SEAT GEEK

Platform Engineering Day · KubeCon NA 2026

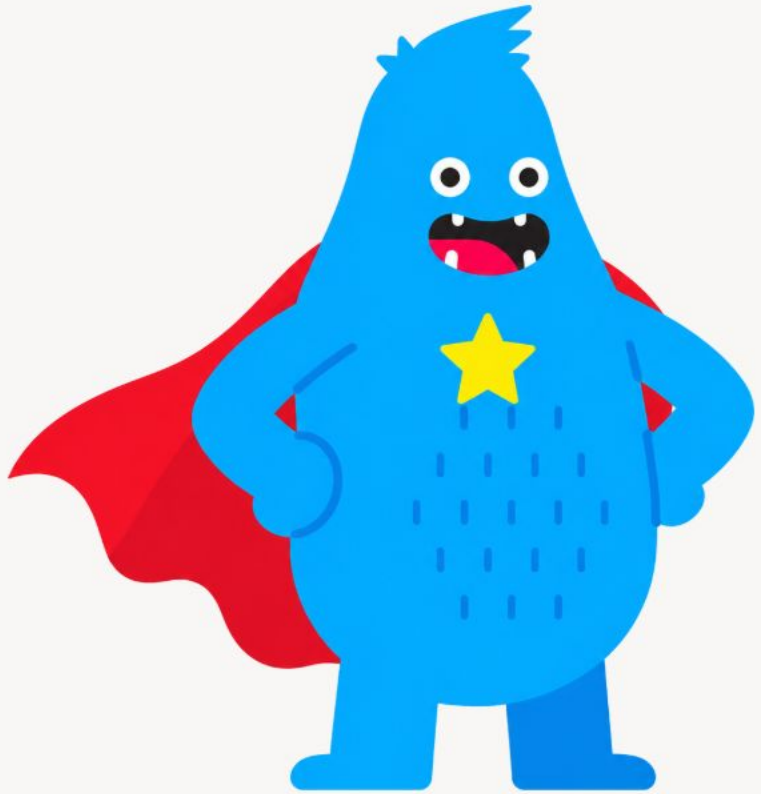
From Clever to Correct

Lessons from rebuilding SeatGeek's platform AI agent

Colin O'Dell · **Lead Software Engineer, SeatGeek**



The vision



Rufus

Lv. 1 · Alpha

Class: **platform engineer + coding agent**

Platform agent — MAX
Who owns this? Why did my deploy fail?

Code generation — MAX
Ask for a change, get a merge request

Code review — MAX
Reviews merge requests in GitLab

Product questions — MAX
How does Deal Score work?

Equipped

Why a combined platform and coding agent

Platform agent
knows the platform

"Why did my deploy fail?" → Diagnosis → Now go fix it yourself

Coding agent
knows the code

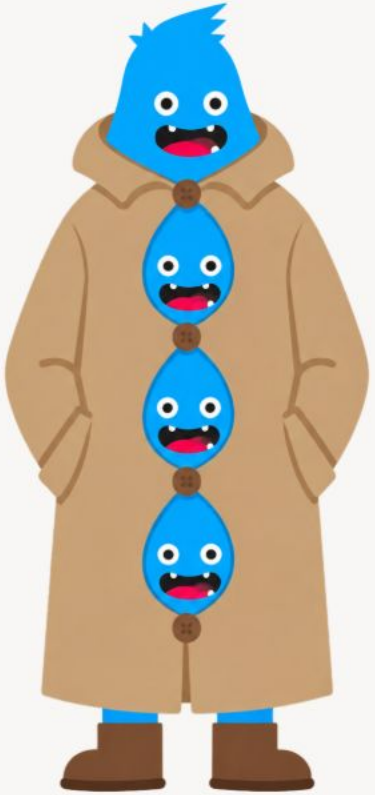
"Fix the failing deploy." → "Which deploy? Paste the logs."

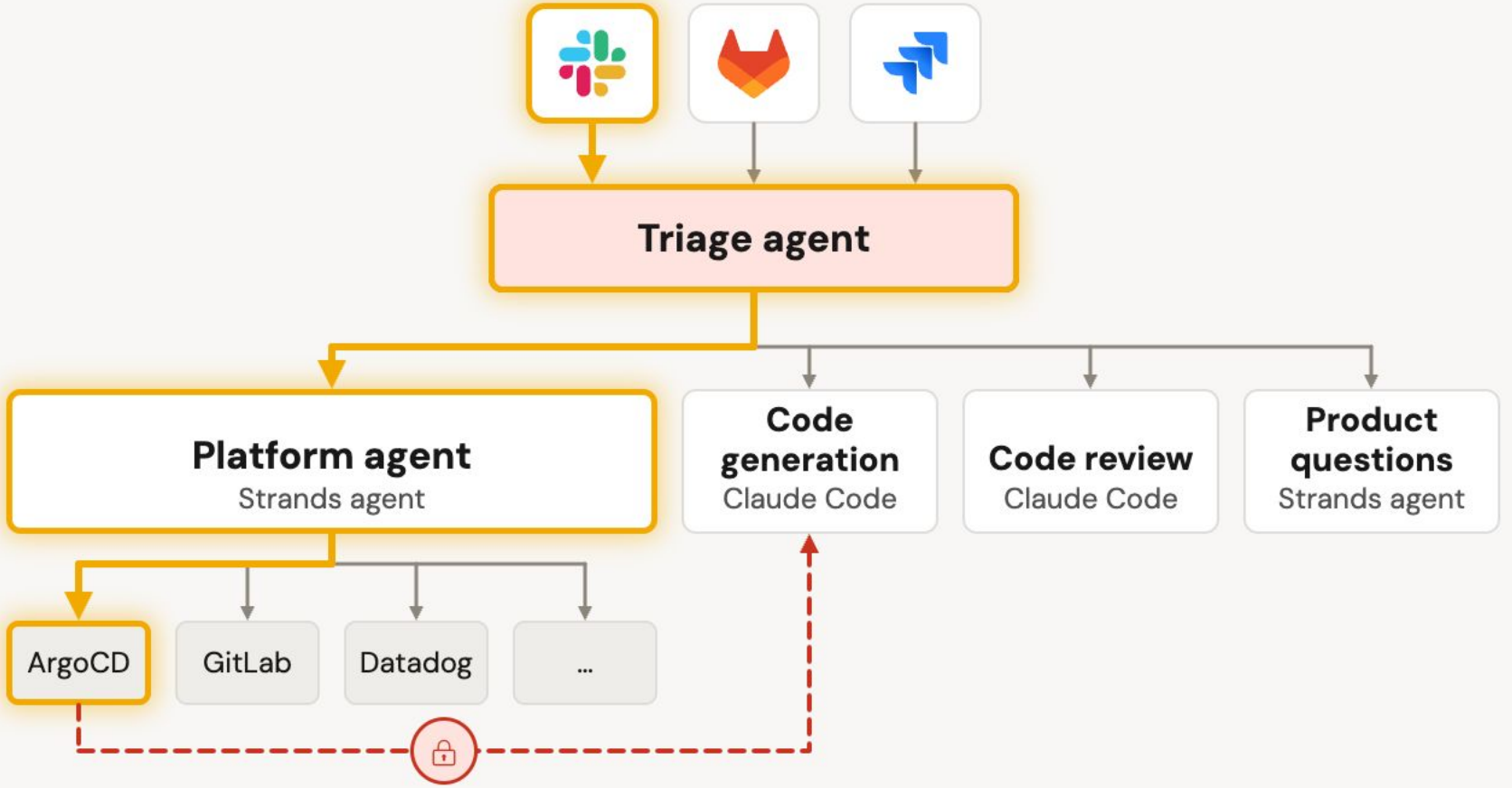
One agent, both
knows the platform and the code, and how they meet

Error → Diagnose → Fix → Merge request → Deploy → Verify on the platform

One conversation, full context, start to finish

The original architecture





Summer 2025

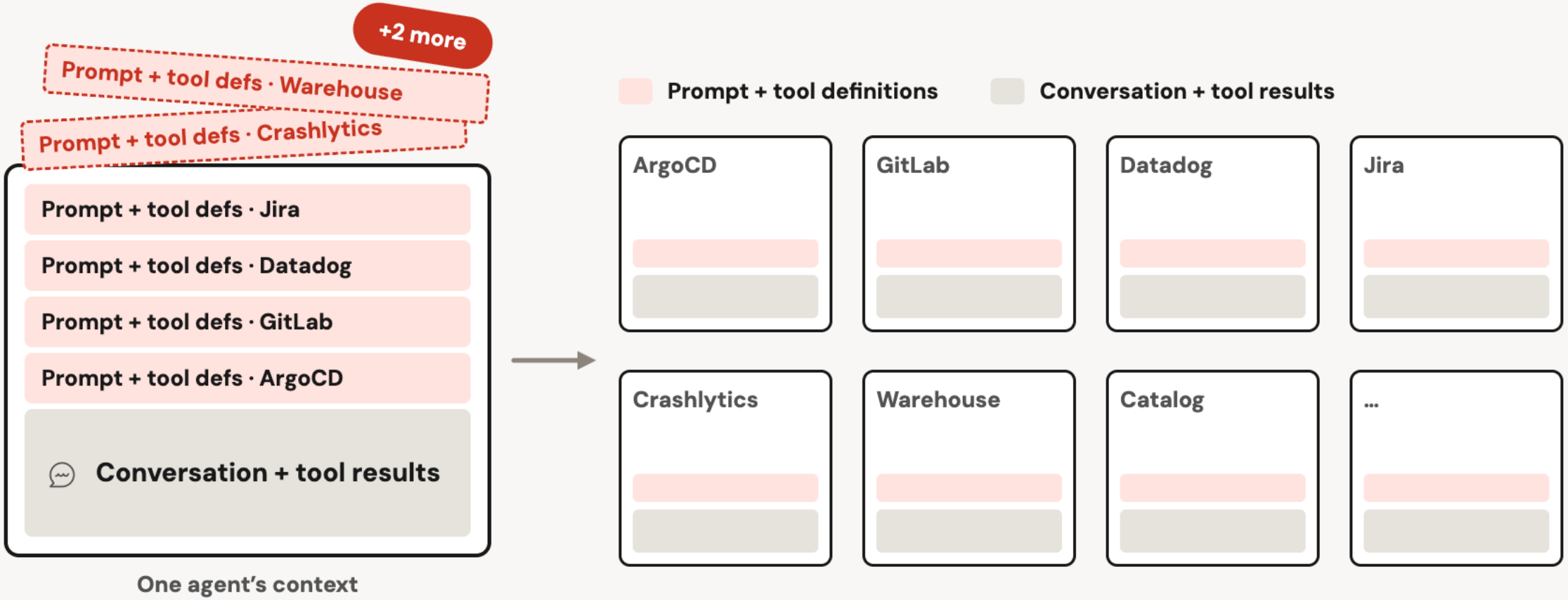
State of the art

- Claude Opus 4
- Long prompts
- MCP tools
- RAG
- Claude Code SDK
- Strands Agents framework
- Context engineering
- Sub-agents
- AGENTS.md standardization

Not yet, or experimental

- Claude Opus 5
- Adaptive thinking
- Auto-compaction
- Kubernetes Agent Sandbox
- SKILL.md standardization
- Skill Creator
- Dynamic tool lookup

Too much for one agent



Lesson 1

Generalist > too many specialists

Rufus · summer 2025

Models
Claude Opus 4
Worked best with RAG and long prompts

+

Harness
Strands · Claude Agent SDK
~~skills~~ ~~dynamic tool lookup~~ ~~compaction~~

=



4 + 8
agents + sub-agents

Sandy · 2026

Models
Newer frontier models
e.g. Claude Opus 5.5; GPT-6.1 Sol

+

Harness
OpenCode
skills dynamic tool lookup compaction

=



1
agent, skills on demand

Lesson 2

Every session gets a sandbox

Before: shared pods



Now: one sandbox per session



Isolated · pause and resume · off the shelf

Lesson 2

Every session gets a sandbox

Sandy's context window

Just the 3 lines that matter
`OOMKilled · memory limit 512Mi`



Sandbox · its own disk and compute

`deploy.log`	`8.4 MB`
`api/`	`cloned repo`

```
$ grep -n OOMKilled deploy.log
1824: … OOMKilled
5310: … OOMKilled
9047: … OOMKilled
```

Isolated: no noisy neighbors

Pauses and resumes

Off the shelf: Daytona

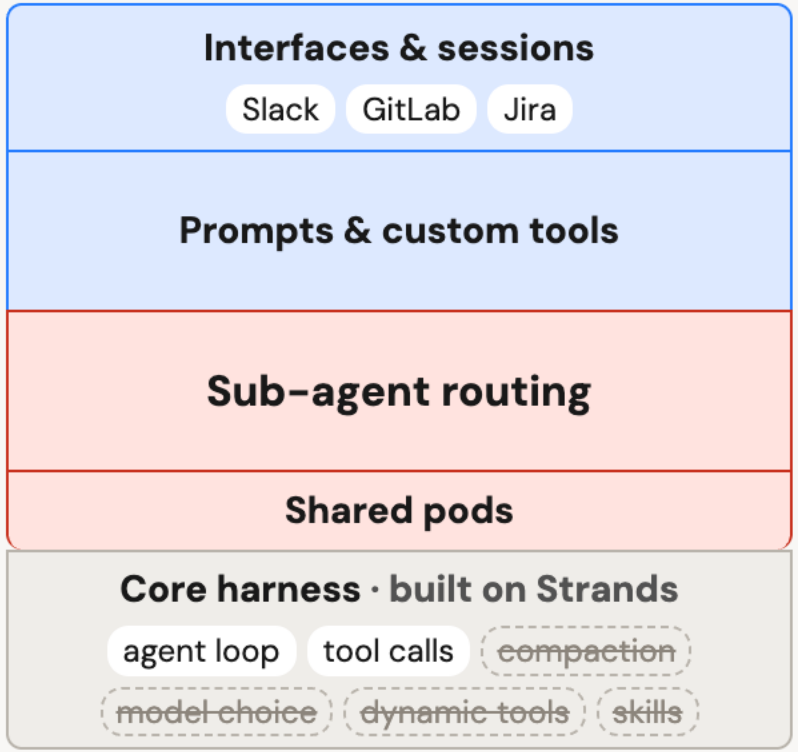
Lesson 3

Build what differentiates, buy what doesn't

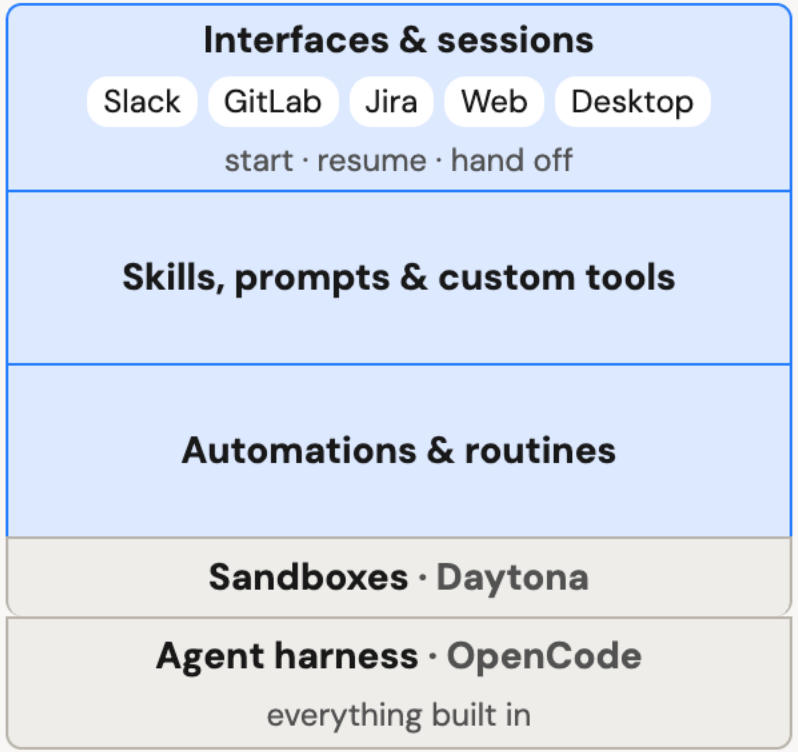
Differentiation
The secret sauce

Commodity (DIY)
Built ourselves

Commodity (Buy)
Existing FOSS or paid solutions



Rufus



Sandy

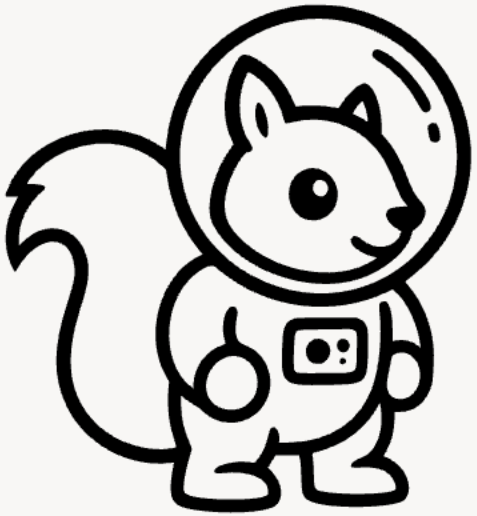
Lesson 4

Close the loop



Lesson 4

Faster loops: let the agent vent



Sandy

Tool definition — simplified

```
name: vent
description: Call this when you're stuck or the user is
frustrated: repeated tool errors, missing access, no
progress. Say what went wrong in a sentence or two.
```

#sandy-vents · **Large Jira .docx attachments can't be read**

fetch_jira_attachment refuses ~2.2 MB docx files. A direct curl to the Jira attachment content URL from the sandbox returns 403. As a result, no tool path can read a large Jira attachment, and the user's contract question was blocked.

Fix with Sandy

Lesson 5

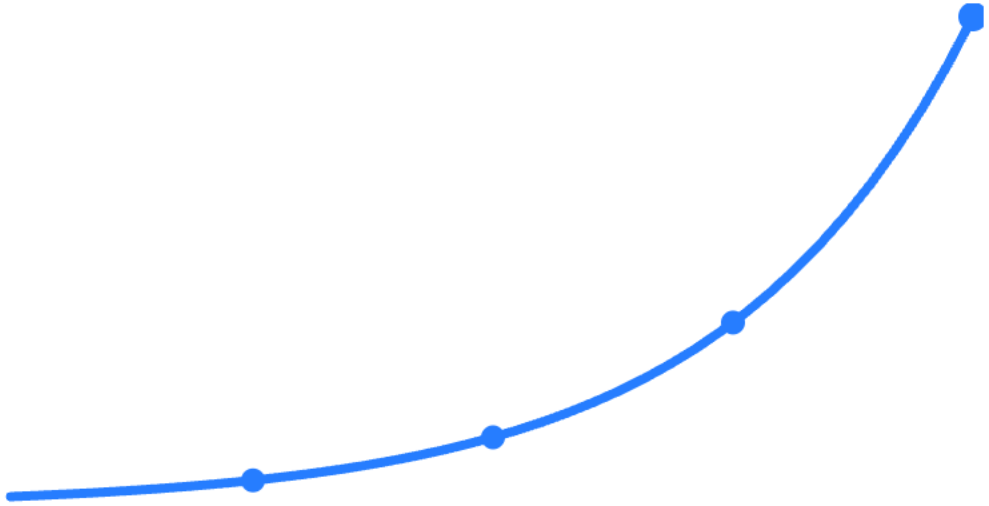
An agent is a product, not a project

Rufus · a project



GA, then maintenance only

Sandy · a product



Keeps adopting what's new

Five lessons

1 One generalist, many skills.

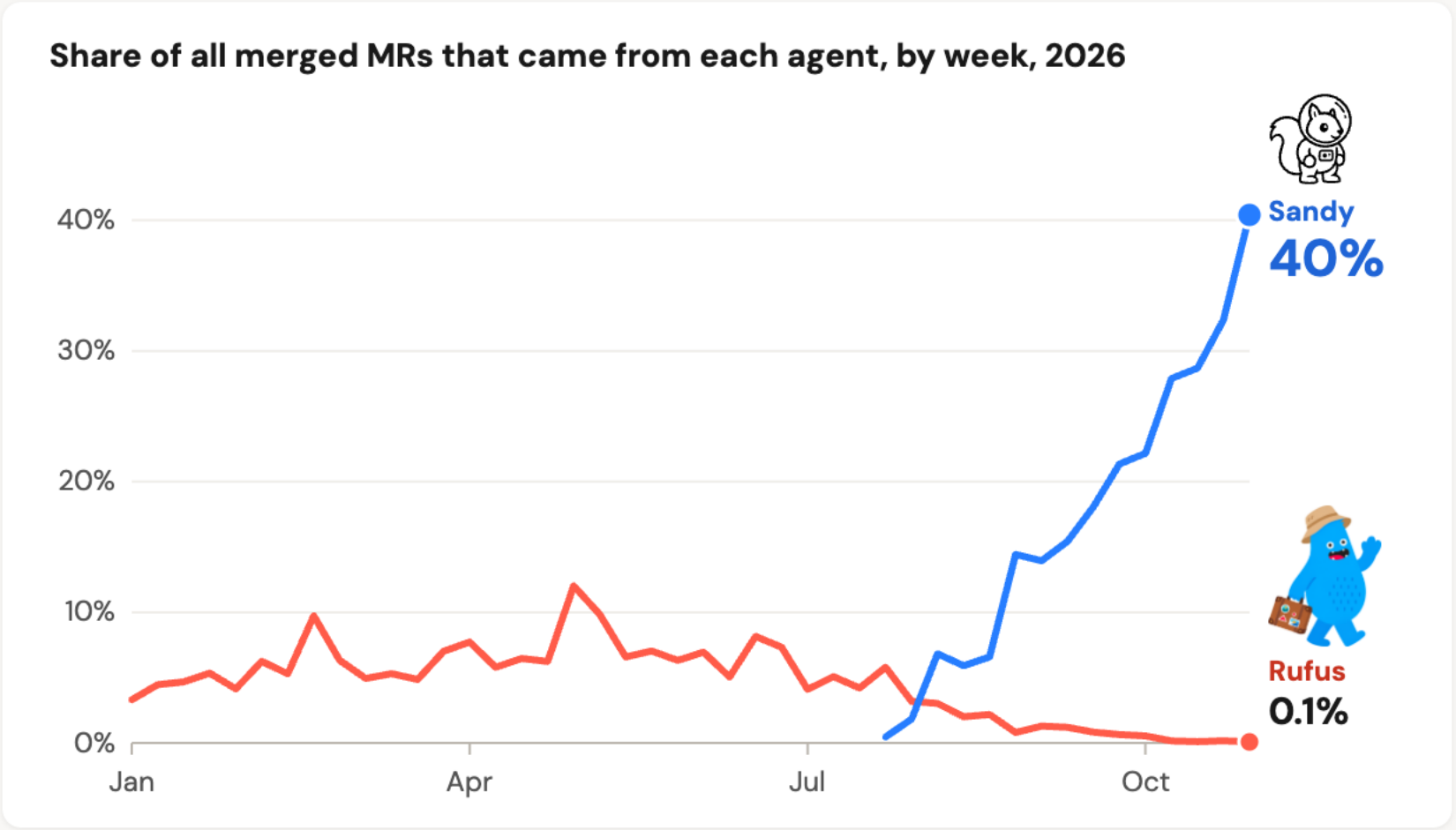
2 Every session gets a sandbox.

3 Build what differentiates, buy what doesn’t.

4 Close the loop.

5 An agent is a product, not a project.

Sandy today



40%

of all merged MRs now come from Sandy

72%

of engineering uses Sandy every week