🌻 Our approach is minimalist – we code only bare causation

Why we stick to bare causation in causal mapping.

Our rule of thumb: record only that "C causes D." No coding of necessity, non-linearity, moderators, or strength. Just who said what causes what.

The short case

- **It avoids false precision.** Labelling links as "necessary," "moderator," "non-linear," or assigning strengths suggests evidence we rarely have. We prefer to show what was claimed and how often, then let readers judge. Maps are primarily **epistemic**—repositories of evidence about people's beliefs—not truth machines.
- **It scales and compares.** Bare links plus rich factor labels let us aggregate, filter, and compare across sources, groups, and contexts without fighting about semantics of special symbols. Our tools then summarise with counts (citations, sources) and simple derived measures (like "outcomeness"), instead of speculative link attributes.

What we record

- **Factors (boxes):** short propositions that do the heavy lifting (e.g., "Not enough money," "Won't take a holiday this year").
- **Links (arrows):** undifferentiated causal influence claims between factors. A link means "P said C influences D." That's it.

What we deliberately don't code on links

- Necessity/sufficiency
- Non-linear forms or feedback classifications
- Moderator/mediator/inhibitor role
- Polarity or strength

Why? Because (a) respondents seldom state these explicitly; (b) analysts rarely agree on them from text alone; and (c) they reduce inter-coder reliability and slow projects down without very much which we can dependably aggregate.

Our analyses are still useful

Coding bare links doesn't make maps "impoverished": Causal mapping produces models you can query to answer questions

Bottom line

Most of the time, we code only: "C causes D (as claimed by P)." That minimal, transparent unit is reliable, scalable, and faithful to the data people actually provide. Everything richer belongs in **analysis and interpretation**, not in speculative link types baked into the coding.