

The Ideation Compiler

The diagram is the analysis: why the case for systems change should be compiled from a drawing, not composed from a blank page, and never guessed.

THE SCIENCE OF TEAMS · CANVAS

The gap every organization falls into

Somewhere in your organization right now there is a whiteboard photograph. On it is the best available understanding of how a team actually works: who does what, what flows where, which output the whole arrangement exists to produce. It was drawn in an afternoon by the people closest to the work, and it is more accurate than anything in the process repository.

And it is analytically inert. The picture cannot say what it costs, what it harms, or what could be removed. To turn it into a case for change, someone must now open a blank document and start writing: re-describing by hand what the drawing already shows, reasoning informally about structure the drawing already encodes, and producing, days later, a deliverable whose connection to the original diagram is a matter of trust. Every diagramming tool on the market stops at the picture. Every analysis begins at the blank page. The most expensive work in organizational design happens in the gap between them, and it happens by hand, differently, every single time.

The Science of Teams Canvas closes that gap by refusing to treat the diagram as a picture. The diagram is a semantic model of the team system, and the Canvas is something no other tool in this market is: an ideation compiler. The diagnosis, the ideation plan, and the deck are compiled from the drawing.

The proposition: draw with semantics, compile the case

The right analogy is not a whiteboard. It is a compiler. To a compiler, source code is not a picture of a program; it is the program, and the build is a deterministic translation of it. The Canvas treats a team diagram the same way. The drawing is the source; the analyst's marks are its type annotations; the structural gaps the engine surfaces (a cost with no benefit weighed against it, a harm with nothing to measure it by) are compiler warnings for your organization's design; and the report and deck are the build artifacts. Same source, same build, every time. Nobody writes a stress report by looking at a CAD sketch and typing what they think they see, and nobody should write the case for restructuring a team that way either.

The Canvas brings that discipline to team systems. An analyst draws the system as it actually is (roles, inputs, outputs, subsystems, referenced systems, and the value flows between them) and marks it with declared judgments: where benefit and cost actually sit, where harm attaches, where a contradiction lives, which elements the analyst intends to make unnecessary and which must stay and be minimized. From that single drawing, in one click, the engine compiles the complete work product: a full written diagnosis in report-ready prose, a framed ideation plan that runs the framework's Solution Algorithm against every system and subsystem at every layer, a gallery of structurally matched solution models, and a boardroom-ready presentation, every figure in them rendered from the same drawing, in the same visual language, down to the marks on the flows.



One drawing, one click: the diagnosis, the ideation plan, and the presentation are compiled from the same semantic model.

The lineage: TRIZ for the 21st century

This did not appear from nowhere. Genrich Altshuller read patents by the hundred thousand and proved that breakthrough solutions are not lightning; they are pattern. TRIZ turned that discovery into a discipline of systematic ideation: contradictions named, ideality measured, an Ideal Final Result to steer toward, and a finite catalog of inventive principles that resolve most problems ever recorded.

The Science of Teams is that discipline's next evolution, carrying the instruments to their most consequential subject yet: the organized effort of people, and now of people and AI together. The instruments matured with the subject: team contradictions, the value and ideality balances, the IFR of a role or an output, twenty-five solution models in place of forty inventive principles, and the Seven Motifs as the operational test of ideal. And where TRIZ has always been practiced by hand by those who mastered it, TSoT ships as software: the compiler routes the right instruments to the right elements for you, at every layer of the system, and hands you the judgment work with the terms already laid out.

TRIZ made invention systematic. The Science of Teams makes team design systematic. The ideation compiler makes it executable.

Three commitments no one else makes together

Deterministic by design

Every generated sentence traces to a mark the analyst placed or a geometric fact of the drawing. Nothing is imagined, estimated, or improvised. The same diagram compiles to the same analysis, every time.

Methodology, embedded

The Science of Teams framework is not a poster on the wall; it is the engine. The three solution tests, the value and ideality balances, twenty-five solution models, and the Seven Motifs run as executable structure.

Whiteboard to boardroom

The output is not notes. It is the finished work product: an illustrated report and a presentation in a coherent visual language, generated in minutes, ready to carry a name and a recommendation.

1 · Deterministic by design: the anti-black-box

This is the claim to lead with, because it is the claim the market cannot match right now. Every competitor racing to add "insight" on top of diagrams is doing it with a large language model, which means the deliverable that reaches a board carries hallucination risk in every paragraph. The Canvas takes the opposite contract, and enforces it in the engine: **the tool frames questions; it never asserts conclusions**. It will tell you the framework's value test now bears on the benefit you marked at one element and the cost you marked at another, and it will refuse, structurally, to tell you the balance holds. Whether an inequality is satisfied, whether a need can truly be removed, which candidate direction to take: those are the analyst's, always. The regression suite literally asserts that no generated sentence claims a satisfied inequality or a real-world conclusion.

For anyone who must defend an analysis in front of a skeptical executive, a client, a works council, or an auditor, this is the difference between a work product and a liability. **No black box. Every line in the report traces to something you drew.**

2 · A methodology that cannot be skipped

Generic canvases give you shapes. Enterprise-architecture suites give you notation. Neither gives you judgment scaffolding. The Canvas operationalizes a complete ideation discipline: it reads the biases the analyst has declared. An element marked **IFR** is to be made unnecessary; an element marked **BPS** stays, and the work is to minimize its cost and harm without eliminating it. It routes the framework's tests to the right elements at every layer of the system, including inside referenced subsystems. It derives the analysis terms from the drawing itself: benefits and costs from the **BENEFIT** and **COST** marks, actual output from what the system actually produces as drawn, problems from the **CONTRADICTION** and out-of-control marks, harms from the harm marks, and states plainly which terms the diagram does not yet supply, handing exactly those to the analyst to define in real terms.

It surfaces the structural gaps that hide in any complex system: a cost with no benefit marked against it, a benefit with no cost, a harm with no measure to weigh it. It matches the framework's twenty-five solution

models against the structure of the drawing and offers them strictly as instruments of reduction: ways to remove a need or shrink what stays, never capabilities to add. And it tests every candidate move against the Seven Operational Motifs, the framework's own operational definition of ideal. **The tool does the structure; the analyst does the judgment.** That is the trust story, the professional-liability story, and the anti-deskilling story in one sentence.

3 · The deliverable is the product

The output of an afternoon at the Canvas is not a diagram export. It is the document a consultancy would bill a week for: a numbered, illustrated report that moves from the system under examination, through the diagnosis, into the Solution Algorithm worksheets (each variable derived step by step, each formula appearing exactly once, at the moment its terms have just been defined) through the matched solution models, each shown with a mini drawing of the specific elements it addresses in *this* system, and closing with the motifs and a decision page. Alongside it, a presentation in the same visual language, where each gap-analysis slide carries the framed question as its title. The diagrams in both are rendered by the same engine that draws the live canvas, in the same clean design language, so the deliverable looks like it came from a design team, because in effect it did.

Who this is for

Practitioners who sell defensible change. Continuous-improvement, organizational-design, and Lean/TRIZ-adjacent consultants, for whom the pipeline collapses days of writing into minutes while raising the floor on rigor, and whose clients can, for once, audit every claim back to the drawing they helped make.

Transformation offices deciding where AI belongs. The framework already carries the tests an AI role must pass (its benefits must meet its costs, its actual output must exceed its required output net of the problems and harms of producing it), and the human/AI designations are first-class marks on the canvas. In 2026, every operations leader is being asked which roles should be people and which should be models. There is a genuinely sharp line available here: *the tool for deciding where AI belongs in your organization, that does not use AI to decide.*

The framework's own teaching channel. For readers and certified practitioners of The Science of Teams, the Canvas is the companion that makes the book executable: the end-of-chapter activities become the worksheets the tool compiles, with the terms already read off the diagram.

Answering the two objections you will hear

"So it's templates." Templates do not reason about structure. The engine computes system ownership from geometry, walks value flows to their deepest common container, detects one-sided balances, matches solution models by the shape of the drawing, and recurses through embedded systems, none of which a

mail-merge can do. The prose itself is drawn from a hand-written phrasebank with seeded variation, so ten reports never read like the same form letter; but within every variant, the facts, names, and formulas are byte-identical, because the facts are not the engine's to vary.

"The vocabulary is a barrier." It is, cold. Which is why the marketing surface should lead with the outcome (*a defensible restructuring case in an afternoon*) and let the methodology be the credibility layer underneath rather than the front door. The IFR and the BPS are what make the analysis rigorous; the one-click boardroom deck is what makes the demo land. Sell the second; deliver on the first.

THE POSITIONING STATEMENT

For consultants and operations leaders who must justify structural change to skeptical audiences, The Science of Teams Canvas is the ideation compiler: it compiles a drawn team diagram into a complete, methodology-governed diagnosis and ideation plan, deterministically, with every sentence traceable to the diagram. Unlike diagramming tools that stop at the picture, or AI assistants that improvise, it produces defensible analysis you can put your name on.

The one-line versions

Not another canvas. Not another copilot. An ideation compiler.

Draw the system. Mark your judgments. Compile the case for change.

No black box: every line in the report traces to something you drew.

The tool does the structure. The analyst does the judgment.