

Lecture 5

L^AT_EX best practices

Scribes notes by: Thomas Hofer

October 2, 2008

1 Introduction

In the field of Computer Sciences, it is common to use L^AT_EX to hand in papers, turn in reports or write thesis. We will take the reports needed for this course as an opportunity to practice writing in L^AT_EX and to get “good” habits. In these lecture notes, we present some best practice recommendations as they were explained by Thomas Henzinger on October 2nd, 2008.

2 Keep your source readable

You should always try and produce a clean and readable source code for your L^AT_EX files. This makes errors easier to detect and reduces the number of bugs. Readable source also makes the work of co-writers simpler, for they do not need to compile every intermediate version of the paper to read it. In particular, the following suggestions help to keep your source clean.

- Introduce automatic line breaks at a fixed width (typically 80 characters). Line breaks ensure that all readers of the source will have it displayed similarly, regardless of the editor they use.
- Use comments sparingly. More specifically, do not comment out obsolete parts of your source. If you’re worried about losing information you only have in an intermediate version of your paper, use a versioning system.
- Start every sentence at a new line. It is much easier to move whole lines around than sentences starting in the middle of a line.
- Use indentation.

3 Macros

By using macros, you can make your source simpler to write, but also more readable and less error-prone. The following rules will help achieving this triple aim.

Lecture by Thomas Henzinger
Scribes by Thomas Hofer

- If you introduced a macro for a single expression/symbol, then *always* use it when the symbol occurs. Never use two different macros for the same symbol.
- Keep the name of the macro close to what the defined expression will look like (e.g. `\Smax` for S_{max}).
- Introduce a macro whenever it makes the source more readable (e.g. because of font commands).¹
- Don't use macros for single symbols. If you or a co-author edit the paper afterwards you're likely to use `\sigma` if you see σ 's everywhere, even if you've introduced a macro for it, say to give it a meaning (e.g. `\minSink`). On later editions of the text you might want to change the symbol used to represent a minSink and you're likely to obtain inconsistencies in your paper.
- Keep the macros definitions clear and readable. Don't import useless packages, get rid of obsolete definitions, *etc.*
- Don't bother about layout details before the very last end and keep a copy of the pure logical source of the paper, *i.e.* without formatting commands.

¹Why use font commands? -Some expressions typeset better this way (e.g. the difference between $S_{function}$ and $S_{function}$).