Technical Comments

- **To switch to full-screen, use the option** `View` $\longrightarrow$ `Full Screen`**, to leave, hit the** `<Esc>` **key;**
- **If some superscript is blocked out by a gray box in the following equation:** $(x+y)^{(a_i-b_{i+1}+c_{i-1})^{(2+e^{x+a})}}$**, then uncheck the** `Edit` $\longrightarrow$ `Preferences` $\longrightarrow$ `Display` $\longrightarrow$ `Use Greek Text` **option;**
- `PgUp`, `PgDn` **or the** `arrow` **keys advance pages, however, in the** `Preferences` $\longrightarrow$ `Full Screen` **section you can set the mouse to advance, too. (You can also hide it, overwrite default transition and many more.)**

Demonstration of the `prosper` class

v 1.9a

János Márkus

markusjancsi@users.sourceforge.net

Oregon State University

Department of Electrical and Computer Engineering

OREGON STATE UNIVERSITY

Powered by LaTeX 2ε

Introduction

The `prosper` class (written by Frédéric Goualard)

- **permits producing high quality slides;**

Introduction

The `prosper` class (written by Frédéric Goualard)

- **permits producing high quality slides;**
- **offers all the advantages (and disadvantages) of LaTeX;**

Introduction

The `prosper` class (written by Frédéric Goualard)

- **permits producing high quality slides;**
- **offers all the advantages (and disadvantages) of LaTeX;**
- **freely available;**

Introduction

The `prosper` class (written by Frédéric Goualard)

- **permits producing high quality slides;**
- **offers all the advantages (and disadvantages) of LaTeX;**
- **freely available;**
- **platform independent;**

Introduction

The `prosper` class (written by Frédéric Goualard)

- **permits producing high quality slides;**
- **offers all the advantages (and disadvantages) of LaTeX;**
- **freely available;**
- **platform independent;**
- **easily extendable;**

Introduction

The `prosper` class (written by Frédéric Goualard)

- **permits producing high quality slides;**
- **offers all the advantages (and disadvantages) of LaTeX;**
- **freely available;**
- **platform independent;**
- **easily extendable;**
- **make the task of recycling materials from previously written article(s) easy.**

Prerequisites

Prosper relies on some recent LaTeX packages and other softwares. The following version-numbers are mandatory:

- a recent distribution of LaTeX with the pstricks and seminar packages with seminar.bg2 (May 22, 1998);
- a recent version of hyperref (version $\geq$ 6.70u);
- dvips (version $\geq$ 5.85);
- GhostScript (version $\geq$ 6.5) for handling embedded fonts.

Styles

Prosper offers some predefined and contributed styles to prepare a presentation. These are as follows:

Styles

Prosper **offers some predefined and contributed styles to prepare a presentation. These are as follows:**

- **Predefined Styles**

  default, alienglow, autumn, azure, contemporain, darkblue, frames, lignesbleues, nuancegris, troispoints;

Styles

Prosper **offers some predefined and contributed styles to prepare a presentation. These are as follows:**

- **Predefined Styles**

  default, alienglow, autumn, azure, contemporain, darkblue, frames, lignesbleues, nuancegris, troispoints;

- **Contributed Styles**

  alcatel **(based on** troispoints**)**, blends, gradboy, pascal, rico, gyom.

Transitions

The following transitions (switching effects between two slides) are defined:

- **R or Replace;**

Transitions

The following transitions (switching effects between two slides) are defined:

- R or Replace;
- Split;

Transitions

The following transitions (switching effects between two slides) are defined:

- R or Replace;
- Split;
- Blinds;

Transitions

The following transitions (switching effects between two slides) are defined:

- R or Replace;
- Split;
- Blinds;
- Box;

Transitions

The following transitions (switching effects between two slides) are defined:

- R or Replace;
- Split;
- Blinds;
- Box;
- Wipe;

Transitions

The following transitions (switching effects between two slides) are defined:

- R or Replace;
- Split;
- Blinds;
- Box;
- Wipe;
- Dissolve;

Transitions

The following transitions (switching effects between two slides) are defined:

- R or Replace;
- Split;
- Blinds;
- Box;
- Wipe;
- Dissolve;
- Glitter.

Predefined styles

The next 10 slides show the different styles developed by the author of the class-file (Frédéric Goualard).

Each slide's title contains the stylename and the transition's name which is used to switch to that particular slide.

“Default” style w/ Replace

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

“Alien glow” style w/ Split

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

“Autumn” style w/ Blinds

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

“Azure” style w/ Box

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

“Contemporain” style w/ Wipe

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

OREGON STATE UNIVERSITY

“Dark blue” style w/ Dissolve

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

OREGON STATE UNIVERSITY

"Frames" style w/ Glitter

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{(99^2)^{2n+1}}$$

OREGON STATE UNIVERSITY

“Lignes bleues” style w/ Replace

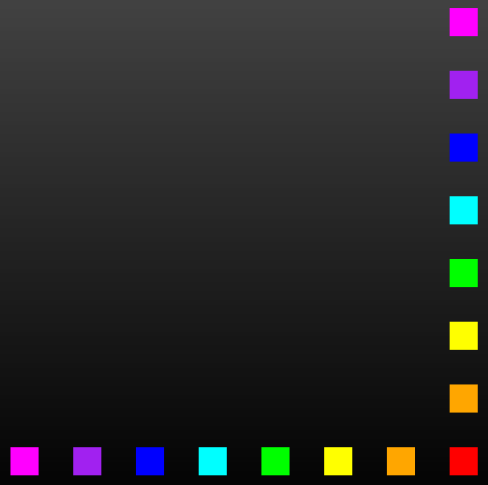
- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{(\frac{1}{4})_n (\frac{2}{4})_n (\frac{3}{4})_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{(99^2)^{2n+1}}$$

“Nuances de gris” style w/ Blinds

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$



“Trois points” style w/ Box

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

Contributed styles

The next 6 slides show the styles developed by different contributors.

Comment: `Alcatel` **slide is based on** `troispoints`**. Using the** `slideColor` **option they look exactly the same.**

“Alcatel” style w/ Wipe

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

“Blends” style w/ Dissolve

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

"Gradient Blue-Orange-Yellow" style w/ Glitter

- **The following formula computes 8 correct digits per iteration (Ramanujan):**

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

“Gyôm personal” style w/ Replace

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

“Pascal” style w/ Split

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

“Rico” style w/ Blinds

- The following formula computes 8 correct digits per iteration (Ramanujan):

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\right)_n \left(\frac{2}{4}\right)_n \left(\frac{3}{4}\right)_n}{n!^3} \left(2\sqrt{2}(1103 + 26390n)\right) \frac{1}{\left(99^2\right)^{2n+1}}$$

Using the class

To start using the class, have a look into the following resources:

- **The homepage of the project;**
- **The documentation of the class (`prosper-doc.pdf`);**
- **Some addition to the documentation – e.g. introducing `letterpaper` option (`prosper-doc-add.pdf`);**
- **Example files in the `doc` directory;**
- **The source of this file (`demo.tex`).**

Testing your workflow

This is a must.

There is a very up-to-date testfile on CTAN, which checks all critical part of your workflow to determine whether your current configuration is capable to produce high-quality ps/pdf files or not.

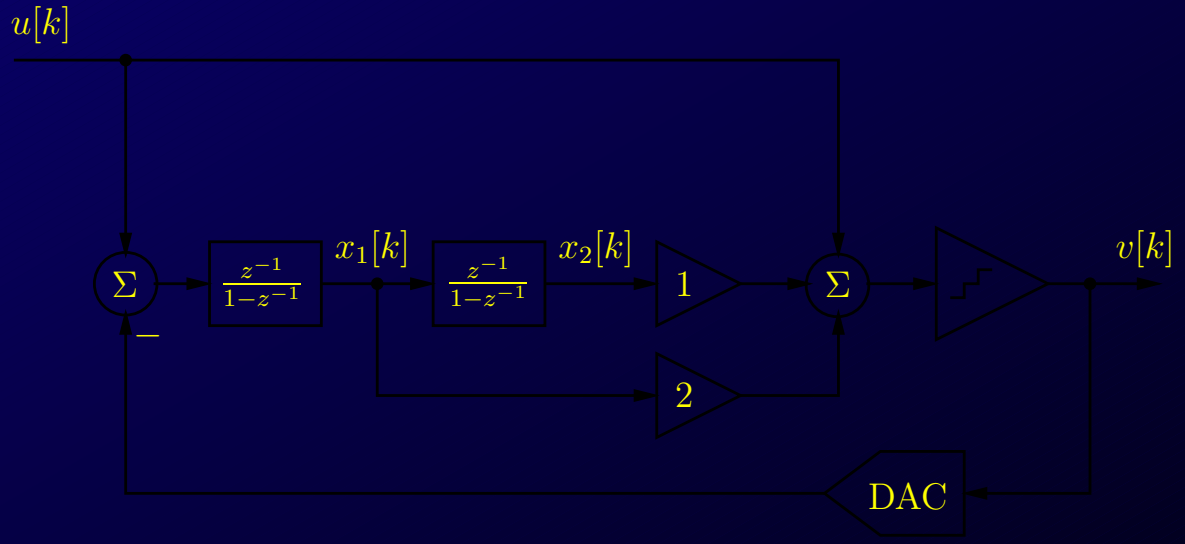
The following URL leads to the directory which has the testfile (`testflow.tex`), ideal output files and a documentation how to detect and correct any errors:

URL: testflow directory on CTAN

Read the documentation (`testflow_doc.txt`) how to use this test.

Recycling xfig-figures (1)

By default, the combined LaTeX and PostScript output of `xfig` gives a transparent background. However, in your paper the figure is probably black. If you include it, it will look like this:



`xfig_normal.eps_t`

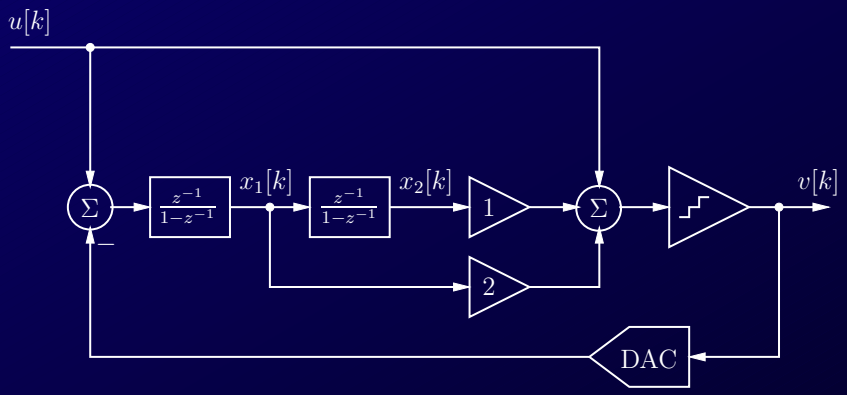
Recycling xfig-figures (2)

Quick-and-dirty trick:

- **Change the fill colors of dots, etc. to white;**
- **Glue every object into one compound;**
- **Update object, set only PenColor option on;**
- **Change PenColor to White;**
- **Apply it to the compound you made (you won't see too much from now on);**
- **Save it to a different file and export it, then include it here.**

Recycling xfig-figures (3)

The text in your figure would inherit the color of the slide text, but, you can make it white with the `\white` command.



`xfig_transp.eps_t`

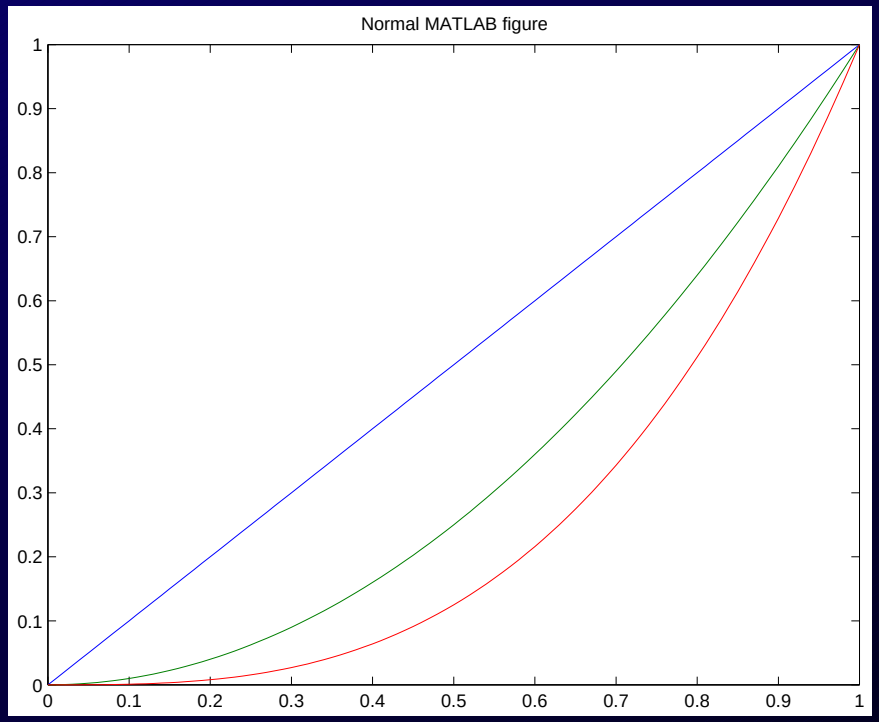
Well, actually you need both versions, one for color and one for white background...

Recycling MATLAB-figures (1)

- **Use `plottransp.m` function shipped with this document;**
- **Use it instead of `plot`;**
- **One can modify it to be used instead of `semilogx`, etc.;**
- **Latest version is here.**

Recycling MATLAB-figures (2)

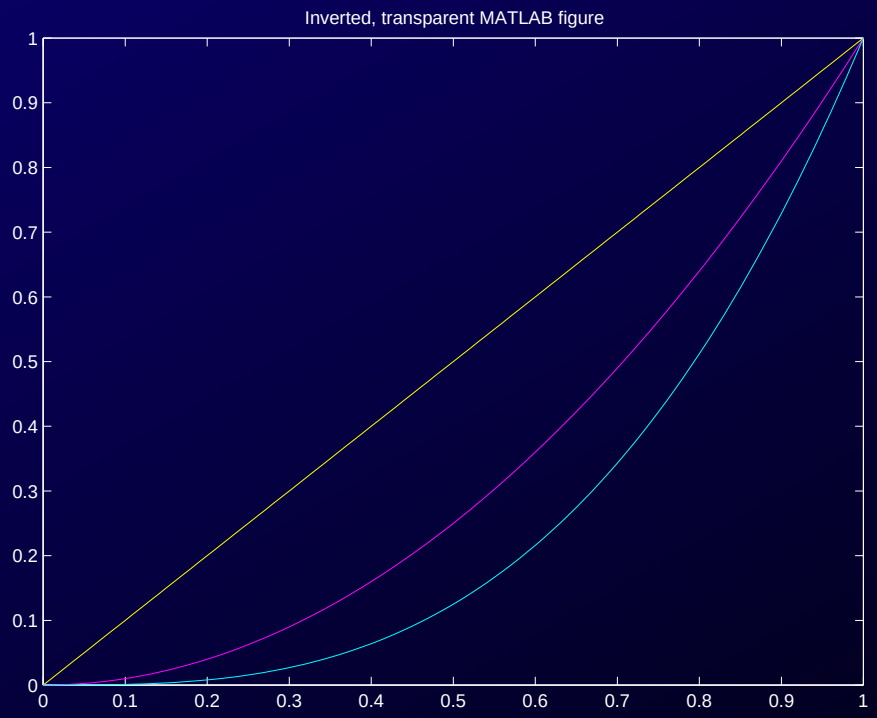
```
plot(0:.01:1,[(0:.01:1);(0:.01:1).^2;(0:.01:1).^3]);
title('Normal MATLAB figure');
print -deps2c matlab_normal.eps
```



matlab_normal.eps

Recycling MATLAB-figures (3)

```
plottransp(0:.01:1,[(0:.01:1);(0:.01:1).^2;(0:.01:1).^3]);
title('Inverted, transparent MATLAB figure');
print -deps2c matlab_transp.eps
```



matlab_transp.eps

Recycling Simulink-diagrams (1)

- **You can print into eps-file by using the** `print -s<modelname> -deps2c <filename.eps>` **command.**
- **Unfortunately, Simulink does not know "transparent colors";**
- **Simulink 3.0.1 (R11.1) does not even print the background (screen) color;**
- **Simulink 4.1 (R12.1) does.**

Recycling Simulink-diagrams (2)

To make the Simulink `eps` file transparent, one have to find the command which fills up the background with white or any other color. It is relative easy to find (at least in these versions).

- **Open the printed eps-file in a text editor;**
- **Search for the following line:**
  `12 dict begin %Colortable dictionary`
  (or part of it, like `Colortable`);

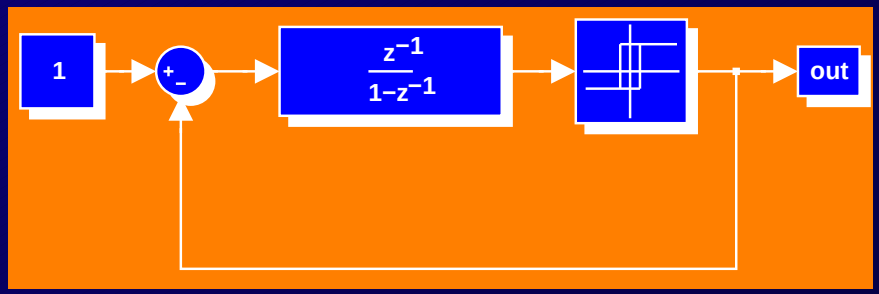
Recycling Simulink-diagrams (3)

- under 8 lines of color definitions, you can see the following two or three lines:

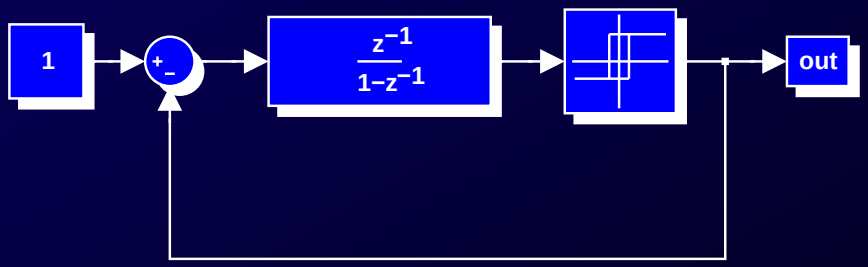
`c0`	this selects black color;
something	if your background color is not black, this line sets up the background color;
`-1 -1` n_1 n_2 `rf`	where n_1 and n_2 are (usually three-digit) numbers.

- The last line (`rf`=`rectfill`) creates the box to be filled. If you uncomment this line using the % character, it results a nice transparent figure.

Recycling Simulink-diagrams (4)



simtest_orange.eps



simtest_transp.eps

Writing new style

To write a new style, the following resources are available:

- **The homepage of the classfile;**
- **The documentation of the class (`prosper-doc.pdf`);**
- **Some addition to the documentation (`prosper-doc-add.pdf`);**
- **various style-files (`PPR*.sty`).**

Acknowledgement

Many thanks to those, who has contributed to the `prosper` class:

- **Timothy Van Zandt (author of `pstricks` and `seminar` classes);**
- **Denis Girou (who is currently maintaining the previous packages);**
- **Frédéric Goualard and the Prosper development team (author and developer(s) of the `prosper` class).**