KOHNEX BRAIN

Thinking With Kohnex

A human guide to research, discovery, and decision intelligence for AI agents.

Human Edition · 2026 · Public manuscript

Contents

A practical field guide for turning unfamiliar questions into better research, experiments, and decisions.

KOHNEX BRAIN / HUMAN EDITION

1. The New Research Problem

Most AI systems are optimized to answer a question. Real work is harder. A researcher does not begin with a perfectly formed question. An engineer often has incomplete evidence. A founder may be looking for a product that does not have a name yet. A security team must decide while facts are arriving and changing.

These are not lookup problems. They are orientation problems.

The difficult part is often knowing what to ask next, which disciplines matter, which assumptions are fragile, and what evidence would change the conclusion. A language model can produce a fluent answer while still missing the structure of the problem. More text does not automatically create better orientation.

Kohnex is designed for this gap. It gives an AI agent a structured research substrate that can connect concepts across domains, surface related patterns, compare ways of thinking, and preserve a path from a question toward a decision. It does not replace judgment. It helps judgment start from a better map.

The shift from answer to orientation

An answer is a destination. Orientation is the ability to see the terrain:

- What is known?
- What is uncertain?
- Which concepts are adjacent?
- Which analogy is useful and which is merely attractive?

- What should be tested next?

The useful Kohnex question is rarely “give me a fact.” It is closer to “help me see the problem from the right angles, then tell me what action follows.”

A practical rule

Use Kohnex when the problem involves ambiguity, multiple domains, competing explanations, design choices, discovery, or a need for traceable reasoning. Use ordinary search or a direct model when the task is a simple fact lookup or a deterministic transformation.

Exercise

Write down a problem you are currently working on. Separate the final decision from the research needed to make it. The research portion is the first candidate for a Kohnex session.

KOHNEX BRAIN / HUMAN EDITION

2. What Kohnex Is

Kohnex Brain is a decision-intelligence layer for AI agents. It is delivered through tools that let an agent research a question, inspect a domain, trace a connection, understand a thinking mode, and inspect the graph at a high level.

The important word is **layer**. Kohnex is not trying to be the only model in a system. It gives the model a structured place to explore before it commits to an answer.

Four capabilities

Research

Kohnex helps an agent orient around unfamiliar territory. It can surface concepts that a narrow prompt would not name and suggest directions for follow-up.

Discovery

Kohnex can combine ideas from different domains to generate product opportunities, architecture patterns, and hypotheses. These are starting points for validation, not guarantees of novelty or truth.

Decision support

Kohnex helps compare alternatives, identify trade-offs, and distinguish a confident conclusion from an interesting possibility.

Explainability

Good use of Kohnex asks not only for a conclusion but also for the concepts, assumptions, evidence, and next questions behind it.

What Kohnex is not

Kohnex is not an oracle that removes uncertainty. It is not a replacement for domain experts, experiments, legal review, security controls, or human accountability. It is not a promise that every cross-domain connection is useful.

The best users treat output as structured material for thinking. They ask follow-up questions, challenge assumptions, and convert ideas into tests.

A compact mental model

```
Question → Orientation → Connections → Evidence → Options → Decision →
Experiment
```

Kohnex is most valuable in the middle of this chain, where an ordinary answer is too early and a final decision is too soon.

KOHNEX BRAIN / HUMAN EDITION

3. The Kohnex Mental Model

You do not need to understand the private implementation of the brain to use it well. You need a useful mental model of what enters the system and what comes back.

Questions are activation requests

When you ask Kohnex a question, you are asking it to orient a search through a structured body of concepts. Specific nouns provide anchors. Verbs provide the desired operation. Constraints tell the system how to judge the result.

Compare:

```
Tell me about consensus.
```

with:

```
Research a leaderless consensus design for a multi-agent system.
Connect distributed systems, biology, and game theory. Compare
failure modes, coordination cost, and a practical first experiment.
```

The second request gives the system a problem, domains, and an output contract.

Connections are hypotheses

A connection is an invitation to investigate. It may be direct, analogical, historical, architectural, or strategic. A strong output explains why the connection matters and where it stops being reliable.

Modes are lenses

A thinking mode changes the behavior of the exploration. Focus narrows. Explore broadens. Skeptic challenges. Mechanistic asks how. Teleological asks toward what purpose. No mode is universally best.

Results need a next move

End a Kohnex session with one of four next moves: ask a sharper question, inspect a domain, trace a path, or design an experiment. If there is no next move, the session may have produced entertainment rather than intelligence.

Exercise

Take one vague question and rewrite it three times: once for Focus, once for Explore, and once for Evidential. Observe how the requested answer changes without changing the topic.

KOHNEX BRAIN / HUMAN EDITION

4. Connecting Your Agent

Kohnex is designed to be used from an AI agent through MCP. The technical setup is documented in the Docs page. This chapter explains the user experience after the connection works.

The first connection

You need an account, an API key, an MCP-compatible client, and a remote server configuration. Add the Kohnex server to the client, restart the client, and confirm that the Kohnex tools appear.

Do not paste private keys into prompts, screenshots, public repositories, or shared documents. Treat the key as a credential with the same care as a production API token.

The first smoke test

Start with a question that is broad enough to prove the graph is reachable but specific enough to recognize a useful answer:

```
Use Kohnex to find design patterns for fault-tolerant consensus
in a multi-agent system. Start in Explore mode and return three
patterns, their trade-offs, and one question for follow-up.
```

The exact output will vary. That is expected. What you are checking is that the agent can call the tool, receive a structured response, name relevant concepts, and continue the conversation.

A healthy connection

A connection is working when the agent can:

- Call a Kohnex tool without silently falling back to a generic answer.
- Identify the requested thinking mode.
- Return cross-domain concepts with explanations.
- Use a follow-up tool to investigate a result.
- State uncertainty rather than inventing evidence.

Troubleshooting

If the tools do not appear, check the JSON configuration, API key, client restart, network access, and account permissions. If the tools appear but the answer is generic, improve the prompt and explicitly request a Kohnex call and a mode.

KOHNEX BRAIN / HUMAN EDITION

5. Asking Better Questions

Kohnex rewards questions with structure. A strong question has five parts:

1. **Objective:** what are you trying to understand or decide?
2. **Context:** what system, user, or constraint surrounds it?
3. **Domains:** which perspectives should be connected?
4. **Mode:** what kind of reasoning should dominate?
5. **Output:** what form should the answer take?

The five-part prompt

```
Objective: design a lower-cost research workflow for an AI agent.
Context: the team has limited GPU memory and needs traceable answers.
Domains: neuroscience, compression, systems engineering.
Mode: Mechanistic, then Evidential.
Output: architecture options, trade-offs, risks, and a validation plan.
```

This is better than adding a long paragraph of background. It makes the desired reasoning visible.

Ask for disagreement

If every answer is framed as confirmation, the agent has little room to correct you. Add a disconfirming request:

```
What assumptions in this proposal are most likely to be wrong?
What evidence would make us reject it?
```

Ask for boundaries

Analogies are useful when bounded. Ask:

```
Explain where this analogy is strong, where it breaks, and what
engineering decision it can responsibly inform.
```

Prompt anti-patterns

Avoid “tell me everything,” vague requests for “the best solution,” and prompts that ask for novelty without defining the user, problem, or constraint. Broad discovery still needs a target.

Exercise

Rewrite one current work prompt using Objective, Context, Domains, Mode, and Output. Save both versions and compare the quality of the next action each one produces.

KOHNEX BRAIN / HUMAN EDITION

6. Your First Research Session

The first session should be a short loop, not a marathon. Reserve twenty minutes and use four passes.

Pass one: orient

Ask Explore mode for a map of the problem. Do not ask for a final answer yet. Look for vocabulary, neighboring domains, and concepts you did not expect.

Pass two: select

Choose the two or three connections that appear most relevant. Ask the agent to explain why each matters and what evidence is missing.

Pass three: challenge

Use Skeptic or Evidential mode. Ask the agent to find weaknesses, counterexamples, and conditions under which the recommendation fails.

Pass four: act

Ask for a small experiment, prototype, interview, measurement, or decision memo. Keep the action small enough to complete quickly.

Example session

```
Explore the problem of detecting coordinated abuse across a marketplace.
Connect cybersecurity, trust systems, collective intelligence, and game
theory. Return a map of concepts, not a final solution.
```

Follow-up:

```
Compare the three strongest approaches using Skeptic mode. What would
cause each approach to fail? Which assumption should we test first?
```

Final:

```
Design a one-week validation experiment. Include data needed, success
criteria, false-positive risks, and a decision rule for continuing.
```

The value is not the first answer. It is the improvement in the question between pass one and pass four.

KOHNEX BRAIN / HUMAN EDITION

7. Understanding Thinking Modes

Modes are practical lenses. They are not personality types and they are not claims about human cognition. They are controls for how a Kohnex session explores, filters, challenges, and synthesizes concepts.

Core modes

Focus seeks precision. Explore broadens the search. Out-of-Box makes lateral combinations. Dream increases serendipity. Intuitive favors rapid pattern matching. Beamform finds intersections between multiple concepts. Swarm emphasizes collective agreement. Predator searches adversarially. Regeneration looks for repair. Hibernate favors low-resource strategies. Immune looks for defense and threat patterns. Photosynthesis explores energy and resource conversion. Quantum keeps multiple possibilities open before choosing.

Logic modes

Inductive moves from examples toward patterns. Deductive moves from rules toward conclusions. Systems examines feedback, dependencies, and boundaries. Normative asks what should be done under values or constraints. Algorithmic turns a goal into steps. Conceptual clarifies categories and abstractions.

Reasoning modes

Scenario compares plausible futures. Evidential tests claims against support and disconfirmation. Reductionist finds the essential core. Mechanistic explains how a process works. Teleological reasons from purpose and function. Model-Based compares explicit models and their assumptions.

Choosing a mode

Choose the mode based on the next decision, not the topic. “Cybersecurity” does not always mean Predator. A security architecture may need Systems. A security claim may need Evidential. A security incident may need Diagnostician.

Combining modes

Use a sequence when the problem has stages:

```
Explore → Beamform → Skeptic → Evidential → Model-Based
```

This moves from possibility to intersection, challenge, evidence, and comparison.

KOHNEX BRAIN / HUMAN EDITION

8. Researching Unfamiliar Territory

Research begins before you know the right vocabulary. The goal of the first Kohnex pass is orientation, not authority.

Step 1: state the unfamiliarity

Tell the agent what you do not know:

```
I understand distributed systems but not clinical decision support.
Help me map the important concepts without assuming that the analogy
is valid. Identify what I should learn first.
```

Step 2: request multiple frames

Ask for technical, operational, user, and risk perspectives. A domain can look simple from one frame and difficult from another.

Step 3: separate signal from metaphor

Metaphor can open a path but cannot prove an engineering claim. Ask the agent to label direct concepts, analogies, assumptions, and validation requirements separately.

Step 4: build a reading and testing plan

End with sources to investigate, terms to search, experts to consult, and experiments to run. Kohnex is the map-maker; primary sources and tests remain the authority.

Research prompt

```
Use Explore mode to orient me to privacy-preserving machine learning.
Connect information theory, distributed systems, and governance.
Separate established principles from analogies. Give me a vocabulary
list, three competing approaches, open questions, and a reading plan.
```

Quality check

Ask whether the output distinguishes what is known from what is suggested. If it does not, run the same question in Evidential mode and request a confidence boundary.

KOHNEX BRAIN / HUMAN EDITION

9. Discovering Product Opportunities

Product discovery is not asking a model for ten startup ideas. It is finding a meaningful problem, a reachable user, and a testable wedge. Kohnex is useful because it can connect a user problem to patterns that are not normally placed beside it.

Start with a job

```
The user is a security operations team. Their job is to decide which
alerts deserve immediate action. Explore where current workflows lose
context, then identify product opportunities that restore that context.
```

This is stronger than “invent a cybersecurity product.”

Use a three-pass discovery loop

First use Out-of-Box or Dream to generate combinations. Then use Reductionist to identify the smallest valuable problem. Finally use Evidential or Scenario to test assumptions about users, competitors, cost, and adoption.

Product opportunity template

For every candidate, request:

- User and painful job.
- Existing workaround.
- Why the timing matters.
- Narrow first product.
- Defensible advantage.
- Technical and behavioral risks.
- One-week validation experiment.

A responsible discovery claim

Kohnex can surface an opportunity. It cannot prove that the opportunity is new, desired, or commercially viable. Novelty requires market research and prior-art review. Demand requires conversations or behavior. Viability requires an experiment.

Exercise

Choose one generated opportunity and delete half its features. If the smaller version still solves a painful job, you may have found a useful wedge.

KOHNEX BRAIN / HUMAN EDITION

10. Generating and Testing Hypotheses

A hypothesis is useful because it can be wrong. A Kohnex session becomes research when its insights become statements that reality can test.

Convert a connection into a claim

Weak:

```
Nature suggests a better way to coordinate agents.
```

Stronger:

```
If agents share compact state summaries before sharing full context,
then coordination quality will remain stable while message cost falls
for this workload.
```

Hypothesis record

```
Claim:
Mechanism:
Expected observation:
Alternative explanation:
Disconfirming result:
Test owner:
Deadline:
```

Ask Mechanistic mode to explain the proposed mechanism. Ask Skeptic to identify alternatives. Ask Evidential to list what would count as support. Ask Scenario to explore consequences if the claim is true or false.

The smallest test

Prefer a test that changes one important assumption. A small benchmark, interview, log replay, prototype, or controlled comparison is more valuable than a large plan that cannot start.

Avoiding false certainty

Use language such as “suggests,” “may,” and “testable hypothesis” until evidence earns stronger language. A beautiful analogy is not a measurement.

KOHNEX BRAIN / HUMAN EDITION

11. Engineering and Architecture

Kohnex can help an engineer move between requirement, mechanism, architecture, and trade-off. It is most useful before implementation, during design review, and after a failure reveals a missing assumption.

Architecture prompt

```
Design a reliable event-processing system for intermittent traffic.
Use Systems mode to map feedback and bottlenecks, Mechanistic mode to
explain failure propagation, and Scenario mode to compare retry,
degradation, and queueing strategies. Return an architecture sketch,
trade-offs, and tests.
```

Use different modes for different design questions

- Systems: where are the feedback loops and boundaries?
- Mechanistic: how does the system behave step by step?
- Reductionist: what is the smallest viable architecture?
- Scenario: what changes under load, failure, or growth?
- Normative: what safety or governance rules must hold?
- Algorithmic: what sequence should the implementation follow?

Architecture review checklist

Ask whether the proposed design identifies state, failure modes, recovery, observability, ownership, and cost. Ask what the design assumes about time, identity, consistency, and human intervention.

Diagram discipline

Have the agent describe a diagram in words before drawing it. Name components, data flows, control flows, trust boundaries, and failure boundaries. This makes the reasoning inspectable and prevents attractive but ambiguous diagrams.

KOHNEX BRAIN / HUMAN EDITION

12. Cybersecurity Reasoning

Security work is adversarial, incomplete, and time-sensitive. Kohnex can help organize hypotheses and connect behaviors, assets, capabilities, and controls. It does not replace telemetry, threat intelligence, incident procedures, or a qualified security team.

A safe investigation prompt

```
Use Predator mode to analyze a defensive zero-trust architecture.
Map likely attack paths, assumptions, detection opportunities, and
containment options. Keep the analysis defensive and identify evidence
needed before treating any path as active.
```

Four questions

1. What is the asset or decision that matters?
2. What behavior would indicate risk?
3. What evidence would distinguish a real incident from noise?
4. What response is proportionate and reversible?

Use Diagnostician for root cause, Immune for adaptive defense, Skeptic for false positives, Systems for control loops, and Evidential for claim review.

Defensive boundary

Keep prompts focused on authorized systems, detection, hardening, simulation, and response. Do not ask Kohnex to operationalize unauthorized access, credential theft, evasion, or destructive action.

Incident memo format

```
Observed:
Hypotheses:
Evidence supporting each:
Evidence against each:
Immediate containment:
Reversible next test:
Escalation owner:
```

KOHNEX BRAIN / HUMAN EDITION

13. Resilience and Fault Tolerance

Resilience is not the absence of failure. It is the ability to preserve important state, reduce harm, recover, and learn. Kohnex helps teams compare recovery patterns across biology, infrastructure, operations, and systems theory without assuming that an analogy is an implementation.

Resilience questions

- What must survive?
- What may be degraded?
- Where can a bypass path exist?
- What is the escalation order?
- When should the system stop trying?
- How does recovery become visible?

Use Regeneration to explore repair and restoration, Systems to map feedback, Mechanistic to explain failure propagation, and Scenario to compare recovery policies.

Design prompt

```
Use Regeneration and Systems modes to design recovery for a service
that may lose dependencies during peak traffic. Compare snapshotting,
redundant paths, graceful degradation, load shedding, and failover.
State what each protects, what it sacrifices, and how to test it.
```

Recovery is a contract

Define recovery point, recovery time, acceptable data loss, user-visible degradation, ownership, and evidence of recovery. A system that “self-heals” invisibly can still fail operationally if nobody knows what happened.

KOHNEX BRAIN / HUMAN EDITION

14. Performance and Resource Decisions

Performance work is a negotiation between quality, latency, memory, energy, and cost. Kohnex helps expose the trade-off instead of treating optimization as a single number.

Begin with a budget

```
I need to reduce inference memory for a transformer service without
losing high-value context. Use Mechanistic mode to identify where memory
is spent, Reductionist mode to find the essential signal, and Scenario
mode to compare pruning, compression, and tiered storage.
```

Ask for a Pareto view

Request alternatives that are not all optimized for the same goal:

- Lowest latency.
- Lowest memory.
- Highest answer quality.
- Lowest operational complexity.
- Lowest energy or cost.

Then choose the trade-off that matches the product, not the one with the most impressive benchmark.

Performance experiment

Record baseline workload, metric, sample size, quality threshold, implementation change, and rollback condition. Ask the agent to predict what the experiment cannot prove. That boundary is as important as the expected gain.

KOHNEX BRAIN / HUMAN EDITION

15. Multi-Agent Consensus

Multiple agents do not automatically produce a better answer. They can reproduce the same blind spot, amplify a bad assumption, or spend more tokens without increasing information. Consensus needs independence, disagreement, and a decision rule.

A useful sequence

1. Ask independent agents to propose solutions.
2. Use Swarm mode to compare agreement and disagreement.
3. Use Skeptic mode to search for shared assumptions.
4. Use Evidential mode to rank claims by support.
5. Use Normative mode when values or governance decide the outcome.

Consensus prompt

```
Compare three independent designs for agent consensus. Identify where
they agree, where they disagree, which assumptions are shared, and what
evidence would resolve the disagreement. Do not treat majority vote as
proof.
```

When not to seek consensus

If the issue is a hard safety constraint, a legal requirement, or a known invariant, do not average opinions. Apply the constraint first and use

agents to explore implementation options inside it.

Decision record

Record the alternatives considered, the decision rule, dissenting view, evidence threshold, and review date. Preserved disagreement prevents a future team from mistaking a temporary consensus for a permanent truth.

KOHNEX BRAIN / HUMAN EDITION

16. Following Evidence and Paths

Kohnex can make a surprising connection visible. The next responsibility is to inspect it. A path is useful when it helps explain why two concepts were placed near each other and what should be checked next.

Path workflow

```
Find a connection → Ask why it matters → Inspect the path
→ Identify assumptions → Find external evidence → Decide whether to use it
```

Use `kohnex_path` when you have two concepts and want the shortest conceptual route between them. Use `kohnex_domain` when you want to inspect the surrounding neighborhood. Use `kohnex_think` when you need synthesis.

Evidence ladder

Separate four levels:

1. **Association:** the graph places ideas in a useful relationship.
2. **Analogy:** a pattern may transfer between domains.
3. **Mechanism:** a plausible process explains the transfer.
4. **Evidence:** observation, research, benchmark, or experiment supports it.

Do not present level one or two as level four.

Follow-up prompt

```
Explain the path between these concepts in plain language. Label each
step as direct relationship, analogy, hypothesis, or evidence. State the
weakest step and the fastest way to test it.
```

KOHNEX BRAIN / HUMAN EDITION

17. Composing Kohnex Tools

The tools become more valuable when used as a deliberate sequence rather than as isolated buttons.

The discovery flow

```
kohnex_think → kohnex_domain → kohnex_path → kohnex_mode
```

Start with a question. Inspect a relevant domain. Trace a surprising connection. Check which mode is best for the next question.

The verification flow

```
kohnex_think(Evidential) → kohnex_path → external research → experiment
```

The graph helps structure what to verify. External sources and experiments establish whether the idea holds.

The architecture flow

```
kohnex_domain → kohnex_think(Systems) → kohnex_think(Mechanistic)
→ kohnex_think(Scenario) → implementation plan
```

The tool roles

- `kohnex_think`: synthesis and activation.
- `kohnex_path`: connection tracing.
- `kohnex_domain`: domain inspection.
- `kohnex_mode`: mode selection and explanation.
- `kohnex_stats`: high-level graph health and coverage.

Use the smallest sequence that answers the next decision. Tool calls should reduce uncertainty, not become ceremony.

KOHNEX BRAIN / HUMAN EDITION

18. From Insight to Experiment

Insight becomes valuable when it changes what you do. End sessions with a transition from language to a measurable action.

The transition template

```
Insight:
Decision it informs:
Assumption to test:
Smallest experiment:
Metric:
Success threshold:
Failure threshold:
Owner:
Review date:
```

Example

```
Insight: compact state summaries may preserve coordination quality.
Decision: whether to prototype summary-based agent communication.
Assumption: summaries retain the information needed for agreement.
Experiment: compare full-context and summary-context runs on the same
workload with independent evaluators.
Metric: decision quality, token cost, latency, and disagreement rate.
```

Review the result

Run the result through Kohnex again, but do not ask it to explain away failure. Ask what the result changes, which hypothesis survived, which failed, and what new question is now more important.

Close the loop

The learning loop is:

```
Question → Hypothesis → Test → Result → Updated question
```

This is how a research tool becomes an organizational capability rather than a source of occasional clever answers.

KOHNEX BRAIN / HUMAN EDITION

19. Working With Uncertainty

Confidence is not the same as truth. A polished answer can have weak support, while an uncertain answer can contain a valuable direction. Use Kohnex to make uncertainty visible and actionable.

Three kinds of uncertainty

Information uncertainty: important facts are missing.
Model uncertainty: different explanations fit the facts.
Execution uncertainty: the idea may work in theory but fail in the environment.

Ask the agent to label which kind is present. Each requires a different next step: research, comparison, or experiment.

A calibrated prompt

```
Give the recommendation, your confidence boundary, the assumptions
that carry the most weight, and what new evidence would change the
recommendation. Do not express confidence as a precise percentage unless
there is a justified measurement.
```

The uncertainty ledger

Keep a short record of claim, support, uncertainty type, owner, and next evidence. This prevents a speculative idea from silently becoming a requirement.

Human responsibility

Kohnex can improve the structure of uncertainty. It cannot accept responsibility for a medical, financial, legal, security, or safety decision. Those decisions still require the appropriate human review.

KOHNEX BRAIN / HUMAN EDITION

20. Team Workflows

Kohnex becomes more useful when a team shares a way of working. Without a shared practice, every prompt becomes a private experiment and the organization cannot learn from it.

Research brief

Before a session, record owner, question, decision deadline, constraints, relevant domains, and definition of useful output. After the session, record the chosen path and unresolved questions.

Review ritual

Ask a second person to review the prompt, not only the final answer. A weak frame can produce a very polished result. Reviewers should challenge the selected mode, missing domains, evidence boundary, and actionability.

Shared prompt patterns

Keep a versioned library of prompts that have produced useful decisions. Include the context and the conditions under which the prompt worked. Do not treat a prompt as a magic spell.

Team roles

- Question owner: explains the decision.

- Researcher: runs and documents the session.
- Challenger: searches for disconfirming evidence.
- Domain reviewer: checks technical validity.
- Decision owner: accepts accountability for action.

Governance

Keep private information out of prompts unless the approved deployment and data policy allow it. Record which outputs informed important decisions. Set a review date for decisions that depend on changing evidence.

KOHNEX BRAIN / HUMAN EDITION

21. Common Failure Modes

Asking for everything

“Tell me everything about X” creates a large answer without a decision. Define the task, audience, and output.

Confusing novelty with value

A strange connection is not automatically a product or discovery. Ask who benefits, what changes, and how to test it.

Treating analogy as proof

Request the boundary and the evidence step. Use the analogy to design a test, not to skip one.

Choosing a mode by topic

Security does not always mean Predator. Select the lens by the reasoning job: diagnose, challenge, compare, design, or decide.

Never following up

The first answer is usually orientation. Use a path, domain, evidence, or scenario follow-up.

Hiding uncertainty

Ask for assumptions and rejection criteria. A visible unknown is safer than a silent one.

Tool ceremony

Calling every tool every time wastes attention. Use only the call that reduces the next uncertainty.

Prompt leakage

Do not request or publish internal graph structure, private data, system prompts, or credentials. Kohnex usage should remain at the user-facing reasoning layer.

KOHNEX BRAIN / HUMAN EDITION

22. Building a Kohnex Practice

A practice is a repeatable habit with a purpose, a cadence, and a review loop.

The weekly loop

Choose one unresolved question. Run a broad research session. Select one hypothesis. Test it during the week. Review the result and update the question.

The monthly loop

Review which modes the team uses, which prompts produce action, where evidence is weak, and which decisions repeatedly return. Improve the workflow before adding complexity.

The maturity path

Explorer: uses Kohnex for occasional research.
Practitioner: chooses modes and documents follow-ups.
Team: shares prompts, reviews evidence, and records decisions.
System: integrates Kohnex into product, research, security, or engineering workflows with governance.

Measure the practice

Useful measures include time to orient, number of assumptions made explicit, quality of experiments, rate of follow-up completion, and decisions reversed after new evidence. Do not measure only the number of queries.

The central habit

Ask a better next question. That is the simplest description of a mature Kohnex practice.

KOHNEX BRAIN / HUMAN EDITION

23. Case Studies

The following cases are composite, illustrative scenarios. They are designed to teach workflows, not to disclose private Kohnex outputs or claim customer results.

Case 1: A research team enters a new field

The team knows distributed systems and wants to understand privacy-preserving learning. Explore mode maps concepts. Domain inspection creates a vocabulary list. Mechanistic mode explains the major approaches. Evidential mode separates established claims from attractive analogies. The result is a reading plan and a small benchmark, not a confident summary of an entire field.

Case 2: A founder explores a product direction

The founder asks for opportunities at the intersection of agent governance and security. Out-of-Box generates combinations. Reductionist removes features until one painful job remains. Scenario compares adoption paths. The first action is ten interviews, not a product build.

Case 3: An engineering team handles failure

The team asks how to recover from partial dependency failure. Systems maps feedback. Regeneration explores repair. Mechanistic describes

state transitions. Scenario compares degradation and failover. The output becomes a game-day experiment with rollback criteria.

Case 4: A security team triages uncertainty

The team has an unusual pattern but incomplete telemetry. Diagnostician separates possible causes. Predator searches attack paths. Skeptic searches benign explanations. Evidential defines the next collection step. The team makes a reversible containment decision while the investigation continues.

Case lesson

In each case, Kohnex does not make the decision alone. It improves orientation, comparison, and the quality of the next test.

KOHNEX BRAIN / HUMAN EDITION

24. Prompt Library

Use these as starting points. Add your context, constraints, and desired output.

Research

```
Use Explore mode to orient me to [field]. Connect [domain A],
[domain B], and [domain C]. Separate established ideas, analogies,
open questions, and the first five things I should investigate.
```

Product discovery

```
Use Out-of-Box mode to explore product opportunities for [user] who
struggles with [job]. Connect [domains]. For each opportunity include
the painful problem, narrow wedge, alternative solutions, risk, and
one-week validation experiment.
```

Architecture

```
Use Systems and Mechanistic modes to design [system] under [constraints].
Map dependencies, feedback, failure modes, recovery, observability,
and the trade-offs between the top three designs.
```

Security

```
Use Predator and Evidential modes for a defensive analysis of [authorized system]. Identify attack hypotheses, evidence needed, containment options, false-positive risks, and a safe next test.
```

Hypothesis review

```
Use Skeptic mode to challenge this proposal: [proposal]. List hidden assumptions, alternative explanations, disconfirming evidence, and the smallest experiment that would change our confidence.
```

Decision memo

```
Turn this research into a decision memo. Include the decision, options, criteria, evidence, uncertainty, dissenting view, recommendation, reversible next action, and review date.
```

KOHNEX BRAIN / HUMAN EDITION

25. Exercises

Exercise 1: Rewrite a question

Take “How do I improve my system?” Add objective, context, domains, mode, output, and success criteria.

Exercise 2: Mode contrast

Ask the same question in Focus, Explore, Skeptic, and Mechanistic. Record how the answer changes and which answer is useful for which decision.

Exercise 3: Connection boundary

Choose one cross-domain analogy. Write why it is helpful, where it breaks, and what experiment could test the transferable part.

Exercise 4: Product wedge

Generate three product opportunities. Remove features until each has one user, one job, and one measurable outcome.

Exercise 5: Incident uncertainty

Write an incident memo with observed facts, hypotheses, evidence for and against, immediate containment, and a reversible next test.

Exercise 6: Decision review

Take a past decision. Reconstruct the question, mode, evidence, uncertainty, and review date. Identify the question that should have been asked earlier.

Exercise 7: Team practice

Run a twenty-minute session with a colleague. One person owns the question; one challenges the frame. Compare the final action with the initial request.

KOHNEX BRAIN / HUMAN EDITION

26. Glossary and Quick Reference

Activation: the process by which a question brings relevant concepts into attention.
Beamform: a mode for finding intersections between multiple concepts.
Domain: a conceptual area used to focus exploration.
Evidence: support that can be inspected, reproduced, or tested.
Eureka connection: an unexpected relationship worth investigating.
Hypothesis: a claim stated so that evidence can support or reject it.
Mode: a user-facing reasoning lens that changes exploration behavior.
Path: a traceable conceptual route between two concepts.
Prompt contract: the objective, context, domains, mode, and requested output.
Reductionist: a mode that searches for the essential core.
Skeptic: a mode that challenges assumptions and searches for disconfirmation.
Substrate: the structured knowledge layer an agent reasons over.

Quick mode selection

```
Need precision?       Focus
Need discovery?       Explore or Out-of-Box
Need surprise?        Dream
Need intersection?    Beamform
Need root cause?      Diagnostician or Reductionist
Need how?             Mechanistic
Need purpose?         Teleological
Need evidence?        Evidential or Skeptic
Need futures?         Scenario
Need strategy?        Strategist
Need systems?         Systems
Need recovery?        Regeneration
Need defense?         Immune or Predator
```

Final checklist

- Is the question attached to a real decision?
- Are the relevant domains named?
- Is the mode chosen for the reasoning job?
- Did the answer distinguish analogy from evidence?
- What assumption is most important?
- What is the smallest next experiment?
- Who owns the decision and when will it be reviewed?

KOHNEX BRAIN / HUMAN EDITION

27. Prompt Anatomy Workbook

A prompt is not a spell. It is a small research brief addressed to a reasoning system. The quality of the result depends less on decorative language than on whether the prompt makes the work legible: what must be decided, what is known, what is constrained, what kind of exploration is wanted, and what artifact should come back.

The six parts of a useful prompt

Part	Question it answers	Common omission
Situation	What is happening now?	The writer assumes shared context.
Decision	What choice or judgment is pending?	The request becomes a topic instead of a decision.
Boundaries	What is in and out of scope?	The answer grows broader than the team can act on.
Posture	How should the problem be explored?	A creative task receives a premature verdict.
Artifact	What form should the result take?	Useful reasoning is buried in a long response.
Test	How will usefulness be checked?	A plausible answer is mistaken for a good one.

The parts can be short. A compact prompt with six clear sentences is usually stronger than a page of background that never names the decision.

From topic to decision

Start with the sentence, “We need to decide whether to...” If the sentence cannot be completed, the work may still be orientation rather than decision support. That is acceptable; name it accurately.

Topic: “We are interested in reducing support load.”

Research question: “What patterns explain the increase in support load?”

Decision question: “Should we spend the next two weeks improving self-service, changing the product flow, or adding support capacity?”

Each version invites different work. The topic needs vocabulary. The research question needs evidence. The decision question needs options, criteria, trade-offs, and a next action.

Exercise: three levels

Choose a current topic and write all three versions.

```
Topic: _____________________________________________________
Research question: _____________________________________________
Decision question: _____________________________________________
What makes the decision timely? _____________________________
Who owns the decision? _____________________________________
```

Context without a data dump

Useful context has a job. Include facts that change interpretation, constraints that change options, and vocabulary that prevents confusion. Exclude details that are merely interesting or that belong in an approved source document.

Use the context test: if removing a sentence would not change the possible answer, move it to background notes. If the sentence contains confidential material, replace it with a category, range, or abstract description when possible.

Weak context: “Here is everything from the last six months.”

Stronger context: “The service is used by small teams, the costly failure is delayed first response, and the next release cannot change the billing system. Compare operational and product interventions. Treat the supplied incident summary as an incomplete sample.”

The stronger version tells the reader what matters and what not to assume.

Choosing the posture

Name the work phase before naming a mode. Are you trying to widen the possibility set, explain a mechanism, challenge a claim, compare models, or select a safe experiment? Then choose a posture that serves that phase.

Worked example: a migration question

Suppose a team asks whether to move a reporting workload to a new platform.

Exploration prompt:

```
We are orienting to options for a reporting workload used by finance and
operations. Explore the relevant architectural patterns, the questions that
distinguish them, and the vocabulary a non-specialist decision owner needs.
Do not recommend a platform yet. Separate established practice from analogy
and list the five most important unknowns.
```

Challenge prompt:

```
Challenge the proposal to move the reporting workload to a new platform.
List the assumptions that must be true, plausible reasons the current system
appears slow, evidence that would disconfirm each explanation, and one cheap
test for the highest-risk assumption. Do not treat migration as the default.
```

Decision prompt:

```
Turn the research into a decision memo. Compare keep, optimize, and migrate.
Use reliability, total effort, reversibility, data governance, and time to
value as criteria. Mark facts, interpretations, and open questions
separately.
Recommend only a bounded next step, with an owner and review date.
```

The subject did not change. The reasoning job changed, so the prompt changed.

Output contracts

An output contract prevents a session from ending with an attractive essay. Specify headings, length, distinctions, and stopping rules.

```
Return:
1. A one-sentence answer to the decision question.
2. A table of three options and five criteria.
3. Evidence, interpretation, and uncertainty in separate sections.
4. The strongest dissenting view.
5. One reversible next action with owner, threshold, and date.
6. Questions that cannot be answered from the supplied material.
```

Avoid asking for certainty when the evidence cannot support it. “State confidence and what would change it” is more useful than “be certain.”

Prompt failure clinic

Symptom	Likely cause	Repair
Generic answer	No decision or audience	Name the decision and reader.
Endless list	No prioritization rule	Ask for ranking criteria and a stopping point.
Confident invention	Evidence boundary is vague	Require source labels and unknowns.
Clever analogy	No transfer test	Ask where the analogy breaks and how to test it.
Overly cautious answer	No action threshold	Define a reversible step and acceptable risk.
Team disagreement	Different jobs hidden in one prompt	Split orientation, critique, and choice.

The repair ladder

When a first result is weak, repair in this order:

1. Restate the decision.
2. Remove irrelevant context.
3. Add the audience and consequence.
4. Name the reasoning posture.
5. Add evidence and uncertainty labels.
6. Specify the artifact.
7. Add a review or experiment.

Do not immediately add adjectives such as “deep,” “brilliant,” or “comprehensive.” They often increase length without improving the task.

Blank prompt builder

```
Situation:
_____________________________________________________________

Decision or question:
_____________________________________________________________

Audience and consequence:
_____________________________________________________________

Known facts and source boundary:
_____________________________________________________________

Constraints, exclusions, and time horizon:
_____________________________________________________________

Reasoning posture and why:
_____________________________________________________________

Required output headings:
_____________________________________________________________

Confidence language and unknowns:
_____________________________________________________________

Next action, owner, threshold, and review date:
_____________________________________________________________
```

Prompt review checklist

- [ ] The decision or research job is explicit.
- [ ] The audience and consequence are named.
- [ ] Facts are distinguished from assumptions.
- [ ] Scope and exclusions are clear.
- [ ] The selected posture fits the next job.
- [ ] The output has a usable structure.

- [ ] Unknowns and disconfirming evidence are requested.
- [ ] The proposed action is bounded and reviewable.

Practice assignment

Take a prompt from your team that produced a vague answer. Keep the topic, but rewrite it three ways: orientation, challenge, and decision memo. Have a colleague predict which output would be most useful before running the sessions. Compare the prediction with the result. Record not only which prompt worked, but what changed in the work.

Advanced anatomy: the hidden contract

Every prompt carries a hidden contract about authority. It may imply that the responder should educate, recommend, decide, or act. Make that contract explicit. A research assistant can organize possibilities; an accountable person decides whether a change is acceptable. A tool can suggest a test; an authorized operator approves and supervises it.

Add one sentence when the boundary matters: “You are advising the decision owner, not making the decision.” Add another when action is possible: “Do not take operational action; return a proposed sequence for review.” This prevents a useful exploration from being mistaken for authorization.

Audience transformation

The same analysis often needs three versions. The researcher needs detail and open questions. The decision owner needs trade-offs and a recommendation. The operator needs steps, stop conditions, and signals. Ask for the transformation explicitly rather than asking every audience to read the same document.

```
Transform this research into three layers:
1. Research notes: claims, evidence, alternatives, and gaps.
2. Decision brief: options, criteria, recommendation, and uncertainty.
3. Operating card: owner, sequence, signal, stop condition, and rollback.
Keep the meaning consistent and mark where detail was intentionally omitted.
```

Prompt version record

```
Prompt name: _____
Version date: _____ Owner: _____
Decision context: _____
What changed from the previous version: _____
Observed benefit: _____
New failure or limitation: _____
Keep, revise, or retire: _____
```

Versioning is valuable when the surrounding context is recorded. A prompt that worked for one audience, deadline, and evidence boundary may fail elsewhere. Keep the conditions beside the wording.

KOHNEX BRAIN / HUMAN EDITION

28. Research Notebook and Evidence Practice

Good research is partly a thinking activity and partly a record-keeping activity. The notebook protects the difference between what happened, what someone inferred, what a tool suggested, and what the team decided to do. Without that separation, a hypothesis slowly becomes a fact through repetition.

The notebook as a chain of custody

A useful notebook lets a later reader answer five questions:

1. What question was being asked?
2. What was known at the time?
3. What reasoning connected the material to the conclusion?
4. Who challenged it, and what remained uncertain?
5. What happened after the decision?

The notebook does not need to be ornate. A dated Markdown page, a ticket, or a reviewed document can work. The essential property is that the record is understandable to someone who was not in the room.

Evidence classes

Class	Meaning	Example wording
Observation	Something directly seen or measured	“The queue reached the stated threshold at 14:10.”
Report	A statement from a person or approved source	“The operator reported that retries began after deployment.”
Interpretation	A meaning assigned to observations	“The timing is consistent with a retry storm.”
Analogy	A comparison that may suggest a question	“This resembles a feedback loop in another system.”
Hypothesis	A claim awaiting a discriminating test	“The timeout policy may be amplifying load.”
Decision	A chosen action under uncertainty	“We will reduce the retry budget for one cohort.”

Use verbs that reveal status: observed, reported, inferred, suspected, tested, rejected, decided. Avoid presenting all statements in the same grammatical voice.

A claim card

Write one card for each material claim. A material claim is one that could change the decision, allocate resources, affect a person, or create a safety or security consequence.

```
Claim ID or short name: _____
Claim: _____
Why it matters: _____
Evidence location: _____
Evidence type: observation / report / interpretation / analogy
Date and collector: _____
Alternative explanations: _____
What would disconfirm it: _____
Freshness or coverage limitation: _____
Current confidence: high / medium / low / unknown
Decision consequence: _____
Next check and owner: _____
```

The short name is for the notebook, not a substitute for the claim. Write enough that a reviewer can understand it without decoding private shorthand.

Source notes

For each source, record why it was consulted and what it can legitimately support. A source may be authoritative for one question and weak for another. A marketing page can establish a published feature claim; it may not establish reliability under your workload.

```
Source title or approved reference: _____
Owner or publisher: _____
Date accessed: _____
Question it can answer: _____
Question it cannot answer: _____
Relevant passage, measurement, or observation: _____
Conflict with other sources: _____
Permission and retention note: _____
```

Do not paste restricted source material into a wider working note. Store it where it is authorized and refer to the approved location.

The evidence matrix

An evidence matrix makes gaps visible before a meeting.

Claim	Decision impact	Supporting material	Contradicting material	Missing check	Owner
_____	high / med / low	_____	_____	_____	____
_____	high / med / low	_____	_____	_____	____
_____	high / med / low	_____	_____	_____	____

Prioritize by decision impact, not by how easy a source is to find. A low-impact claim can remain provisional. A high-impact claim deserves a deliberate attempt to challenge it.

Worked illustrative example: a delayed notification

An operations team notices that users receive notifications late. The first conversation says, “The provider is unreliable.” The notebook should resist that premature conclusion.

Observation: On three recorded occasions, the notification appeared more than ten minutes after the triggering event.

Report: An operator says the delay began after a configuration change.

Alternative explanations: queue saturation, clock mismatch, downstream batching, provider delay, or a display issue.

Discriminating check: Compare event time, enqueue time, delivery acknowledgement, and display time for a small approved sample. The check is useful because each explanation predicts a different interval.

Bounded action: Route a low-risk test cohort through a conservative retry and logging configuration. Define a rollback and a success threshold before the change.

The notebook does not need to decide the root cause in advance. It needs to preserve competing explanations long enough to learn.

Confidence is a statement about support

Confidence is not a mood and not a property of elegant prose. Rate it according to evidence quality, relevance, independence, freshness, and remaining alternatives.

Rating	Use when	Required companion
High	Direct, relevant, current, checked support exists	State the remaining limitation.
Medium	Signals agree but coverage or causality is incomplete	Name the gap and next check.
Low	The claim depends on analogy, sparse observation, or unresolved alternatives	Call it a working hypothesis.
Unknown	The needed observation or definition is missing	Ask for the missing input.

Never upgrade a claim merely because several people repeat it. Independent support means the sources are not all copying the same original assertion.

Dissent and rejection logs

Dissent is evidence about the reasoning process. Record the strongest objection in terms its author recognizes, then record whether it changed the plan.

```
Proposal: _____________________________________________
Strongest dissent: ____________________________________
Assumption challenged: ________________________________
Evidence supplied: ____________________________________
Response or test: _____________________________________
What changed: _________________________________________
What remains open: ____________________________________
```

Rejected paths deserve a page when they were plausible, expensive, or likely to reappear. “Rejected” is not “forgotten.”

Review meeting protocol

Before the meeting, circulate the decision page and evidence matrix. During the meeting, ask the question owner to state the decision in one sentence. Ask the challenger to select one high-impact claim. Ask the domain reviewer to identify the most consequential technical limitation. End by naming the smallest safe action, owner, threshold, and review date.

Do not use the meeting to rewrite history. If new information appears, date it and mark the change.

Notebook checklist

- [ ] The question and deadline are dated.
- [ ] The decision owner is named.

- [ ] Material claims have cards.
- [ ] Evidence types are separated.
- [ ] Alternative explanations are recorded.
- [ ] A disconfirming check exists for important claims.
- [ ] Dissent and rejected paths are preserved.
- [ ] Sensitive source material remains in its approved location.
- [ ] The decision, action, threshold, and review date are explicit.

Blank research page

```
Date: _____  Session owner: _____
Question: _____
Decision deadline: _____
Audience: _____

What we know:
1. _____
2. _____
3. _____

What we think:
1. _____
2. _____

What we do not know:
1. _____
2. _____

Best disconfirming check: _____
Bounded next action: _____
Owner: _____ Threshold: _____ Review: _____
```

When evidence conflicts

Conflicting evidence is not automatically a problem with the evidence. It may reveal different populations, time periods, definitions, measurement methods, or causal stages. Before choosing a side, align the claims.

Use a conflict card:

```
Claim A: _____________________________________________________
Claim B: _____________________________________________________
Do they use the same definition? ____________________________
Same population or scope? ___________________________________
Same time window? ___________________________________________
Same measurement method? ____________________________________
Most credible difference: ____________________________________
Additional check: ____________________________________________
```

Worked conflict

One report says response time improved; operators say the service feels slower. Both may be right if the report measures server processing while operators experience queueing and display delay. The repair is not to average the statements. Define the user-visible interval, locate each measurement, and add the missing stages.

Evidence freshness

Evidence has a shelf life that depends on the claim. A stable definition may remain useful for years. A capacity measurement may become stale after a release, traffic change, or policy change. Record the event that would invalidate a source, not only an arbitrary expiry date.

```
Claim: _____________________________________________________
Fresh as of: _______________________________________________
Likely invalidating events: ________________________________
Review trigger: ____________________________________________
Current owner: _____________________________________________
```

This makes research operational. The team knows when a conclusion deserves another look instead of treating every review date as ceremony.

From notebook to decision record

At the end of a cycle, compress the notebook into a decision record without deleting the supporting trail. The short record should contain the decision, alternatives considered, decisive evidence, unresolved uncertainty, dissent, action, and review date. Link to the longer pages by stable internal references appropriate to the project.

Ask a reader who missed the work to summarize the decision after reading only the short record. If they cannot identify the owner, reason, or next check, the compression removed too much.

A weekly evidence clinic

Teams benefit from a short recurring clinic that is separate from urgent decision meetings. Bring one claim card, one rejected path, and one recent result. The clinic asks whether the evidence standard was appropriate, whether the team searched for alternatives, and whether the record would help someone reproduce the reasoning without receiving restricted source material.

Use a rotating reviewer so the same person does not become the permanent gatekeeper. The reviewer should not rewrite the researcher’s

conclusion. They should ask questions that expose missing definitions, stale support, confounded measurements, and unowned follow-up.

Twenty-minute clinic agenda

Minutes	Question
0-3	What decision did this claim influence?
3-7	What is directly observed, and where is it recorded?
7-11	What alternative explanation remains plausible?
11-15	What check would most change confidence?
15-18	Is the current action proportionate to support?
18-20	Who updates the record, and by when?

The clinic should end with a small number of corrections. If every claim receives a new investigation, the practice becomes a bottleneck. Prioritize claims by consequence and uncertainty.

Measurement traps

Measurements become misleading when the unit changes, the denominator disappears, or an average hides a harmful tail. Ask whether the measure is counting events, people, requests, time, or cost. Ask which cases were excluded and whether the exclusions are related to the outcome. Ask whether an apparent improvement came from a changed definition.

```
Measure: _____________________________________________________
Unit: ________________________________________________________
Denominator: _________________________________________________
Included cases: ______________________________________________
Excluded cases and why: ______________________________________
Average or distribution: _____________________________________
Possible confounder: _________________________________________
Decision relevance: __________________________________________
```

An evidence notebook should make these questions ordinary. They are not accusations; they are part of understanding what a number can and cannot support.

Closing a research cycle

Close a cycle when the decision is made, the experiment is complete, or the team has identified a blocker that requires another authority. Write the closure reason. A closed page can still be reopened when a trigger occurs, but the trigger should be named. This prevents old hypotheses from quietly becoming permanent background beliefs.

KOHNEX BRAIN / HUMAN EDITION

29. Mode-by-Mode Field Guide

The modes are a practical vocabulary for changing posture. They do not certify a conclusion, and they do not replace domain expertise. Use them to choose the kind of work a session should perform, then inspect the result with ordinary professional judgment.

A quick selection rule

Ask, “What must be different when this session ends?” If the answer is “we need more possibilities,” widen. If it is “we need to choose,” compare. If it is “we need to know whether a claim survives challenge,” test. If it is “we need to repair a system,” diagnose, model, and recover.

Core exploration modes

Mode	Best for	Ask for	Watch for
Focus	Narrow, precise work	Definitions, priorities, exact output	Premature closure
Explore	Orientation and vocabulary	Neighbors, boundaries, competing views	Scope that never narrows
Out-of-Box	Novel combinations	Unusual pairings and useful transfer	Novelty without a user or test
Dream	Serendipitous discovery	Surprising connections and questions	Decorative imagination
Intuitive	Fast pattern triage	Initial patterns labeled as provisional	Unchecked familiarity
Beamform	Intersection finding	Shared structure across selected areas	Forced similarity
Swarm	Many perspectives	Independent views and convergence points	Agreement mistaken for truth
Predator	Adversarial search	Weak points, attack paths, failure incentives	Unauthorized or sensational testing
Regeneration	Repair and recovery	What can be restored, replaced, or isolated	Recovery without verification
Hibernate	Low-resource continuity	Minimum viable operation and preservation	Saving resources while missing danger
Immune	Defensive patterning	Threat signals, boundaries, response	Threat-only framing

Mode	Best for	Ask for	Watch for
		tiers	
Photosynthesis	Resource conversion	Inputs, transformation, stored value	Metaphor replacing measurement
Quantum	Keeping options open	Plausible states and collapse criteria	Indefinite postponement

Field notes

Focus. Use when the team has too many interpretations of one term or needs a precise deliverable. Ask for a definition, exclusions, and a short answer. Pair with Evidential to prevent a precise answer to a false premise.

Explore. Use at the beginning of unfamiliar work. Ask for a map in plain language, neighboring concepts, disagreements, and a reading plan. End with a narrowing question; otherwise exploration becomes a comfortable substitute for action.

Out-of-Box. Use after the job is defined. Require each idea to name a user, transferable mechanism, break point, and cheap test. A surprising connection is a lead, not evidence.

Dream. Use for ideation or recovery from a stuck frame. Capture the ideas without judging them for a short interval, then switch to Reductionist or Skeptic. Do not carry Dream language into an operational recommendation without translation.

Intuitive. Use for triage when speed matters and stakes are low enough for a later check. Record the initial impression before gathering support. This makes calibration possible.

Beamform. Use when two or more areas may share a structure. Ask for the common pattern, the non-transferable details, and a test. It is

particularly useful for generating hypotheses across product, engineering, and operations.

Swarm. Use when a decision benefits from independent perspectives. Give each perspective the same question before asking for convergence. Preserve minority views; consensus is a coordination result, not proof.

Predator. Use only for authorized defensive analysis. Ask what an adversary would exploit, what evidence would distinguish threat from benign behavior, and what containment is reversible. Pair with Immune so the work produces protection rather than fear.

Regeneration. Use after failure or degradation. Separate restoration of service from restoration of trust. Ask what state can be recovered, what must be rebuilt, and how recovery will be verified.

Hibernate. Use when capacity, attention, or budget is constrained. Identify the minimum safe service, preserved state, wake condition, and expiry of the reduced mode. Low resource is not the same as low consequence.

Immune. Use to design layered defense and response. Ask what is recognized, blocked, isolated, learned from, and escalated. Require false-positive analysis.

Photosynthesis. Use when a team has scattered inputs but needs durable value. Ask what raw material becomes what stored capability, what energy it costs, and what by-products or waste appear.

Quantum. Use before an irreversible choice when several states remain plausible. Define the evidence or deadline that will collapse the choice. This is disciplined uncertainty, not a license to avoid commitment.

Logic modes

Mode	Best for	Failure to challenge
Inductive	Pattern formation from cases	Overgeneralization
Deductive	Applying explicit rules	A bad or incomplete premise
Systems	Dependencies and feedback	Boundary blindness
Normative	Values and obligations	Hidden value judgments
Algorithmic	Repeatable procedures	Optimizing the wrong goal
Conceptual	Clarifying categories	Abstraction without use

Inductive should show the cases and the limits of the pattern. Ask what sample would break it. **Deductive** should list premises before the conclusion. Ask whether the rule applies in this context. **Systems** should trace inputs, delays, feedback, and failure boundaries rather than drawing a list of components. **Normative** should state whose values and duties are in scope. **Algorithmic** should include stopping, exception, and rollback conditions. **Conceptual** should define categories by what they include, exclude, and enable.

Reasoning modes

Mode	Best for	Strong output
Scenario	Plausible futures	Conditions, signals, and actions
Evidential	Claim checking	Support, alternatives, and tests
Reductionist	Finding essentials	Minimum variables and decision rule
Mechanistic	Explaining how	Causal sequence and boundaries
Teleological	Purpose and function	User or system purpose plus trade-offs
Model-Based	Comparing explanations	Assumptions, predictions, and fit

Scenario is not prediction theater. Give each future a trigger, early signal, exposure, and response. **Evidential** should seek disconfirmation, not just supporting material. **Reductionist** should make the simplification visible and state what it loses. **Mechanistic** should explain timing and interaction, not merely name a component. **Teleological** should challenge assumed purpose by asking who benefits and what outcome counts. **Model-Based** should compare models against observations and admit when the data cannot distinguish them.

Three worked sequences

Product discovery

```
Explore -> Out-of-Box -> Reductionist -> Evidential -> Algorithmic
```

First learn the user's landscape. Then generate alternatives. Reduce each to one job and one measurable outcome. Test the riskiest claim. Finally turn the surviving idea into a small learning procedure.

Incident review

`Diagnostician -> Mechanistic -> Skeptic -> Systems -> Regeneration`

Collect symptoms and competing causes. Explain the likely process. Challenge the attractive story. Inspect surrounding dependencies and feedback. Choose recovery and verification steps.

Architecture decision

`Explore -> Systems -> Model-Based -> Normative -> Scenario`

Orient to patterns. Map dependencies and boundaries. Compare explicit designs. State values and constraints. Examine plausible futures and review triggers.

Mode worksheet

```
Decision or research job: _____________________________________
What must be different at the end: _____________________________
Starting mode: _____________ Why: _____________________________
Companion mode: _____________ Why: ____________________________
Evidence boundary: ____________________________________________
Main failure to watch for: ____________________________________
Switch condition: _____________________________________________
Output artifact: ______________________________________________
```

Mode transitions in practice

Most difficult work fails at a transition rather than inside a mode. A team explores but never narrows. It narrows before understanding. It challenges a proposal but never returns to construction. Treat the transition as a deliverable.

From	To	Carry forward	Leave behind
Explore	Focus	Definitions, options, unknowns	Unbounded curiosity
Out-of-Box	Evidential	Transferable mechanism and test	Novelty as a virtue by itself
Focus	Skeptic	Exact claim and scope	Confidence from precision
Skeptic	Algorithmic	Surviving assumptions and constraints	Objections without action
Mechanistic	Systems	Causal sequence and boundaries	Component-only explanation
Quantum	Decision	Alternatives and collapse criteria	The comfort of indefinite choice
Predator	Immune	Threat conditions and evidence	Adversarial drama
Regeneration	Focus	Recovery objective and verification	Repair as an endless project

At each transition, write a one-sentence handoff: “We explored X; we will now compare Y using Z.” This prevents the next mode from restarting the work from scratch.

Mode calibration exercise

Select ten past sessions. For each, record the chosen mode, the actual job, the output quality, and the failure you observed. Look for mismatches. Perhaps Explore was used when the team needed a decision, or Focus was used when terms were not yet understood. Change the selection rule, not merely the mode label.

```
Session: _____ Job: _____
Chosen mode: _____ Better mode: _____
Useful result: _____
Failure: _____
New selection lesson: _____
```

The exercise turns mode choice into a learnable skill. It also prevents a team from treating mode names as a fixed taxonomy rather than practical tools.

Stop conditions by mode

Focus stops when the definition and requested artifact are stable. Explore stops when the team can name the next three questions. Out-of-Box stops when each idea has a job and test. Skeptic stops when the strongest objection has a response or experiment. Systems stops when normal and degraded paths are described. Scenario stops when triggers and actions are explicit. These stopping rules protect attention and make sessions easier to schedule.

Field checklist

- [ ] The mode serves the next job, not the topic label.
- [ ] The session has a stopping condition.
- [ ] Creative outputs will be challenged before use.
- [ ] Defensive work is authorized and bounded.
- [ ] Minority views and alternatives are retained.
- [ ] A mode switch is planned when the work changes.
- [ ] The final artifact states evidence and uncertainty.

KOHNEX BRAIN / HUMAN EDITION

30. Team Workshop Facilitation Guide

A workshop is a temporary decision system. Its purpose is not to produce the most impressive conversation. Its purpose is to help a group frame a question, expose disagreement, create useful options, and leave with accountable next steps.

Before the room

The facilitator needs a decision owner, a narrow question, a timebox, an audience for the result, and a clear boundary around information. Invite only people who add a perspective, authority, or relevant experience. A large room can hide responsibility.

Send a one-page pre-read containing the question, known facts, open assumptions, constraints, and the decision date. Ask each participant to write one answer privately before the session. This reduces the influence of the first confident speaker.

Pre-work checklist

- [ ] The decision is reversible or the approval path is explicit.
- [ ] The owner can accept or reject the next action.
- [ ] Sensitive material has an approved handling plan.
- [ ] The room has the necessary domain reviewer.
- [ ] A challenger is assigned before the conversation becomes agreeable.

- [ ] The output format is known.
- [ ] The follow-up date is on the calendar.

The 90-minute workshop

Time	Activity	Facilitator output
0-10	Frame the decision	One sentence and success condition
10-20	Silent individual notes	Initial views and assumptions
20-35	Shared facts	Evidence list with gaps
35-50	Broad exploration	Options, questions, alternatives
50-65	Challenge round	Risks, disconfirming checks, dissent
65-78	Compare options	Criteria and trade-offs
78-87	Choose next action	Owner, threshold, rollback
87-90	Read-back	Decision record and review date

The facilitator should protect transitions. Exploration that continues after the choice phase has begun often feels productive while avoiding commitment.

Opening script

“We are not here to prove that one person is right. We are here to improve a decision. We will separate observations from interpretations, preserve a serious dissenting view, and finish with a bounded action. If the question is too broad, narrowing it is a successful outcome.”

This script sets a useful norm: the workshop rewards clarity, not performance.

Framing the question

Write the decision on a visible board. Add the owner, deadline, constraints, and what cannot be changed in this cycle. Ask each participant to complete the sentence, “If we learn one thing today, it should be...” Compare the answers. If they differ, repair the frame before generating options.

Framing card

```
Decision: _____________________________________________________
Owner: _____________________ Deadline: _____________________
Who is affected: ____________________________________________
Success condition: __________________________________________
Non-negotiable constraint: __________________________________
What is explicitly out of scope: ____________________________
```

Silent thinking and equal participation

Use two to five minutes of silent writing before discussion. Ask for one observation, one assumption, one concern, and one possible next check. Read all cards before debate. The goal is not forced equality of speaking time; it is to prevent status from deciding what gets examined.

When a senior person speaks early, invite others to add independent views before responding. When one participant dominates, use a round with a strict time limit. When the room is quiet, ask a specific question rather than “Any thoughts?”

Evidence sorting exercise

Create four columns: observed, reported, inferred, and unknown. Place each material statement in one column. A statement may move later, but

the move must be explained. This exercise is particularly useful when a team has been repeating a diagnosis for several weeks.

Facilitator prompts

- “What did we directly observe?”
- “Who reported this, and what could they see?”
- “What are we inferring?”
- “What would we expect to see if that explanation were true?”
- “What observation would make us reconsider?”

Do not shame an inference for being an inference. Inference is necessary. Hidden inference is the problem.

Generating and narrowing options

During the broad phase, separate option generation from evaluation. Ask for at least one option that changes the process, one that changes the product, one that changes the operating practice, and one that deliberately does less. Then apply criteria.

Option	User value	Effort	Reversibility	Risk	Learning value
_____	1-5	1-5	1-5	1-5	1-5
_____	1-5	1-5	1-5	1-5	1-5
_____	1-5	1-5	1-5	1-5	1-5

Scores do not make judgment objective. They make differences discussable. Ask why two people scored the same option differently.

The dissent round

Assign one person to argue against the leading option, even if they personally support it. The role is to improve the proposal, not to win. Require a specific claim, consequence, and test. “This feels risky” is a useful signal but not a finished objection.

```
Leading option: _____
Strongest objection: _____
Assumption underneath it: _____
Evidence that would support the objection: _____
Evidence that would reduce it: _____
Design change or test: _____
```

If dissent reveals an unbounded risk, stop the decision and escalate. A workshop is not a license to approve what the organization has not authorized.

From insight to action

End with one of four outcomes: decide, test, defer with a named condition, or stop. “Continue discussing” is not an outcome unless it has a specific missing input and date.

Action card

```
Outcome: decide / test / defer / stop
Action: _____
Owner: _____ Support: _____
Start date: _____ Review date: _____
Success threshold: _____
Rollback or stop condition: _____
Evidence to collect: _____
```

Read the card aloud. The owner must confirm the interpretation, and the decision owner must confirm authority.

Remote and hybrid rooms

Use one shared workspace rather than a mixture of private chats and visible notes. Make the timer and current phase visible. Ask remote participants to write first. If a room contains both in-person and remote people, have everyone contribute through the same channel so the remote group is not reduced to an audience.

After the workshop

Send the decision record within one working day. Mark facts, assumptions, dissent, action, and review date. Ask participants to correct factual errors, not rewrite conclusions merely because they dislike the choice. At the review date, compare expected and observed results. Update the notebook and the team playbook.

Facilitation repair table

Problem	Intervention
Topic expands	Return to the decision and remove one branch.
Debate becomes personal	Restate the claim and ask for evidence or a test.
Consensus arrives too quickly	Run a private dissent round.
No one owns action	Ask who has authority and who accepts the risk.
Evidence is inaccessible	Mark the claim unknown and assign retrieval.
The tool fails	Use the ordinary facts-assumptions-options-risks template.

Workshop scorecard

```
Question was clear: yes / partly / no
Evidence and inference were separated: yes / partly / no
Dissent was recorded: yes / partly / no
One owner accepted the action: yes / partly / no
Threshold and rollback were defined: yes / partly / no
Review date scheduled: yes / partly / no
One improvement for next time: _____
```

Difficult participant patterns

Facilitation is not about controlling personalities; it is about making the work observable. When someone speaks in conclusions, ask for the observation underneath. When someone supplies a long history, ask which part changes today’s decision. When someone rejects every option,

ask what test would make one option acceptable. When someone proposes an action outside their authority, ask who must approve it.

Pattern	Useful response
The expert answers before the question is framed	Thank them, park the answer, and finish the frame.
The skeptic rejects without a test	Ask for the smallest observation that could settle the concern.
The optimist skips failure	Request a pre-mortem and a stop condition.
The quiet domain owner says little	Ask one precise question and allow written response.
The decision owner delegates accountability	Clarify who can accept the risk and record the gap.
The group argues over terminology	Define the term, examples, exclusions, and decision relevance.

A 30-minute micro-workshop

When the full format is impossible, use three rounds. First, five minutes of silent framing. Second, fifteen minutes to sort observations, assumptions, and unknowns. Third, ten minutes to choose one next check with an owner and date. Do not try to generate a complete strategy in the short format. The micro-workshop is a triage instrument.

```
Question: _____________________________________________________
Three observations: ___________________________________________
Two assumptions: ______________________________________________
One unknown that matters: _____________________________________
Next check: ___________________________________________________
Owner: _____________________ Review: _________________________
```

Measuring workshop quality

Do not measure a workshop only by the number of ideas. Track whether decisions become clearer, whether important claims receive checks, whether actions have owners, and whether later reviews find fewer surprises. A workshop can produce no new idea and still succeed by stopping an unsafe or unsupported action.

Ask participants one week later: What did you use? What remained unclear? What should the facilitator change? Compare the answers with the original scorecard.

Workshop artifacts that survive the meeting

A conversation disappears unless it leaves a few durable artifacts. Keep the framing card, evidence sort, option comparison, dissent card, and action card. Do not preserve every sentence by default. Excessive notes make it difficult to find the actual decision and can create a false impression that every spoken idea received equal support.

Name each artifact with the date, decision, and status. If a note contains a claim, give it a source or mark it as an assumption. If it contains an action, give it an owner and review date. If it contains a proposed connection, label it as a hypothesis until a test supports it.

Artifact quality check

```
Could a person absent from the meeting identify the decision? ______
Could they see what was known at the time? ____________________
Could they find the strongest dissent? _______________________
Could they tell who acts next? ______________________________
Could they tell when the decision will be revisited? ____________
What should be removed as distracting detail? ___________________
```

Workshop variants

The research kickoff

Use this when the team does not yet understand the field. Spend more time defining terms and less time scoring options. End with a reading plan, three evidence questions, and a date for returning to the decision. Do not force a recommendation from an orientation session.

The design critique

Use this when a proposal exists. Ask the proposer to describe the normal path and a degraded path before discussion. Invite critique by failure boundary, user impact, operational burden, and reversibility. End with changes to the design or a specific test, not a general feeling of confidence.

The incident learning review

Use this after service has stabilized. Establish a factual timeline before assigning causes. Ask what signals were available, what was missed, how the response escalated, and which recovery step reduced harm. Keep the review blameless without making it consequence-free: accountability belongs in the process and system design, not in humiliation.

The governance review

Use this when scope, users, data, or impact changes. Recheck authorization, access, retention, appeal, correction, and human approval. A workflow that was appropriate for low-consequence research may require different controls when connected to a consequential decision.

Facilitation self-review

After each workshop, the facilitator should answer three questions. Where did I let the room move faster than the evidence? Where did I over-control a useful disagreement? What question should have been asked earlier? Share one answer with the next facilitator. A team improves its decision practice when facilitation itself becomes inspectable.

KOHNEX BRAIN / HUMAN EDITION

31. Publishing, Ethics, and Disclosure

A human-facing book about an intelligence tool has two responsibilities. It should make the practice understandable and useful, and it should avoid turning private implementation detail into public material by accident. These responsibilities reinforce each other. Clear boundaries make a book more trustworthy because readers can tell what is demonstrated, what is illustrative, and what remains an opinion.

Describe the experience, not the private machinery

Readers need to know what they can ask, what kind of output to expect, how to inspect it, and where human judgment remains necessary. They do not need undisclosed design details to learn those practices. Prefer descriptions such as “a structured exploration of related concepts” or “a comparison of several reasoning postures” when those are accurate at the user-facing level.

The abstraction test is simple: could a practitioner use the explanation without receiving a blueprint of private internals? If yes, the level is probably appropriate. If no, step back and describe purpose, behavior, boundary, and evaluation instead.

Claim categories

Every meaningful statement in a manuscript belongs to a category.

Category	How to present it
Public fact	Give a source or identify the public reference.
Author interpretation	Use language such as “we interpret” or “a useful way to think.”
Illustrative example	Label it clearly and avoid invented performance claims.
Composite case	State that details are combined or altered.
Observed result	Identify the setting, limits, permission, and evidence.
Recommendation	Explain the reasoning and conditions under which it applies.

The reader should not have to guess whether a story happened. Put the label near the story, not in a distant disclaimer.

Examples and composite cases

An illustrative example is a teaching device. It should use invented names, safe quantities, and a clearly bounded scenario. A composite case combines patterns from multiple situations; say so. Do not add precise dates, outcomes, customer details, or technical fingerprints merely to make a story feel real.

Safe label: “The following is a composite, illustrative scenario. It teaches the workflow and does not report a customer result.”

If a real case is essential, obtain permission for the scope of publication. Permission should cover names, quotations, outcomes, screenshots, and the ability to revise or withdraw sensitive details where applicable. A verbal assumption is not a publication process.

Privacy and data minimization

Before drafting, classify inputs and keep the working manuscript at the least detailed level that supports the lesson. Remove personal names, contact details, unique identifiers, precise timestamps, internal URLs, access paths, and combinations of facts that could re-identify a person or organization.

Redaction is not only blacking out text. A rare job title plus a small location plus an unusual incident may identify someone even when the name is gone. Review the whole passage for linkage.

Store source material in its approved location. The book should contain the lesson, not a duplicate archive of the source.

Security and responsible description

Defensive material can accidentally become operational guidance for misuse. Keep examples at the level needed to understand authorization, detection, containment, review, and recovery. Do not publish access tokens, private access details, undisclosed weaknesses, or step-by-step instructions that would enable unauthorized intrusion.

For security examples, state the authorized scope and the stop condition. A responsible pattern is: identify the asset owner, define the permitted test, minimize data, preserve evidence, prefer reversible containment, and escalate when impact or scope is uncertain.

Intellectual property and attribution

Keep a source ledger while writing. Record the title, creator, publisher, access date, permission status, and the passage or idea used. Paraphrase faithfully; changing a few words does not make a distinctive

expression yours. Obtain permission for substantial quoted material, images, tables, screenshots, and proprietary diagrams.

Attribution is also a reasoning practice. When a concept comes from a field outside the author's expertise, name the source and avoid implying original discovery. A cross-domain connection can be valuable without pretending that the original field said what the book needs it to say.

Generated material and disclosure

If an AI system helped draft, classify, summarize, or connect material, retain human responsibility for the final text. Check every factual statement, especially citations, quotations, historical claims, numerical claims, and statements about product behavior. Generated prose can be fluent while being wrong.

An author note can be concise:

```
This book contains human-reviewed illustrative workflows. Automated tools
may
have assisted with drafting or organization, but the authors remain
responsible
for factual accuracy, permissions, privacy, security, and editorial
judgment.
```

Adjust the note to match the actual process. Do not imply that a tool was used in a way it was not.

Review gates

Use separate review passes because one reviewer rarely sees every risk.

Gate	Reviewer asks
Accuracy	Are claims supported and examples internally consistent?
Privacy	Could a person or organization be identified?
Security	Does the passage disclose an actionable weakness or access detail?
Legal and rights	Are quotations, names, images, and outcomes permitted?
Product	Does the description match the public experience?
Editorial	Is the distinction between fact, analogy, and recommendation clear?

Record reviewer, date, scope, requested changes, and resolution. A review that exists only in memory cannot be audited.

Publication boundary worksheet

```
Passage or asset: _____
Purpose in the book: _____
Claim category: _____
Source or permission: _____
Could it identify a person or organization? _____
Could it reveal restricted design information? _____
Could it enable misuse? _____
Safer abstraction or replacement: _____
Required reviewer: _____
Decision: keep / revise / remove
```

Corrections and accountability

A responsible book has a correction path. Publish a contact or editorial route appropriate to the project. When a material error is found, record the issue, assess affected passages, correct the source and rendered editions, and explain the correction when readers could otherwise rely on the old claim.

Do not quietly change a case study from “observed” to “illustrative” after criticism. Explain the change and what evidence was missing. Trust grows when uncertainty is made visible.

Final preflight

- [ ] Every example is labeled as sourced, observed, illustrative, or composite.
- [ ] Names, identifiers, screenshots, and outcomes have permission or are removed.
- [ ] Claims are sourced or clearly presented as interpretation.
- [ ] Generated material was checked by a responsible human.
- [ ] Security descriptions remain authorized and non-operational for misuse.
- [ ] The manuscript contains no confidential implementation detail.
- [ ] Source, permission, and review records are complete.
- [ ] The rendered PDF and web version were reviewed, not only the source files.
- [ ] A correction route exists.

Ethics of persuasive writing

A book can mislead without containing a false sentence. Selective examples, dramatic headings, unjustified precision, and confident pacing

can create an impression stronger than the evidence. Ethical writing includes proportion: a small illustrative exercise should not be surrounded by language that implies a guaranteed transformation.

Prefer bounded verbs. “Can help a team examine” is different from “solves.” “Suggests a question” is different from “discovers the truth.” “In this example” is different from “in practice.” These are not timid substitutions; they tell the reader how much weight to place on the material.

Persuasion audit

```
Most persuasive claim: _____________________________________
What evidence supports it: _____________________________________
What evidence is absent: _____________________________________
Could the example be mistaken for a result? _____________________
Could the heading overstate the section? _____________________
Safer wording: _____________________________________
```

Accessibility and reader dignity

Ethics includes how readers are treated. Define specialized terms before using them. Do not assume every reader has the same technical access, budget, language background, or authority. Offer a tool-independent fallback for every important workflow. Use examples that do not make a person or group the punchline of a failure story.

If the book discusses safety, security, or high-impact decisions, make the limitation visible near the advice. A buried disclaimer is not a meaningful safeguard.

Release packet

Before release, assemble one packet containing the final source, rendered outputs, source ledger, permissions, review sign-offs, disclosure decisions, correction contact, and a short change log. Inspect the rendered artifacts because formatting can reveal text omitted from the source review, including captions, alternate text, footers, and metadata.

```
Release version: _____
Source reviewed by/date: _____
Rendered formats reviewed by/date: _____
Permissions complete: yes / no
Privacy review: _____
Security review: _____
Correction contact: _____
Known limitation to disclose: _____
```

KOHNEX BRAIN / HUMAN EDITION

32. A Thirty-Day Practice Journal

This journal turns the ideas in the book into a habit. Each day asks for a small, observable action. Do not try to redesign your organization in a month. The aim is to learn a repeatable loop: frame a question, choose a posture, inspect evidence, make a bounded move, and review what happened.

How to use the journal

Choose one low-to-medium consequence decision. Keep the same decision or research theme for the first two weeks so that you can compare sessions. Use a separate approved location for any restricted source material. The journal should contain questions, abstractions, observations, decisions, and links to approved records, not copied sensitive content.

At the end of each day, write for ten minutes. A short honest entry is better than a polished retrospective written from memory.

Week 1: Frame the work

Day 1: Choose the decision

Name one decision with an owner and a date. State why it matters now and what happens if nothing changes.

```
Decision: _____________________________________________________
Owner: _____________________ Deadline: _____________________________
Cost of delay: _____________________________________________________
Why this is safe enough for practice: _____________________________
```

The next month should preserve the same promise: make the question clearer, make the uncertainty visible, and make the next responsible action easier to review.

Small records compound. A dated question today becomes a better comparison next month, a clearer handoff next quarter, and a more honest account when results differ from expectations now.

Day 2: Write the three questions

Write the topic, research question, and decision question. Circle the one you will work on today. Note what would make you switch.

Day 3: Set the boundary

List included domains, excluded topics, approved sources, and prohibited inputs. Write the audience for the eventual artifact.

Day 4: Gather baseline observations

Record three direct observations and three assumptions. Do not interpret them yet. Mark where each observation came from.

Day 5: Choose the first posture

Select a mode based on the job. Write why another mode would be wrong today. Run a short session and save the output with a date.

Day 6: Inspect the output

Mark every statement as observation, report, interpretation, analogy, hypothesis, or unknown. Record one useful result and one unsupported leap.

Day 7: Weekly review

What became clearer? What became more uncertain? Which question should be narrowed? What will you stop doing next week?

```
Most useful discovery: _____
Most important uncertainty: _____
Question for week 2: _____
Practice to stop: _____
```

Week 2: Compare and challenge

Day 8: Run a mode contrast

Ask the same bounded question in Focus, Explore, Skeptic, and Mechanistic postures. Compare what each reveals and hides.

Day 9: Build a claim card

Choose one material claim. Record support, alternatives, what would disconfirm it, and the decision consequence.

Day 10: Search for disconfirmation

Spend a fixed period looking for evidence against the leading explanation. If you find none, record the search boundary rather than claiming proof.

Day 11: Find a useful connection

Use a cross-domain analogy only to generate a question. Write why it helps, where it breaks, and the cheapest safe test.

Day 12: Reduce the problem

Ask what minimum variables must be true for the decision. Remove one attractive but nonessential branch. Record what the simplification loses.

Day 13: Compare models

Write two plausible explanations or designs. For each, list assumptions, predictions, costs, and an observation that would distinguish them.

Day 14: Second weekly review

Rate your current confidence: high, medium, low, or unknown. Explain the rating in two sentences. Name the next observation that could change it.

Week 3: Make a bounded move

Day 15: Define the experiment

Choose the smallest test that could change the decision. State baseline, cohort or scope, duration, success threshold, and rollback.

```
Hypothesis: _____
Test: _____
Baseline: _____
Success threshold: _____
Stop or rollback condition: _____
Owner and date: _____
```

Day 16: Check authorization and harm

Ask who could be affected, what could go wrong, and who can stop the test. Confirm the data and operational boundary.

Day 17: Run the pre-mortem

Imagine the test failed or caused harm. List five plausible reasons. Add one prevention or detection measure for each.

Day 18: Run or schedule the test

Do the smallest authorized version. If the test cannot run, record the blocker and what evidence would remove it. A blocked experiment is still a useful result when documented.

Day 19: Record observations promptly

Write what happened before discussing why. Capture exceptions, timing, and deviations from the plan.

Day 20: Interpret with alternatives

Generate at least two explanations for the result. Which explanation is supported? Which remains possible? What did the test fail to measure?

Day 21: Third weekly review

Did the test change the decision? Did it change the confidence without changing the decision? Record the difference. Update the claim card and decision page.

Week 4: Share and institutionalize

Day 22: Draft the one-page memo

Include decision, options, criteria, evidence, uncertainty, dissent, recommendation, action, owner, and review date. Keep facts separate from interpretation.

Day 23: Ask for a challenge review

Give the memo to someone who did not run the session. Ask them to find one hidden assumption, one missing alternative, and one claim that needs better support.

Day 24: Revise without erasing history

Make corrections visible. Preserve the original claim and date when it matters for learning. Record why the conclusion changed.

Day 25: Teach the loop

Explain the practice to a colleague in five minutes: question, posture, evidence, action, review. Ask them to give an example from their work.

Day 26: Create a reusable template

Keep only fields you actually used. Add one field that would have prevented a mistake this month. Remove decorative complexity.

Day 27: Review the team boundary

Ask who may access inputs, who may act on outputs, who approves high-impact choices, and how corrections are distributed. Record unresolved governance questions.

Day 28: Write the case note

Describe the starting question, the useful turn, the failure or surprise, the evidence, and the result. Label it as personal experience or illustrative teaching material. Do not add invented precision.

Day 29: Design the next month

Choose one capability to deepen: prompt quality, evidence discipline, mode switching, facilitation, or experiment design. Define a small metric and a weekly ritual.

Day 30: Close the loop

Complete the scorecard and write a letter to your future self. State what you will keep, stop, and question.

Thirty-day scorecard

```
Days written: ______ / 30
Questions narrowed: ______
Material claims documented: ______
Disconfirming checks attempted: ______
Experiments run: ______
Decisions with owners and dates: ______
Dissenting views preserved: ______
Corrections made visible: ______

Keep: _____________________________________________________
Stop: _____________________________________________________
Question next month: ______________________________________
```

Daily blank page

```
Date: _____ Day: _____
Today I am trying to learn: _____
Mode or posture and why: _____
Observation: _____
Interpretation: _____
Unknown: _____
Best challenge: _____
Smallest next action: _____
Owner: _____ Review date: _____
What I will not claim yet: _____
```

The final habit

The practice is working when the team becomes faster at saying what it does not know, better at finding the next useful observation, and more disciplined about matching action to evidence. The goal is not to remove judgment. It is to make judgment visible enough to improve.

Optional depth days

If a day is interrupted, do not restart the month. Use one of these depth days instead. They are designed to strengthen the habit without changing the decision theme.

Depth day A: replay. Take yesterday's notes and mark the first point where an assumption entered. Ask whether the assumption was necessary, visible, and testable.

Depth day B: translation. Turn the same conclusion into a researcher note, a decision brief, and an operator card. Check that each version preserves uncertainty.

Depth day C: adversarial kindness. Write the strongest version of the opposing view. Then identify the smallest experiment that could support or weaken it. The goal is not to defeat a colleague; it is to improve the choice.

Depth day D: recovery. Imagine the next action partially fails. Write what state must be preserved, who is notified, what is rolled back, and how the team knows recovery worked.

Depth day E: governance. Recheck the audience, authorization, data boundary, retention, and correction path. A good reasoning practice can become unsafe when its scope expands without review.

End-of-month conversation

Invite the decision owner and one challenger to a twenty-minute review. Read the original question, the final decision, and the observed result. Ask what the team would do earlier, later, or differently. Preserve one quote from the conversation only if permission is clear; otherwise summarize the lesson without attribution.

```
Original belief: _____________________________________________
What happened: _____________________________________________
Where we were right: _____________________________________________
Where we were wrong: _____________________________________________
What the process caught: _____________________________________________
What it missed: _____________________________________________
Next practice to test: _____________________________________________
```

The journal ends, but the loop does not. Choose the next decision, reset the page, and carry forward only the practices that improved clarity or action.

A compact continuation card

Use this card after the first month when a full journal entry is not needed. It keeps the essential discipline alive in five minutes.

```
Decision or question: _____
What changed since the last review: _____
New observation: _____
Current belief and confidence: _____
Best alternative explanation: _____
Smallest useful next check: _____
Owner: _____ Threshold: _____ Date: _____
What would make us stop or reverse: _____
```

Monthly practice review

At the end of each month, compare the practice with the work it is meant to serve. Did the records make decisions faster, safer, or more reversible? Did the team spend too much time documenting low-consequence questions? Did a template clarify a disagreement or merely add administration? Keep the fields that changed behavior and remove the fields that only create the appearance of rigor.

Invite one person who uses the records and one person who does not. The first can describe utility; the second can reveal friction. Ask both to identify one missing safeguard and one unnecessary step. A mature practice is not the one with the most paperwork. It is the one that reliably helps people notice what matters before they act.

```
Practice that improved judgment: _____
Practice that added friction: _____
Safeguard to add: _____
Field or ritual to remove: _____
Next month decision theme: _____
```

KOHNEX BRAIN / HUMAN EDITION

Appendices

These appendices turn the book into a working field guide. They are intentionally tool-agnostic. Examples are illustrative and must be checked against the policies, people, data, and laws of the actual project.

Appendix A: First 30 Days

Days 1–3: establish the boundary

Choose one reversible decision. Name the owner, audience, deadline, acceptable risk, and data boundary. Read the protected-information policy with the team. Decide which information may enter a session and which must remain in approved systems. Create a folder for questions, evidence, decisions, experiments, and rejected paths.

Days 4–7: learn the loop

Run two short sessions on low-consequence topics. Use Explore for orientation, Skeptic for challenge, and Evidential for next steps. Ask a colleague to review the notes for invented facts, unmarked analogies, and missing ownership. Do not optimize prompts yet. Learn where the workflow breaks.

Days 8–14: produce one artifact

Select a real decision and create a one-page memo. Include options, criteria, evidence, uncertainty, dissent, reversible action, owner, and review date. Invite a domain reviewer. If the memo cannot distinguish evidence from interpretation, return to the research question rather than decorating the document.

Days 15–21: run a small experiment

Choose the least expensive test that can change the decision. Define a threshold and rollback before starting. Record the baseline, conditions, exceptions, and result. A negative result is valuable when it prevents a larger investment.

Days 22–30: institutionalize learning

Review the month. Count decisions with owners, claims with sources, experiments completed, and assumptions retired. Remove templates nobody uses. Keep the smallest workflow that improves decision quality. Agree on a weekly review and a monthly governance check.

Thirty-day scorecard

```
Decisions framed: ____
Sessions completed: ____
Material claims checked: ____
Experiments run: ____
Reversible actions: ____
Unresolved risks with owners: ____
One practice to keep: _____________________
One practice to remove: _____________________
```

Appendix B: Mode Decision Matrix

Need	Start	Useful companion	Watch for
Vocabulary	Explore	Focus	scope that never narrows
Precision	Focus	Evidential	premature closure
Novel combinations	Out-of-Box	Skeptic	attractive but empty analogy
Serendipity	Dream	Reductionist	ideas without a job
Intersections	Beamform	Model-Based	forced connection
Root cause	Diagnostician	Mechanistic	single-cause bias
Adversarial review	Predator	Evidential	unauthorized testing
Defensive posture	Immune	Systems	threat-only framing
Repair	Regeneration	Scenario	recovery without verification
Low resource	Hibernate	Focus	under-investigating risk
Multiple possibilities	Quantum	Decision owner	indefinite postponement
Collective view	Swarm	Skeptic	conformity mistaken for truth
Rule from examples	Inductive	Evidential	overgeneralization
Consequence from rule	Deductive	Skeptic	bad premise
Dependencies	Systems	Mechanistic	boundary blindness
Values	Normative	Scenario	hidden value judgment

Need	Start	Useful companion	Watch for
Steps	Algorithmic	Model-Based	optimizing the wrong goal
Categories	Conceptual	Reductionist	abstraction without use
Futures	Scenario	Evidential	forecast theater
Mechanism	Mechanistic	Systems	missing environment
Purpose	Teleological	Normative	assumed purpose
Explicit models	Model-Based	Skeptic	false precision
Strategy	Strategist	Scenario	action without capability

Selection rule

Select the mode that serves the next reasoning job, not the label of the subject. A security problem may need Systems, a product problem may need Evidential, and an architecture problem may need Normative. Change mode when the work changes from possibility to choice, from choice to proof, or from proof to operation.

Appendix C: Tool Reference

Research tool

Use for orientation, connections, and follow-up questions. Supply a decision, context, public domains, and desired artifact. Request distinctions between evidence, inference, analogy, and uncertainty. Do not use it as the sole authority for consequential claims.

Domain inspection

Use when a topic is too broad or unfamiliar. Ask for definitions, neighboring concepts, disagreements, common methods, and boundaries. Follow with external reading or expert review.

Path exploration

Use when you need to understand how two ideas relate. Ask what is direct, what is inferred, and what evidence would validate the relationship. A short conceptual path is a research lead, not a citation.

Mode reference

Use when choosing the posture for a session. State the decision and the desired output first; then ask which mode fits and why. Do not select a mode merely because its name sounds relevant.

Session record

Use a plain text or Markdown record for the question, time, participants, data boundary, output, evidence, dissent, action, and review date. Keep sensitive source material in its approved repository and link to it rather than duplicating it.

Failure fallback

If a tool is unavailable, switch to an ordinary decision template: list facts, assumptions, options, risks, next evidence, owner, and date. A useful practice survives tool downtime.

Appendix D: Research Notebook Templates

Question page

```
Date and participants:
Decision under study:
Decision owner:
Audience:
Deadline:
Constraints and exclusions:
Data classification:
What we currently believe:
What would change our mind:
```

Evidence page

```
Claim:
Source or observation:
Collected by and when:
Direct observation or interpretation:
Independent corroboration:
Known limitations:
Confidence and reason:
Decision consequence:
```

Connection page

```
Concept A:
Concept B:
Why the connection is interesting:
Analogy or proposed mechanism:
What it does not imply:
Testable question:
Cheapest safe test:
Result and next action:
```

Rejection page

```
Idea or path considered:
Why it was attractive:
Why it was rejected:
Evidence or constraint:
What would reopen it:
Date and owner:
```

Appendix E: Product Discovery Canvas

```
User segment: _____
Job to be done: _____
Recent painful episode: _____
Current workaround: _____
Cost of the problem: _____
Existing alternatives: _____
Reachable first users: _____
Narrow product wedge: _____
Riskiest assumption: _____
Manual one-week test: _____
Success threshold: _____
Refusal or harm risk: _____
Decision after test: _____
```

Discovery interview prompts

Ask what happened last time, what the person did next, who noticed, what it cost, and what they tried already. Ask what would make a proposed change unacceptable. Avoid asking whether they “like the idea”; preference without a recent behavior is weak evidence.

Review questions

Is this a painful job or a fashionable topic? Is the first user reachable? Can value be measured without building the whole product? What data or

authority would the product require? What is the safest way to learn?

Appendix F: Architecture Review Canvas

```
Decision and owner: _____
User promise: _____
Workload and latency: _____
State and consistency: _____
Dependencies: _____
Normal request path: _____
Degraded request path: _____
Failure signal: _____
Recovery and rollback: _____
Security boundary: _____
Operational burden: _____
Cost and resource budget: _____
Open assumption: _____
Review date: _____
```

Architecture walk-through

Read the path aloud from input to outcome. At every boundary ask what can be late, duplicated, missing, unauthorized, or malformed. Then ask how an operator detects it and what action is safe. A design is not reviewed until the team has walked at least one normal and one degraded path.

Appendix G: Security Review Canvas

```
Authorization and scope: _____
Asset and owner: _____
User or business impact: _____
Observed behavior: _____
Benign explanations: _____
Threat hypotheses: _____
Evidence needed: _____
Privacy and retention boundary: _____
Reversible containment: _____
Escalation path: _____
Stop condition: _____
Review and closure criteria: _____
```

Defensive review principles

Protect people and systems while investigating. Use only authorized environments and approved tools. Minimize copied data. Preserve timestamps and provenance. State uncertainty. Prefer containment that can be reversed while evidence is incomplete. Escalate when impact is high, scope is unclear, or the investigation could disrupt service.

Appendix H: Evidence and Confidence Rubric

Rating	Appropriate when	Required wording
High	Relevant evidence is direct, current, and independently checked.	State the evidence and remaining limitation.
Medium	Several signals agree but coverage, causality, or freshness is incomplete.	State the gap and the test that would raise or lower confidence.
Low	The idea rests on analogy, sparse observation, or unresolved alternatives.	Label it a working hypothesis, not a finding.
Unknown	The necessary observation is unavailable or the question is undefined.	Ask for the missing definition or evidence.

Calibration exercise

Take ten past predictions. Record the confidence at the time and whether the prediction was supported, partly supported, or rejected. Look for systematic overconfidence. If high-confidence claims are often wrong, lower the rating threshold or improve the evidence standard. A rubric is useful only when checked against outcomes.

Appendix I: Team Governance

Minimum policy

Define approved users, data classes, review roles, retention, logging, escalation, and prohibited uses. Require human approval for high-impact decisions. Require disclosure when an illustrative connection or

generated draft influenced a recommendation. Review access periodically and remove access that is no longer needed.

Decision rights

The researcher may propose. The domain reviewer may challenge. The system owner may approve operational change. The accountable leader accepts business risk. Security, privacy, legal, and safety reviewers retain their defined vetoes. Write these rights down so disagreement is a process rather than a personality contest.

Governance review questions

Who can see the input? Who can change the prompt or workflow? Who can act on the output? What is logged? How is a correction distributed? How can a user appeal? What happens when the tool is wrong, unavailable, or misused? Revisit these questions after every material scope change.

Appendix J: Publication and Legal Disclosure Boundary

Before publication, remove raw internal exports, private identifiers, confidential mappings, credentials, restricted source collections, unpublished customer details, and implementation material that would reveal protected design information. Describe public concepts at a user-facing level. Label fictional and composite examples. Do not imply customer results, scientific validation, legal compliance, or safety guarantees without evidence and permission.

Disclosure review checklist

- [ ] Every example is labeled illustrative, observed, or sourced.

- [ ] No private data or secret appears in prose, code, screenshots, or metadata.
- [ ] Public claims have a source or are clearly framed as interpretation.
- [ ] Product, security, legal, and privacy reviewers have approved relevant sections.
- [ ] Customer names and outcomes are used only with written permission.
- [ ] Technical descriptions stay at the user-facing abstraction level.
- [ ] The final PDF and web output were scanned, not only the Markdown source.

Author note template

```
This manuscript describes public, user-facing concepts and illustrative
workflows. Examples are not customer claims or guarantees. Readers remain
responsible for domain review, authorization, privacy, security, legal
review,
and decisions made from the material.
```

KOHNEX BRAIN / HUMAN EDITION

Chapter Workbook Expansions

This companion volume expands the practical layer of chapters 1–26. It is part of the human manuscript, not a technical export. All examples are illustrative. Replace fictional details with approved evidence before using an exercise in a real project.

1. The New Research Problem

Decision-surface drill

Write a topic, a decision, and a decision owner in three separate lines. Then write what happens if the team does nothing for one week, one month, and one quarter. This reveals whether the problem is urgent, important, or merely interesting. Add a sentence beginning “We will know more when…” and make the ending observable. If the sentence ends in “feel confident,” add a behavior or measurement that would justify that feeling.

Illustrative worked example

Topic: knowledge management. Decision: whether to introduce a shared incident index. Owner: the operations lead. Evidence: time required to find a prior incident, duplicate investigations, and operator adoption. A safe first test is a manually curated index for ten incidents, not an automated platform. The team can stop if the index takes more time to maintain than it saves.

Checklist

- [ ] Topic, decision, owner, and deadline are distinct.
- [ ] Inaction has a visible cost.
- [ ] The first test is smaller than the proposed solution.

2. What Kohnex Is

Boundary map

Draw three circles labeled agent, research layer, and accountable team. Put question formation in the agent circle, conceptual orientation in the research circle, and approval of consequential action in the team circle. Items that cross boundaries need an explicit handoff. A recommendation may cross from research to team; a private customer record should not cross into a general exploration session without authorization.

Worked comparison

For a simple date conversion, use a deterministic tool. For a question involving unfamiliar terminology, competing designs, and uncertainty, use a research session. For a safety-critical decision, use research only as preparation and require specialist review. Good judgment includes knowing when not to use a broad reasoning tool.

Checklist

- [ ] The tool is matched to the task.
- [ ] Human approval remains visible.
- [ ] The output is labeled as draft, evidence, or decision.

3. The Kohnex Mental Model

Concept-to-test table

Take five interesting connections from a session. For each, write the concept, the implied claim, the mechanism that might connect them, and the cheapest test. Delete any connection that cannot produce a question. This is an effective antidote to decorative metaphor.

Illustrative example

“A team resembles an ecosystem” is a concept. “Local specialists may respond faster than one central coordinator” is a claim. “Local decisions reduce coordination delay under bounded conditions” is a proposed mechanism. A tabletop simulation can compare response time and divergence. The test may reject the analogy while still improving the design conversation.

Checklist

- [ ] Every metaphor becomes a question.
- [ ] The mechanism is stated in ordinary language.
- [ ] The test includes a failure condition.

4. Connecting Your Agent

Integration rehearsal

Create a fictional request with a missing constraint, an ambiguous user, and one high-impact action. Ask the agent to identify the gaps before researching. Next, make the research tool unavailable and verify that the agent reports the limitation plainly. Finally, provide two conflicting sources and verify that the output preserves disagreement.

Operating note

A connection is ready for a pilot when its data boundary, timeout, fallback, audit record, and owner are documented. It is not ready merely because a demonstration is impressive. Keep a small synthetic test suite in version control and run it after prompt, model, or tool changes.

Checklist

- [ ] Missing information is requested, not invented.
- [ ] Tool unavailability is visible.
- [ ] Conflicting evidence survives synthesis.

5. Asking Better Questions

Rewrite ladder

Rewrite “What should we build?” as “Which user job is costly enough to solve this quarter?” Rewrite “What is the best architecture?” as “Which design meets the latency and recovery target with the least operational risk?” Rewrite “Is this secure?” as “Which authorized threat scenario has the greatest plausible impact, and what evidence distinguishes it from benign behavior?”

The improvement comes from adding a user, criterion, scope, and decision. A question can be broad during orientation, but it must become narrow before action.

Checklist

- [ ] The question names the user or system.
- [ ] Success and failure are observable.
- [ ] The requested artifact matches the decision.

6. Your First Research Session

Timed worksheet

At minute zero, write the decision. At minute five, circle terms you do not understand. At minute ten, select one connection. At minute fifteen, write the strongest objection. At minute twenty, assign an action. Keep the first session short enough that the team can repeat it weekly.

Debrief prompts

What did the session help us see? What did it tempt us to overclaim? Which source or person should we consult? What did we decide not to investigate? A good debrief rewards disciplined stopping as much as discovery.

Checklist

- [ ] The session had a timebox.
- [ ] The strongest objection was recorded.
- [ ] A person and date own the next action.

7. Understanding Thinking Modes

Mode rotation exercise

Take one proposal and run four short passes. Explore lists adjacent ideas. Mechanistic explains how the proposal would work. Skeptic searches for disconfirming cases. Normative asks whether it is acceptable under the team’s values. Compare what each pass made visible and what it ignored.

Facilitation rule

Do not call a mode a personality. A careful engineer can use Dream, and an imaginative designer can use Evidential. Mode selection is a

temporary work instruction. Rotate the facilitator so the same person does not always define what counts as evidence.

Checklist

- [ ] The mode is named in the session notes.
- [ ] A creative pass is balanced by a challenge pass.
- [ ] No mode is treated as an authority.

8. Researching Unfamiliar Territory

Field-entry notebook

Record ten terms, three central problems, two competing interpretations, one measurement practice, and one boundary of the field. For each term, write a plain-language definition and a source or expert who could correct it. Do not begin with an executive summary. Begin with the vocabulary that lets you recognize a meaningful disagreement.

Illustrative example

Entering reliability engineering, a product team may confuse availability, durability, recoverability, and resilience. The notebook forces these apart. It then asks which property matters to the proposed product and what operational evidence would demonstrate it.

Checklist

- [ ] Similar terms are distinguished.
- [ ] The map contains disagreements.
- [ ] One expert review is scheduled.

9. Discovering Product Opportunities

Interview-to-wedge workflow

Interview before ideating. Ask for the last real episode, the workaround, the person who paid the cost, and what would make a new approach unacceptable. Translate each interview into a job statement. Only then use cross-domain exploration to propose wedges.

Opportunity card

```
User and job:
Painful recent episode:
Current workaround:
Cost of failure:
Reachable first users:
Smallest manual service:
Evidence that would justify building:
Reason a user may refuse:
```

Checklist

- [ ] The idea is anchored in an observed job.
- [ ] A manual test precedes a platform build.
- [ ] Refusal and adoption costs are explicit.

10. Generating and Testing Hypotheses

Three-level test plan

Level one checks language and assumptions at a desk. Level two observes behavior with a small sample. Level three compares alternatives under controlled conditions. Move levels only when the previous one resolves its uncertainty. Do not use a large launch to answer a question a small interview could answer.

Worked illustrative example

Claim: a decision brief reduces meeting time. Desk check: define “brief” and “meeting time.” Observation: compare meetings that use a brief with similar meetings that do not. Alternative explanation: the brief works because a senior owner attended. Record attendance and agenda quality so the test can challenge that explanation.

Checklist

- [ ] The hypothesis predicts behavior.
- [ ] The alternative explanation is measurable.
- [ ] The test has a stopping rule.

11. Engineering and Architecture

Boundary review

For every component, name its input, output, state, owner, timeout, retry rule, and failure signal. Then ask whether a human can disable it safely. A diagram that omits state and failure behavior is a presentation, not an architecture review.

Trade-off worksheet

Compare three designs on user impact, operational burden, recovery, observability, security, cost, and reversibility. Write a paragraph for the option you do not recommend. This prevents a comparison table from becoming advocacy disguised as analysis.

Checklist

- [ ] Failure paths are first-class.
- [ ] State changes can be explained after the fact.
- [ ] The recommendation names what it sacrifices.

12. Cybersecurity Reasoning

Authorized defensive review

Write the scope, asset owner, permitted methods, time window, and stop conditions before looking at suspicious behavior. Separate indicator, hypothesis, impact, and response. Preserve evidence. Avoid copying secrets into a general-purpose prompt. A defensive session should make the system safer, not merely produce a vivid attack story.

Triage card

```
Observed behavior:
Affected asset:
Benign explanations:
Threat hypotheses:
Evidence needed:
Reversible containment:
Escalation owner:
Review time:
```

Checklist

- [ ] Authorization is explicit.
- [ ] Benign causes are considered.
- [ ] Containment is proportionate.

13. Resilience and Fault Tolerance

Failure rehearsal

Pick one dependency and remove it on paper. What does the user see? What state is retained? What work is retried? What work must not be retried? Who receives the signal? Define a graceful degradation path and

a recovery verification step before discussing sophisticated resilience patterns.

Illustrative example

If a pricing service fails, a storefront might show the last verified price, pause checkout, or route to manual review. Each option has a different risk. “Keep serving” is not automatically resilient if it creates incorrect charges.

Checklist

- [ ] User impact is named.
- [ ] Retry safety is explicit.
- [ ] Recovery is verified, not assumed.

14. Performance and Resource Decisions

Resource budget

List latency, memory, compute, network, money, operator attention, and environmental cost. Identify which budget is binding. Optimize the binding constraint first. A faster component may be a poor choice if it consumes scarce memory or increases incident burden.

Measurement discipline

Record workload shape, warm-up conditions, sample size, percentile choice, and variance. An illustrative benchmark is useful for a design conversation but must not be presented as a production guarantee. Compare the baseline and the proposed change under the same conditions.

Checklist

- [ ] The scarce resource is identified.
- [ ] Baseline and workload are documented.
- [ ] Cost includes operator attention.

15. Multi-Agent Consensus

Agreement protocol

Give each reviewer the same question, evidence, and output schema. Collect independent views before showing the group summary. Ask each reviewer for confidence, strongest objection, and missing evidence. Consensus is useful when it exposes agreement and disagreement; it is dangerous when it merely averages confident repetition.

Decision record

```
Shared conclusion:
Material disagreement:
Evidence everyone used:
Evidence interpreted differently:
Minority view and condition:
Human decision owner:
```

Checklist

- [ ] Views were independent before synthesis.
- [ ] Dissent is preserved.
- [ ] Agreement is not treated as proof.

16. Following Evidence and Paths

Traceability chain

For every material conclusion, record the question, observation, interpretation, source, transformation, and decision consequence. If a conclusion cannot be traced, label it as a working hypothesis. Use short links in the team record rather than relying on memory or a fluent paragraph.

Path audit

Ask where the chain branches, which assumption connects two steps, and whether an alternative path reaches a different conclusion. A path is strongest when another person can reproduce it without needing the original conversation.

Checklist

- [ ] Claims have inspectable support.
- [ ] Interpretation is separate from observation.
- [ ] A reviewer can reproduce the path.

17. Composing Kohnex Tools

Pipeline design

Compose tools around a bounded artifact: orientation notes, a comparison, a hypothesis card, or a decision memo. Pass only the context needed for the next step. Preserve the original question and constraints at every handoff. Add a human checkpoint before external action.

```
Orient → select → challenge → verify → synthesize → approve
```

Failure handling

Each step should state what happens when it returns no result, conflicting results, or a timeout. A pipeline that silently converts absence into a plausible paragraph is unsafe. Test empty and contradictory inputs deliberately.

Checklist

- [ ] Each step has one job.
- [ ] Context loss is detectable.
- [ ] External action requires approval.

18. From Insight to Experiment

Experiment canvas

Write the insight in one sentence, the assumption it depends on, the smallest intervention, the measurement, the threshold, and the rollback. Decide what result would cause you to stop, continue, or redesign. This converts a compelling connection into a learning instrument.

Illustrative example

Insight: teams lose time reconstructing ownership. Intervention: add a structured owner field to the incident template. Measure reconstruction time and unanswered handoffs for four weeks. Roll back if the field is routinely ignored or creates duplicate records. The experiment tests a specific mechanism, not a broad promise of “better collaboration.”

Checklist

- [ ] The intervention is small.
- [ ] Success and failure thresholds are written first.
- [ ] Rollback is practical.

19. Working With Uncertainty

Confidence statement

Use calibrated language: high confidence for directly verified, relevant evidence; medium for several consistent but incomplete signals; low for analogy, sparse data, or unresolved alternatives. Always include the reason and what would change the rating. Confidence is not a mood or a property of prose.

Uncertainty budget

List unknowns by consequence, not by quantity. Resolve high-consequence unknowns first. If the decision is reversible, a lower confidence may be acceptable. If it is irreversible, require stronger evidence or an explicit risk owner.

Checklist

- [ ] Confidence has a reason.
- [ ] Unknowns are ranked by consequence.
- [ ] Reversibility affects the evidence standard.

20. Team Workflows

Meeting pattern

Before the meeting, circulate the decision question and evidence boundary. During the meeting, spend separate time on orientation, disagreement, and decision. Afterward, record owner, action, review date, and dissent. Do not use the meeting to discover basic vocabulary if a short pre-read can do that work.

Roles

The question owner frames the job. The researcher develops the map. The challenger searches for failure and disconfirmation. The domain reviewer checks consequential claims. The decision owner accepts the trade-off. One person may hold multiple roles in a small team, but the functions should remain visible.

Checklist

- [ ] Roles are explicit.
- [ ] Dissent has a safe channel.
- [ ] Decisions are revisited on a date.

21. Common Failure Modes

Recovery clinic

When a session fails, classify the failure: vague question, wrong mode, unsupported analogy, missing evidence, excessive scope, tool failure, or unclear ownership. Choose a repair that matches the cause. A vague question needs reframing; a missing source needs research; unclear ownership needs governance. More prompting is not the universal fix.

Warning signs

Watch for polished summaries with no sources, many connections with no tests, consensus with no independent views, and architecture diagrams with no failure path. Invite a reviewer to point to the weakest sentence rather than asking whether the whole output “looks good.”

Checklist

- [ ] The failure has a category.
- [ ] The repair addresses the category.
- [ ] The team records the lesson.

22. Building a Kohnex Practice

Maturity path

Start with individual notes, then a shared decision template, then recurring review, then measured experiments. Do not begin by mandating a large platform. A practice is mature when teams can explain why they used a mode, how they checked a claim, and what happened after the decision.

Weekly ritual

Review one successful session, one misleading connection, one rejected hypothesis, and one decision whose review date arrived. Share the smallest improvement to the workflow. This creates learning without turning every meeting into process administration.

Checklist

- [ ] Adoption starts with a real decision.
- [ ] Templates remain lightweight.
- [ ] Outcomes are reviewed, not just prompts.

23. Case Studies

Case reading method

For each case, identify the initial ambiguity, the selected mode, the evidence boundary, the reversible action, and the result that would disconfirm the plan. Then rewrite the case from the perspective of a skeptical operator. This turns stories into transferable judgment rather than inspirational anecdotes.

Composite-case caution

Composite cases teach patterns but do not establish customer outcomes. Label fictional details and never imply that an illustrative workflow is a verified benchmark. Replace story-based confidence with a local pilot and a review record.

Checklist

- [ ] Case facts and teaching devices are separated.
- [ ] A local test is proposed.
- [ ] No customer claim is inferred.

24. Prompt Library

Prompt adaptation

Before using a prompt, add the audience, decision, constraints, data boundary, mode, and artifact. After using it, inspect whether the response obeyed those contracts. Save the smallest prompt that reliably produces the desired structure. Do not paste confidential material simply because a template has a blank field.

Safe prompt pattern

```
Help me reason about [decision] for [audience]. Use [mode] to explore
[public domains]. Separate known evidence, inference, analogy, uncertainty,
and open questions. Do not invent sources or private facts. End with three
options, a reversible test, an owner, and a review date.
```

Checklist

- [ ] Data boundaries are in the prompt.
- [ ] The output asks for uncertainty.
- [ ] A human reviews consequential results.

25. Exercises

Capstone exercise

Choose a real but reversible decision. Write the decision surface, run one orientation session, create a hypothesis card, ask a challenger to review it, and run the smallest safe test. At the end, write what changed in your belief and what you would do differently. Grade the process on clarity, traceability, reversibility, and learning, not on whether the first idea was correct.

Peer review rubric

Ask a peer to score whether the question is clear, the evidence is inspectable, alternatives are present, the next action is bounded, and the owner is named. Require comments for every score below four. The point is to improve the work before consequences grow.

Checklist

- [ ] The exercise has a real owner.

- [ ] The test is safe and reversible.
- [ ] Learning is recorded regardless of outcome.

26. Glossary and Quick Reference

Personal glossary method

For each important term, write a plain-language definition, a non-example, a related term, and the decision where it matters. Revisit the glossary after every unfamiliar-field session. A glossary is not decoration; it is a shared contract that prevents teams from agreeing on different meanings.

Final self-check

Can you state the decision? Can you name the owner? Can you distinguish evidence from analogy? Can you select a mode for the next reasoning job? Can you name the smallest safe test? Can another person follow the path? If any answer is no, the work is not finished, even if the prose is polished.

Checklist

- [ ] Terms have examples and non-examples.
- [ ] The quick reference points to action.
- [ ] The final decision has a review date.

KOHNEX BRAIN / HUMAN EDITION

Practice Atlas

This atlas supplies reusable patterns for the decisions that recur across research, product, engineering, security, and operations. It is deliberately concrete. Each pattern asks a person to define the decision, inspect evidence, and choose a reversible next action. None of the patterns is a substitute for professional advice or authorization.

Pattern 1: The one-page brief

Use when a team is circling a decision in conversation. Put the decision at the top, followed by owner, deadline, affected people, options, criteria, evidence, uncertainty, recommendation, dissent, next action, and review date. Limit the first draft to one page. If the team cannot fit the decision on one page, it may be mixing several decisions. Split them. A brief is not a complete research report. It is a compact interface between research and accountability.

Pattern 2: The assumption inventory

List assumptions under four headings: user, environment, mechanism, and governance. User assumptions concern need, behavior, consent, and access. Environment assumptions concern load, failure, change, and dependency. Mechanism assumptions concern why the proposal should work. Governance assumptions concern authority, review, and acceptable harm. Rank assumptions by consequence and uncertainty. Test the high-consequence, high-uncertainty items first.

Pattern 3: The evidence ladder

Move from definition to observation, from observation to comparison, and from comparison to controlled intervention. A definition tells you what a term means. An observation tells you what happened. A comparison shows whether two conditions differ. An intervention tests whether changing something changes an outcome. The ladder is not always linear, but skipping lower steps makes higher claims fragile. Mark which rung supports each sentence in a decision memo.

Pattern 4: The alternative explanation table

Observation	Leading explanation	Benign explanation	Evidence that separates them	Safe next action
A process is slow	dependency saturation	unusual workload	compare load and dependency timing	sample before scaling
Users abandon a flow	confusing design	wrong audience	observe sessions and segment data	test one copy change
Alerts increase	new threat activity	detector change	compare rule and baseline versions	review a sample
Costs rise	inefficient computation	usage growth	normalize by workload	measure before optimizing

The table prevents the first plausible story from becoming the official story.

Pattern 5: The smallest safe test

Describe the smallest change that can generate useful evidence without exposing people or production to disproportionate risk. Use synthetic data, a shadow mode, a limited audience, a manual process, or a tabletop exercise. State what the test cannot prove. A small test is not valuable merely because it is small; it must discriminate between meaningful options.

Pattern 6: The stoplight decision

Green means evidence and safeguards are sufficient for the next bounded step. Amber means the step is reversible but a material uncertainty remains; name its owner and review date. Red means the proposed action could create unacceptable harm, expose protected information, or exceed authorization. Stop, escalate, or redesign. Do not let a green color hide an unresolved legal, privacy, or safety question.

Pattern 7: The decision journal

Record the decision before results are known. Include what you expected, why, confidence, alternatives, and what would change your mind. Revisit after the review date. Judge the process separately from the outcome. A good decision can have a bad result because of uncertainty; a bad process can have a lucky result. Journaling makes that distinction visible.

Pattern 8: The disagreement protocol

Ask each reviewer to state the conclusion they support, the evidence they rely on, their confidence, and the observation that would change their view. Then identify whether the disagreement is about facts, definitions, predictions, values, or risk tolerance. Different disagreements require

different repairs. More data may resolve a factual dispute, while a values dispute needs an explicit owner and principle.

Pattern 9: The source interview

When a claim matters, interview the person or read the source closest to the observation. Ask what was measured, under which conditions, what was excluded, and what failed. Preserve the source's limitations. A summary that removes caveats may be easier to read and less truthful. If direct access is impossible, lower confidence and say why.

Pattern 10: The glossary contract

Choose terms that have caused confusion. For each, provide a definition, non-example, related term, and decision where the distinction matters. Ask two team members to explain the term independently. If their explanations differ, resolve the definition before debating the proposal. Shared language is a control against silent disagreement.

Pattern 11: The domain bridge

When connecting two fields, name the shared structure and the non-shared context. For example, both distributed systems and ecosystems involve local interactions, but their materials, timescales, incentives, and failure modes differ. Use the bridge to generate questions, not to transfer conclusions. A good bridge includes a warning label: "This analogy may help us inspect X; it does not establish Y."

Pattern 12: The architecture walk

Walk one request from arrival to user-visible outcome. At every boundary ask what is validated, stored, delayed, duplicated, denied, retried, and

observed. Walk a second request with one dependency unavailable. Walk a third with malformed input. If the team cannot explain the user-visible result and operator action, the architecture is incomplete.

Pattern 13: The failure budget

Define what amount of failure the service can tolerate over a review period and what users experience during that failure. Include availability, correctness, recovery time, data loss, and operator effort. A system that remains available while returning incorrect information may violate the most important budget. Discuss budgets with product and operations together.

Pattern 14: The resource ledger

Track compute, memory, network, storage, money, latency, human attention, and environmental impact. Record the baseline and the proposed change. Note whether the optimization shifts cost rather than removing it. For example, reducing latency through aggressive caching may increase staleness, storage, invalidation work, and privacy exposure. Optimization is a trade, not a free improvement.

Pattern 15: The safe security hypothesis

Write “If this were a security issue, we would expect…” followed by observable indicators. Write “If this were benign, we would expect…” followed by competing indicators. Define the authorized collection step and its stop condition. Protect evidence and minimize access. Never turn a hypothesis into an accusation without corroboration and appropriate process.

Pattern 16: The recovery card

```
Failure:
User-visible effect:
Last known good state:
Automatic response:
Human escalation:
Data that must not be lost:
Rollback:
Recovery verification:
Post-incident learning:
```

Fill the card before the incident. During stress, a pre-agreed response is safer than improvisation. Review it after drills and real events. Retire steps that nobody can perform under the stated conditions.

Pattern 17: The experiment ledger

For every test, record owner, hypothesis, intervention, population or environment, baseline, measure, threshold, duration, exceptions, result, and decision. Keep negative results. A result without conditions cannot be reused; a result without a decision becomes trivia. Where people are affected, document consent, privacy, and escalation requirements.

Pattern 18: The prompt contract

Use this compact contract for user-facing research:

```
Objective: [decision or question]
Context: [people, system, timeline, constraints]
Allowed material: [public concepts, approved sources, synthetic examples]
Lens: [mode and reason]
Output: [memo, table, test plan, glossary, or questions]
Safety: separate evidence, inference, analogy, and uncertainty; do not
invent.
```

The contract is useful because it makes omissions visible. Add a request for a dissenting view whenever a recommendation is expected.

Pattern 19: The review cadence

Use daily review during an incident, weekly review during an experiment, monthly review for a practice, and quarterly review for governance. At each cadence ask what changed, which assumptions were retired, which actions were safe, and which controls need adjustment. Cadence should follow consequence and rate of change, not organizational habit.

Pattern 20: The handoff packet

A handoff includes the decision, current state, evidence, uncertainty, next action, owner, deadline, and escalation condition. Add what not to assume. The receiver should be able to act without reconstructing an entire conversation. Handoffs are a reliability boundary; test them with someone who was not in the original meeting.

Pattern 21: The publication packet

Before sharing a case or guide, assemble sources, permissions, disclosure labels, redactions, reviewer names, and a change log. Inspect screenshots, metadata, examples, and appendices. Public concepts can

still become sensitive when combined with context. Have the appropriate product, legal, privacy, and security reviewers inspect the final rendered artifact.

Pattern 22: The correction path

Publish how readers can report an error or unsafe example. Assign an owner, response target, severity categories, and correction method. When a material correction is made, update derivative formats and notify affected users where appropriate. Trust is strengthened by a visible correction process, not by pretending drafts are infallible.

Pattern 23: The team retrospective

Ask four questions: What did we expect? What happened? Which assumption failed? What change will we make? Avoid turning the meeting into a list of personal faults. Examine incentives, information flow, interfaces, and decisions. End with one owned improvement and one practice to stop. A retrospective without changed behavior is a story about learning, not learning.

Pattern 24: The teaching session

Have a learner explain a concept without notes, produce a non-example, and apply the concept to a new case. The facilitator corrects definitions before debating conclusions. Ask the learner to state uncertainty. This is a stronger test of shared understanding than asking whether the material was clear.

Pattern 25: The readiness review

Before moving from research to action, check objective, evidence, safety, ownership, reversibility, monitoring, rollback, communication, and review. Use a written “not ready because” field. A refusal to proceed is a valid outcome when the evidence standard has not been met.

Pattern 26: The closing reflection

End a project by asking what the team now knows, what it only believes, what it should stop doing, which connection was useful, which was misleading, and what should be added to the shared glossary. Close the decision record with the result and next review. Closure turns a session into institutional memory.

Atlas worksheet

```
Pattern used:
Decision:
Why this pattern fits:
Evidence available:
Uncertainty remaining:
Safe next action:
Owner and date:
What would make us stop:
What we learned:
```

KOHNEX BRAIN / HUMAN EDITION

Worked Examples and Workshop Cards

The examples in this file are fictional composites. They demonstrate how to move from a broad prompt to a bounded decision. They are not customer stories, benchmarks, legal advice, medical advice, or security authorization.

Example 1: Choosing a research direction

A small team wants to study trustworthy automation. The first prompt asks for “everything important.” The facilitator rejects it because there is no decision, audience, or timebox. The revised question is whether to spend the next month on auditability, human override, or evaluation. Explore creates vocabulary. Conceptual separates accountability from explainability. Evidential asks which claims are supported by existing tests. The output is a reading list and three interviews, not a product roadmap.

The team writes a hypothesis: “Operators will adopt automation faster when they can inspect why a recommendation appeared.” The alternative is that adoption depends more on the ability to undo an action. A tabletop comparison tests both explanations. Participants receive identical fictional incidents and alternate between inspectable rationale and easy reversal. The team measures time to approve, questions asked, and willingness to use the feature again. The result determines which assumption deserves a larger study.

Example 2: A product wedge

A founder notices that security teams and finance teams both struggle with evidence handoffs. The connection is interesting but too broad. Interviews reveal a narrower job: an incident commander needs to hand a timeline to a finance approver who was not present. The first wedge is a structured handoff packet, assembled manually from approved records. The success measure is reconstruction time and number of clarification messages.

The founder asks for three implementation ideas, then uses Reductionist to remove anything not required for the handoff. Out-of-Box suggests a visual timeline, a confidence marker, and a review queue. Skeptic asks whether the problem is actually missing ownership. A controlled pilot compares the packet with the existing channel transcript. The team delays automation until it knows whether structure or ownership produces the improvement.

Example 3: A queue under failure

An engineering group must choose between direct calls and a queue for a partner integration. Systems mode identifies coupling, ordering, duplicate delivery, and recovery. Mechanistic mode describes the state transitions. Scenario compares normal traffic, a slow partner, a partner outage, and a replay after recovery. Normative asks whether stale data may be shown to users.

The recommendation is not “always queue.” It is a hybrid: acknowledge a request only when the local state is durable, display a clear pending status, and make the external operation idempotent. The experiment uses a simulated partner and synthetic requests. It measures duplicate effects, user-visible delay, and operator effort. A stop condition is any path that can charge twice or silently lose a request.

Example 4: An unusual login

A security team sees a login from an unfamiliar location. Predator mode produces possible attack paths, but the team does not jump to a conclusion. Diagnostician lists travel, a corporate proxy, a service account, logging error, and stolen credentials. Evidential specifies checks: authentication context, device history, approved travel, recent configuration, and related activity. The team applies reversible step-up verification while preserving records.

The review notes that location alone is a weak indicator. The stronger signal is the combination of new device, unusual time, and an access pattern not seen for the account. The result is a detector improvement and an investigation record, not a claim that the user was compromised. Authorization, privacy, retention, and escalation remain explicit.

Example 5: Researching a new scientific field

A software team enters a field with unfamiliar terminology. Explore produces a vocabulary map. The team asks for formal definitions, common measurement methods, competing interpretations, and limitations. A domain reviewer corrects two conflated terms. The team reads an introduction, a survey, a criticism, and an applied report. Mechanistic mode explains a process, while Skeptic identifies where the explanation may not transfer to the team’s setting.

The first deliverable is a glossary and a one-week replication exercise using public material. The exercise is intentionally modest. It tests whether the team can apply the field’s concepts correctly before it makes a product claim. The team records what it cannot establish and the expert questions that remain.

Example 6: A performance trade-off

A service is slow during peak traffic. The initial idea is to cache everything. A resource ledger shows that memory is scarce and freshness matters to users. The team measures latency percentiles, memory pressure, invalidation work, stale-result rate, and operator incidents. Focus narrows the question to the most expensive request class. Model-Based compares caching, precomputation, and workload shaping.

The smallest test caches one low-risk response class for a limited audience. The threshold is a latency improvement without a material increase in stale responses or memory eviction. If the threshold is missed, the team does not declare caching a failure in general; it records which assumption failed and tests the next option.

Example 7: A resilient workflow

A support workflow depends on a classification service. Regeneration asks how the workflow can recover. Systems distinguishes the user request, the classification state, the human queue, and the notification state. The design allows unclassified tickets into a visible review queue rather than rejecting them or silently guessing. A manual fallback protects continuity. Recovery verification checks that queued work is processed once and that users can see its status.

The team runs a game day with the classification service unavailable. Operators practice the fallback, record confusion, and improve the runbook. Success is not zero disruption. Success is a bounded, understandable disruption with no lost requests and a clear return to normal operation.

Example 8: A disagreement in architecture

One reviewer wants a centralized policy service. Another wants local policy copies. The disagreement is first classified: they agree on the goal but differ about freshness, availability, and administrative control. Each writes a prediction. Centralized policy should reduce drift but create a dependency. Local copies should preserve availability but create convergence delay.

The team measures policy update time, behavior during dependency loss, operator effort, and disagreement between copies. It chooses a design for the current risk, documents the trade-off, and schedules a review when workload or policy complexity changes. Consensus comes after the disagreement is understood, not before.

Example 9: A governance decision

A team wants to let an agent draft external communications. The research session identifies audience, data classes, legal review, brand risk, correction process, and approval rights. The initial scope is internal drafts using synthetic examples. The agent must label uncertainty, cite approved material, and stop when a source is missing. A human approves every external message.

After a month, the team reviews correction rate, review time, and inappropriate claims. It does not measure success by the number of drafts produced. The governance review asks whether access, retention, and approval still match the scope. Expansion is possible only when the evidence and controls justify it.

Workshop cards

Card 1: Missing owner

Question: Who can accept the trade-off? Test: ask three participants to name the owner independently. Repair: appoint an accountable decision owner before further research.

Card 2: Missing deadline

Question: When does delay create cost? Test: compare this week, next month, and next quarter. Repair: state a decision window or explicitly choose to wait.

Card 3: Attractive analogy

Question: What mechanism is claimed? Test: write a prediction that could fail. Repair: label the analogy and design a small test.

Card 4: Conflicting sources

Question: Are the sources measuring the same thing? Test: compare definitions, populations, dates, and conditions. Repair: preserve disagreement until the difference is explained.

Card 5: Overbroad prompt

Question: What artifact is needed? Test: ask for a one-page memo. Repair: split topic, decision, and experiment into separate requests.

Card 6: False precision

Question: What does the number mean? Test: inspect method, sample, range, and limitations. Repair: use an interval or qualitative confidence when exactness is unjustified.

Card 7: Missing baseline

Question: Better than what? Test: record current behavior before changing it. Repair: establish a baseline under stated conditions.

Card 8: Unclear failure

Question: What does the user experience? Test: walk one unavailable dependency. Repair: document degraded behavior and recovery.

Card 9: Hidden sensitive data

Question: Does the session require this detail? Test: replace it with a synthetic value. Repair: minimize, redact, or move the data to an approved workflow.

Card 10: Tool dependence

Question: Can the team continue without the tool? Test: use the ordinary decision template. Repair: document a manual fallback.

Card 11: Consensus pressure

Question: What does the minority believe? Test: collect independent views first. Repair: preserve dissent and its condition.

Card 12: Scope drift

Question: Is this still the original decision? Test: compare each new request with the decision sentence. Repair: open a new decision record when the owner or consequence changes.

Card 13: Unmeasured benefit

Question: What behavior should improve? Test: define an observable outcome. Repair: replace “better” with a measure and threshold.

Card 14: Unpriced burden

Question: Who operates this? Test: simulate a bad day. Repair: include attention, training, maintenance, and escalation in the trade-off.

Card 15: Unreviewed high impact

Question: Who must approve? Test: inspect the decision-rights map. Repair: pause action and involve the required specialist.

Card 16: Repeated prompt failure

Question: Is the issue wording or missing context? Test: compare a short prompt with a contract containing objective, evidence, and output. Repair: change the workflow, not only the adjectives.

Card 17: Unsupported novelty

Question: What is actually new? Test: compare public prior work and alternatives. Repair: call it a hypothesis until novelty and value are checked.

Card 18: Unclear review date

Question: When will the decision be revisited? Test: inspect the calendar. Repair: assign a date tied to evidence arrival or a change in conditions.

Card 19: Incomplete handoff

Question: What must the receiver know to act? Test: have an uninvolved person use the packet. Repair: add state, owner, uncertainty, and escalation.

Card 20: Lucky outcome

Question: Was the process sound? Test: compare the original prediction with the result and alternatives. Repair: improve the decision record even when the result was favorable.

Workshop close

Choose one card that describes the current project. Write the repair, owner, evidence, and review date. Then choose a different card that could appear next. Preventive practice is more valuable than an impressive postmortem.