

Writing, running and including the output of external documents from within a main L^AT_EX document –v. 0.31

Herbert Voß herbert@dante.de*

20. Juni 2022

Inhaltsverzeichnis

1	Loading the package	3
2	Syntax	3
3	First examples	3
3.1	Without showing the code	3
3.2	Showing code and output of a Python example	4
4	Setting marker in the source	7
5	Optional arguments	8
5.1	Programs and runs	8
5.2	Grafik options	10
5.3	Listings options	11
5.4	Background color	11
5.5	Type of the source code	12
5.6	Output more than one page	12
5.7	Output as floating object with caption and label	13
5.8	Cropping the PDF	14
5.9	Code and output side by side	14
5.10	Horizontal alignment of the output	15
5.11	Inline images	16
5.12	Input text instead of an image	16
5.13	Running additional external programs	18
5.14	Using listings	19
5.15	Vertical space	22
5.16	No output	22
6	Defining new marker	23

*Thanks to Karl Berry; Rolf Niepraschk

7	Supported engines	23
7.1	METAPOST example	23
7.2	plainT _E X example	24
7.3	LuaL _A T _E X example	25
7.4	ConT _E Xt example	26
8	Running external commands	27
9	Other options	28

1 Loading the package

```
\usepackage[Option]{hvextern}
```

There exists only one option `checkCode` which is only valid for the use of `LuaLaTeX`. In this case an already existing external file will only be compiled, if the code changed. This doesn't depends on the setting of the option `force`. The package option will speed up the compilation time.

2 Syntax

This package allows to write external `METAPOST`, `TeX`, `ConTeXt`, `LaTeX`, `LuaTeX`, `LuaLaTeX`, `XLaTeX`, `XLaTeX`, `Lua`, `Perl`, `Java` and/or `Python` source code, which will then be run via `shell escape` to create a PDF oder text output to include it into the main `LaTeX` document.

There is only one environment and one command:

```
\begin{externalDocument}[<options>]{<external filename without extension>}
...
source code
...
\end{externalDocument}

\runExtCmd[<options>]
  {<command with arguments>}
  {<external filename without extension>}
```

The main document *must* be run with the `--shell-escape` option, otherwise it won't work, e.g.:

```
lualatex --shell-escape <file>
```

The purpose for this package is to show the output of documents which have to be compiled with a different preamble or a different engine or a complete different system, but integrating the output automatically in the main document..

All examples in this document are created on-the-fly while running this `LaTeX` document with `lualatex` with the `--shell-escape` option.

3 First examples

3.1 Without showing the code

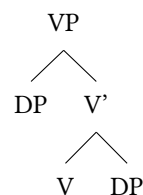
This document was run with `LuaLaTeX`. Suppose you want to insert the output of a document which needs for several reasons a `XLaTeX` run. Instead of created and running a document outside of the main document and then to insert the output we can do this from within this `LuaLaTeX` document itself. The external document is compiled with `XLaTeX` and the output is insert as pdf image: 美好的一天.

Another example which needs a `pdfLaTeX` run. The source code itself is also not shown by the environment `externalDocument`.

```

\begin{externalDocument}[
  compiler=pdflatex,force=false,cleanup]{voss
}
\documentclass{standalone}
%StartVisiblePreamble
\usepackage{fontenc}
\usepackage{libertinus}
\usepackage[linguistics]{forest}
\forestapplylibrarydefaults{linguistics,
  edges}
%StopVisiblePreamble
\begin{document}
\begin{forest}
[VP
  [DP]
  ['V
    [V]
    [DP]
  ]
]
\end{forest}
\end{document}
\end{externalDocument}

```



3.2 Showing code and output of a Python example

The png image is created on the fly with the following arguments of externalDocument:

```

\begin{externalDocument}[
  compiler=python3,
  code,
  ext=py,
  docType=py,
  usefancyvrb,
  grfOptions={width=\linewidth}]{python}
... Python code ...
\end{externalDocument}

```

The code which is declared as header and main can be marked by:

```

\hv@extern@exampleType{py}
{\NumChar StartVisibleMain}
{\NumChar StopVisibleMain}
{\NumChar StartVisiblePreamble}
{\NumChar StopVisiblePreamble}

```

\NumChar is the default Python comment character # and needs to be saved with a different category, which is done internally by the package. The complete definition of the code is:

```

\begin{externalDocument}[
  compiler=python3,
  code,

```

```

ext=py,
force=false,
docType=py,
usefancyvrb,
grfOptions={width=\linewidth}}{python}
import os
#StartVisiblePreamble
from PIL import Image
import subprocess
# drawing area (xa < xb and ya < yb)
xa = -0.1716
xb = -0.1433
ya = 1.022
yb = 1.044
maxIt = 1024 # iterations
imgx = 1000 # image size
imgy = 750
image = Image.new("RGB", (imgx, imgy))
#StopVisiblePreamble

#StartVisibleMain
for y in range(imgy):
    cy = y * (yb - ya) / (imgy - 1) + ya
    for x in range(imgx):
        cx = x * (xb - xa) / (imgx - 1) + xa
        c = complex(cx, cy)
        z = 0
        for i in range(maxIt):
            if abs(z) > 2.0: break
            z = z * z + c
        r = i % 4 * 6
        g = i % 8 * 32
        b = i % 16 * 16
        image.putpixel((x, y), b * 65536 + g * 256 + r)
#StopVisibleMain
# now get the filename created by the latex document
imageName = os.path.basename(os.path.splitext(__file__)[0])+".png" # get filename
image.save(imageName, "PNG")
\end{externalDocument}

```

And with using this code we get the image as png inserted. The given filename of the external document is internally extended by a consecutive number which isn't known to the Python code. However, it is no problem in any programming language to get the name of a running file. The forlast line in the above code shows how it can be done with Python.

```

from PIL import Image
import subprocess
# drawing area (xa < xb and ya < yb)
xa = -0.1716

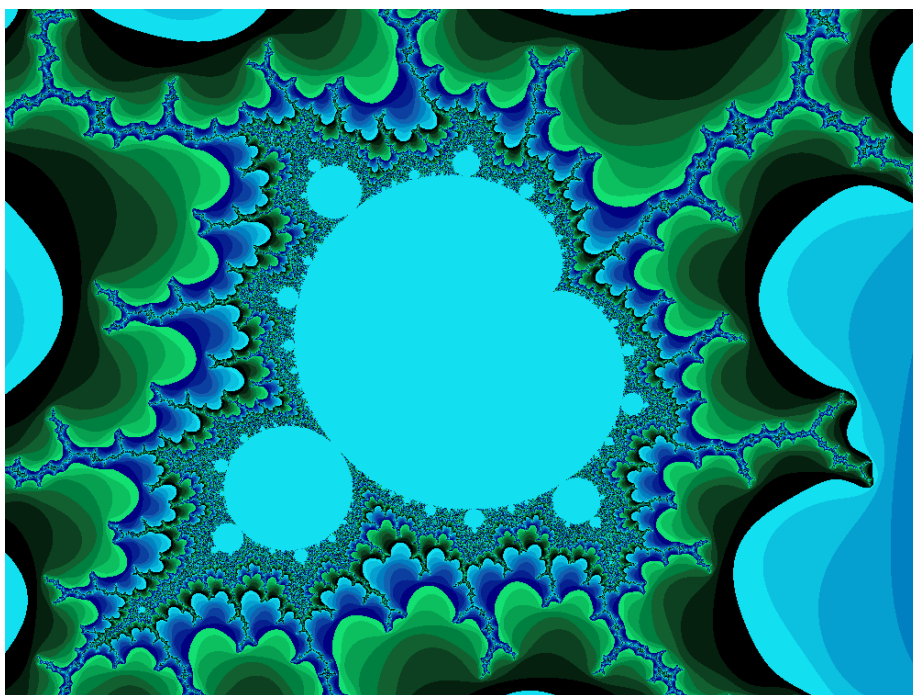
```

```

python-3.py
xb = -0.1433
ya = 1.022
yb = 1.044
maxIt = 1024 # iterations
imgx = 1000 # image size
imgy = 750
image = Image.new("RGB", (imgx, imgy))

for y in range(imgy):
    cy = y * (yb - ya) / (imgy - 1) + ya
    for x in range(imgx):
        cx = x * (xb - xa) / (imgx - 1) + xa
        c = complex(cx, cy)
        z = 0
        for i in range(maxIt):
            if abs(z) > 2.0: break
            z = z * z + c
        r = i % 4 * 6
        g = i % 8 * 32
        b = i % 16 * 16
        image.putpixel((x, y), b * 65536 + g * 256 + r)

```

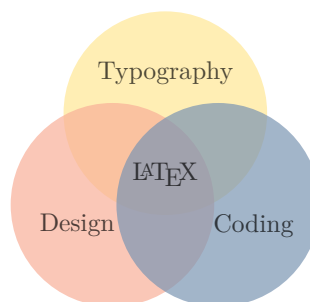


The external filename, extended by a consecutive number, can be printed in the margin by setting the keyword `showFilename`. In general it is printed in the outer margin or in twocolumn mode in the outer column. If the example is set in twocolumn mode but inside a starred floating environment over both column, then use the keyword `outerFN`. Then `hvxtern` doesn't test for twocolumn mode.

A vertical shift of the filename is possible by setting a length to the keyword `shiftFN`, e.g. `shiftFN=5ex`.

```
\usepackage{tikz}
\usepackage[hks,pantone,xcolor]{xespotcolor}

\SetPageColorSpace{HKS}
\definecolor{HYellow}{spotcolor}{HKS05N,0.5}
\definecolor{HRed}{spotcolor}{HKS14N,0.5}
\definecolor{HBlue}{spotcolor}{HKS38N,0.5}
\begin{tikzpicture}[scale=0.7,fill opacity
=0.7]
\fill[HYellow]( 90:1.2) circle (2);
\fill[HRed] (210:1.2) circle (2);
\fill[HBlue] (330:1.2) circle (2);
\node at ( 90:2) {Typography};
\node at ( 210:2) {Design};
\node at ( 330:2) {Coding};
\node {\LaTeX};
\end{tikzpicture}
```



4 Setting marker in the source

The marker for the code ranges which should be listed depend to the used programming language:

```
[...]
%StartVisiblePreamble
[... listed preamble code ]
%StopVisiblePreamble
[...]
\begin{document}
[...]
\end{document}
```

everything between `%StartVisiblePreamble` and `%StopVisiblePreamble` will be listed as preamble and in case of a `L\LaTeX` source everything between `\begin{document}` and `\end{document}` as body. The marker must be defined with an own macro, e.g.:

```
\hv@extern@exampleType{py}
{\NumChar StartVisibleMain}
{\NumChar StopVisibleMain}
{\NumChar StartVisiblePreamble}
{\NumChar StopVisiblePreamble}
```

`\NumChar` is the comment character `#`, which needs a special handling. This version of `hvxextern` supports the following programming languages (option `compiler`): `mpost`, `tex`, `latex`, `luatex`, `python3`, `perl`, `lua`, `xetex`, `pdflatex`, `lualatex`, `xelatex`, and `context`. The default is `pdflatex`. The option `docType` selects the config file, which must be one of `context`, `lua`, `pl`, `tex`, `latex`, `mp`, and `py`. For Lua it is

```
\hv@extern@exampleType{lua}
{--StartVisibleMain}
```

```

{--StopVisibleMain}
{--StartVisiblePreamble}
{--StopVisiblePreamble}

```

It defines the marker strings for the listed code sequences. In some cases you have to use multiple times the same value for different optional arguments, e.g.

```
ext=lua, compiler=lua, docType=lua, ...
```

5 Optional arguments

The default setting is always shown in brackets.

5.1 Programs and runs

The progpah should only in some rare cases needed. In general all used compilers will be found by the system. A given progpah must be end with a slash, e.g. ./bin/

```

\define@key{hv}{progpah}[]{\def\hv@extern@progpah{#1}}
\define@key{hv}{compiler}[pdflatex]{\def\hv@extern@compiler{#1}}
\define@key{hv}{runsequence}[]{\def\hv@extern@runsequence{#1}}
\define@key{hv}{runs}[1]{\setcounter{hv@extern@runs}{#1}}

```

Instead of the optional arguments compiler, biber, and xindex one can define an individual command sequence by using the optional argument runsequence. It must be comma separated list:

```
runsequence={lualatex,biber,xindex -l de -c AU,lualatex,lualatex}
```

voss-5.tex

```

\usepackage[ngerman]{babel}
\usepackage{libertinus,hvindex}
\usepackage{makeidx}\makeindex
\usepackage{biblatex}\addbibresource{biblatex-examples.bib}

```

```

Sort with xindex \verb|-l DE --config AU|
\blindtext
\Index{Österreich} \Index{Öresund}
\Index{Ostern} \Index{Ober} \Index{Oberin}
\Index{Österreich} \Index{Öresund}
\Index{Ödem} \Index{Oligarch} \Index{Oder}
\Index{Goldmann}
\printindex
\nocite{*}\printbibliography
\blindtext
\blinddocument

```


[illegible]

The following Java-program creates the Mandelbrot set as png image. The valid setting for the environment `externalDocument` is:

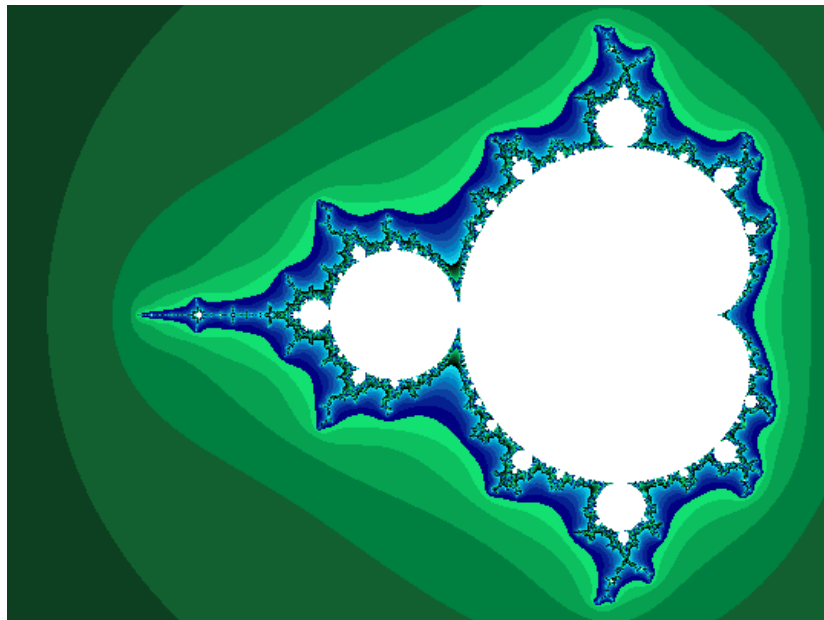
```
compiler=java,ext=java,docType=java,
```

```
public static int iterZahl(final double cx, final double cy, int maxIt,
    final double radius) {
    // bestimmt Anzahl der Iterationen
    int zaehler = 0;
    double zx = 0.0, zy = 0.0, tmp;
    do {
        tmp = zx*zx - zy*zy + cx;
        zy = 2*zx*zy + cy; zx = tmp;
        zaehler++;
    } while (zx*zx + zy*zy <= radius && zaehler < maxIt);
    return zaehler;
}
```

```
double xa = -2.5, xe = 0.7, ya = -1.2, ye = 1.2; // Ratio 20:15
double dx = (xe-xa)/(imageBreite-1), dy = (ye-ya)/(imageHoehe-1);
double cx, cy; int R, G, B;
double radius = 10.0; int maxIt = 1024;
cx = xa;
for (int sp = 0; sp < imageBreite; sp++) {
    cy = ye; // von oben nach unten
    for (int ze = 0; ze < imageHoehe; ze++) {
        int zIter = iterZahl(cx, cy, maxIt, radius);
        if (zIter == maxIt) {
            g.setColor(Color.WHITE);
            g.drawLine(sp, ze, sp, ze);
        } else {
            R = zIter % 4 * 6 ; G = zIter % 8 * 32; B = zIter % 16 * 16;
            g.setColor(new Color(R,G,B));
            g.drawLine(sp, ze, sp, ze);
        }
        cy = cy - dy;
    } // for ze
}
```

java-6.java

```
cx = cx + dx;
} // for sp
```



5.2 Grafik options

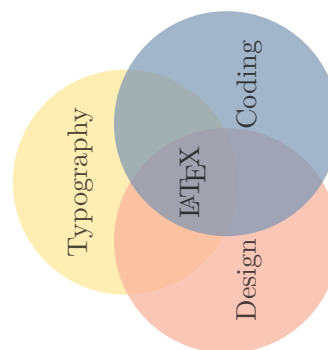
```
\define@key{hv}{grfOptions}[]{\def\hv@extern@grfOptions{#1}}
```

The option is passed to `\includegraphics`, e.g. `angle=90,width=\linewidth` for the following example.

voss-7.tex

```
\usepackage{tikz}
\usepackage[hks,pantone,xcolor]{xspotcolor}

\SetPageColorSpace{HKS}
\definecolor{HYellow}{spotcolor}{HKS05N,0.5}
\definecolor{HRed}{spotcolor}{HKS14N,0.5}
\definecolor{HBlue}{spotcolor}{HKS38N,0.5}
\begin{tikzpicture}[scale=0.7,fill opacity
=0.7]
\fill[HYellow] (90:1.2) circle (2);
\fill[HRed] (210:1.2) circle (2);
\fill[HBlue] (330:1.2) circle (2);
\node at (90:2) {Typography};
\node at (210:2) {Design};
\node at (330:2) {Coding};
\node {\LaTeX};
\end{tikzpicture}
```



5.3 Listings options

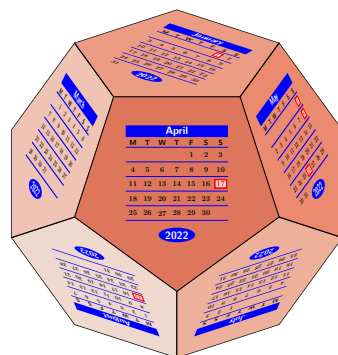
```
\define@key{hv}{\lstOptions}[]{\def\hv@extern@lstOptions{#1}}
```

The option is passed either to `\lstinputlisting`, or, if `usefancyvrb` is active, to `\VerbatimInput`. The following example uses

```
lstOptions={basicstyle=\sfamily\itshape\scriptsize},
```

```
\usepackage{pst-calendar}
```

```
\psscalebox{0.3}{%
  \psCalDodecaeder[Year=2022,style=april]%
}
```



voss-8.tex

5.4 Background color

There are different colors for the preamble and body listing: the background and frame color.

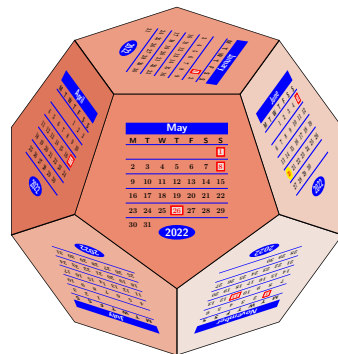
```
\define@key{hv}{BGpreamble}[black!2]{\def\hv@extern@BGpreamble{#1}}
\define@key{hv}{BGbody}[black!8]{\def\hv@extern@BGbody{#1}}
\define@key{hv}{BOpreamble}[black!2]{\def\hv@extern@BOpreamble{#1}}
\define@key{hv}{BObody}[black!8]{\def\hv@extern@BObody{#1}}
```

The options are passed to `tcolorbox` and preset to `black"!12` and `black"!8`. The color of the frame is set to the same values, hence not seen. The following example uses

```
BGpreamble=red!10, BOpreamble=red,
BGbody=blue!8, BObody=blue,
```

```
\usepackage{pst-calendar}
```

```
\psscalebox{0.3}{%
  \psCalDodecaeder[Year=2022,style=may]%
}
```



voss-9.tex

5.5 Type of the source code

The current version of hvextern supports code written as METAPOST, plain T_EX, L^AT_EX, ConT_EXt, and Python. Every type has its own keywords for the linerange which should be printed for the preamble and the body. For example the latex config is:

```
\hv@extern@exampleType{latex}%           for _all_LaTeX engines
  {\string\begin\string{document\string}}%
  {\string\end\string{document\string}}%
  {\perCent StartVisiblePreamble}%
  {\perCent StopVisiblePreamble}%

% only for the sequence latex->dvips->ps2pdf
\def\hv@extern@runLATEX#1#2#3#4{% path compiler file extension
  \ifhv@extern@verbose \typeout{>>> running #1#2 #3#4}\fi
  \ShellEscape{#1#2\space #3#4}%
  \ifhv@extern@verbose \typeout{>>> running #1dvips #3}\fi
  \ShellEscape{#1dvips\space #3.dvi}%
  \ifhv@extern@verbose \typeout{>>> running ps2pdf #3.ps}\fi
  \ShellEscape{#1ps2pdf\space -dAutoRotatePages=/None\space -dALLOWPSTRANSPARENCY\space #3.ps}%
}
```

If a source needs more than running the defined compiler, it must be defined by a macro

```
\def\hv@extern@run<NAME>#1#2#3#4{% path compiler file extension
...}
```

The type of the source code can be different to the compiler, e.g. source latex, but compiler lualatex.

5.6 Output more than one page

The pages which should be printed can be defined by

```
\define@key{hv}{pages}[1]{\def\hv@extern@pages{#1}}
\define@key{hv}{pagesep}[1em]{\hv@extern@pagesep=#1}
\define@boolkey{hv}[hv@extern@]{frame}[true]{}%
```

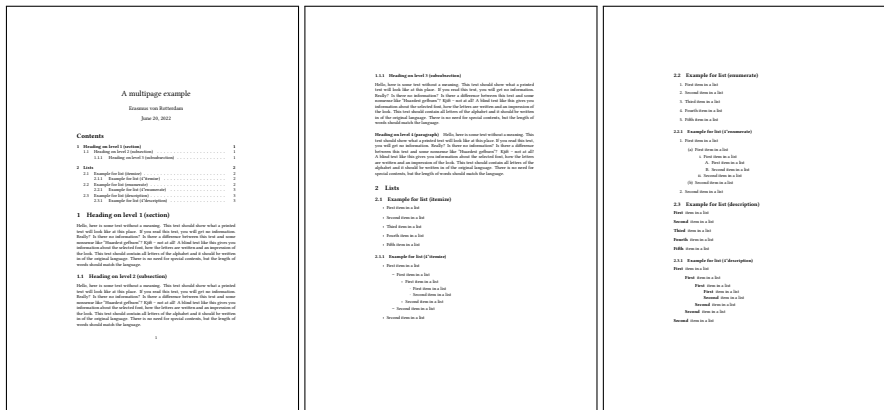
With frame the pages can be framed (internally by \fbox). It is leaved to the user to choose the correct image width for the pages. The separation between the pages is defined by the length pagesep. The following example uses:

```
pages={1,2,3},
pagesep=2pt,
grfOptions={width=0.3\linewidth},
compiler=lualatex, runs=2, % for the TOC
frame,
```

voss-10.tex

```
\usepackage[american]{babel}
\usepackage{libertinus}
\usepackage{blindtext}
```

```
\title{A multipage example}
\author{Erasmus von Rotterdam}
\maketitle
\tableofcontents
\blinddocument
```



5.7 Output as floating object with caption and label

By default the images are not inserted as a float. This can be changed by the keyword `float`, a caption and a label are optional. The float type is always `figure`.

```
\define@boolkey{hv}[hv@extern@]{float}[true]{}
\define@key{hv}{floatsetting}[]{\def\hv@extern@floatsetting{#1}}
\define@key{hv}{caption}[]{\def\hv@extern@caption{#1}}
\define@key{hv}{label}[]{\def\hv@extern@label{#1}}
```

The image Figure 1 shows an example for a floating object, which uses the `floatsetting !htb`, which is the default. Using a caption and a label are optional.

```
\usepackage{pst-coxeterp}
```

```
\begin{pspicture}(-1,-1)(1,1)\Simplex[dimension=2]\end{pspicture}
\begin{pspicture}(-1,-1)(1,1)\Simplex[dimension=3]\end{pspicture}
\begin{pspicture}(-1,-1)(1,1)\Simplex[dimension=5]\end{pspicture}
\begin{pspicture}(-1,-1)(1,1)\Simplex[dimension=7]\end{pspicture}
```

voss-11.tex

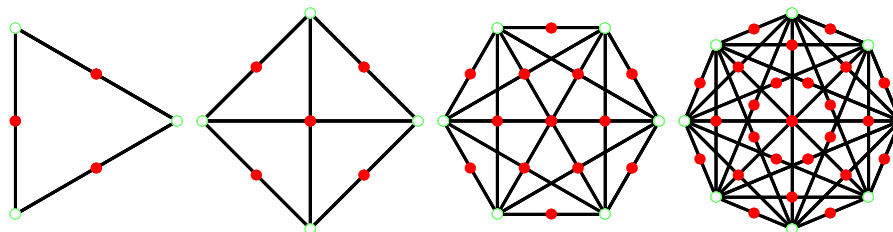


Abbildung 1: An example for Coxeter images

5.8 Cropping the PDF

Instead of using the documentclass standalone, which already crops the created PDF, one can use the optional argument crop.

```
\define@boolkey{hv}[hv@extern@]{crop}[true]{}
\define@key{hv}{cropmargin}[2]{\def\hv@extern@cropmargin{#1}}% length in pt
```

It is also possible to crop a document with more than one page. In this case the beginning and end of the pages should be on the same height. Otherwise the pages will have different heights after cropping (see next image). The following example was created with

```
pages={1,2,3},
pagesep=2pt,
grfOptions={width=0.3\linewidth},
compiler=lualatex, runs=2, % for the TOC
frame,
crop, cropmargin=5,% 5pt margin
```

voss-12.tex

```
\usepackage[american]{babel}
\usepackage{libertinus}
\usepackage{blindtext}
\pagestyle{headings}

\title{A multipage example}
\author{Erasmus von Rotterdam}
\maketitle
\tableofcontents
\Blinddocument
```

	1	HEADING ON LEVEL 1 (SECTION)	2	1	HEADING ON LEVEL 1 (SECTION)	3
<p>A multipage example</p> <p>Erasmus von Rotterdam</p> <p>June 30, 2022</p>						
Contents						
1	Heading on level 1 (section)	1				
1.1	Heading on level 2 (subsection)	2				
1.1.1	Heading on level 3 (subsubsection)	3				
2	Lists	5				
2.1	Example for list (itemize)	5				
2.1.1	Example for list (itemize)	6				
2.2	Example for list (enumerate)	6				
2.2.1	Example for list (enumerate)	9				
2.3	Example for list (description)	10				
2.3.1	Example for list (description)	11				
<p>1 Heading on level 1 (section)</p> <p>Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Hundred gefahren"? Kjilt - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special contents, but the length of words should match the language.</p> <p>This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Hundred gefahren"? Kjilt - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special contents, but the length of words should match the language.</p> <p>And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Hundred gefahren"? Kjilt - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special contents, but the length of words should match the language.</p> <p>After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Hundred gefahren"? Kjilt - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special contents, but the length of words should match the language.</p> <p>After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Hundred gefahren"? Kjilt - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special contents, but the length of words should match the language.</p>						
<p>1.1 Heading on level 2 (subsection)</p> <p>This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Hundred gefahren"? Kjilt - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special contents, but the length of words should match the language.</p> <p>And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Hundred gefahren"? Kjilt - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special contents, but the length of words should match the language.</p> <p>After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Hundred gefahren"? Kjilt - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special contents, but the length of words should match the language.</p> <p>After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Hundred gefahren"? Kjilt - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special contents, but the length of words should match the language.</p>						
<p>1.1.1 Heading on level 3 (subsubsection)</p> <p>And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Hundred gefahren"? Kjilt - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special contents, but the length of words should match the language.</p> <p>After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Hundred gefahren"? Kjilt - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special contents, but the length of words should match the language.</p> <p>After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Hundred gefahren"? Kjilt - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special contents, but the length of words should match the language.</p>						

5.9 Code and output side by side

By default the code and the output is on top of each other. With setting the width of a minipage with mpwidth greater than 0pt the output will be side by side.

```
\define@key{hv}{mpwidth}[0pt]{\setlength{hv@extern@mpwidth}{#1}}
\define@key{hv}{mpvalign}[0pt]{\def{hv@extern@mpvalign}{#1}}
```

`mpwidth` is the width of the code. The rest of the line, minus 1em for the space between the minipages, will be the possible width for the output and will be calculated automatically. The two minipages are aligned by defaults to its top. This can be changed by setting `mpvalign` to `c` or `b`.

5.10 Horizontal alignment of the output

`align=\centering, % default`

```
\rule{0.5\linewidth}{5mm}
```



`align=\raggedright,`

```
\rule{0.5\linewidth}{5mm}
```



`align=\raggedleft,`

```
\rule{0.5\linewidth}{5mm}
```



`align=\centering, mpwidth=0.5\linewidth, % default for side by side`

```
\rule{0.25\linewidth}{5mm}
```



`align=\raggedright, mpwidth=0.5\linewidth,`

```
\rule{0.25\linewidth}{5mm}
```



`align=\raggedleft, mpwidth=0.5\linewidth,`

```
\rule{0.25\linewidth}{5mm}
```



5.11 Inline images

By default code and image are own paragraphs. With the optional argument `inline` the created image can be part of the current line. This may make sense, if you need characters which are not part of your current font.

```
\define@boolkey{hv}{hv@extern@}{inline}[true]{%
  \hv@extern@codefalse
  \hv@extern@showFilenamefalse}
```

With the setting `inline=true` the optional keyword `code` and `showFilename` is automatically set to false. The next Chinese characters % 美好的一天 are inserted as inline image without showing the code. The complete code looks like:

```
With \Lkeyset{inline} the optional argument \Lkeyword{code} is
automatically set to false. The next Chinese characters
\begin{externalDocument}[
  compiler=xelatex, inline, runs=2, grfOptions={height=8pt},
  crop, cropmargin=0, cleanup, force=false, docType=latex]{voss}
\documentclass{ctexart}
\pagestyle{empty}
\begin{document}
  美好的一天
\end{document}
\end{externalDocument}
```

are inserted as inline image without showing the code. The complete code looks like:

5.12 Input text instead of an image

By default the created pdf which can be, of course, only text, will be inserted by `\includegraphics`. If you have only text as output and don't want to create a pdf you can insert this kind of output as verbatim text by setting `includegraphic=false`.

```
\define@boolkey{hv}{hv@extern@}{includegraphic}[true]{}
```

The textfile must have the same main filename with the extension `.txt`. As already mentioned, in every programming language you can get the current used filename from within the code itself. The following Perl example which calculates the Kaprekar constants uses

```
my $filename = $0;           # the current filename
$filename =~ s/\.pl//;       # without extension .pl
$filename = "${filename}.txt"; # for the output
```

Only for some completeness: a Kaprekar constant is a number A with $\max(A) - \min(A) = A$. \max and \min are the sorted digits of the number A : $495 = 954 - 459$.


```
my $zahl = 1;
my $anfang = 1;
my $ende = 9;
```

```
print $fh "Finding Kaprekarconstants ...\n";
while ($zahl < 8) {
  print $fh "${zahl}-stellig: ";
  foreach ($anfang...$ende) { # for every
    row $_
    @Zeichen = split(//,$_);
    $Min = join("",sort(@Zeichen));
    $Max = reverse($Min);

    $Dif=$Max-$Min;
    if($_ eq $Dif) {
      $found = 1;
      print $fh $_, " ";
    }
  }
  if (!$found) { print $fh "---\n"; }
  else { print $fh "\n"; }

  $found = false;
  $zahl = $zahl+1;
  $anfang = $anfang*10;
  $ende = $ende*10;
}
}
```

Finding Kaprekarconstants

```
...
1-stellig: ---
2-stellig:
3-stellig: 495,
4-stellig: 6174,
5-stellig:
6-stellig: 549945, 631764,
7-stellig:
```

voss-20.pl

Another example with running Lua to calculate and print the Pascal's triangle. The internal filename is available with

```
local filename = arg[0]
local shortFN = str:match("(.)%.+") -- delete extension
outFile = io.open(shortFN..".txt","w+") -- open external file
```

```
function nextrow(t)
  local ret = {}
  t[0], t[#t+1] = 0, 0
  for i = 1, #t do ret[i] = t[i-1] + t[i] end
  return ret
end

function triangle(n)
  t = {1}
  for i = 1, n do
    m = (n - i)
    for j = 1, m do outFile:write(" ") end
    for k = 1, i do outFile:write(string.format("%8s", t[k])) end
    outFile:write("\n")
    t = nextrow(t)
  end
end
```

voss-21.lua

```
triangle(10)
```

```

      1
    1 1
  1 2 1
1 3 3 1
 4 6 4
 1 5 10 10 5 1
 1 6 15 20 15 6 1
 1 7 21 35 35 21 7 1
 1 8 28 56 70 56 28 8 1
 1 9 36 84 126 126 84 36 9 1

```

5.13 Running additional external programs

For a L^AT_EX additional programs for bibliography, index, a.s.o. maybe needed.

```

\define@boolkey{hv}[hv@extern@]{biber}[true]{}
\define@boolkey{hv}[hv@extern@]{xindex}[true]{}
\define@key{hv}{xindexOptions}[]{\def\hv@extern@xindexOptions{#1}}
\define@key{hv}{runsequence}[]{\def\hv@extern@runsequence{#1}}

```

The biber run needs no additional options, but for xindex it maybe useful. The following examples uses

```

\begin{externalDocument}[
  compiler=lualatex, runs=2, pages=2, crop,
  xindex, xindexOptions={-l DE --config AU},
  mpwidth=0.6\linewidth, usefancyvrb=false,
  docType=latex,
  ...
]{voss}

```

voss-22.tex

```

\usepackage{makeidx}\makeindex
\usepackage{hvindex}

```

```

Sort with xindex \verb|-l DE --config AU|
\Index{Österreich} \Index{Öresund}
\Index{Ostern} \Index{Ober} \Index{Oberin}
\Index{Österreich} \Index{Öresund}
\Index{Ödem} \Index{Oligarch} \Index{Oder}
\Index{Ostern} \Index{Ober} \Index{Oberin}
\Index{Obstler} \Index{Öl} \Index{ölen}
\Index{Oder|seealso{Fluss}} \Index{Göbel}
\Index{oder} \index{Fluss!Oder}
\Index{Goethe} \Index{Göthe} \Index{Götz}
\Index{Goldmann}
\printindex

```

Index

F	Obstler, 1
Fluss	oder, 1
- Oder, 1	Oder, 1, <i>siehe auch</i> Fluss
	Oligarch, 1
	Ostern, 1
G	
Goethe, 1	Ö
Goldmann, 1	Ödem, 1
Göbel, 1	Öl, 1
Göthe, 1	Öresund, 1
Götz, 1	Österreich, 1
O	
Ober, 1	ö
Oberin, 1	ölen, 1

Instad of using the options compiler, biber, and xindex one can also use only the optional argument runsequence to define an individuell sequence of commands, e.g.:

```
runsequence={lualatex,biber,{xindex -l de -c AU},lualatex,lualatex}
```


voss-24.tex

```

beforekip=-3.25ex plus -1ex minus -.2ex,
afterskip=1.5ex plus .2ex,
font=\normalsize,
indent=0pt,
counterwithin=subsubsection
]{subsubsubsection}
\RedeclareSectionCommand[
  level=5,
  toplevel=5,
  tocindent=13em,
  tocnumwidth=5.9em,
  counterwithin=subsubsubsection
]{paragraph}
\RedeclareSectionCommand[
  level=6,
  toplevel=6,
  tocindent=15em,
  tocnumwidth=6.8em
]{subparagraph}
\setcounter{secnumdepth}{\subsubsubsectionnumdepth}
\setcounter{tocdepth}{\subsubsubsectionnumdepth}

\tableofcontents
\chapter{Einführung}
\section{Ein Abschnitt}
\subsection{Ein Unterabschnitt}
\subsubsection{Ein Unter-Unterabschnitt}
\subsubsubsection{Ein Unter-Unter-Unterabschnitt}
\paragraph{Der normale Paragraph}
\blindtext
\subparagraph{Der normale Unterparagraph}
\blindtext
\blinddocument

```

Inhaltsverzeichnis

1	Einführung	3
1.1	Ein Abschnitt	3
1.1.1	Ein Unterabschnitt	3
1.1.1.1	Ein Unter-Unterabschnitt	3
1.1.1.1.1	Ein Unter-Unter-Unterabschnitt	3
2	Überschrift auf Ebene 0 (chapter)	5
2.1	Überschrift auf Ebene 1 (section)	5
2.1.1	Überschrift auf Ebene 2 (subsection)	5
2.1.1.1	Überschrift auf Ebene 3 (subsubsection)	6
2.2	Listen	6
2.2.1	Beispiel einer Liste (litemize)	6
2.2.1.1	Beispiel einer Liste (litemize)	7
2.2.2	Beispiel einer Liste (litemize)	7
2.2.2.1	Beispiel einer Liste (litemize)	7
2.2.3	Beispiel einer Liste (description)	8
2.2.3.1	Beispiel einer Liste (description)	8

1

Kapitel 1

Einführung

1.1 Ein Abschnitt

1.1.1 Ein Unterabschnitt

1.1.1.1 Ein Unter-Unterabschnitt

1.1.1.1.1 Ein Unter-Unter-Unterabschnitt

Der normale Paragraph

Der normale Unterparagraph

3

```
\documentclass[chapterprefix=on,parskip=half-,DIV=12,fontsize=12pt]{scrbook}
\DeclareNewSectionCommand[
  style=section,
  level=4,
  beforeskip=-3.25ex plus -1ex minus -.2ex,
  afterskip=1.5ex plus .2ex,
  font=\normalsize,
  indent=0pt,
  counterwithin=subsubsection
]{subsubsubsection}
\RedeclareSectionCommand[
  level=5,
  toplevel=5,
  tocindent=13em,
  tocnumwidth=5.9em,
  counterwithin=subsubsubsection
]{paragraph}
\RedeclareSectionCommand[
  level=6,
  toplevel=6,
  tocindent=15em,
  tocnumwidth=6.8em
]{subparagraph}
\setcounter{secnumdepth}{\subsubsubsectionnumdepth}
\setcounter{tocdepth}{\subsubsubsectionnumdepth}
```

```
\tableofcontents
\chapter{Einführung}
\section{Ein Abschnitt}
\subsection{Ein Unterabschnitt}
\subsubsection{Ein Unter-Unterabschnitt}
\subsubsubsection{Ein Unter-Unter-Unterabschnitt}
\paragraph{Der normale Paragraph}
\blindtext
\subparagraph{Der normale Unterparagraph}
\blindtext
\blindeddocument
```

voss-25.tex

[illegible]

5.15 Vertical space

```
\define@key{hv}{aboveskip}[\medskipamount]{%
    \setlength{hv@extern@aboveskip}{#1}}
\define@key{hv}{belowpreambleskip}[\smallskipamount]{%
    \setlength{hv@extern@belowpreambleskip}{#1}}
\define@key{hv}{belowbodyskip}[\smallskipamount]{%
    \setlength{hv@extern@belowbodyskip}{#1}}
\define@key{hv}{belowskip}[\medskipamount]{%
    \setlength{hv@extern@belowskip}{#1}}
```

aboveskip Vertical space *before* the environment `externalDocument` or the command `\runExtCmd` (default `\medskipamount`)

belowpreambleskip Vertical space between preamble and body (default `\smallskipamount`)

belowbodyskip Vertical space between body and output (default `\smallskipamount`)

belowskip Vertical space *after* the environment `externalDocument` or the command `\runExtCmd` (default `\medskipamount`)

5.16 No output

By default there is an image or text as output of the external run. In a case, where you are only interested in the code, which should be formatted in the same style as other examples, you can set `showoutput` to `false`.

```
\documentclass[chapterprefix=on,parskip=half-,DIV=12,fontsize=12pt]{scrbook}
\DeclareNewSectionCommand[
  style=section,
  level=4,
  beforekip=-3.25ex plus -1ex minus -.2ex,
  afterskip=1.5ex plus .2ex,
  font=\normalsize,
  indent=0pt,
  counterwithin=subsubsection
]{subsubsubsection}
```

```
\tableofcontents
\chapter{Einführung}
\section{Ein Abschnitt}
\subsection{Ein Unterabschnitt}
\subsubsection{Ein Unter-Unterabschnitt}
\subsubsubsection{Ein Unter-Unter-Unterabschnitt}
\blindtext
```

voss-26.tex

6 Defining new marker

Suppose you do not want for a L^AT_EX document the complete body part between `\begin` and `\end` printed. In this case you can define own markers, e.g.:

```
\defMarkerType{ltx}
  {\perCent StartVisibleBody}
  {\perCent StopVisibleBody}
  {\perCent StartVisiblePreamble}
  {\perCent StopVisiblePreamble}
```

With this definition and the setting `docType=ltx` the last example looks like:

```
\DeclareNewSectionCommand[
  style=section,
  level=4,
  before=skip=-3.25ex plus -1ex minus -.2ex,
  after=skip=1.5ex plus .2ex,
  font=\normalsize,
  indent=0pt,
  counterwithin=subsubsection
]{subsubsubsection}

\subsubsubsection{Ein Unter-Unter-Unterabschnitt}
```

voss-27.tex

Inhaltsverzeichnis

1	Einführung	2
1.1	Ein Abschnitt	2
1.1.1	Ein Unterabschnitt	2
1.1.1.1	Ein Unter-Unterabschnitt	2
1.1.1.1.1	Ein Unter-Unter-Unterabschnitt	2

7 Supported engines

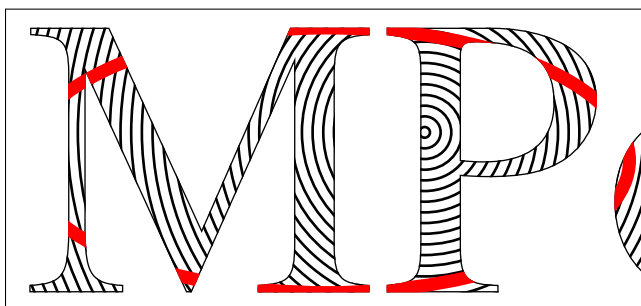
7.1 METAPOST example

Needs the run sequence setting to get a pdf from the created dvi output. It is already internally defined.

```
defaultfont:="ptmr8r";
warningcheck:=0;

draw fullcircle shifted (0.5,0.6) xscaled 8cm yscaled 3.5cm
  withpen pencircle scaled 5bp withcolor red;
special( " /Times-Roman findfont 150 scalefont setfont " &
  " 0 10 moveto (MPost) false charpath clip stroke gsave 150 70 translate "
  &
  " 2 4 600 {dup 0 moveto 0 exch 0 exch 0 360 arc stroke} for grestore ");
```

voss-28.mp



7.2 plain \TeX example

Needs the run sequence setting to get a pdf from the created dvi output. It is already internally defined.

voss-29.tex

```
\footline={\footsc the electronic journal of combinatorics
  {\footbf 16} (2009), \#R00\hfil\footrm\folio}
```

```
\font\bigrm=cmr12 at 14pt
```

```
\centerline{\bigrm An elementary proof of the reconstruction conjecture}
```

```
\bigskip\bigskip
```

```
\centerline{D. Remifa\footnote*{Thanks to the editors of this journal!}}
```

```
\smallskip
```

```
\centerline{Department of Inconsequential Studies}
```

```
\centerline{Solatido College, North Kentucky, USA}
```

```
\centerline{\tt remifa@dis.solatido.edu}
```

```
\bigskip
```

```
\centerline{\footrm
```

```
Submitted: Jan 1, 2009; Accepted: Jan 2, 2009; Published: Jan 3, 2009}
```

```
\centerline{\footrm Mathematics Subject Classifications: 05C88, 05C89}
```

```
\bigskip\bigskip
```

```
\centerline{\bf Abstract}
```

```
\smallskip
```

```
{\narrower\noindent
```

The reconstruction conjecture states that the multiset of unlabeled vertex-deleted subgraphs of a graph determines the graph, provided it has at least 3 vertices. A version of the problem was first stated by Stanisław Ulam. In this paper, we show that the conjecture can be proved by elementary methods. It is only necessary to integrate the Lenkle potential of the Broglington manifold over the quantum supervacillatory measure in order to reduce the set of possible counterexamples to a small number (less than a trillion). A simple computer program that implements Pipletti's classification theorem for torsion-free Aramaic groups with symplectic socles can then finish the remaining cases.}

```
\bigskip
```

```
\beginsection 1. Introduction.
```

This is the start of the introduction.



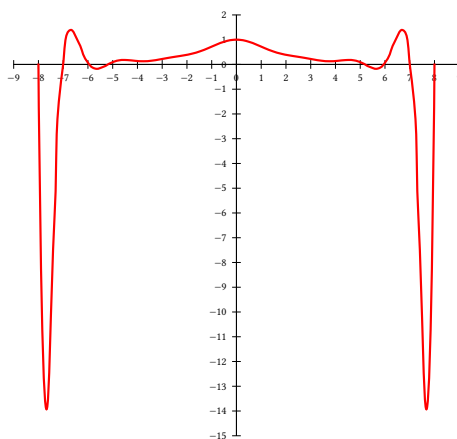
7.3 Lua \LaTeX example

With Lua \LaTeX and using PostScript code the intermediate GhostScript run is not needed. The pdf is directly created.

```
\usepackage{fontenc}\usepackage{libertinus}
\usepackage{pst-all}

\psset{unit=0.8cm}
\begin{pspicture}(-9,-15)(9,2)
\psaxes(0,0)(-9,-15)(9,2)
\psplot[algebraic,plotstyle=curve,curvature=1 1 0,
linewidth=2pt,linewidth=red]{-8}{8}{
1 - 3876218985722260225*x^2/10892114744073986176
+ 14975974793271450625*x^4/174273835905183778816
- 317095420958296875*x^6/26811359370028273664
+ 194412970920703125*x^8/214490874960226189312
- 2090988251953125*x^10/53622718740056547328
+ 99480224609375*x^12/107245437480113094656
- 7879638671875*x^14/697095343620735115264
+ 152587890625*x^16/2788381374482940461056}
\end{pspicture}
```

voss-30.tex



7.4 ConT_EXt example

voss-31.tex

```

\definehead
  [myhead]
  [section]
\setuphead
  [myhead]
  [numberstyle=bold,
   textstyle =bold,
   before    =\hairline\blank,
   after     =\nowhitespace\hairline]

\startstandardmakeup
\midaligned{From Hasselt to America}
\midaligned{by}
\midaligned{J. Jonker and C. van Marle}
\stopstandardmakeup
\placecombinedlist[content]
\chapter{Introduction}
\input knuth \input knuth
\chapter[rensselaer]{The Rensselaer family}
\input knuth
\section{The first born}
\input knuth
\section{The early years}
... in those days Hasselt was ...
\input knuth
\section{Living and workin in America}
\input knuth
\chapter[lansing]{The Lansing family}
... the Lansing family was also ...
\input knuth
\chapter[cuyler]{The Cuyler family}
... much later Tydeman Cuyler ...
\input knuth
\myhead[headlines]{And the end of all}
foo

```

<p>Press Hardy to Austin</p> <p>J. Jucker and C. van Hatt</p>	<p>1</p> <p>1 Introduction</p> <p>2 The Rensselaer family</p> <p>2.1 The first horns</p> <p>2.2 The early years</p> <p>2.3 Living and working in America</p>	<p>2</p> <p>1 Introduction</p> <p>Then I came to the conclusion that the designer of a new system must not only be the implementer and first long-walk user, the designer should also write the first user manual.</p> <p>The separation of any of these four components would have hurt TQS significantly. If I had not participated fully in all these activities, literally hundreds of improvements would never have been made, because I would never have thought of them or perceived why they were important.</p> <p>But a system cannot be successful if it is too strongly influenced by a single person. Once the initial design is complete and fairly robust, the real test begins as people with many different viewpoints undertake their own experiments.</p>
<p>2</p> <p>2 The Rensselaer family</p> <p>Then I came to the conclusion that the designer of a new system must not only be the implementer and first long-walk user, the designer should also write the first user manual.</p> <p>The separation of any of these four components would have hurt TQS significantly. If I had not participated fully in all these activities, literally hundreds of improvements would never have been made, because I would never have thought of them or perceived why they were important.</p> <p>But a system cannot be successful if it is too strongly influenced by a single person. Once the initial design is complete and fairly robust, the real test begins as people with many different viewpoints undertake their own experiments.</p>	<p>4</p> <p>2.3 Living and working in America</p> <p>Then I came to the conclusion that the designer of a new system must not only be the implementer and first long-walk user, the designer should also write the first user manual.</p> <p>The separation of any of these four components would have hurt TQS significantly. If I had not participated fully in all these activities, literally hundreds of improvements would never have been made, because I would never have thought of them or perceived why they were important.</p> <p>But a system cannot be successful if it is too strongly influenced by a single person. Once the initial design is complete and fairly robust, the real test begins as people with many different viewpoints undertake their own experiments.</p>	<p>5</p> <p>3 The Lausung family</p> <p>... the Lausung family was also ... Then I came to the conclusion that the designer of a new system must not only be the implementer and first long-walk user, the designer should also write the first user manual.</p> <p>The separation of any of these four components would have hurt TQS significantly. If I had not participated fully in all these activities, literally hundreds of improvements would never have been made, because I would never have thought of them or perceived why they were important.</p> <p>But a system cannot be successful if it is too strongly influenced by a single person. Once the initial design is complete and fairly robust, the real test begins as people with many different viewpoints undertake their own experiments.</p>

8 Running external commands

Integrating the current directory of this document we can use the macro `\runExtCmd` with the optional argument `redirect`

```
\runExtCmd[redirect]{ls -la}{voss}
```

to get the directory listed:

```
total 4776
drwxr-xr-x  19 voss  staff    608 20 Jun 18:34 .
drwxr-xr-x 201 voss  staff   6432 10 Jun 08:01 ..
drwxr-xr-x   3 voss  staff    96 24 Apr 19:50 .ctan
drwxr-xr-x   4 voss  staff   128 27 Apr 11:29 .test
-rw-r--r--   1 voss  staff  1694 20 Jun 18:22 Changes
drwxr-xr-x 101 voss  staff  3232 20 Jun 18:34 Exa
-rwxrwxrwx   1 voss  staff  1170 27 Mai 08:16 Makefile
-rw-rw-rw-   1 voss  staff   713 27 Mai 08:56 README
-rwxrwxrwx   1 voss  staff  3998 27 Mai 08:59 hvdoctools.sty
-rw-r--r--   1 voss  staff 16384 20 Jun 18:34 hvextern.aux
-rw-r--r--   1 voss  staff  8192 20 Jun 18:34 hvextern.idx
-rw-r--r--   1 voss  staff   600 20 Jun 18:33 hvextern.ilg
-rw-r--r--   1 voss  staff  8674 20 Jun 18:33 hvextern.ind
-rw-r--r--   1 voss  staff 168491 20 Jun 18:34 hvextern.log
```

```

-rw-r--r--  1 voss  staff      364 16 Jun 20:46 hvextern.lua
-rw-r--r--@  1 voss  staff 1295945 20 Jun 18:34 hvextern.pdf
-rw-r--r--  1 voss  staff   28490 20 Jun 18:32 hvextern.sty
-rw-r--r--  1 voss  staff   51419 20 Jun 18:33 hvextern.tex
-rw-r--r--  1 voss  staff      0 20 Jun 18:34 hvextern.toc

```

9 Other options

force=false can speed up the compiling time for the document. If a created image/output already exists, there is no need to create it with the next run again and again. This option is not valid if the package option `checkCode` exists.

cleanup the auxiliary files of a L^AT_EX-run are deleted, preset to `aux, log`. It must be a comma separated list of the extensions of the main file, s.g. `cleanup={aux, log}`.

copyToExampleDir name of a directory for the examples, must first be created by the user himself

ExamplesDir move all examples into a directory

tclbox=false Can be used if there are some negative interactions between package `listings` and package `tcolorbox`.

framesep Value for `\fbox` if keyword `frame` is used.

mpsep Distance between code and output (default 1 em).

pagesep Distance between pages for multipage output (default 1 em).

verbose Print control messages into the terminal and logfile.

eps create an eps from the pdf (historical).

Index

Symbols

-shell-escape (package option), 3
./bin/ (value), 8
#, 4
%StartVisiblePreamble, 7
%StopVisiblePreamble, 7

A

aboveskip (keyword), 22

B

b (value), 15
\begin, 23
\begin{document}, 7
belowbodyskip (keyword), 22
belowpreambleskip (keyword), 22
belowskip (keyword), 22
biber (keyword), 8, 18
bibliography, 18
black!12 (color), 11
black!8 (color), 11

C

c (value), 15
checkCode (package option), 3, 28
cleanup (keyword), 28
code (keyword), 16
compiler (keyword), 7f, 18
context (value), 7
copyToExampleDir (keyword), 28
crop (keyword), 14
cropping, 14
current line, 16

D

docType (keyword), 7, 23
documentclass, 14

E

\end, 23
\end{document}, 7
eps (keyword), 28
ExamplesDir (keyword), 28
externalDocument (environment), 3f, 9, 22

F

false (value), 16, 28
fancyvrb (package), 21

\fbox, 12, 28
figure (environment), 13
float (keyword), 13
force (keyword), 3, 28
frame (keyword), 12
framesep (keyword), 28

H

hvextern (package), 6f, 12

I

includegraphic (keyword), 16
\includegraphics, 10, 16
index, 18
inline (keyword), 16
inline image, 16

J

Java, 3

K

Kaprekar, 16
Keyword

- ExamplesDir, 28
- aboveskip, 22
- belowbodyskip, 22
- belowpreambleskip, 22
- belowskip, 22
- biber, 8, 18
- cleanup, 28
- code, 16
- compiler, 7f, 18
- copyToExampleDir, 28
- crop, 14
- docType, 7, 23
- eps, 28
- float, 13
- force, 3, 28
- frame, 12
- framesep, 28
- includegraphic, 16
- inline, 16
- mpsep, 28
- mpvalign, 15
- mpwidth, 14f
- outerFN, 6
- pagesep, 12, 28
- progpas, 8

- redirect, 27
- runsequence, 8, 18
- shiftFN, 6
- showFilename, 6, 16
- showoutput, 22
- tclbox, 28
- usefancyvrb, 11, 21
- verbose, 28
- xindex, 8, 18

L

latex (value), 7, 12
 listings (package), 28
 \lstinputlisting, 11, 19
 ltx (value), 23
 lua (value), 7
 Lua, 3
 lua_latex (value), 7, 12
 luatex (value), 7

M

\medskipamount, 22
 minipages, 15
 mp (value), 7
 mpost (value), 7
 mpsep (keyword), 28
 mpvalign (keyword), 15
 mpwidth (keyword), 14f

N

\NumChar, 4, 7

O

outerFN (keyword), 6

P

pagesep (keyword), 12, 28
 Pascal's triangle, 17
 pdf_latex (value), 7
 perl (value), 7
 Perl, 3
 pl (value), 7
 png, 4
 progbath (keyword), 8
 py (value), 7
 Python, 3, 5
 python3 (value), 7

R

redirect (keyword), 27
 \runExtCmd, 22, 27

runsequence (keyword), 8, 18

S

shiftFN (keyword), 6
 showFilename (keyword), 6, 16
 showoutput (keyword), 22
 \smallskipamount, 22
 source, 3
 standalone (doc class), 14

T

tclbox (keyword), 28
 tcolorbox (package), 11, 28
 tex (value), 7
 .txt (file extension), 16

U

usefancyvrb (keyword), 11, 21

V**Value**

- false, 16, 28
- ltx, 23

\VerbatimInput, 11, 21
 verbose (keyword), 28

X

xelatex (value), 7
 xetex (value), 7
 xindex (keyword), 8, 18