PPTIMPORT Package

Matteo Oldoni (matteo.oldoni@polimi.it), May 3rd, 2024

May 3, 2024

Abstract

Automatic import of pictures and slides from Microsoft Power Point with optional replacement of placeholder text in Power Point or in Latex. Works only under Microsoft Windows.

1 User's Guide

Drawing pictures in Latex is a pain in the ass, even with the available drawing packages, most notably the all-powerful TikZ (a brilliant work: the REAL way to draw pictures in Latex). Mostly, I rather prefer to draw my pictures and diagrams in Microsoft Power Point, and here's the problem.

1.1 The traditional way

If you need to import a picture drawn in a slide in Microsoft Power Point into a Latex document, you would typically need to open Power Point, go to the slide that you want to export, select all the shapes, export as PNG into the Latex document's folder, insert a figure snippet in the Latex document like `\figure\includegraphics{fig.png}\end{figure}`. PNG is exported as a bitmap file, thus any shape in Power Point will be rendered as a pixel image, thus loosing definition when zooming.

If you would then wish to do the same but in vector form, so as to retain vector shapes such as lines, rectangles, circles... drawn in Power Point, then PDF is your option. However, in Power Point you can either save the whole presentation as a PDF from which you should then extract the slide with your picture, or you may resort to printing one page to PDF. Either way, you end up with a full slide in a pdf. Cropping the PDF can be done by either `Briss` (available in its original https://sourceforge.net/projects/briss/ and enhanced version https://github.com/mbaeuerle/Briss-2.0) or by other 3rd-party tools to crop. Finally, your cropped PDF can be placed in the Latex document `\figure\includegraphics{fig.pdf}\end{figure}`

1.2 The "pptimport" way

To save this hassle, pptimport was born. It provides 6 commands to automatically import a slide's content into a Latex figure, providing alternative forms under-the-hood to use a PNG or a PDF file. In its simplest form, a figure with the content of slide 3 of a Power Point presentation figures.pptx is created by

```
\begin{figure}\includegraphicspptpdf{figures.pptx}{3}\end{figure}
```

1.3 Package Installation

Three files are needed for this package: pptimport.sty, exportPptToPng.vbs, exportPptToPngReplace.vbs. All must be in a directory reachable from the system path, i.e. we suggest to add the parent folder to the %PATH% environment variable.

The package pptimport does the steps outlined above automatically and creates the needed pdf or png files on the fly, verifying they are up-to-date with the source PPT or PPTX file and cropping to the content. Under-the-hood, the two Visual Basic scripts written for Windows are used, and a working installation of Microsoft Power Point must be available to do the background work of opening and exporting. But none of this should concern you: once installed, the package is very simple to use.

1.4 Package Loading

You must import `\usepackage{pptimport}` Then the following commands become available:

- `\includegraphicspptpdf`: import a slide from PPT/PPTX "as-is"
- `\includegraphicspptpng`: import a slide from PPT/PPTX
- `\includegraphicspptpdfsubs`: import a slide from PPT/PPTX replacing text placeholders in Power Point
- `\includegraphicspptpngsubs`: import a slide from PPT/PPTX replacing text placeholders in Power Point
- `\includegraphicspptpdfsubstex`: import a slide from PPT/PPTX replacing text placeholders in Latex
- `\includegraphicspptpngsubstex`: import a slide from PPT/PPTX replacing text placeholders in Latex

All work by opening a PPT or PPTX file containing the figures. When importing, a specific slide will be converted to a figure, therefore separate figures should be placed on separate slides or separate PPT/PPTX files. The PPT or PPTX file or files must be available to Latex, therefore we recommend placing the figure PPT/PPTX files in the same folder as the Latex document.

1.5 Import a slide content "as-is"

The two following commands simply import the content of a PPT or PPTX slide into Latex:

```
\includegraphicspptpdf[OPTIONS]{pptfile}{slidenumber}
```

```
\includegraphicspptpng[OPTIONS]{pptfile}{slidenumber}
```

Once executed, they create respectively a PDF or PNG file with the content from a PPT/PPTX file named as specified in the pptfile argument, scrolling to the required slide number (slidenumber, numbered from 1) and cropping the output to the true extent of the content of the slides, removing white borders.

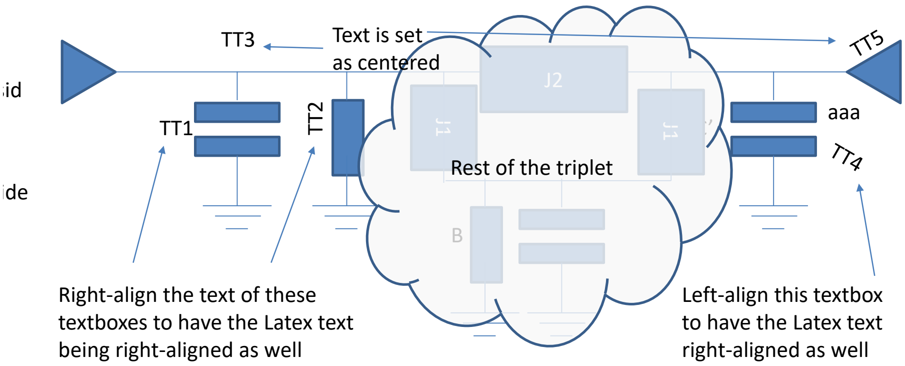


Figure 1: PPT to PDF: Original Power Point Slide with placeholders TT1, TT2... including some objects partially outside the slide (which become clipped):”includegraphicspptpdf[width=columnwidth]TestFigures.pptx1”

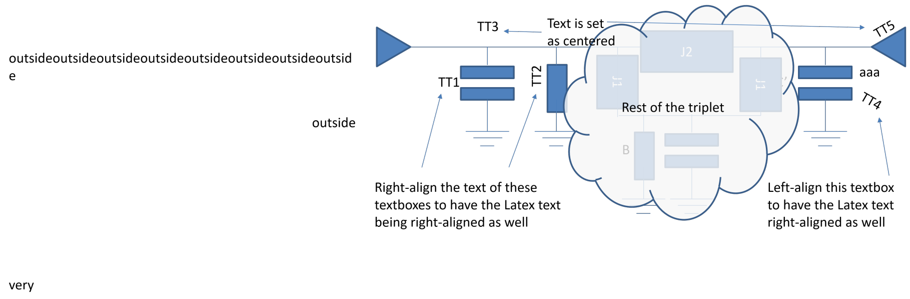


Figure 2: PPT to PNG: Original Power Point Slide with placeholders TT1, TT2... including some objects partially outside the slide, as png: ”includegraphicspptpng[width=columnwidth]TestFigures.pptx1”

The PDF version will retain all vector capabilities of the shapes drawn in Power Point (i.e. lines, rectangles, circles...), whereas the PNG will rasterize everything to a pixel-based image, which is exported as a very large file but will anyway have a finite resolution at close range.

The PDF version will automatically crop away any object or part of object beyond the slide border, therefore some notes and comments not meant to be included in the Latex document can be placed outside of the slide. The PNG version instead will retain also all objects outside of the slide border.

1.6 Importing a slide replacing text in Power Point

The two following commands replace ”TT1”, ”TT2”... placeholder texts in the content of a PPT or PPTX slide and import the resulting figure into Latex:

```
\includegraphicspptpdfsubs[OPTIONS]{pptfile}{slidenum}{rep1;rep2;...}
```

```
\includegraphicspptpngsubs[OPTIONS]{pptfile}{slidenum}{rep1;rep2;...}
```

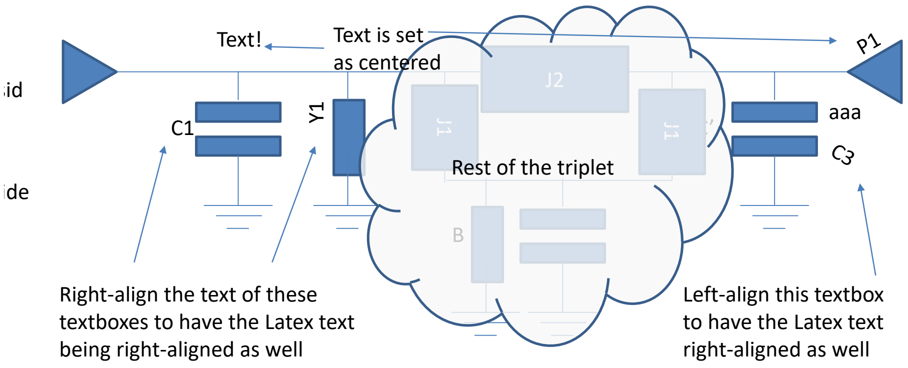


Figure 3: PPT to PDF: Using replacement in Power Point

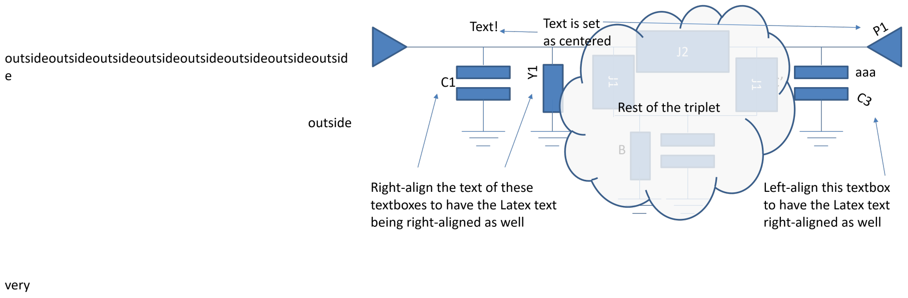

outsideoutsideoutsideoutsideoutsideoutsideoutsideoutsid
e
outside
very
Text!
Text is set
as centered
J2
J1
J1
Y1
C1
Rest of the triplet
B
aaa
C3
P1
Right-align the text of these
textboxes to have the Latex text
being right-aligned as well
Left-align this textbox
to have the Latex text
right-aligned as well

Figure 4: PPT to PNG: Using replacement in Power Point

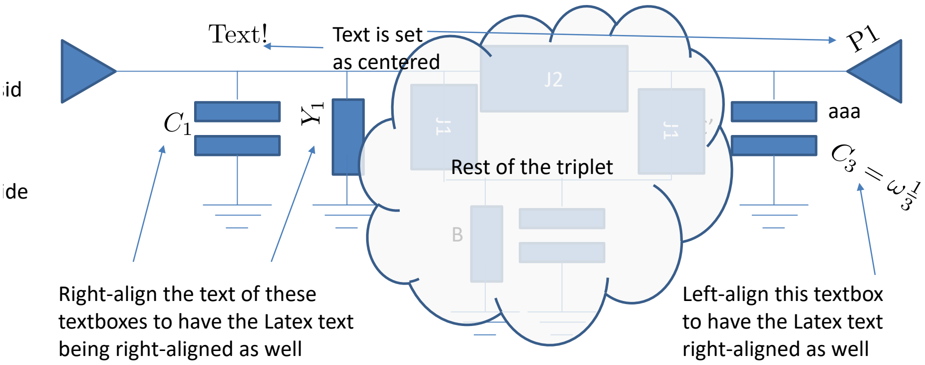
Their behavior is very similar to `\includegraphicsppptpdf` and its companion `\includegraphicsppptpng`, although a text replacement is done before saving each PDF/PNG figure. Specifically, the Power Point slide of interest should contain some shapes with placeholder text in the form "TT1", "TT2"... Every occurrence of the text "TT1" will be replaced by the text given in rep1, and similarly for the other repN strings. The text properties and effects set in Power Point (e.g. font, text size, margins, placement, color...) will be preserved in the replacement. The original Power Point file is not affected by the replacement.

```
\includegraphicsppptpdfsubs[width=\columnwidth]{TestFigures.pptx}{1}{C1;Y1;Text!;C3;P1}%
\includegraphicsppptpngsubs[width=\columnwidth]{TestFigures.pptx}{1}{C1;Y1;Text!;C3;P1}%
```

Again, options for scaling, width and rotation can be provided as options, which are directly passed on to the usual `\includegraphics` command. Scaling will also scale the text.

1.7 Importing a slide replacing text in Latex

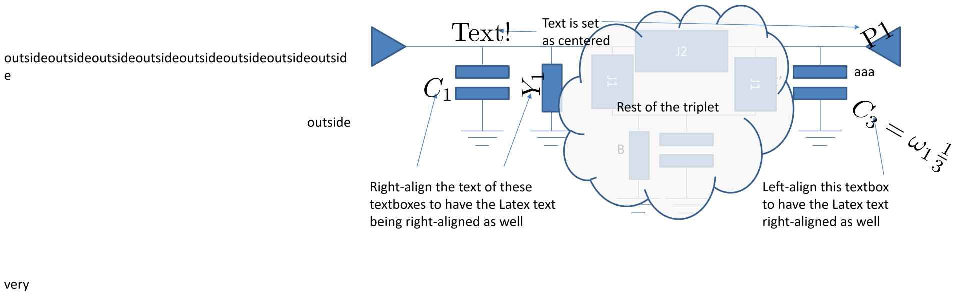
The two following commands replace "TT1", "TT2"... placeholder texts in the content of a PPT or PPTX slide and import the resulting figure into Latex, in



Right-align the text of these textboxes to have the Latex text being right-aligned as well

Left-align this textbox to have the Latex text right-aligned as well

Figure 5: PPT to PDF: Doing replacement in Latex


outsideoutsideoutsideoutsideoutsideoutsideoutsideoutsid
e
outside
very
Text!
Text is set
as centered
P1
$C_1$
$Y_1$
J2
J1
J1
Rest of the triplet
aaa
$C_3 = \omega_1\frac{1}{3}$
B

Right-align the text of these textboxes to have the Latex text being right-aligned as well

Left-align this textbox to have the Latex text right-aligned as well

Figure 6: PPT to PNG: Doing replacement in Latex

this case by typesetting the replacement in Latex:

```
\includegraphicspptpdfsubstex[OPTIONS]{pptfile}{slidenum}{rep1;rep2;...}
```

```
\includegraphicspptpngsubstex[OPTIONS]{pptfile}{slidenum}{rep1;rep2;...}
```

The syntax is identical to `\includegraphicspptpdfsubs` and its companion `\includegraphicspptpngsubs`, but in this case the replacement strings can be Latex-formatted blocks, i.e. formulas, boxes, math... E.g., TT1, TT2, TT3, TT4 and TT5 can be replaced as: `$C_1$;$Y_1$;Text`;$C_3 = \omega\frac{1}{3}$;P1!

Therefore, we advise to make the placeholders to contain only the ”TTn” text (not other fixed text in Power Point), otherwise overlaps may occur.

```
\includegraphicspptpdfsubstex[width=\columnwidth]{TestFigures.pptx}{1}{$C_1$;$Y_1$;Text!;$C_3=\omega\frac{1}{3}$;P1}
\includegraphicspptpngsubstex[width=\columnwidth]{TestFigures.pptx}{1}{$C_1$;$Y_1$;Text!;$C_3=\omega_1\frac{1}{3}$;P1}
```

2 Details

2.1 Flow

Importing slides ”as-is” requires the exportPptToPng.vbs script: it is responsible for checking the existence and/or modification timestamp against the

PPT/PPTX file and running the Power Point export operation again as needed. The generated figure is named `pptfile_slidenumber.pdf`. When uploading or sending the Latex source files, the `\includegraphicspptpdf[options]{pptfile}{slidenumber}` command can be replaced by `\includegraphics[options]{pptfile_slidenumber.pdf}`, so that the recipient is not required to rebuild figures from Power Point and directly uses the provided PDF/PNG files.

Importing slides with replacement in Power Point or in Latex instead relies on the exportPptToPngReplace.vbs script: it is responsible for checking the existence and/or modification timestamp of the PDF/PNG files and running the Power Point export operation again as needed. The generated figure by `\includegraphicspptpdfsubs` is named `pptfile_slidenumber_checksum.pdf`, where checksum is a numeric value computed in Latex accounting for the replacement strings. It changes when any of the replacement strings is changed, and hence triggers a regeneration of the figures at the next compilation. When uploading or sending the Latex source files, the `\includegraphicspptpdfsubs[options]{pptfile}{slidenu` command can be replaced by `\includegraphics[options]{pptfile_slidenumber_checksum.pdf}`, so that the recipient is not required to rebuild figures from Power Point and directly uses the provided PDF/PNG files.

When doing a replacement in Latex, a further ".t" file is generated: it contains the latex commands to typeset the replacement text at the needed coordinates (a sequence of `\put\makebox{...}` commands). Similarly, the figure and auxiliary .t files generated by `\includegraphicspptpdfsubstex` are named `pptfile_slidenumber_tchecksum.pdf` and `pptfile_slidenumber_tchecksum.pdf.t`, where checksum is a numeric value computed in Latex accounting for the replacement strings. In this case, when uploading or sending the Latex source files, the `\includegraphicspptpdfsubstex[options]{pptfile}{slidenumber}{repl...}` command can be replaced by `\includegraphics[options]{pptfile_slidenumber_tchecksum.pdf}` followed by `\input{pptfile_slidenumber_tchecksum.pdf.t}`, so that the recipient is not required to rebuild figures from Power Point and directly uses the provided PDF/PNG files and .t file.

2.2 Regeneration of figures from a slide "as-is"

The first time the Latex document is compiled, the package will do the required export, opening Power Point in the background and closing it afterwards. It happens once for every figure imported in this way, creating the required PDF/PNG files and saving them in the same folder.

At the next compilation of the Latex document, if the PDF/PNG files are already present and their modification date coincides with the source PPT/PPTX file, they are considered still valid and thus no longer exported: thus is faster than exporting from Power Point every time. If the expected PDF/PNG file is instead not present in the same folder, or if the file exist but its "Modified date" timestamp does not coincide with the source PPT/PPTX file, the figures are considered outdated, hence the export is carried out again.

Therefore, saving the PPT/PPTX file will invalidate all existing PNG/PDF files which will hence be regenerated at the next Latex compilation.

2.3 Regeneration of figures from a slide with replaced text

The criterion to decide that a PDF/PNG figure needs to be regenerated explained in Section 2.2 based on the timestamp is still valid.

However, when a replacement has been needed, the generated PDF/PNG figures need to be regenerated also when the replacement strings are updated in Latex, which is not apparent from the timestamp of the PPT/PPTX file. Therefore, the whole replacement list (sequence of semicolon-separated replacement strings) is "hashed" by a checksum. The computed checksum is appended to the PDF/PNG pictures. When the replacement strings are changed, the checksum changes and therefore a new PDF/PNG figure is regenerated. This might lead to many unused versions of the figures. However, cleaning them at the end and recompiling the final version will get rid of the unnecessary files.

The hashing is done on a simple algorithm in Latex and might break under some weird circumstances (i.e. weird expanded Latex commands used as replacement strings). The hashing algorithm moreover only considers the first 200 characters. Therefore, if the replacement list is longer, future modifications may not be detected and the figure may not be regenerated. In these cases, please manually remove the PPT/PNG figures before compiling the Latex document again.

These considerations also apply to the case of replacement done in Latex. However, in this case, an additional .t file is generated and is checked to determine that regeneration is necessary.

2.4 Replacement strings in Power Point

The replacement strings are separated by semicolon. The provided replaceN strings should only contain simple text and must not contain semicolon (otherwise these will split and be considered as two replacement strings). Currently no support for subscripts or superscripts is provided in a single placeholder. As a workaround, if needed, one can draw in Power Point a shape with the text "TT1" as main text size and add raised/lowered placeholder boxes "TT2" and "TT3" with a smaller font to emulate subscripts and superscripts around the main text.

If less strings are provided than the available TTn, the non-replaced placeholders will remain visible as TTn. If necessary, to make all placeholders invisible even when not replaced, make them with same font color as the background in Power Point (if possible). To not replace a placeholder, for instance TT3, pass an empty replacement string: i.e. firststring;secondstring;;fourthstring... (two successive semicolons) If a TTn placeholder is not found, no warning or error is issued, but the replacement of that string will not occur.

2.5 Replacement strings in Latex

The replacement in this case is approximate: the font will be the Latex one and the size is also determined by Latex. Rotation will be preserved; instead, vertical alignment (i.e. if the placeholder text is set in Power Point to be aligned to the bottom of the shape) will always be assumed at the top of the shape. The look of some placeholders might be different than in the original Power Point,

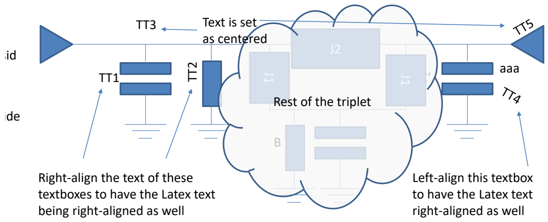


Figure 7: PPT to PDF "as-is" scaled down

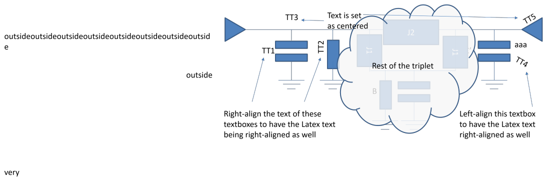


Figure 8: PPT to PNG "as-is" scaled down

due to margins and precise placement. Moreover, only one occurrence of each TTn will be replaced.

2.6 Scaling and other options

The imported figures can be scaled with the usual options of includegraphics, such as `width=0.7\columnwidth` or `scale=0.28` or others:

```
\begin{figure}\centering
\includegraphicspptpdf[width=0.6\columnwidth]{TestFigures.pptx}{1}%
\label{fig:}\caption{Caption}\end{figure}
```

Scaling a picture imported "as-is" from the slide scales everything including any text. Scaling a picture imported by replacing text in Power Point also scales everything including any text.

Instead, scaling a picture imported by replacing text in Latex in this case scales everything except for the Latex-replaced text, which remains readable at every scale of the rest of the picture. However, as the text might become too big to fit and thus overlap other shapes, the usual Latex commands can be used to change the size: `\small`, `\tiny`,`\large`... These commands can be used within each replace string (e.g. `$C_1$;\small$Y_1$;`) or a single text size command can be provided before includegraphics...:

```
\begin{figure}%
\centering\tiny %Scale text, otherwise it remains the default size
\includegraphicspptpdfsubstex[width=0.7\columnwidth]{TestFigures.pptx}{1}{$C_1$;$Y_1$;Text!;$\frac{C_3}{Y_1}$;$P1$}%
\end{figure}
```

3 Minimum Working Example

1. Installation:

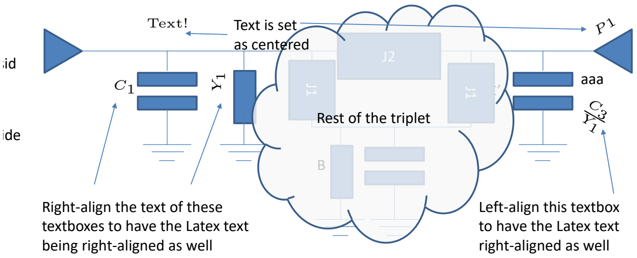


Figure 9: PPT to PDF with replacement in Latex, scaled down with default text

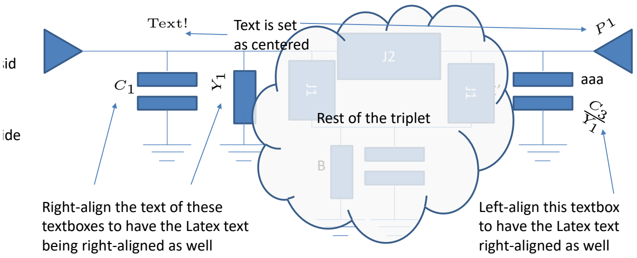


Figure 10: PPT to PDF with replacement in Latex, scaled down with ”tiny” text

   - Copy pptimport.sty, exportpptpng.vbs, exportpptpngreplace.vbs in a folder
   - Add that folder to the system Path (environment variable) or copy those 3 files in a new folder

2. Preparation:

   - Prepare a file Figures.pptx in a new empty folder or in the folder where the 3 files were copied.
   - In slide 1 remove any preexisting shapes and draw a rectangle and a circle. In the circle type ”Test”.
   - In slide 2 remove any preexisting shapes draw a rectangle and a circle. In the circle type ”TT1”.
   - Save the Figures.pptx file

3. Latex file: create the main.tex file and save it in the same folder as Figures.pptx:

```
\documentclass[10pt,,a4paper]{article}
\usepackage{pptimport}
\begin{document}
\begin{figure} \includegraphicspptpdf[width=0.7\columnwidth]{Figures.pptx}{1}
\caption{First example: import "as-is".}\end{figure}

\begin{figure} \includegraphicspptpdfsubs[width=0.7\columnwidth]{Figures.pptx}{2}{New test!}
\caption{Second example: import replacing in Power Point.} \end{figure}

\begin{figure} \includegraphicspptpdfsubstex[width=0.7\columnwidth]{Figures.pptx}{2}{$E=mc^2\frac{\omega}{4}$}
\caption{Third example: import replacing in Latex.} \end{figure}
\end{document}
```

4. Compile and observe the result.

5. Altering the PPTX file and saving will require recompiling the Latex document.

The result is as follows:

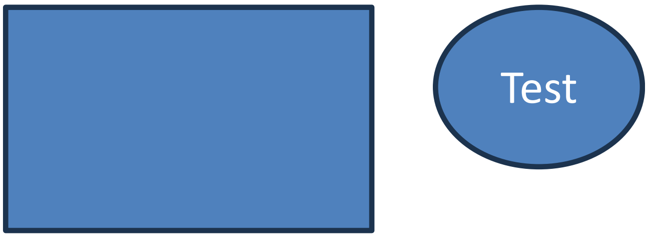


Figure 11: First example: import ”as-is”.

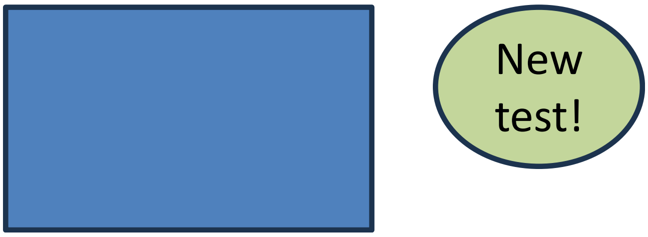


Figure 12: Second example: import replacing in Power Point.

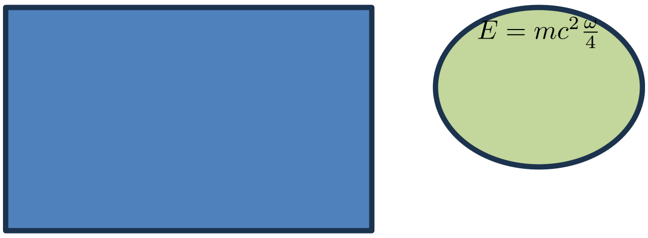


Figure 13: Third example: import replacing in Latex.