What an evaluator does to your numbers

Six OECD DAC criteria, the question each one asks of your indicator set, and the gaps you can only close before the evaluation rather than during it.

Pierre Robentz Cassion

Lesson 8 of 8

Indicator Design and the LogFrame — Judgement

What this lesson covers

- The review is the test your indicators were built for
- The six criteria, and the question each asks of your data
- Effectiveness: where the reference sheet pays for itself
- Impact: say what you cannot claim
- Efficiency needs a denominator you probably do not have
- The four gaps you can only close beforehand
- Build the evidence pack as you go
- The one-page summary an evaluator actually wants
- Where this course leaves you

The review is the test your indicators were built for

- At some point somebody external — a mid-term review, an end-of-project evaluation, a donor's monitoring adviser — takes your LogFrame, your reference sheets and your reported figures, and asks whether they support what the programme claims.

The six criteria, and the question each asks of your data

Criterion	The question	What it needs from you
Relevance	Was this the right thing to do?	Needs assessment data, and indicators that measure the need, not only the activity
Coherence	Did it fit with everything else?	Comparable definitions — the standards lesson, cashed in
Effectiveness	Did it achieve its objectives?	Baseline, target, and an outcome indicator that is not an output
Efficiency	Was it a reasonable use of resources?	Reach or volume with a denominator, and cost data joined to it
Impact	What difference did it make overall?	A counterfactual, or an honest statement that you do not have one
Sustainability	Will the benefits last?	Something measured after handover, which almost nobody has

Effectiveness: where the reference sheet pays for itself

- **What was the target, and where did it come from?** The four methods from the previous lesson. "Negotiated" is an...
- **What was the baseline, measured how, and when?** Same indicator, same method, before.
- **Is the reported value computed the same way as the baseline?** Version the sheet or you cannot answer this.
- **What else could explain the change?** The alternatives you listed when answering question two of lesson 2.

Effectiveness: where the reference sheet pays for itself — In Python

```
evidence = pd.DataFrame({
    "indicator": ["penta3_coverage_percent_monthly"],
    "baseline": [61.4],
    "baseline_period": ["2023 mean"],
    "target": [85.0],
    "target_basis": ["negotiated, assumed full outreach budget"],
    "latest": [72.8],
    "sheet_version": ["2.1"],
    "same_method": [True],
    "confounders": ["reporting completeness rose from 71% to 77% over the period"],
})
```

Effectiveness: where the reference sheet pays for itself — In R

```
evidence <- tibble::tribble(
  ~indicator,                          ~baseline, ~target, ~latest, ~sheet_version, ~same_
    method,
  "penta3_coverage_percent_monthly",      61.4,    85.0,    72.8,          "2.1",
    TRUE
)
```

Impact: say what you cannot claim

- **A before-and-after change**, labelled as such, with the alternative explanations named.
- **A comparison against a non-programme area**, with an explicit statement of why the two are or are not comparable.
- **Consistency with the theory of change** — the intermediate steps moved in the order the theory predicted, which is...

Efficiency needs a denominator you probably do not have

- Efficiency questions are usually cost per unit of result, and they fail on the join rather than on the arithmetic: financial data is by budget line and month, programme data is by activity and facility, and nothing connects them.

The four gaps you can only close beforehand

- **No baseline on the same definition.** Fixed by versioning the sheet at the start.
- **No data on the assumptions.** Fixed by giving the two or three critical assumptions their own indicators.
- **No outcome indicator, only outputs.** Fixed at LogFrame design, and lesson 1 is about spotting it.
- **No post-handover measurement.** Fixed by budgeting one follow-up round, which almost no proposal does and every...

Build the evidence pack as you go — Example

```
evidence/
  indicators/              reference sheets, versioned
  baseline/                values, method notes, extract date
  targets/                 values, basis, revision history
  series/                  every period, every indicator, one row each
  dqa/                     assessment rounds and follow-ups
  assumptions/             stock-outs, access days, vacancy rates
  limitations.md           what the data cannot support
```

The one-page summary an evaluator actually wants — In Python

```
summary = (
    evidence[["indicator", "baseline", "target", "latest", "same_method"]]
    .assign(progress=lambda d: (d["latest"] - d["baseline"])
                               / (d["target"] - d["baseline"]))
)
print(summary.round(2))
```

The one-page summary an evaluator actually wants — In R

```
evidence |>
  mutate(progress = (latest - baseline) / (target - baseline)) |>
  select(indicator, baseline, target, latest, progress, same_method)
```

The one-page summary an evaluator actually wants

An evaluation is not an examination of your programme. It is an examination of whether your evidence supports what your programme said. Those come apart more often than anyone expects, and the reference sheet is where they are held together.

Where this course leaves you

- You can trace an indicator from the change it is meant to evidence down to the arithmetic, write a reference sheet a stranger can compute from, defend a denominator, distinguish reach from coverage, adopt a standard definition and document your departure from it, set a baseline and a target that survive scrutiny, and assemble the evidence a review will ask for.

Read the full lesson, with runnable code

https://data-analysis.cassion.dev/courses/indicator-design-logframe/indicators-08-the-review