients In\n Envelope"]
        "Incoming Mailer" -> "Queue Entry for\n Recipient #1"
```

```
   [label = "Recipient #1"];
"Incoming Mailer" -> "Queue Entry for\n Recipient #2"
   [label = "Recipient #2"];
"Incoming Mailer" -> "..." [style = "dotted"];
"Incoming Mailer" -> "Queue Entry for\n Recipient #n"
   [label = "Recipient #n"];
subgraph cluster_0 {
         style = filled;
        color = lightgrey;
        label = "Incoming Queue";
        "Queue Entry for\n Recipient #1"
           [style=filled,color=white];
        "Queue Entry for\n Recipient #2"
           [style=filled,color=white];
        "..." [style=filled,color=white];
        "Queue Entry for\n Recipient #n"
           [style=filled,color=white];
}
"Queue Entry for\n Recipient #1" -> "Outgoing Mailer";
"Queue Entry for\n Recipient #2" -> "Outgoing Mailer";
"..." -> "Outgoing Mailer" [style=dotted];
```

```
        "Queue Entry for\n Recipient #n" -> "Outgoing Mailer";
}
```