

Advanced L^AT_EX Topics

Edsko de Vries (edsko@edsko.net)

Hosted by Netsoc



This is a follow-on from part 1, *Introduction to L^AT_EX*. We covered

- The basic tools (`latex`, `xdvi`, `dvips`, `ps2pdf`, `pdflatex`)
- Basic document structure and table of contents
- Basic font changing commands (`\small`, `\textit`, etc.)
- Itemized and numbered lists
- Tables
- Footnotes and marginal notes
- Cross referencing
- Simple mathematics
- Importing graphics
- Typesetting code listings

What will we cover today?

- More mathematics
- Using a bibliography
- Advanced listings typesetting
- Creating an index
- “Floats”
- Custom commands/environments

Recall

```
\begin{equation}
\sum_{n=0}^i \sqrt{n^i}
\label{eq:summation}
\end{equation}
```

$$\sum_{n=0}^i \sqrt{n^i} \tag{1}$$

So, we have $\sum_{\text{lower}}^{\text{upper}}$, where `lower` and `upper` are the lower and upper bounds to the sum. Also, x^y for powers, and \sqrt{x} for square roots. See `symbols.pdf`

Equations over multiple lines

```
\begin{equation}
\begin{split}
x &= a + b \\
&\quad + c - d \\
&= e + f
\end{split}
\end{equation}
```

$$\begin{aligned} x &= a + b \\ &\quad + c - d \\ &= e + f \end{aligned} \tag{2}$$

Choice

```
\begin{equation}
\text{fac } x =
\begin{cases}
1 & \text{if } x = 0 \\
x \cdot \text{fac } (x-1) & \text{if } x > 0
\end{cases}
\end{equation}
```

$$\text{fac } x = \begin{cases} 1 & \text{if } x = 0 \\ x \cdot \text{fac } (x - 1) & \text{if } x > 0 \end{cases} \quad (3)$$

Matrices

```
\begin{equation}
\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \times
\begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}
\end{equation}
```

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \times \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} \quad (4)$$

Symbols over or under other symbols

`\left(A \xrightarrow{x \in \mathcal{A}} B \right)`

`\overrightarrow{x} \circ \underleftarrow{y}`

`\overset{*}{\underset{\wedge}{\alpha}}`

$$\left(A \xrightarrow{x \in \mathcal{A}} B \right) \tag{5}$$

$$\overrightarrow{x} \circ \underleftarrow{y} \tag{6}$$

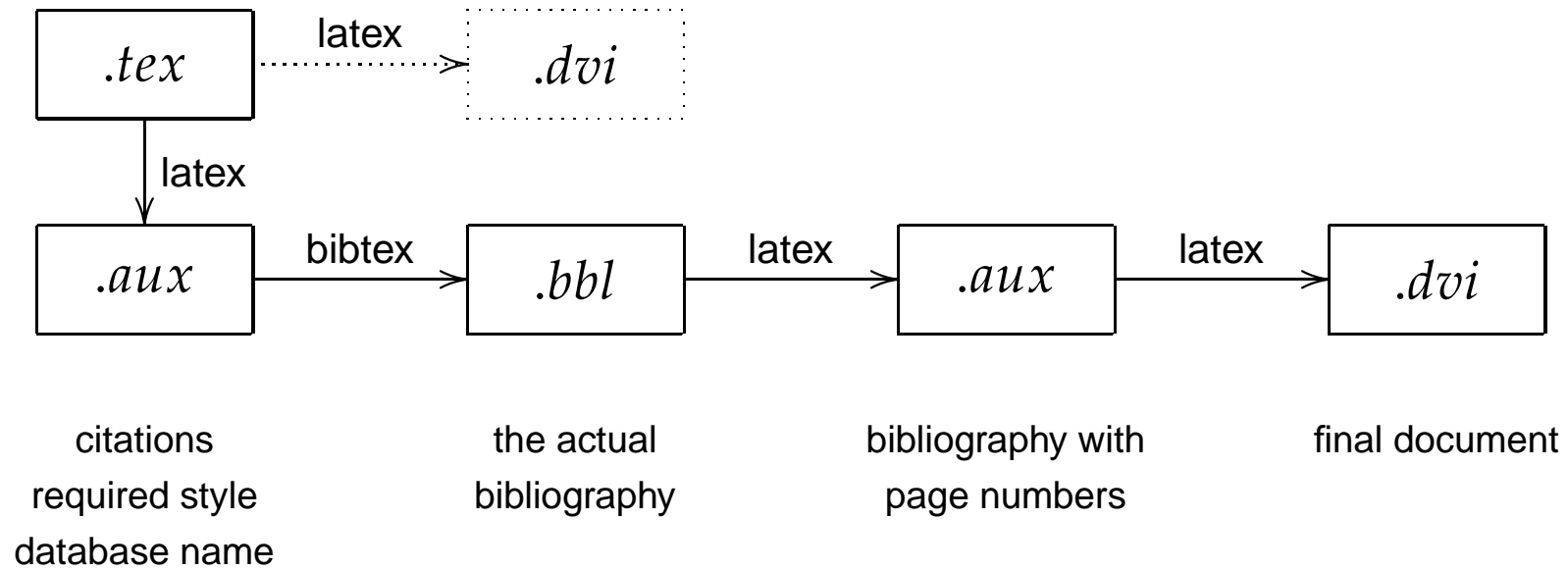
$$\overset{*}{\underset{\wedge}{\alpha}} \tag{7}$$

Commutative diagrams (needs amscd package)

```
\begin{equation}
\begin{CD}
\wp V @>\text{Add}_0>> \wp V \\
@V{\Re}VV @AA{\Re}A \\
V^* @<<\text{Add}_1<< V^*
\end{CD}
\end{equation}
```

$$\begin{array}{ccc} \wp V & \xrightarrow{\text{Add}_0} & \wp V \\ \Re \downarrow & & \uparrow \Re \\ V^* & \xleftarrow{\text{Add}_1} & V^* \end{array} \quad (8)$$

How does it work?



Using a Bibliography (ctd.)

example.tex:

```
\documentclass{article}
\begin{document}
```

As explained in `\cite{some_label}`, ...

```
\bibliographystyle{acm}
\bibliography{references}

\end{document}
```

Using a Bibliography (ctd.)

references.bib:

```
@article{some_label,  
  author = {Edsko de Vries},  
  title = {This is an example},  
  journal = {The title of a famous journal},  
  year = 2004  
}
```

Using a Bibliography (ctd.)

To generate the DVI file:

```
[~/example]> latex example.tex  
[~/example]> bibtex example  
[~/example]> latex example.tex  
[~/example]> latex example.tex
```

Yields

As explained in [1], ...

References

[1] DE VRIES, E. This is an example. *The title of a famous journal* (2004).

Some other remarks

- `\cite[pp. 4--6]{label}` will give [6, pp. 4–6]
- `\cite{label1, label2, label3}` will give [3, 6, 10]
- **Some other styles:** `amsalpha`, `amsplain`, `abbrv`, `alpha`, `apalike`, `ieeetr`, `siam`, `unsrt`, `acm`
- **Style affects layout of bibliography as well as layout of in-text citations:** [1] versus [Aam86a] versus [Aamport, 1986a]
- **See `btxdoc.dvi` or `xampl.bib` for more information on the database entries itself (articles, book, dissertations, etc.)**

Defining a custom language

```
\lstdefinlanguage{68kasm}{%  
  keywords={move.b,add.b,cmp.b,bne},%  
  alsoletter={.},%  
  comment=[l]{*},morecomment=[n]{/*}{*/}%  
}
```

Then

```
\begin{lstlisting}[language=68kasm]  
/* Example assembly listings */  
  
    move.b  #0,d0          * counter  
CLEAR    * do {  
    move.b  #0,(a0,d0)     * clear buckets  
    add.b   #1,d0          * }  
    cmp.b   #14,d0        * while( counter != 14 )  
    bne     CLEAR  
  
\end{lstlisting}
```

Advanced Listings Typesetting (ctd.)

Yields

/ Example assembly listings */*

```

        move.b  #0,d0           * counter
CLEAR   * do {
        move.b  #0,(a0,d0)      * clear buckets
        add.b   #1,d0           * }
        cmp.b   #14,d0          * while( counter != 14 )
        bne     CLEAR
```

Literate programming

```
\lstdefinlanguage{cfg}{%  
  literate={%  
    {<}{{{\$\\langle$}}}{1}%  
    {>}{{{\$\\rangle$}}}{1}%  
    {::=}{{{\$\\rightarrow$}}}{1}%  
    {-|}{{{\$\\dashv$}}}{1}%  
  }%  
}
```

Every entry in the `literate` block defines a replacement, where

$$\{x\} \quad \left\{ \{a\}\{b\}\{c\} \right\} \quad \{n\}$$

Defines that x should be replaced by abc , and n should be the length of abc . Note that every character must be delimited in curly braces.

Advanced Listings Typesetting (ctd.)

Then

```
\begin{lstlisting}[language=cfg]
<S> ::= <E> - |
<E> ::= <E> + <T>
<E> ::= <T>
\end{lstlisting}
```

Yields

$$\langle S \rangle \rightarrow \langle E \rangle - |$$
$$\langle E \rangle \rightarrow \langle E \rangle + \langle T \rangle$$
$$\langle E \rangle \rightarrow \langle T \rangle$$

Example

```
\documentclass{article}
\usepackage{makeidx}

\makeindex

\begin{document}

\index{entry}
This is some section about \textit{entry}.

\index{another}
This is another section about \textit{another}.

\printindex
\end{document}
```

Generating an Index (ctd.)

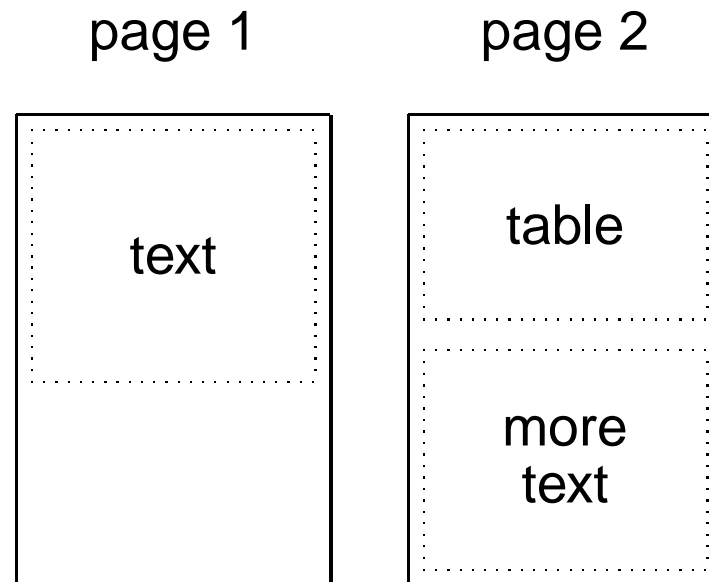
Then to compile

```
[~/example]> latex example.tex  
[~/example]> makeindex example  
[~/example]> latex example.tex
```

Remarks

- Refer to other entries: `\index{another|see{entry}}`
- Subentries: `\index{grammar!context free}`
- Special characters: `\index{p@{ ϕ }}` (necessary to make sure ϕ is in the right place alphabetically)

Problem: you have a large table (or image) in your document, and it does not fit on the current page, so \LaTeX moves it to the next page, leaving a large empty space:

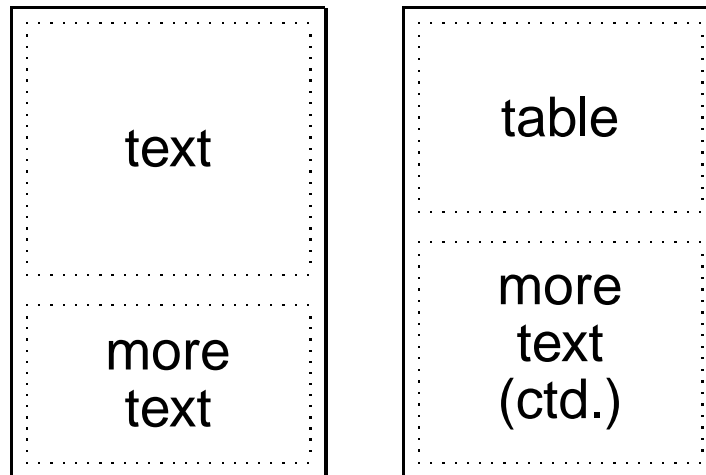


Ugly.

Better to put the table in a *float*. A float is an object in a \LaTeX document that can “float” around the document. \LaTeX decides what the best position for the float is:

page 1

page 2



The table is still in the same position (this is not necessarily the case), but it has “floated” over part of “more text”. Much better.

How?

```
\begin{table}  
... your table definition ...  
\caption{\label{tab:label}Table caption}  
\end{table}
```

`\begin{table}` introduces a float that holds a table (it does not define the table itself; use `\begin{tabular}` ...). The float is also given a caption and a label.

\LaTeX will position the table somewhere, and give it a caption like “Table 4.1 – Derivation of the LR(0) Automaton”.

It is important to give the float a label, because you don't know where the float will be in your document. Thus

As shown in the following table, ...

Will usually not make any sense, because the table might not immediately follow this paragraph (but be on the next page, for example). Better:

As shown in table `\ref{tab:label}`, ...

Note: `\begin{figure}` introduces a float that is intended to hold an image of some form, and will get a caption like “Figure 6.2 – Finite State Machine”.

Custom Commands

Suppose you need a standard definition over and over again. You can define a new command that does the work for you (a macro):

```
\newcommand{\arsdef}{%  
  Let  $\mathcal{A}$  be an ARS,  $\alpha \in \mathcal{A}$ }
```

Then you can simply use `\arsdef` to get “Let \mathcal{A} be an ARS, $\alpha \in \mathcal{A}$ ”.

A macro can also have parameters:

```
\newcommand{\vect}[1]{%  
   $\overrightarrow{\#1}$ }
```

Then use `\vect{a}` to get “ \overrightarrow{a} ”.

Custom Commands (ctd.)

There is a problem with this definition though:

```
\newcommand{\vect}[1]{\$\overrightarrow{#1}$}
```

Consider:

To multiply two vectors $\$\vect{a} \cdot \vect{b}\$,$

This will give a “Missing \$ inserted. error.” Why?

The macro enters math mode, but we are already *in* math mode!

Solution:

```
\newcommand{\vect}[1]{  
  \ensuremath{\overrightarrow{#1}}}
```

Suppose you often use something like

```
\begin{table}
\begin{center}
\begin{tabular}{...}

...

\end{tabular}
\end{center}
\caption{\label{tab:...}...}
\end{table}
```

Custom Environments (ctd.)

In general to make a new environment:

```
\newenvironment {EnvName} [NumArgs] {begin} {end}
```

Where `EnvName` is the name of the new environment, `NumArgs` is the number of arguments (may be omitted if no arguments), `begin` is the start of the environment, and `end` is the end of the environments.

Note that the arguments can only be used in `begin`, not in `end`. If you do want to use them in `end`, you must explicitly save them (see example).

Custom Environments (ctd.)

For our example:

```
\newenvironment{FloatTable}[3]{%
  \begin{table}%
  \newcommand{\FTlabel}{#2}    % Save arguments #2 and #3
  \newcommand{\FTcaption}{#3} % for later use
  \begin{center}%
  \begin{tabular}{#1}%
  }{%
  \end{tabular}%
  \end{center}%
  \caption{\label{tab:\FTlabel}\FTcaption}%
  \end{table}%
}
```

Custom Environments (ctd.)

Then

```
\begin{FloatTable}{l1}{foo}{bar}  
1 & 2 \\  
3 & 4 \\  
\end{FloatTable}
```

Will yield a table with label `tab:foo`

1	2
3	4

Table 1: bar

Finally, two special lists

```
\newcounter{cenum}
\newenvironment{cenum}%
  {\begin{list}{\arabic{cenum}.}{%
    \usecounter{cenum}%
    \setlength{\itemsep}{0pt}%
    \setlength{\parsep}{0pt}}}
  {\end{list}}
```

```
\newenvironment{citem}%
  {\begin{list}{$\bullet$}{%
    \setlength{\itemsep}{0pt}%
    \setlength{\parsep}{0pt}}}
  {\end{list}}
```

Custom List Environments (ctd.)

Then `citem` can be used the same way as `itemize`, but yields

- First item
- Second item
- Third item

Instead of the standard `itemize`

- First item
- Second item
- Third item

And similarly for `cenum` (instead of `enumerate`). This more compact list is useful when the items of the list are short.

A few closing remarks:

- The following is a useful standard document preamble

```
\documentclass{report}
\usepackage{amsmath,amstext,amssymb,amscd}
\usepackage{mathptm,times}
```

This includes most of $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{T}\mathcal{E}\mathcal{X}$, and uses a more modern and higher quality font

- If you run Latex, and you get an error, and latex does not want to return to the command line, type `x` (followed by pressing Return)
- Check my website (www.edsko.net) for more information (also for answers to questions asked at the tutorials)