

Typst Render

Quarto Extension

Mickaël CANOUIL, *Ph.D.*

A Quarto filter extension that compiles ````{typst}` code blocks to images (PNG, SVG, or PDF) using the Typst binary bundled with Quarto. This makes Typst diagrams, figures, tables, and equations usable across all output formats (HTML, PDF via LaTeX, DOCX, RevealJS, and more).

When the output format is Typst, blocks pass through natively without image conversion.

Installation

```
quarto add mcanouil/quarto-typst-render@0.5.0
```

This will install the extension under the `_extensions` subdirectory.

If you are using version control, you will want to check in this directory.

Usage

To use the extension, add the following to your document's front matter:

```
filters:  
  - typst-render
```

For cross-referencing support, use timing control:

```
filters:  
  - path: typst-render  
    at: pre-quarto
```

Then write Typst code blocks in your document:

```
```${typst}  
#set text(size: 16pt)
Hello from *Typst*!
```
```

Basic Usage

A simple Typst block rendered as an image:

```
```{typst}
#set text(size: 16pt)
Hello from *Typst*!
This is rendered as an image in non-Typst outputs.
```
```

Hello from **Typst**! This is rendered as an image in non-Typst outputs.

Per-Block Options

Use comment+pipe syntax (`/// key: value) at the top of the code block to set per-block options.`

Page Dimensions

Control the width, height, and margin of the rendered image page. These options set the Typst page dimensions for image compilation (non-Typst output formats). In native Typst output, `width` and `height` are ignored; `margin` is applied as block inset.

```
```{typst}
/// width: 12cm
/// height: auto
/// margin: 1em
#set text(size: 14pt, fill: eastern)
#align(center)[
 #box(
 fill: luma(240),
 inset: 1em,
 radius: 4pt,
```

## Cross-Referencing

Use `/// label:` and `/// cap:` for Quarto cross-references. Generic `cap` and `alt` options work with any label prefix. Prefix-specific variants (e.g., `fig-cap`, `tbl-alt`) override the generic options when the label matches. Any Quarto cross-reference type is supported (`fig-`, `tbl-`, `lst-`, etc.), including custom types.

See Figure 1 for a captioned figure.

```
```{typst}
/// label: fig-typst-shapes
/// cap: "A simple Typst diagram with shapes."
/// alt: "A blue circle and orange rectangle arranged horizontally."
#set text(size: 12pt)
#align(center)[
  #stack(dir: ltr, spacing: 1em)[
    #circle(radius: 1cm, fill: blue.lighten(60%))
  ][
    #rect(width: 2cm, height: 2cm, fill: orange.lighten(60%))
  ]
]
```

Echo and Eval Control

The `echo` and `eval` options control whether source code is displayed and whether the block is compiled.

Echo: Source Code Alongside Output

Show the source code followed by the rendered output with `echo: true`:

```
#set text(size: 14pt)
$ sum_(k=0)^n k = (n(n+1))/2 $
```

$$\sum_{k=0}^n k = \frac{n(n+1)}{2}$$

Echo: Fenced Source Code

Use `echo: fenced` to display source code wrapped in fenced code block markers (````{typst}`), including any `comment+pipe` options (except `echo` itself). This mirrors Quarto's native `echo: fenced` behaviour for computational cells.

```
```{typst}
```

Output Location (Reveal.js)

## Input Variables

Pass key-value pairs to Typst code via `sys.inputs`. Global inputs are set in YAML; per-block inputs override them using comma-separated syntax.

```
```{typst}
//| input: greeting=Hello,subject=World
#set text(size: 16pt)
#sys.inputs.at("greeting"), #sys.inputs.at("subject")!
```
```

Hello, World!

## Multi-Page Output

When Typst produces multiple pages, all pages are included by default. Use `pages` to select specific pages and `layout-ncol` to arrange them in columns.

### All Pages Side by Side

```
```{typst}
//| layout-ncol: 2
//| width: 8cm
//| height: 6cm
#set text(size: 14pt)
#align(center + horizon)[
  *Page 1*: First page content.
]
#pagebreak()
#align(center + horizon)[
  *Page 2*: Second page content.
]
```
```

## External File Rendering

Render an external `.typ` file instead of inline code:

```
`` `{typst}
//| file: _extensions/typst-render/_example.typ
#set text(size: 14pt)
#align(center)[
 This content was loaded from an external `.typ` file.
]
...`
```

This content was loaded from an external `.typ` file.

# Preamble

Typst code can be prepended before each block using the `preamble` option. This is useful for setting global styles or importing packages.

## Inline Preamble

```
extensions:
 typst-render:
 preamble: '#set text(font: "Libertinus Serif")'
```

## File-Based Preamble

Any value ending in `.typ` is treated as a file path:

```
extensions:
 typst-render:
 preamble: "preamble.typ"
```

## Social Card

A repository social card built entirely in Typst, rendered as a high-resolution PNG.

```
```{typst}
//| label: fig-social-card
//| cap: "Social card for quarto-typst-render."
//| format: png
//| dpi: 288
//| width: 1280pt
//| height: 640pt
//| margin: 0pt

#let gradient-bg = gradient.linear(
  rgb("#1a1a2e"),
  rgb("#16213e"),
  rgb("#0f3460"),
  angle: 135deg,
)
```

Global Configuration

Configure the filter globally in your document YAML. Most options can be set globally and overridden per block using comment+pipe syntax (`///| key: value`). See [Global-Only Options](#) for options that cannot be overridden per block.

```
extensions:  
  typst-render:  
    dpi: "288"  
    margin: "1em"  
    format: png  
    cache: true
```

Options

Option	Type	De- fault	Description
<code>format</code>	string	(auto)	Image format: png, svg, pdf.
<code>dpi</code>	string	"144"	Pixels per inch (PNG only).
<code>width</code>	string	"auto"	Page width for image compilation (ignored in Typst)