

Seven dashboards, and what each one refuses to show

Eight kilobytes, no JavaScript, no external request, byte-identical on rebuild.
Each of the seven states the decision it informs and ends by naming what it does not show — and one of them found an error in this platform's own teaching.

Pierre Robentz Cassion

Lesson 8 of 8

Dashboards and Reporting — The room

What this lesson covers

- What is actually shipped
- The four constraints that decided the design
- What each one is for, and what it refuses
- The one that found an error
- What these dashboards do not do
- The pattern, portable
- Report it whole
- What comes next

What is actually shipped — Shell

```
pnpm project:dashboards  
ls apps/data-analysis/public/projects/*/dashboard/
```

What is actually shipped

- Seven projects, two languages, fourteen files, about 8 kB each — Static HTML built from committed CSVs, with the...
- No JavaScript, no external request, no generation date — Rebuilding on unchanged data produces byte-identical files,...

The four constraints that decided the design

- The audience works offline and on metered connections — That removed anything needing a live server or a CDN before any...
- The Content-Security-Policy forbids inline script bodies — `script-src` has no `'unsafe-inline'`, so a script in a...
- The deploy has no Python, no TeX and no pandoc — Rendering the project notebook would need Quarto, Python, pandas and a...
- The output has to be reviewable — No clock, no build stamp, no generated identifier — so the page is a function of the...
- Three of those four are constraints rather than preferences — and the design is what is left once they are respected

What each one is for, and what it refuses

Project	Decision	Refuses
WASH coverage	Which six communities get rehabilitation	Unweighted proportions from a clustered survey; the district column normalised from six spellings
Nutrition screening	Where the next round goes	The commune ranking beyond the three highlighted; screening is not a survey
Protection pathway	Which gate to fix first	Four-way disaggregation — nineteen cells of one
Food security	Which instrument the caseload rests on	Averaging the four prevalences; none is an IPC classification
School attendance	Which schools need a response	Counting the strike period as absence; any causal claim about feeding
SMART survey	Whether a prevalence is publishable	The prevalence itself, until quality control passes
Vaccination	Whether coverage is service or denominator	Months with no report drawn as zero

What each one is for, and what it refuses

- The refusal column is the one to read first — It is the section each page ends with, and it took longer to write than. . .
- The SMART row is the strongest of the seven — The dashboard for a nutrition survey declines to show the prevalence, . . .

The one that found an error

- The documentation was right — The JMP improved list includes packaged or delivered water — tanker-truck here — and...
- Nothing else would have found it — The lesson was internally consistent, the figure used the same set so it agreed, the...
- That is the argument for a definition panel, made by a failure rather than by assertion — and it is why the correction...

What these dashboards do not do

- No interactivity — A reader cannot filter, drill or re-slice
- No live refresh — The page is rebuilt when the data is, by a script somebody runs
- No access control — They are public static files, which is fine because every dataset here is synthetic — and would not. . .
- No accessibility audit — They are semantic HTML with a documented reading order and they have not been tested with a. . .

The pattern, portable

- One file per audience-question, not per data source
- Every number computed by a script from a versioned file — with nothing typed
- Provenance and definitions on the page — not in a separate document nobody opens
- A section naming what the page does not show — written before the page is considered finished
- Deterministic output — so "is this current?" is a question the version control system answers

Report it whole — Example

Project dashboards

Seven projects, two languages, 14 files, ~8 kB each. Static HTML built by `scripts/export/build-project-dashboards.mjs` from the committed dataset files, with each project's committed SVG figure inlined.

No JavaScript (the CSP forbids inline script bodies), no external request, no generation date. Rebuilding on unchanged data is byte-identical.

Each page states the decision it informs and ends with what it does not show. `dashboard.ready` is true on all seven projects and `project-files.test.ts` fails if a declared file is missing.

Not done: no interactivity, no scheduled refresh, no access control, no screen-reader audit.

Report it whole

- The last line is the one to copy — A dashboard section without a "not done" is a section nobody has audited

What comes next

- Twenty-one courses and one argument — A number in a programme report is a claim, and everything here is about what has. . .
- The last step is the one the others are for — An analysis that never reaches a decision is an analysis that did not. . .

Read the full lesson, with runnable code

```
https://data-analysis.cassion.dev/courses/dashboards-and-reporting/  
dash-08-the-dashboards-this-platform-ships
```