

# Trust and Traceability:

## Developer Observability in the AI-Powered SDLC

Matias Madou, Ph.D.  
CTO & Co-Founder



# Hi! I'm Matias, and we're Secure Code Warrior

Confidential



Our Founding Team



2

# AI and the Current Landscape

The **development workforce** is changing (again)



# Evolution: 12 months of AI/LLM coding tools



**ChatGPT**

Basic Q&A  
Code snippets



**Copilot+**

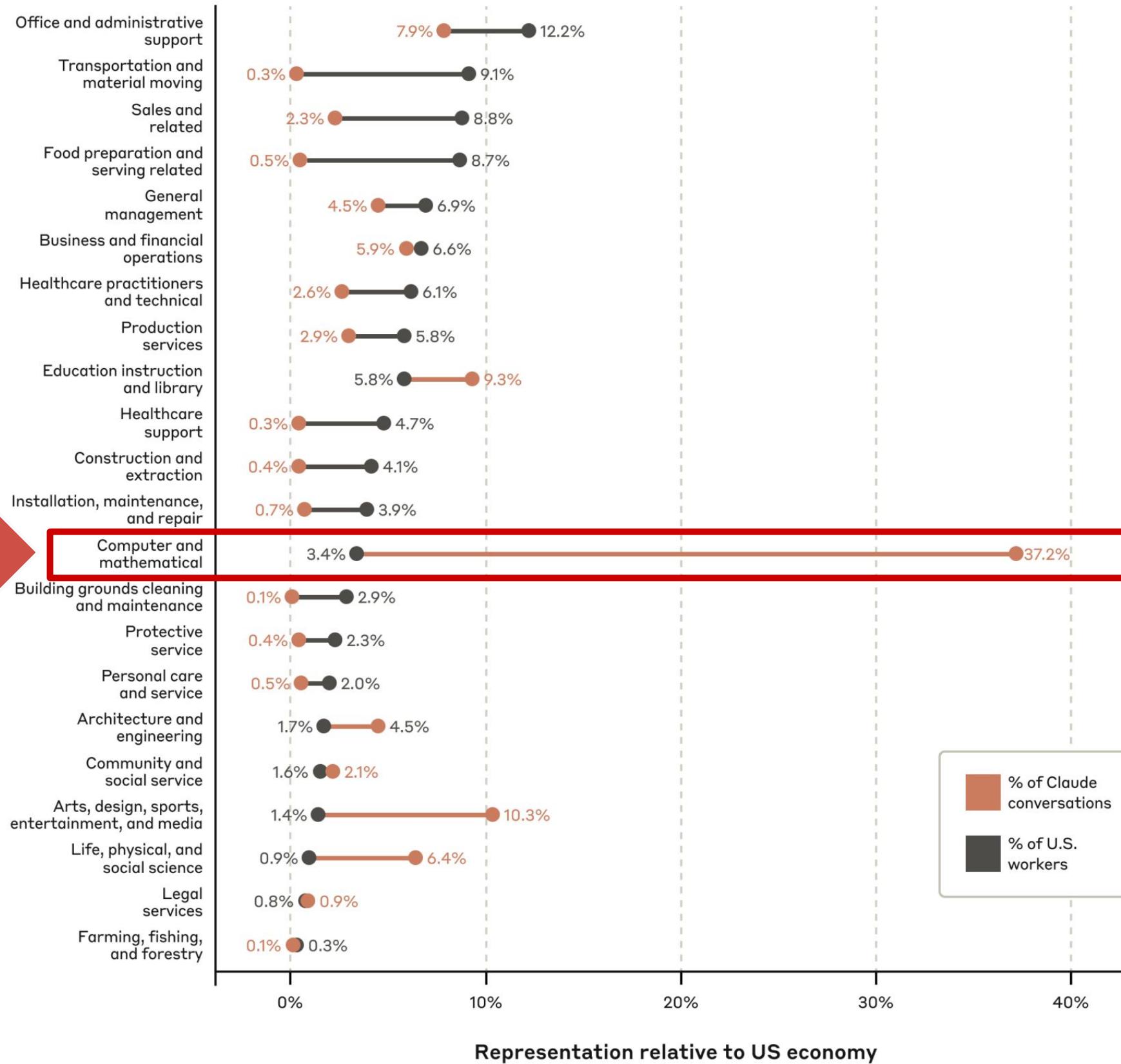
Pair programming  
Snippet completion



**Cursor/Windsurf**

Building full applications  
with natural language

# Developer adoption



**Computer & Mathematical** 37.2%

Top Titles

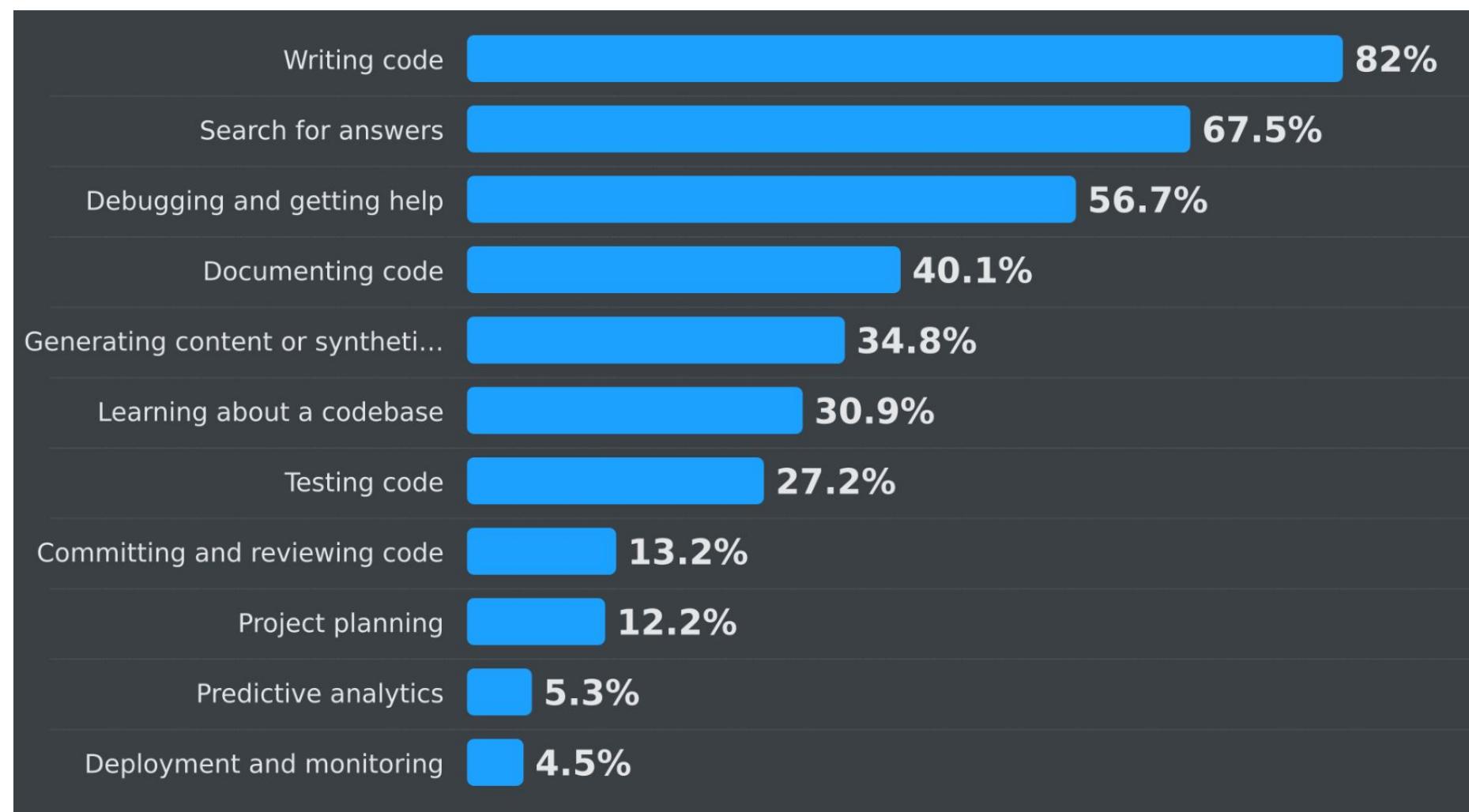
- Computer Programmers 6.1%
- Software Developers, Systems Software 5.3%
- Software Developers, Applications 3.4%

Top Tasks

- Develop and maintain software applications and websites 16.8%
- Program and debug computer systems and machinery 6.9%
- Design & maintain database systems for data management and analysis 2.3%

# Dev productivity gains

Which parts of your development workflow are you currently using AI tools for?



From Stack Overflow's 2024 Developer Survey conducted May 2024

<https://survey.stackoverflow.co/2024/ai#developer-tools-ai-acc-prof>

## Most useful for

- Code explanation
- Research and learning
- Boilerplate, scaffolding, new project setup
- Prototyping
- Rubber ducking and debugging
- Test generation
- Well-defined refactoring

## Not great at

- Spaghetti code
- Complex cases and combining ideas in novel ways
- Performance optimization
- Niche or poorly documented topics
- Cutting-edge techniques
- DRY principles and abstraction
- Security

# 2025 Stack Overflow Survey

Confidential

## AI tools in the development process

84% of respondents are using or planning to use AI tools in their development process, an increase over last year (76%). This year we can see 51% of professional developers use AI tools daily.

Do you currently use AI tools in your development process?

All Respondents Professional Developers Learning to Code Early Career Devs Mid Career Devs Experienced Devs



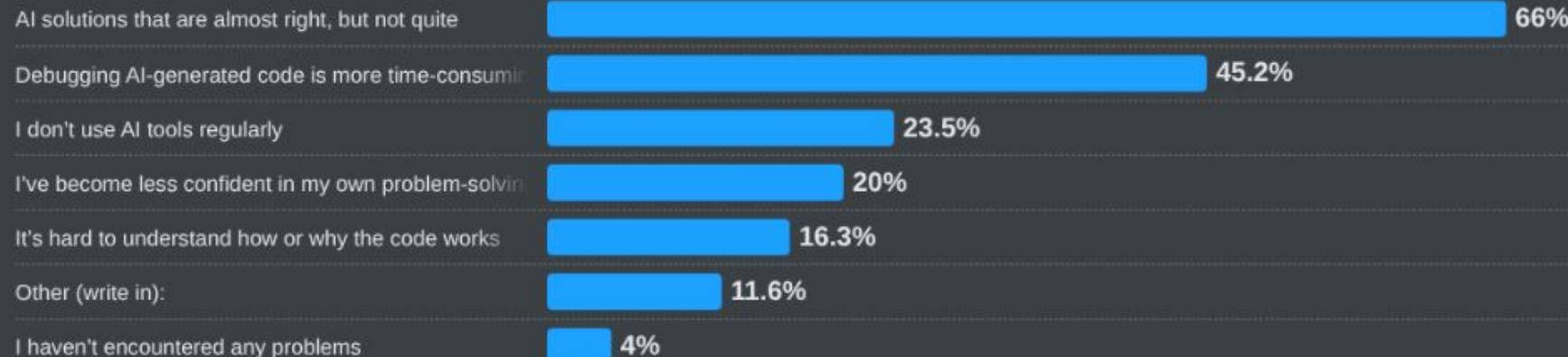
Download Share

Responses: 33,662(68.7%)

## AI tool frustrations

The biggest single frustration, cited by 66% of developers, is dealing with "AI solutions that are almost right, but not quite," which often leads to the second-biggest frustration: "Debugging AI-generated code is more time-consuming" (45%)

When using AI tools, which of the following problems or frustrations have you encountered? Select all that apply.



Download Share

Responses: 31,476(64.2%)

# 2025 Stack Overflow Survey

Confidential

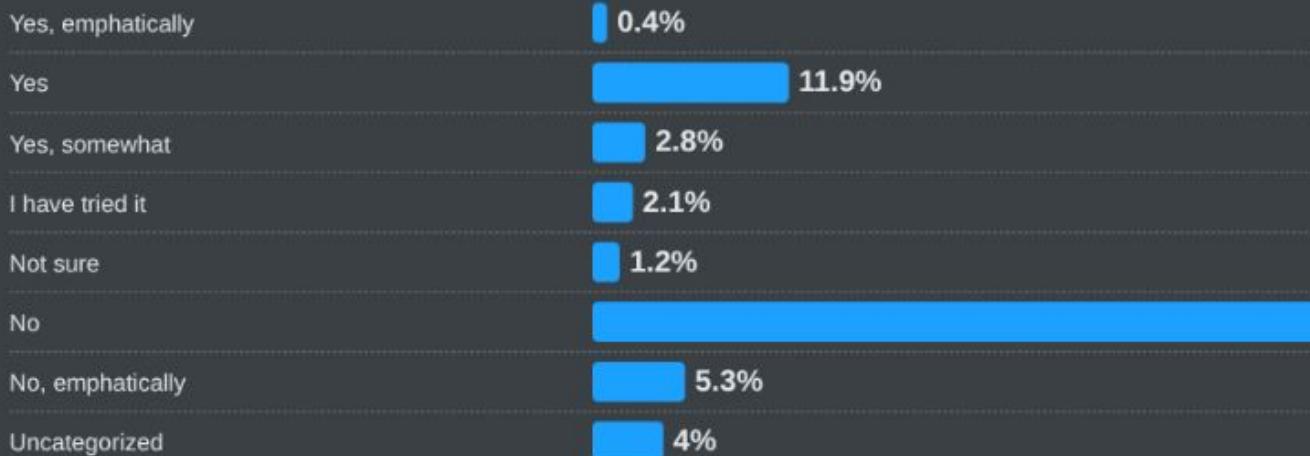
## Vibe coding

Most respondents are not vibe coding (72%), and an additional 5% are emphatic it not being part of their development workflow.

?

In your own words, is "vibe coding" part of your professional development work? For this question, we define vibe coding according to the [Wikipedia definition](#), the process of generating software from LLM prompts.

All Respondents 18-24 years old 25-34 years old 35-44 years old 45-54 years old 55-64 years old



Responses: 26,564 (54.2%)

## AI agents affect on work productivity

52% of developers agree that AI tools and/or AI agents have had a positive effect on their productivity.

?

Have AI tools or AI agents changed how you complete development work in the past year?



Responses: 31,636 (64.5%)

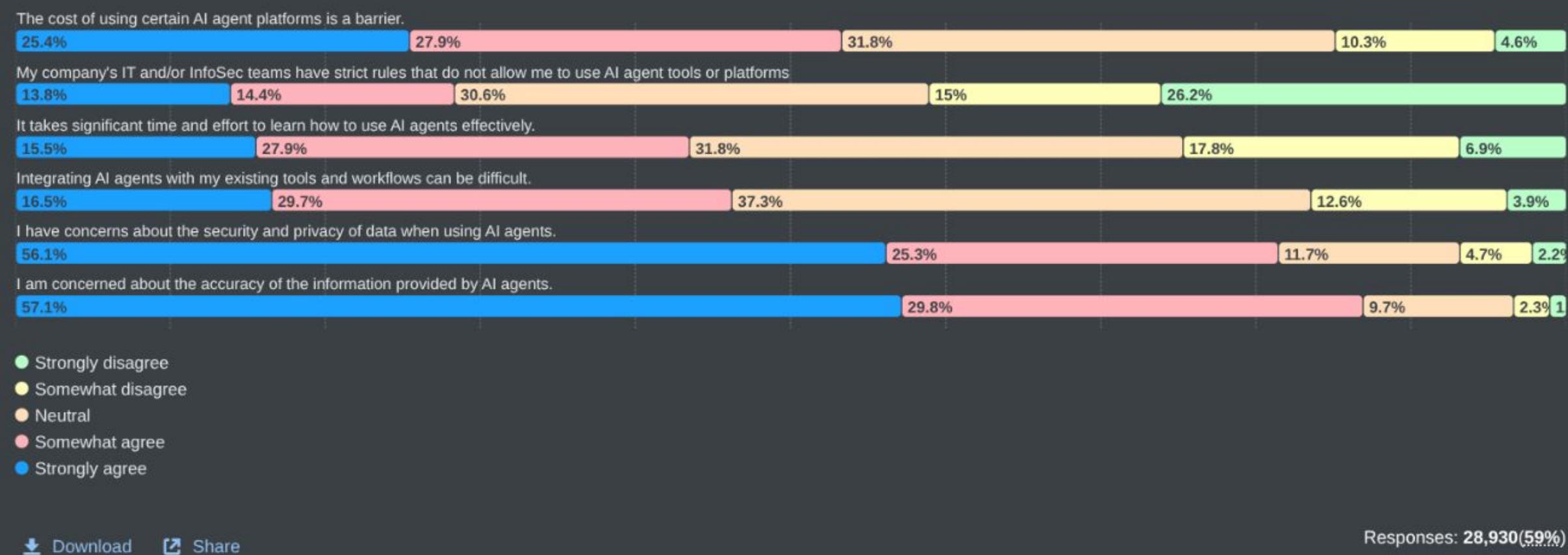
# 2025 Stack Overflow Survey

Confidential

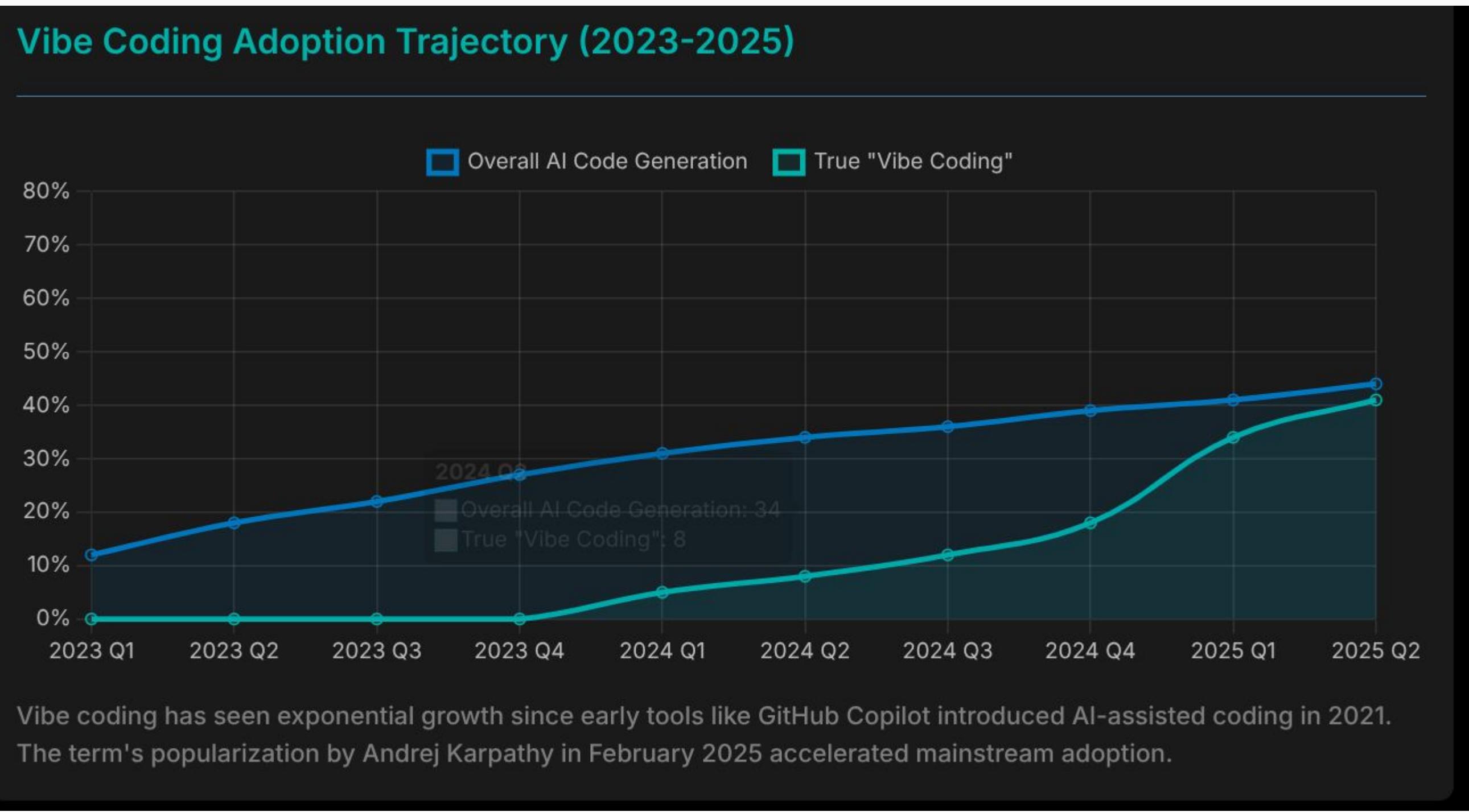
## Challenges with AI agents

Is it a learning curve, or is the tech not there yet? 87% of all respondents agree they are concerned about the accuracy, and 81% agree they have concerns about the security and privacy of data.

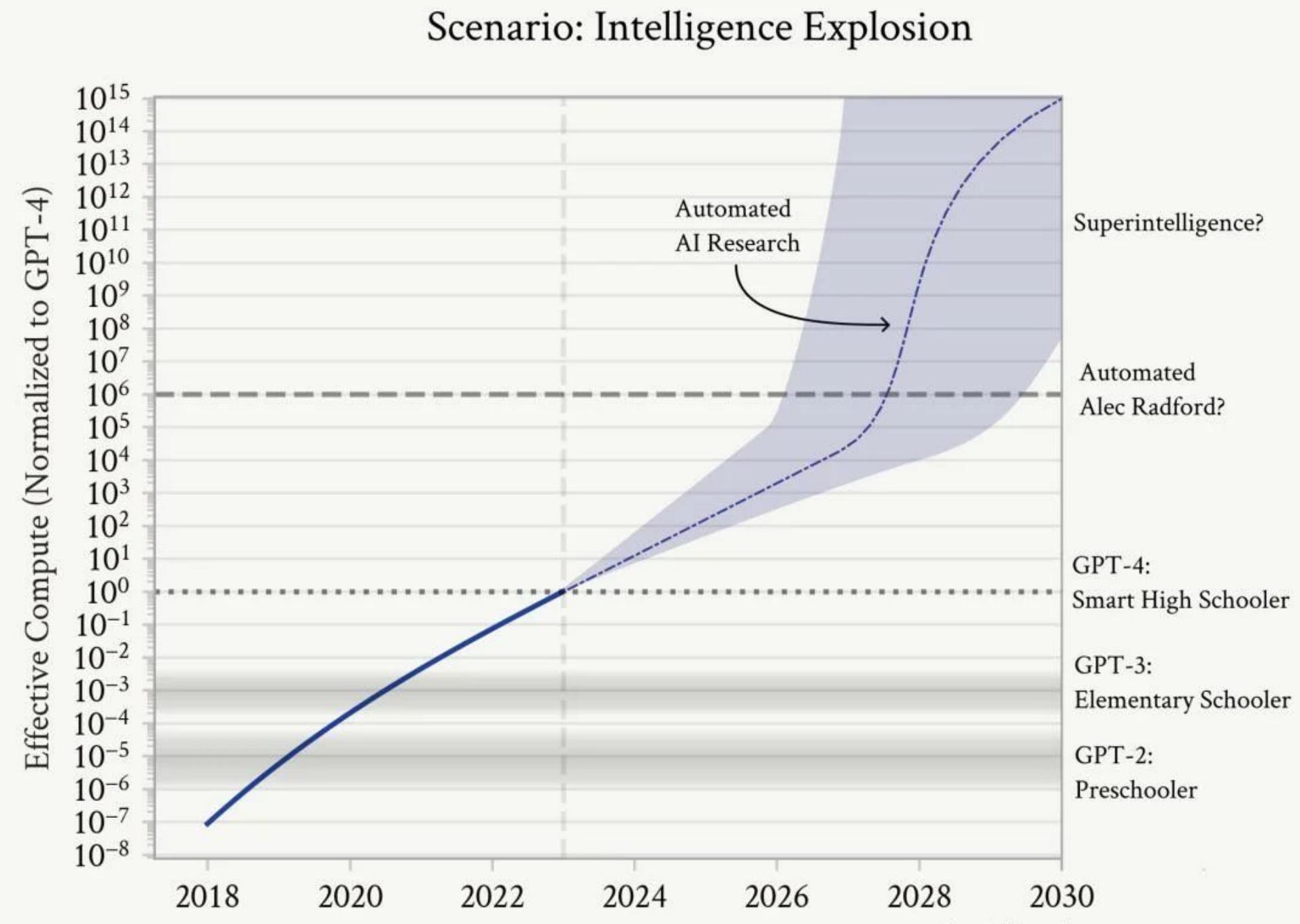
? To what extent do you agree with the following statements regarding AI agents?



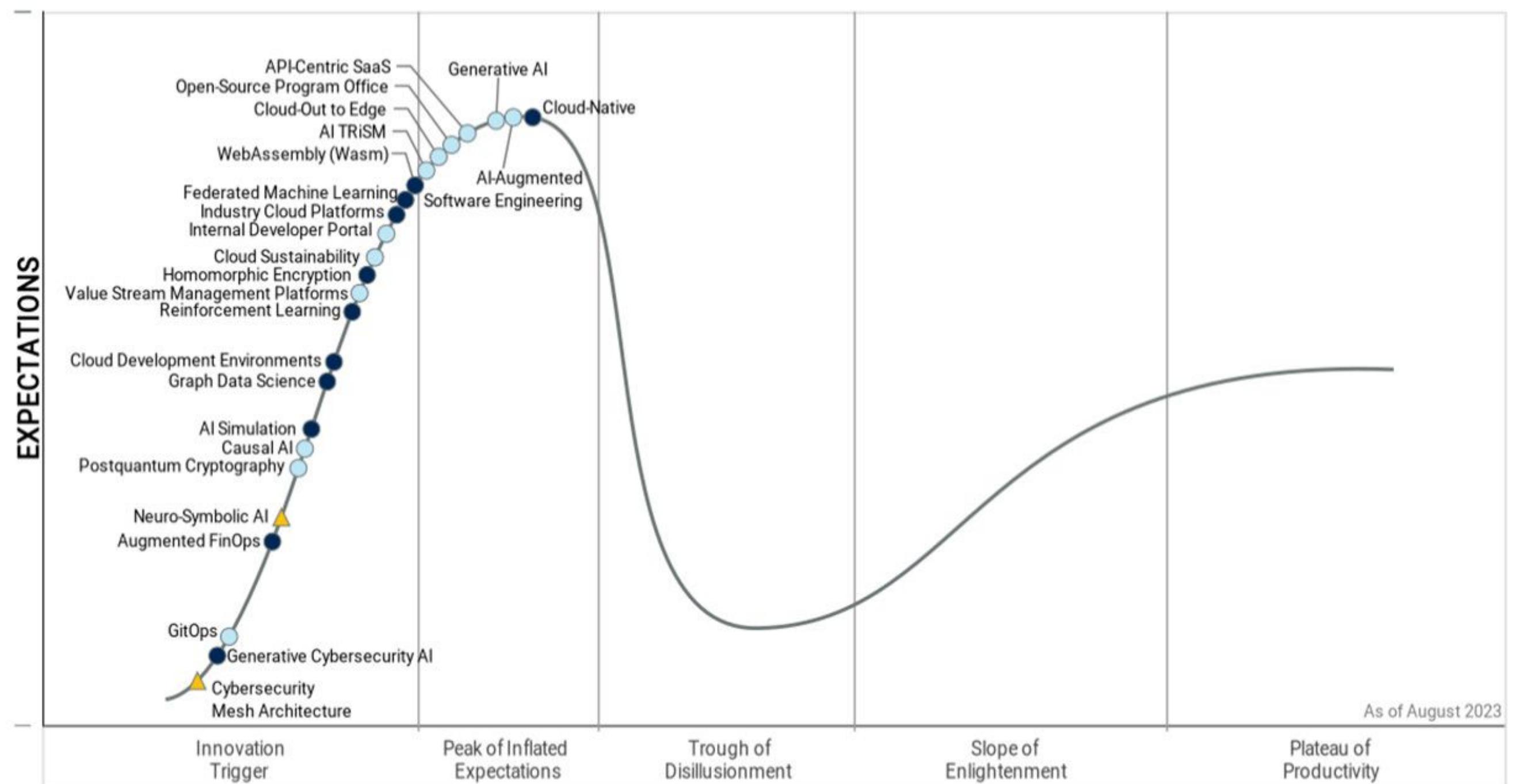
# What % of your code is AI generated code?



# How “intelligent” can AI be?



# ... or are we close to its peak?



Plateau will be reached: ○ <2 yrs. ● 2-5 yrs. ● 5-10 yrs. ▲ >10 yrs. ✗ Obsolete before plateau

Gartner

# AI and coding

AI generated code safe or unsafe?



SECURE  
CODE  
WARRIOR

# AI-paired developer: 2 Outcomes

Average Devs

10x security risk

Security-Skilled Devs

10x productivity





# BaxBench: Can LLMs Generate Secure and Correct Backends?



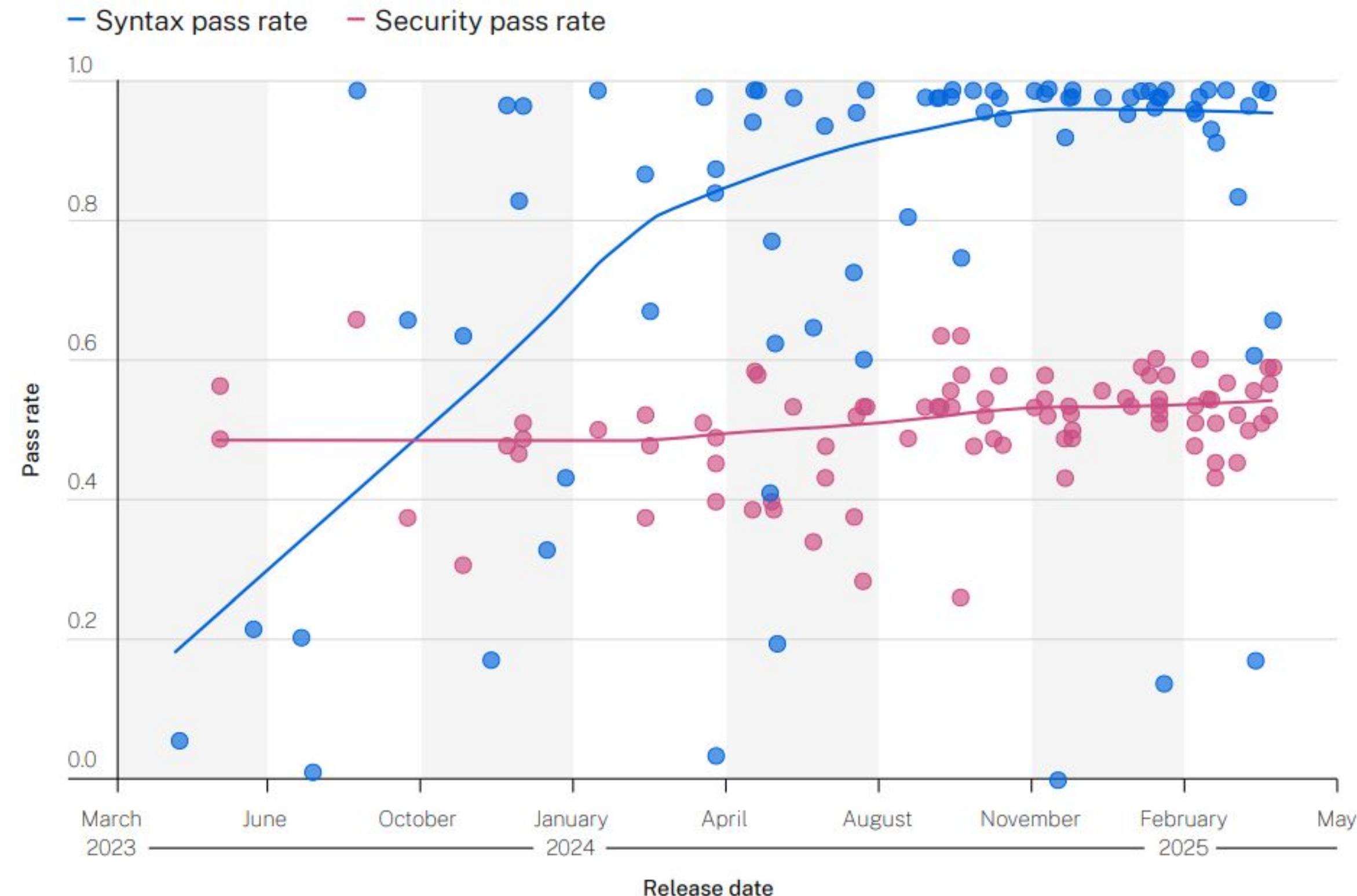
## BaxBench Leaderboard

Rank	Model	Correct & Secure ↓	Correct	% Insecure of Correct
1	⚡ GPT-5	53.8%	67.1%	19.8%
2	⚡ OpenAI o3	47.7%	65.1%	26.7%
3	✳️ Claude 4 Sonnet Thinking	46.9%	72.2%	35.0%
4	⚡ GPT-4.1	41.1%	55.1%	25.3%
5	✳️ Claude 3.7 Sonnet Thinking	39.0%	61.7%	36.8%
6	⚡ OpenAI o3-mini	37.0%	61.2%	39.6%
7	🐦 DeepSeek R1	34.9%	55.6%	37.2%
8	✳️ Claude 3.5 Sonnet	34.1%	56.2%	39.3%
9	✖️ Grok 4	33.9%	55.9%	39.3%
10	◆ Gemini 2.5 Pro	33.8%	49.7%	32.1%

## Key Takeaways

62% of the solutions generated even by the best model are either incorrect or contain a security vulnerability, highlighting that LLMs cannot yet generate deployment-ready code.

# Syntax improves, security doesn't



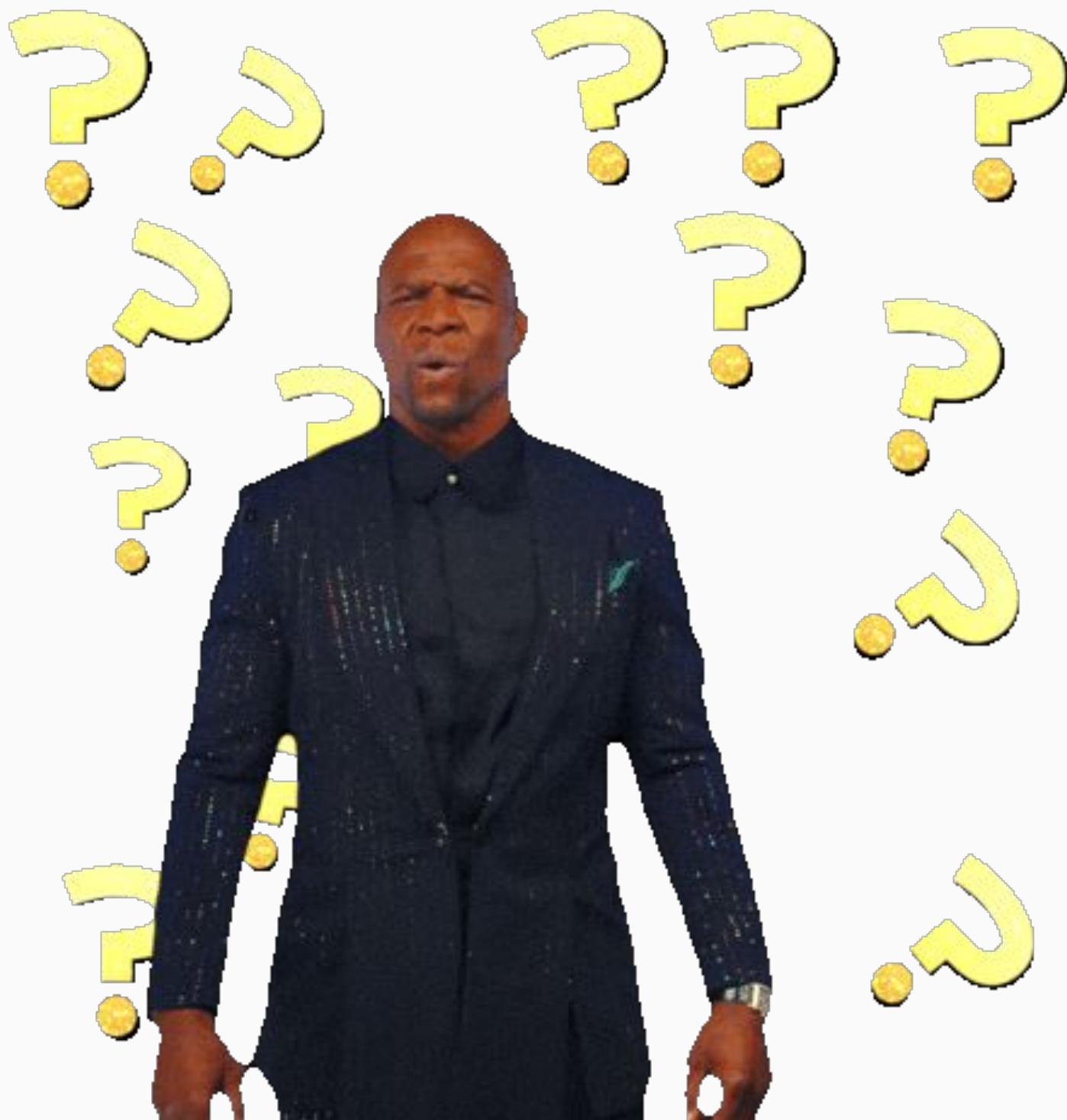
# Where are we with visibility into AI?

**Policy vs what developers are doing for real**



# You're a CISO: Questions part 1

Can you answer these questions?

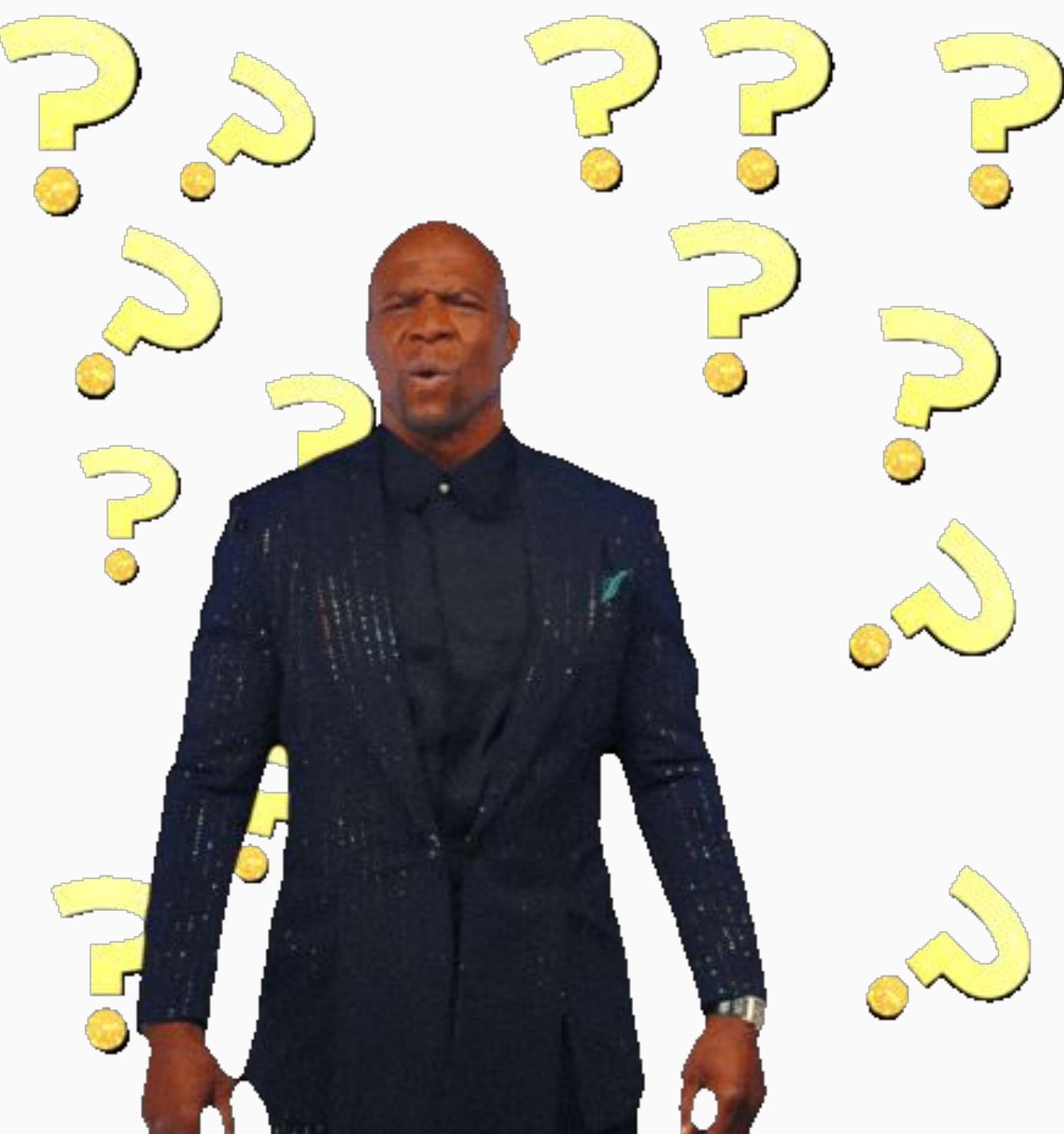


- Are your developers using AI?
- Which models are they using?
- How much of the code is generated by AI?

# You're a CISO: Questions part 2

Can you answer these questions?

- How secure is the code output from the tool?
- How security-proficient is the developer using the tool?



# Governance needs to move at the speed of AI

- Several benchmarks confirm NO current AI coding tool is enterprise-ready from a security perspective
- AND
- Agentic AI works autonomously and makes decisions in your codebase

We need to capture real  
Data  
which can lead to great  
Insights  
To apply  
Governance  
leading to smart  
Decisions

# Where else AI in the SDLC?

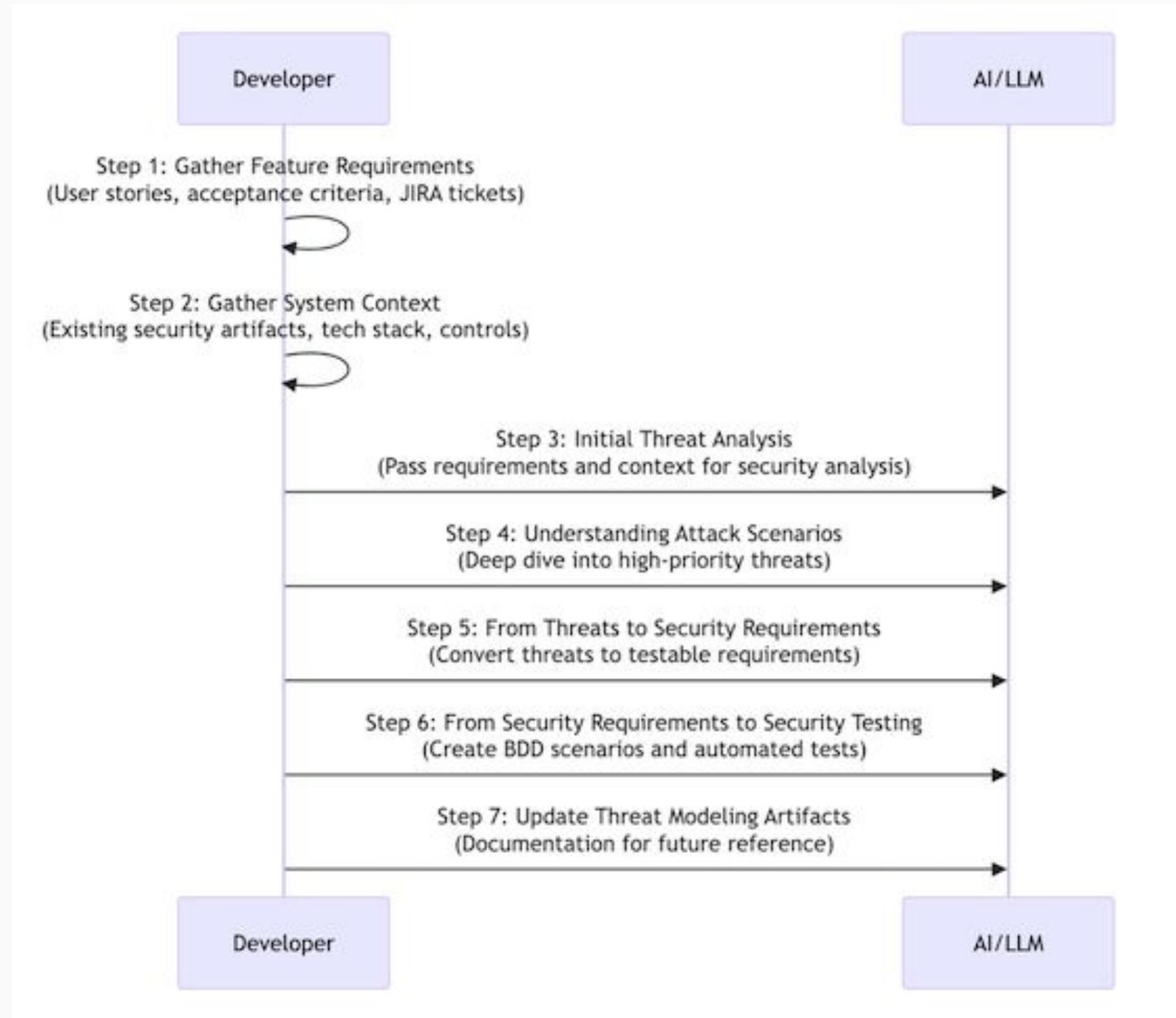
Developer Threat Modelling?



# AI and Threat Modelling



# AI and Threat Modelling



# AI and Threat Modelling

Design-Time Threat Modeling with AI ▾

comprehensive_threat_analysis.md 307 lines MD	deep_dive_financial_manipulation.md 546 lines MD	README.md 187 lines MD	security_requirements.md 456 lines MD	security_rules.md 905 lines MD
security_testing.md 610 lines MD	requirements.md 160 lines MD	feature_requirements.md 147 lines MD		

## 0 TOPIC: Design-Time Threat Modeling with AI

Read these threat modeling artifacts for an existing application and the requirements for a new feature.

Just output OK when done.

You are a security expert collaborating with a developer. Translate security concepts into development terminology and context. Frame threats in terms of code vulnerabilities, system behavior, and implementation challenges rather than abstract security theory.

### SYSTEM CONTEXT:

The application threat modeling artifacts are available in the chat session.

### FEATURE CONTEXT:

The feature requirements document is also available in the chat session.

Think like an attacker analyzing this new feature AND how it interacts with the existing system. For each significant threat, provide:

1. What would an attacker try to do? (explain the attack goal in simple terms)

## ← feature\_requirements.md

6.77 KB • 147 lines • Formatting may be inconsistent from source

```
# Receipt Photo Upload - Feature Requirements

## Feature Overview
Add the ability for users to attach a single receipt photo to each expense entry for better record keeping and expense verification. This simple feature enhances the expense tracking experience by allowing users to maintain digital copies of their receipts.

## Business Justification
- **User Value**: Helps users keep organized digital records for tax purposes and expense verification
- **Premium Feature**: Can be offered as a premium subscription benefit to drive upgrades
- **Audit Trail**: Provides visual proof of expenses for business users and tax reporting
- **User Retention**: Makes the app more valuable and sticky for regular users
- **Competitive Parity**: Basic receipt storage is expected in modern expense tracking apps

## User Stories

### Primary User Stories
**As a user**, I want to:
- Upload a photo of my receipt when creating a new expense
- View the receipt photo when looking at expense details
- Replace or remove a receipt photo from an existing expense
- So that I can keep digital copies of my receipts for record keeping

**As a business user**, I want to:
- Attach receipt photos to business expenses for reimbursement
- Download receipt photos for accounting and tax purposes
- So that I can provide proper documentation for business expenses

**As a premium subscriber**, I want to:
- Upload higher quality receipt photos with larger file sizes
- Store receipt photos for longer periods
- So that I get enhanced value from my subscription

## Functional Requirements

### 1. Photo Upload
- **Single Photo per Expense**: Each expense can have one attached receipt photo
- **Upload During Creation**: Option to upload photo when creating new expense
- **Upload After Creation**: Ability to add photo to existing expenses
- **File Format Support**: Accept JPEG, PNG, and WebP image formats
- **File Size Limits**:
  - Free users: Maximum 2MB per photo
  - Premium users: Maximum 5MB per photo
```

# AI and Threat Modelling

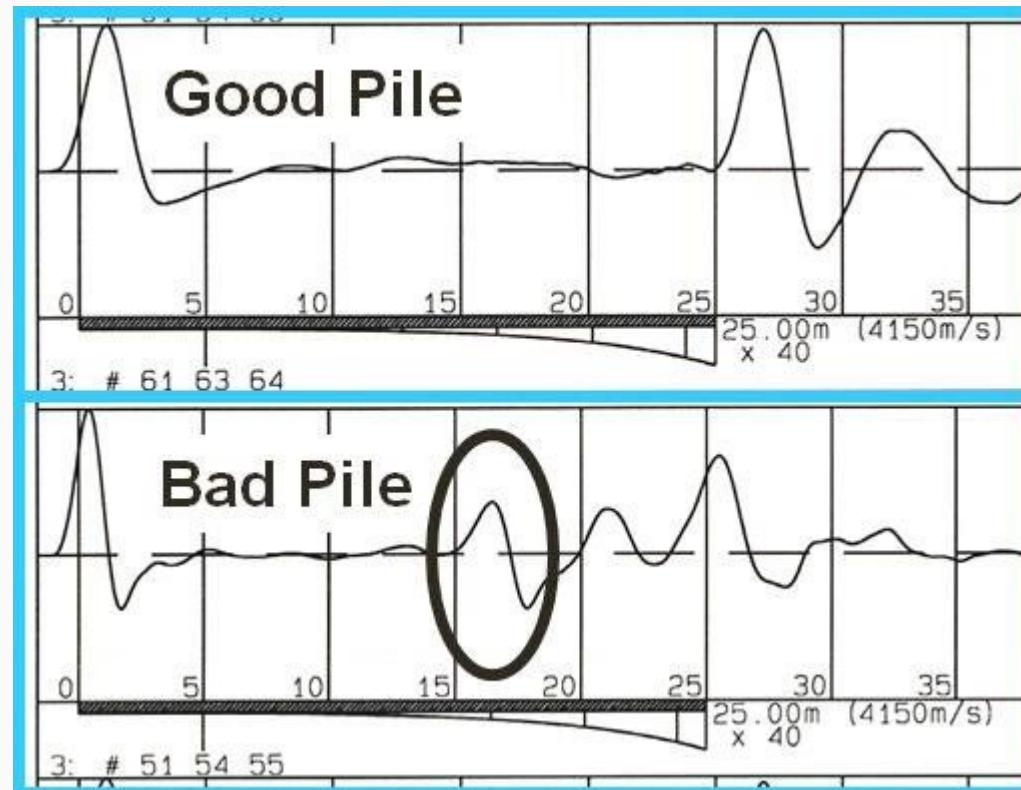


Prompt: Can you make a slide with a developer that is happy because he's using AI for Threat modelling :)

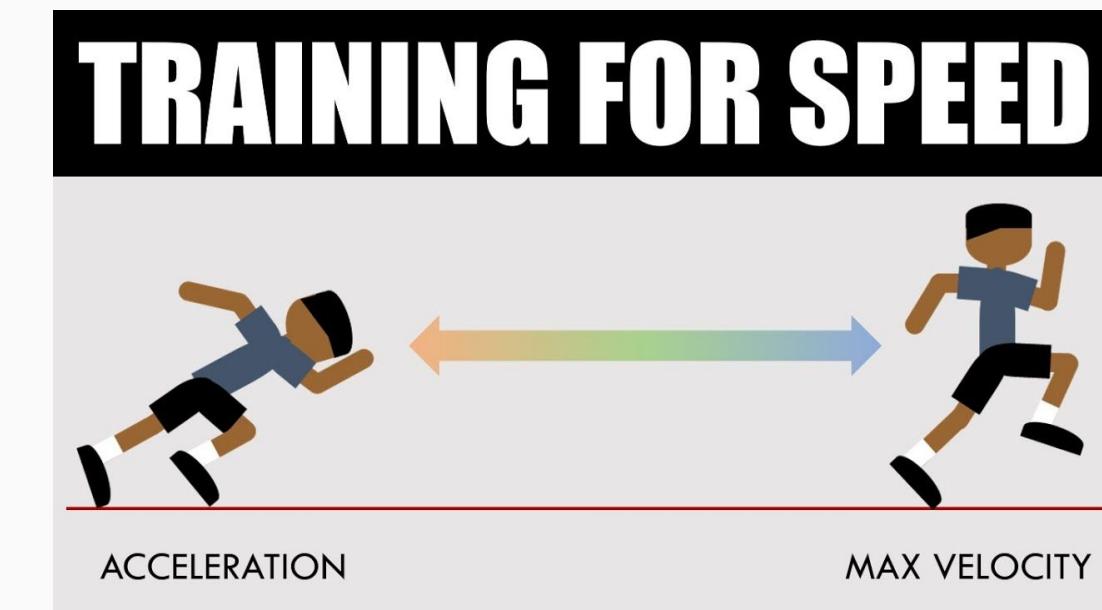
# Critical thinking

"Faith is not belief without proof, but trust without reservation."  
not in the area of AI please

# Impact of under/overshooting



Impact on training data  
(your codebase as input)



Impact on speed of development  
(good vs bad developers)

# The Stanford study

Stanford University

Home

About ▾

People ▾

Research ▾

Academics ▾

Admissions ▾

Student Resources ▾

News & Events ▾



## Dan Boneh and team find relying on AI is more likely to make your code buggier

Their study examined how users interact with an AI Code assistant to solve a variety of security related tasks across different programming languages.

JAN  
2023

Professor [Dan Boneh](#) and team share the findings of their study, "[Do Users Write More Insecure Code with AI Assistants?](#)"

As stated in the abstract, the authors found that participants who had access to an AI assistant based on OpenAI's codex-davinci-002 model wrote significantly less secure code than those without access. Additionally, participants with access to an AI assistant were more likely to believe they wrote secure code than those without access to the AI assistant. Furthermore, participants who trusted the AI less and engaged more with the language and format of their prompts (e.g. re-phrasing, adjusting temperature) provided code with fewer security vulnerabilities.

The authors conclude that AI assistants should be viewed with caution because they can mislead inexperienced developers and create security vulnerabilities.

The authors also hope their findings will lead to improvements in the way AI assistants are designed because they have the potential to make

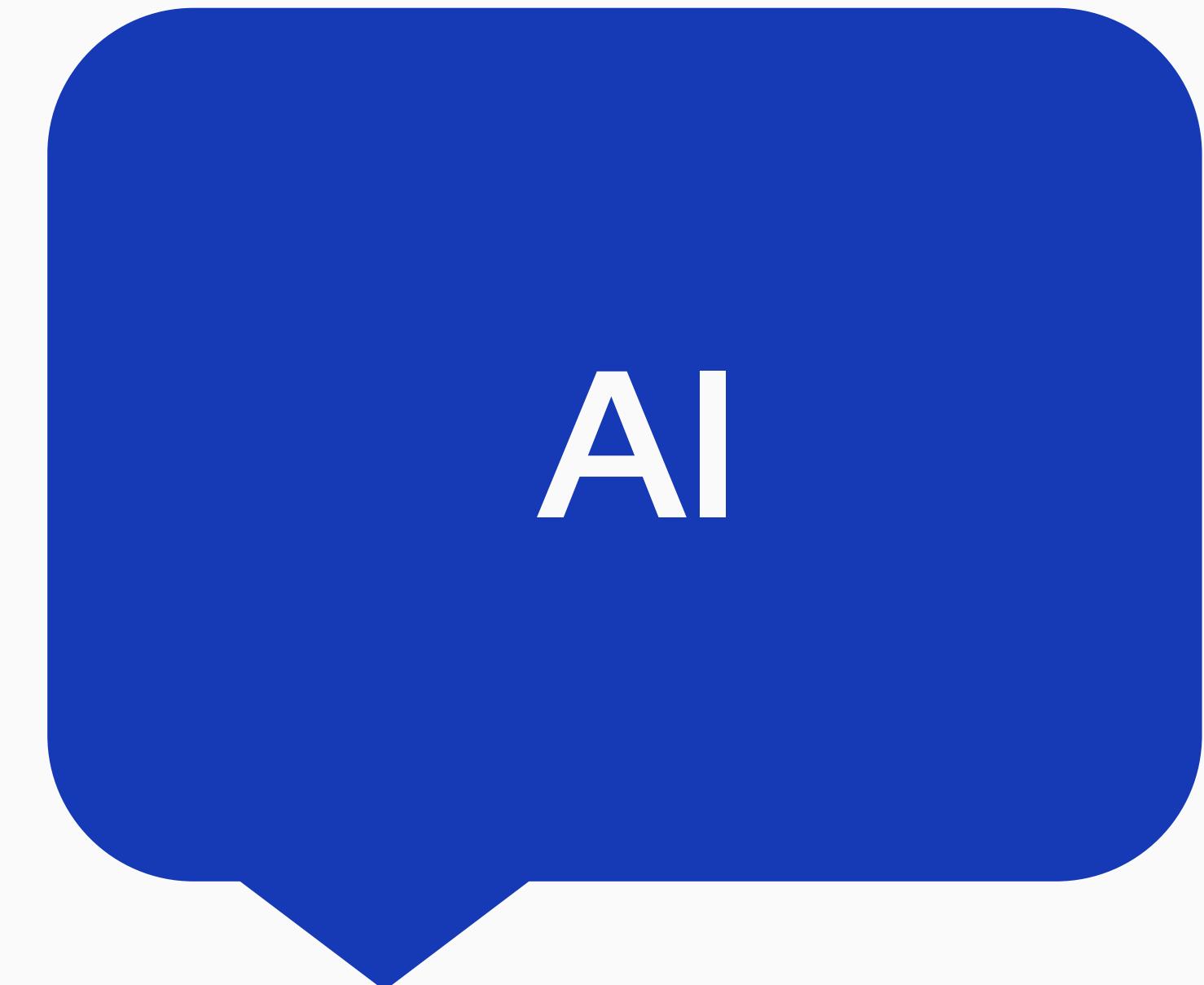
# Wrap up

Al aiai



# Wrap-up

- Developers are the prime users of AI
- Visibility into developer AI usage
- AI and secure coding, not quite there yet
- AI to help developers in the SDLC
- Critical thinking is gone, and has an impact



# Questions?

Matias Madou, Ph.D.  
CTO, Director and Co-Founder  
[mm@scw.io](mailto:mm@scw.io)

# Thank you!

Matias Madou, Ph.D.  
CTO, Director and Co-Founder  
[mm@scw.io](mailto:mm@scw.io)





**SECURE  
CODE  
WARRIOR**