Strategies for including graphics in \LaTeX\ documents

Klaus Höppner

GuIT meeting 2005
<table>
<thead>
<tr>
<th></th>
<th>Graphics Formats</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>\LaTeX{} graphicx package</td>
</tr>
<tr>
<td>3</td>
<td>Supported formats</td>
</tr>
<tr>
<td>4</td>
<td>Tools</td>
</tr>
</tbody>
</table>
Overview of graphics formats

Classification of graphics formats:

**Vector graphics** set up by geometrical elements like lines, curves, polygons, circles, . . .

**Bitmap graphics** store image information as a set of colored pixels with a given resolution and color depth. Different compression methods exist:

- bitmaps with data compression only
- bitmaps with lossy compression
Example: vector drawing

- Vector drawings are fine for geometrical drawings
- Advantage: Easily scaleable
- Advantage: Optimal quality independent of resolution of output device
Example: bitmap

a bitmap with few colors and sharp borders

- Sometimes you have to use bitmaps when you don’t have a mathematical representation of your drawing (e.g. no data set for land and political borders in shown example)
- Disadvantage: Loss of quality when scaling or zooming
- Disadvantage: Loss of quality when image resolution doesn’t fit to resolution of output device
Example: photo

A photograph has many colors (typically 16 mio) and smooth transitions.

No mathematical representation.

Again: Loss of quality when zooming into the photo (low resolution photo of big size).
Comparison

- Vector drawing
- Low resolution bitmap (pixels visible)
- Artifacts in a bitmap with lossy compression
Graphic formats in detail

**EPS** encapsulated postscript can contain vector drawings and bitmaps

**PNG** bitmapped portable network graphics format is a successor of GIF, supporting both compression with and without lossy compression

**JPG** bitmap format with lossy compression, often used for photographs (e.g. digital cameras)

**TIFF** a bitmap format often used for high quality DTP, supports CMYK color space
<table>
<thead>
<tr>
<th>Graphics Formats</th>
<th>\LaTeX\ graphicx package</th>
<th>Supported formats</th>
<th>Tools</th>
</tr>
</thead>
</table>

**Guidelines**

- For geometrical drawings (e.g. technical drawings, data plots) use a vector format like EPS or PDF.
- If you have a bitmap with sharp borders, use PNG.
- For photographs with high color depth and smooth transitions use JPEG (100–150 dpi are enough in most cases).
- In high quality DTP, use TIFF for photographs (especially if you need to support CMYK color space).
Including graphics in \LaTeX documents is supported by the packages `graphics` and `graphicx`

`graphicx` is an extension of `graphics` supporting key-value-options for e.g. scaling and rotating

Load `graphicx` package with

\usepackage{graphicx}

Modern \TeX systems assume `dvips` as backend when using `latex` as compiler and `pdftex` as backend when using `pdflatex`

For other backends use

\usepackage[backend]{graphicx}
Including a graphics file

- You can include an image in its natural size with
  \includegraphics{sample}

- Use options as key-value-pairs (graphicx):
  \includegraphics[key1=opt1,key2=opt2,...]{sample}

- Common options are:
  - `scale` to scale the image by a factor
  - `width` to scale the image to fit a width
  - `height` to scale the image to fit a height
  - `angle` to rotate the image by an angle with the lower left corner as fix point (positive: counter-clockwise)
  - `keepaspectratio` scale uniquely in x- and y-direction even if both width and height are given
Examples

\includegraphics[width=.3\linewidth]{sample}

\includegraphics[width=.3\linewidth,angle=20]{sample}

\includegraphics[angle=20,width=.3\linewidth]{sample}

\includegraphics[width=.3\linewidth,angle=20]{sample}
More examples

\includegraphics[width=1in,height=1in]{sample}

\includegraphics[width=1in,height=1in,keepaspectratio]{sample}
Supported graphics formats

- Support for graphics file formats and support for features like scaling and rotating depend on the used backend.
- Both `dvips` and `pdftex` support scaling and rotating.
- `dvips` supports EPS.
- `pdftex` supports PNG, PDF, JPEG, and MPS (METAPOST output).
- Include images without extension and the backend driver will look for a supported format (so it’s easy to switch between `latex` and `pdflatex` without changing the document).
## Converting to a supported format

<table>
<thead>
<tr>
<th>Source</th>
<th>Target</th>
<th>Tool</th>
<th>Target</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>EPS</td>
<td>✔</td>
<td>–</td>
<td>PDF</td>
<td>epstopdf</td>
</tr>
<tr>
<td>PDF</td>
<td>EPS</td>
<td>gs</td>
<td>✔</td>
<td>–</td>
</tr>
<tr>
<td>PNG</td>
<td>EPS</td>
<td>ImageMagick</td>
<td>✔</td>
<td>–</td>
</tr>
<tr>
<td>JPEG</td>
<td>EPS</td>
<td>ImageMagick</td>
<td>✔</td>
<td>–</td>
</tr>
<tr>
<td>TIFF</td>
<td>EPS</td>
<td>ImageMagick</td>
<td>✔</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td></td>
<td>or tif2eps</td>
<td>PNG</td>
<td>ImageMagick</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>PDF</td>
<td>tif2eps+epstopdf</td>
</tr>
</tbody>
</table>
Tools for graphics conversion

**ImageMagick** command line tool for graphics conversion and manipulation (changing size, gamma correction, ...), available for Unix and Windows

**netpbm** command line conversion tools, mainly on Unix but Windows binaries exist

**gs** Ghostscript is a PostScript interpreter available for various OS

**epstopdf** is a Perl script to convert EPS to PDF using gs

**tif2eps** by Bogusław Jackowski et al. uses gs to convert TIFF to EPS. Nice tool, also supporting CMYK color space.

**GUI tools** like Gimp, Adobe Photoshop, Corel Draw, ...
Additional tools: potrace

- potrace is a tool to trace a pure black and white bitmap and produce a vector drawing
- potrace is a command line tool, binaries available for Unix, Mac OS X and Windows
- input formats are PBM, PGM, PPM
- output format is EPS
- Cool!
Example

original bitmap

traced vector drawing
Additional tool: package overpic

- LaTeX package written by Rolf Niepraschck
- overlays an image with a LaTeX picture environment
- you can add new elements to the picture (text, symbols, ...)

Example:

\begin{overpic}[grid,tics=5]{map}
\put(32,74){\includegraphics[scale=.3]{busstop.mps}}
\put(32,77){\llap{\scriptsize\colorbox{back}{Windm"uhle}}}
\put(28,63){\textcolor{red}{\ding{55}}}
...\put(6.3,13){\colorbox{back}{\Pisymbol{ftsy}{68}}}
\put(29.8,61.4){\color{blue}\vector(-1,-3){2}}
\put(38.6,63){\color{blue}\vector(1,3){2}}
\end{overpic}
Example

original  with grid  final

Windmühle
Mainzer Str.
7
Haus für Industriekultur
A Fr