

Most project tools store information. Fewer help a team understand how that information fits together. A project knowledge graph is meant to do the latter: connect people, organizations, decisions, documents, claims, and open questions so that a team can reason across its work instead of starting from scratch in every conversation.

In Suprmind, the project knowledge graph is best understood as a shared, evolving map of project context. It can help a team keep track of entities and relationships—for example, which customer raised a concern, which product decision addresses it, what evidence supports that decision, and what still needs validation. The graph is not simply a folder of files or a transcript archive. Its value comes from making relevant connections available as work continues.

The practical question is research.com not only what the feature does, but who gets it. In Suprmind's plan structure, the knowledge graph is a capability for **Pro and higher** plans. Spark is the entry point; Pro, Frontier, Power, and Enterprise are the higher tiers to evaluate when shared project context, more involved workflows, and structured decision support matter. Exact limits and feature access can change, so teams should confirm current plan details before buying.

What a Project Knowledge Graph Represents

A knowledge graph organizes information as **entities and relationships**. An entity might be a customer, competitor, product feature, regulation, vendor, or decision. A relationship describes how two entities connect: a feature depends on a platform, a policy applies to a region, or a customer objection influenced a roadmap choice.

Consider a product team evaluating a new analytics feature. The project may contain interview notes, a product brief, market research, and a debate about engineering effort. A conventional chat can summarize each item. A project knowledge graph aims to connect the items: the interview evidence relates to a customer segment; the segment relates to a revenue opportunity; the proposed feature relates to a technical dependency; and the decision relates to the evidence and trade-offs that informed it.

Those connections make it easier to ask questions that span sources: "Which customer needs support this proposal?" "What assumptions have not been validated?" or "What decisions depend on the pricing model?" The graph does not guarantee that an answer is correct. It gives the team a more structured context in which to inspect the answer and check its supporting material.

Why Multi-Model Orchestration Matters

Research and decision work rarely fits one prompt or one model's strongest use case. A team may want to summarize a long set of documents, compare conflicting claims, draft an options memo, and then test the memo for missing evidence. Multi-model orchestration in a shared thread lets users coordinate work across models, rather than treating each model interaction as a disconnected session.

Tools such as OpenAI and Anthropic provide models that can be used for different kinds of reasoning and generation. In a multi-model workflow, a user might ask one model to organize source material, another to challenge an emerging recommendation, and then bring the results back into the same project conversation. The important distinction is that the shared thread preserves the project's working context. Users can compare outputs and connect them to the underlying entities and relationships, instead of copying fragments between isolated chats.

Orchestration should not be confused with automatic truth-checking. Models can disagree, overlook a source, or produce a confident but unsupported statement. A good workflow makes those differences visible and gives people a way to verify claims against the project record. For regulated or high-stakes decisions, human review remains essential.

From Chat to Decision Intelligence

Decision intelligence is the process of turning information into a decision that can be explained, evaluated, and revisited. A project knowledge graph can support that process by preserving how evidence, assumptions, options, and decisions relate to one another.

For example, a strategy team could use a project to compare three market-entry options. The team might collect source material, identify the relevant competitors and customer segments, record assumptions about costs, and define criteria such as time to launch or regulatory exposure. A structured workflow can then help the team see which option is supported by which evidence—and where the case depends on an untested assumption.

Structured validation is especially important. Instead of asking only for a polished recommendation, teams can request a check of the decision itself:

- Are the key claims linked to evidence?
- Are there contradictory findings in the project sources?
- Which assumptions have not been tested?
- What is missing from the comparison criteria?
- Does the recommendation follow from the stated priorities?

This approach can make an investment memo, roadmap debate, vendor review, or policy analysis easier to audit. It does not remove judgment; it helps make the judgment and its basis more legible.

Who Gets the Project Knowledge Graph?

The short answer is: teams considering the knowledge graph should look at **Pro and higher**. Spark is a lower-cost starting point for individuals or lighter use, while the higher plans are the place to assess when a team needs deeper project context and decision workflows. The exact feature boundaries should be verified against Suprmind's current plan page or with its sales team.

Plan How to think about it Knowledge graph fit Spark Entry-level access for lighter or individual workflows Do not assume the full project knowledge graph is included; confirm the current feature set. Pro For users who need more capable, ongoing project work The starting tier to evaluate for the project knowledge graph. Frontier For more demanding workflows and model use Evaluate if your projects require broader orchestration or greater capacity. Power For intensive usage and more advanced work patterns Consider when project workflows are frequent, complex, or central to team operations. Enterprise For organizational deployment and procurement requirements Assess for team-wide needs, administration, and enterprise-specific controls.

That table is a decision guide, not a substitute for a current feature matrix. Plan names and inclusions can evolve. Before choosing a tier, check whether the graph is enabled, how many projects or collaborators are covered, what usage limits apply, and which administration or security controls are included.

Plan-Based Pricing Versus Per-Token Billing

One reason teams compare AI products closely is that pricing can be difficult to forecast. With per-token billing, cost depends on how much text is processed and generated, which models are used, and how often people run their workflows. That can be appropriate for flexible, developer-led use, but it may make team budgeting less predictable.

Plan-based pricing packages access into a subscription tier. A simple example is **Spark at \$19/month**. A plan can make the monthly cost easier to understand, though teams still need to review included usage, model access, collaboration limits, and any overage or add-on charges. “Plan-based” does not automatically mean unlimited.

For procurement, compare the expected cost of a real workflow rather than just the headline price. Estimate how many people will use it, how many projects they will maintain, how often they will run multi-model reviews, and whether the plan includes the features the workflow depends on. Also ask whether usage is shared across a workspace or allocated per user.



How to Evaluate It Against Other Business Software

Pricing models vary widely across business software, so comparisons are useful mainly as procurement context—not as a direct feature or value comparison. For example, Gaviti, an AR software provider, uses pricing by quote and offers both a free trial and a demo. Paylocity uses custom pricing and does not offer a free trial. Zoho Books lists a plan at \$15 per month and offers a 14-day free trial.

Those examples illustrate why a buyer should identify the evaluation path as well as the price. A quote-based vendor may require a sales conversation; a product with a free trial allows hands-on testing; a fixed-price subscription can be easier to assess from a published plan. For Suprmind, the relevant question is whether a team can test its own project workflow at the appropriate tier and understand what the subscription includes.

Who Is Most Likely to Benefit?

The project knowledge graph is likely to be most useful when a team has a body of information that grows over time and must inform repeated decisions. That includes strategy groups, research teams, product operations, compliance teams, and organizations that need to explain why a decision was made.

- **Strategy teams** can connect market evidence, assumptions, and recommendations.

- **Research teams** can organize findings across sources and track unresolved questions.
- **Product teams** can relate customer needs to roadmap proposals and dependencies.
- **Compliance teams** can map policy language, obligations, and review decisions—subject to appropriate human and legal validation.
- **Cross-functional groups** can preserve context as work moves between departments or contributors.

It may be less valuable for a person who only needs occasional, one-off answers and does not need persistent project context. In that case, a lightweight plan or standalone chat tool could be sufficient. The graph earns its place when connections between pieces of information matter as much as the pieces themselves.

Questions to Ask Before Choosing a Plan

Run a small evaluation using a real project rather than a generic demo prompt. Bring together a few representative documents, ask a question that requires connecting them, and test whether the resulting answer makes its evidence and uncertainty clear. Then have a colleague challenge the recommendation or introduce a conflicting source.

1. Is the project knowledge graph available on the plan being evaluated?
2. Can the team inspect or correct entities and relationships?
3. How does a shared thread preserve context across models such as OpenAI and Anthropic?
4. Can users distinguish sourced facts from model-generated interpretation?
5. What usage, collaborator, and project limits apply?
6. Are the necessary privacy, access, and retention controls available?

The Bottom Line

Suprmind's project knowledge graph is a way to organize project context around entities and relationships, making it easier to connect evidence, assumptions, and decisions. Combined with multi-model orchestration in a shared thread, it can support decision intelligence and structured validation—but it should be evaluated as a workflow tool, not as an automatic source of truth.

For buyers, the clearest plan distinction is that the knowledge graph is aimed at **Pro and higher**, while Spark—illustrated at \$19/month—is the entry-level option to examine for lighter needs. Compare the current feature gates across Pro, Frontier, Power, and Enterprise, and test the plan against a real decision workflow. The right tier is the one that gives your team enough shared context and validation capability without paying for capacity or controls it does not need.

