The phflplx package — include PDF graphics with hyperlinks

Philippe Faist *philippe.faist@bluewin.ch*

April 09, 2021

phflplx—A handy LaTeX package for including graphics defined via LaTeX source code files. Designed for use with the `ltxpdflinks` tool in order to include PDF graphics in documents with external or internal hyperlinks.

1 Introduction

WRITE DOC!

Usage:

```
\usepackage{phflplx}
\DeclareGraphicsExtensions{.lplx,.pdf}
```

2 Implementation

Load some useful packages.

```
\RequirePackage{etoolbox}
```

Check that the user is loading the hyperref package! (We don't load it automatically to avoid issues of package loading order, link appearance, etc.)

```
\AtBeginDocument{%
  \@ifpackageloaded{hyperref}{}{%
    \PackageWarning{phflplx}{The package 'hyperref' was not loaded. You
      probably forgot to load it.}%
  }%
}
```

Declaring the graphics rule for the LPLX extension. Note that the LPLX file has the line %%BoundingBox 0 0 <W> <H> so that the size of the graphics can be parsed by the default graphics graphic size inspector, which is designed to parse %%BoundingBox commands in EPS files.

```
\DeclareGraphicsRule{.lplx}{lplx}{*}{}
```

Declare the driver functions for the graphics package internals.

```
\def\Ginclude@lplx#1{%
  \message{<#1>}%
  \input{#1}%
}
```

This command gets called from within the LPLX files to actually include the underlying graphics.

```
\def\lplxIncludeGraphics{%
  \edef\x{\noexpand\hbox to 0pt{%
      \noexpand\Ginclude@graphics{\Gin@base\lplxiGraphicExt}}}%
  \x
}
```

Declare some simple helpers that are used in our automatically generated LPLX files.

```
\newdimen\lplxiBboxWcrdim
\newdimen\lplxiBboxHcrdim
\def\lplxSetBbox#1#2#3#4{%
  \def\lplxiBboxX{#1}%
  \def\lplxiBboxY{#2}%
  \def\lplxiBboxW{#3}%
  \def\lplxiBboxH{#4}%
  \ifnum\lplxiBboxX>0\relax\lplx@bboxzerowarn\fi
  \ifnum\lplxiBboxX<0\relax\lplx@bboxzerowarn\fi
  \ifnum\lplxiBboxY>0\relax\lplx@bboxzerowarn\fi
  \ifnum\lplxiBboxY<0\relax\lplx@bboxzerowarn\fi
}
\def\lplx@bboxzerowarn{%
  \PackageWarning{phflplx}{LPLX bounding box is not pinned at (0,0), not supported}%
}
```

Utility to set up the appropriate command arguments to use for \scalebox, etc.

```
\def\lplx@noscale{1,!}
\def\lplx@exclam{!}
\def\lplxSetupScaleAndBbox{%
```

First, we locally define a macro \lplxaDoScale{...} that will generate the correct \scalebox call with the given contents, according to the requested size.

```
  \def\lplx@tmp{\Gin@scalex,\Gin@scaley}%
```

```
\ifx\lplx@tmp\lplx@noscale%
  \def\lplxaDoScale{}%
\else
  \ifx\Gin@scaley\lplx@exclam
    \edef\lplxaDoScale{\noexpand\scalebox{\Gin@scalex}}%
  \else
    \ifx\Gin@scalex\lplx@exclam
      \edef\lplxaDoScale{\noexpand\scalebox{\Gin@scaley}}%
    \else
      \edef\lplxaDoScale{\noexpand\scalebox{\Gin@scalex}[\Gin@scaley]}%
    \fi
  \fi
\fi
```

Second, we need to take care of setting the bounding box correctly. Define `\lplxvCropX` and `\lplxvCropY` which are the (X, Y) position of the lower left corner the part of the image we want to pick out, in user space units. Define `\lplxvCropW` and `\lplxvCropH` as the requested width & height of the subimage we want to use.

```
  \edef\lplxvCropX{\Gin@llx}%
  \edef\lplxvCropY{\Gin@lly}%
  \edef\lplxvCropW{\strip@pt\dimexpr\Gin@urx pt-\Gin@llx pt\relax}%
  \edef\lplxvCropH{\strip@pt\dimexpr\Gin@ury pt-\Gin@lly pt\relax}%
}
```

Finally, a utility that will help us place links in the picture environment. Usage is `\lplxPutLink{`⟨*x*⟩`}{`⟨*y*⟩`}{`⟨*w*⟩`}{`⟨*h*⟩`}{`⟨*hyperstartcmd*⟩`}{`⟨*hyperendtokens*⟩`}`. Here `{`⟨*hyperstartcmd*⟩`}` and `{`⟨*hyperendtokens*⟩`}` are any tokens that will be inserted immediately before and immediately after an invisible rule of the given width and height. The rule will be in curly braces, so it can be considered as a mandatory argument to the last macro token in `{`⟨*hyperstartcmd*⟩`}`.

```
\def\lplxPutLink{%
  \ifGin@clip
    \expandafter\lplx@clipputlink
  \else
    \expandafter\lplx@doputlink
  \fi
}%
\def\lplx@doputlink#1#2#3#4#5#6{% x,y,w,h,hyperstart,hyperend
  \put(#1,#2){#5{\phantom{\rule{#3bp}{#4bp}}}#6}%
}%
\newdimen\lplx@tmpx
\newdimen\lplx@tmpy
\newdimen\lplx@tmpw
\newdimen\lplx@tmph
\def\lplx@clipputlink#1#2#3#4#5#6{% x,y,w,h,hyperstart,hyperend
```

Some notes: 1) We use “pt” as dummy unit of measure here just to do the floating point arithmetic and we use `\strip@pt` at the end. 2) Here `\lplx@maybeskip`

serves as a flag that if set, asserts the link was entirely cropped out of the picture. Initially it expands to an empty string but when set it expands to `\p@<\z@\relax` (="`1pt < 0pt`"), so it can be placed in front of all `\ifdim`'s so that they are skipped if the link was determined to be out of the picture.

```
\def\lplx@maybeskip{}%
\def\lplx@setskip{\def\lplx@maybeskip{\p@<\z@\relax}}%
\lplx@tmpx=#1pt\relax
\lplx@tmpy=#2pt\relax
\lplx@tmpw=#3pt\relax
\lplx@tmph=#4pt\relax
\ifdim\lplx@maybeskip\lplx@tmpx<\lplxvCropX\p@\relax
  \ifdim\dimexpr\lplx@tmpx+\lplx@tmpw>\lplxvCropX\p@\relax
    \lplx@tmpw=\dimexpr\lplx@tmpx+\lplx@tmpw-\lplxvCropX\p@\relax
    \lplx@tmpx=\lplxvCropX\p@\relax
  \else
    \lplx@setskip
  \fi
\fi
\ifdim\lplx@maybeskip\dimexpr\lplx@tmpx+\lplx@tmpw>\dimexpr\lplxvCropX\p@+\lplxvCropW\p@\rel
  \ifdim\lplx@tmpx<\dimexpr\lplxvCropX\p@+\lplxvCropW\p@\relax
    \lplx@tmpw=\dimexpr\lplxvCropX\p@+\lplxvCropW\p@-\lplx@tmpx\relax
  \else
    \lplx@setskip
  \fi
\fi
\ifdim\lplx@maybeskip\lplx@tmpy<\lplxvCropY\p@\relax
  \ifdim\dimexpr\lplx@tmpy+\lplx@tmph>\lplxvCropY\p@\relax
    \lplx@tmph=\dimexpr\lplx@tmpy+\lplx@tmph-\lplxvCropY\p@\relax
    \lplx@tmpy=\lplxvCropY\p@\relax
  \else
    \lplx@setskip
  \fi
\fi
\ifdim\lplx@maybeskip\dimexpr\lplx@tmpy+\lplx@tmph>\dimexpr\lplxvCropY\p@+\lplxvCropH\p@\rel
  \ifdim\lplx@tmpy<\dimexpr\lplxvCropY\p@+\lplxvCropH\p@\relax
    \lplx@tmph=\dimexpr\lplxvCropY\p@+\lplxvCropH\p@-\lplx@tmpy\relax
  \else
    \lplx@setskip
  \fi
\fi
\ifdim\lplx@maybeskip\p@>\z@\relax
  \edef\x{\noexpand\lplx@doputlink%
    {\strip@pt\lplx@tmpx}{\strip@pt\lplx@tmpy}{\strip@pt\lplx@tmpw}{\strip@pt\lplx@tmph}}%
  \message{*****\detokenize\expandafter{\x}*******}%
  \x{#5}{#6}%
\fi
}%
```

Change History

v0.1

General: Initial version 1

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.