

From Documents to Networks: Graph-Based RAG for Tax Investigation

IOTA Forum on Tax and Customs Data Science

20 October 2026

Jacob Coles · Redfield





Welcome

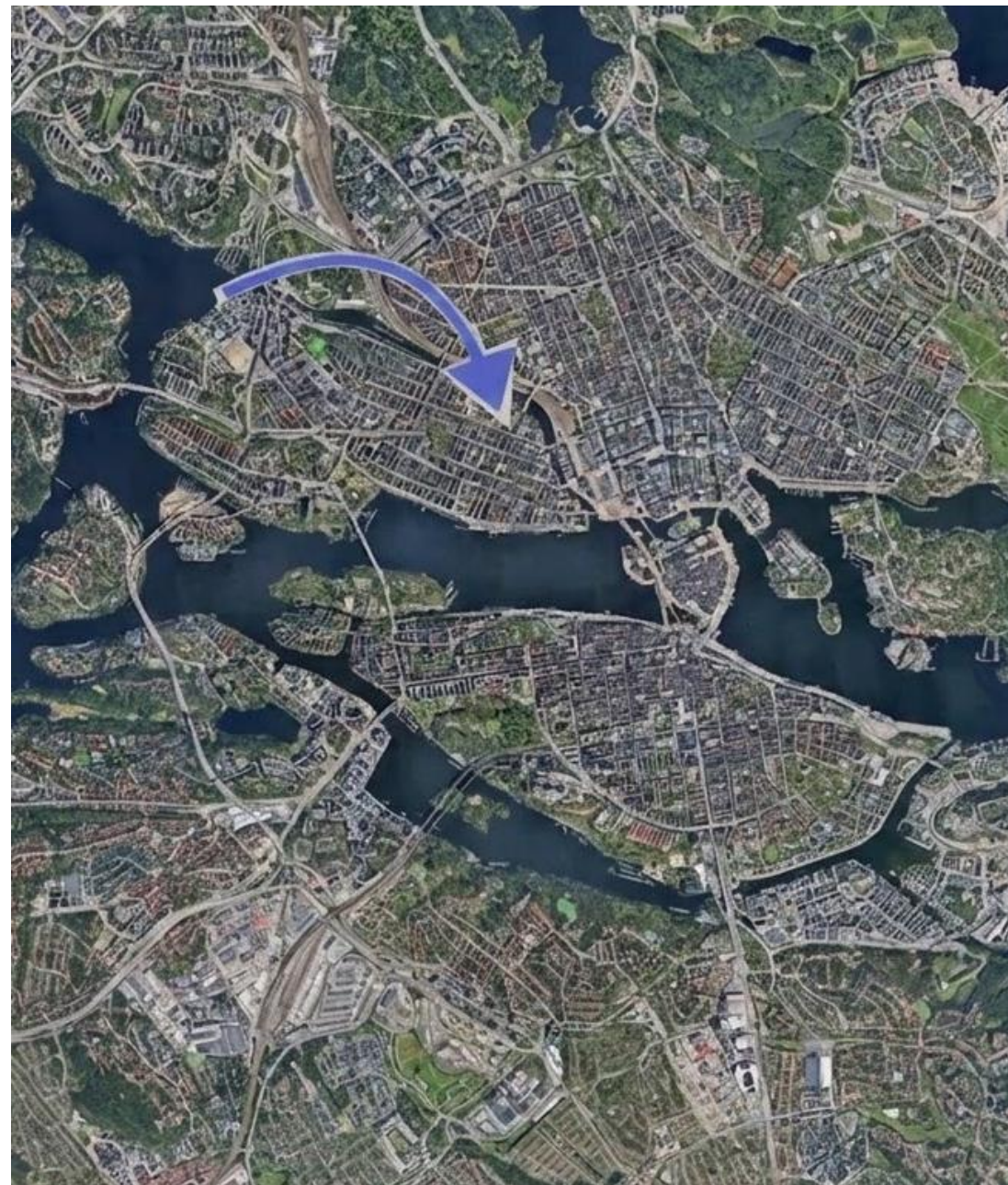
- My name is **Jacob Coles**
- Data scientist at **Redfield**, based in **Stockholm, Sweden**
- Have a presence in Brussels working with the public sector
- We recently won an award at an EU institution's Hackathon
- **Key focuses of ours:**
 - Local AI & ML models with frameworks (including LLMs)
 - Risk detection, AI sovereignty & compliance
 - AI knowledge management (RAG pipelines)
 - Graph analytics and implementations
- **Today:** How we implemented private AI for advanced risk detection and analysis using LLMs & graph-networks



Redfield.AI



Jacob Coles



To Start: A simulated case

A simulated case:

- **MCP Risker SA** pays its IT supplier **JF Apex**
- **Benjamin Dark** approved the payments
- **Jason Dark** is JF Apex's Managing Director

"What supports the concerns about JF Apex when we connect its payments, approvals and management roles, and how much was paid?"

Why this question:

Finding vs checking

The audit memo already raises the concern. We can check supporting records rather than repeat its conclusion.

Follow the connections

Relevance of links between JF Apex, its payment approver and its Managing Director.

Check the total

Check the payments against the supplier ledger as evidence.

Today's Goal: GraphRAG for Tax Investigators

- Understand why searching documents directly isn't enough
- What a **knowledge graph** adds → a map of **who** and **what** is **connected to whom**, built from the case file
- How an assistant helps investigate graphs, showing its work
- A live demonstration

Three clarifications before we continue:

Local LLM

We use locally deployed, private models, for:

- Chat
- Query generation
- Data extraction
- Chat completions

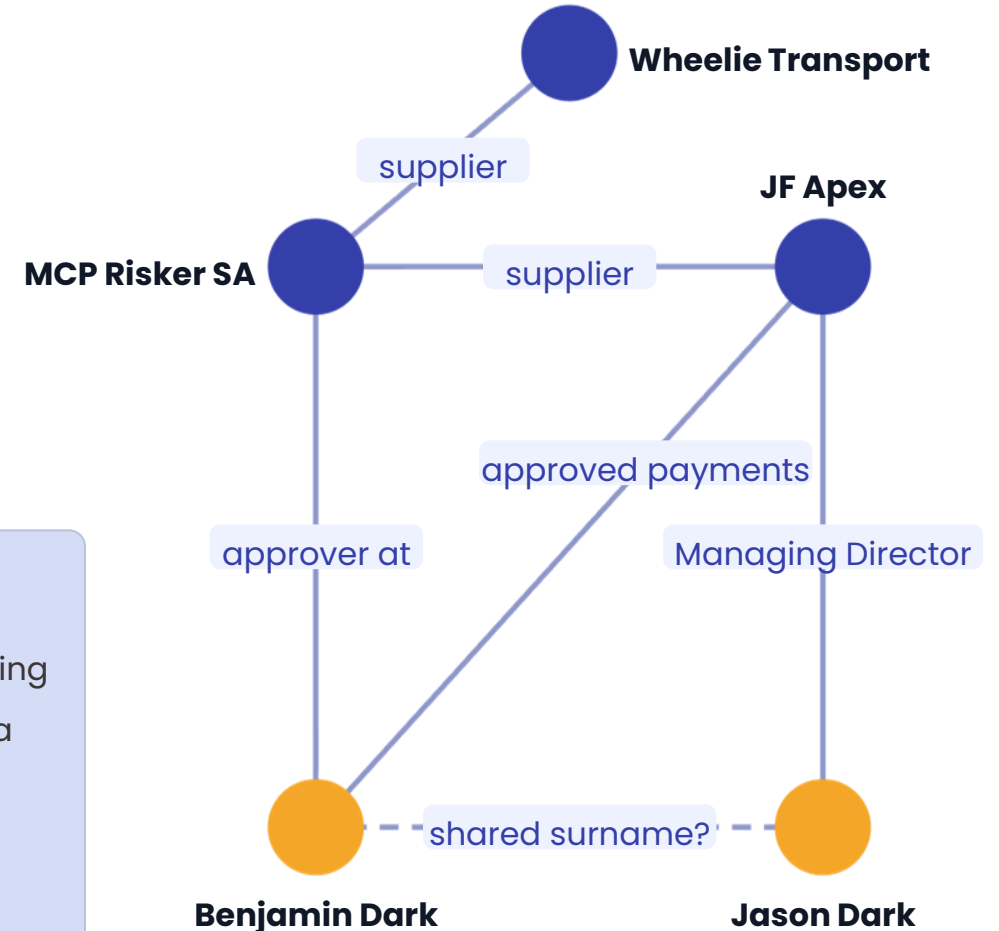
Knowledge graph

A map of people, companies and payments, and the lines between them. From:

- Structured data
- Unstructured text

Cypher

The language for asking that map questions, a bit like SQL for connections



Blue → company

Yellow → person

Dashed line → a lead

The Data Is There. Detecting Risk Is Difficult.

- Overwhelming volumes
- Many formats and languages
- Hidden connections

Issue	In a tax investigation
Missed relationships	A concern about an approver and a supplier is noted, but its supporting records are not checked together
Incomplete decisions	A risk assessment built on the few passages a search returned, not the full file
Wasted time	Cross-referencing ledgers, contracts and emails by hand
Unmanaged risk	A pattern across payments stays invisible until it becomes a liability



You can't search for what you don't know exists.

The Case File: MCP Risker SA

A small simulated company file

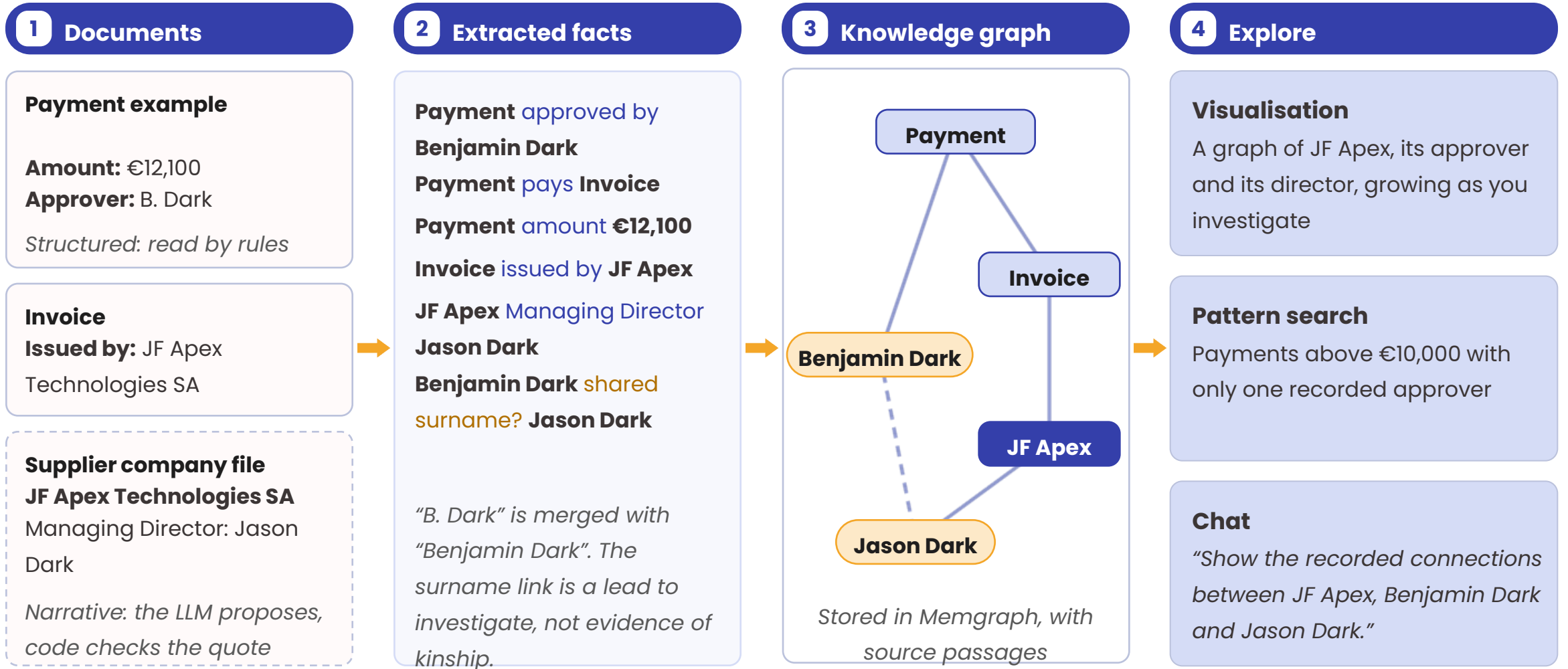
- ~ 50 documents
- Suppliers such as **JF Apex Technologies** (IT consultancy) and **Wheelie Transport** (logistics)
- People:
 - Finance Manager
 - Managing Director
 - Supplier directors
 - Staff

In the file	What we want to connect
Invoices, payment ledgers, GL extracts	Amounts, invoice references, periods, approvals
Contracts, amendments, policies	What <i>should</i> happen
Emails, audit material, whistleblower log	Explanations, concerns, denials
Company files, conflict-of-interest register	Who is who, and declared relationships



Concerns were planted in this case, some already raised in the audit material. Today isn't about the AI "discovering" fraud: it's about how fast and thoroughly you can **test a suspicion** across the supplied records.

The Workflow: From Document to Answers



Prepared example from the simulated MCP Risker case file: records combined across three documents. The demo follows the same path live.

Trust, Up Front: Four Principles

Four principles before any technology. If you remember one slide, remember this one.

1 Start with the records

Rules read structured amounts; values proposed by the model are checked against the source. Both still need review.

2 You decide when it digs deeper

Long, multi-step searches only start when you say so, and you can stop them.

3 It tells you what it couldn't do

Nothing found, a known gap in the data, too many results: it says so, and why.

4 A lead, not a verdict

Inspect supporting sources and available excerpts. The judgement is yours.

What Is RAG?

- RAG is “Retrieval-Augmented Generation”
- Like a search engine, it lets the AI read the relevant documents before answering
- It enables answers over private documents

Remember: this *vector-search approach finds passages by meaning, beyond exact keywords.*

Indexing: Preparing the documents

1. Chunk

Split documents into small passages

2. Embed

Turn each chunk into an ‘embedding’, a numerical fingerprint of its meaning

3. Store

Save the chunks with their embeddings

Search: Answering a question

1. Retrieve

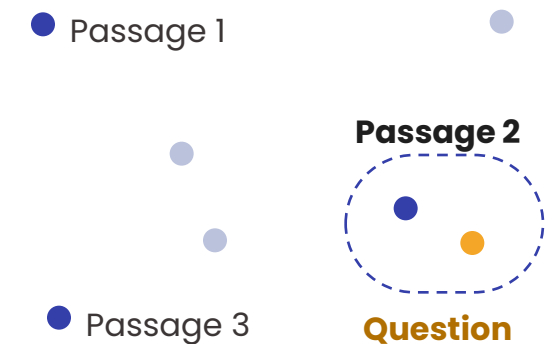
Find the chunks whose meaning is closest to the question

2. Generate

Pass them to a language model, which writes a grounded answer

SEARCH BY MEANING

Every passage, and the question, becomes a vector: a list of numbers
For example: [0.61, 0.32, 0.08, ...]



“Who has to sign off large payments?” lands next to passage 2: the two-approver rule.
Illustrative: real vectors have hundreds of numbers.

Where RAG Falls Short

Works well

"What does the JF Apex invoice say?"

The answer is in one or a few passages

Starts to fall short

"Which payments, approvals and supplier records support the concerns about JF Apex?"

Checking the answer requires assembling supporting records

Three failure modes:

1 Relevant context can be missed

Asking about the Wheelie fuel surcharge may find the documents that mention it, but not the amendment register you need to check it.

2 Spread across documents

The ledger has the payments, the company file the director, the register the declarations.

3 Checking requires connections

The surname match is in the documents. Checking it is a **multi-hop** question: follow both people to their roles, approvals and records.

Remember: finding the memo is useful. Checking its claims across records is a different task.

GraphRAG (Graph-Based RAG): The Idea

- Here, GraphRAG organises the case file *before* any question is asked
- People, companies, payments, contracts and their relationships are extracted and stored as a **knowledge graph**

SQL or rules can do a fixed reconciliation. The graph adds flexible exploration of relationships across structured records and narrative documents.

Simple passage retrieval

A “bag of document chunks”

Search returns passages

Relies on the question to describe what to find

Passage similarity alone cannot add up totals or follow links across records

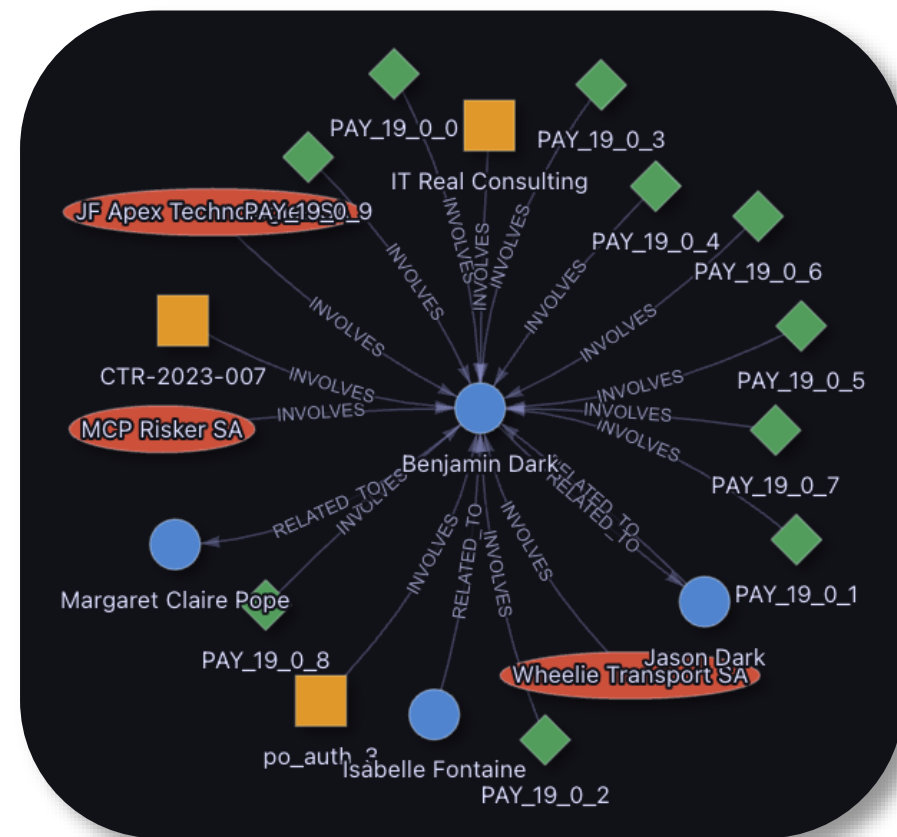
Graph plus passage retrieval

An organised map of entities and relationships

Search returns records, links and paths, with their passages

Follows connections the question never mentions

Counts, totals and paths computed over the extracted records



Building the Knowledge Graph: Four Steps

From documents to a graph

1. **Structured records** → **rules**. Rules read supported fields in invoices, ledgers and contracts: identifiers, amounts, dates.
2. **Narrative documents** → **a local LLM**. Emails, memos and logs: the model fills in a fixed set of record types and points to the supporting quote.
3. **Check and merge**. Check model-proposed values and relationships against the source text. Merge name variants ("B. Dark" = "Benjamin Dark").
4. **Store** the graph and the passages together in **Memgraph**, a graph database.

What we extract

A fixed set of record types, the **schema**:

People

Organisations

Transactions invoices, payments

Agreements contracts, amendments

Amounts

Claims what a source says

You can inspect supporting sources and available excerpts.

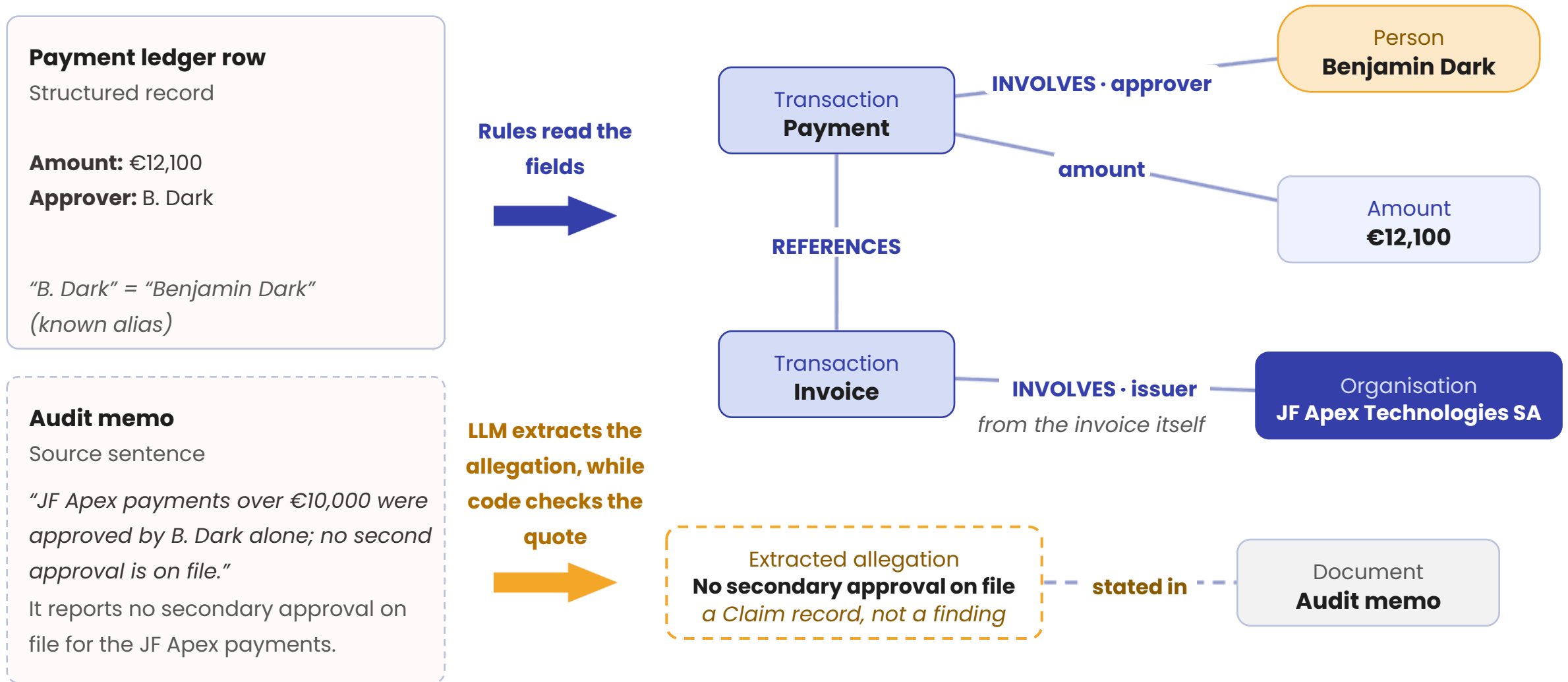
A **"triplet"** is one line on the map: who → how they're connected → to what.

📌 **Remember:** use rules for structured values, and check what the model proposes against the source.

From the ledger row. Approver = B. Dark



From Records to the Graph: Structured Extraction



What Should Happen vs What Did Happen

*"An investigation can start with the gap between what **should** happen, in policies and contracts, and what is **recorded**, in the transactions."*

- The graph puts both side by side: contracts, amendments and policies **beside** invoices, payments and approvals
- A gap is a **lead to review**, not proof of wrongdoing

From today's case:

What should happen

Payments above **€10,000** need **two approvers**.

What is recorded

Ten payments of **€12,100** to JF Apex; the ledger names B. Dark as approver. The audit memo reports no secondary approval on file.

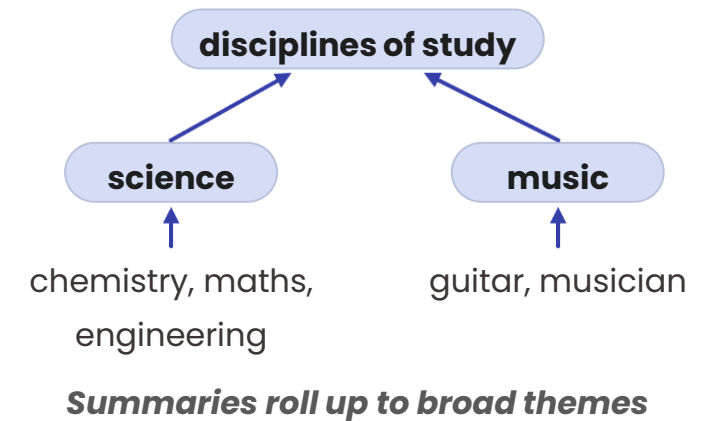
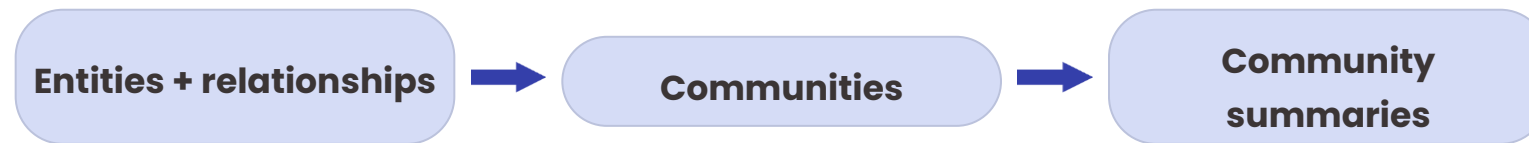


Tax-investigation link: these records help review whether recorded expenses are supported and authorised. Establishing the tax treatment or any tax loss needs the relevant tax records, applicable rules and further review.

Taking GraphRAG Further: Communities

- A graph answers **local** questions about named parties; a **global** one, like “What are the main concerns in this file?”, has no single passage or path
- Microsoft's GraphRAG (2024) groups well-connected entities into **communities**: a supplier, its invoices, its staff and the people who approve its payments
- Communities form a **hierarchy**, from small groups to broad themes, and an LLM writes a short **summary** of each

Before any question is ever asked:



Remember: things that are frequently connected belong together. That gives you a table of contents for the whole case.

Reference: D. Edge et al., “From Local to Global: A Graph RAG Approach to Query-Focused Summarization”, [Microsoft Research](https://www.microsoft.com/research/publication/from-local-to-global-a-graph-rag-approach-to-query-focused-summarization), 2024. arxiv.org/abs/2404.16130

Communities in Practice

1 Detect

Leiden community-detection runs inside the graph database via a query to detect 'clusters' of relationships

2 Tune

A few settings are tested, such as gamma (γ), which sets group size, in order to get optimal groupings

3 Summarise

One short LLM summary over every community (i.e. group/cluster) yields a cited report per community

4 Use

Deep investigations start here, then go to the records

Simplified, and not yet done

- **No roll-up:** one level of reports, built directly from source passages
- **Partial coverage:** on a sparse file many entities have too few links to group
- **Heuristic score:** hand-weighted, with case-specific anchor names
- **Not yet done:** modularity or conductance metrics; a connected-components baseline; configurable anchors per case

Searching the Graph: Three Ways

Pick the search that fits the question

Remember: structured queries for recorded facts, meaning search for context, and a deep investigation to turn a broad question into specific checks.

Graph query (Cypher)

In plain words

A structured query over the stored records

Good for

Counts, totals, paths, “who approved what”

Passage search (vector)

In plain words

Search by meaning

Good for

Explanations, emails, concerns

Deep investigation

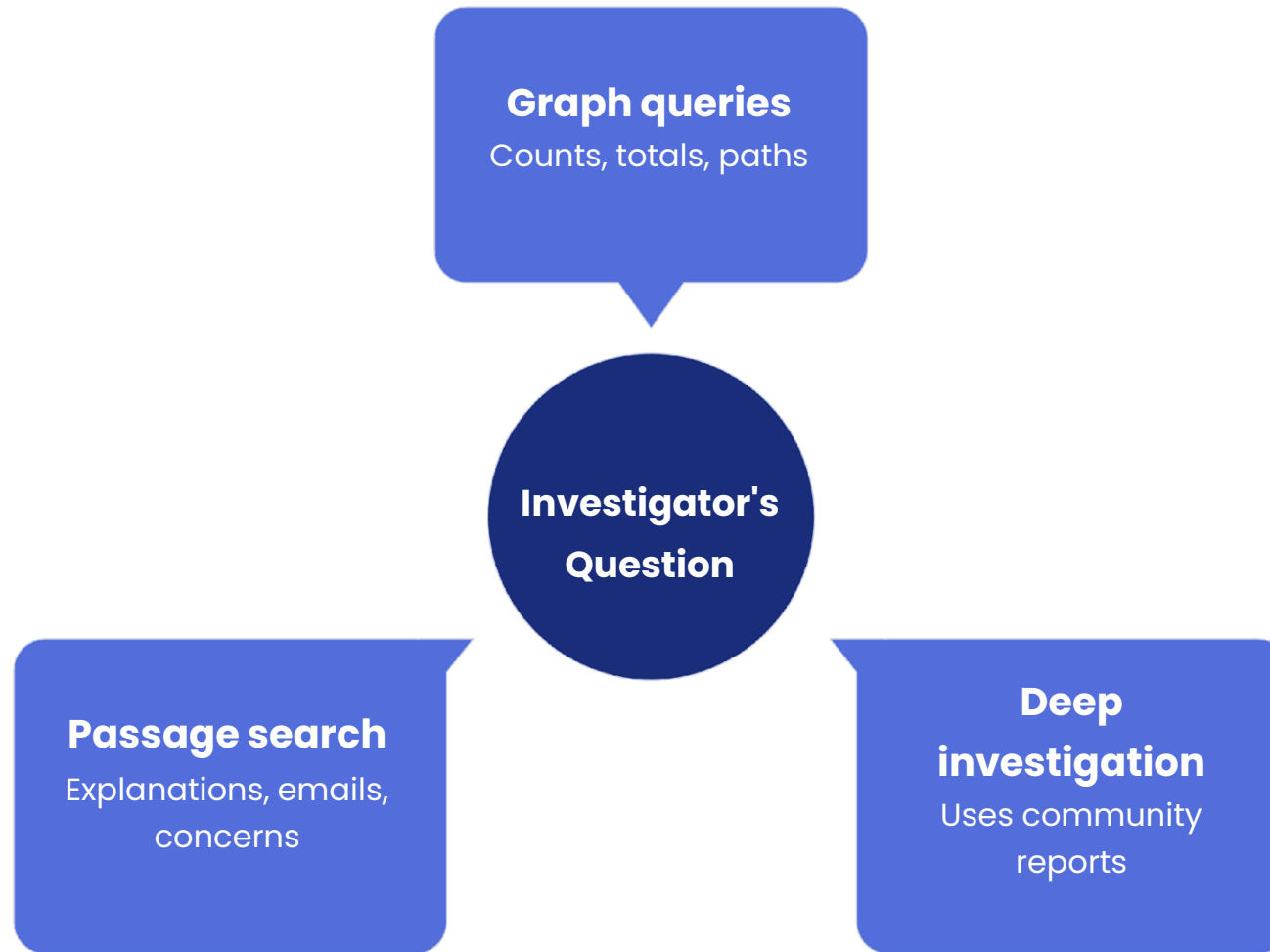
In plain words

The AI runs a mini-investigation: passages, recorded connections and community reports, over several rounds

Good for

Open questions

The Investigator's Question at the Centre



Questions to Cypher: Structure Does the Work

Ask in plain language → the assistant writes the graph query → **you can see and check it.**

Knows the map

It knows the real structure (its **schema**): which kinds of things exist and how they connect.

Checked before it runs

Code checks the query's structure before it runs, not its meaning; the model can make one targeted correction.

Shows its work

A short summary of what the query did and, on request, a plain-English account of how it understood you.

QUESTION → **Becomes a generated CYPHER-query (Similar to SQL)**

"How much was paid to JF Apex in 2024, and who approved those payments?"

 **Remember:** the structure around the model (schema, checks, one targeted correction) does much of the work.

Six Lessons for Builders

1

Let the model point and let code read where possible

Use rules for supported values; expose the cases that still rely on model interpretation.

EXTRACT

2

Lint generated queries against the live schema

Check them before they run and allow one targeted repair. Valid isn't the same as correct.

QUERY

3

Check known data limitations before generating a query

Our payment-flow check is one targeted example.

QUERY

4

Probe result size within a cap and disclose reductions

Distinguish filtered records from aggregated rows.

INSPECT

5

Watch for silent truncation

Our local runtime cut over-long prompts from the start without an error.

INSPECT

6

A citation-ID check isn't a support check

It proves the passage was supplied, not that it backs the sentence.

INSPECT

Deep Investigation

1 Propose

What it will check, who's involved, what the records *cannot* establish. **The deep investigation starts only when you press Start.**

2 Start broad

Community reports and the most relevant passages

3 Follow up

It asks itself a few follow-up questions and digs into each

4 Answer

Findings with citations, counterevidence and innocent explanations, plus what's still unanswered

ILLUSTRATIVE EXAMPLE

1 The report it started from

JF Apex: an IT supplier paid ten times in 2024, every payment approved by B. Dark.

2 A follow-up it asked itself

Follow-up 2/4: Is a second approval recorded for any JF Apex payment?

3 The source passage it led to

Audit memo: no secondary approval is on file for the JF Apex payments.

4 What it says is still unanswered

Still open: whether the services were delivered and accepted.

One Investigation, Two Views

ASK: the chat

- Ask questions in plain language
- See the graph query and the passages used
- A graph workspace that grows as you investigate

VERIFY: the evidence workspace

- Choose a prepared review or rule-based check
- Confirm its scope and run it
- Inspect the results and sources, record your assessment, and export the review

Calculated results, prepared guidance and optional AI explanations are labelled separately.

-  The chat helps you find things. The workspace helps you check the supporting records and document your assessment.
The decision is always yours.

Honest by Design

A good assistant tells you what it *couldn't* do.

Still working?

One live progress line: what it's doing and for how long.

Nothing found?

It checks why ("this name isn't in the file") and may offer a clearly labelled alternative.

Can't be answered from this data?

For known gaps, like payment flows, it says so and offers what *can* be shown.

Too many results?

It explains how the result was reduced and lets you request a larger, bounded result.

- ❏ This is what separates a demo from a tool. Known limitations and reduced results are disclosed; other mistakes can still look plausible, so check the evidence.



Demo

Extraction → Orient → Ask → Connect → Verify → Should vs did → Honesty

Demo: From One Document to the Graph

Extraction

Orient

Ask

Connect

Verify

Should vs did

Honesty

WE SHOW

One prepared sentence from the audit memo
→ the allegation the LLM extracted → its
record in the graph

WATCH FOR

Stored as a claim linked to its source: what the memo says, not
a finding. Ledger rows are read by rules instead.

Demo: Orient

Extraction

Orient

Ask

Connect

Verify

Should vs did

Honesty

WE ASK

"Give me an overview of the case: the main companies, people and any concerns raised."

WATCH FOR

The live progress line, the query it ran, and the sources it used.

Demo: Ask About JF Apex

Extraction

Orient

Ask

Connect

Verify

Should vs did

Honesty

WE ASK

"How much was paid to JF Apex in 2024, and who approved those payments?"

WATCH FOR

Three different things: what a source says, what we extracted from it, and how the model interprets it.

Demo: Connect, the Opening Question

Extraction

Orient

Ask

Connect

Verify

Should vs did

Honesty

WE ASK

"Show the recorded connections between JF Apex Technologies SA, its payment approver Benjamin Dark and its Managing Director Jason Dark."

WATCH FOR

A shared surname is a lead to investigate, not evidence of kinship or a conflict of interest.

Demo: Verify Against the Ledger

Extraction

Orient

Ask

Connect

Verify

Should vs did

Honesty

WE DO

Build the prepared JF Apex review from the supplied records, then compare its ledger rows with the extracted graph.

WATCH FOR

Agreement verifies extraction consistency, not whether the services occurred or whether every chat answer is correct.

Demo: A Balanced Picture

Extraction

Orient

Ask

Connect

Verify

Should vs did

Honesty

What the records show

Ten payments to JF Apex

€100,000 excl. VAT + **€21,000** VAT =
€121,000 paid

Invoice periods January–October 2024;
paid through November 2024

One recorded approver: B. Dark

Not an assessed tax loss.

The explanation on file

Benjamin Dark's recorded
explanation: JF Apex provided
complementary specialist services.

*His account, to be tested, not an
established fact.*

Next evidence request

- Deliverables
- Service acceptance
- Independent approvals

*What would support or contradict his
account.*



A shared surname is a lead to investigate, not evidence of kinship or an established conflict of interest.

Demo: Should vs Did, a Pattern Search

Extraction

Orient

Ask

Connect

Verify

Should vs did

Honesty

WE ASK

“Which payments above €10,000 have only one recorded approver?”

WATCH FOR

The policy requires two approvers above €10,000. A gap is a lead to review, not proof.

Demo: Finding evidence

Extraction

Orient

Ask

Connect

Verify

Should vs did

Honesty

WE ASK

"Show me circular payments that leave Jason Dark through 2 to 4 companies and return to him."

WATCH FOR

It says what this data can't establish, and offers what it can show instead.

Demo: Deep Investigation

The assistant's own multi-step research, with your approval

WE ASK

“What do the records establish about Benjamin Dark’s payment approvals and Jason Dark’s role at JF Apex, and what remains unverified about their relationship?”

WATCH FOR

The plan it proposes, your Start, its follow-up questions, and what it says is still unanswered.

Demo: A Separate Synthetic VAT Case

A separate synthetic case, unrelated to MCP Risker, built to show VAT checks

WE RUN

Run VAT reconciliation on the supplied invoices.
Panelworks: one invoice posted twice, and both postings counted in the VAT working paper, so the VAT is claimed twice. Then Toolcare: two postings plus a reversal, so the VAT matches.

WATCH FOR

A duplicate VAT claim is not a duplicate cash payment, and a reversal can explain an apparent duplicate. Code, not the AI, computes the VAT difference.

Kinds of Investigative Question

Entity-centric

"What do the documents say about Benjamin Dark?"

Network

"How are JF Apex's recorded approvers and management connected, and what do those links establish?"

Temporal

"When did the Wheelie surcharge start relative to the contract?"

Cross-document

"For which suppliers do the supplied records report missing contracts or delivery evidence?"

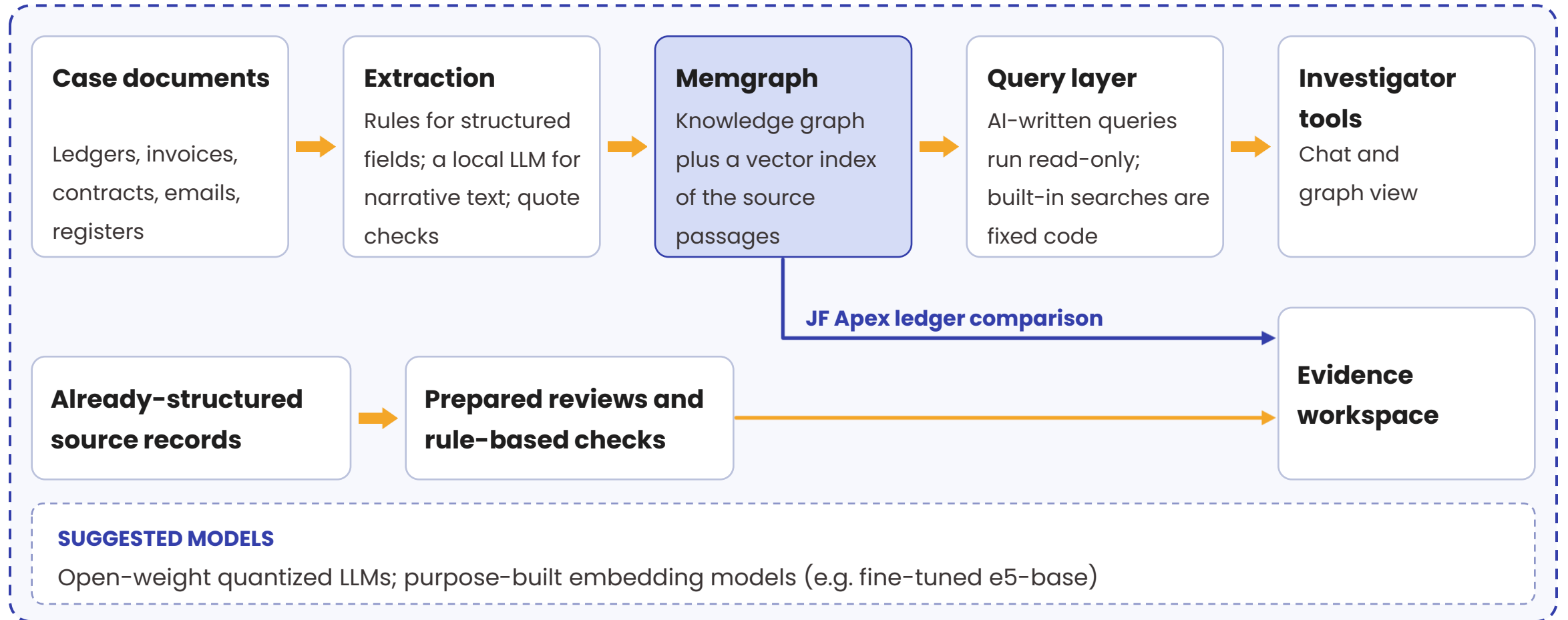
Investigative summary

Investigate deeper. "What explains the JF Apex overspend?"

Out of scope

For known gaps, like payment flows, it says so.

How It Runs: Architecture



📄 In our setup, no case data is sent to a cloud AI provider.

Sovereign and Responsible

Designed to run fully on your own infrastructure

The language model, embeddings, graph database and interface all run locally, on components you host yourself.

In our setup, the case file isn't sent to any cloud AI provider.

Built with care for high-stakes use

- **Human oversight:** approval before deep investigations, Stop, “a lead, not a verdict”, and a separate verify step
- **Inspection:** retrieved evidence, generated queries and result-reduction notices can be reviewed
- **Records:** each review run retains its inputs and results; supporting records and analyst notes can be exported
- **People, carefully:** inferred links (like a shared surname) are shown as hints, and people are ranked only when you ask

Prototype: access control, audit logging and a deployment assessment are still to do.

What We've Tested, and What We Haven't

Tested

- One simulated, English, roughly fifty-document company file, end to end
- The prepared JF Apex supplier-ledger reconciliation; selected case facts checked against the documents (not a complete graph-accuracy evaluation)
- The answers, warnings and disclosures you'll see in today's demo

Not yet measured

- Extraction accuracy on a labelled case
- Query accuracy against a gold question set
- Time saved compared with a manual review
- Large, scanned or multilingual files
- Whether community reports improve answers (with vs without)

Known gaps

- Occasional extraction errors
- Answers vary between runs
- Communities are weak on sparse records
- The quote check has a loose fallback

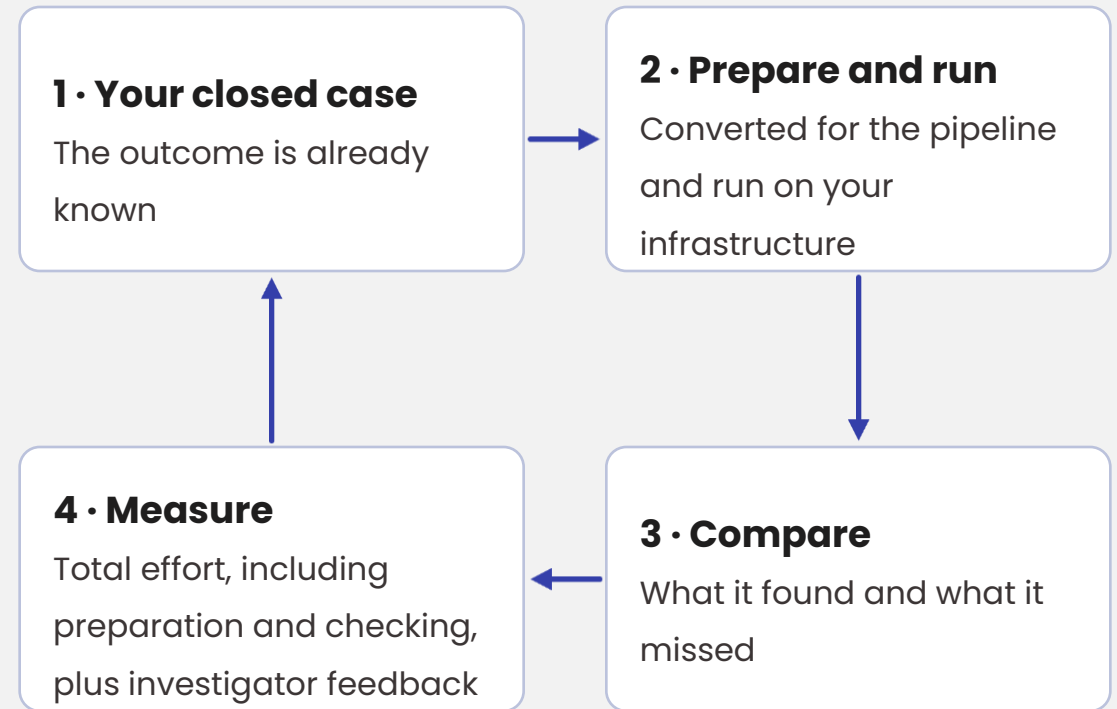
"We'd rather show you exactly where it stands than oversell it."

Next Step: Test It on a Closed Case

A shared evaluation, on your infrastructure.

- A case where you already know the outcome
- Your documents, converted for the pipeline, on your infrastructure
- We measure together: what it found, what it missed, the total effort including preparation and checking, and what your investigators thought

The aim: evidence on your cases, not ours.



A closed case gives known outcomes to measure against.

Summary

The problem	Our answer
Overwhelming volume	Prepare the records, then explore what they can answer
Hidden connections	A knowledge graph combined with passage search
Open questions	Deep investigation, with your approval
Trust	Inspect supporting sources, queries and disclosed limitations
Sovereignty	Designed to run fully on your own infrastructure

📄 **The opening question, answered:** ten payments to JF Apex, €100,000 + €21,000 VAT = €121,000 paid (not an assessed tax loss); one recorded approver; a supplier director with the same surname; and Benjamin Dark's recorded explanation of complementary specialist services. **Next evidence request: deliverables, service acceptance and independent approvals.**

An investigative prototype that supports evidence review and professional judgement, not automatic fraud determination.



Q&A + Discussions



Contact

Redfield: redfield.ai · linkedin.com/company/redfield-ab

Jacob Coles: jacob.coles@redfield.ai · linkedin.com/in/jacoblcoles