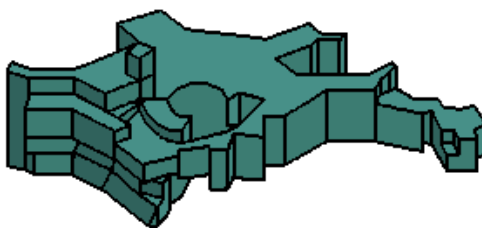# The Prosper package for LaTeX

Presentations without PowerPoint

Georg Drenkhahn

georg@mpa-garching.mpg.de

Introduction

- *Prosper* is a plain LaTeX packet. All common LaTeX macros are available.

Introduction

- *Prosper* is a plain LaTeX packet. All common LaTeX macros are available.

- Produces Postscript for printing slides and *P*ortable *D*ocument *F*ormat (PDF) files for presentations.

Introduction

- *Prosper* is a plain LaTeX packet. All common LaTeX macros are available.

- Produces Postscript for printing slides and *P*ortable *D*ocument *F*ormat (PDF) files for presentations.

- Presentations are viewed with the Acrobat reader. Common on many computer systems.

Introduction

- *Prosper* is a plain LATEX packet. All common LATEX macros are available.

- Produces Postscript for printing slides and *P*ortable *D*ocument *F*ormat (PDF) files for presentations.

- Presentations are viewed with the Acrobat reader. Common on many computer systems.

- System independent (Windows, UNIX, Mac, . . .)

Features

- Simple structure, few new commands

Features

- Simple structure, few new commands
- New step-by-step itemisation environment (as seen)

Features

- Simple structure, few new commands
- New step-by-step itemisation environment (as seen)
- Various transition types between slides: Replace, Split, Blinds, Box, Wipe, Dissolve, Glitter

Features

- Simple structure, few new commands

- New step-by-step itemisation environment (as seen)

- Various transition types between slides: Replace, Split, Blinds, Box, Wipe, Dissolve, Glitter

- Possibility to create own styles

Add, replace, delete

Items can be added

Add, replace, delete

. . . or replaced

Add, replace, delete

Items can be added, replaced and deleted

Add, replace, delete

Items can be added, replaced and deleted
This works in normal text, itemisation environments and math formulae:
Induction equation (ideal MHD)

$$\partial_t \vec{B} = \mathbf{curl}\left(\vec{v} \times \vec{B}\right)$$
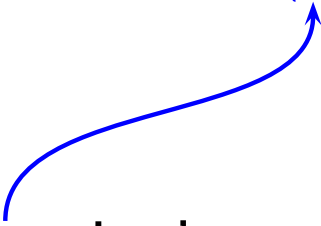
Add, replace, delete

Items can be added, replaced and deleted
This works in normal text, itemisation environments and math formulae:
Induction equation

$$\partial_t \vec{B} = \mathbf{curl}\left(\vec{v} \times \vec{B}\right) + \frac{c^2}{4\pi\sigma}\triangle\vec{B}$$

'flux freezing'   diffusion term

Standard styles

Alienglow

Autumn

Azure

Contemporain

Standard styles 2

Darkblue

Frames

Lignesbleues

NuanceGris

Standard styles 3

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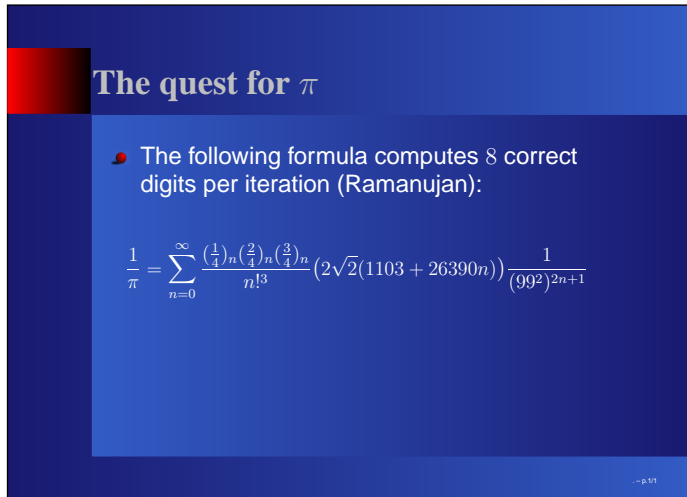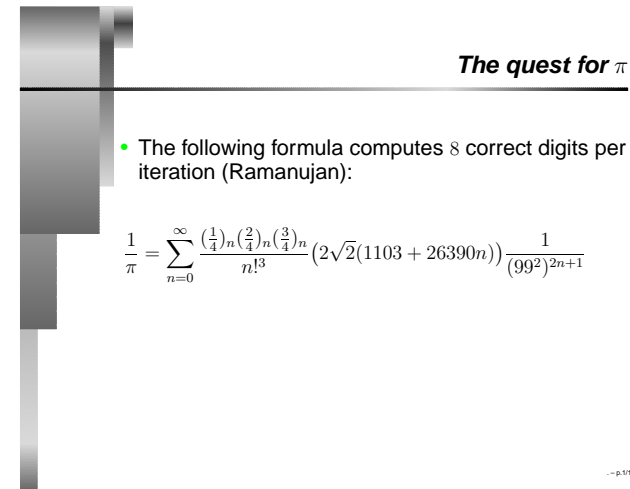
- The following formula computes 8 correct digits per iteration (Ramanujan):

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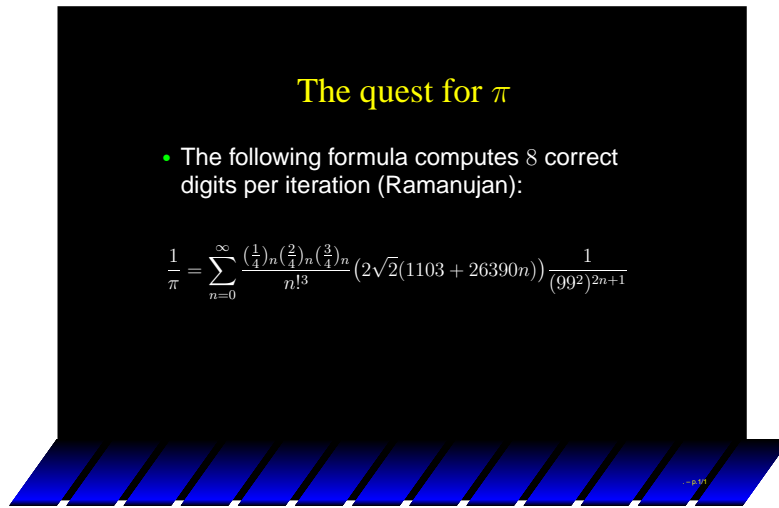
TroisPoints

The quest for π

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

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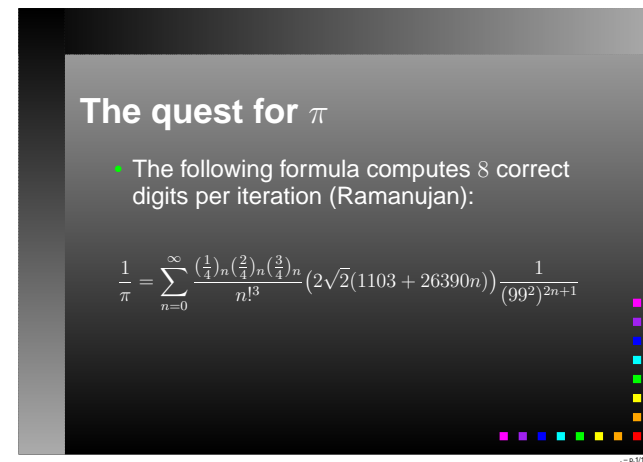
gyom

The quest for π

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

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

rico

Structure

```
\documentclass[pdf,mpa]{prosper}
\title{Some presentation}
...
\maketitle
\begin{slide}{1. Slide}
  normal text/graphics on slide
\end{slide}
\overlays{4}{
  \begin{slide}{Slide with more parts}
    ...
  \end{slide}
}
\begin{document}
```

Overlays

```
\overlay{4}{
  \begin{slide}{Title}
    \onlySlide*{1}{...}
    \fromSlide{2}{...}
    \untilSlide{3}{...}
    \FromSlide{4}

    ...
  \end{slide}
}
```

Creation from source

- `latex` (twice for cross-references):
  .tex → .dvi

Creation from source

- `latex` (twice for cross-references):
  .tex → .dvi

- for slides: `dvips`:
  .dvi → .ps

Creation from source

- `latex` (twice for cross-references):
  .tex → .dvi

- for slides: `dvips`:
  .dvi → .ps

- for pdf presentations: `dvips`,`ps2pdf` or `dvi2pdf`:
  .dvi → .pdf

Creation from source

- `latex` (twice for cross-references):
  .tex → .dvi

- for slides: `dvips`:
  .dvi → .ps

- for pdf presentations: `dvips`,`ps2pdf` or `dvi2pdf`:
  .dvi → .pdf

- dvi2pdf is a Perl script in the pdsoft area. Takes care of all pdf issues: vector fonts, paper size, optimisation.