HEAD OF PRODUCT DESIGN & UX

Joshua Venter

Product design for complex, high-stakes software, where the work is rarely to make the interface prettier, and almost always to change what a user is able to **trust**.

CONTENTS / THREE CASE STUDIES

JOSHVNTR@GMAIL.COM · WWW.JOSHVENTER.COM · LONDON, UK

Galactic Co-Scientist

Head of Product Design & UX · Biorelate · 2025 – present (beta)

Scientists do not struggle to get an answer out of a generic LLM. They struggle to trust it. A fabricated citation looks exactly like a real one until you have done the work to check it, so in drug discovery, the real cost shifted from getting the answer to proving it. I owned the design and product definition of Galactic Co-Scientist end to end: an AI assistant built on a single, uncompromising principle. It may never make a claim the user cannot trace back to a real paper. That principle held from the first designs through to a live enterprise beta with one of the world's largest pharmaceutical companies.

LIVE ENTERPRISE BETA

6-MONTH EVALUATION

GROUNDED, NOT GENERATED

TRUST AS SUCCESS METRIC

FIG. 03

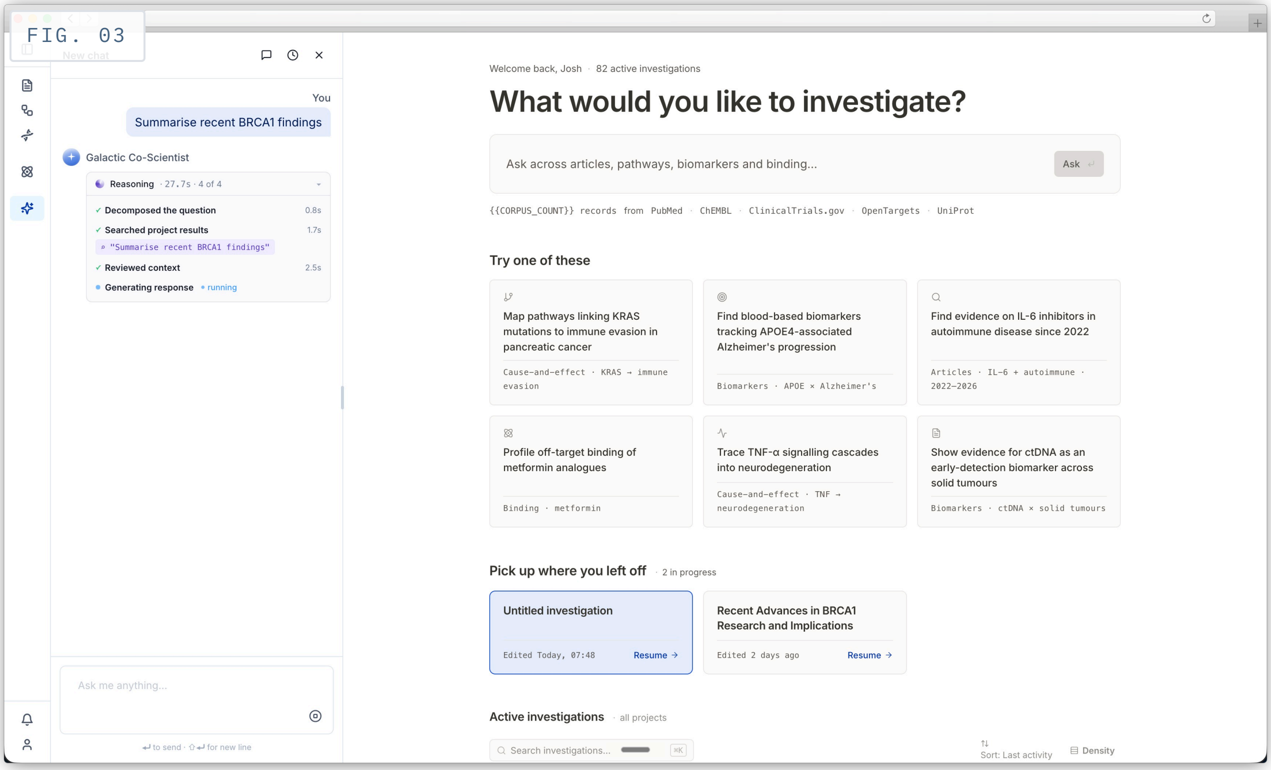

Fig. 03 · The persistent assistant panel with its reasoning trace, sitting alongside the investigation workspace.

CONTEXT & MY ROLE

Galactic Co-Scientist is a persistent assistant panel inside Biorelate's web app. You ask a question in plain English; it works out which biomedical concepts you mean, takes you to the right part of the platform, and answers in a few lines where every claim is backed by a linked scientific paper. When the data is not there, it says so rather than inventing something.

I led it as the entire design function. At Biorelate, that is a role of one: I own product design, UX, front-end specification, the design system and brand across every product line. On this feature I authored the product requirements, drove the naming, held the scope against steady pressure to widen it, and designed the trust model below. I worked alongside the PM, a tech lead on the backend, the data-science team evaluating answer quality, and the professional-services team behind the earlier proof of concept. The decision to build it, and the commercial strategy behind our enterprise pharma accounts, sat with the CSO and CEO; the product and its design were mine to define.

THE REAL PROBLEM: THE VERIFICATION TAX

Before this, researchers bounced between our platform, public tools like PubMed, and a general LLM to summarise, then spent longer checking whether the references were even real. The pain was not speed, it was doubt. A scientist at a global animal-health company put it bluntly: with general LLMs, most of the time the references are not real. A researcher at a national drug-discovery institute said the same in gentler terms: the fear is that the information is not reliable, and they would want to see the papers it was drawing from.

> **The job was not to build a faster answer engine. It was to remove the verification tax, so checking the answer was trivial, not a second job.**

THE CORE BET: GROUNDED, NOT GENERATED

The central decision: the assistant retrieves only relationships extracted from real literature at query time, and is forbidden from answering from the model's own training. Everything downstream follows from that. Every factual statement ends in a linked citation badge to a real paper: title, journal and year on hover. The provenance runs all the way down: from the model's sentence, to the relationship in our knowledge graph, to the exact evidence sentence, to the source publication and its DOI. Confidence is shown as evidence (the number of supporting documents and mentions), not as a number the model invented about its own certainty. A user can always step sideways into the full evidence page and read the underlying papers. The phrase I kept coming back to: this should be AI with data you can touch.

DESIGNING THE CONSTRAINTS

Most of the interesting work here was subtraction, not addition. Deciding what the AI must *not* do mattered more than what it could.

01 No general knowledge for any biological fact: every statement comes from a returned result.	02 Resolve input to a canonical ontology ID before acting: models guess IDs, and a wrong ID means a silently wrong answer.
03 Never show raw internal IDs; never dump every reasoning step by default.	04 When evidence does not load, refuse: say it cannot answer without grounded results, rather than fill the silence.

None of these are visual decisions, but all of them are design decisions: each is about what the user is allowed to trust.

TWO DECISIONS I WILL DEFEND

Where the assistant lives. I chose a persistent side panel available on every page, over three alternatives:

OPTION A

Upgrade the standalone chat

Lived outside the workflow: constant context-switching.

OPTION B

Per-feature contextual chat

Could not navigate across the product; multiplied maintenance.

OPTION C

Dashboard-only home

Reintroduced the same context-switch problem.

OPTION D

SELECTED

Persistent side panel

The only option that let the assistant move the user *through* the app rather than sit beside it.

Questioning my own project. The one I am proudest of. I refused to assume that because the thing was agentic it would automatically be better. We tested the grounded approach against a generic model, and on well-known targets the knowledge graph added little: the general model already knew the textbook biology. An uncomfortable result for the premise of the feature. Rather than bury it, I let it move the bet: the real differentiation was not answering well-known questions, it was surfacing indirect connections and novelty a general model cannot know, and leaning harder on document search grounded in our data. Finding my own premise half-wrong in testing changed the product for the better.

WHAT I GOT WRONG

I proposed simplifying the scientific language in responses, assuming plainer wording would help. Data science overruled me, and they were right: a scientist whose first language is not English still knows exactly what *apoptosis* means, and dumbing the language down would have told our users we did not understand them. I dropped it.

In testing we found the worst possible behaviour: when no articles came back, the system would answer from the model's own weights and append a note saying references would follow: a confident, unsourced claim dressed up as evidence-backed, handed to a scientist at a top-tier pharma client. Precisely the failure the product exists to prevent. A prompt instruction was not a strong enough guarantee, so we moved the guarantee into code: no grounded results, no answer. Trust of this kind cannot live in a suggestion to the model: it has to be enforced at a layer the model cannot override.

WHERE IT STANDS

Co-Scientist is in a live beta with one of the world's largest pharmaceutical companies: a six-month evaluation of Biorelate as their AI co-scientist, with a broader launch to follow. I deliberately argued against usage as the measure of success. Success was defined as whether the client's scientific champions trusted it enough to advocate for it internally, the honest thing to measure for a trust product at this stage. The early signal was exactly that: a lead scientist reported that having the assistant sit beside their work, rather than in a separate tab, had already changed how they moved through the platform.

The grounded approach is not theoretical: a sibling implementation of the same principle is already in daily use at another enterprise account, which is part of why I am confident the bet is right even while the panel is still proving out. There is more to build: an entity-disambiguation flow, a lighter and punchier answer mode, and a re-architected backend as the feature outgrows the structure we reused to hit the enterprise timeline. That reuse was the right call for the deadline and is now the thing most in need of replacing, the honest tension between shipping to a real customer window and building for the long term.

Cause-and-Effect Paths

Head of Product Design & UX · Biorelate · July – November 2023

Researchers were abandoning Biorelate's most-used feature and exporting raw data to spreadsheets instead. I led a redesign that re-encoded how evidence strength was shown in the visualisation, so researchers could judge which relationships were worth investigating at a glance. Engagement rose 42% and people stayed in the tool rather than leaving for Excel.

+42%	+35%	+54%
FEATURE ENGAGEMENT	USER RETENTION	SESSION DEPTH

FIG. 01

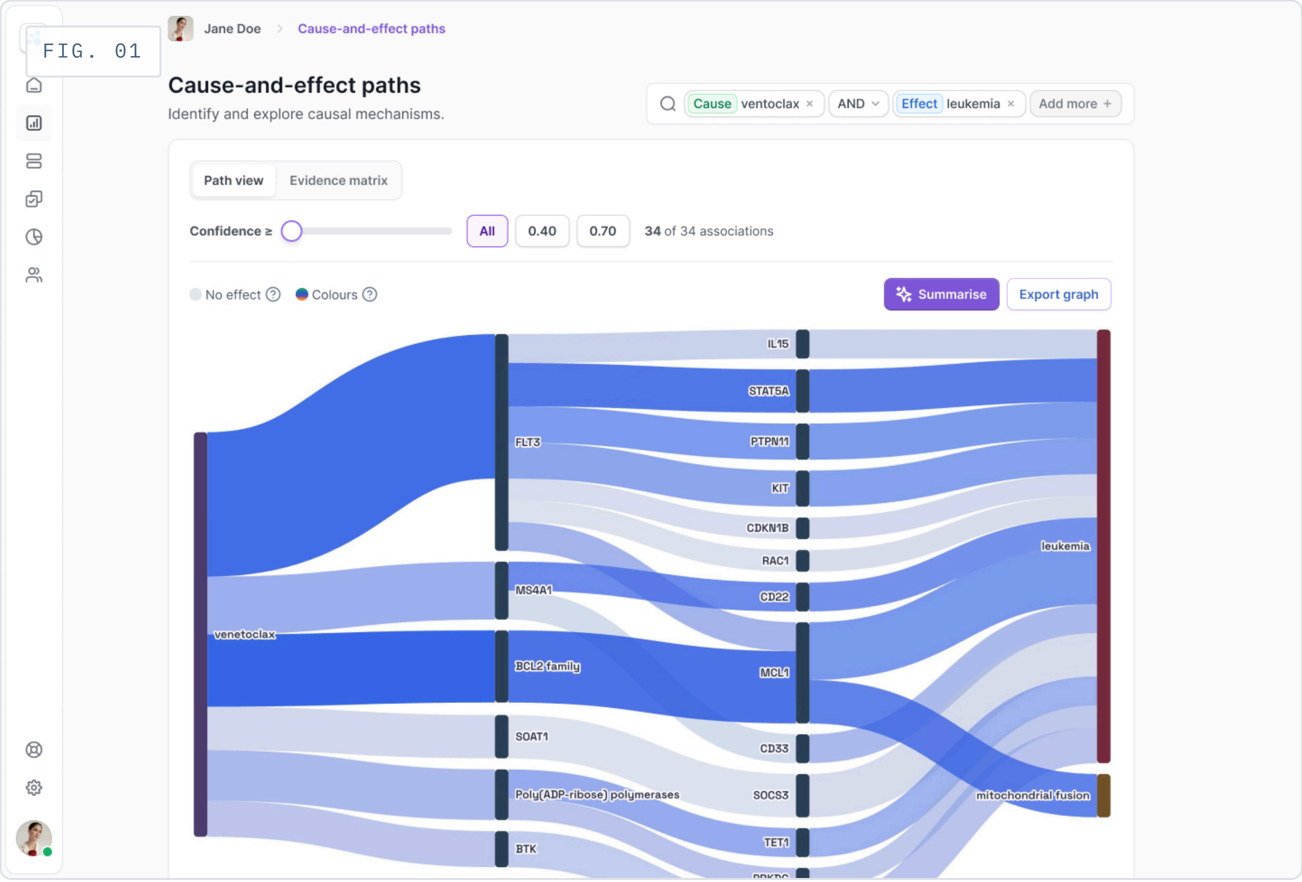

Fig. 01 · Cause-and-Effect Paths: relationship paths colour-coded by confidence, with a linked Graph / Table view.

CONTEXT

Galactic uses NLP and machine learning to pull causal relationships between drugs, genes and diseases out of published biomedical literature. The core feature, a Sankey diagram called Cause-and-Effect Paths, was the most-used part of the platform and also the most complained about. I led the redesign as design lead, working with one engineer and drawing on interviews with 12 researchers.

There was a deadline behind it. Biorelate was about to push its largest-ever data update, which would multiply the volume and complexity of the relationships on screen. The existing design already struggled at current scale; at the new scale it would have been unusable. So this was not a tidy-up. It was a question of whether the feature survived the update at all.

THE PROBLEM I WAS ACTUALLY SOLVING

I started with the 12 interviews and a stack of Pendo session recordings. The thing that changed my approach: researchers were not struggling because the diagram was ugly. They were struggling because they could not tell which relationship paths were worth their time and which were noise. A query could return dozens of connections that all looked equally important, so people gave up on the visualisation, exported everything to a spreadsheet, and filtered by hand.

> **The fix was not visual. It was about the information model: evidence strength had to be encoded into the diagram itself.**

Treat it as a styling job and I would have shipped a prettier version of the same dead end. A researcher needed to prioritise without reading every path.

THE DECISION THAT MATTERED

I worked through four approaches, constrained by a Sankey library that barely supported per-path styling, which ruled out a full rebuild and pushed me toward tight, high-leverage changes.

OPTION A

Brand-new chart type

Most control, but abandons a Sankey researchers already knew, plus far more engineering.

OPTION B

Separate filter panel

Filtering alongside the Sankey, but leaves the diagram itself just as unreadable.

OPTION C — SELECTED

Colour-code paths by confidence

Plus interactive filtering. Improves readability with no new visualisation to learn, and the only option that fit engineering capacity.

OPTION D

Hybrid Sankey + detail table

Simplified overview with a linked table beneath: strongest, but beyond our engineering window.

WHAT TESTING CHANGED

We tested with 8 users, and the most useful result was the one that proved part of my plan wrong. We had kept the numerical confidence scores as tooltips, treating them as a harmless secondary detail. They were not harmless. Users kept asking what a score of 0.73 actually meant. One researcher said he would ignore any path unless someone told him the threshold for "reliable," which defeated the entire point of showing confidence.

So I cut the numbers. We replaced them with High, Medium and Low, tied to the colour-coding, with a short explanation on hover. We did not reduce the data; we changed how it was communicated. That single change did more for usability than the visual redesign around it.

IMPACT

We launched to all users in November 2023 and tracked the result over three months in Pendo. Engagement with Cause-and-Effect Paths rose 42%, with more researchers using it as a primary exploration tool rather than defaulting to raw tables. Retention improved 35%. Session depth rose 54%: researchers followed more relationship paths per session instead of giving up early.

WHAT I WOULD DO DIFFERENTLY

I under-invested in the table view. We treated it as secondary to the Sankey, but the session data showed many researchers preferred the table for detailed analysis and only used the visualisation for initial exploration. By the time I understood that, most of the design and engineering effort had already gone to the graph. A roughly even split would have served them better. The recent data update has since introduced new scale challenges this design will need to handle, so the work is not finished.

Native Competitive Bidding

Senior Product Designer · DeepStream · 2021

Teams running competitive bids were leaving DeepStream to do it in spreadsheets, email chains and standalone tools, then reconciling results back by hand. I led the design of a native auctioning feature that pulled the whole process into the platform. In the first 12 weeks, setup and monitoring time dropped 25% and supplier participation rose 20%. The insight that shaped all of it: the real problem was not speed, it was trust.

−25%	+20%	+30%
BID MANAGEMENT TIME	SUPPLIER PARTICIPATION	USER SATISFACTION

FIG. 02

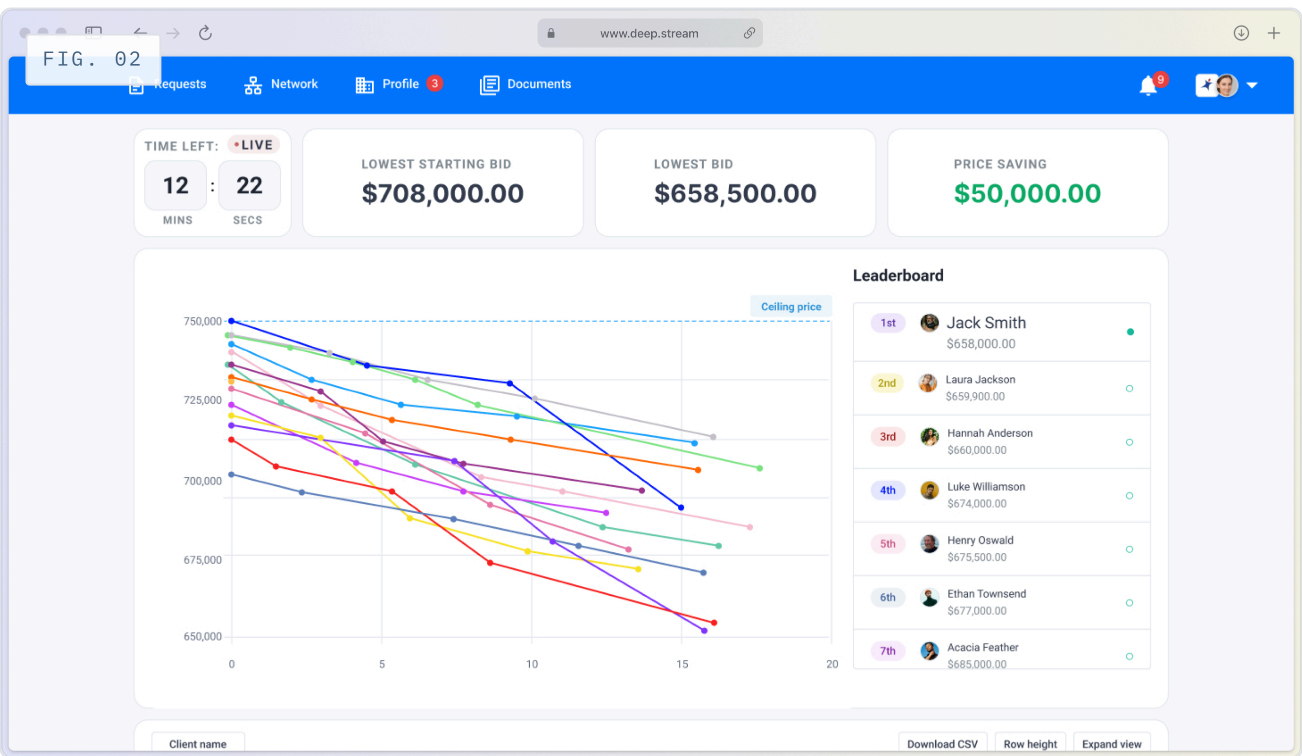

Fig. 02 · Live auction: real-time bid progression against a ceiling price, with a live leaderboard.

CONTEXT

DeepStream is an enterprise procurement platform for sourcing and supplier relationships. Competitive bidding was a core workflow the product could not actually do, so buyers and suppliers improvised with external tools and paid the reconciliation tax afterwards. I owned the design end to end, from research through to launch, as the senior designer on the project. The brief looked like an efficiency job. It was not.

WHAT THE RESEARCH CHANGED

I interviewed 15 procurement professionals and ran a survey that drew 56 responses. The finding that redirected the project: the pain was not mainly about time, it was about trust. Buyers did not trust that every supplier was seeing the same information. Suppliers did not trust that evaluations were fair. A faster tool that ignored that would have shipped the same suspicion with a nicer interface.

> I set transparency as the design goal, not speed. Speed became a by-product of getting transparency right.

THE DECISIONS THAT MATTERED

Auction setup. A competitive bid has many moving parts: specifications, bid rules, timing windows, invitations, evaluation criteria. The spreadsheet workaround was messy but flexible; the risk was rebuilding that mess as one monster form. I structured setup as a guided, step-by-step flow instead. More screens, usually a thing to avoid, but in testing it cut errors sharply. People set up valid auctions on the first attempt.

The live bidding interface. The hardest part, because real-time multi-user bidding is unlike anything else in DeepStream. I explored three approaches:

OPTION A

All-bids dashboard

Everything updating at once: visually busy, hard to scan under pressure.

OPTION B

Feed-style activity log

Timestamped events: reads as a stream, not a comparison.

OPTION C

SELECTED

Live-updating table

Matches the Excel mental model (scan rows, compare values) that procurement teams already had.

The key iteration came from testing. Suppliers did not just want their current standing; they wanted the pattern. Were competitors front-loading aggressive bids or making incremental cuts? That shaped their own strategy. So I added a bid-history timeline alongside the table, showing the full progression rather than only the latest number. Under both decisions sat one constraint I held to: the feature had to feel like a native part of DeepStream, not a module bolted on.

WHAT TESTING CAUGHT

We ran a closed beta with 10 users across both buyer and supplier roles. The problem that surfaced was notifications. Suppliers were not reliably aware when they had been outbid or when a window was about to close, so they missed chances to respond, quietly undermining the whole point of a live auction. We rebuilt the notification hierarchy to separate urgent alerts (being outbid, a deadline approaching) from informational updates (a new bid landing, a status change), and added persistent in-app indicators so nobody depended on email alone. That round was what made us confident to launch.

IMPACT

Measured over the first 12 weeks: bid management time fell 25%, mostly because buyers no longer reconciled data from external tools. Supplier participation rose 20%, with more suppliers bidding per auction now that responding in-platform was lower friction than email. Satisfaction rose 30% across in-app surveys and follow-up interviews. Both sides described the process as more transparent and more trustworthy, which had been the actual goal all along.

WHAT I WOULD DO DIFFERENTLY

I under-scoped the post-auction reporting. We cut it to a basic summary screen under time pressure, and after launch it was clear buyers wanted real analysis of bidder behaviour: who bid when, how prices moved, which suppliers were consistently competitive. The data existed; we just had not designed a way to surface it. For repeat users running auctions regularly, that reporting layer would have done more for the value proposition than almost anything else. It is the first thing I would build next.