

The `ltmermaid` package

安藤 遼哉 (Ryoya Ando)
<https://ryoya9826.github.io/>

April 20, 2026

Abstract

The `ltmermaid` package renders Mermaid diagrams at $\text{\LaTeX} 2_{\epsilon}$ compile time when using **LuaLaTeX** and embeds them in the PDF. Rendering uses external tools such as the Mermaid CLI, so **shell escape** (`-shell-escape`) is required. The machine that runs `lualatex` must have a working **Node.js** installation, `mmdc` (or `npx`), and headless Chromium for the CLI.

1 Requirements

Item	Requirement
\TeX engine	LuaLaTeX (<code>lualatex</code>).
Format	$\text{\LaTeX} 2_{\epsilon}$.
Shell escape	<code>lualatex -shell-escape</code> (the package invokes the CLI via <code>os.execute</code>).
External tool	Mermaid CLI (<code>mmdc</code>).

If the Mermaid CLI is missing from the environment, `ltmermaid` does **not** auto-install `mmdc` or similar. A typical setup uses **Node.js** with `npm` (or `npx`). If you pass a command that includes `npx -y` via the package option `Renderer`—for example `npx -y @mermaid-js/mermaid-cli ...`—then `npx` may fetch and install packages as needed (**network access** may be required).

When you run `lualatex`, unless the `Renderer` option points to another command, the default renderer `mmdc` must be visible in the same environment (if you set `Renderer`, that command must be visible instead). The Mermaid CLI depends on headless Chromium (Puppeteer); see the Mermaid CLI documentation for details.

2 License and source

Distributed under the \LaTeX Project Public License (LPPL), version 1.3c or later.

Source and issues: <https://github.com/ryoya9826/ltMermaid>

3 Usage

3.1 Minimal document

```
1 \documentclass{article}
2 \usepackage{ltmermaid}
3
4 \begin{document}
5 \begin{mermaid}
6 flowchart LR
7   A --> B
8 \end{mermaid}
9 \end{document}
```

Compile with:

```
1 lualatex -shell-escape yourfile.tex
```

3.2 Layout adjustments (optional)

By default, included graphics are scaled with `adjustbox`. Example:

```
1 \MermaidAdjustBoxOpts{max width=0.8\linewidth,center}
2 \MermaidAdjustBoxOpts{max width=0.9\linewidth,center,valign=T}
```

3.3 Beamer

The package works with the `beamer` class. Any frame that contains a `mermaid` environment must use the `fragile` option (the environment relies on `fancyvrb`), for example `\begin{frame}[fragile]{Diagram}`.

```
1 \documentclass{beamer}
2 \usepackage{ltmermaid}
3
4 \begin{document}
5 \begin{frame}[fragile]{Mermaid}
6 \begin{mermaid}
7 flowchart LR
8   A --> B
9 \end{mermaid}
10 \end{frame}
11 \end{document}
```

4 Package options (optional)

- **Renderer:** Prefix for the renderer command. If omitted, the default is `mmdc`. You can set it explicitly, for example `npx -y @mermaid-js/mermaid-cli`.

5 User macros

- `\MermaidRenderOptions{...}`: Extra CLI arguments before `-i` / `-o` (merged after the built-in `-f` when PDF fit is enabled).
- `\MermaidNoPdfFit`: Disables `-f` / `--pdfFit` for `mmdc` (by default this is *on*).
- `\MermaidAdjustBoxOpts{...}`: Full `adjustbox` key list around `\includegraphics` (default `max width=0.9\linewidth,center`).
- `\MermaidGraphicsOpts{...}`: Extra keys for `\includegraphics` (e.g. `rotation, trim`); width is usually set with `\MermaidAdjustBoxOpts`.

6 Output files

Intermediate `.mmd` and `.pdf` files are written under the `mermaid/` directory relative to the directory where compilation runs. With `-output-directory`, they are written under `mermaid/` inside the directory indicated by the environment variable `TEXMF_OUTPUT_DIRECTORY`.

7 Examples

Left: source for the `mermaid` environment as typed. **Right:** rendered diagram.

```

1 \begin{mermaid}
2 flowchart TB
3   subgraph client["Client tier"]
4     WEB["Browser / SPA"]
5     CLI["CLI / batch"]
6   end
7   subgraph edge["Edge"]
8     GW{{API Gateway}}
9   end
10  subgraph svc["Services"]
11    AUTH["Auth"]
12    API["Business API"]
13    WORK["Workers"]
14  end
15  subgraph store["Data"]
16    DB[("PostgreSQL")]
17    CACHE[("Redis")]
18    QUEUE["Job queue"]
19  end
20  WEB --> GW
21  CLI --> GW
22  GW --> AUTH
23  GW --> API
24  API --> WORK
25  API --> DB
26  API --> CACHE
27  WORK --> QUEUE
28  WORK --> DB
29 \end{mermaid}

```

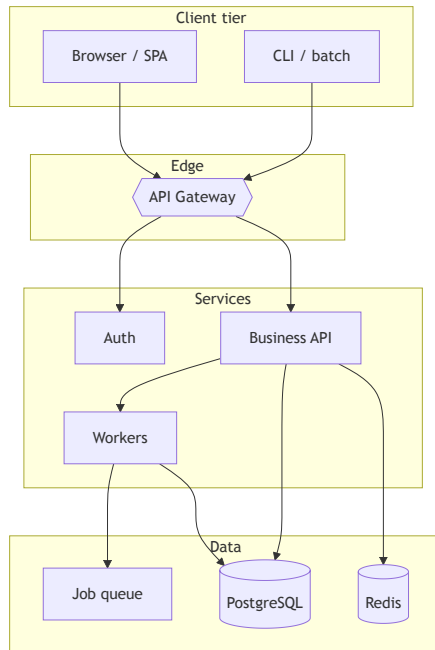


Figure 1: Layered architecture (subgraphs and node shapes)

```

1 \begin{mermaid}
2 sequenceDiagram
3   autonumber
4   actor U as User
5   participant B as Browser
6   participant A as Auth API
7   participant S as Business API
8   participant D as DB
9   U->>B: Log in
10  B->>A: POST /token
11  A->>D: Verify user
12  D-->>A: Row
13  A-->>B: JWT
14  B->>S: GET /orders ( Bearer)
15  S->>A: Validate token
16  A-->>S: Claims
17  S->>D: SELECT
18  D-->>S: Rows
19  S-->>B: 200 JSON
20  B-->>U: Show list
21 \end{mermaid}

```

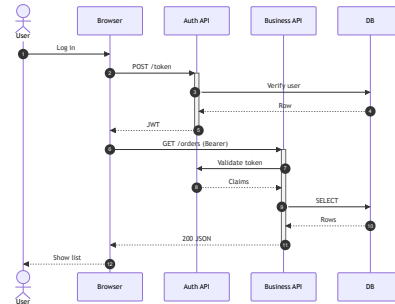


Figure 2: Sequence diagram (autonumber, async arrows, multiple participants)

```

1 \begin{mermaid}
2 stateDiagram-v2
3   [*] --> Draft: Create
4   Draft --> Review: Submit
5   Review --> Draft: Send
6     back
7   Review --> Approved:
8     Approve
9   Approved --> Published:
10    Publish
11   Published --> Archived:
12    Close
13   Review --> Rejected:
14    Reject
15   Rejected --> [*]
16   Archived --> [*]
17 \end{mermaid}

```

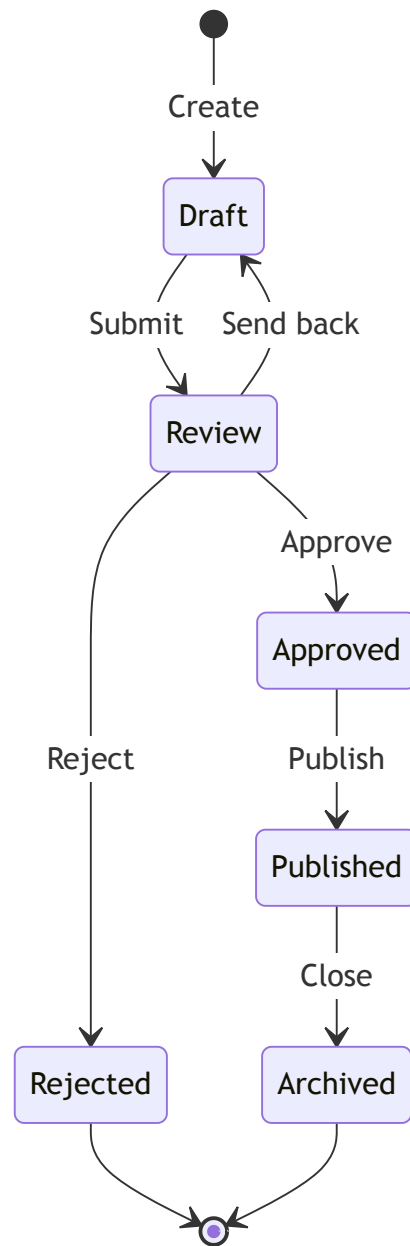


Figure 3: State machine (`stateDiagram-v2`)

8 Version history

Version 1.0a (2026-04-20) Renamed sample documents.

Version 1.0 (2026-04-16) Stable release.

Version 0.2 (2026-04-13) Removed MERMAID_MMDC and MERMAID_MMDC_OPTIONS.

Version 0.1 (2026-04-08) Initial release.