🌻 Automating the standard questions – slide creation functions

10 Jul 2026

The describes what a skilled analyst asks of a causal map and how they answer it: which filters, in what order, counting by sources or citations, with which caveats. A slide creation function (SCF) is that recipe written down as code, so the app can apply it for you. The 1-click report in the Causal Map app runs a catalogue of SCFs to draft a first set of report views.

What an SCF is

Each SCF answers one question from the catalogue and has three parts:

- **An applicability test.** The SCF reads some basic facts about the project and decides whether its question is worth asking here. The sentiment discrepancy view only appears when sentiment is coded and sources disagree about at least one factor; the group comparison only appears when there is a metadata field with a handful of values covering enough of the data; the zoomed view only appears when labels use the ; hierarchy convention. A question that does not apply produces no slide.
- **A parameterisation.** The SCF sets its own thresholds and choices from the data, the way an analyst would: thin the link bundles only when there are enough to clutter the map, pick the path-tracing endpoints from the factors with the highest and lowest , choose the grouping field with the best coverage.
- **A fixed filter recipe.** The output is an ordered stack of the app's own filters, the same ones you would apply by hand (). Nothing is invented: every slide can be explained by pointing at its filters, and reproduced by loading its bookmark.

The catalogue, with examples

Each SCF answers one question from the . A sample of the current set shows the pattern:

Question	SCF	Only when	Adapts how
	Overview map; factors and links tables	always	thins link bundles only when there are enough to clutter; tables sort by the unit they claim

Question	SCF	Only when	Adapts how
	Consensus map, counted by sources	5 or more sources	below that, source counting means nothing and the view stays away
/	One step up/downstream of the top factors	20+ links	ranks by (incoming share of citations), because raw degree mislabels busy intermediates as outcomes
	Source-traced paths between them	whole stories exist	uses , so every chain shown was told end to end by one source
	The Pathways filter's most-told two-step chain	quotes have text positions	on a single-source project the ranking falls back to how often that source tells it
	Factors shaded by incoming sentiment spread	sentiment coded AND real disagreement	the threshold is one +1 against one −1 claim; below it the shading would colour noise
Contested claims	Links carrying claims both ways, sentiment mix shown	2+ positive and 2+ negative claims on one link	treats the disagreement itself as the finding, never averaged away
	Map with per-group counts and chi-square markers	a field with 2 to 6 values covering half the links, 8+ sources	picks the grouping field with the best coverage
	A top-links map per group value	as above, 4 or fewer values	one slide per value
	The map at hierarchy level 1	a quarter of labels use ; levels	otherwise zooming changes nothing
Tribes	Sources clustered by causal story	the stated aim asks about subgroups	never appears unrequested, because cluster membership is unstable on typical data

Adaptation also runs across the whole report. A project with hardly any sources switches every count from sources to citations and drops the views that need a population. A map whose first paint comes out portrait gets its factor labels widened and repaints before the screenshot.

What the AI does and does not do

The deterministic part, which views to offer and how to parameterise them, is plain code. AI is used in three narrow places, each of which can fail without breaking the report:

- **Selection.** Given the researcher's stated aim, one call orders the applicable views for a reader (orientation first, then the views that speak to the aim, then supporting detail) and drops only the clearly irrelevant, before anything is built.
- **Review.** After the slides are built, a model reads each slide's title, description and most-cited links, discards slides that say nothing, and writes a short commentary for the rest, shown as a widget on the slide.
- **Narrative.** Each surviving map gets a vignette: a narrative description with verbatim quotes, through the same vignette channel you can run by hand, placed on the slide after its map with a cross-reference each way. The report closes with a typical source's own story, and opens with a contents slide stating the aim it worked from.

The AI never authors filters. It only chooses from things that already exist: slides, factor names, field names.

Why the commentary is worded the way it is

The commentary and narratives follow the reasoning rules of this chapter. They report evidence rather than fact: "12 sources said X influenced Y", never "X drives Y". Counts are treated as salience of mention, never effect size. And the is respected in the tracing views: the drivers-to-outcomes slide uses , so any pathway it narrates was told end to end by at least one source; if nobody told a whole story, there is no slide, rather than a stitched chain nobody claimed.

Antecedents

The pieces have separate histories. The filters themselves descend from the standard analyses that causal mapping software has offered since the 1990s (Decision Explorer's domain, centrality and loop analyses in the Eden and Ackermann tradition), from CAQDAS query tools (NVivo's matrix coding queries, MAXQDA's model templates), and from Miles and Huberman's catalogue of named displays, each with guidance on when to use it. The layer above, which decides whether a question is worth asking of this dataset, sets the parameters, ranks the results and writes the caption, has so far been automated only for numeric data: the "auto-insights" systems in visualisation research (QuickInsights, DataShot, Calliope, and commercially Tableau's Explain Data), chart-recommendation systems that encode "when is this view a good idea" as explicit constraints (Draco, CompassQL), and the Automatic Statistician's generated model reports. In qualitative analysis that layer has always been the analyst, guided at most by when-to-use advice in methods texts. SCFs apply the same architecture to qualitative causal evidence, where the applicability tests are methodological (is sentiment coded? did anyone tell a whole story?) and the caption's voice carries the .

Where it is going

The report reads its aim from the project description. The next step is an intake conversation that asks "what do you want to find out?", asks follow-ups until the answer is usable, and saves it there, so both the

coding and the report are steered by the same stated aim.