

From Evals to Experiments

How to Ship Successful AI Initiatives by Failing Cheaply



DATADOG

**How do you know if your AI
initiatives are working?**

**How *quickly* do you know if
your AI initiatives are working?**

Without a plan, initiatives could take months or years to measure

Bloomberg Subs

The AI Race: Startups to Watch | AI Is Rewiring Finance | AI Unlocks Ancient Secrets | How AI Chatbots Work | Gemini Backlash

Technology

Call-Center Firm Sinks on Klarna Claims AI Is Doing Agents' Jobs

- Teleperformance shares slump on fears over AI impact
- Klarna says AI helps solve customer errands faster than humans

● ● ● ● ● ●

By [Paul Jarvis](#) and [Henry Ren](#)
February 28, 2024 at 10:58 AM UTC
Updated on February 28, 2024 at 2:24 PM UTC

Time to course correct: ~14 months

Bloomberg Subscribe

Industries | Finance

Klarna Rethinks AI Cost-Cutting Plan With Call for Real People

● ● ● ● ● ●



Sebastian Siemiatkowski Source: Klarna Holding AB

By [Charles Daly](#)
May 8, 2025 at 8:00 AM UTC

Copy Li

If we can't fail fast, the cost of failure quickly piles up

Disappointed users and subsequent hit to our reputation

Ceding **competitive ground** by moving too slowly

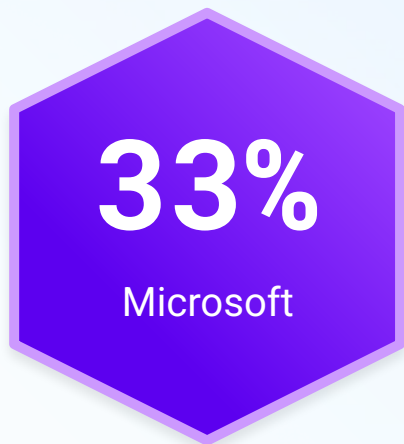
Wasted budget on an initiative that didn't work

The hidden **opportunity cost** of not working on something better

**Over the last three years,
25%
of AI initiatives have delivered
expected ROI**

Source: IBM CEO Study, 32nd ed. (2025), n = 2,000

A/B tests across disciplines have similar success rates, but only take weeks to measure



Sources: Kohavi, Deng, Vermeer. "A/B Testing Intuition Busters" (2022)

Adams and Ou-Yang. "Scaling Experimentation for Machine Learning at Coinbase" (2023)

**How do we bring that power of
A/B testing to AI initiatives?**

From Evals to Experiments



Use Evals to prioritize and tune potential solutions



Use Experiments to prove business impact and make decisions

Use the right tool for the each part of the process



Our ultimate goal is to solve a business problem

Template: We want to build for use case **X**, in order to drive **Y** business outcome

E.g., “We want to build AI-powered **customer support agents** to drive **decreased support costs**.”

This becomes our *hypothesis*, and we run an experiment to close the feedback loop.



To get there, we need to answer smaller questions about “how”

Evals help guide *how* we approach the build.

- What model(s) will do the best job here?
- Can we solve this affordably?
- How should our prompts/datasets look?
- How should we set parameters like temperature?

Evals = pre-production “experiments”

Try out multiple combinations of models/parameters

Use **fixed inputs** in order to compare (e.g. *expected, atypical, adversarial*)

Conduct **error analysis** to define failure modes and **metrics/evaluators**

Look at your **evaluators** plus metrics like **cost, token use, duration**

Eval strategy starts with product strategy

Who are you building for?

What are their **jobs-to-be-done**?

How are they likely to prompt your product?

What about this may **evolve over time**?

Look at traces to identify and categorize errors

LLM Observability

[Traces](#)[Clusters](#)[Configuration](#)[Dashboards](#)

Visualize as

[List](#)[Timeseries](#)[Table](#)

90 Errors

All Monitors

1 ALERT1 OK

Search facets

Showing 14 of 18

CORE

Duration

Min 1000msMax 23.4s

Status

OK

Error 90

ML Application

shopist-chat-v2 90

Version

Service

Env

ML Application Version

EVALUATIONS

Quality

Hide Controls

90 traces found

CONTENT

INPUT

OUTPUT

Do you have any futons or pull out sofa sleepers?

no content

INPUT

OUTPUT

Great, I want to buy the

no content

INPUT

OUTPUT

Do you have any futons or pull out sofa sleepers?

no content

INPUT

OUTPUT

Great, I want to buy the

no content

INPUT

OUTPUT

I want a sofa I want a sofa

no content

INPUT

OUTPUT

For shoe 7 numby exec 7

Hi, I'm sorry but I'm unal

INPUT

OUTPUT

I want a sofa I want a sofa

no content

INPUT

OUTPUT

Great, I want to buy the

no content

INPUT

OUTPUT

For shoe 7 numby exec 7

Hi, I'm sorry but I'm unal

INPUT

OUTPUT

Do you have any futons or pull out sofa sleepers?

no content

INPUT

OUTPUT

For shoe 7 numby exec 7

Hi, I'm sorry but I'm unal

INPUT

OUTPUT

Great, I want to buy the

no content

shopist-chat-v2 > trace_id 66bdd7c0000000bd8b7804081dd2b4

Aug 13, 2024 at 3:26:08.403 pm (47 minutes ago)

Open full page

Duration 1.52sTotal Tokens 0LLM Calls 1Models gpt-4-turbo

INPUT

OUTPUT

Do you have any futons or pull out sofa sleepers?

no content

Assorted Product Inquiries

No Response

cheerful.chihuahua@dog.com

Single prompt session

Trace

Custom Evaluations

Quality

Security & Safety

Errors 3

openai.AuthenticationError: Error code: 401 - {'error': {'message': 'Incorrect API key provided: sk-n47VW*****EfbW. You can find your API key at https://platform.openai.com/account/api-keys.', 'type': 'invalid_request_error', 'param': None, 'code': 'invalid_api_key'}}

Traceback (most recent call last):

File "/usr/local/lib/python3.10/site-packages/ddtrace/llmobs/decorators.py", line 70, in wrapper

return func(*args, **kwargs)

Show all 21 lines

openai.AuthenticationError: Error code: 401 - {'error': {'message': 'Incorrect API key provided: sk-n47VW*****EfbW. You can find your API key at https://platform.openai.com/account/api-keys.', 'type': 'invalid_request_error', 'param': None, 'code': 'invalid_api_key'}}

Traceback (most recent call last):

File "/usr/local/lib/python3.10/site-packages/ddtrace/contrib/openai/patch.py", line 258, in patched_endpoint

resp = func(*args, **kwargs)

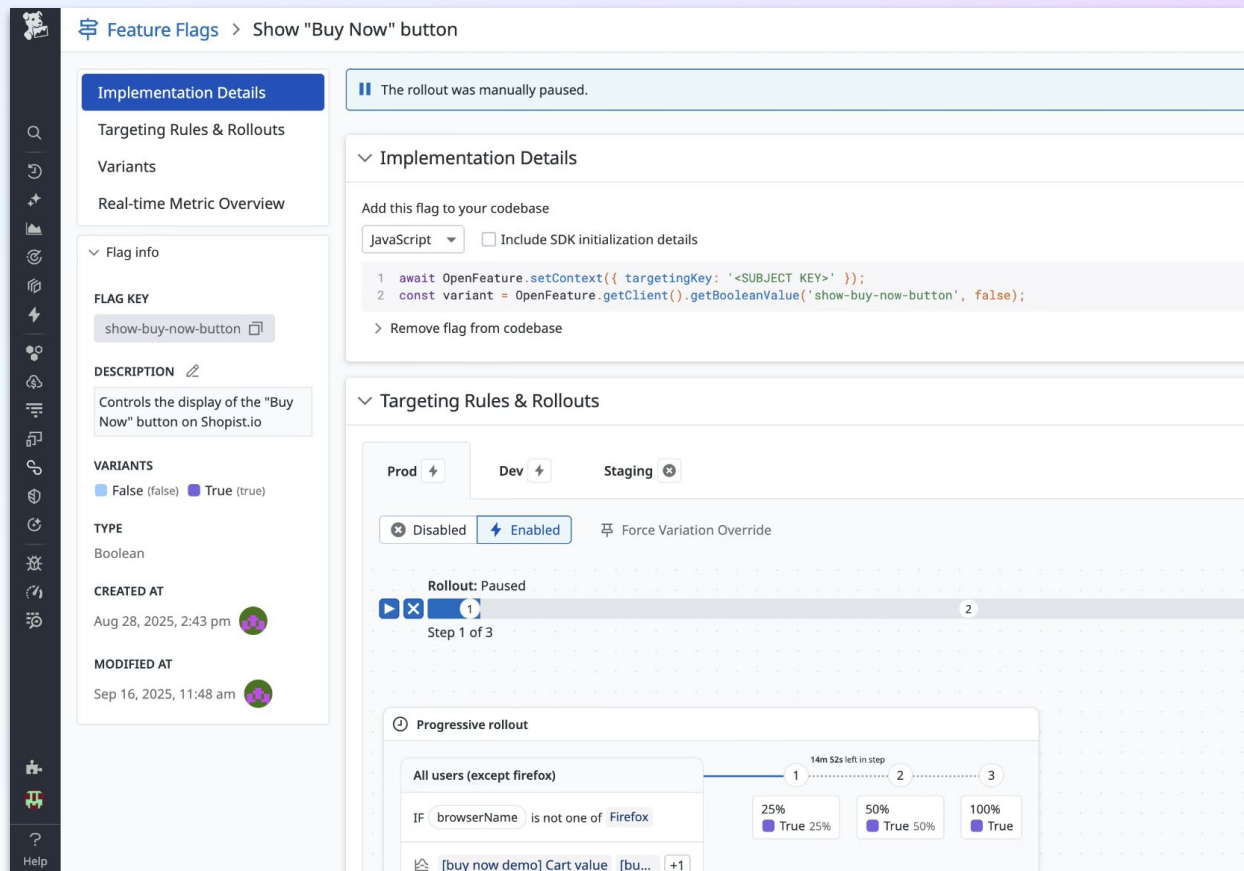
Show all 15 lines

openai.AuthenticationError: Error code: 401 - {'error': {'message': 'Incorrect API key provided: sk-n47VW*****EfbW. You can find your API key at https://platform.openai.com/account/api-keys.', 'type': 'invalid_request_error', 'param': None, 'code': 'invalid_api_key'}}

Example: Datadog Feature Flags

AI Feature #1:
Feature flag implementation

AI Feature #2:
Stale flag cleanup



The screenshot displays the Datadog Feature Flags management interface for a flag named "Show \"Buy Now\" button".

Left Sidebar: Contains navigation icons and a "Help" button at the bottom.

Header: Shows the breadcrumb "Feature Flags > Show \"Buy Now\" button".

Implementation Details Panel:

- Targeting Rules & Rollouts** (selected)
- Variants**
- Real-time Metric Overview**
- Flag info**
 - FLAG KEY:** show-buy-now-button
 - DESCRIPTION:** Controls the display of the "Buy Now" button on Shopist.io
 - VARIANTS:** False (false) [selected], True (true)
 - TYPE:** Boolean
 - CREATED AT:** Aug 28, 2025, 2:43 pm
 - MODIFIED AT:** Sep 16, 2025, 11:48 am

Main Content Area:

- Notification:** "The rollout was manually paused."
- Implementation Details:**
 - Instruction: "Add this flag to your codebase"
 - Language: JavaScript
 - Code snippet:

```
1 await OpenFeature.setContext({ targetingKey: '<SUBJECT KEY>' });
2 const variant = OpenFeature.getClient().getBooleanValue('show-buy-now-button', false);
```
 - Action: "Remove flag from codebase"
- Targeting Rules & Rollouts:**
 - Environments: Prod, Dev, Staging
 - Status: Disabled, Enabled (selected), Force Variation Override
 - Rollout: Paused** (Step 1 of 3)
 - Progressive rollout:**
 - Target: All users (except firefox)
 - Condition: IF browserName is not one of Firefox
 - Progress: 14m 52s left in step
 - Steps: 1 (25% True), 2 (50% True), 3 (100% True)

Use errors to craft evaluators

(*not real examples)

AI Feature Flag Creation

Problem: Model is incorrectly implementing flag

Idea: Did the model read our documentation?

Problem: Model is writing really inefficient code or hallucinating

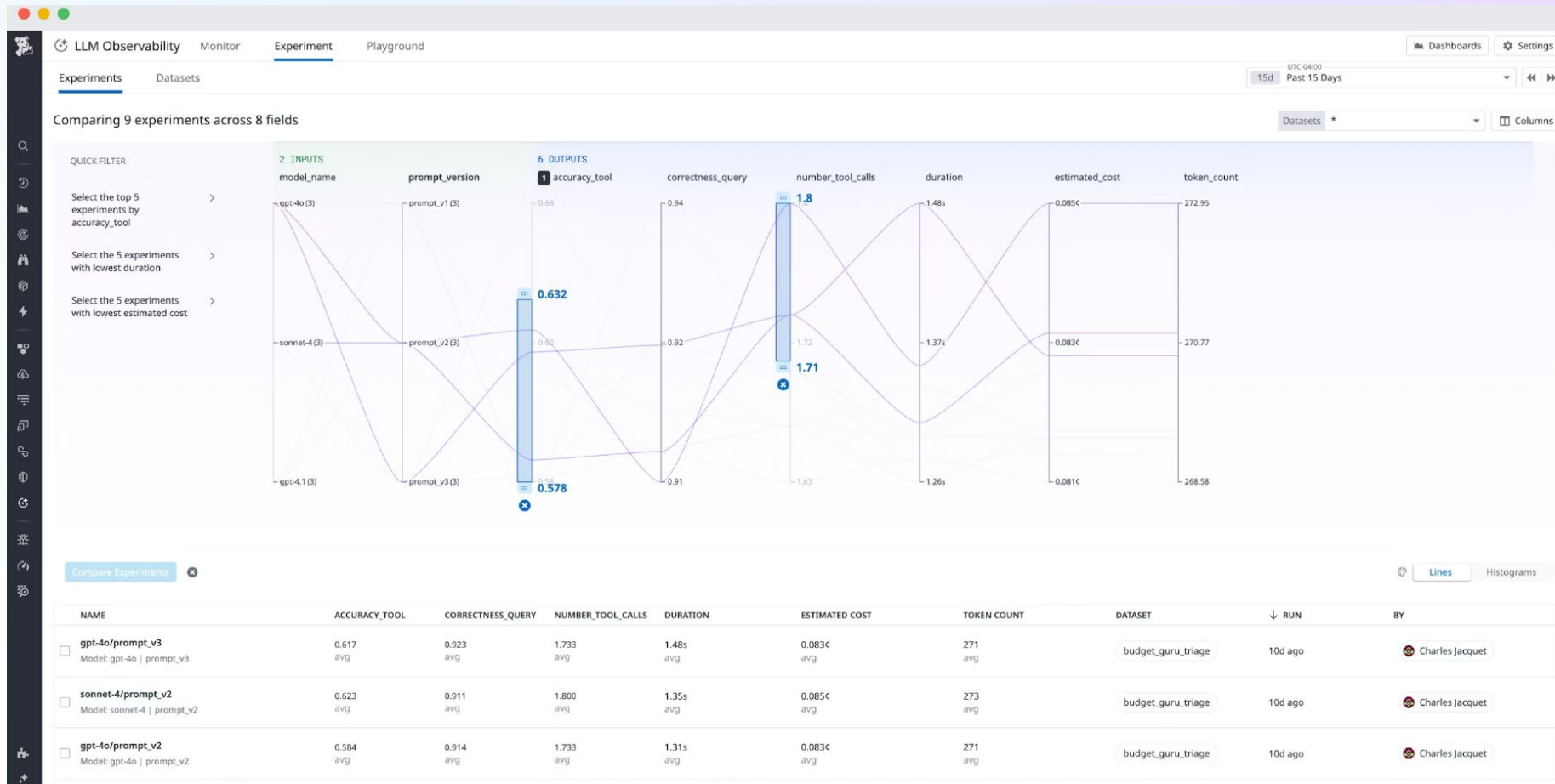
Idea: What was the size of the code change generated?

AI Stale Flag Cleanup

Problem: Model does a partial job - correctly removes some code, but leaves some behind

Idea: “LLM as judge” - ask a second LLM to check the first LLM’s work

Use evals to test prompt or model changes



Why aren't pre-prod evals alone sufficient?



Limited sample size
may not catch all
real cases



Evaluating text output
is often **subjective or**
qualitative



Pre-prod metrics often
fail to correlate to
business metrics

Experiments bring AI to the real world

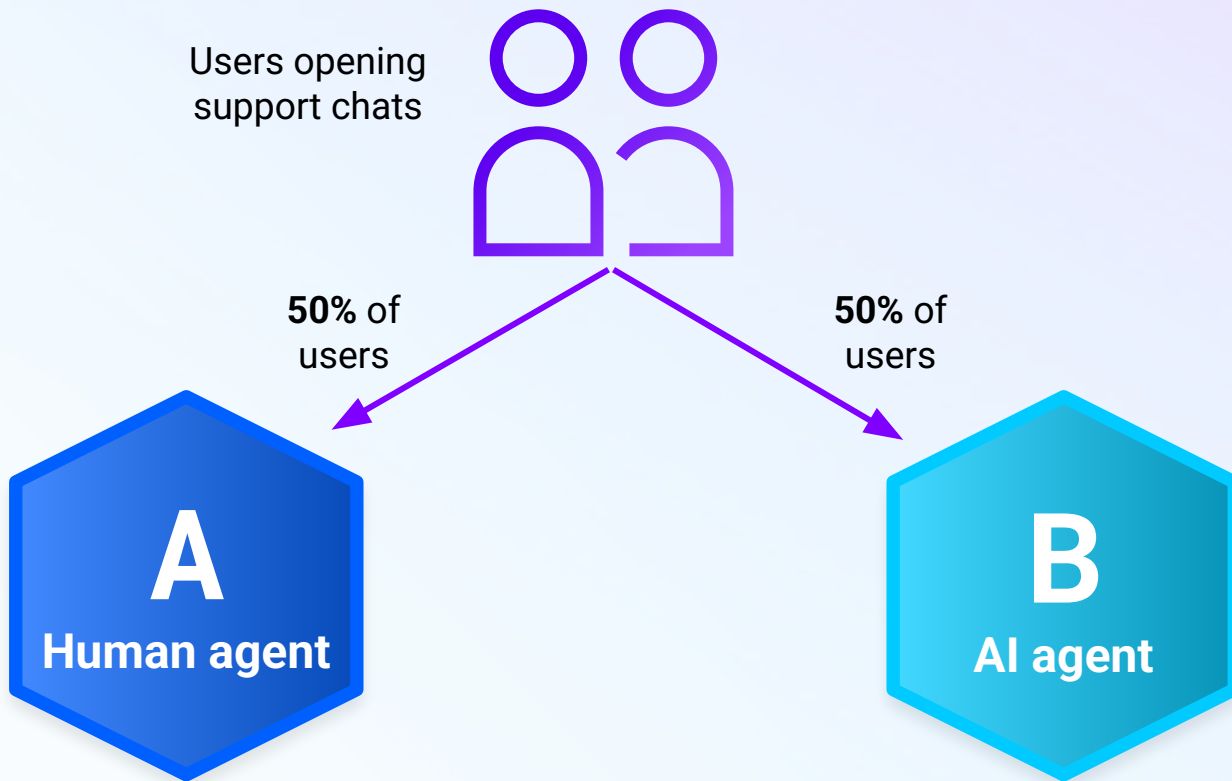
Test our solutions in **real-world environments, with real users**

Randomized, controlled trials **prove what works** (*causality*)

Tie the AI product to **business metrics like revenue or cost**

Experimenting at scale gives insights in **weeks, not quarters**

Example: Testing an AI support agent



The building blocks of an experiment plan



Hypothesis



Audience



Metrics



Statistical Analysis
Plan

Three Types of Experiment Metrics

Primary Metric(s)

Business outcome

- Examples
 - Revenue
 - Completed Transactions
 - Profit Margins
 - Return/Refund rate
 - Value-driving usage (e.g. # of nights booked for a hotel)

Guardrail Metrics

Avoid negative second-order effects

- Examples
 - Repeat purchase revenue
 - Customer LTV
 - Average Order Volume
 - Support Costs
 - Error rates

Storytelling Metrics

Fuel further hypotheses and future exploration

- Examples
 - Time to ticket resolution
 - Customer satisfaction
 - Latency / performance
 - Time on site

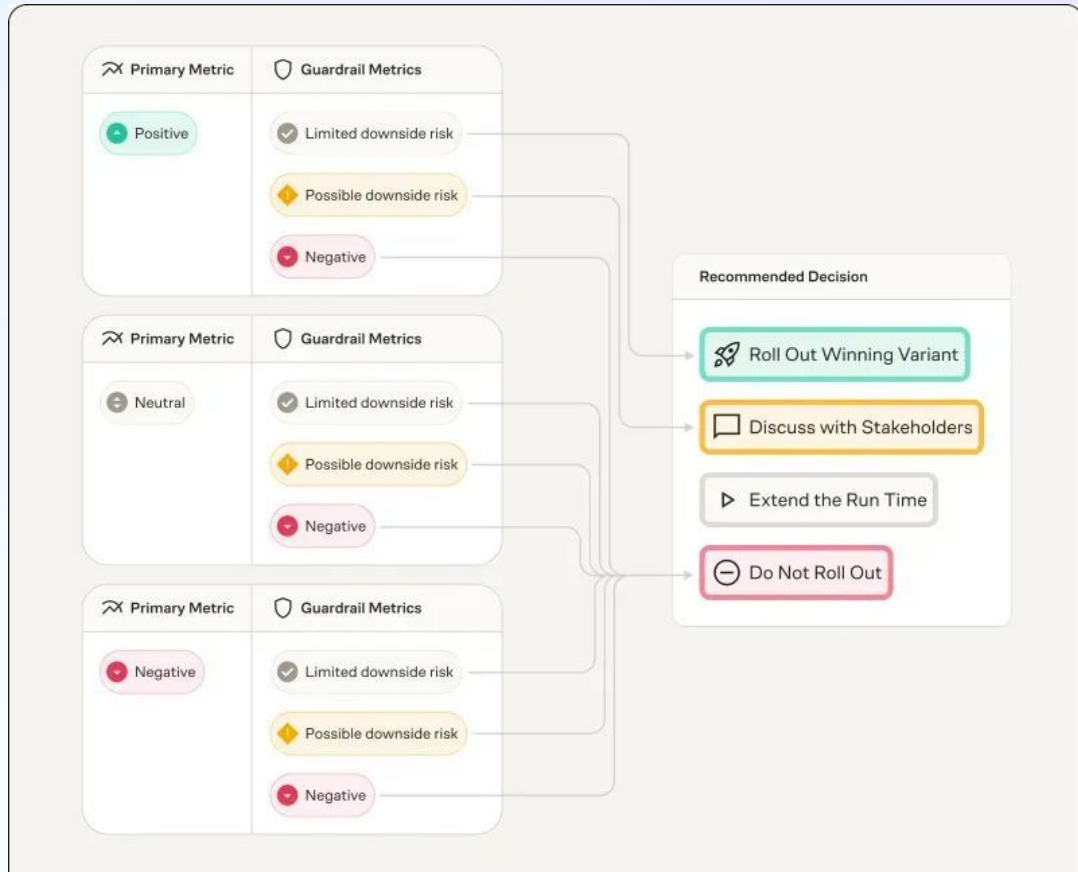


Transactional

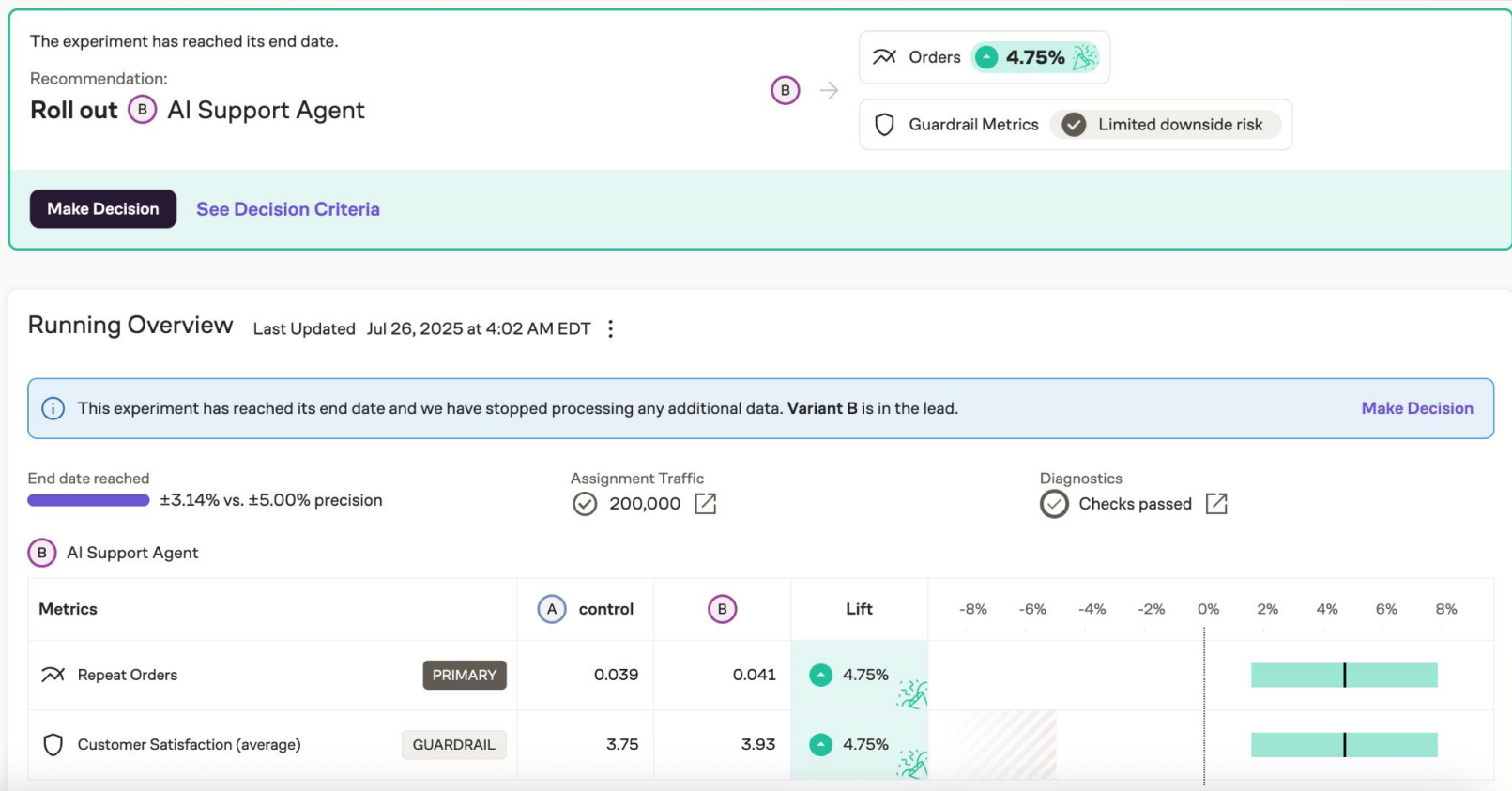


Event Stream

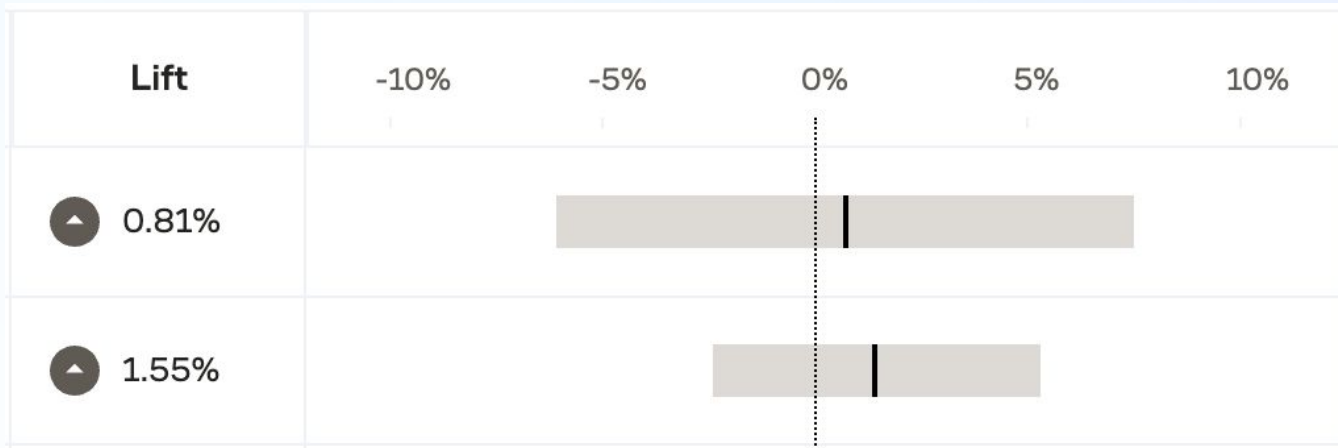
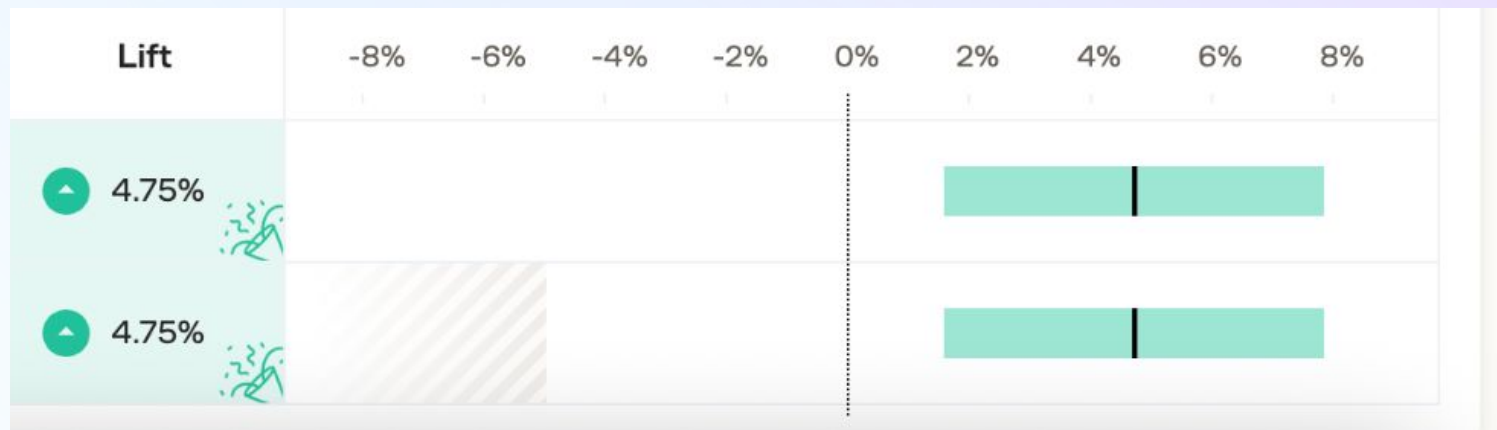
One more step: make decisions in advance



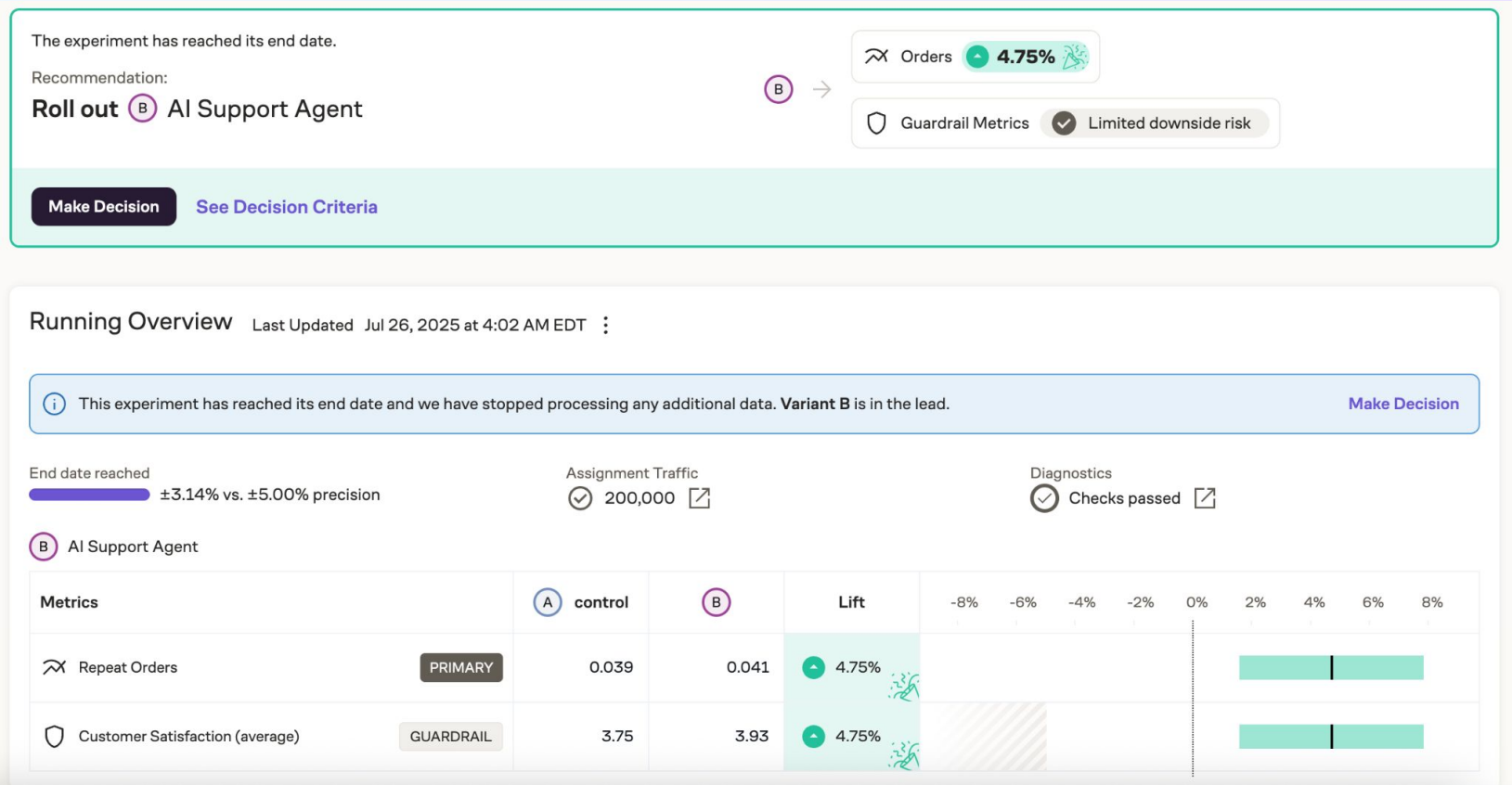
Example experiment results



Stats hack: look at confidence intervals



Output: Confident business decisions



Combine Evals + Experiments

Don't fall prey to long feedback loops

Define success metrics as business outcomes

Run A/B tests to go from idea to proof



Thank you!

Ryan Lucht

ryan.lucht@datadoghq.com

linkedin.com/in/ryanlucht