A quarto document with mermaid diagrams

Georg W Otto

07/11/2025

Summary

This is a test where I try out the many and somehow confusing options how to integrate mermaid diagrams in documents rendered from a quarto file. The results depend a to a large degree on the output format: `html` is the most flexible format showing the output as intended, `pdf` does not (in my setting) render the mermaid node definition syntax introduced in `Mermaid 11.3.0`, and github flavoured `markdown` does not render mermaid code and does not support automated references to figures.

Rendering mermaid diagrams from code

Rendering quarto documents with mermaid diagrams directly from code can be done either by including the code directly in the quarto document or by loading it from external .mmd files.

To run mermaid code blocks directly in the document, the code blocks need to have eval and include options set to true.

````
```{mermaid}
%%| eval: true
%%| include: true
%%| label: fig-test_1
%%| fig-cap: "Test 1"

%% mermaid code here

```
````

To load mermaid diagrams from external .mmd files and render them in the html output, the code blocks need to specify the file name and have eval and include options set to true.

````
```{mermaid}
%%| eval: true
%%| include: true
````
````

```
%%| label: fig-qc_preimputation_ncds
%%| fig-cap: "NCDS, Pre-imputation workflow"
%%| file: qc-preimputation-ncds.mmd
```

For quarto to render mermaid code, it needs a configuration html file to include the mermaid js library in the html header.

```
<script type="module">
  import mermaid from "https://cdn.jsdelivr.net/npm/mermaid@11.9.0/dist/mermaid.esm.min.mjs'
  document.addEventListener("DOMContentLoaded", () => {
    mermaid.initialize({ startOnLoad: true });
  });
</script>
```

To load the html configuration file in quarto, specify it in the document yaml header like this:

```
format:
  html:
    include-in-header: config-mermaid.html
```

Rendered mermaid code in html output sometimes shows a strange behaviour. The resultin html file in the location where it is created shows an error message instead of the rendered diagram, while the same html file opened from a different directory shows the diagram correctly. This seems to be related to the relative paths used by mermaid js library to load resources.

Example Flowcharts

The diagrams in Figure 1 and Figure 2 use the **original Mermaid node syntax**. Figure 1 shows a simple flowchart defined directly in the quarto document, while Figure 2 loads the diagram from an external .mmd file.

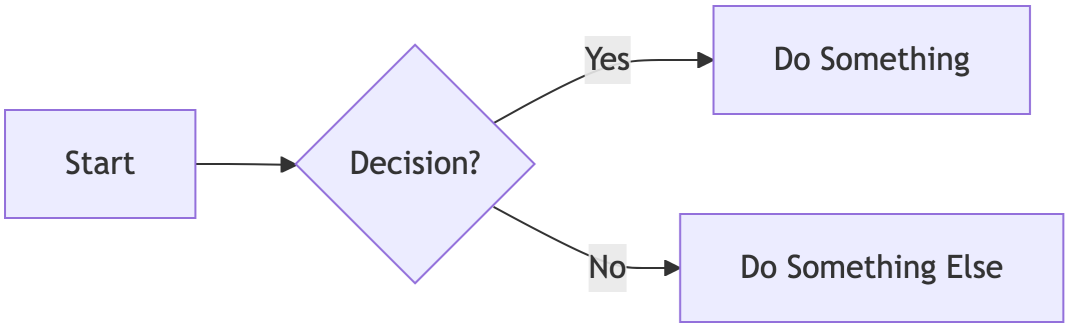

Figure 1: Diagram rendered from mermaid code in code block.

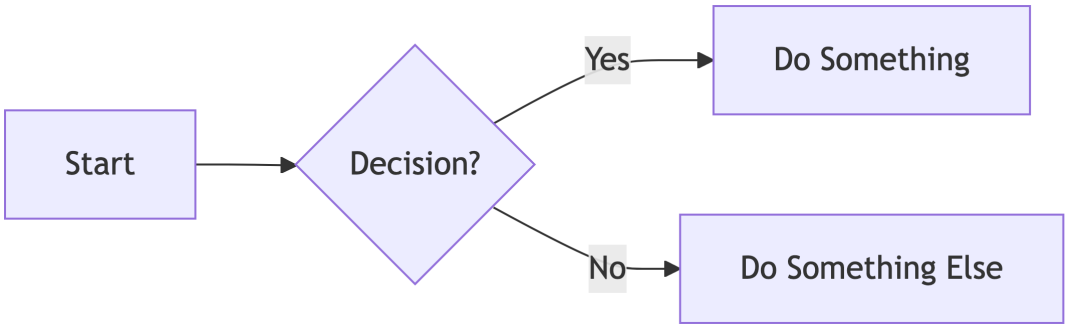

Figure 2: Diagram rendered from mermaid code in imported file.

The diagrams in Figure 3 and Figure 4 use the **Mermaid 11.3.0+ node syntax**. The diagram in Figure 3 is defined directly in the quarto document, while Figure 4 loads the diagram from an external .mmd file. This code gives an error with pdf output, but works with html output.

Including mermaid diagrams from rendered files

Mermaid diagrams can also be included from rendered files in the results/figures folder, e.g. if there are rendering problems for pdf output. This puts the imported figures in a window with zoom and pan functionality in both html and pdf output. The document also looks cleaner.

The default format for graphics in quarto is svg for html output and pdf for pdf output. To include mermaid diagrams from a pdf file into html output, one has to set the default image extension for html output to pdf in the document yaml header like this:

```
format:
  html:
    default-image-extension: pdf
```

The diagram in Figure 5 loads a pdf file with a mermaid diagram generated with the old mermaid node syntax and the diagram in Figure 6 loads a pdf file with a mermaid diagram generated with the Mermaid 11.3.0+ node syntax.

Figure 3: Diagram rendered from mermaid code in code block.

Figure 4: Diagram rendered from mermaid code in imported file.

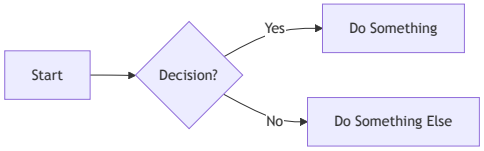

Figure 5: The same diagram as in Figures 1 and 2, but imported from a pdf file

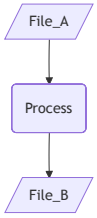

Figure 6: The same diagram as in Figures 3 and 4, but imported from a pdf file