



Margo

Markdown in. Durable outputs out.

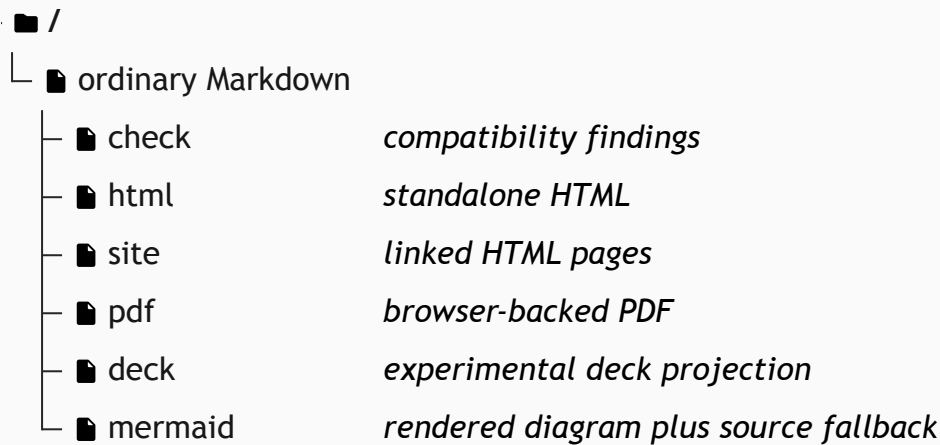


Margo is one Go module and one `margo` command for turning ordinary Markdown into standalone HTML, linked static sites, PDF documents, and an experimental presentation deck projection. This showcase is the public path through those capabilities: each page is a small, runnable-looking example of one feature.

Start with the surface

Feature ↕	Output ↕	Best first stop ↕
Markdown compiler	A semantic document model	Markdown
Standalone HTML	One portable HTML page	HTML
Static sites	Linked pages plus a manifest	Static sites
PDF documents	Print-ready PDF bytes	PDF
Presentation decks	Experimental HTML/PDF projection	Decks
Charts	Static SVG, accessible data, optional interaction	Charts
Mermaid diagrams	Rendered flowcharts with a text fallback	Mermaid
Policy and diagnostics	Actionable validation	Policy

One source, several projections



Mermaid diagram for One source, several projections, with its source available as a text fallback.

The compiler keeps the document content separate from the host-owned frame, metadata, assets, and publication choices. The CLI adds explicit output and diagnostic boundaries around the same module surface. This tree uses Mermaid's [TreeView syntax](#), so the visual structure remains source text with a readable fallback.

A predictable workflow

```
margo check guide.md
margo html guide.md --output guide.html
margo pdf guide.md --output guide.pdf
```

For a larger publication, point `margo site` at a directory of Markdown files. For a programmatic integration, import the root module and choose the package that owns the projection you need.

This is a feature tour, not the project's internal decision log. For source, releases, and the complete public README, visit the [Margo repository](#).

