

# CONSTRAINT MORPH

TRANSFORM-F-0003

**FAMILY****CODE**

TRANSFORM

TRANSFORM-F-0003

Adapt an existing concept, plan or system to new constraints while preserving its defining core and making every compromise visible.

**WHAT IT IS**

CONSTRAINT MORPH is for the moment after reality changes: the budget drops, the platform changes, the deadline moves, a feature disappears, the team shrinks, a requirement becomes mandatory or the original solution no longer fits its environment.

Rather than starting over, it identifies what makes the existing solution itself, classifies the new constraints, and rebuilds only what must change. The output shows which parts survived, which mutated and which trade-offs were introduced so adaptation does not become accidental redesign.

**BEST FOR**

Product concepts, workflows, creative systems, project plans, technical approaches, service designs, content systems and operating models that need to survive a changed environment.

**WHAT YOU GIVE IT**

The existing solution, the new constraints or environment, the elements that define its identity, and optionally the priorities that should govern trade-offs.

**WHAT IT RETURNS**

An adapted version, a constraint-response map, a preservation ledger and explicit trade-offs or unresolved conflicts.

**HOW TO USE IT**

1. Describe the existing solution before listing the new constraints; the formula needs a baseline to know what counts as change.
2. Mark non-negotiables separately from preferences. Treating everything as sacred makes adaptation impossible.
3. Use the trade-off ledger to decide whether the transformed solution is still the same thing or whether the constraints justify a new concept entirely.

**GOOD FIT IF...**

Choose this formula when you want to save the core of an existing solution after the conditions around it have changed.

NOT SURE KITTY - LAB / FORMULA SERIES