

# Building a Data Foundation Fit for the Human-AI Partnership

How structured context turns retrieval into reasoning.

## CONTENTS

- 01 The Problem in a Nutshell
- 02 The Problem Explained: Why RAG Alone Is Not Enough
- 03 The Three Missing Layers
- 04 Building a Context Layer
- 05 How We Build It
- 06 Getting Started

## 01

### The Problem in a Nutshell

Somewhere in your organisation right now, someone is asking an AI tool a perfectly reasonable question about your business, and the tool is confidently giving them an answer that is either wrong, incomplete, or six months out of date. Nobody knows this yet, because the answer sounds plausible, and that, in a nutshell, is the problem with how pharma has been implementing AI.

The technology behind most of these implementations is called Retrieval-Augmented Generation, or RAG, and it genuinely works well for simple tasks:

finding a clause in a contract, summarising a study report, pulling out key points from a regulatory submission. Where it falls apart is with the kind of questions that brand teams, medical affairs leaders, and commercial directors actually need answered, which are rarely simple.

Consider what happens when a brand director asks: "*Which content themes are driving the highest HCP engagement for our respiratory portfolio across digital channels in Northern Europe?*" Answering that properly requires the system to understand what content exists, how it has been classified, which HCPs saw it, through which channels, in which markets, over what time period, and what happened next. That information lives in at least six different systems: The MLR platform for content approvals, your CRM for field activity, your marketing automation platform for email engagement, your analytics tools for web behaviour, your brand plan for strategic context, and your finance system for spend data. No amount of clever document searching will connect those dots, because the connections between those systems do not exist as searchable text. They exist as relationships, and relationships require structure.

This is where most enterprise AI implementations are stuck right now. The finding-stuff part works. The writing-an-answer part works. What needs sorting out is the bit in the middle: getting the system to actually understand how your content, your customers, your channels, and your commercial outcomes relate to each other, and whether any of that information is still true today.

**57%**

of organisations estimate their data is not AI-ready

Gartner, 2025

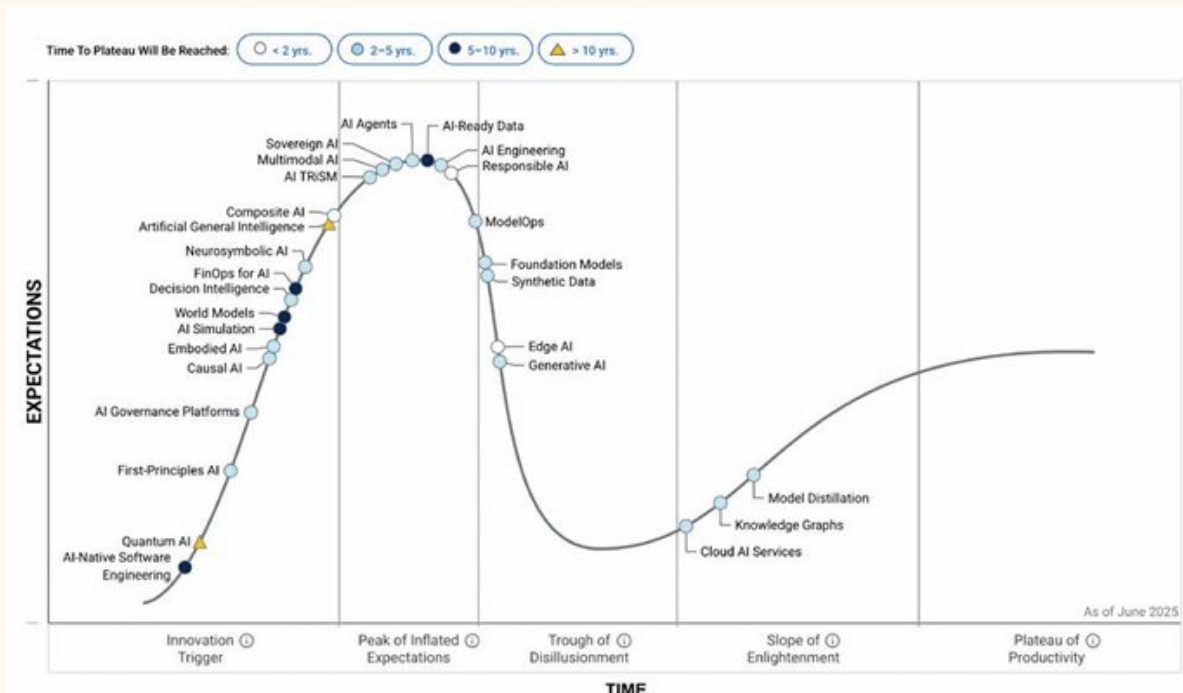
**\$1.9M**

median spend per GenAI initiative, with widespread dissatisfaction

Gartner, 2025

In Gartner's 2025 Hype Cycle for Artificial Intelligence, generative AI has slid into the "Trough of Disillusionment," Gartner's term for the stage where inflated expectations give way to the realities of integration, governance, and scaling. Fewer than 30% of CEOs are satisfied with returns on their GenAI investments. The data architecture gap is one of several factors driving this disillusionment,

but it is the one most directly addressable, and the one where structured foundations have the clearest impact on outcomes.



Source: Gartner (2025). Hype Cycle for Artificial Intelligence. Khandabattu, H. and Tamersoy, B., 11 June 2025.

Better models will not necessarily fix this. The problem is structural, and the solution is giving your AI the context it needs to actually reason about your business. The rest of this paper explains how.

02

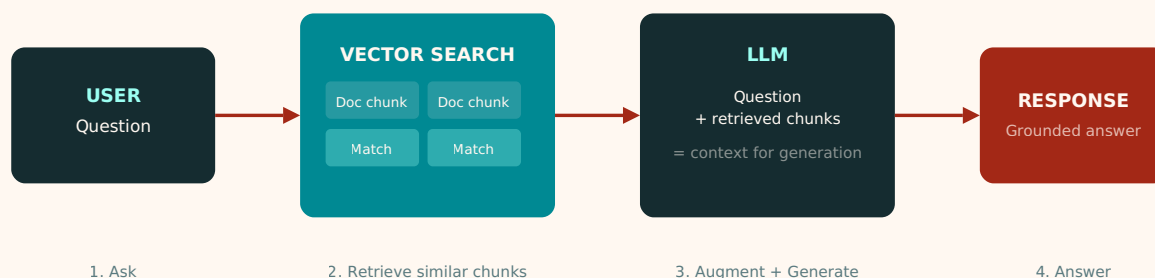
## The Problem Explained: Why RAG Alone Is Not Enough

To understand why some AI implementations underperform in pharma, it helps to understand how they actually work, because once you see the mechanics, the limitations become obvious.

### How RAG Works

A standard RAG system does three things. First, it takes your question and converts it into a mathematical representation. Second, it searches a database

of pre-processed document chunks to find the ones whose mathematical representations are most similar to your question. Third, it feeds those chunks to a large language model along with your original question, so the model can generate a response grounded in your actual data rather than its general training.



This is a genuine improvement over using a language model on its own. The model is no longer reasoning solely from its training data, which is generic. It is working with real information from your organisation. So far, so good.

#### WANT TO GO DEEPER ON RAG?

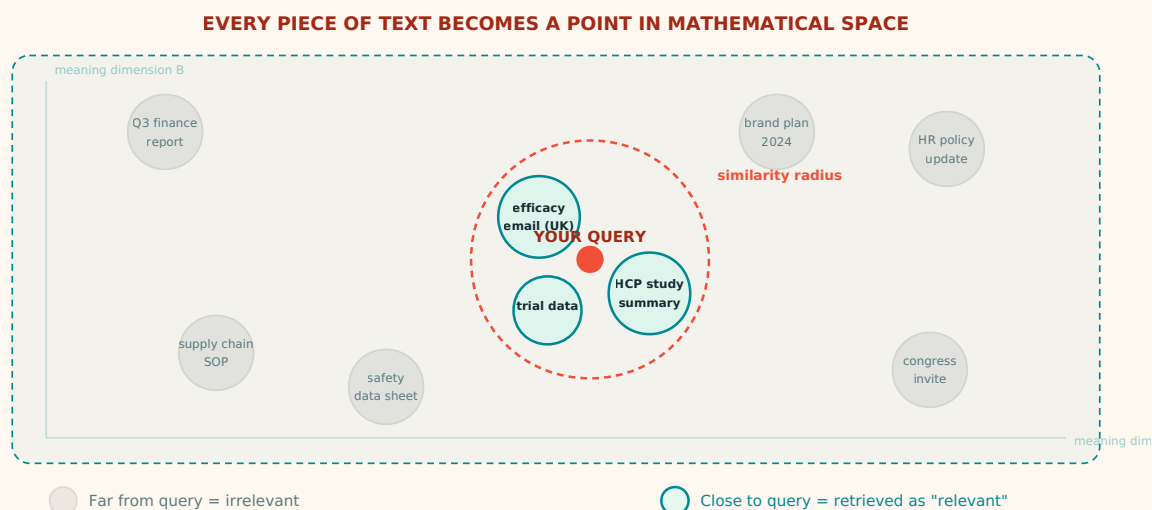
IBM's 7-minute explainer video is the clearest non-technical introduction available: "*What is Retrieval-Augmented Generation (RAG)?*" - [ibm.com/think/videos/rag](https://ibm.com/think/videos/rag). For a more detailed written explanation, IBM Research's article walks through the full architecture with practical examples: [research.ibm.com/blog/retrieval-augmented-generation-RAG](https://research.ibm.com/blog/retrieval-augmented-generation-RAG)

## How Vector Similarity Search Works

The search step is where it gets interesting, and also where it starts to go wrong.

Most RAG systems find relevant documents using something called vector similarity search. Every piece of text, your question, and every chunk of every document in the database, gets converted into a long list of numbers (called a vector) that captures the meaning of that text in mathematical form. Words and

phrases that mean similar things end up with similar numbers. When you ask a question, the system finds the document chunks whose vectors are closest to your question's vector, and those become your "relevant" results.



If this sounds like a very sophisticated version of keyword search, that is because it essentially is, just one that understands synonyms and concepts rather than exact words. It works well when the answer lives inside a single document. It struggles when the answer requires connecting information across multiple sources, understanding when something was true, or resolving contradictions between different pieces of data.

The questions that matter to brand, medical, and commercial teams almost always span multiple systems, time periods, and data types. The result is four recurring limitations that explain why AI tools in pharma consistently underdeliver despite the underlying technology being genuinely capable.

## The Four Limitations of RAG

### 1. It Struggles to Connect the Dots

Standard RAG retrieves isolated chunks of text based on similarity, but it has no reliable mechanism to traverse the relationships between different pieces of information and build a synthesised answer. Microsoft Research tested this directly in their 2024 GraphRAG study (Edge et al., 2024). They took a dataset

of thousands of news articles and asked both a standard RAG system and their graph-enhanced version to answer questions that required pulling together information from across the entire corpus, things like "What are the main themes in this dataset?" or "How does entity X relate to entity Y?" The standard RAG system consistently failed on these global questions because it could only retrieve individual text chunks matching the query. The graph-enhanced system, which first built a knowledge graph from the corpus and used community summaries to organise related entities, substantially outperformed standard RAG: in head-to-head comparisons judged by an LLM evaluator, GraphRAG was selected as the more comprehensive answer 70-80% of the time, and as the more diverse answer 57-82% of the time, depending on the level of graph summarisation used (Edge et al., 2024). A subsequent independent evaluation noted that while GraphRAG consistently outperforms standard RAG, the magnitude of improvement is more moderate when evaluation biases are corrected (Han et al., 2025), and that GraphRAG's advantage grows with query complexity.

#### **WHAT THIS LOOKS LIKE IN PHARMA**

A brand manager asks "Which HCP segments are most responsive to our efficacy-led messaging in the UK?" Answering this requires connecting content assets (tagged with "efficacy" messaging) to deployment records (filtered by UK market) to engagement data (broken down by HCP segment) to behavioural outcomes. That chain spans at least four systems, and no single document contains the answer. The RAG system will dutifully return whatever document chunks happen to mention efficacy, HCPs, and the UK in close proximity.

## **2. It Has No Sense of Time**

Standard retrieval systems treat every piece of information as equally current. There is no awareness of when facts were true, when they changed, or when they expired. Rasmussen et al. (2025) at Zep AI built a system called Graphiti specifically to address this. The idea is straightforward: instead of storing facts as flat records, you store them with timestamps that track when each fact became true and when it stopped being true. The system knows that a piece of information from last quarter may have been superseded, and can distinguish

between what is current and what is historical. Tested against existing benchmarks, this temporal awareness improved accuracy by up to 18.5% while reducing response time by 90%, with the strongest gains on questions that required understanding how information had changed over time (though performance decreased on simpler queries where temporal context was irrelevant). While these results come from the team that built the system, the underlying principle, that AI needs to know when things were true, is well-established in the knowledge graph literature.

#### **WHAT THIS LOOKS LIKE IN PHARMA**

Your system retrieves engagement data showing email open rates for a specific brand at 34%, which looks healthy. But that data is from Q2 last year, before a competitor launched in the same therapeutic area. Current open rates have dropped to 19%. The system has no way to know which figure is more relevant, because it treats everything it finds as equally valid. In an industry where product lifecycle stages, competitive entries, and seasonal prescribing behaviour all affect performance, a time-blind system routinely serves yesterday's truth as today's insight, and you might not notice until a decision has already been made.

### **3. It Does Not Know That Your Systems Are Describing the Same Things**

Limitation 1 is about the retrieval mechanism struggling to follow chains of relationships. This one is more fundamental: the system has no way of knowing that records in different systems are describing the same entity. A HCP in your CRM, a recipient in your mass email platform, and an attendee in your congress system might all be the same person, but to a retrieval system they are three unrelated data points with nothing connecting them.

#### WHAT THIS LOOKS LIKE IN PHARMA

Dr. X exists in your CRM as a cardiologist in Manchester, in your email platform as recipient ID 4472, and in your congress registration system as an attendee at ESC 2025. Ask an AI tool "What is Dr. X's engagement history?" and it will return whatever fragments it can find in whichever system it searches first. It will never build a complete picture, because it does not know these records are related. Every insight it generates will be based on a partial view, and there is no way for the user to know what is missing.

#### 4. It Fills the Gaps with Guesswork

When an AI system lacks sufficient context to answer a question properly, it will often generate a plausible-sounding answer anyway - this is referred to as hallucination. The original RAG paper (Lewis et al., 2020) introduced the retrieval-augmented approach specifically to reduce this problem, by giving the model real documents to reference rather than relying on its training data alone. It helped, but it did not solve the problem. Subsequent research has shown that hallucination rates increase when the retrieved chunks are incomplete, contradictory, or pulled from the wrong context. Deloitte's whitepaper on knowledge-enriched generative AI (2023), published alongside their expanded alliance with Neo4j, argues that structured knowledge graphs can reduce hallucination by providing verified, relationship-aware context rather than raw text fragments, essentially giving the AI a fact-checked reference rather than a pile of loosely related documents. This argument is supported by subsequent empirical work: the Lettria Hybrid GraphRAG benchmark (December 2024), conducted by graph-enhanced RAG vendor Lettria, tested across four domains (finance, healthcare, industry, and law) and found that graph-enhanced retrieval achieved 80% accuracy versus 50.83% for vector-only RAG, with healthcare domains showing 85.7% versus 55.2%.

#### WHAT THIS LOOKS LIKE IN PHARMA

A commercial analytics team asks their AI tool to summarise a campaign's content ROI. The system finds some spend figures and some engagement numbers in its retrieved chunks and generates a confident-sounding summary attributing a 3.2x return. The problem is that it pulled the spend from one campaign and the engagement from a different one with a similar name, because the system had no structural way to keep them separate.

These limitations are well documented, and the research consistently points to the same solution: structured knowledge. When RAG is backed by a context layer rather than a flat document store, the improvements are significant.

**70-80%**

win rate for GraphRAG over standard RAG on comprehensiveness in head-to-head comparison

Microsoft Research, 2024

**18.5%**

accuracy gain from temporal awareness in knowledge graph reasoning

Rasmussen et al., 2025

**A note on modern RAG architectures.** Since the original RAG paper (Lewis et al., 2020), retrieval-augmented approaches have evolved considerably. Agentic RAG systems use autonomous agents to plan multi-step retrieval, cross-reference sources, and self-correct, partially addressing the "connecting the dots" limitation described above. Hybrid search combining lexical retrieval (matching exact keywords and phrases) and semantic retrieval (matching meaning, even when different words are used) has become standard in production deployments. These advances are genuine improvements. However, they address the retrieval problem without solving the structure problem. An agentic system that performs five retrieval steps across your document store is still searching text: it cannot traverse typed relationships between entities, distinguish when a fact was true, or resolve that records in different systems describe the same person. Multi-step retrieval finds more documents. A context layer understands how those documents relate to each other.

A related counter-argument is that expanding context windows will make retrieval unnecessary altogether. LLM context windows have grown from 4,000 tokens in 2022 to 200,000-2,000,000 tokens in 2025, with experimental models reaching 10 million. Why not just feed everything in? Because research consistently shows that performance degrades well before stated limits. Models exhibit a "lost in the middle" phenomenon where information buried in large contexts is missed (Liu et al., 2024), and independent benchmarks show accuracy dropping significantly at extended context lengths (Paulsen, 2026). Larger windows also increase cost linearly. You can give the model access to more data, but without explicit relationships between entities, it still cannot reliably connect a content asset to its deployment channel to its engagement outcome to its strategic objective. More hay does not help you find the needle. It makes the haystack bigger.

The limitations described above apply to any retrieval-first approach, however sophisticated, that operates over unstructured or semi-structured data without an explicit relationship model.

The fundamental limitation across all four problems is the same: standard RAG gives your AI access to words, but not to meaning. It can find documents that look relevant. It cannot understand the relationships between the things those documents describe. For that, you need structure. A context layer provides it, and building one is what the rest of this whitepaper is about.

**WANT TO GO DEEPER ON RAG LIMITATIONS IN ENTERPRISE SETTINGS?**

McKinsey's analysis "*Generative AI in the pharmaceutical industry: Moving from hype to reality*" covers where AI is delivering value in pharma and where it is falling short: [mckinsey.com](https://www.mckinsey.com). The original GraphRAG paper (Edge et al., 2024) includes side-by-side comparisons of standard vs graph-enhanced retrieval: [arxiv.org/abs/2404.16130](https://arxiv.org/abs/2404.16130)

# The Three Missing Layers to Creating Structure

If RAG gives your AI the ability to find information, the three layers described here give it the ability to understand what it found. Each builds on the one before it, and together they turn a retrieval system into a reasoning system that can work within the reality of pharmaceutical commercial teams, which, as anyone who has tried to pull a cross-channel engagement report knows, is considerably more complicated than it might seem on the surface.

## Layer 1: Rules

Rules are the explicit instructions that tell the system how to interpret what it finds.

In pharma, this means encoding the logic that is currently trapped inside people's heads: the regulatory constraints, the commercial environment, the therapeutic area knowledge, and the operational processes that shape how content gets created, approved, deployed, measured, and optimised. Without rules, an AI system is trying to make sense of your data using only general knowledge, which is a bit like asking someone who has never worked in pharma to explain your MLR workflow by reading the user manual. Technically possible, but not something I'd bet on.

Rules do several things at once. They enforce consistency by catching contradictions, flagging when an asset has been classified under two categories that cannot both be true. They validate data quality by restricting what values a property can take. And they complement the pattern-matching that AI models are good at by providing a layer of logic that humans can actually inspect, verify, and trust.

### **What rules look like in practice:**

#### **RULE 1: CHANNEL CLASSIFICATION**

"If an asset is deployed via CRM with a send date, classify the channel as 'Rep Triggered Email'. If the same asset is deployed via marketing automation to a segment list, classify it as 'Mass Email'.

*Why this matters:* the same content asset performs completely differently depending on how it reaches the HCP. If the system cannot distinguish between them, downstream metrics, engagement rates, content effectiveness, channel ROI, will be structurally wrong.

#### **RULE 2: MARKET-SPECIFIC INDICATION GATING**

"If a content asset references an indication for a brand, only tag that indication to the asset in markets where it is approved. If the indication is pending or not approved in a given market, flag the asset and exclude it from that market's content library."

*Why this matters:* a brand can be approved across multiple indications, but those indications may not be approved at the same time in every market. A content asset promoting one indication cannot be tagged and served in a market where that indication has not yet been approved. Without this rule, the system has no way to distinguish between globally available content and market-specific content, and risks surfacing assets that reference indications your local team is not permitted to promote.

Rules are not the AI model. They sit alongside it and act as guardrails, ensuring that everything the system produces respects the constraints of your business, your regulations, and your domain.

A natural question is whether these rules could simply be encoded as instructions in an LLM prompt. They can, up to a point. But prompt-based rules are fragile: they cannot be versioned, audited, or tested independently of the model. When a prompt-based rule fails, it fails silently, the model simply classifies incorrectly, and nobody knows until the downstream analysis is wrong. Encoding rules as structured logic that sits alongside the model means they can be inspected, validated, and maintained by your teams without requiring AI expertise.

## Layer 2: Classification

Classification is the bridge between your business language and the system's understanding of your data, and frankly, this is where you've got to put in a good bit of work.

Every pharmaceutical company has its own vocabulary, its own way of describing therapeutic areas, brand strategies, customer segments, channel types, content categories, and campaign structures. These vocabularies are rarely documented in one place, and they are unlikely to match the generic categories that off-the-shelf AI systems understand. When a brand team talks about "Differentiation" key messages, they mean something very specific to their brand, their market, and their therapeutic area. A generic AI system will nod along and pretend it understands.

Classification typically covers three areas:

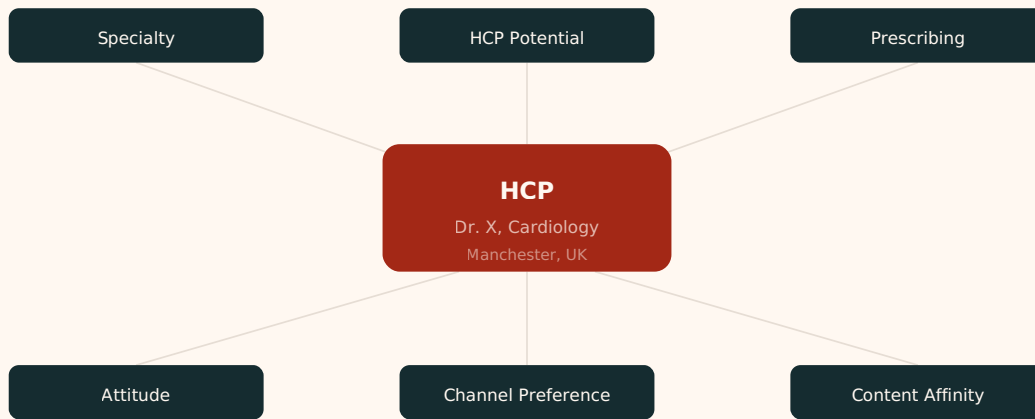
### Content Taxonomy

How assets are categorised by theme, key message, format, and strategic intent.



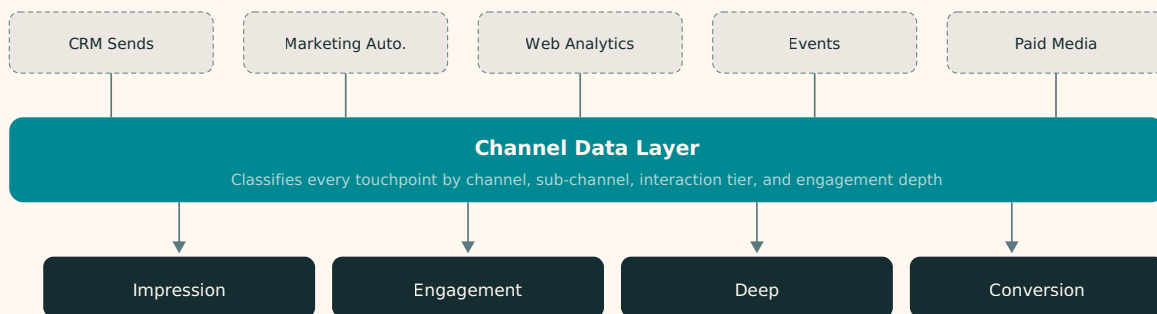
### Customer Data Model

How HCPs are described by specialty, prescribing behaviour, attitudinal segment, and HCP potential.

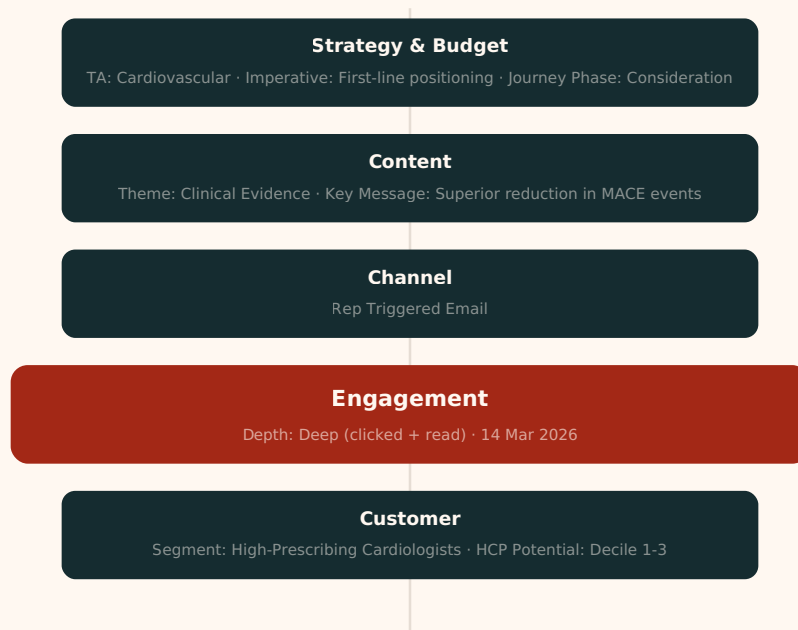


## Channel Data Layer

How touchpoints are classified across digital, field, and event-based interactions.



Together, these give the system a consistent vocabulary to work with across every part of your commercial operations.



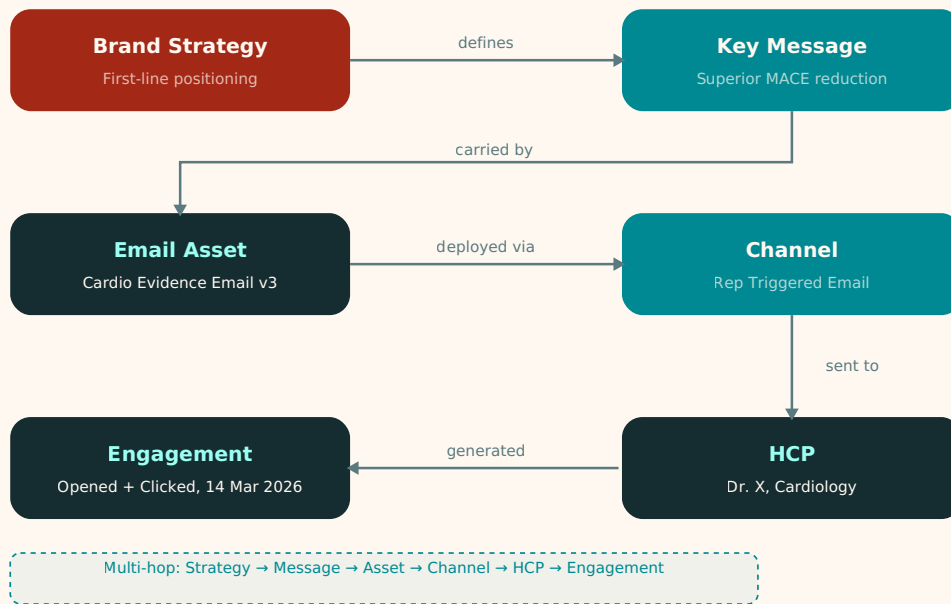
A single email asset might be classified as: Strategic Imperative = "Establish first-line positioning," Therapeutic Area = Cardiovascular, Journey Phase = Consideration, Content Theme = Clinical Evidence, Key Message = "Superior reduction in MACE events," Channel = Rep Triggered Email, Target Segment = High-Prescribing Cardiologists, HCP Potential = Decile 1-3. Without these labels applied consistently across every asset, every channel, and every customer segment, the system cannot compare performance across campaigns, cannot identify which messages resonate with which audiences, and cannot trace what actually drove the outcome.

Classification is the foundation. Until the system understands your business language, it cannot reason about your business.

### Layer 3: Knowledge Graph

A knowledge graph is a way of representing data as a network of entities (things) and the relationships between them.

Rather than storing information as isolated records, a knowledge graph makes the connections between those records explicit and traversable. The system can start from one entity, follow its relationships to other entities, follow those relationships further, and keep going until it reaches an answer. Complex questions that span multiple data sources become answerable, because the connections are built into the structure itself.



For pharmaceutical commercial teams, a knowledge graph connects a brand strategy to the key messages it defines, connects those key messages to the content assets that carry them, connects those assets to the channels they were deployed through, connects those channels to the HCPs who received them, and connects those HCPs to what they actually did next. Every connection is an explicit, typed relationship: a defined link that says "this asset carries this key message" or "this HCP received this asset through this channel on this date."

#### WANT TO GO DEEPER ON KNOWLEDGE GRAPHS AND GRAPH RAG?

IBM's explainer "*What is a Knowledge Graph?*" is a good starting point: [ibm.com/think/topics/knowledge-graph](https://ibm.com/think/topics/knowledge-graph). Microsoft Research's blog post explains how GraphRAG outperforms standard RAG with clear comparisons: [microsoft.com/research/blog/graphrag](https://microsoft.com/research/blog/graphrag). The Neo4j *GraphRAG Manifesto* includes LinkedIn's 28.6% reduction in resolution time: [neo4j.com/blog/genai/graphrag-manifesto](https://neo4j.com/blog/genai/graphrag-manifesto). A note on terminology: RAG, GraphRAG, and Knowledge Graphs are all industry terms that describe variations of the same core concept: giving AI structured access to your data. The labels differ, but the underlying principle is consistent.

# Building a Context Layer

The three layers described above, rules, classification, and the knowledge graph, give your AI the ability to understand your data. A context layer is what you get when you bring all three together and enrich them with additional dimensions that reflect how information actually behaves in the real world.

Rules tell the system how to interpret what it finds. Classification maps your business language onto a structure the system can work with. The knowledge graph connects everything into a traversable network of relationships. A context layer takes that connected network and adds the dimensions that make it genuinely useful for decision-making: when those relationships were true, where they apply, how confident you should be about them, and where the information came from in the first place.

The research literature uses the term "context graph" to describe knowledge graphs enriched with these additional dimensions. We prefer "context layer" because it describes more accurately what it does in a data architecture: it sits on top of your existing systems and makes them queryable, connected, and useful to both humans and AI.

This matters because pharma commercial operations are not static. Campaign performance shifts week by week. HCP attitudes move with product lifecycle stages. Competitive landscapes change with new market entries. Content effectiveness varies by market, channel, and audience. A knowledge graph that connects your data is valuable. A context layer that understands how that data changes over time, where it came from, and how much you should trust it is the fundamental plumbing to effective use of AI.

A context layer encodes five dimensions alongside the data, and these dimensions mirror how humans actually reason about complex situations.

**Temporal context** tracks when relationships existed, changed, or expired. The system understands that Dr. X's prescribing behaviour last quarter may differ from today, that a content asset was approved for a specific period, or that a campaign ran in a defined window. Without it, you get the time blindness problem described earlier.

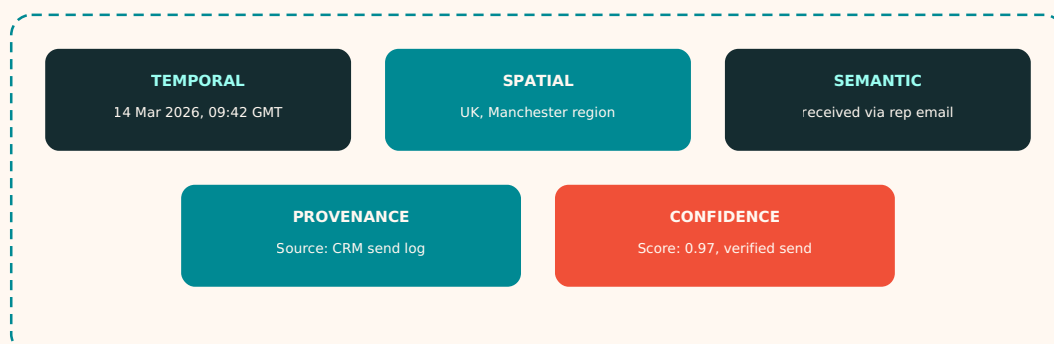
**Spatial context** captures how geography affects relationships. Market access in France differs from the UK. HCP engagement norms vary by region. Regulatory environments shape what content can be deployed where. Insights generated for one market should not be blindly applied to another.

**Semantic context** defines the precise nature of relationships. Instead of generic "related to" connections, the context layer specifies whether one thing "supports" another, "contradicts" it, "depends on" it, or "replaced" it. Distinguishing between a content asset approved to carry a key message and one merely tagged with a similar theme matters for attribution and compliance alike.

**Provenance** tracks where information came from. In pharma, understanding the origin of a data point matters for trust and auditability. Knowing whether engagement data came from your CRM, your marketing automation platform, or a third-party data provider gives your teams the audit trail they need to validate insights and trace decisions back to their source.

**Confidence** assigns certainty scores based on data quality and source reliability. Not all data is equally trustworthy. Confidence scoring ensures the system does not present a weakly-supported insight with the same authority as a well-evidenced one, which helps your people calibrate their trust appropriately, rather than either blindly accepting or wholesale rejecting everything the AI produces.

A context layer adds five dimensions to every relationship:



*The system knows what happened, when, where, through which channel, where the data came from, and how much to trust it.*

With all five dimensions in place, the context layer can tell you what happened, when it happened, where it applies, how the information got there, and how much you should trust it.

#### WANT TO GO DEEPER ON CONTEXT GRAPHS?

Atlan's comparison piece "*Context Graph vs Knowledge Graph*" is the most accessible explanation: [atlan.com/know/context-graph-vs-knowledge-graph](https://atlan.com/know/context-graph-vs-knowledge-graph). The Zep paper demonstrates how temporal awareness improves AI accuracy in practice: [arxiv.org/abs/2501.13956](https://arxiv.org/abs/2501.13956). Deloitte's whitepaper covers the enterprise case for knowledge-enriched GenAI: [deloitte.com](https://deloitte.com)

**A note on effort and investment.** Building a context layer requires meaningful upfront investment. Knowledge graph construction costs significantly more in compute and human effort than standard RAG approaches: entity extraction needs domain-specific tuning, ontology design requires subject matter expertise, and the resulting graph needs ongoing maintenance as your data evolves. The payoff is that once built, the context layer reduces the ongoing cost of answering complex questions from days of analyst time to seconds of computation, but the path to that payoff requires discipline and realistic timelines, which is something we discuss with clients upfront.

05

## How We Build It

Our approach has three stages, and they are sequential: each one depends on the one before it. There are no shortcuts here, which is something we tell clients upfront because it matters. The quality of your AI outputs is determined by the quality of these foundations, and rushing through them to get to the exciting automation part is how you end up with a very fast system that is confidently wrong.

## Stage 1: Define Meaning, Rules, and Relationships

Before anything gets automated, we work with your teams to define three things.

**Meaning:** we build the three classification structures described earlier: a content taxonomy that maps your assets to strategic objectives, key messages, and themes; a customer data model that defines how HCPs are described by specialty, HCP potential, prescribing behaviour, and attitudinal segment; and a channel data layer that standardises how engagements are classified across rep email, marketing automation, events, digital, and paid media.

**Rules:** the explicit business logic that governs classification. For example, which indications are approved in which markets, how content gets labelled by channel type, or how conflicts between data sources get resolved. These rules are specific to your brands, your markets, and your compliance environment.

**Relationships:** how entities connect across strategy, content, channel, engagement, and customer, forming the knowledge graph described in Section 3. We also define the five context dimensions that enrich every relationship: temporal context (when was a campaign active, when was a brand plan published), spatial context (which market, which region), semantic context (does this asset support a key message), provenance (where this data came from), and confidence (how reliable is this data point). In pharma, these dimensions are what separate an AI that retrieves information from one that understands your commercial reality.

## Stage 2: Automate Tagging and Modelling

Once the rules and relationships are defined, we build the automation that applies them at scale.

For content, this means extracting your key business artefacts (brand plans, campaign briefs, approved marketing content), tagging them against the content taxonomy from Stage 1, and structuring the output so that every asset is connected to its strategic context. For customer and channel data, the focus is on building clean data models that structure how HCPs and touchpoints are described consistently across systems, rather than automated tagging.

The tagging automation uses a combination of rule-based logic (from Stage 1) and LLM-powered classification, with human-in-the-loop validation for edge cases. This hybrid approach balances the precision of explicit rules with the flexibility to handle content that does not fit neatly into predefined categories.

The automation follows the rules: it does not invent its own classifications. Every tagging decision comes with a confidence score and a justification, which means the system is auditable, explainable, and compatible with human-in-the-loop validation. Where the system is uncertain, it flags assets for review rather than guessing. In pharma, an AI that says "I am not sure about this one" is considerably more useful than one that confidently gets it wrong.

### Stage 3: Connect and Serve

With the context layer built and automation running, we connect it to your existing AI tools. In the first instance, this means connecting the data foundation to large language models so that your teams can query it using natural language. A brand director can ask a question in plain English and receive an answer grounded in the full connected context of their data, spanning content, customer, channel, and strategy, without needing to know which systems the data lives in or how to write a query.

### What This Looks Like in Practice

Once a context layer is in place, the way your teams interact with commercial data changes fundamentally.

**Trace impact from plan to outcome.** Every commercial decision can be followed from the brand plan through to the HCP response. You can see what was spent, what it produced, for whom, through which channels, and carrying which messages. This creates the kind of accountability that most pharma commercial teams talk about wanting but have never had the data architecture to deliver.

How are we performing against our strategic imperatives this quarter? CardioMax, UK.

AI Here is your Q1 2026 strategic performance summary for CardioMax UK:

STRATEGIC IMPERATIVE	CONVERSION GOAL	CONVERTED	TARGET	STATUS
First-line positioning Top asset: Cardio Evidence Email v3	Dosing Guide Download	312	250	Above
Safety profile confidence Top asset: AE Profile Detail Aid	Objection resolved	89	100	On
Access & pricing Top asset: Formulary Value Dossier	Formulary submission	18	120	Below
Earlier diagnosis Top asset: Symptom Checker Email	Screening referral	34	200	Below

AI Two imperatives are below target: Access & pricing (18 of 120) and Earlier diagnosis (34 of 200).  
Would you like me to run diagnostics on these? I can identify which content, channels, and audience segments are underperforming and suggest where to focus.

Ask a follow-up question... ↑

**Ask your data a question and get an answer that actually accounts for context.** A brand manager can type "Which HCPs have raised objections in the last month, and what content do I have to handle these?" and the system will return the specific HCPs, the nature of their objections (safety concerns, cost queries, dosing hesitation), the approved assets mapped to each objection type, and any gaps where no content exists. That query spans your CRM, your content library, your engagement data, and your brand strategy. Today it would take an analyst days to pull together. With a context layer, it takes seconds, because the relationships between those systems are already built.

▶ Which HCPs have raised objections in the last month, and what content do I have to handle these?

<b>OBJECTIONS</b> <b>12</b> HCPs flagged Safety concerns, cost queries, dosing hesitation	<b>MATCHED CONTENT</b> <b>4</b> approved assets RWE study, formulary guide, patient case, dosing summary	<b>GAP IDENTIFIED</b> <b>1</b> content gap No objection handling content for cost objection in digital
---	--	--

SOURCES: CRM, Content Library, Engagement Data, Brand Strategy

## Build engagement strategies grounded in what is actually working.

Instead of planning campaigns based on assumptions, your teams can query the context layer to find which audiences to target, which messages are resonating, and where the gaps are. A brand manager asks "I want to drive earlier diagnosis in my therapy area, which HCPs should I be targeting?" and the system builds audience segments based on prescribing behaviour, engagement history, and strategic fit: GPs with low symptom recognition, specialists seeing late-stage presentations, practice nurses with screening opportunities. Each segment comes with a count, ready to activate. The strategy is built on evidence from your own data, connected across every system that contributed to it.

I want to drive earlier diagnosis in my therapy area, which HCPs should I be targeting?

**AI** Based on your brand strategy, I have identified three audience segments. Here are the recommended key messages and available content assets for each:

Audience Segment	HCPs
<b>GPs — Low Symptom Recognition</b>	<b>1,240 HCPs</b>
<b>Specialists — Late-Stage Presentations</b>	<b>380 HCPs</b>
<b>Practice Nurses — Screening Opportunity</b>	<b>510 HCPs</b>

Ask a follow-up question...

## Connect strategy to execution without the manual translation layer. A

context layer serves structured strategic context through a natural language interface. Your teams describe what they need in plain English, and the system builds it from connected data: audience briefs with recommended content and channel strategies, journey configurations grounded in actual engagement patterns, campaign briefs that account for what has worked before.

▶ Build me a segment of high-value cardiologists in the UK who have seen our efficacy data but are still prescribing a competitor.

**AUDIENCE CONDITIONS**

Specialty is **Cardiology**

HCP Potential is **Decile 1-3**

Content Engagement includes **Efficacy messaging** in last **90 days**

Micro-Brick Rx is **Competitor molecule**

Geography is **UK**

**AUDIENCE VOLUME**

**247** HCPs matched of 3,200 target cardiologists (7.7%)

[Export Audience](#)

06

## Getting Started

If you want to explore what a context layer would look like for your organisation, we would be happy to walk you through it: where your data is today, where the gaps are, and what it would take to close them.

If you are interested in understanding your current maturity, take our Marketing Maturity Assessment.

[Book a Discovery Session](#)

07

## Glossary of Key Terms

**RAG (Retrieval-Augmented Generation):** A technique where an AI model searches a database of documents to find relevant information before generating a response. Think of it as giving the AI a reference library to check before answering.

**Knowledge Graph:** A way of representing data as a network of entities (brands, HCPs, content assets, campaigns) connected by explicit, defined

relationships ("targets," "supports," "deployed through"). The connections between entities are built into the structure, which allows the system to follow chains of relationships to answer complex questions.

**Context Layer / Context Graph:** A knowledge graph enriched with five dimensions: temporal awareness (when was this true?), provenance (where did this come from?), confidence scoring (how reliable is this?), spatial context (where does this apply?), and semantic precision (what is the exact nature of this relationship?).

**Vector Similarity Search:** The method most RAG systems use to find documents. It converts text into mathematical representations (vectors) and finds documents whose vectors are closest to the query. Effective for finding relevant text. Limited at understanding relationships, time, or context.

**GraphRAG:** An approach from Microsoft Research that combines knowledge graphs with RAG, using graph structure to find information through entity relationships and community summaries rather than similarity alone.

**Taxonomy:** A structured classification system: a shared dictionary of terms and categories used to describe content, customers, channels, and processes consistently across an organisation.

**Multi-hop Reasoning:** Answering a question by following a chain of relationships rather than finding one document. Each connection is a "hop": key messages > assets > channels > HCPs > engagement > geography.

**MCP (Model Context Protocol):** A standard allowing AI tools to connect to external data sources. The mechanism through which AI applications access the context layer using natural language.

**Hallucination:** When an AI model generates plausible-sounding information that is factually wrong or unsupported. In pharma, particularly dangerous because it can drive decisions based on fabricated information.

**Provenance:** The tracked origin of information. Knowing whether a data point came from your CRM, a marketing automation platform, or a third-party data provider gives teams an audit trail to validate insights and trace decisions back to their source.

**Content Intelligence Layer:** Forge DC's term for the automated infrastructure that continuously classifies, connects, and contextualises commercial data using domain-specific rules and taxonomies.

*Forge DC builds data foundations for pharmaceutical commercial operations. We connect content, customer, channel, and impact data into unified structures that serve both human teams and AI tools. Learn more at [forgedc.com](https://forgedc.com).*

09

## References

1. Hogan, A. et al. (2021). "Knowledge Graphs." ACM Computing Surveys, 54(4).
2. Lewis, P. et al. (2020). "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks." NeurIPS.
3. Edge, D. et al. (2024). "From Local to Global: A Graph RAG Approach to Query-Focused Summarization." Microsoft Research.
4. Rasmussen, P. et al. (2025). "Zep: A Temporal Knowledge Graph Architecture for Agent Memory." arXiv:2501.13956.
5. Serra, A. et al. (2025). "An update on knowledge graphs and their current and potential applications in drug discovery." Expert Opinion on Drug Discovery, 20(5), 599-619.
6. Deloitte (2023). "Responsible Enterprise Decisions with Knowledge-enriched Generative AI."
7. Gartner (2025). Hype Cycle for Artificial Intelligence. Khandabattu, H. and Tamersoy, B., 11 June 2025.
8. McKinsey (2024). "Generative AI in the pharmaceutical industry: Moving from hype to reality."
9. Veeva Systems (2025). Pulse Field Trends Report.
10. DT Consulting (2024). CXQ Study. 6,100 HCPs, 13 countries.

11. Contextual Knowledge Graphs. arXiv:2406.11160 (2024).
12. Han, H. et al. (2025). "How Significant Are the Real Performance Gains? An Unbiased Evaluation Framework for GraphRAG." arXiv:2506.06331.
13. Lettria (2024). Hybrid GraphRAG Benchmark. Reported via AWS ML Blog, December 2024.
14. Paulsen, N. (2026). "Context Is What You Need: The Maximum Effective Context Window for Real World Applications." OAJAIML, 6(1), 4853-4878.
15. Liu, N. et al. (2024). "Lost in the Middle: How Language Models Use Long Contexts." TACL, 12, 157-173.

07

## Ready to explore what a context layer could do for your organisation?

We would be happy to walk you through it: where your data is today, where the gaps are, and what it would take to close them.

[Book a Discovery Session](#)

This whitepaper is provided for informational purposes only and does not constitute professional advice. The statistics, research findings, and third-party sources cited are accurate to the best of our knowledge at the time of publication (March 2026) but may be subject to change. Forge DC makes no guarantees regarding specific outcomes from implementing the approaches described. All trademarks and product names referenced belong to their respective owners.

Forge DC builds data foundations for pharmaceutical commercial operations. We connect content, customer, channel, and impact data into unified structures that serve both human teams and AI tools.

[foragedc.com](https://foragedc.com) · 4th Floor, Silverstream House, 45 Fitzroy St, London W1T 6EB