

Wide-range METAPOST tutorial

Luís Nobre Gonçalves
(L. Nobre G.)

10. november 2012

Sammendrag

Sort of workshop about T_EX–things, focusing on the METAPOST language.

The Church of Free Software

Pope: Richard Stallman.

Evil

WYSIWYG

Evil

WYSIWYG

What you get is what you **don't** see

Freedom

You choose what you get

Freedom

You choose what you get
but
you only get it at the end of a long path.

① Introduction

A bit of history

What is METAPOST?

Main reason to use METAPOST

② For Almost Absolute Beginners

Availability

Mathematically oriented

Programmability/Scriptability

Forced Simplicity

Consistency

③ Workflow

Workflow example

A bit of history

Donald Knuth – *The Art of Computer Programming*

- Volume1, 1969, hot metal
- Volume2, 2nd ed., 1977, photographic
- T_EX, v1, 1978, $v \longrightarrow \pi$
- METAFONT, v1, 1979, $v \longrightarrow e - \text{affine}$
- AMS, 1983
- APS, AIP, OSA, AAS, Springer-Verlag
- METAPOST, 1994, John Hobby

A bit of history

Donald Knuth – *The Art of Computer Programming*

- Volume1, 1969, hot metal
- Volume2, 2nd ed., 1977, photographic
- T_EX, v1, 1978, $v \longrightarrow \pi$
- METAFONT, v1, 1979, $v \longrightarrow e - \text{affine}$
- AMS, 1983
- APS, AIP, OSA, AAS, Springer-Verlag
- METAPOST, 1994, John Hobby

METAPOST is the realm of graphic **parameterization**.

Main features of METAPOST

- It is easy to express geometry as METAPOST code
- METAPOST outputs postscript or SVG

outputformat := "svg";

- Very good control of 2D Bézier splines
- Special 2D operators
- Many operators work the same way in 1, 2, 3 or 4D
- Linear equations
- May include \LaTeX
- May be included in \LaTeX

Use METAPOST because

METAPOST helps the user to experiment different diagram layouts without changing specified **geometric relationships** among diagram elements.

For Almost Absolute Beginners

- Availability
- Mathematically oriented
- Programmability/Scriptability
- Forced simplicity
- Consistency

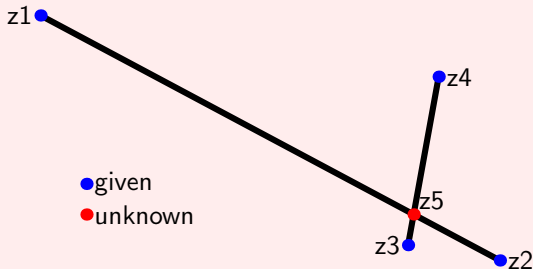
Availability

- Development site
- CTAN
 - /graphics/metapost/contrib/macros

Availability

- Development site
- CTAN
 - /graphics/metapost/contrib/macros
- T_EX Users Group
 - texlive
- Debian
 - texlive-metapost
 - texlive-publishers includes RevTeX
 - latex-beamer
 - tex4ht
- “The L^AT_EX Companions”

Linear equations



- Constraints may be expressed as linear equations

$$z5 = \text{whatever}[z1, z2];$$

$$z5 = \text{whatever}[z3, z4];$$

- No need to explicitly assign calculations to unknowns

pair doublets

- pair $(x,y) \longrightarrow (X,Y)$
origin $(0,0)$
- Coordinates
 $z[i] = (x[i], y[i]);$

color triplets

- color $(red, green, blue) \longrightarrow (X, Y, Z)$

black	$(0,0,0)$
red	$(1,0,0)$
green	$(0,1,0)$
blue	$(0,0,1)$
white	$(1,1,1)$

cmykcolor tetraplets

- cmykcolor
(*cyan, magenta, yellow, black*) \longrightarrow (X, Y, Z, W)
(quaternions, homogeneous coordinates, 4D splines, animation frames, straight line segments, etc.)

Special operators

- Pythagorean addition and subtraction

$$h = a ++ b +-+ c; (h = \sqrt{a^2 + b^2 - c^2})$$

Special operators

- Pythagorean addition and subtraction

$$h = a ++ b +-+ c; (h = \sqrt{a^2 + b^2 - c^2})$$

- Interval 0—1 ND operator s[,]

$$\begin{array}{ll} 0.5[(1, 1), (3, 5)] & \longrightarrow (2, 3) \\ -0.5[(1, 1, 1), (3, 5, 7)] & \longrightarrow (0, -1, -2) \\ 1.5[(1, 1, 1, 1), (3, 5, 7, 9)] & \longrightarrow (4, 7, 10, 13) \end{array}$$

Special operators

- Pythagorean addition and subtraction

$$h = a ++ b +-+ c; (h = \sqrt{a^2 + b^2 - c^2})$$

- Interval 0—1 ND operator $s[,]$

$$\begin{array}{ll} 0.5[(1,1), (3,5)] & \longrightarrow (2,3) \\ -0.5[(1,1,1), (3,5,7)] & \longrightarrow (0,-1,-2) \\ 1.5[(1,1,1,1), (3,5,7,9)] & \longrightarrow (4,7,10,13) \end{array}$$

- N-dimensional scalar arithmetic

$$\begin{array}{l} b = 2a; \\ z1 = 2*(1,2); \\ cc2 = 2*(1,2,3); \text{ grey} = 0.5\text{white}; \\ cc3 = 2*(1,2,3,4); \end{array}$$

Other features

- Throw-in-the-garbage unknowns
whatever

Other features

- Throw-in-the-garbage unknowns

whatever

- Special functions

dir() abs() angle() unitvector()
sind() cosd()

Other features

- Throw-in-the-garbage unknowns

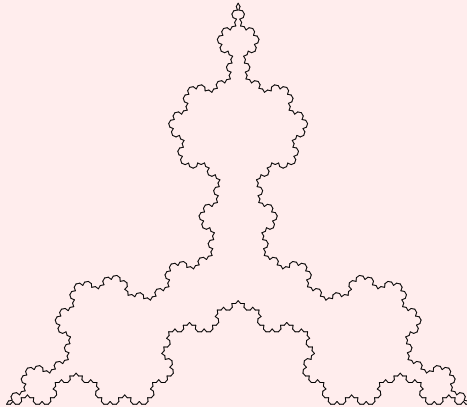
whatever

- Special functions

dir() abs() angle() unitvector()
sind() cosd()

- Affine transforms (linear)
- Totally parameterized 2D cubic Bézier splines

Programmability



ø æ å ì § ü ç © £ œ ß « » “ ” ... $3.14159e-2 = 3,1416 \cdot 10^{-2}$

Scriptability

- `scantokens "string of commands";`
- MPlib

Forced Simplicity

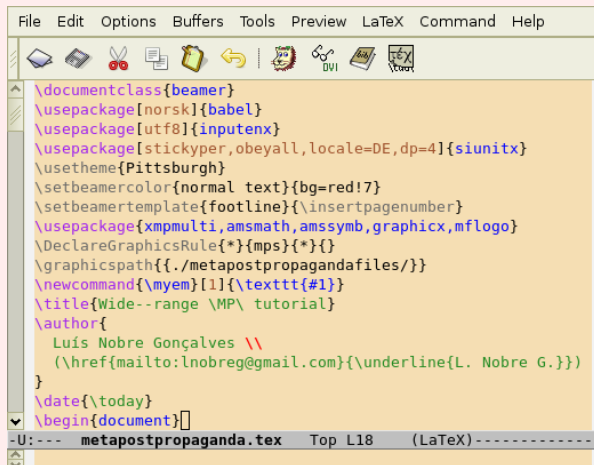
- $(8 \times) 4096$
- $1/65536$
- 2000 points in a path
- Memory
- Number of variables

Not for long. . .

Consistency

- emacs+auctex/texmacs
- latexmp
- emp/LuaT_EX
- graphicx
- beamer+xmpmulti
- GNUPLOT
- xfig
- dia/expressg
- gs+potrace+pstoedit+PstoeditMpostPreeditor
- siunitx

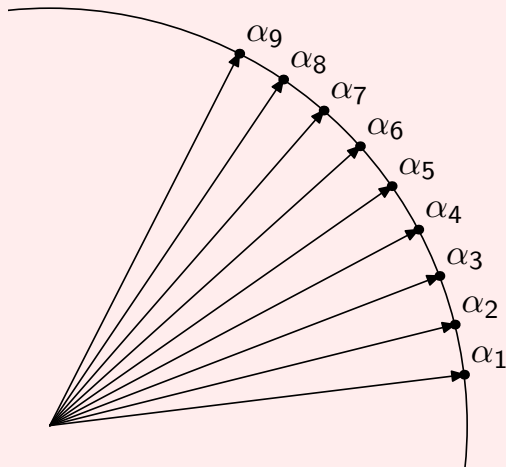
emacs+auctex



```
\documentclass{beamer}
\usepackage[norsk]{babel}
\usepackage[utf8]{inputenx}
\usepackage[stickyper,obeyall,locale=DE,dp=4]{siunitx}
\usetheme{Pittsburgh}
\setbeamercolor{normal text}{bg=red!7}
\setbeamertemplate{footline}{\insertpagenumber}
\usepackage{xmpmulti,amsmath,amssymb,graphicx,mflogo}
\DeclareGraphicsRule{*}{mps}{*}{}
\graphicspath{./metapostpropagandafiles/}
\newcommand{\myem}[1]{\texttt{#1}}
\title{Wide--range \MP\ tutorial}
\author{
  Luís Nobre Gonçalves \\\
  (\href{mailto:lnobreg@gmail.com}{\underline{L. Nobre G.}})
}
\date{\today}
\begin{document}
```

-U:--- metapostpropaganda.tex Top L18 (LaTeX)-----

See code



See code.

graphicx

```
\usepackage{graphicx}  
\DeclareGraphicsRule{*}{mps}{*}{}  
—  
\includegraphics[height=61mm]{fekslatexmp.1}
```

beamer+xmpmulti

```
\usepackage{xmpmulti}
```

```
\multiinclude[< + >][graphics={width=8cm},end=10]  
{feksmulti}
```


GNUPLOT

```
set term mp color solid latex  
set output "feksgnuplot.mp"
```

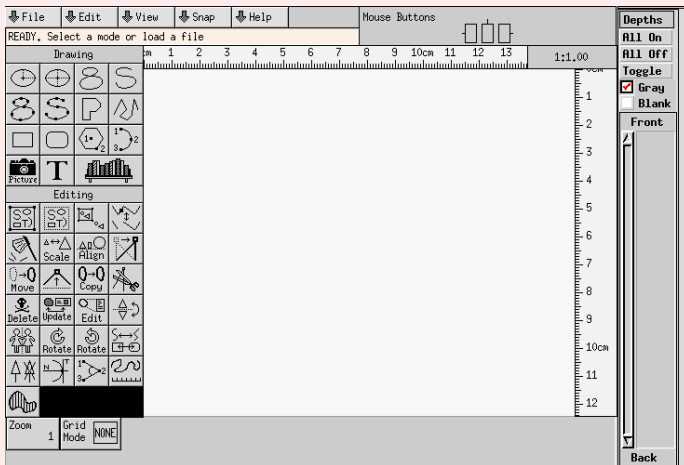
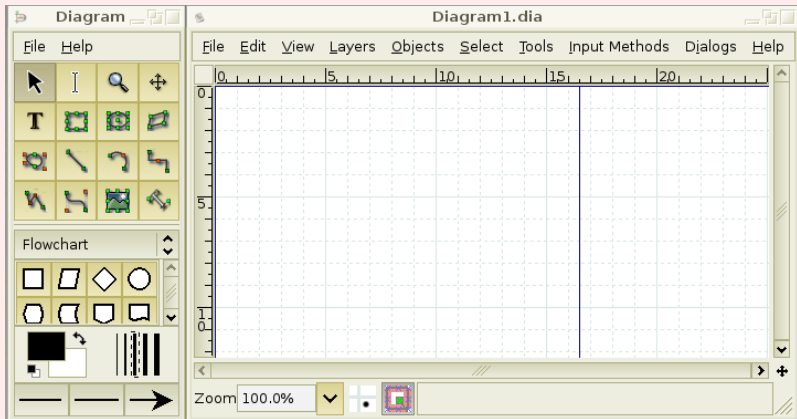


fig2ps

dia



epspdf+gs

epspdf minimal-1.mps

```
gs -q -sDEVICE=jpeggray -r100 -dNOPAUSE  
-sOutputFile=minimal.jpg minimal-1.pdf < /dev/null
```

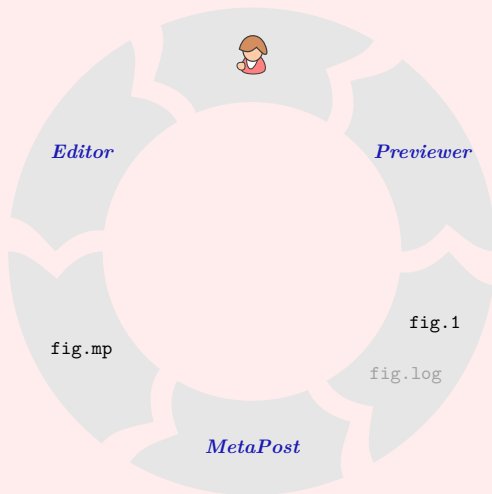
potrace+pstoedit+PstoeditMpostPreeditor

Example

PstoeditMpostPreeditor

Example

Basic workflow



Workflow

- ① Draw scheme by hand
- ② Identify parameters

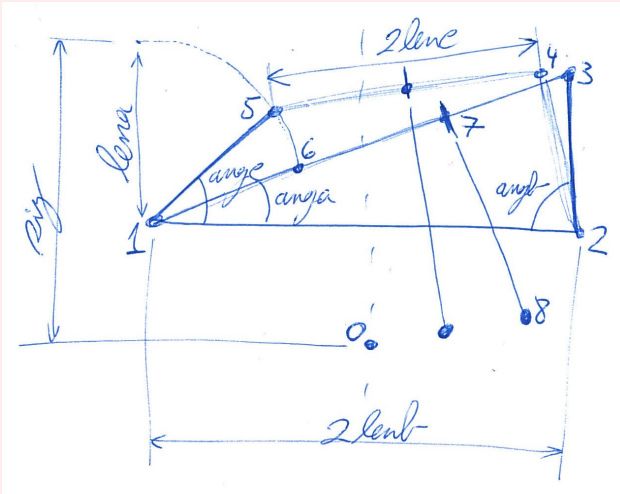
Workflow

- 1 Draw scheme by hand
- 2 Identify parameters
- 3 Express constraints
- 4 Determine valid ranges
- 5 Program
- 6 Test

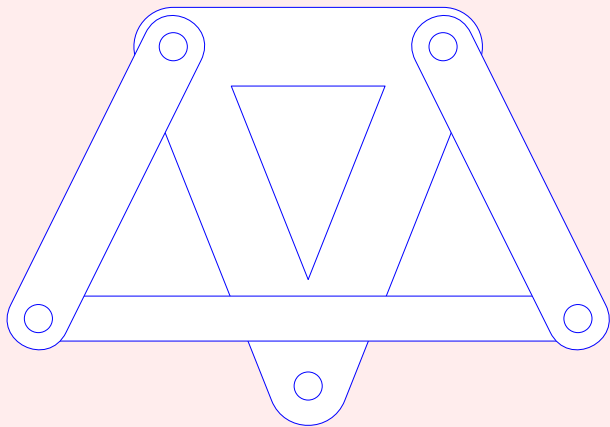
Workflow example (0)



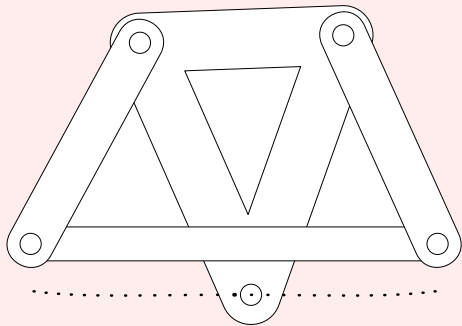
Workflow example (1)



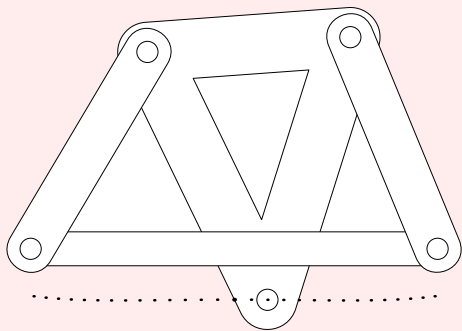
Workflow example (2)



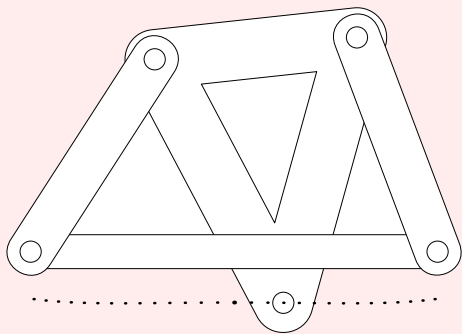
Workflow example (2)



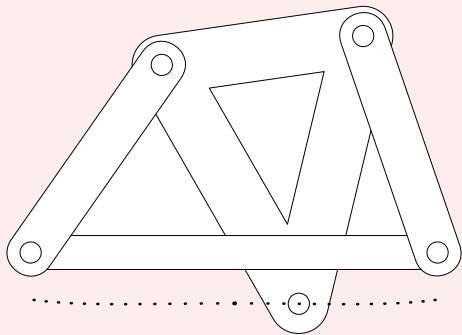
Workflow example (2)



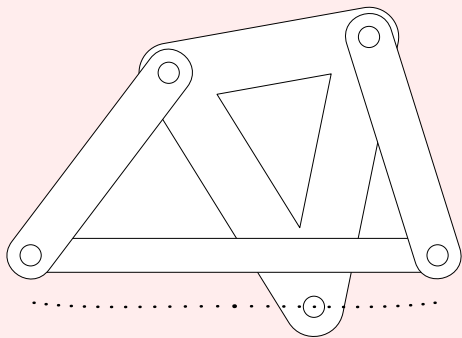
Workflow example (2)



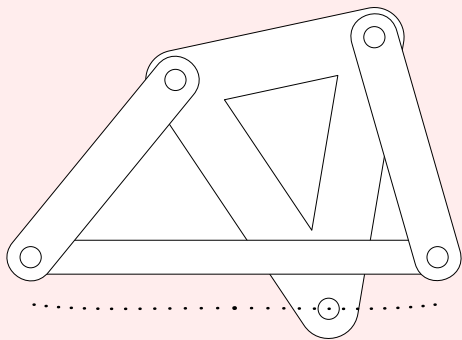
Workflow example (2)



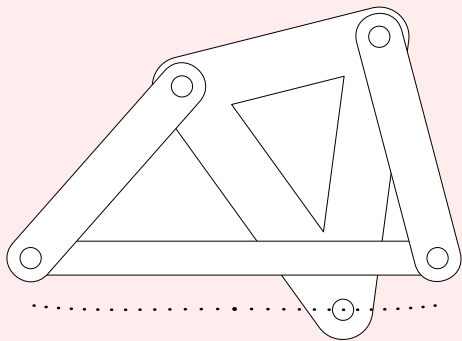
Workflow example (2)



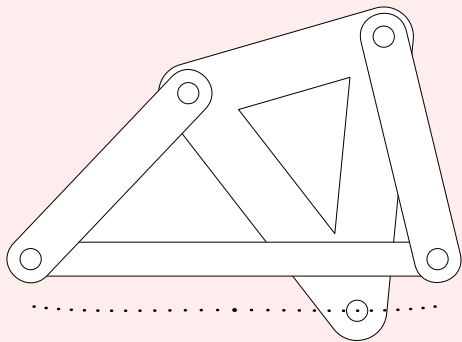
Workflow example (2)



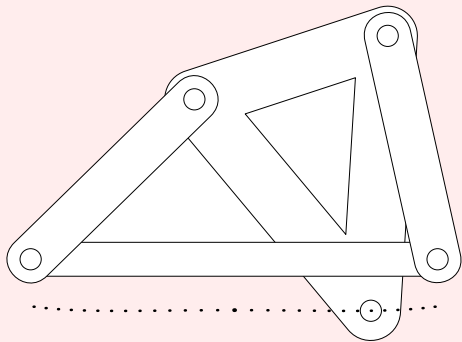
Workflow example (2)



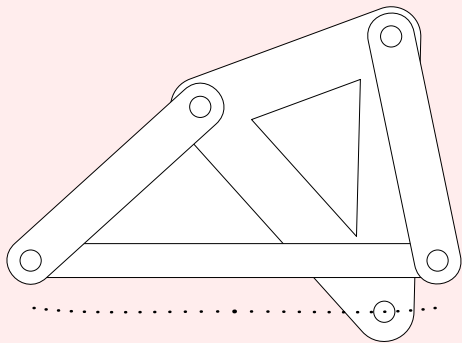
Workflow example (2)



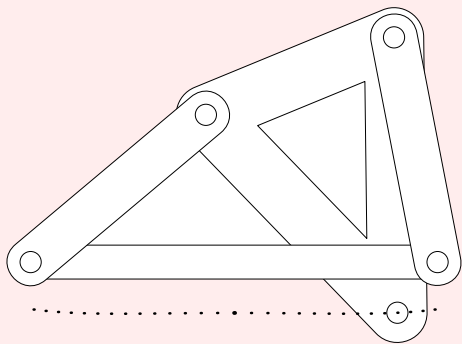
Workflow example (2)



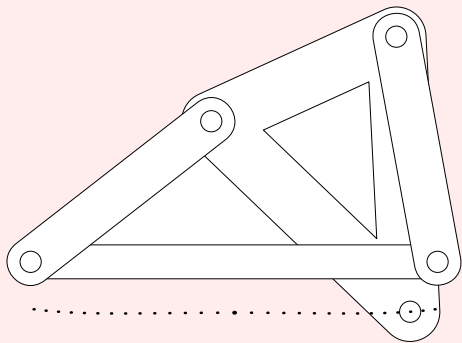
Workflow example (2)



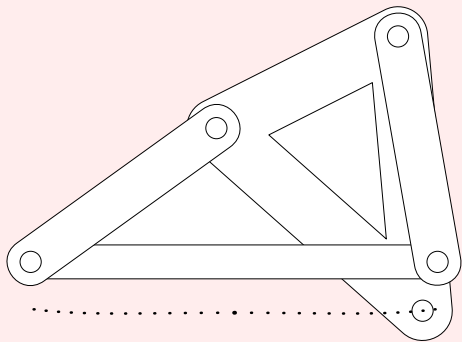
Workflow example (2)



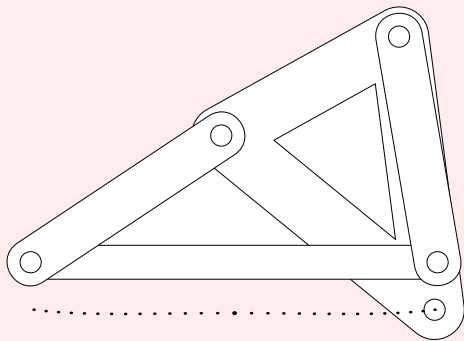
Workflow example (2)



Workflow example (2)



Workflow example (2)



Workflow example (2)

