ConTEXt

title : ConTEXt User Module
subtitle : Degrading JPEG images
author : Peter Münster
date : September 12, 2006

```
\writestatus{loading}{Degrading JPEG images on the fly}
```

```
\unprotect
```

In this module we suppose, that original figures are either eps or jpg. When creating pdf, eps is converted to pdf, and when creating dvi, jpg is converted to eps. Furthermore, jpg-images can be degraded to a given resolution, to get smaller documents.

When `http://context.literatesolutions.com/collector/63` is solved, we can make it better using `\appliedfigurefilename`, `\figurewidth` etc, and without redefining `\externalfigure`.

`\setupDegr..` Sets the resolution, the directory for degraded images and the directory, where the original images can be found (source directory).

Default setup:

```
\setupDegrade[Res=100,Dir=degrade,SDir=.]
```

Example:

```
\setupDegrade[Res=600,Dir=degraded-images,SDir=/home/peter/jpegs]
```

```
\def\setupDegrade[#1]{\getparameters[Deg][#1]}
\setupDegrade[Res=100,Dir=degrade,SDir=.]
```

In order to use this module, you must use `\externalfigure` in the following way: `\externalfigure[file-prefix][options]`, where `options` must contain the width or the height. Furthermore, `\write18` must be enabled. The shell-script is in a buffer, just to keep this module in only one file.

```
\startbuffer[degrade-script]
```

```
for i in jpeg2ps convert identify bc; do
 if ! type $i &>/dev/null; then
  echo Error: $i is not installed.
  exit 1
 fi
done
```

```
EPSTOPDF=epstopdf
```

```
if ! type epstopdf &>/dev/null; then
 if type texmfstart &>/dev/null; then
  EPSTOPDF="texmfstart newpstopdf"
 else
  EPSTOPDF="texutil --figures --epstopdf"
 fi
fi
```

```
if [ -s "$1.jpg" ]; then
 if [ "$2" = eps ] && [ "$1.jpg" -nt "$1.eps" ]; then
  jpeg2ps "$1.jpg" >"$1.eps"
 fi
else
 if [ "$2" = pdf ] && [ "$1.eps" -nt "$1.pdf" ]; then
  $EPSTOPDF "$1.eps"
 fi
fi
```

Degrading JPEG images

```
X=
[ $3 = h ] && X=x
FORMAT="%$3"
FILE="$6/$1.jpg"
DFILE="$7/$1.jpg"
DFILE_EPS="$7/$1.eps"
DPI="$4"
IW="$5"
MODE="$2"
INCH=72.27

mkdir -p $7

if [ -s "$FILE" ]; then
 W=`identify -format "$FORMAT" "$FILE"`
else
 rm -f "$DFILE"
 exit 0
fi

NW=`echo "$DPI * $IW / $INCH" | bc`

if [ $NW -ge $W ]; then
 rm -f "$DFILE"
else
 if [ -s "$DFILE" ] && \
  [ `identify -format "$FORMAT" "$DFILE"` -eq $NW ]; then
  exit 0
 else
  convert -verbose -resize $X$NW "$FILE" "$DFILE"
  [ $MODE = eps ] && jpeg2ps "$DFILE" >"$DFILE_EPS"
 fi
fi
\stopbuffer

\def\Command{\immediate\write18}
\let\externalfigureO=\externalfigure
\def\externalfigure[#1][#2]{%
  \getparameters[Deg][height=,#2]%
  \processaction[\Degheight][%
    \s!default =>\scratchdimen=\Degwidth  \def\Deg@WH{w},
    \s!unknown =>\scratchdimen=\Degheight \def\Deg@WH{h}]%
  \Command{bash ./\jobname-degrade-script.tmp #1 \ifcase\pdfoutput eps\else
    pdf\fi\space\Deg@WH\space\DegRes\space\withoutpt\the\scratchdimen\space
    \DegSDir\space\DegDir}%
  \doiffileexistselse{\DegDir/#1.jpg}{\def\Deg@File{\DegDir/#1}}{%
    \def\Deg@File{#1}}%
  \externalfigureO[\Deg@File][#2]}

\protect

\doifnotmode{demo}{\endinput}
```

Usage example:

```
17  \starttext
    \externalfigure[hacker][width=0.2\textwidth]
    \stoptext
```