

# A presentation with Polylux via Quarto using **projector**

Author One  
Harvard University

Author Two  
Yale University

April 8, 2026

# Table of contents

A Presentation Demo

Features

Callouts

Typst Specialties

Styling your slides

# A Presentation Demo

# Introduction

`projector` is a Quarto extension to create Polylux presentations in Typst while only writing Markdown.

It is designed to mimic all of the properties of beamer so that you can (hopefully) convert just by:

1. Changing the format to `projector-typst`
2. Changing `incremental` to `bullet-incremental`
3. Setting a theme with a Typst file

This Quarto extension is written by Christopher T. Kenny.

# More Information

This demo explains the template. For more information, see the readme.

Please report any issues or make suggestions at <https://github.com/christopherkenny/projector/issues>.

Look at some theme options at: <https://github.com/christopherkenny/projector-themes>

# Features

# Incremental Lists

- First item
- 
-

# Incremental Lists

- First item
- Second item
-

# Incremental Lists

- First item
- Second item
- Third item

# Non-Incremental

- one-th
- two-th
- three-th

# Numbered lists

1. A one
- 2.

# Numbered lists

1. A one
2. And a twoooo

# Non-Incremented Numbered lists

1. And 3 (really 1) but static
2. And 4 (really 2)

# And manual pauses

Sometimes you write text

# And manual pauses

Sometimes you write text  
that needs time to be ready

An untitled slide?

Yes! It's possible with ---.

# Split slides

left column in 40% column center aligned

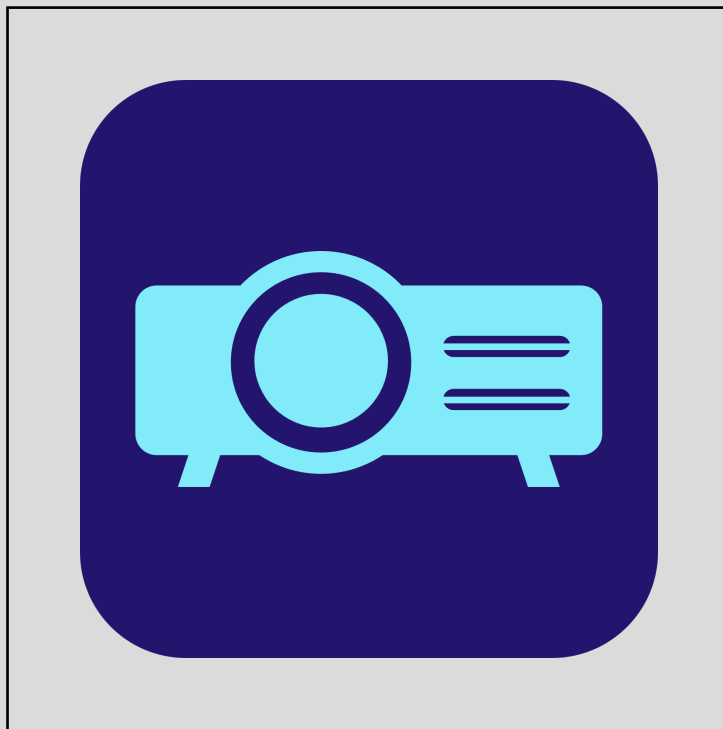
right column with 60% column right aligned

# Split slides, fixed width and centered

contents...

contents...

# Images



# Code

Include code with triple backticks:

```
library(dplyr)

starwars %>%
  filter(species == "Droid")
```

# Adding speaker notes

Polylux uses pdfpc to store notes. Speaker notes are temporarily disabled pending upstream fixes for Typst 0.14. You can create them via

```
 :::{.notes}
```

```
Adding this will create notes readable to the presenter in pdfpc.
```

```
 :::
```

# Callouts

# Callouts: Mapped from Beamer

We can use the heading values to control callouts, like so:

 Caution!

This is an important warning.

 Try This

This demonstrates something useful.

Note

Plain information.

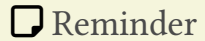
# Callouts: Custom to this template

And here's a little extra flair



Tip

A tip for the viewer



Reminder

A thing to remember



Warning!

For a little less caution than an alert

# Typst Specialties

# Using Typst functions

This template comes pre-equipped with a Quarto extension that converts divs and spans to function calls. To use any Typst functions, list them under the `functions` key in the YAML header.

```
functions: align
```

Then you can use them like:

```
::: {.align arguments="right"}  
this  
:::
```

or

```
[other text]{.align arguments="right"}
```

For full documentation of this feature, see the [typst-function](#) extension page.

# Styling your slides

# YAML-controlled features

Some features can be controlled via yaml options.

- `mainfont`: sets font (see options with `quarto typst fonts`)
- `margin`: sets page margins
- `papersize`: the paper size to use (choices listed here)
- `toc`: whether to display the table of contents
- `toc_title`: title of the table of contents
- `background-image`: the path to an image to put as the background
- `handout`: display as a handout, removing incrementals

# Template files

For additional customization, you can supply a template style file to use.

This must contain a function `projector-theme` that contains your styling choices. That file will be imported before any of the YAML options, so the YAML options take precedence.

This should look like

```
#let projector-theme(doc) = {  
  // your styling choices  
  doc  
}
```

Omitting `doc` at the end will create a single, empty slide as your output.

To supply a template, add `theme: path_to_file.typ` to the YAML.

# Controlling Title, ToC, and Sections Slides

To modify these three particular slides, you need to adjust them in your template file. Simply redefine the functions that produce them with your own (pretty) version.

The function signatures should be as follows:

- `title-slide(title, subtitle, authors, date)`
- `toc-slide(toc_title)`
- `section-slide(name)`

You do not need to use the arguments in the functions, but they must take those arguments or it will break.