

# Package ‘polish’

June 3, 2026

**Title** Polishing Content for 'Word' and 'PowerPoint'

**Version** 0.3.1

**Description** Set of functions to polish content for Microsoft 'Word' and 'PowerPoint' into 'OOXML'. Polishing is the conversion of the R object into an 'OOXML' representation of the object that can then be added to 'Word' or 'PowerPoint' files.

**License** Apache License ( $\geq 2$ )

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**URL** <https://gsk-biostatistics.github.io/polish/>,  
<https://github.com/GSK-Biostatistics/polish/>

**BugReports** <https://github.com/GSK-Biostatistics/polish/issues/>

**Depends** R ( $\geq 4.1.0$ )

**Imports** cli, commonmark, dplyr, flextable, ggplot2, glue, gt,  
htmltools, magick, officer ( $\geq 0.6.9$ ), purrr, rlang, rmarkdown,  
sloop, tibble, webshot2, xml2

**Suggests** testthat ( $\geq 3.0.0$ ), withr

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** Romain François [aut],  
Ellis Hughes [aut, cre],  
Shannon Haughton [aut],  
GlaxoSmithKline Research & Development Limited [cph, fnd]

**Maintainer** Ellis Hughes <ellis.h.hughes@gsk.com>

**Repository** CRAN

**Date/Publication** 2026-06-03 14:10:12 UTC

## Contents

as_file . . . . .	2
as_is . . . . .	3
as_md . . . . .	3
as_xml_nodese . . . . .	4
polish_check_dots_empty . . . . .	5
polish_content . . . . .	6
polish_content_pptx.flextable . . . . .	7
polish_content_word.flextable . . . . .	9
show_xml . . . . .	11
sp_shell . . . . .	12
xml_nodese_ns_spec . . . . .	12
<b>Index</b>	<b>13</b>

---

as_file	<i>Mark as a file by adding classes</i>
---------	---

---

### Description

Mark as a file by adding classes

### Usage

```
as_file(path)
```

### Arguments

path	file path
------	-----------

### Value

path with classes set to "file" and "file\_{ext}" where ext is the file extension

### Examples

```
tf <- tempfile(fileext = ".txt")
writeLines("", tf)

as_file(tf)
```

---

as_is	<i>Mark as a node to inject as is</i>
-------	---------------------------------------

---

**Description**

Mark as a node to inject as is

**Usage**

```
as_is(x, error_call = current_env())
```

**Arguments**

x	raw ooxml passed through <a href="#">as_xml_node()</a>
error_call	The execution environment of a currently running function, e.g. <code>caller_env()</code> . The function will be mentioned in error messages as the source of the error. See the call argument of <a href="#">abort()</a> for more information.

**Value**

adds the class "as\_is" to the input text the content "as is" and not modify it when polishing

---

as_md	<i>Mark text as markdown</i>
-------	------------------------------

---

**Description**

Mark text as markdown

**Usage**

```
as_md(x)
```

**Arguments**

x	text to be interpreted as markdown
---	------------------------------------

**Value**

adds the class "as\_ms" to the input text the content "as markdown" apply markdown formatting to the text when polishing

---

as_xml_nodese	<i>Convert to xml nodes</i>
---------------	-----------------------------

---

**Description**

Convert to xml nodes

**Usage**

```
as_xml_nodese(  
  x,  
  ns = c("word", "pptx"),  
  ...,  
  .envir = parent.frame(),  
  .open = "{",  
  .close = "}",  
  error_call = current_env()  
)
```

```
as_xml_node(  
  x,  
  ...,  
  .envir = parent.frame(),  
  .open = "{",  
  .close = "}",  
  error_call = current_env()  
)
```

```
as_xml_node_word(  
  x,  
  ...,  
  .envir = parent.frame(),  
  .open = "{",  
  .close = "}",  
  error_call = current_env()  
)
```

```
as_xml_node_pptx(  
  x,  
  ...,  
  .envir = parent.frame(),  
  .open = "{",  
  .close = "}",  
  error_call = current_env()  
)
```

```
as_xml_word(  
  x,  
  ...,  
  .envir = parent.frame(),  
  .open = "{",  
  .close = "}",  
  error_call = current_env()  
)
```

```

    x,
    ...,
    .envir = parent.frame(),
    .open = "{",
    .close = "}",
    error_call = current_env()
)

as_xml_pptx(
  x,
  ...,
  .envir = parent.frame(),
  .open = "{",
  .close = "}",
  error_call = current_env()
)

```

### Arguments

<code>x</code>	an object to convert to xml nodes
<code>ns</code>	word or pptx
<code>...</code>	These dots are for future extensions and must be empty.
<code>.envir</code> , <code>.open</code> , <code>.close</code>	forwarded to <a href="#">glue::glue()</a>
<code>error_call</code>	The execution environment of a currently running function, e.g. <code>caller_env()</code> . The function will be mentioned in error messages as the source of the error. See the <code>call</code> argument of <a href="#">abort()</a> for more information.

### Value

an `xml_nodeset`

---

`polish_check_dots_empty`

*Check ... empty*

---

### Description

Wrapper around [rlang::check\\_dots\\_empty\(\)](#) that only does the check if the `polish.dots_error` is set to `TRUE`

### Usage

```
polish_check_dots_empty(env = caller_env(), error = NULL, call = caller_env())
```

**Arguments**

env	Environment in which to look for . . .
error	An optional error handler passed to <code>try_fetch()</code> . Use this e.g. to demote an error into a warning.
call	The execution environment of a currently running function, e.g. <code>caller_env()</code> . The function will be mentioned in error messages as the source of the error. See the <code>call</code> argument of <code>abort()</code> for more information.

---

polish_content	<i>Polish content</i>
----------------	-----------------------

---

**Description**

Polish content

**Usage**

```
polish_content(x, type, ..., error_call = current_env(), error_class = NULL)
```

**Arguments**

x	object to be polished into ooxml
type	output type. Valid values are "word" and "pptx".
...	additional arguments for methods
error_call	The execution environment of a currently running function, e.g. <code>caller_env()</code> . The function will be mentioned in error messages as the source of the error. See the <code>call</code> argument of <code>abort()</code> for more information.
error_class	Additional error class.

**Details**

depending on type either the `polish_content_pptx()` or `polish_content_word()` function is called.

**Value**

an object of class "polish\_xml\_nodeset". Contains XML that is viable Word or PowerPoint OOXML for addition to Word documents or PowerPoint presentations based on the "type".

---

polish\_content\_pptx.flextable  
*Polish content for Powerpoint*

---

## Description

Polish content for Powerpoint

## Usage

```
## S3 method for class 'flextable'
polish_content_pptx(
  x,
  ph = "<p:ph/>",
  ...,
  full_width = FALSE,
  page_size = NULL,
  table_style = NULL,
  pptx = NULL,
  error_call = current_env()
)

## S3 method for class 'ggplot'
polish_content_pptx(
  x,
  ph = "<p:ph/>",
  pptx,
  ...,
  height = NULL,
  width = NULL,
  res = 300,
  device = c("png", "svg", "jpeg"),
  scale = 1,
  error_call = current_env()
)

polish_content_pptx(x, ph = "<p:ph/>", pptx, ..., error_call = current_env())

## S3 method for class 'character'
polish_content_pptx(
  x,
  ph = "<p:ph/>",
  pptx,
  ...,
  escape = TRUE,
  collapse = FALSE,
  font_color = NULL,
```

```

    font_style = NULL,
    font_size = NULL,
    font_typeface = NULL,
    error_call = current_env()
)

## S3 method for class 'list'
polish_content_pptx(
  x,
  ph = "<p:ph/>",
  pptx,
  ...,
  list_type = c("none", "unordered", "ordered"),
  error_call = current_env()
)

## S3 method for class 'file_png'
polish_content_pptx(
  x,
  ph = "<p:ph/>",
  pptx,
  ...,
  height = NULL,
  width = NULL,
  units = "in",
  image_fit = c("stretch", "scale"),
  error_call = current_env()
)

```

### Arguments

x	object to be polished
ph	placeholder
...	These dots are for future extensions and must be empty.
full_width	Logical, used to fit full size of given width
page_size	Number of rows to accommodate in a slide
table_style	Named list with two entries: style and options. style is the name of table style to apply as a character, options a character vector of options to apply from the table style, including: "Banded Columns", "Banded Rows", "First Column", "Header Row", "Last Column", and "Total Row"
pptx	The Powerpoint Presentation object
error_call	The execution environment of a currently running function, e.g. caller_env(). The function will be mentioned in error messages as the source of the error. See the call argument of <a href="#">abort()</a> for more information.
width, height	defaults to match the height and width of the placeholder provided. When that cannot be derived, it assumes the plot should be 5 inches tall and 6 inches wide. Users may set the width and height, but they must both be set.

res	Plot resolution. Also accepts a string input: "retina" (320), "print" (300), or "screen" (72). Applies only to raster output types.
device	what graphic device to use when saving the plot. Can be one of "png", "svg", or "jpeg".
scale	Multiplicative scaling factor to use when saving the plot. See <a href="#">ggplot2::ggsave</a> .
escape	Should x be escaped with <a href="#">htmltools::htmlEscape()</a>
collapse	If TRUE a <code>&lt;a:br w:type="textWrapping" w:clear="all"&gt;</code> element is added between the <code>&lt;a:r&gt;</code> nodes
font_color	color of text. Enter as a hex code.
font_style	styling of text, vector including "bold", "italic", "underline".
font_size	point size of text if different from document default.
font_typeface	typeface of the text to be added if different from document default.
list_type	How to print the list. Options are "none", "unordered" and "ordered". <ul style="list-style-type: none"> <li>• "none" (default) puts contents on new lines and indents lists accordingly.</li> <li>• "unordered" makes the content bulleted list with intents.</li> <li>• "ordered" makes a numbered list with intents.</li> </ul>
units	see <a href="#">officer::external_img</a>
image_fit	Should the image be distorted to match the dimensions of the placeholder, or scaled up/down and keep dimension ratio (scale). Default is "stretch".

**Value**

a "polish\_xml\_nodeset" of viable PowerPoint OOXML for addition to PowerPoint presentations.

---

polish\_content\_word.flextable

*Polish content for Word*

---

**Description**

Polish content for Word

**Usage**

```
## S3 method for class 'flextable'
polish_content_word(
  x,
  inline = TRUE,
  ...,
  align = NULL,
  split = NULL,
  topcaption = TRUE,
  keepnext = NULL,
```

```

    error_call = current_env()
  )

## S3 method for class 'ggplot'
polish_content_word(
  x,
  inline = FALSE,
  ...,
  height = 5,
  width = 6,
  units = "in",
  res = 300,
  error_call = current_env()
)

polish_content_word(x, inline = FALSE, ..., error_call = current_env())

## S3 method for class 'character'
polish_content_word(
  x,
  inline = FALSE,
  ...,
  font_color = NULL,
  font_style = NULL,
  font_size = NULL,
  font_typeface = NULL,
  error_call = current_env()
)

## S3 method for class 'file_png'
polish_content_word(
  x,
  inline = FALSE,
  ...,
  height = 5,
  width = 6,
  units = "in",
  error_call = current_env()
)

```

### Arguments

<code>x</code>	object to be polished
<code>inline</code>	if FALSE the comments are wrapped in a <w:p>
<code>...</code>	used in some methods
<code>align</code>	alignment of table and caption
<code>split</code>	should table be allowed to split

topcaption	should caption be above or below the table
keepnext	should content be marked for keep next.
error_call	The execution environment of a currently running function, e.g. caller_env(). The function will be mentioned in error messages as the source of the error. See the call argument of <code>abort()</code> for more information.
width, height, units	Plot size in units ("in", "cm", "mm", or "px"). defaults to 5 inches tall and 6 inches wide.
res	Plot resolution. Also accepts a string input: "retina" (320), "print" (300), or "screen" (72). Applies only to raster output types.
font_color	color of text. Enter as a hex code.
font_style	styling of text, vector including "bold", "italic", "underline".
font_size	point size of text if different from document default.
font_typeface	typeface of the text to be added if different from document default.

**Value**

an object of class "polish\_xml\_nodeset". Contains XML that is viable Word OOXML for addition to Word documents.

---

show_xml	<i>Show the xml of the polished version of a display in the browser</i>
----------	---

---

**Description**

Show the xml of the polished version of a display in the browser

**Usage**

```
show_xml(xml, wrap = TRUE)
```

**Arguments**

xml	an xml node or nodeset
wrap	if TRUE, the xml is wrapped in a w:body node

**Details**

This opens the xml in the browser, so that you can do View source and inspect the structure of the xml

**Value**

used for side effect of previewing xml in a browser. Helpful for inspecting the OOXML generated by this package.

---

sp_shell	<i>Wrap the placeholder into a &lt;p:sp&gt; shell</i>
----------	---

---

**Description**

Wrap the placeholder into a <p:sp> shell

**Usage**

```
sp_shell(ph, error_call = current_env())
```

**Arguments**

ph	placeholder
error_call	The execution environment of a currently running function, e.g. <code>caller_env()</code> . The function will be mentioned in error messages as the source of the error. See the <code>call</code> argument of <code>abort()</code> for more information.

**Value**

an object of class "polish\_xml\_nodeset". Contains XML that is viable PowerPoint OOXML for addition to PowerPoint documents.

---

xml_nodeset_ns_spec	<i>xml namespaces</i>
---------------------	-----------------------

---

**Description**

xml namespaces

**Usage**

```
xml_nodeset_ns_spec(ns = c("word", "pptx"), error_call = caller_env())
```

**Arguments**

ns	word or pptx
error_call	The execution environment of a currently running function, e.g. <code>caller_env()</code> . The function will be mentioned in error messages as the source of the error. See the <code>call</code> argument of <code>abort()</code> for more information.

**Value**

xml namespaces for word or pptx

# Index

`abort()`, [3](#), [5](#), [6](#), [8](#), [11](#), [12](#)  
`as_file`, [2](#)  
`as_is`, [3](#)  
`as_md`, [3](#)  
`as_xml_node` (`as_xml_nodeset`), [4](#)  
`as_xml_node()`, [3](#)  
`as_xml_node_pptx` (`as_xml_nodeset`), [4](#)  
`as_xml_node_word` (`as_xml_nodeset`), [4](#)  
`as_xml_nodeset`, [4](#)  
`as_xml_pptx` (`as_xml_nodeset`), [4](#)  
`as_xml_word` (`as_xml_nodeset`), [4](#)

`ggplot2::ggsave`, [9](#)  
`glue::glue()`, [5](#)

`htmltools::htmlEscape()`, [9](#)

`officer::external_img`, [9](#)

`polish_check_dots_empty`, [5](#)  
`polish_content`, [6](#)  
`polish_content_pptx`  
    (`polish_content_pptx.flextable`),  
    [7](#)  
`polish_content_pptx()`, [6](#)  
`polish_content_pptx.flextable`, [7](#)  
`polish_content_word`  
    (`polish_content_word.flextable`),  
    [9](#)  
`polish_content_word()`, [6](#)  
`polish_content_word.flextable`, [9](#)

`rlang::check_dots_empty()`, [5](#)

`show_xml`, [11](#)  
`sp_shell`, [12](#)

`try_fetch()`, [6](#)

`xml_nodeset_ns_spec`, [12](#)