

Who gets to fly first class?

Value per token in the AI factory

Agentic SDLC @ Red Hat

Infrastructure software
provider:

Linux, Kubernetes,
PyTorch, vLLM

> **5k** engineers using
AI-assisted
development

> **500** engineers
using factory model

Leading public
adoption of AI for
product

Hi, I'm Jeremy.

17+ years

At Red Hat, from entry level to Distinguished Engineer.

Open source

Kubernetes
performance and scale.
PyTorch Foundation
Governing Board.

Now

Studying the
economics of
AI-written code: cost,
governance, security,
sovereignty.

Co-author, *When Agents Decide* (IT Revolution Press).

AI-driven codegen is not economically viable (without discipline)

Uber: entire 2026 AI coding budget spent in four months.

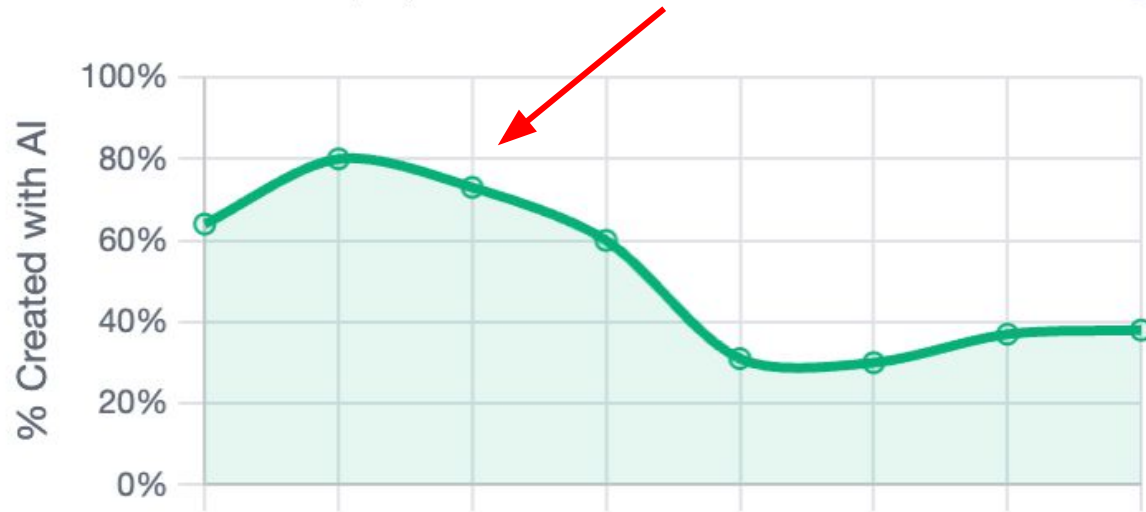
Microsoft: one employee, \$28,000 in 28 days. Median reported: about \$300.

Red Hat: token budgets per engineer + centralized AI-gateway + custom factory

WSJ, Oct. 4: only 11% of 396 companies forecast AI spend within 10%.

The bull-whip effect

Created with AI (%)





Matt Terry (Thrive Market) Today at 12:36 PM

Positive constraints can encourage creativity and decision making ... Time constraints worked for Dylan ... What are some other constraints y'all are employing?

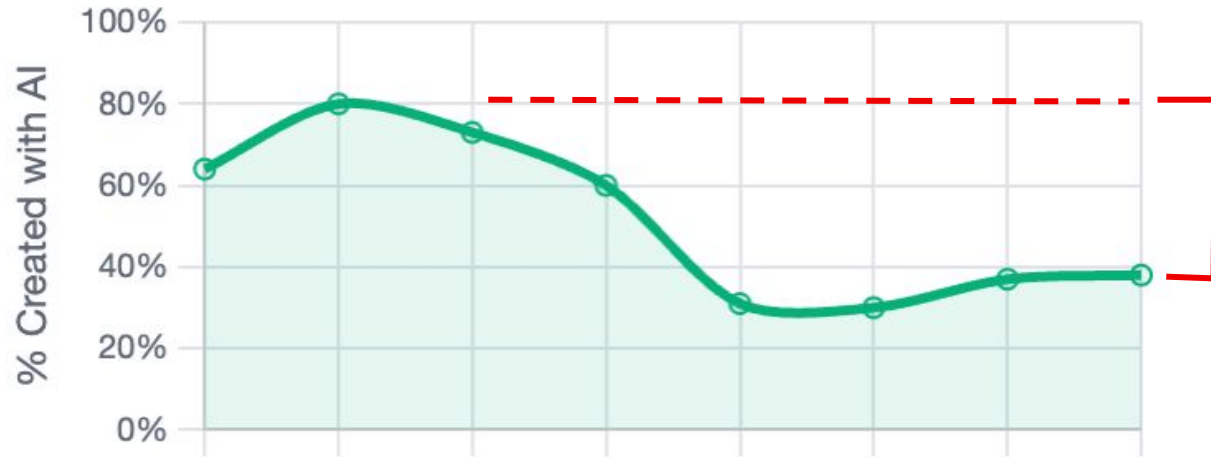


2



The bull-whip effect

Created with AI (%)



Discipline
goes here

5 economic strategies for overcoming the bull-whip

1

Enterprise
Assistant

2

Agentic
Security

3

Flying first
class

4

Who does
this work?

5

Help we can
offer

Value per token, defined.

Value per token = business value delivered ÷ tokens spent.

VALUE

What the job is worth.

Deal size, urgency, blast radius. Read from the CRM and the issue tracker.

DECISION 1

÷

TOKENS

What it costs to pass.

Tokens spent to clear the eval bar. Retries and failed runs count.

DECISION 2

=

RESULT

Value per token.

One number per job. Higher flies first. Rank the value. Route the tokens.

ONE NUMBER

Airline view: priority sets the boarding group. Model tier sets the cabin.

Enterprise Assistant

What does the next gen knowledge-work look like?
What do those people need?

THE PROBLEM

Re-envision enterprise-level data connections, governance and collaboration.

WHAT WE TRIED

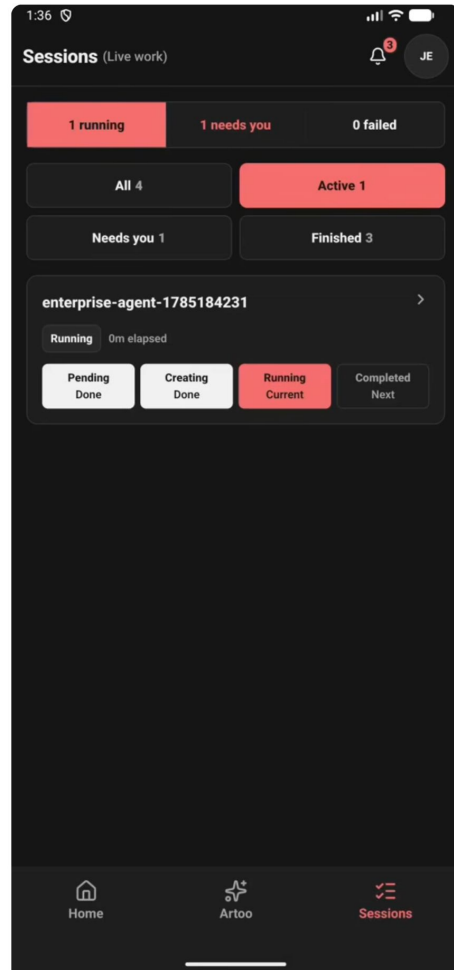
Retrofit onto existing systems.
Adopt latest vendor features.

WHY IT MATTERS

Existing systems can't support what we want.

STILL UNPROVEN

Affordability. Open-model capabilities.



Memory buys efficiency

Sonnet solved 26 to 27 of 42 tasks with memory, 27 with an empty bank.

27%

cheaper

Frozen run alone: 30%

27%

fewer tokens

Frozen run alone: 33%

16%

faster

Frozen run alone: 18%

HAIKU

Memory solved 3 more tasks (8 to 11 of 42). Cost per solve fell 43%. It did not close the gap to Sonnet (27 of 42).

Warm + frozen runs vs empty bank. Exploratory: 42 tasks, 3 attempts each.

Agentic Security

The Platform's role: IBM Lightwell

The problem

Exploit speed
Endpoint security
API keys / Auth
IP escapes
Context
poisoning

Why it matters

Suddenly existing
release speed,
guardrails and
policies became
ineffective

What we tried

First wide-scale
use-case for
internal software
factory

Still unproven

Cost controls,
authentication
standard - ODIS,
AAuth and how
upstream
communities
evolve

When Agents Decide, part 2

OpenClaw Enterprise

An open control plane for persistent agents across users and teams. Enterprise assistants at scale: the “second brain.” Red Hat contributes and will run it internally.

Open Agent Safety Platform

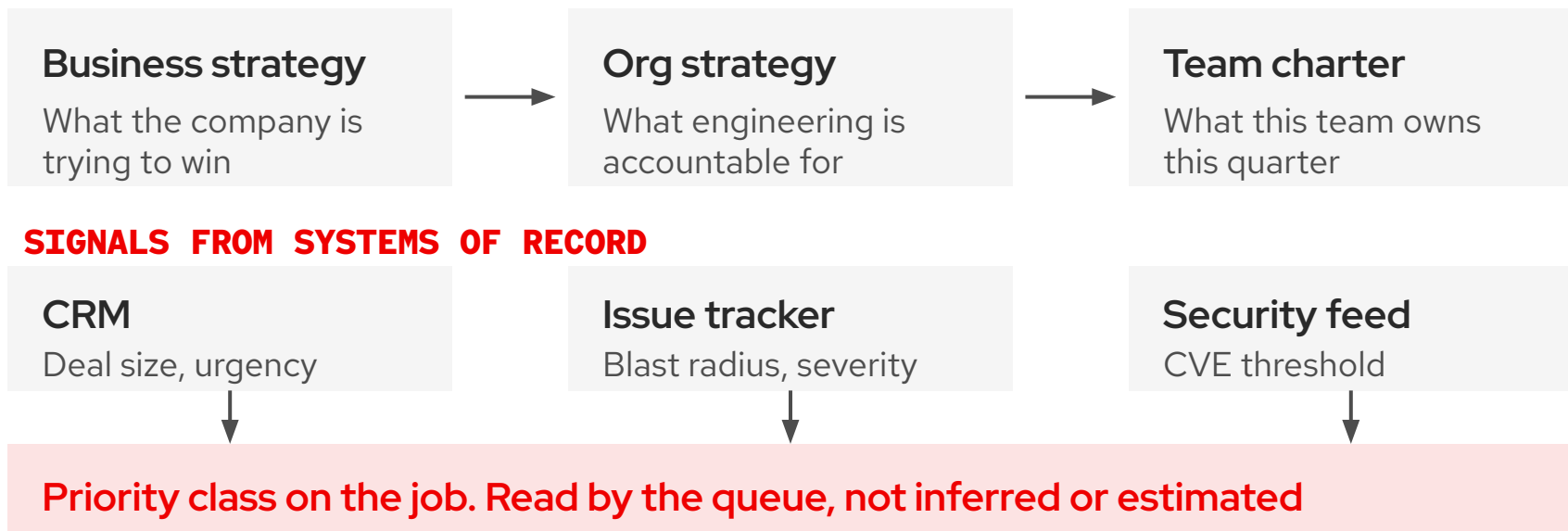
Launched with NVIDIA in September. OpenShell sandboxes agents with policy. Sentry extends it into hardware, watching from a BlueField-4 DPU outside the agent.

This talk

The next step: controlled decisions at scale. DiffusionGemma, a decision model, decides live on OpenShift with a custom vLLM build. A prototype, not production.

"Output to Outcome" → codified

Where a priority number is supposed to come from



Which job flies first? Which model flies it?

Decision 1: What runs first

- Priority comes from CRM and issue tracker.
- Deal size, urgency, blast radius, dynamically ranked

Decision 2: What runs it

- Routed to the cheapest model that clears the eval bar.
- **Don't pay for the same tokens twice**
- If it passes, it ships, model-independent.

Urgency starts with the requester.

THREE REQUESTS

Everyone asks first.

Requested urgency starts the conversation. Asked priority stays attached to every request.

A Reporting redesign

Product · \$14k · 3 days · due Day 8

Asked P1

Ranked #1

B Customer renewal fix

Sales · \$18k · 2 days · due Day 3

Asked P2

Ranked #2

C Cloud cost reduction

Engineering · \$8k · 1 day · due Day 7

Asked P3

Ranked #3

Illustrative scenario. \$ in thousands. Cap \$60k. Scope \$40k.

A tighter budget changes it.

Start savings sooner while keeping the renewal commitment on time.

RULE 01 | BUDGET

Savings go first.

The cap drops from \$60k to \$45k. Cost reduction goes first. The renewal still lands on Day 3.

Rank: #3 to #1

C Cloud cost reduction

Asked P3

Ranked #1

B Customer renewal fix

Asked P2

Ranked #2

A Reporting redesign

Asked P1

Ranked #3

Illustrative scenario. \$ in thousands. Cap \$45k (was \$60k). Scope \$40k.

Rank picks the job. The eval bar picks the model.

DECISION 2

Token price isn't the whole story.

In 32% of 6,800+ tasks, the lower-priced model cost more. One run: \$14 and failed, vs \$1 and passed.

Source: WSJ, Oct 4

C Cloud cost reduction

Economy

Ranked #1

B Customer renewal fix

First

Ranked #2

A Reporting redesign

Business

Ranked #3

Illustrative scenario. Cabins are examples. Study data via WSJ, Oct. 4, 2026.

Who does this work?

Nobody has the jobs defined yet.

Three “on-the-loop” roles

ROLE 1

Curator

Reconciles accumulated memories against the codebase and business priorities.

ROLE 2

Self-improving agent lifecycle

Owens versions, rollout and governance of agents that change themselves.

ROLE 3

Trajectory Miner

Finds value signals and outliers in agent traces. Performance engineering for agents.

Exceptional trace mining

Find the common failure modes across thousands of agent runs.

THE INPUT

6,901 traces in MLflow,
across experiments. 2,356
of them failed.

THE METHOD

A Jupyter notebook.
DuckDB queries the local
trace corpus. k-means
clusters the failed traces.

THE OUTPUT

A fix recommendation for
each cluster. Each one links
to representative traces in
the MLflow UI.

DEMO VIDEO https://youtu.be/pc_ZZ2m18wY

Help we can offer

Five posts from redhat.com, plus the proof of concept.

- 01** The AI-enabled enterprise: Why we are applying software engineering principles to business operations
- 02** Reimagining the enterprise innovation engine in the agentic era
- 03** The AI code paradox: Moving fast without breaking security
- 04** Securing AI agents requires securing the systems around them
- 05** Building the future: Core concepts of Red Hat's agentic software development life cycle
- POC** Proof of concept: <https://github.com/jeremyeder/first-class-tokens>

Help I'm Looking For

**Do you have a similar
"value" metric?**

How do you iterate?

**You set an eval bar per
task**

Are teams "good" at
building evals yet?

**You shipped agent
identity**

ODIS, AAuth, something
else... Do they hold up in
production?

POC

How does the [first-class-tokens](#) decision model approach work for you?