

Dobby

A Meta-Agent Manager for Research



"The research bottleneck in research isn't intelligence — it's autonomously coordination."

Simon Yu · March 2026

The Problem

As researchers, we have access to powerful tools...

- **Claude Code/Codex** — autonomous coding agent
- **GPU clusters** — EC2, university Slurm clusters, Lambda machines, local machines
- **Multiple projects** — papers, experiments, infrastructure

...but the human becomes the bottleneck.

"You have to remove yourself as the bottleneck. You can't be there to prompt the next thing. You need to take yourself outside. Arrange things such that they're completely autonomous — maximize your token throughput and not be in the loop." — Andrej Karpathy

One researcher, one agent, one machine, one task at a time. The human-in-the-loop is the serialization point.

Dobby: The Meta-Agent Manager

How It Differs

Claude Code / Codex

- One agent, one repo, one machine
- You are always in the loop, context-switching between projects wastes hours

OpenClaw

- One agent, one task, sandboxed
- No distributed execution
- No human-in-the-loop approval

Dobby

- **Multi-agent** across machines, 24/7 monitoring
- **Manager instead of executor**
- **Plan → Approve → Execute** flow
- **Backend-agnostic** (Claude + Codex)
- **Slack/MCP/CLI** from anywhere

Recursive Agent Dispatch

Each agent can call Dobby to create more sub-tasks:

This enables **unlimited depth of delegation**

Overnight Autonomy — Real Example

10:00 PM — Submit 6 tasks across 3 projects via Slack, go to sleep

2:00 AM — Dobby detects vLLM crash on EC2 (OOM) → Diagnoses root cause (CUDA graph allocation)

6:00 AM — Dobby detects pipeline completion → Pivots to next phase, submits 3 sequential training jobs

8:00 AM — Posts morning summary:

Project	Result
Meta-Agent	✅ Research report + experiments (asymmetric transfer confirmed)
User Simulator RL	✅ Evalaution done, 3 models IFEval trained
RL Jobs	✅ 3 slurm jobs submitted and running

Total overnight cost: Claude code usage

Live Dashboard

<https://dobby.chats-lab-gui-agent.uk/dashboard>

- **Agent Network** — hub + spokes with live status
- **Heatmap** — GitHub-style activity across projects
- **Flow View** — VibeLens-style agent execution timeline
- **Usage Tracking** — Claude API utilization over time
- **Project Control** — per-project status with spoke labels

Some interface get insights from VibeLens, developed by Jingheng

Summary

Dobby is the operations layer that coding agents are missing.

	Claude Code	OpenClaw	Dobby
Machines	1	1	N (distributed)
Always-on	✗	✗	✓ VPS hub
Submit from	Terminal	Browser	Slack / MCP / CLI / API
Backends	Claude only	OpenAI only	Claude + Codex
Overnight	✗	✗	✓ Self-healing
Approval flow	✗	✗	✓ Plan → Approve

Claude Code is the engineer. Dobby is the engineering manager.