WHEN YOUR USER IS AN AI AGENT

Product Management Beyond the Human Interface

TALK

MICHAEL HABLICH | GOOGLE

SPEAKER PROFILE

Michael Hablich

Product Manager @ Google

Leading Chrome DevTools & Puppeteer.

- 20 years: Developer → Product manager
- Building tools for millions of developers (and bots)
- Lecturer at University of Applied Sciences St. Pölten

User shift

TRADITIONAL PERSONA

"As a **human developer**, I want to inspect network requests..."

THE 2025 REALITY

"As a coding agent, I need programmatic access..."

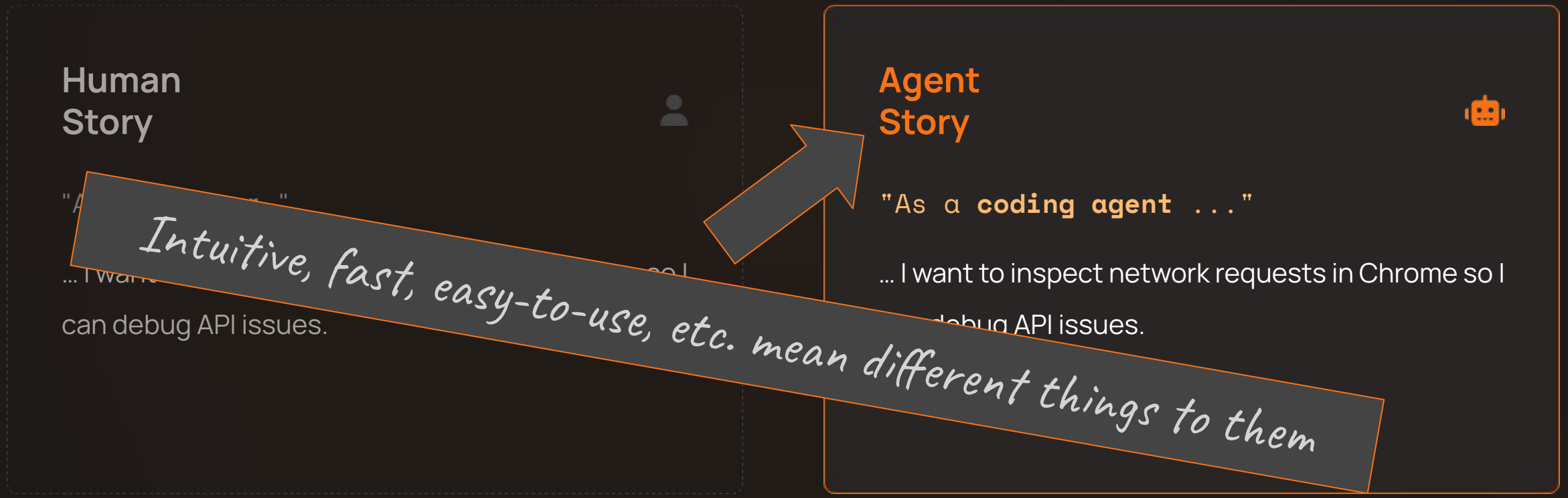
> IMPACT: Chrome, GitHub, Slack all shipped products in 2025 where the primary user is AI.

THE CORE PROBLEM

The blindfolded agent

Coding agents generate code but cannot see what it actually does.

The evolution of user personas



Writing agent personas

Define Technical Constraints: Unlike humans, agents have hard limits. Specify "200ms latency" or "4K token context" directly in the persona.

Zero-Shot Onboarding: Agents can onboard in a short period of time.

Autonomy over Assistance: Assume the agent runs autonomously, without human intervention.

Structured Output: What "language" does it speak?

Example: Chrome DevTools

Agent Story

"As a **coding agent** ..."

I need programmatic access to Chrome DevTools via MCP, so I can autonomously analyze & fix performance problems.

A(gent) (e)X(perience) metrics

Old world (Human)

Usage: DAU / MAU

Satisfaction: NPS / CSAT

etc. …

New world (Agent)

- **Usage:** DAU / MAU
- **Satisfaction:** NPS / CSAT of human handler
- **Reliability:** % of successful tool calls
- **Value:** Duration reduction & token decrease (Cost/Task)

The cost of cognitive load

up to 70 %

cost reduction

- Consider returning structured data
- Chunk content semantically
- Write clear error messages
- …

Tokens = Mental effort

For agents, verbose API responses aren't just annoying; they are a direct financial cost and a latency bottleneck.

Cost: 15 $ / MTok

300 Tok / API call

1M API calls per day

4500 $ per day → **1.6M $ per year**

30 % reduction is suddenly 0.5M $ per year.

Should we build for agents?

Do agents already TRY to use your product? Check logs for headless browsers, Browser use user agents, or superhuman form submission speeds.

Can you enable autonomous tasks? GitHub (PRs) and Slack (Messages) are natural fits. Pure creative judgment tools may not be ready.

Is there a gap? In Chrome DevTools, agents could write code but not see it. That gap was the product opportunity.

What's the effort? Reusing existing infrastructure and exposing it to agents is "relatively" cheap.

Product discovery

> AI agents don’t have emotions (yet).

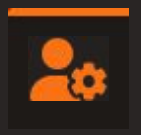
Four agent discovery Methods

Telemetry

Monitor logs for retry spikes. Where do agents get stuck?

Synthetic Tests

Run agent simulations through your product. Integration tests for AX.



Proxy Research

Interview the *builders*. "Where does your agent spend money?"

Comparative

A/B test API designs. Compare successful vs. struggling agents.

Chrome DevTools MCP - AX design principles 1/2

Agent-Agnostic API: Use standards like MCP. Don't lock in to one LLM. Interoperability is strategy.

Token-Optimized: Return semantic summaries. "LCP was 3.2s" is better than 50k lines of JSON.

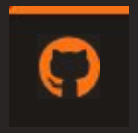
Small, Deterministic Blocks: Give agents composable tools (Click, Screenshot) not magic buttons.

Chrome DevTools MCP - AX design principles 2/2

Self-Healing Errors: Return actionable errors.

Human-Agent Collaboration: Output must be parseable by machines (JSON) AND readable by humans (Summaries).

The MCP ecosystem (~2,000 Servers)



GitHub

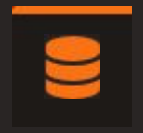
"Fix bug & create PR"

Exposes existing APIs to allow autonomous repo management.

Slack

"Summarize & DM me"

Interface to team comms.



Postgres

"Analyze churn data"

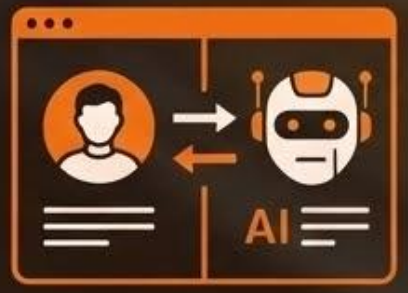
Turns agents into data analysts with safe read access.

COMMON THREAD: ENABLING AUTONOMOUS WORKFLOWS.

Expanding the PM skillset

Understanding agent architecture

Designing for dual interfaces



Measuring agent (user) success

Discovery without empathy

Action Items

This week:

- **Check your logs.** Look for headless browsers. Are they already here?
- **Talk to 3 customers**: Are they building agents?

This month:

- **Run the framework:** Should we build for agents?

Goal: Design for a future where your user base is a hybrid of humans and AIs.

Questions?

Have a good one!

linkedin.com/michael-hablich
@mhablich
hablich.dev