

---

## Preparing for scientific conferences with $\LaTeX$ : A short practical how-to

Paweł Łupkowski and Mariusz Urbański

### Abstract

In this paper we will present a short practical how-to guide considering the complete experience of preparing materials for a scientific conference. This will cover the preparation of: a paper with figures and charts, a PDF presentation and a conference poster. The paper is based on our hands-on experience in this area.

### Introduction

The main aim of this paper is to present a short practical how-to guide considering the complete experience of preparing materials for a scientific conference. Attending conferences is one of the essential aspects of being a researcher. There are a huge number of conferences which involve preparing different materials, such as papers, presentations and posters. What is more, organisers of many of these conferences demand preparing at least some of these materials (usually papers) in  $\LaTeX$ . This paper covers our experience in attending conferences in different domains (cognitive science, psychology, logic, computer science, linguistics). Our experience is that  $\LaTeX$  constitutes a single extremely efficient environment to prepare all the necessary materials. One major virtue of  $\LaTeX$  is that it allows for easy use of the content in different forms (whether a paper, a presentation, or a conference poster).<sup>1</sup>

We are not aiming at novelty in this paper. Our aim is to write down and share our tips and tricks used in everyday work with  $\LaTeX$  in science. We hope that it will be a useful guide, especially for  $\LaTeX$  beginners and also for scientists wondering which combination of tools will be useful in their work and is worth the time and effort to learn. We assume that our reader has a basic knowledge and skills in typesetting in  $\LaTeX$ .<sup>2</sup> Preparing for a conference is often done under time pressure. We hope that this paper will serve as a quick reference for those of you who don't have time to read manuals and search for specific solutions.

The paper is structured as follows. The first section covers preparing a conference paper, the second section a presentation, and the third a poster.

---

Originally presented at EuroBach $\TeX$  2013.

<sup>1</sup> It is worth mentioning that there is a method which allows to produce three types of outputs (a paper, a poster and a presentation) from one input file — see [1].

<sup>2</sup> For a paper with convincing reasons to start using  $\LaTeX$  in the first place, see [6].

In the summary we point at some issues that we find difficult or disturbing in our work with  $\LaTeX$  in preparing materials for conferences.

## 1 Conference paper

We attend conferences in different domains. Usually, the organisers ask for submission of a talk proposal in a form of a short or a long paper. In such a case you should check for a paper template provided by the organisers. Sometimes it is a specially prepared class for the paper (as for the *Logic and Cognition* conference<sup>3</sup>), or a general class used for certain domains, like the EACL class for linguistic conferences (cf. *Semdial*<sup>4</sup>). Also, specific requirements for the most popular *article* class might be given (cf. *IWCS 2013*<sup>5</sup>). You should always search for a *Call for Papers* or *Information for Authors* section. Many different solutions used in different  $\LaTeX$  classes (especially for author and title fields) might be difficult to grasp, and sometimes reading example articles and guidelines provided by the organisers is necessary. When you prepare your paper in advance, our advice is to simply use the *article* class without any custom commands and modifications. Then adjusting your source to the conference requirements should be fairly easy.

One of the problems that we often encounter preparing our papers is how to produce high quality pictures in an easy way. Below we present a class for preparing bar charts within  $\LaTeX$ . We also describe how you can use Inkscape to generate pictures, which might be then exported into a  $\LaTeX$  code and used directly in your paper. It is worth mentioning that using the presented methods involves compiling your file first with  $\LaTeX$  and then with a DVI-to-PostScript converter such as *dvips*.

**Bar charts** One of the simplest ways to put a chart into a paper (not using software other than  $\LaTeX$  itself) is the *bchart* package. If you don't already have it installed, it can be obtained from CTAN: <http://www.ctan.org/pkg/bchart>.

The basic structure of a chart is the following:

```
\begin{bchart}[max=5, step=1]
\bcbar{4.5}
\end{bchart}
```

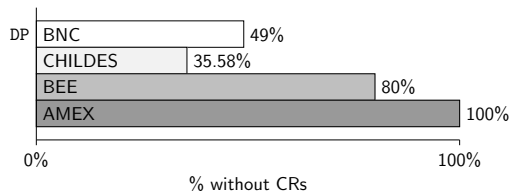
The option `max` defines the maximum value on the  $x$ -axis (the default is 100). If you want to define regular steps along the  $x$ -axis, you can use the `step` option (or, an option `steps={...}` allows you to give irregular step values, e.g. `steps={1, 3, 5}`).

---

<sup>3</sup> <http://logicandcognition.org/>

<sup>4</sup> <https://sites.google.com/site/semdial2012seinedial>

<sup>5</sup> <http://www.ling.uni-potsdam.de/iwcs2013/>



**Figure 1:** An example bar chart using *bchart*

The command `bcbar` puts a bar with a given value. For this command you can use several options, like `text` (to print some text inside the bar), `color` (to define the bar's colour), `label` (to assign a label to the bar, which will be visible on the left side of the bar). To add a caption to the *x*-axis use the command `\bcxlabel{...}`.

A very convenient feature is that the charts are scalable. There are two ways to obtain this effect. First, you can scale only the chart (without a text) by adding the option `scale` into the `bchart` command. The second one is to put the chart into the command `\scalebox{<factor>}{...}`.

An example chart presented in Figure 1 is generated by the following code:

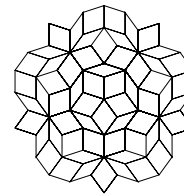
```
\scalebox{0.7}{
\begin{bchart}[max=100, unit=%]
  \bcbar[label={\tt DP}, text=BNC,
    color=white]{49}
  \bcbar[text=CHILDES, color=gray!10]{35.58}
  \bcbar[text=BEE, color=gray!50]{80}
  \bcbar[text=AMEX, color=gray!80]{100}
\bcxlabel{\% without CRs}
\end{bchart}
}
```

Other useful commands available via this package can be found in [8].

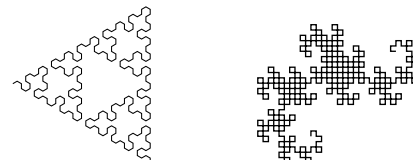
**L<sup>A</sup>T<sub>E</sub>X and Inkscape** To include a more sophisticated picture into your paper, you may want to use Inkscape (an open source vector graphic editor).<sup>6</sup> Using Inkscape you can prepare your picture, then export it to *\*.png* or *\*.pdf* format, and afterwards include it into your paper in the usual way.<sup>7</sup> However, one of the most convenient ways is to use the L<sup>A</sup>T<sub>E</sub>X export function in Inkscape. To do this choose *File > Save as ...* and then pick the *L<sup>A</sup>T<sub>E</sub>X with PSTricks* option. This will save your drawing into a *\*.tex* file. Then you can put it in your paper with the `\input{foo.tex}` command, or—to obtain a self-contained paper—simply copy and paste the code into the paper's source. Remember also to load the `pstricks` package in your preamble. An example picture prepared in Inkscape and then embed-

<sup>6</sup> <http://inkscape.org>

<sup>7</sup> For a short, but comprehensive, guide see [2].



**Figure 2:** A drawing prepared in Inkscape and exported to L<sup>A</sup>T<sub>E</sub>X (figure from the Inkscape examples set, file *l-systems.svgz*)



(a) Sierpinski triangle (b) Dragon curve

**Figure 3:** Side by side figures (figures from the Inkscape examples set, file *l-systems.svgz*)

ded in this paper is presented in Figure 2. Such a picture might be easily scaled using the same command as in the *bchart* case, namely `\scalebox{...}`.

Inkscape (from version 0.48) can also export the graphics to PDF/EPS/PS, and the text to a L<sup>A</sup>T<sub>E</sub>X file, so you can obtain a vector image with the same font and size as in normal text. This might be especially useful when you want to prepare a picture with mathematical formulas included.<sup>8</sup> For the Inkscape manual, see [5].

**Side by side figures** Sometimes there is a need to arrange pictures inside a figure environment (e.g. to save some space for the main text). This can easily be done by using the *subfigure* package.<sup>9</sup> An example of such a solution is presented in Figure 3. Below you may trace the code used to generate the example. Each picture is placed with a `subfigure` command. A very useful feature is the optional argument to supplement each figure with a description. As a consequence, you may easily produce a complex figure with a caption and description for each of its elements.

```
\mbox{\subfigure[Sierpinski triangle]{%
  \input{sierpinski.tex}
  \quad
  \subfigure[Dragon curve]{\input{dragon.tex}}
}}
```

<sup>8</sup> L<sup>A</sup>T<sub>E</sub>X and Inkscape fans might be also interested in an extension for Inkscape that allows using L<sup>A</sup>T<sub>E</sub>X inside the drawing program itself. See <http://www.johndcook.com/blog/2009/12/22/including-latex-in-inkscape/>.

<sup>9</sup> From <http://www.johndcook.com/blog/2009/01/14/how-to-display-side-by-side-figures-in-latex/>.

## 2 Conference presentation

Probably the most popular way of preparing presentations with L<sup>A</sup>T<sub>E</sub>X is by using the *Beamer* class.<sup>10</sup> It allows for fast and easy preparation of presentations (especially on the basis of a previously written paper). Beamer produces a PDF file on output, which guarantees that the presentation will look exactly the same on different machines. In addition, it allows using sections and subsections to structure the presentation in a paper-like style—very convenient when we build a presentation on the basis of a paper. Last but not least, it allows preparing attractive presentations with well-balanced colour schemes and slide elements that make it easier for the audience to track the presentation (number of slides, short title, author etc.).

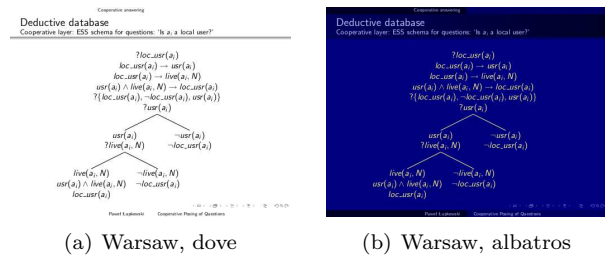
If you would like to take a look at the complete set of default Beamer themes available, see [9, Section 30]. You will find there names and pictures of Beamer themes (title slide and a regular one). If you are interested in the range of possible modifications of the standard themes see [9, Section 29].

**Slide customisations** The first modification you may want to make is to change the basic colour of your presentation. You may have a situation when you want to match the colour scheme to the colours used in your poster, or your corporate identity colour scheme etc. The simplest way is to use RGB colour description (the numerical values for a given colour might be obtained easily using Inkscape or GIMP). To use the colour you've picked, include a `\usecolortheme` command into the preamble of your presentation, as shown:

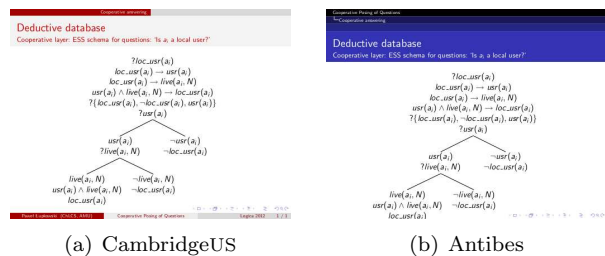
```
\documentclass{beamer}
\usecolortheme[RGB={241,200,144}]{structure}
\settheme{Warsaw}
```

One of the useful features of Beamer is the possibility of changing the theme (and its colour scheme) of the whole presentation after it is prepared. This is might be useful especially when technical conditions for your presentation will not match the theme you used for the presentation (e.g. the room where you are giving your talk is too bright or the beamer is of poor quality). Changing one line of L<sup>A</sup>T<sub>E</sub>X code (an argument of the `usecolortheme` command) might make your slides visible again—see Figure 4, where one slide is presented in two different colour themes (*dove* and *albatros*). However, you should remember

<sup>10</sup> Other options are e.g. the *powerdot* class (<http://www.ctan.org/pkg/powerdot>), the *prosper* class (<http://www.ctan.org/pkg/prosper>), and an interesting KOMA-script based presentation—see [7] and <http://www.latextemplates.com/template/koma-script-presentation>.



**Figure 4:** Beamer allows for easy modification of the colour scheme of your presentation, so you can adjust it to your needs (even after the presentation is ready)

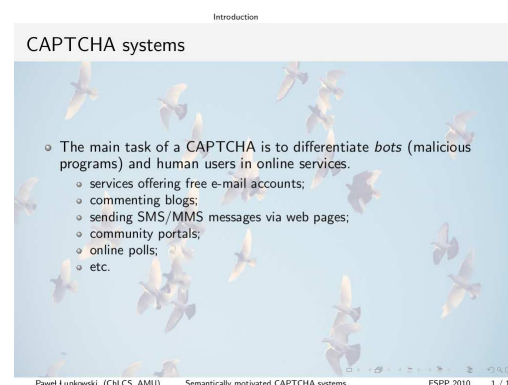


**Figure 5:** When changing the slide theme remember that themes have different slide architecture, and sometimes the content does not fit the new theme

that some themes have more extensive slide structure than others (see e.g. Figure 5; the content which fits the slide in the CambridgeUS theme does not fit the slide in the Antibes theme). Thus it is useful to check in advance which theme changes are safe for your content.

Beamer offers also the possibility of setting a background image for the slides (see Figure 6). To do this, use the following command in the preamble of your document [9, Section 27]:

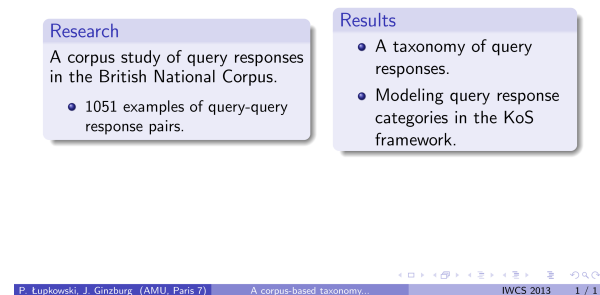
```
\setbeamertemplate{background canvas}
{\includegraphics[width=\paperwidth,
height=\paperheight]{backgrounding.jpg}}
```



**Figure 6:** Custom background for a Beamer slide

## A corpus-based taxonomy of question responses

Paweł Lupkowski, Jonathan Ginzburg



**Figure 7:** Side by side content of a Beamer slide with the `columns` environment

**Side by side content on a slide** Sometimes you may need to arrange the content of a slide in two (or three) columns. One way to do this is to use the `columns` environment. Below you will find sample code, which generates the slide presented in Figure 7. You may put pictures, blocks, lists and regular text inside columns. The option `[c]` of the `columns` command centres columns vertically. Each `column` environment constitutes one column, where you give its width as an argument of the `\begin{column}` command.

```
\begin{columns}[c]
\column{5.5cm}
\begin{block}{Research}
  A corpus study of query responses in the
  British National Corpus.
  \begin{small}
  \begin{itemize}
    \item 1051 examples of query-query
    response pairs.
  \end{itemize}
  \end{small}
\end{block}
\column{5cm}
\begin{block}{Results}
  \begin{itemize}
    \item A taxonomy of query responses.
    \item Modeling query response categories
    in the KoS framework.
  \end{itemize}
\end{block}
\end{columns}
```

**References with BIB<sub>T</sub>E<sub>X</sub>** If you want to include references in your presentation, and you want to use BIB<sub>T</sub>E<sub>X</sub>, we recommend the following method:

```
\begin{frame}[allowframebreaks]{References}
\def\newblock{
```

```
\bibliographystyle{plain}
\bibliography{mybibliography}
\end{frame}
```

The option `allowframebreaks` allows Beamer to produce new slides for references if they will not fit in the initial one. We recommend using the *plain* bibliography style. This is due to the fact that Beamer does not fully support BIB<sub>T</sub>E<sub>X</sub> (it will generate compilation errors with the *natbib* package, for example). If you would like to have author-year citations in Beamer (with BIB<sub>T</sub>E<sub>X</sub>) you may try ignoring compilation errors and see if the output looks fine.<sup>11</sup>

**Hyperlinks between slides** One of our favoured features of Beamer presentations is the possibility of defining non-linear changes of slide via the use of hyperlinks. Using this feature you may skip slides in case of lack of time or unveil hidden slides if it appears that you have some time left. You can also have extra slides prepared for discussion after presentation and easily access them using hyperlinks (without the necessity of skipping lots of slides in front of the audience).

First you should define the target for the hyperlink. You do this adding a label for a slide in the following way:

```
\begin{frame}[label=yourlabel]
```

Afterwards you define the hyperlink to the labelled slide:

```
\hyperlink{yourlabel}{hyperlink text}
```

This method produces a hyperlink which is typeset as regular text. If you want fancier hyperlinks you may use the following modification to the command:

```
\hyperlink{yourlabel}{\beamergotobutton
  {hyperlink text}}
```

The `\beamergotobutton` command produces an attractive graphic button which is a hyperlink to the labelled slide.<sup>12</sup>

**Navigation symbols** Our last tip is about changing a small detail in Beamer presentations,<sup>13</sup> namely the navigation symbols visible in the bottom right corner of each slide. As it happens, we never use them, so it would be nice to get rid of them. This is very simple; it is enough to add the following line to the preamble:

```
\setbeamertemplate{navigation symbols}{}

```

<sup>11</sup> Another solution — perhaps preferable if you have prepared your paper earlier — is to use the `*.bbl` file obtained from compilation of your paper. See <http://tex.stackexchange.com/questions/3542/bibtex-and-beamer>.

<sup>12</sup> You can find tips and tricks for adjusting the button's appearance at <http://tex.stackexchange.com/questions/63171/beamergotobuttons-color>.

<sup>13</sup> This comes from <http://nickhigham.wordpress.com>.

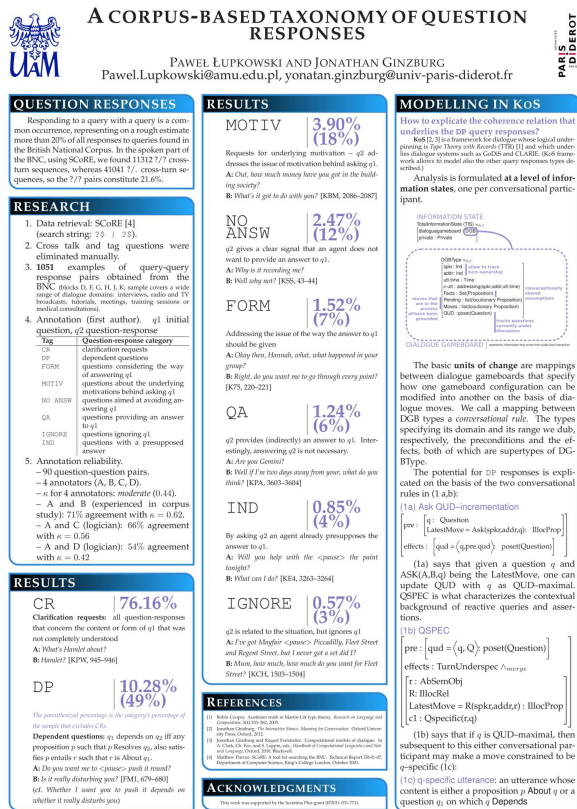


Figure 8: Example poster with the *baposter* class

### 3 Conference poster

Designing a conference poster is a very demanding task. You have to combine carefully picked and structured content with an attractive visual form (similar to the presentation, but here we have to be even more aware of space limitations).  $\LaTeX$  allows us to prepare a good-looking poster with minimal effort spent on the visual form and allows you to focus on the most important thing — what content it should have.

One basic document class for preparing posters with  $\LaTeX$  is *aoposter*.<sup>14</sup> However, our favoured class for typesetting posters is *baposter*; it is also quite simple, and allows for various modifications of a poster structure and appearance. You will find the manual and examples at the project homepage: <http://www.brian-amberg.de/uni/poster/>. The simplest way to learn how to use this class is to download examples and analyse them. Here we will list some of interesting and useful commands available in this class. An example of a poster prepared with this class is presented in Figure 8.

<sup>14</sup> <http://www.ctan.org/pkg/aoposter>

**The class options** In the class options you may declare — *int. al.* — the following:

- **page layout:** *landscape*, *portrait*;
- **paper size:** *a0paper*, *a1paper*, *a2paper*, *a3paper*, *a4paper*;
- **showframe:** draw a frame around the page (mainly useful for debugging).

**Poster options** The main environment for this class is *poster*. The environment has some options available, such as:

- **grid={yes,no}** — turns on/off the visibility of a grid (useful for designing the layout);
- **columns= $\langle n \rangle$**  — number of columns (default 4 in landscape and 3 in portrait format; maximum number is 6);
- **colspacing= $\langle length \rangle$**  — defines the distance between the columns of the poster;
- **background= $\langle bgtype \rangle$**  — defines the background for the poster. The simplest options are *none* for no background and *plain* for a single background colour, which is defined by the option **bgColorOne= $\langle pgf-colour-name \rangle$** .<sup>15</sup>

The content is formatted in columns and placed into attractive boxes. The box is declared with a command:

```
\headerbox{ $\langle boxtitle \rangle$ }{name= $\langle boxname \rangle$ ,
column=0,row=0}{
 $\langle content \rangle$ 
}
```

Thus, as you can see, each box has its own label and column indication. You use the labels to place one box under another one, e.g.:

```
\headerbox{RESEARCH}{name=research,below=intro,
column=0,row=0}{
 $\langle content \rangle$ 
}
```

This code will place the box entitled “RESEARCH” in the first poster column under the box labelled as *intro* (see Figure 8).

You can easily modify the appearance of boxes (like headers, borders, shapes and fill types). Try the following poster options:

- **textborder= $\langle type \rangle$**  — type of border for the lower part of the box (possible values: *none*, *bars*, *coils*, *triangles*, *rectangle*, *rounded*, *faded*);
- **headerborder= $\langle type \rangle$**  — which sides of the text box headers should have a border (*none*, *closed*, *open*);

<sup>15</sup> Other types of background are available, such as different gradients; see the class manual for details: [http://www.brian-amberg.de/uni/poster/baposter/baposter\\_guide.pdf](http://www.brian-amberg.de/uni/poster/baposter/baposter_guide.pdf).

- `headershape=metatype` — type of ornament for the text box headers (`rectangle`, `small-rounded`, `roundedright`, `roundedleft`, `rounded`).

The example poster presented in Figure 8 uses the following options:

- `textborder=roundedleft`
- `headerborder=closed`
- `headershape=roundedright`

**Saving space** If you want to save some space in your poster you can make lists more compact. To do this just use the command `\compresslist` after the `\begin{enumerate}` or `\begin{itemize}`.

If you are interested in other classes or general guidelines for preparing posters with L<sup>A</sup>T<sub>E</sub>X, see [3, 4, 10].

### Summary

We hope that we have convinced you to use L<sup>A</sup>T<sub>E</sub>X as a unified environment for preparing your conference materials. We think that the biggest advantage of L<sup>A</sup>T<sub>E</sub>X is that it allows you to focus on the important thing, namely the quality of the content. It is L<sup>A</sup>T<sub>E</sub>X that takes care of the form of your materials (and we have to admit that we like the results). However, we should also admit that there are some issues that we sometimes find disturbing. Many L<sup>A</sup>T<sub>E</sub>X document classes use different approaches to some standard commands (e.g. `author`, `title`) or define completely new commands to use, so there are cases when we need to meticulously study guidelines or examples provided by the conference organisers. This — unfortunately — might take some time.

As we have pointed out, we value L<sup>A</sup>T<sub>E</sub>X for the possibility of easily reformatting the content from paper into a presentation or a poster. Last but not least, sometimes we find it very difficult to fit our content in the page limit using the document class provided by the organisers. However, one may argue that this is to some extent a universal problem.

**Acknowledgements** This work was supported by funds of the National Science Council, Poland (DEC-2012/04/A/HS1/00715).

### References

- [1] David Allen. Screen presentations, manuscripts, and posters from the same L<sup>A</sup>T<sub>E</sub>X source. *The PracT<sub>E</sub>X Journal*, 2005. <http://tug.org/pracjourn/2005-1/allen/allen.pdf>.
  - [2] Patrick Daly. Graphics and Colour with L<sup>A</sup>T<sub>E</sub>X. <http://tex.loria.fr/graph-pack/grf/grf.pdf>, 1998.
  - [3] Tomas Morales de Luna. Writing posters in L<sup>A</sup>T<sub>E</sub>X. *The PracT<sub>E</sub>X Journal*, 2008(3). <http://tug.org/pracjourn/2008-3/morales/>.
  - [4] Paulo Rogério de Souza e Silva Filho and Rian Gabriel Santos Pinheiro. Design and Preparation of Effective Scientific Posters using L<sup>A</sup>T<sub>E</sub>X. *The PracT<sub>E</sub>X Journal*, 2010(2). <http://tug.org/pracjourn/2010-2/rogerio.html>.
  - [5] Johan Engelen. How to Include an SVG Image in L<sup>A</sup>T<sub>E</sub>X. <http://tug.ctan.org/tex-archive/info/svg-inkscape/InkscapePDFLaTeX.pdf>, 2010.
  - [6] Peter Flom. L<sup>A</sup>T<sub>E</sub>X for academics and researchers who (think they) don't need it. *The PracT<sub>E</sub>X Journal*, 2005(4). <http://tug.org/pracjourn/2005-4/flom/flom.pdf>.
  - [7] Marius Hofert and Markus Kohm. Scientific Presentations with L<sup>A</sup>T<sub>E</sub>X. *The PracT<sub>E</sub>X Journal*, 2010(2). <http://tug.org/pracjourn/2010-2/hofert.html>.
  - [8] Tobias Kuhn. `bchart`: Simple Bar Charts in L<sup>A</sup>T<sub>E</sub>X. <http://www.ctan.org/pkg/bchart>, 2012.
  - [9] Rouben Rostamian. A Beamer Quickstart. <http://www.math.umbc.edu/~rouben/beamer/quickstart.html>, 2004.
  - [10] Han Lin Shang. Writing Posters with Beamerposter Package in L<sup>A</sup>T<sub>E</sub>X. *The PracT<sub>E</sub>X Journal*, 2012(1). <http://tug.org/pracjourn/2012-1/shang.html>.
- ◇ Paweł Lupkowski  
Institute of Psychology  
Department of Logic and Cognitive Science  
Adam Mickiewicz University  
Poznań, Poland  
[pawel.lupkowski \(at\) gmail dot com](mailto:pawel.lupkowski@gmail.com)  
[http://amu.edu.pl/~p\\_lup/](http://amu.edu.pl/~p_lup/)
- ◇ Mariusz Urbański  
Institute of Psychology  
Department of Logic and Cognitive Science  
Adam Mickiewicz University  
Poznań, Poland  
[Mariusz.Urbanski \(at\) amu dot edu dot pl](mailto:Mariusz.Urbanski@amu.edu.pl)  
<http://mu.edu.pl>