

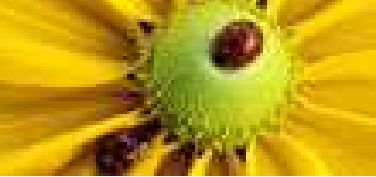
Creative use of PDF files in L^AT_EX environments

What Adobe won't tell you ...

Ralf Koenig

Chemnitz University of Technology, Dept. of Computer Science

<http://www.tu-chemnitz.de/informatik>



Brainstorming

- The World of PDF and \LaTeX

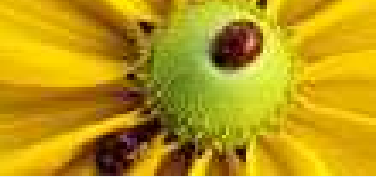
PDF and Postscript

Font Tools

Page Extraction and Arranging

Thanks four your attention.

Brainstorming



The World of PDF and L^AT_EX

Brainstorming

● The World of PDF and L^AT_EX

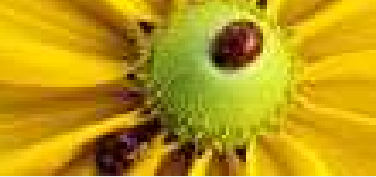
PDF and Postscript

Font Tools

Page Extraction and Arranging

Thanks for your attention.

- pdflatex, tex \Rightarrow pdf
- dvi_{pdfm}, tex \Rightarrow dvi \Rightarrow pdf
- ps2pdf (ghostscript) tex \Rightarrow dvi \Rightarrow ps \Rightarrow pdf
- pdftmark (hyperref)
- Adobe Distiller ps \Rightarrow pdf
- Adobe Acrobat (edit pdfs) and Adobe Reader (show pdfs)
- epdf (Encapsulated PDF)
- thumbpdf (create thumbnails, obsolete now)
- encryption of PDFs
- no eps images, why?
- jpg, bitmap and other image formats
- bad rendering of bitmap fonts (foremost in Acrobat Reader)
- using standard postscript fonts leads to small PDFs
- pstricks, pdftricks, ps4pdf



PDF and Postscript

Brainstorming

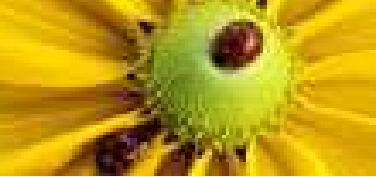
PDF and Postscript

- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- PDF Tree
- Compressed streams
- Uncompressing streams
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.



History

Brainstorming

PDF and Postscript

● History

- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- PDF Tree
- Compressed streams
- Uncompressing streams
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.

■ created by Adobe founder John Warnock in 1990.

■ web suitable brother of Postscript

■ versions:

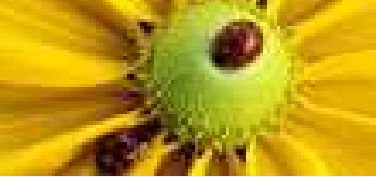
PDF 1.1 Acrobat 2

PDF 1.2 Acrobat 3

■ PDF 1.3 Acrobat 4

PDF 1.4 Acrobat 5

PDF 1.5 Acrobat 6



Common to PS and PDF

Brainstorming

PDF and Postscript

● History

● Common to PS and PDF

● Differences between PS and PDF

● PS to PDF and vice versa

● latex+tools vs. pdftex

● Demo

● Structure of a PDF

● Demo

● PDF Tree

● Compressed streams

● Uncompressing streams

● Extracting parts

● Extracting fonts

● Protection against Font Ripping

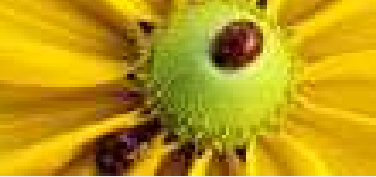
● Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.

- describe physical pages with graphical elements
- many printing related features: support of CMYK
- have similar operator sets and same logic for drawing commands
- only difference: operator names were abbreviated in PDF to reduce file sizes (adopted from Adobe Illustrator AI format)



Differences between PS and PDF

Brainstorming

PDF and Postscript

- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- PDF Tree
- Compressed streams
- Uncompressing streams
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.

Postscript

stack-based programming language with all the features of a programming language

only printable objects

pages depend on each other (linear structure)

only fonts and images make sense to be embedded

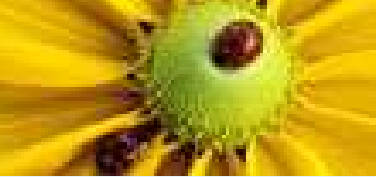
PDF

page description language (+ JavaScript for limited programmability)

additional web features: hyperlinks, forms, encryption, more built-in compression algorithms

pages are independent from each other (tree structure)

objects of almost every type can be embedded (multimedia items such as sounds, movies, clickable buttons)



PS to PDF and vice versa

Brainstorming

PDF and Postscript

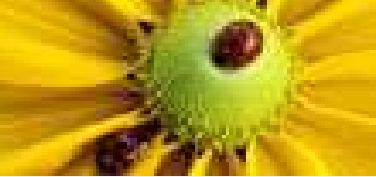
- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- PDF Tree
- Compressed streams
- Uncompressing streams
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.

- From PS to PDF (e.g. for deployment on the web):
 - ◆ process the PS, compute all the drawing commands (only these will be shown, all the code will be removed)
 - ◆ compute a graphics state for every page (this graphics state will make the pages independant from each other)
 - ◆ replace all the custom macros by the relevant drawing commands
 - ◆ compress embedded objects (fonts, images)
- From PDF to PS (eg. for laser printers with built-in Postscript RIP):
 - ◆ replace the short names by the long names (or input mapping code)
 - ◆ decompress and convert the font and image objects
 - ◆ disregard the rest (non printable objects such as JavaScript code, Annotations, . . .)



latex+tools vs. pdftex

Brainstorming

PDF and Postscript

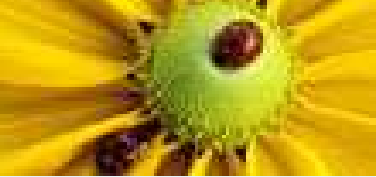
- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- PDF Tree
- Compressed streams
- Uncompressing streams
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.

- pdftex is the integrated approach, direct control over output
- latex, dvips, gs is the traditional approach
- more features in stable versions of gs than in pdftex
 - ◆ gs supports encryption, only special `pdftex` versions support encryption (see `pdfcrypt`)
 - ◆ gs (since version 7) (loss-lessly) compresses fonts to Compact Font Format (also called Type1C or Type2), `pdflatex` only embeds CFF data from OTF fonts (commercial $\text{T}_{\text{E}}\text{X}$ s (Bakoma, VTeX) have more complete support for CFF)
 - ◆ BTW, Han The Thanh works on `pdfTeX 2.0` (current 1.11b)



Demo

Brainstorming

PDF and Postscript

- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex

● Demo

- Structure of a PDF
- Demo
- PDF Tree
- Compressed streams
- Uncompressing streams
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

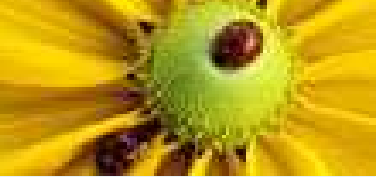
Page Extraction and Arranging

Thanks four your attention.

```
latex test1.tex  
dvips test1.tex  
ps2pdf test1.tex
```

```
pdftex test1.tex
```

- see the difference: Type1C vs. Type1
- BTW: different default paper size by pdflatex and latex in this MikTeX version



Structure of a PDF

Brainstorming

PDF and Postscript

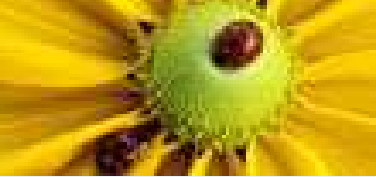
- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- **Structure of a PDF**
- Demo
- PDF Tree
- Compressed streams
- Uncompressing streams
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.

- tree of objects
- each object has an object number and revision level, such as
`75 0 obj`
- encoding or compression algorithms can be applied to streams (object content)
- built-in lookup table with offsets to find objects in $O(1)$ \Rightarrow makes manipulations difficult
- look-up table is (more or less) repaired when invalid



Demo

Brainstorming

PDF and Postscript

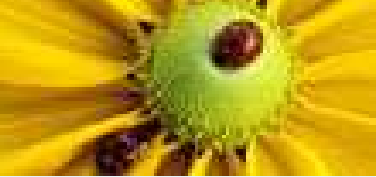
- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- PDF Tree
- Compressed streams
- Uncompressing streams
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.

- let's start with a \LaTeX one liner: Hello World.
- `pdflatex test1`
- Element:
 - ◆ Version tag
 - ◆ 3 stream object, compressed with zip
 - ◆ 2 Page Object
 - ◆ 1 Font
 - ◆ 5 stream object, compress with zip
 - ◆ 6 Font
 - ◆ 4 Dictionary
 - ◆ 9 enum object
 - ◆ 8 Encoding
 - ◆ 7 Pages
 - ◆ 10 Catalog
 - ◆ 11 Dictionary
 - ◆ 12 Trailer



PDF Tree

This is a tree!!

```
trailer (12) - info node: (11)
  - catalog (10) - pages (7) - page (2) - contents: (3)
    - resources (1) - font (6) - encoding (8)
      - widths (9)
      - fontdesc (4)
      - fontfile (5)
```

Brainstorming

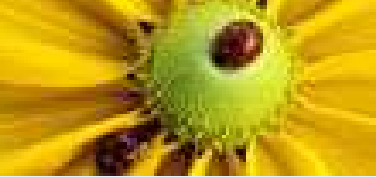
PDF and Postscript

- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- **PDF Tree**
- Compressed streams
- Uncompressing streams
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.



Compressed streams

Brainstorming

PDF and Postscript

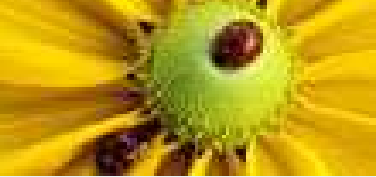
- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- PDF Tree
- **Compressed streams**
- Uncompressing streams
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.

- page contents is a postscript like program with shortened operators
- many of the operators resemble postscript ones (m moveto, l lineto, d setdash, h closepath . . .)
- a postscript representation can be generated by adding a preamble of macro mapping code (proc set)
- represents text, vector graphics, small inline bitmap graphics and references to XObjects
- XObject are resources in separate objects, usually in compressed form such as images



Uncompressing streams

Brainstorming

PDF and Postscript

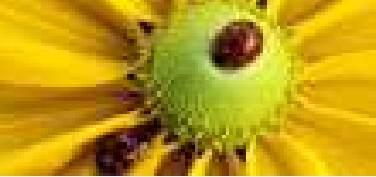
- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- PDF Tree
- Compressed streams
- **Uncompressing streams**
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.

- for \LaTeX documents: `\pdfcompresslevel=0`
- for any PDF: extract and uncompress a *single* object:
`pdftosrc <file> <object-nr>`
- for any PDF: uncompress *all* compressed objects *inside* the PDF file with Help of tools from Multivalent:
`java -jar Multivalent.jar
edu.multivalent.tools.pdfUncompress
file-in.pdf file-out.pdf`
- There is also a Compress in Multivalent Tools.



Extracting parts

Brainstorming

PDF and Postscript

- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- PDF Tree
- Compressed streams
- Uncompressing streams
- **Extracting parts**
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.

almost every PDF parser includes a few sample applications to extract certain kinds of information from PDF files

Bitmap Images. in XObjects: `pdftosrc` (one single image, low-level) or `pdfimages` (from `xpdf`, converts bitmap graphics to pnm, JPEG is left in DCT coded form, no loss)

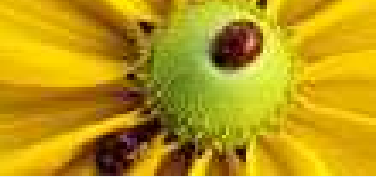
Text. `pdftotext` (from `xpdf`), be careful with CID-type fonts used in document, you must specify the encoding on the command line, it may occur that text is not in its natural order

subsetted fonts sometimes need an extra encoding vector

Vector Graphics. conversion to eps: `pdftops` (`xpdf`) or `ghostscript`, there is commercial software for conversion to svg (`pdf2svg`), I do not know of any free one.

Alternatively, extract page and alter `MediaBox` or `CropBox`.

Fonts. see next slide



Extracting fonts

Brainstorming

PDF and Postscript

- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- PDF Tree
- Compressed streams
- Uncompressing streams
- Extracting parts
- **Extracting fonts**
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

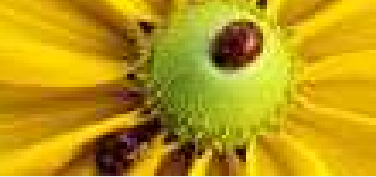
Page Extraction and Arranging

Thanks four your attention.

- find the right objects (`FontFile`)
- extract them with `pdftosrc`
- Type1 and Truetype fonts are ready at this point
- use `cfftot1` to uncompress Type1C fonts
- inspect the result in a font editor such as `fontforge` (formerly `pfaedit`)
- generate fake metrics, `gs printafm.ps` or (better) with a font editor
- All this can be automated to extract all fonts embedded in a PDF file.
- Imagine this in conjunction with a web crawler or Google search: file type pdf plus name of the font.



<http://www.linotype.com/14330/birkaroman-fontsample.html>



Protection against Font Ripping

Brainstorming

PDF and Postscript

- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- PDF Tree
- Compressed streams
- Uncompressing streams
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- Encrypting pdf files

Font Tools

Page Extraction and Arranging

Thanks four your attention.

- the Standard Security simply sets some flags and depends on the behaviour of the viewer, in other words: itt's useless anyway
- full encryption may not be an option, as your partners shall see the contents of the PDF without a password
- finally only option: subsetting (standard with most newer PDF making software)
- This is typically also part of most license agreements. Sometimes a special web embedding license is required.

Encrypting pdf files

Brainstorming

PDF and Postscript

- History
- Common to PS and PDF
- Differences between PS and PDF
- PS to PDF and vice versa
- latex+tools vs. pdftex
- Demo
- Structure of a PDF
- Demo
- PDF Tree
- Compressed streams
- Uncompressing streams
- Extracting parts
- Extracting fonts
- Protection against Font Ripping
- **Encrypting pdf files**

Font Tools

Page Extraction and Arranging

Thanks four your attention.

- different security handlers: 40 bits and 128 bits (since PDF 1.4)

- patched pdflatex: package `pdfcrypt` by Heiko Oberdiek

- generic for all PDFs

- ◆ Ghostscript:

```
http://casper.ghostscript.com/~ghostgum/pdftips.htm#encrypt
```

- ◆ `ps2pdf14 -sOwnerPassword#owner`

```
-sUserPassword#user -dEncryptionR#3
```

```
-dKeyLength#128 -dPermissions#-3904 in.ps
```

```
out.pdf
```

Permissions: Print, Extract text and graphics, ...

- ◆ Multivalent Tools:

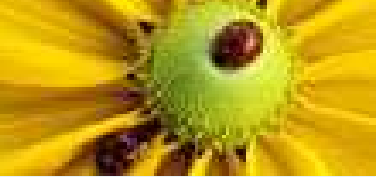
```
http://multivalent.sourceforge.net/Tools/index.html
```

```
java tool.pdf.Encrypt [options]
```

```
PDF-files-or-directories
```

- ◆ DTK 1/2002, S.22 : Tool `pdlin` by Glance AG (commercial)

- Password Recovery: <http://www.elcomsoft.com/prs.html>



Brainstorming

PDF and Postscript

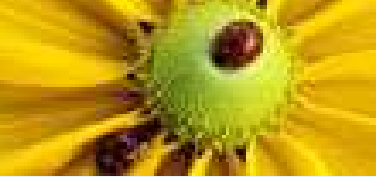
Font Tools

- T1 Utilities
- LCDF Typetools
- Multivalent Font Tools

Page Extraction and Arranging

Thanks four your attention.

Font Tools



T1 Utilities

[Brainstorming](#)

[PDF and Postscript](#)

[Font Tools](#)

● **T1 Utilities**

● LCDF Typetools

● Multivalent Font Tools

[Page Extraction and Arranging](#)

[Thanks four your attention.](#)

<http://www.lcdf.org/~eddielwo/type/#t1utils>

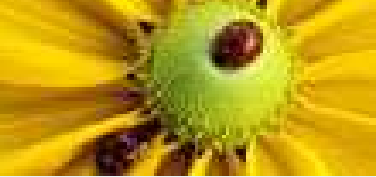
t1ascii, t1binary `t1ascii` changes PFB (binary) fonts into PFA (ASCII) format; `t1binary` goes the opposite direction. *Good for UNIX fonts where PFA seems to be common.*

t1disasm, t1asm `t1disasm` translates PFBs or PFAs into a human-readable and -editable format; `t1asm` goes the opposite direction. *Most useful for font manipulation and font comparison.*

t1mac, t1unmac `t1unmac` (formerly `unpost`) translates a Macintosh Type 1 font into either PFB or PFA format, and `t1mac` goes the opposite direction. *I do not own a Mac.*

Demo

Results: font outlines are described by small postscript programs, hints are `vstem`, `hstem`, `dstem`



LCDF Typetools

[Brainstorming](#)

[PDF and Postscript](#)

[Font Tools](#)

● [T1 Utilities](#)

● **[LCDF Typetools](#)**

● [Multivalent Font Tools](#)

[Page Extraction and Arranging](#)

[Thanks four your attention.](#)

<http://www.lcdf.org/~eddielwo/type/#typetools>

cfftot1 cff to Type 1, handles hints and subrs.

mmafm, mmpfb create instances of multiple master fonts.

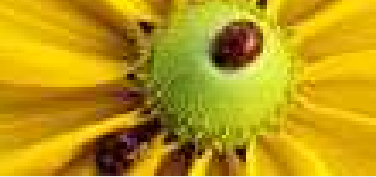
otfinfo list features of OpenTyp fonts.

otftotfm extract metrics from Postscript flavoured Opentype fonts.

t1dotlessj create matching dotless j for a font.

t1lint checks a Type 1 font for correctness.

t1testpage Creates a PostScript proof for a Type 1 font.
Preliminary.



Multivalent Font Tools

[Brainstorming](#)

[PDF and Postscript](#)

[Font Tools](#)

● T1 Utilities

● LCDF Typetools

● **Multivalent Font Tools**

[Page Extraction and Arranging](#)

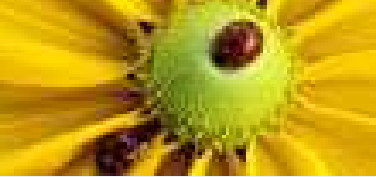
[Thanks four your attention.](#)

<http://multivalent.sourceforge.net/Tools/index.html>

Info on Type 1, TrueType, and OpenType fonts.

Convert between Type 1 PFA and PFB, decrypt Type 1, or convert Mac resource font.

View fonts in system, from files or directories, or even embedded in PDF.



Brainstorming

PDF and Postscript

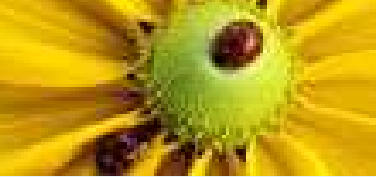
Font Tools

Page Extraction and Arranging

- Multivalent PDF Tools
- Package `pdfpages`
- `pdfpages` usage
- Sample control file

Thanks four your attention.

Page Extraction and Arranging



Multivalent PDF Tools

[Brainstorming](#)

[PDF and Postscript](#)

[Font Tools](#)

[Page Extraction and Arranging](#)

[Multivalent PDF Tools](#)

[Package pdfpages](#)

[pdfpages usage](#)

[Sample control file](#)

[Thanks four your attention.](#)

<http://multivalent.sourceforge.net/Tools/index.html>

Split also rearrange or delete pages.

Merge

Impose simple n-up, booklets, sophisticated imposition

Validate

Undo



Package pdfpages

From T_EX Catalogue Online:

This package makes it easy to include pages from external PDF documents in LaTeX documents (using pdf_latex). It is possible to arrange two or four logical pages from a PDF document on each physical sheet of paper from LaTeX. It is similar to includegraphics in some respects. Landscape format is supported. Pages are centered automatically. The package provides hypertext operations like hyperlinks and article threads. You can select pages to be included with, e.g.,
`pages=3 , 5-11 , 20 , 38`. Support is included for VTeX. This package was implemented by Andreas Matthias.

Brainstorming

PDF and Postscript

Font Tools

Page Extraction and Arranging

● Multivalent PDF Tools

● **Package pdfpages**

● pdfpages usage

● Sample control file

Thanks four your attention.



pdfpages usage

Brainstorming

PDF and Postscript

Font Tools

Page Extraction and Arranging

● Multivalent PDF Tools

● Package pdfpages

● pdfpages usage

● Sample control file

Thanks four your attention.

- a control file is needed that includes the pdfpages options

- psbook like approach, that generates the control file:

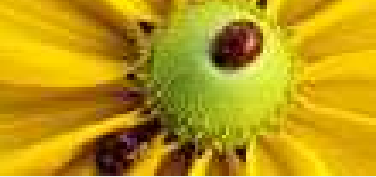
`http://www.tug.org/pipermail/pdftex/2004-February/00476`

- pdfpages

- ◆ `pdflatex myfile // make the source file`

- ◆ `pdflatex control-file`

Sample control file



Brainstorming

PDF and Postscript

Font Tools

Page Extraction and Arranging

● Multivalent PDF Tools

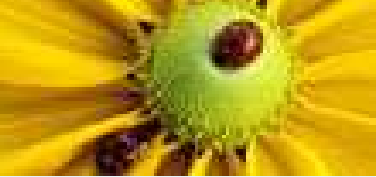
● Package pdfpages

● pdfpages usage

● Sample control file

Thanks four your attention.

```
\documentclass[a4paper]{scrartcl}
\usepackage{pdfpages}
\begin{document}
% Beispiel fr 8-seitiges Dokument
\includepdf[nup=2, landscape, pages={
    8,1,2,7, %% diese Zahlen musst du selbst zusammenbauen
    6,3,4,5 %% mit einer Tabellenverarbeitung
}]{myfile} % % hier wird dein Dokument genannt
% Beispiel fr 11-seitiges Dokument
\includepdf[nup=2, landscape, pages={
    -,1,2,11, %% - heit Leerseite
    10,3,4,9,
    8,5,6,7
}]{myfile}
\end{document}
```



[Brainstorming](#)

[PDF and Postscript](#)

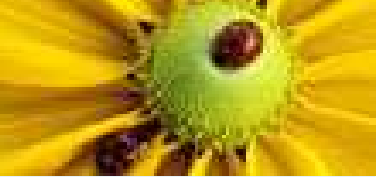
[Font Tools](#)

[Page Extraction and Arranging](#)

Thanks four your attention.

● How this document was made

Thanks four your attention.



How this document was made

Brainstorming

PDF and Postscript

Font Tools

Page Extraction and Arranging

Thanks four your attention.

● How this document was made

- \LaTeX , dvips, ps2pdf
 - ◆ `latex slides.tex`
 - ◆ `dvips -o slides.ps slides.dvi`
 - ◆ `ps2pdf slides.ps`
- document class: `prosper.cls`
- additional package: `HA-prosper.sty`
- style: `HAPHA.sty` (comes with `HA-prosper`)
- Direct generation with `pdflatex` is not possible due to `prosper` using `pstricks`, obviously noone so far thought about using `pdftricks` and `pdflatex`, so far.