

INTERVIEW PREP: BORIS CHERNY

Head of Claude Code, Anthropic

Prepared: January 15, 2026

GUEST PROFILE

Boris Cherny is the creator and Head of Claude Code at Anthropic. His journey spans startups, venture capital, and a decade at Meta (Facebook/Instagram) where he rose from mid-level engineer to Principal (IC8) and led code quality for all of Meta's codebases. He authored "Programming TypeScript" (O'Reilly, 2019) and organizes the San Francisco TypeScript Meetup.

PREVIOUS MEDIA APPEARANCES

The Peterman Pod (December 2025) - 1hr 25min

Topics covered:

- Career journey: startups → VC → Facebook → Instagram → Meta code quality → Anthropic
- Why every engineer needs "side quests"
- Why being underleveled is a good thing
- All promotion stories from IC6 to IC8 at Meta
- How to find side projects and build credibility
- Vibe coding limitations and when it works
- Claude Code competition landscape

AI & I Podcast / Every.to (October 2025)

Topics covered:

- Origin story of Claude Code from internal experiment
- How to use subagents effectively
- Favorite slash commands
- Lessons from watching Anthropic engineers use it

AI Engineer World's Fair (June 2025)

Topics covered:

- Evolution of coding UX since the 1950s
- Programming abstractions vs. programming UX parallel tracks
- Philosophy: "We don't know what the right UX is yet"
- The "bitter lesson" applied to agent UX

Anthropic Financial Services Webinar

- Claude Code applications in regulated industries

QUESTIONS TO AVOID (ALREADY ANSWERED)

These have been thoroughly covered elsewhere:

1. **"How did you create Claude Code?"** → Covered extensively in Every.to and Peterman Pod
2. **"Walk me through your career at Meta"** → Full 1hr+ coverage on Peterman Pod
3. **"What's your workflow with Claude Code?"** → Twitter thread + VentureBeat article went viral
4. **"How do you use CLAUDE.md files?"** → Multiple interviews covered this
5. **"Opus vs Sonnet - which do you prefer?"** → He's stated publicly: Opus 4.5 with thinking, always
6. **"What do you think of Cursor/Copilot?"** → Answered in Latent Space: 'awesome products' but different goals
7. **"Is vibe coding good or bad?"** → Nuanced answer already public: good for prototypes, bad for maintainable code

HIS STRONGEST OPINIONS (POTENTIAL TENSION POINTS)

1. "The model is moving faster than the product"

He believes AI coding capabilities are on an exponential curve, but product/UX is "struggling to keep up." This creates tension with companies building polished IDE experiences. **PUSH: Does this mean Cursor/Windsurf are building the wrong thing?**

2. "Claude Code is a Unix utility, not a product"

Intentionally bare-bones and terminal-based to give raw model access. This is contrarian in an industry obsessed with beautiful UIs. **PUSH: Is this philosophy or just rationalization of limited resources?**

3. "Agents are workforce, not assistants"

Runs 10-15 concurrent Claude sessions. Treats AI as parallel workers with separate contexts, not one smart helper. **PUSH: Doesn't this require a level of expertise most developers don't have?**

4. "New grads use AI better than senior engineers"

Claims newer engineers without "legacy memories" of model limitations use AI most effectively. **PUSH: Is this actually true, or survivorship bias from Anthropic's hiring?**

5. "I throw away 10-20% of Claude sessions"

Surprisingly candid about failure rate even as the creator. **PUSH: What patterns predict which sessions will fail?**

6. "Code is no longer the bottleneck"

If this is true, it fundamentally changes what it means to be a software engineer. **PUSH: What IS the bottleneck now? Taste? Architecture? Understanding users?**

3 FRESH ANGLES (NEW FOR BOTH OF YOU)

ANGLE 1: THE TYPESCRIPT BOOK → AI CODING PIPELINE

THE HOOK	Boris wrote the definitive TypeScript book in 2019 advocating for "type-driven development" - sketching types first, then filling in code. Now he uses AI that generates code from natural language. How does his type-system thinking influence how he prompts AI?
WHY FRESH	Nobody has connected his TypeScript expertise to his Claude Code approach. His book emphasized types as "communication with the compiler" - does he now think of prompts as "communication with the model"? Is CLAUDE.md basically a type system for AI?
QUESTIONS	"Your TypeScript book talks about types as contracts that catch errors at compile time. Is CLAUDE.md essentially a 'prompt type system' that catches AI errors before they happen?" / "Do you think in types when you write prompts?" / "Has AI made your TypeScript expertise more or less valuable?"

ANGLE 2: THE META CODE QUALITY → AI CODE QUALITY PARADOX

THE HOOK	Boris was responsible for code quality across ALL of Meta's codebases. Now he advocates for a tool where he admits throwing away 10-20% of sessions and vibe coding produces "janky" results. How does the person who enforced code quality at Meta reconcile this?
WHY FRESH	Nobody has explored this tension. At Meta he spent years building systems to improve code quality at scale. Now he's building tools that generate code fast but admit quality tradeoffs. Has his definition of "quality" fundamentally changed?
QUESTIONS	"You spent years building code quality infrastructure at Meta. Does AI-generated code meet the standards you used to enforce?" / "What would 2022 Boris, the Meta code quality guy, think of 2026 Boris's workflow?" / "Is 'speed of iteration' replacing 'code quality' as the primary metric?"

ANGLE 3: THE ANTHROPIC CULTURE EXPERIMENT

THE HOOK	At Anthropic, everyone is "Member of Technical Staff" - no distinction between engineer, PM, or designer. Claude Code was a side project that became a flagship product. 80-90% of it is written by itself. This is a radical organizational experiment that nobody is discussing.
WHY FRESH	Most coverage focuses on Claude Code the tool. Nobody has explored what Anthropic is learning about organizations when AI writes most of the code. If titles don't matter and AI does the coding,

	what DOES an engineer do? This is relevant to your PM audience navigating AI transformation.
QUESTIONS	"If 90% of code is AI-written and titles don't exist, what do engineers actually DO at Anthropic?" / "You look for engineers who make kombucha - what does 'good engineer' mean when the AI codes?" / "Is Anthropic a preview of how all tech companies will work in 3 years?"

UNIQUE TALKING POINTS (NOT COVERED ELSEWHERE)

On Product Management & AI

- Your Claude Code masterclass covered usage, but not the PM philosophy behind WHY Claude Code is intentionally "not a product."
- His statement: "Build for the model 6 months from now, not today" - what does this mean for PMs scoping AI features?
- Cowork (Anthropic's new productivity tool) was built in 1.5 weeks with AI writing "pretty much all" the code. What does this mean for PM timelines?

On Engineering Leadership

- His approach to technical disagreements: "You have to earn trust first" - he brought beer and food to convince Meta skeptics
- Strong technical disagreement with senior TL Bob actually strengthened relationship and led to promotion
- Hiring philosophy: looks for "cool weekend projects" like kombucha brewing as signal of curiosity

Unexplored Tensions

- Claude Code controversy: "I heard Claude Code doesn't use evals" - this came up in your recent evals episode
- The job displacement debate: his workflow sparks both excitement and concern about team sizes
- Early Facebook quality was "super jank" by modern standards - his perspective on quality evolution

SOURCES

- The Peterman Pod (Dec 2025): developing.dev/p/boris-cherny-creator-of-claude-code
- AI & I Podcast: every.to/podcast/how-to-use-claude-code-like-the-people-who-built-it
- AI Engineer World's Fair: classcentral.com/course/youtube-claude-code-the-evolution-of-agentic-coding
- VentureBeat workflow article: venturebeat.com/technology/the-creator-of-claude-code-just-revealed-his-workflow
- MIT Tech Review: technologyreview.com/2025/12/15/1128352/rise-of-ai-coding-developers-2026
- The Neuron: theneuron.ai/explainer-articles/claude-codes-co-creator-just-revealed-how-coding-will-change
- Latent Space podcast: latent.space/p/claude-code
- Pragmatic Engineer: newsletter.pragmaticengineer.com/p/how-claude-code-is-built
- Programming TypeScript (O'Reilly): oreilly.com/library/view/programming-typescript/9781492037644
- X/Twitter: x.com/bcherny