

# mermaid パッケージ (mermaid.sty)

安藤 遼哉 (Ryoya Ando)

<https://ryoya9826.github.io/>

2026 年 4 月 20 日

## 概要

mermaid.sty は shellesc や \write18 を介して Mermaid CLI を起動し、 $\text{\LaTeX} 2_{\epsilon}$  におけるコンパイル時に Mermaid 図をレンダリングして PDF として取り込むパッケージです。up $\text{\LaTeX}$ , pdf $\text{\LaTeX}$ , Lua $\text{\LaTeX}$  などのエンジンで利用可能です。Mermaid 図のレンダリングには、Mermaid CLI (mmdc など) とヘッドレス Chromium (Puppeteer) が必要です。また、コンパイルの際は必ずシェルエスケープ (`-shell-escape`) を有効にしてください。

## 1 動作条件

項目	条件
フォーマット	$\text{\LaTeX} 2_{\epsilon}$
シェルエスケープ	<code>-shell-escape</code> (shellesc / \write 18 経由で CLI を起動)
外部ツール	Mermaid CLI (mmdc)

環境内に Mermaid CLI が存在しない場合でも、mermaid.sty が mmdc などを自動的にインストールすることはありません。Mermaid CLI を一般的な方法でインストールするには、**Node.js** と npm (または npx) を利用します。ただし、パッケージオプションの `Renderer` に `npx -y @mermaid-js/mermaid-cli` … のように `npx -y` を含むコマンドを指定した場合は、npx の仕様により、必要に応じてパッケージの**取得およびインストール**が動的に行われます (ネットワーク通信を伴います)。

コンパイルを実行する際、`Renderer` オプションで別のコマンドを指定しない限り、既定のレンダラである mmdc へのパスが通っており、実行可能である必要があります (別のコマンドを指定した場合も同様です)。なお、Mermaid CLI はヘッドレス Chromium (Puppeteer) に依存しています。詳細は Mermaid CLI の公式ドキュメントを参照してください。

ログは `\typeout` で [mermaid] プレフィックス付きで出力されます。トラブル時は mermaid/ 下の `.err` を確認してください。

## 2 ライセンスとソース

L<sup>A</sup>T<sub>E</sub>X Project Public License (LPPL) 1.3c 以降の下で配布されます。

ソースと issue: <https://github.com/ryoya9826/ltMermaid>

## 3 使い方

### 3.1 最小構成の文書

---

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

---

コンパイル例:

```
pdflatex -shell-escape yourfile.tex
```

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

```
uplax -shell-escape yourfile.tex && dvipdfmx yourfile.dvi
```

### 3.2 レイアウトの調整 (任意)

既定では `adjustbox` で取り込んだ図をスケールします。例:

---

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

---

### 3.3 Beamer

`beamer` クラスでも利用できます。 `mermaid` 環境を含む各フレームには `fragile` オプションが必要です (環境が `fancyvrb` に依存するため)。例: `\begin{frame}[fragile]{図}`。

---

```
1 \documentclass{beamer}
2 \usepackage{mermaid}
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 パッケージオプション (省略可能)

- `Renderer`: 使用するレンダラコマンド。省略時は `mmdc` が使用されます。 `npx -y @mermaid-js/mermaid-cli` のように、使用するコマンドを明示的に指定することも可能です。

## 5 ユーザー命令

- `\MermaidRendererOptions{...}`: `-i / -o` の前に付ける追加の CLI 引数 (PDF fit 有効時は組み込みの `-f` の後に結合)。
- `\MermaidNoPdfFit`: `mmdc` の `-f / --pdfFit` を無効化 (デフォルトでは **有効**)。
- `\MermaidAdjustBoxOpts{...}`: `\includegraphics` を囲む `adjustbox` のキー式 (デフォルトでは `max width=0.9\linewidth, center`)。
- `\MermaidGraphicsOpts{...}`: `\includegraphics` への追加キー (回転, `trim` など)。幅は通常 `\MermaidAdjustBoxOpts` で指定します。

## 6 出力ファイル

中間ファイルである `.mmd` と、レンダリング結果の `.pdf` は、コンパイルを実行したディレクトリ内の `mermaid/` フォルダに書き出されます。ファイル名には `mermaid-doc-ja-mermaid-1.mmd` や `mermaid-doc-ja-mermaid-1.pdf` のように、`mermaid-doc-ja` (ジョブ名) と通し番号が付与されます。

## 7 図の例

左に入力した `mermaid` 環境のソース、右にレンダリング結果を示します。

```

1 \begin{mermaid}
2 flowchart TB
3   subgraph clientクライアント層 [""]
4     WEBブラウザ [" / SPA"]
5     CLI["CLI / バッチ"]
6   end
7   subgraph edgeエッジ [""]
8     GW{{API Gateway}}
9   end
10  subgraph svcサービス層 [""]
11    AUTH認証 [""]
12    API業務 [" API"]
13    WORKワーカー [""]
14  end
15  subgraph storeデータ [""]
16    DB[("PostgreSQL")]
17    CACHE[("Redis")]
18    QUEUEジョブキュー [""]
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}

```

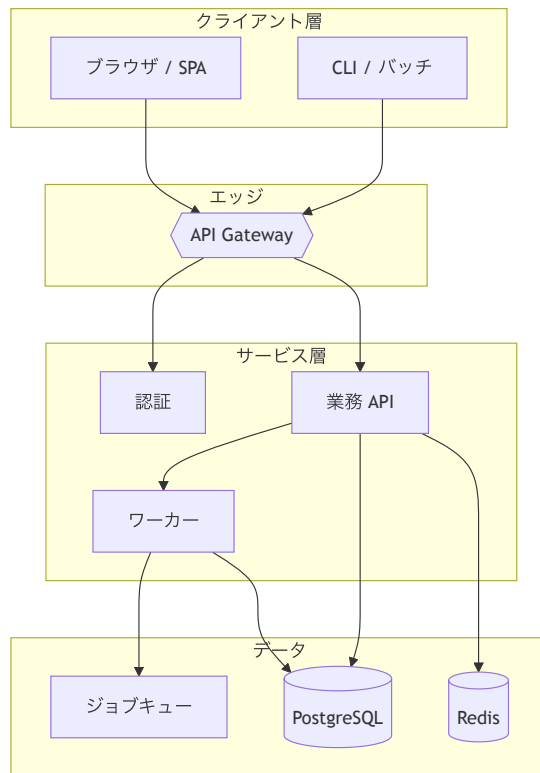


図1 多層アーキテクチャ (サブグラフ・各種ノード形)

```

1 \begin{mermaid}
2 sequenceDiagram
3   autonumber
4   actor U as 利用者
5   participant B as ブラウザ
6   participant A as 認証 API
7   participant S as 業務 API
8   participant D as DB
9   U->>B: ログイン
10  B->>A: POST /token
11  A->>D: ユーザー検証
12  D-->>A: レコード
13  A-->>B: JWT
14  B->>S: GET / () ordersBearer
15  S->>A: トークン検証
16  A-->>S: クレーム
17  S->>D: SELECT
18  D-->>S: 行集合
19  S-->>B: 200 JSON
20  B-->>U: 一覧表示
21 \end{mermaid}

```

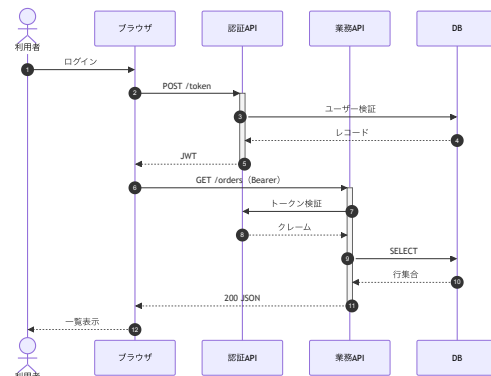


図2 シーケンス図 (番号・非同期矢印・複数参加者)

---

```

1 \begin{mermaid}
2 stateDiagram-v2
3   [*] --> Draft: 新規作成
4   Draft --> Review: 提出
5   Review --> Draft: 差し戻し
6   Review --> Approved: 承認
7   Approved --> Published: 公開
8   Published --> Archived: 終了
9   Review --> Rejected: 却下
10  Rejected --> [*]
11  Archived --> [*]
12 \end{mermaid}

```

---

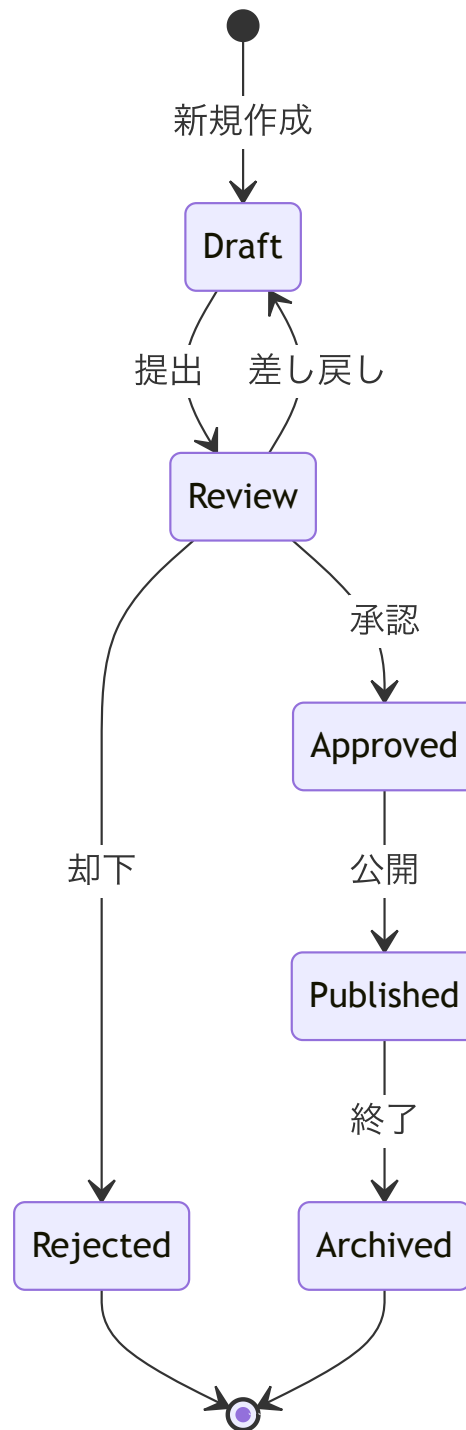


図3 状態遷移 (stateDiagram-v2)

## 8 更新履歷

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

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