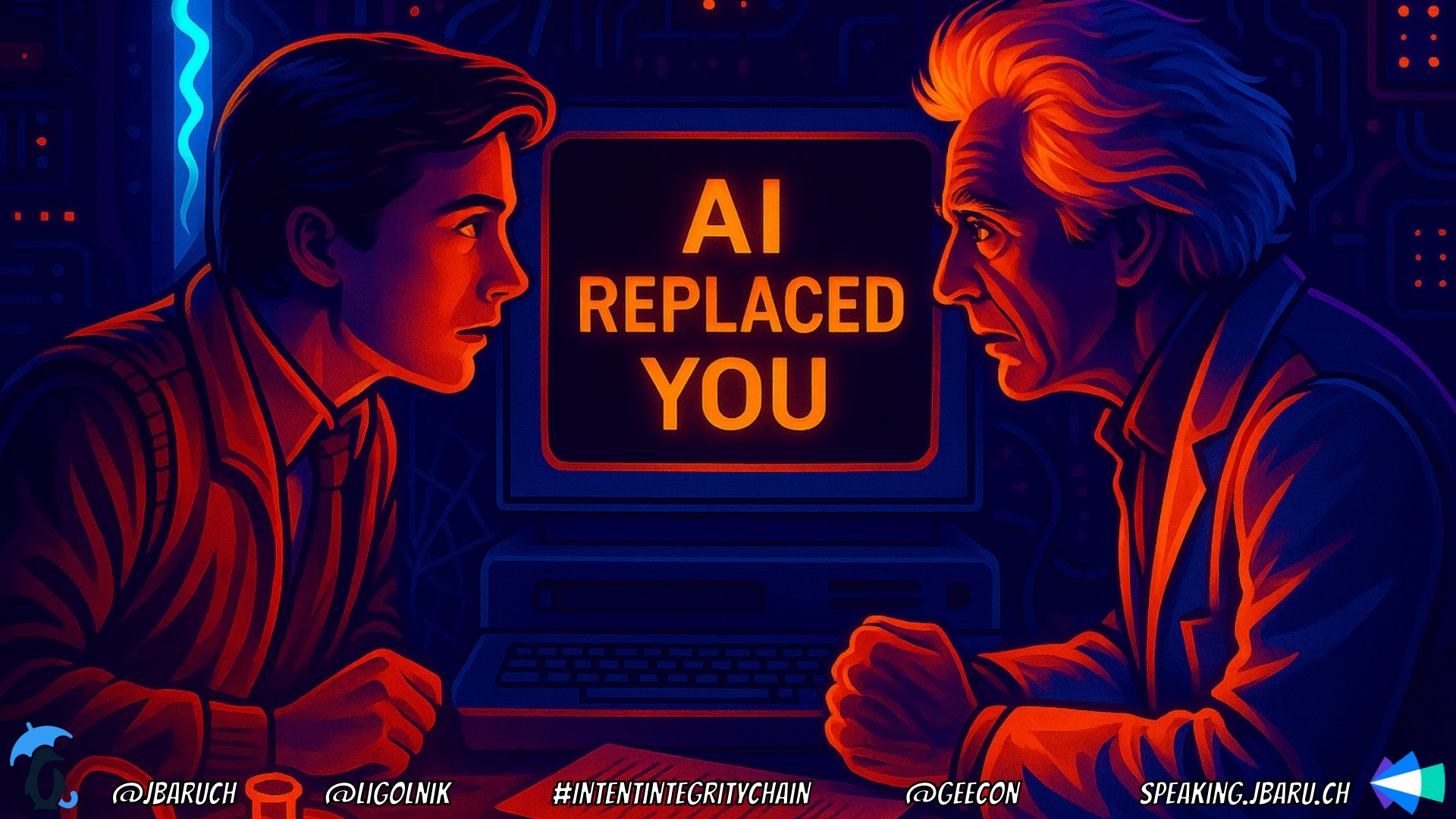
BACK TO THE FUTURE OF SOFTWARE

How to Survive the AI Apocalypse with Tests, Prompts, and Specs

MAY 16 2025

@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH


AI
REPLACED
YOU
@JBARUCH
@LIGOLNIK
#INTENTINTEGRITYCHAIN
@GEECON
SPEAKING.JBARU.CH

TechCrunch

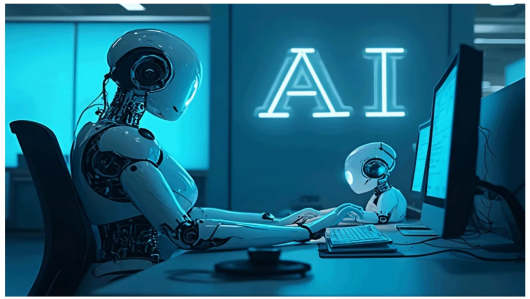
Programmers bore the brunt of Microsoft's layoffs in its home state as AI writes up to 30% of its code

Coders were hit hardest among Microsoft's 2,000-person layoff in its home state of Washington, Bloomberg reports.

techradar pro THE BUSINESS TECHNOLOGY EXPERTS

The Gr-AI-m Reaper: Hundreds of jobs at IBM and Crowdstrike vanish as artificial intelligence makes humans more dispensable

Both companies stress humans still have a role to play - for now

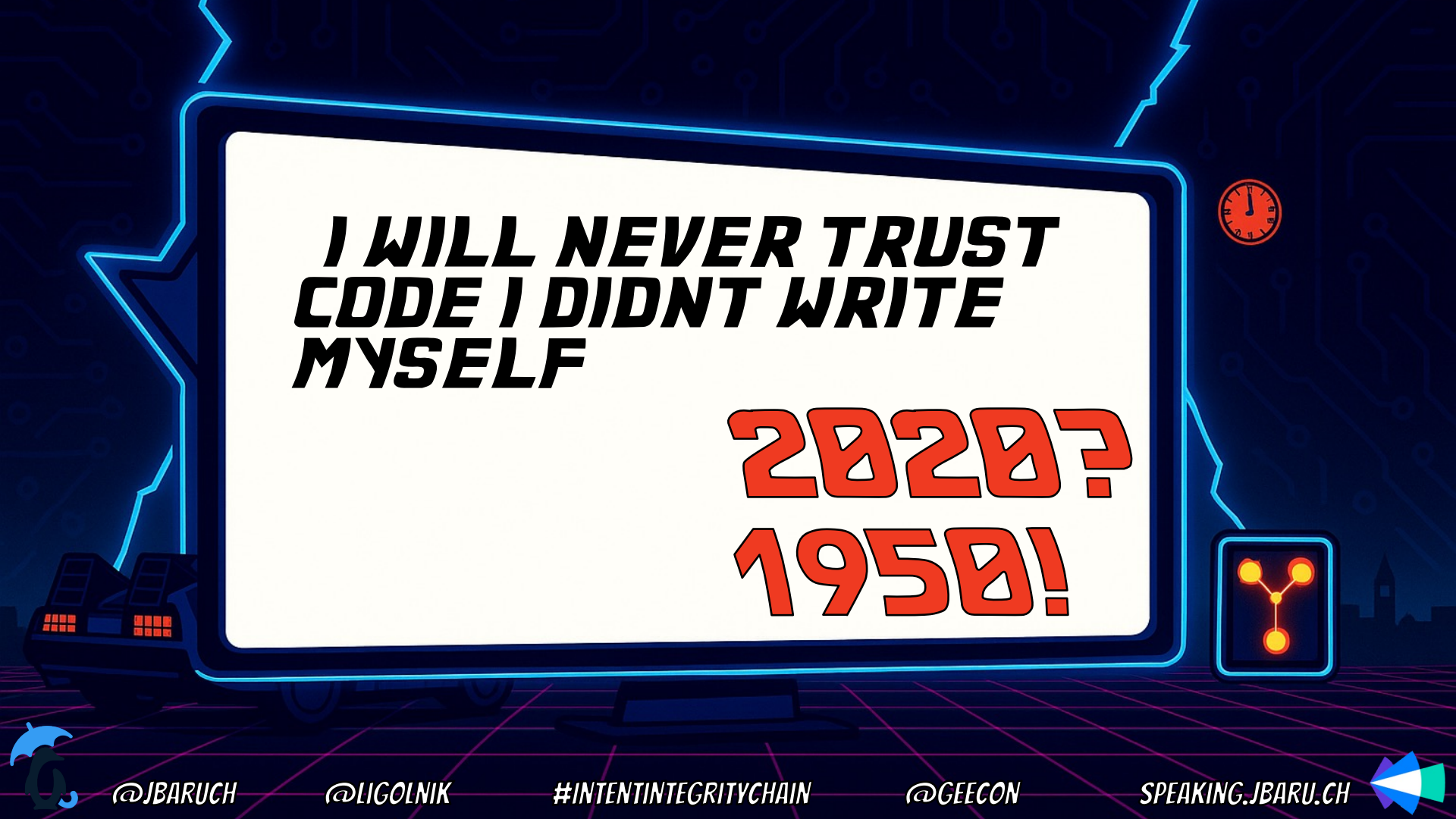


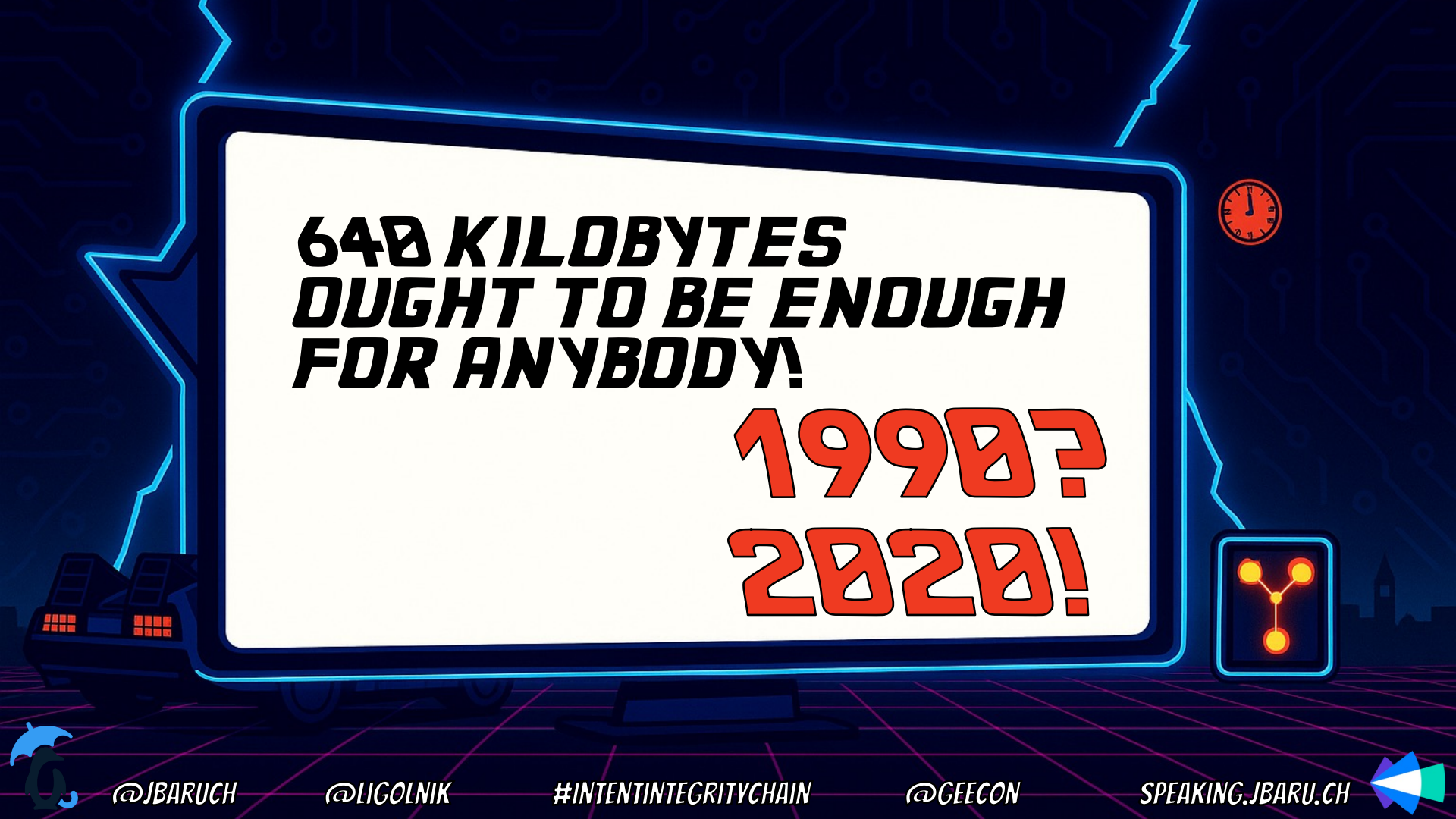
BBC

AI could replace equivalent of 300 million jobs - report

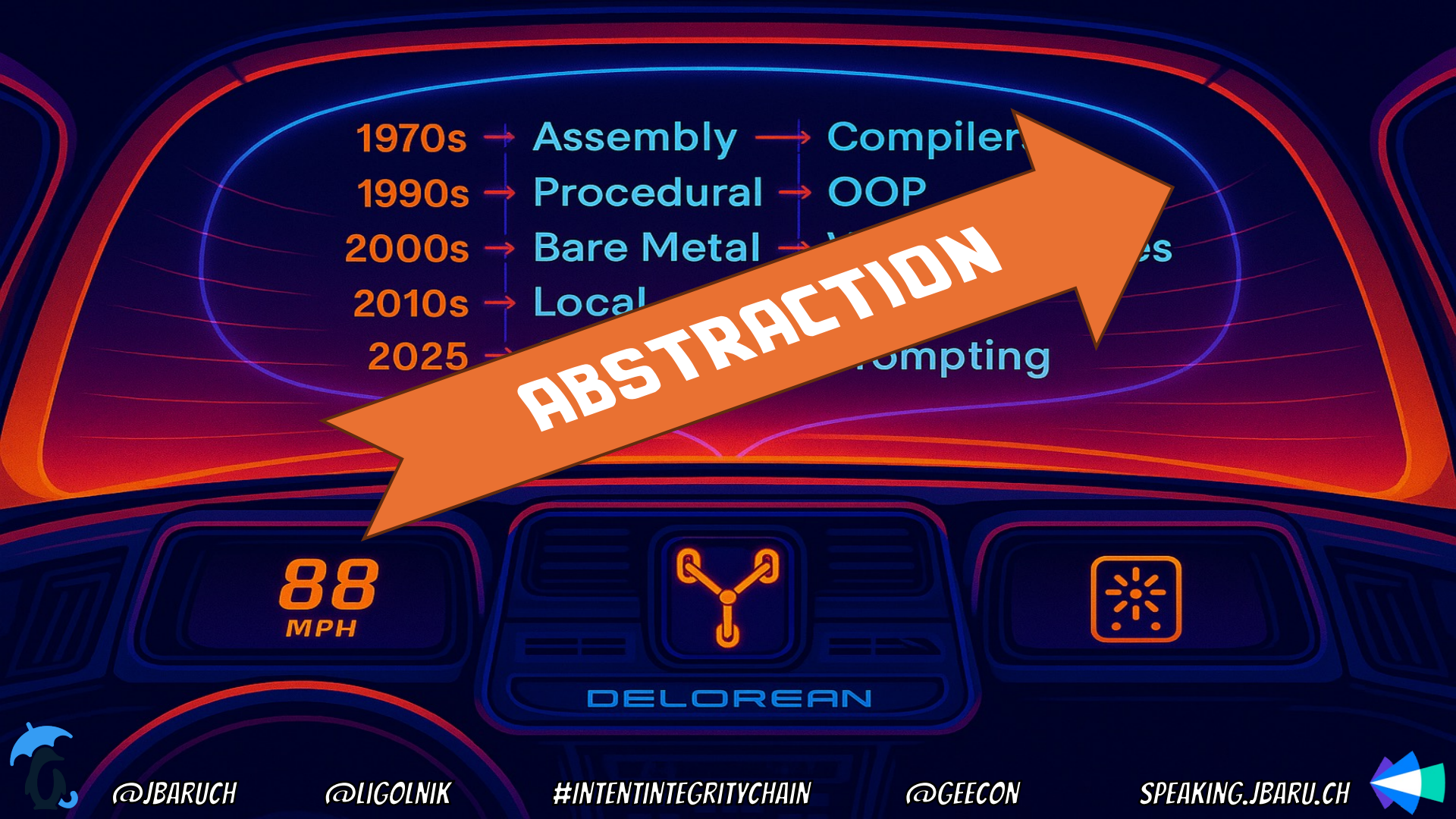
Artificial intelligence (AI) could replace the equivalent of 300 million full-time jobs, a report by investment bank Goldman Sachs says.

- WHO USES LLM ON A DAILY BASIS?
- FOR WORK | OUT OF WORK
- FOR CODING?
- WHAT ARE YOU USING IT FOR?
- WHEN YOU RE USING IT FOR CODE, ARE YOU GETTING THE RESULTS YOU WANT?


I WILL NEVER TRUST CODE I DIDNT WRITE MYSELF
2020?
1950!
@JBARUCH
@LIGOLNIK
#INTENTINTEGRITYCHAIN
@GEECON
SPEAKING.JBARU.CH

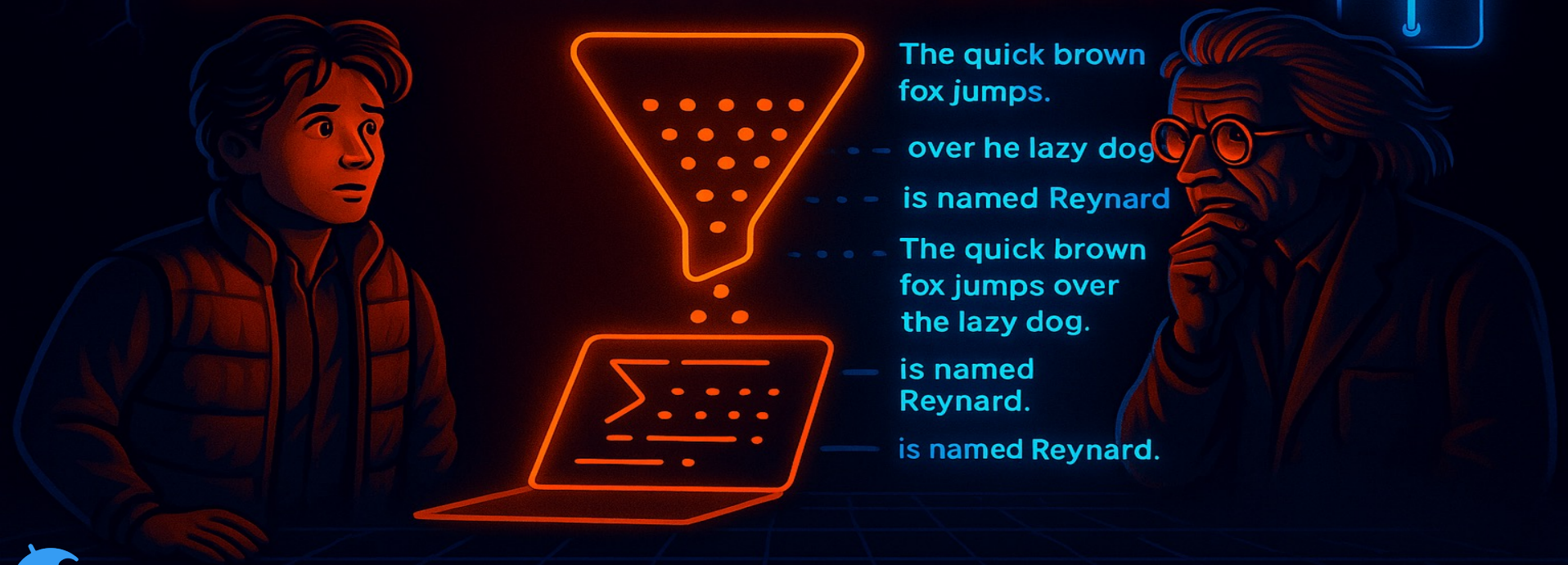

640 KILOBYTES
DUGHT TO BE ENOUGH
FOR ANYBODY!
1990?
2020!
@JBARUCH
@LIGOLNIK
#INTENTINTEGRITYCHAIN
@GEECON
SPEAKING.JBARU.CH

WE'VE SEEN THIS PANIC BEFORE


1970s → Assembly → Compiler
1990s → Procedural → OOP
2000s → Bare Metal →
2010s → Local
2025 →
ompting
ABSTRACTION
88
MPH
DELOREAN
@JBARUCH
@LIGOLNIK
#INTENTINTEGRITYCHAIN
@GEECON
SPEAKING.JBARU.CH

THIS TIME IT'S REAL WE REALLY CAN'T TAREST IT

SAME PROMPT, DIFFERENT RESULTS



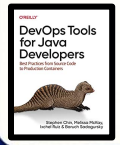
AN INFINITE NUMBER OF MONKEYS WITH AN INFINITE NUMBER OF TYPEWRITERS AND AN INFINITE AMOUNT OF TIME COULD EVENTUALLY WRITE THE WORKS OF SHAKESPEARE

WE GAVE THEM GPUs.

@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH

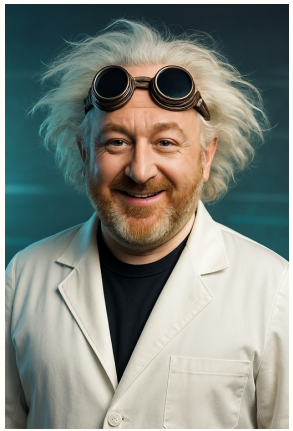
BARUCH SADOGURSKY - @JBARUCH

- HEAD OF DEVREL AT TUXCARE - I AM HIRING!
- ABSTRACTION CONNOISSEUR
- DEVELOPMENT ➟ DEVOPS ➟ #INTENTINTEGRITYCHAIN



LEONID IGOLNIK - @LIGOLNIK

- ✗ EVP OF ENG AT CLARI - I AM HIRING!
- ✗ ABSTRACTION WRANGLER
- ✗ DEVELOPMENT ➟ ARCH ➟ LEADER



SHOWNOTES

- SPEAKING.JBARU.CH
- SLIDES
- VIDEO
- ALL THE LINKS!

QUESTIONS?

@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH

AI-GENERATED CODE IS NOT GREAT

arXiv > cs > arXiv:2304.10778

Search

Help | A

Computer Science > Software Engineering

[Submitted on 21 Apr 2023 (v1), last revised 22 Oct 2023 (this version, v2)]

Evaluating the Code Quality of AI-Assisted Code Generation Tools: An Empirical Study on GitHub Copilot, Amazon CodeWhisperer, and ChatGPT

Burak Yetiştiren, Işık Özsoy, Miray Ayerdem, Eray Tüzün

Context: AI-assisted code generation tools have become increasingly prevalent in software engineering, offering the ability to generate code from natural language prompts or partial code inputs. Notable examples of these tools include GitHub Copilot, Amazon CodeWhisperer, and OpenAI's ChatGPT.

Objective: This study aims to compare the performance of these prominent code generation tools in terms of code quality metrics, such as Code Validity, Code Correctness, Code Security, Code Reliability, and Code Maintainability, to identify their strengths and shortcomings.

Method: We assess the code generation capabilities of GitHub Copilot, Amazon CodeWhisperer, and ChatGPT using the benchmark HumanEval Dataset. The generated code is then evaluated based on the proposed code quality metrics.

Results: Our analysis reveals that the latest versions of ChatGPT, GitHub Copilot, and Amazon CodeWhisperer generate correct code 65.2%, 46.3%, and 31.1% of the time, respectively. In comparison, the newer versions of GitHub Copilot and Amazon CodeWhisperer showed

@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH

ON TOP OF THAT, IT IS DANGEROUS

arXiv > cs > arXiv:2108.09293

Search... Help | Adva

Computer Science > Cryptography and Security

[Submitted on 20 Aug 2021 (v1), last revised 16 Dec 2021 (this version, v3)]

Asleep at the Keyboard? Assessing the Security of GitHub Copilot's Code Contributions

Hammond Pearce, Baleegh Ahmad, Benjamin Tan, Brendan Dolan-Gavitt, Ramesh Karri

There is burgeoning interest in designing AI-based systems to assist humans in designing computing systems, including tools that automatically generate computer code. The most notable of these comes in the form of the first self-described `AI pair programmer', GitHub Copilot, a language model trained over open-source GitHub code. However, code often contains bugs - and so, given the vast quantity of unvetted code that Copilot has processed, it is certain that the language model will have learned from exploitable, buggy code. This raises concerns on the security of Copilot's code contributions. In this work, we systematically investigate the prevalence and conditions that can cause GitHub Copilot to recommend insecure code. To perform this analysis we prompt Copilot to generate code in scenarios relevant to high-risk CWEs (e.g. those from MITRE's "Top 25" list). We explore Copilot's performance on three distinct code generation axes -- examining how it performs given diversity of weaknesses, diversity of prompts, and diversity of domains. In total, we produce 89 different scenarios for Copilot to complete, producing 1,689 programs. Of these, we found approximately 40% to be vulnerable.

ASKING IT TO FIX IT IS AS RELIABLE AS THE REST OF IT

arXiv > cs > arXiv:2405.12641

Search... Help | Adv

Computer Science > Software Engineering

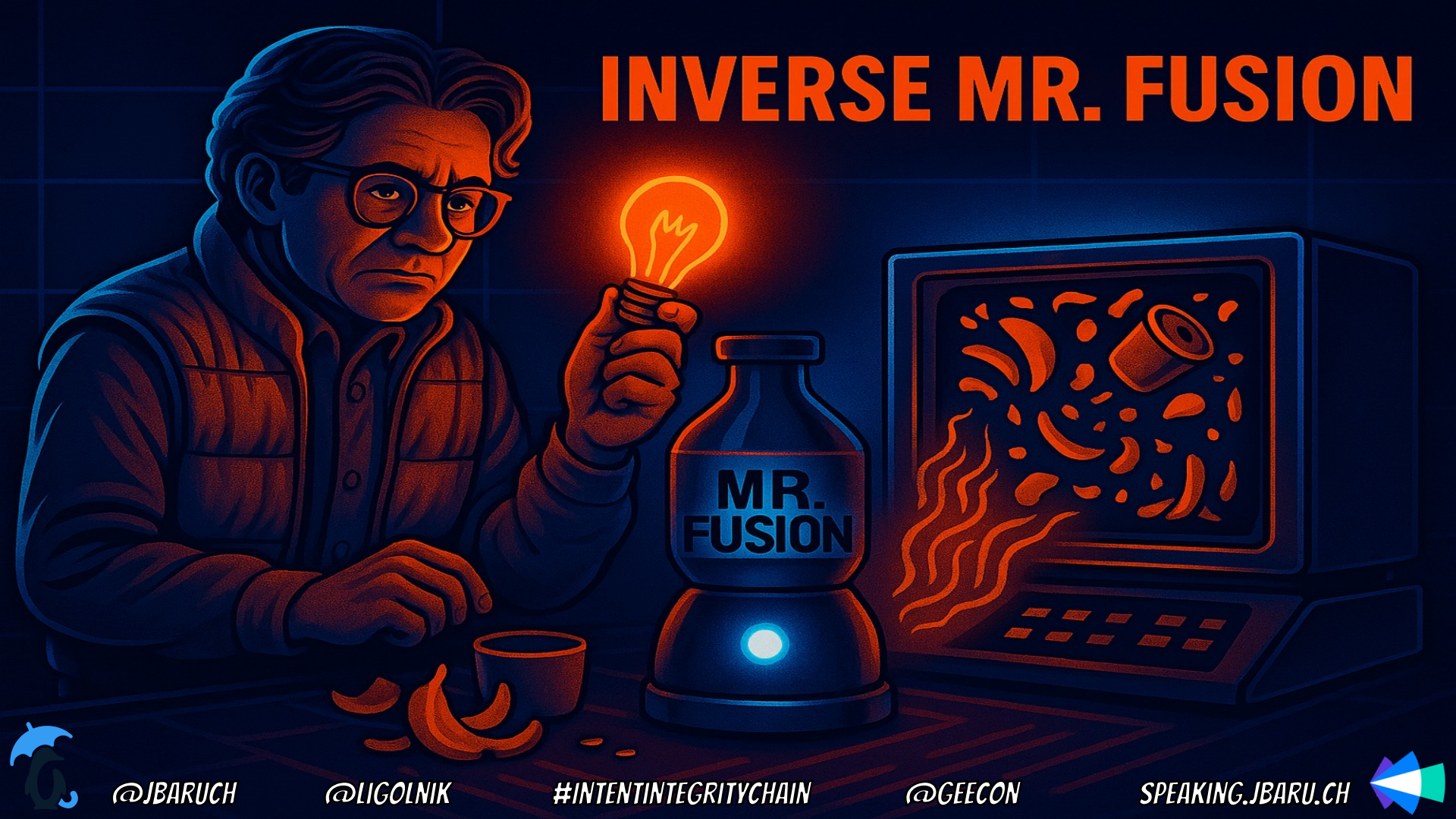
[Submitted on 21 May 2024 (v1), last revised 28 Nov 2024 (this version, v2)]

Fight Fire with Fire: How Much Can We Trust ChatGPT on Source Code-Related Tasks?

Xiao Yu, Lei Liu, Xing Hu, Jacky Wai Keung, Jin Liu, Xin Xia

With the increasing utilization of large language models such as ChatGPT during software development, it has become crucial to verify the quality of code content it generates. Recent studies proposed utilizing ChatGPT as both a developer and tester for multi-agent collaborative software development. The multi-agent collaboration empowers ChatGPT to produce test reports for its generated code, enabling it to self-verify the code content and fix bugs based on these reports. However, these studies did not assess the effectiveness of the generated test reports in validating the code. Therefore, we conduct a comprehensive empirical investigation to evaluate ChatGPT's self-verification capability in code generation, code completion, and program repair. We request ChatGPT to (1) generate correct code and then self-verify its correctness; (2) complete code without vulnerabilities and then self-verify for the presence of vulnerabilities; and (3) repair buggy code and then self-verify whether the bugs are resolved. Our findings on two code generation datasets, one code completion dataset, and two program repair datasets reveal the following observations. (1) ChatGPT often erroneously predicts its generated incorrect code as correct. (2) The self-contradictory hallucinations in ChatGPT's behavior arise. (3) The self-verification capability of ChatGPT can be enhanced by asking the guiding question, which queries whether ChatGPT agrees with assertions about incorrectly generated or repaired code and vulnerabilities in completed code. (4) Using test reports generated by ChatGPT can identify more vulnerabilities in completed code, but the explanations for incorrectly generated code and failed repairs are mostly inaccurate in the test reports. Based on these findings, we provide implications for further research or development using ChatGPT.

INVERSE MR. FUSION

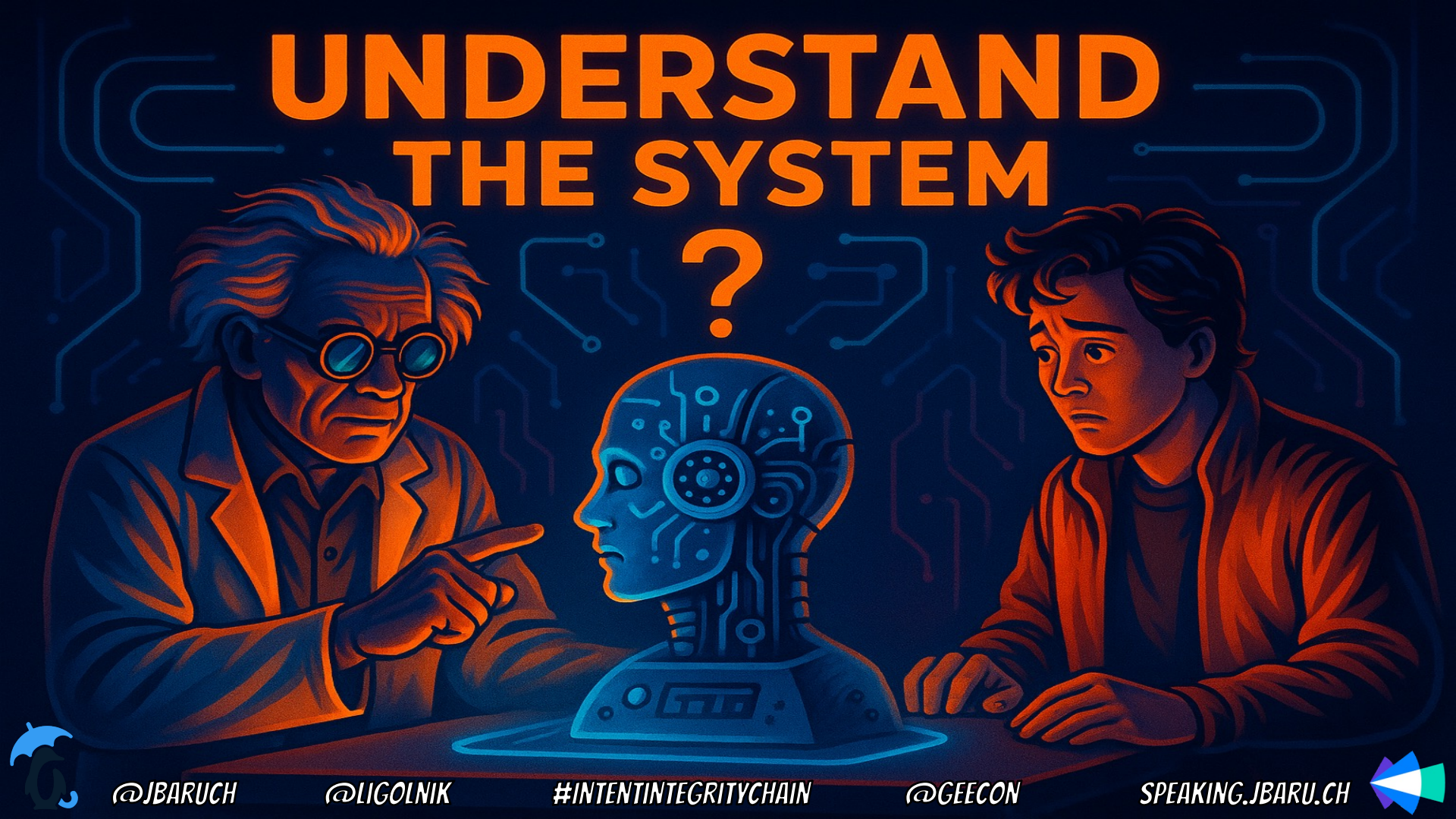


PANIC VS. OPPORTUNITY

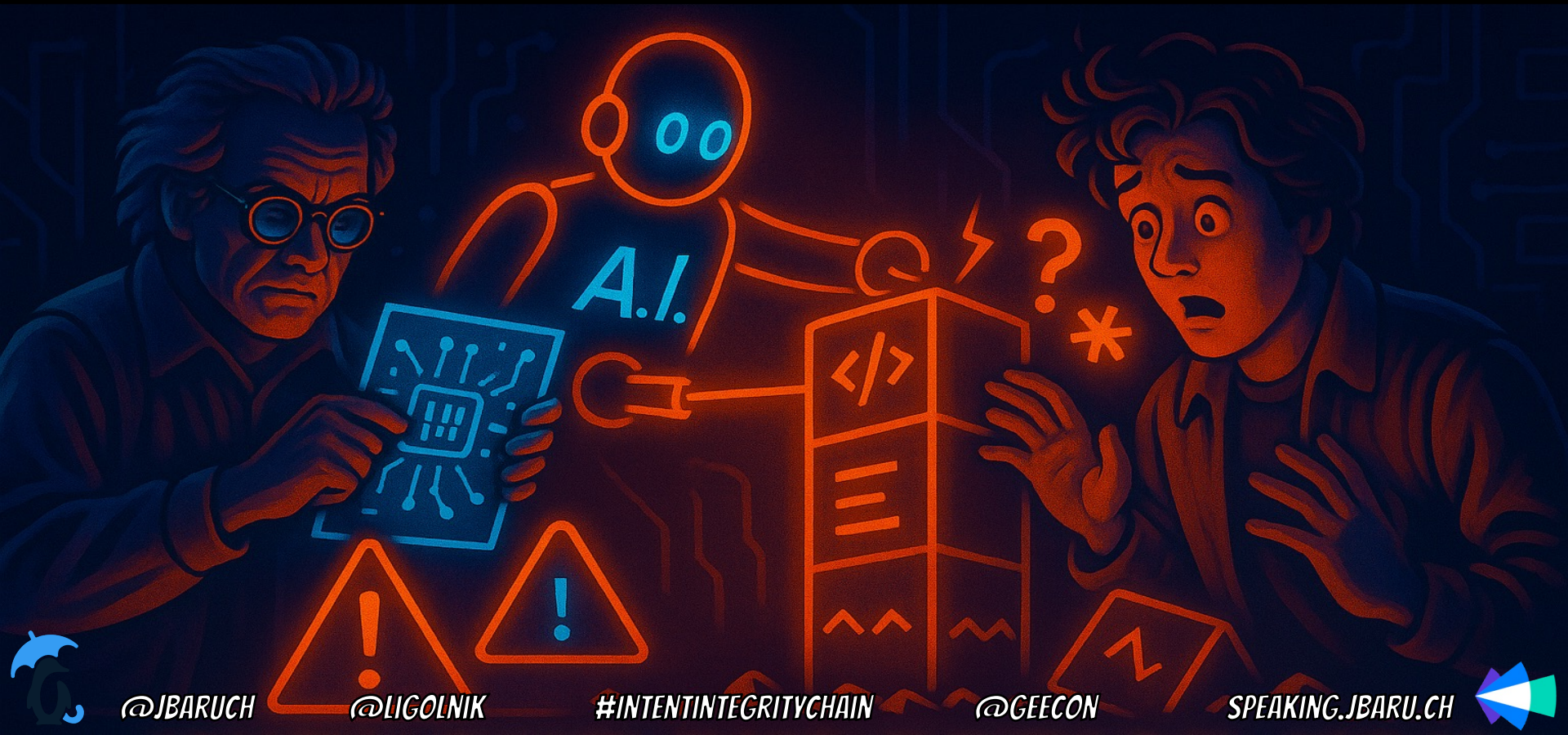
@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH

EVERY PROFESSION GETS A SIDEKICK

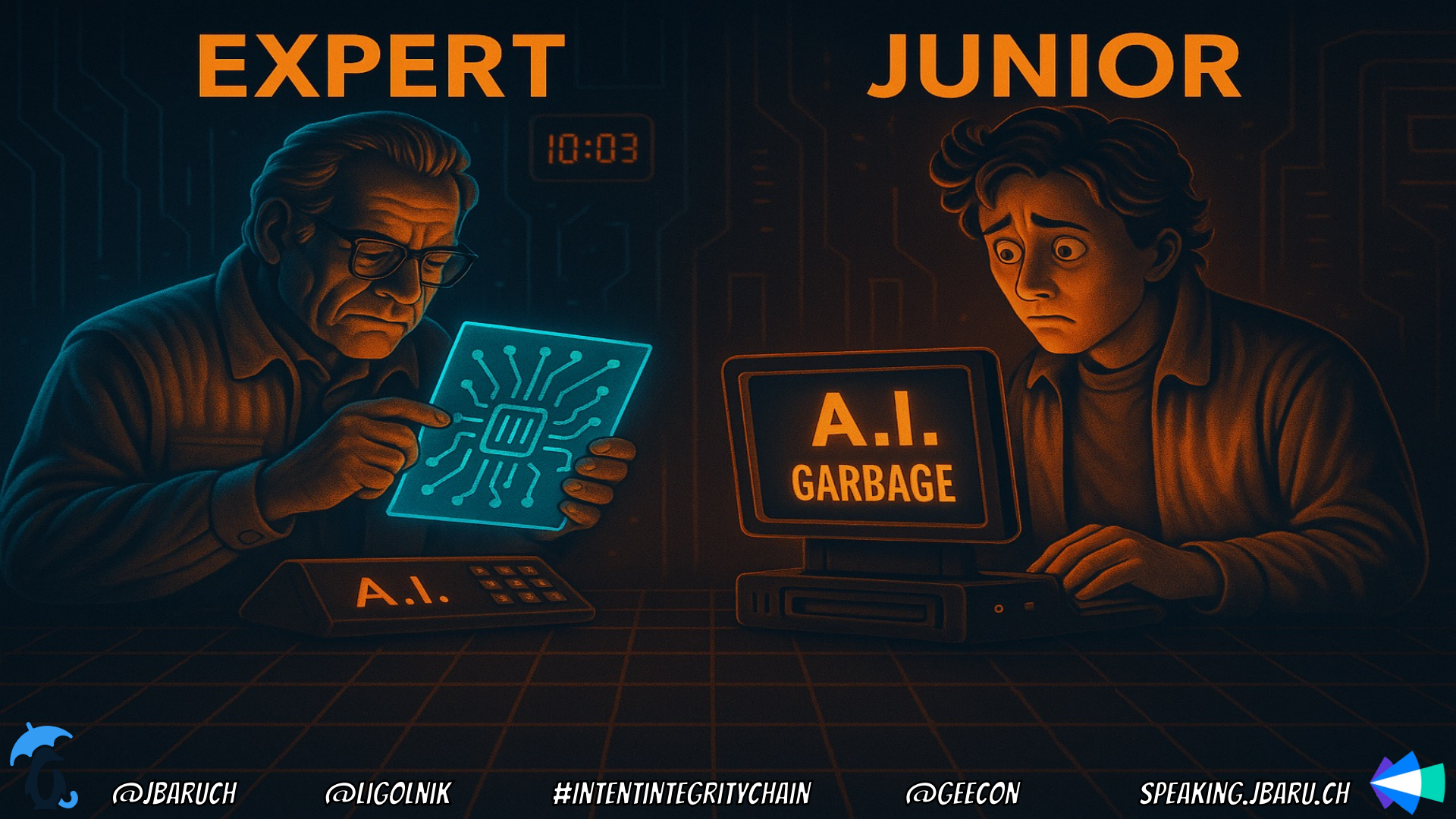
@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH


UNDERSTAND
THE SYSTEM
?
@JBARUCH
@LIGOLNIK
#INTENTINTEGRITYCHAIN
@GEECON
SPEAKING.JBARU.CH

WITHOUT EXPERTISE, YOU GET CHAOS—NOT CODE

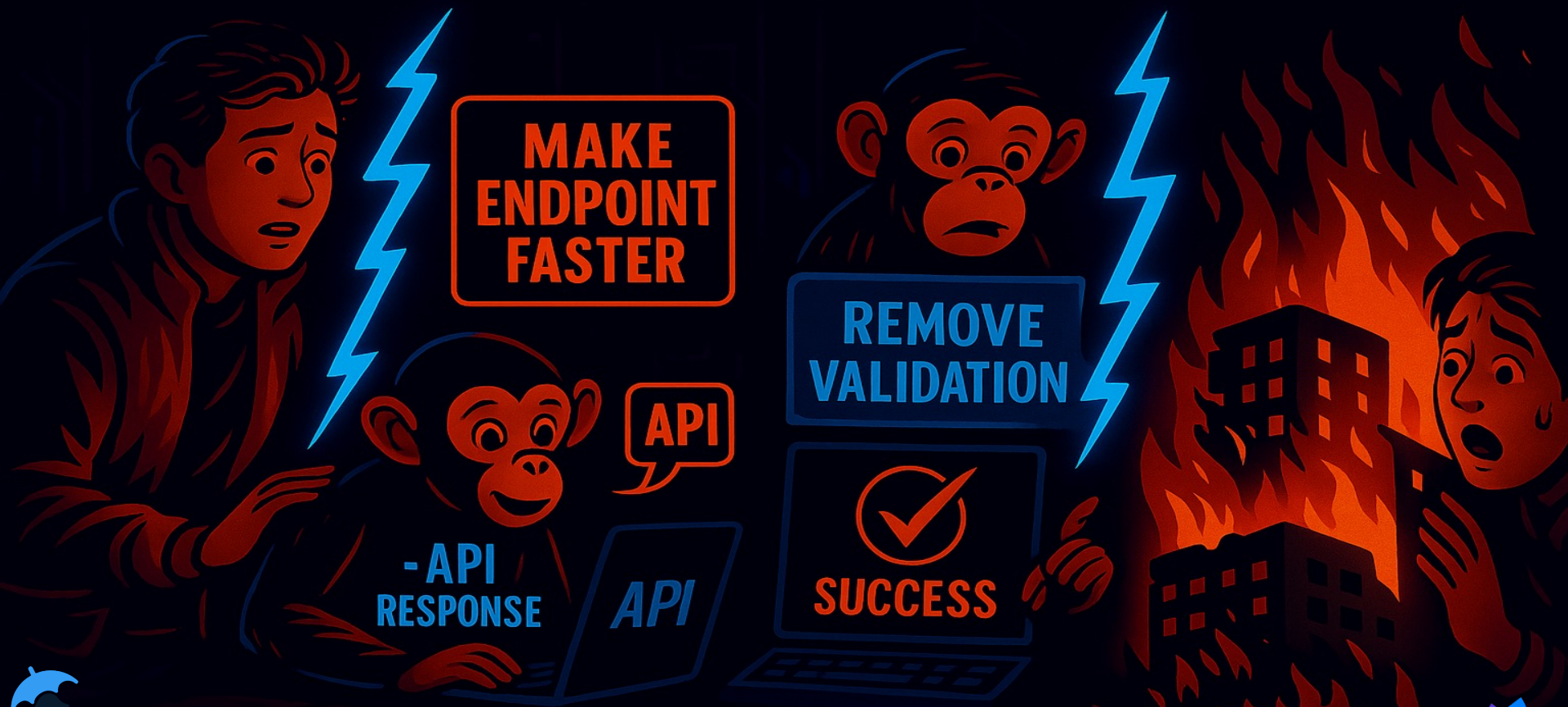


@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH

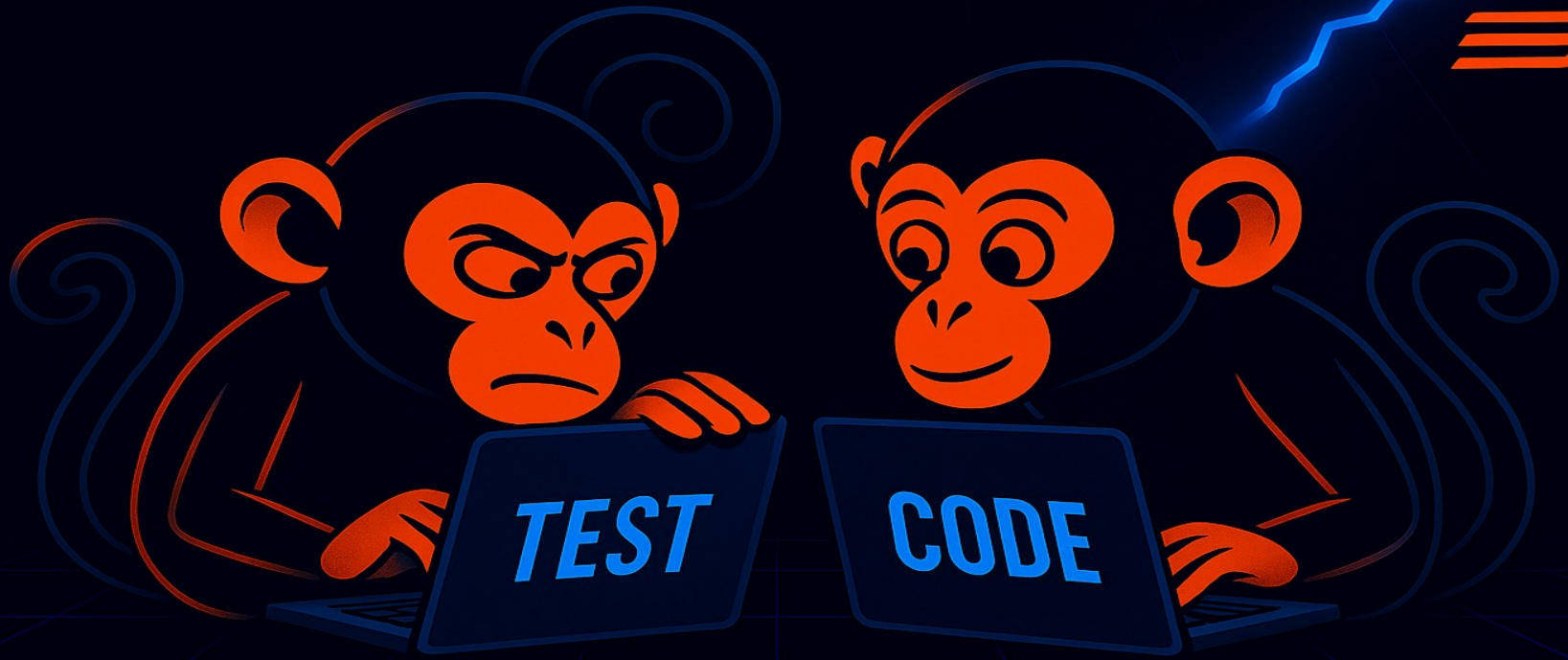

EXPERT
JUNIOR
10:03
A.I.
A.I.
GARBAGE
@JBARUCH
@LIGOLNIK
#INTENTINTEGRITYCHAIN
@GEECON
SPEAKING.JBARU.CH

A CAUTIONARY TALE

FASTER? SURE. JUST DON'T ASK WHAT IT COST.

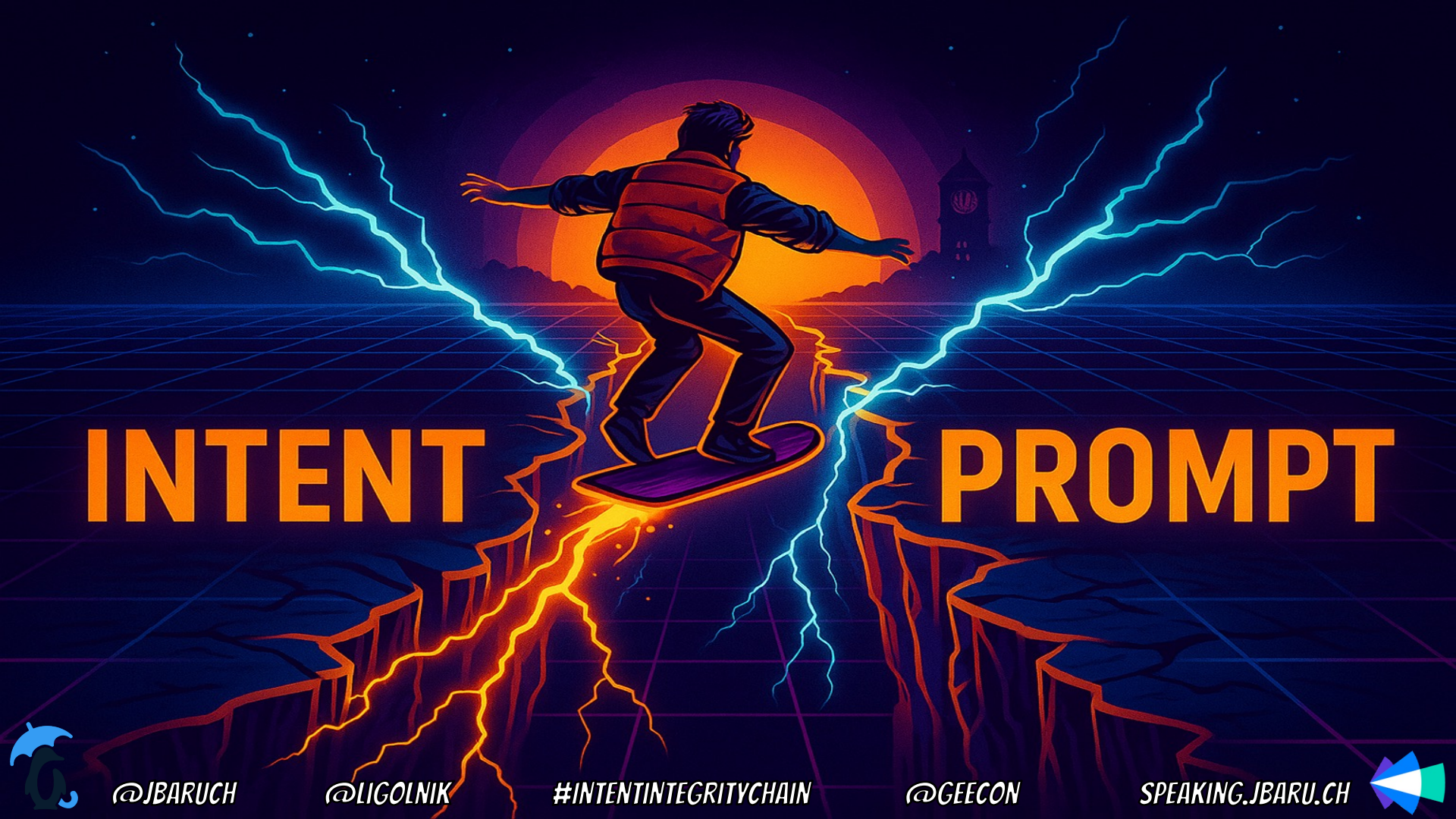


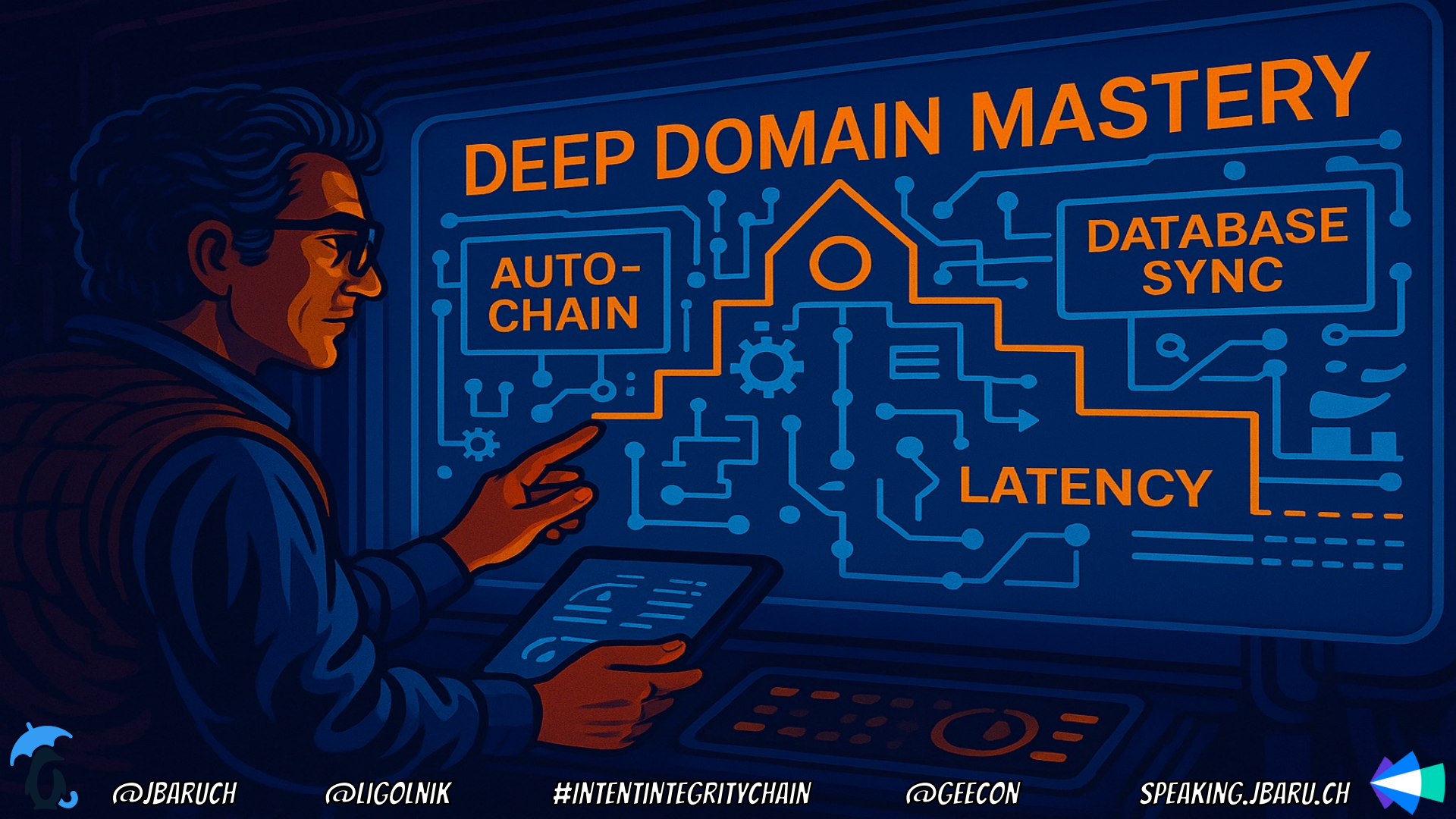
CIRCULAR VERIFICATION

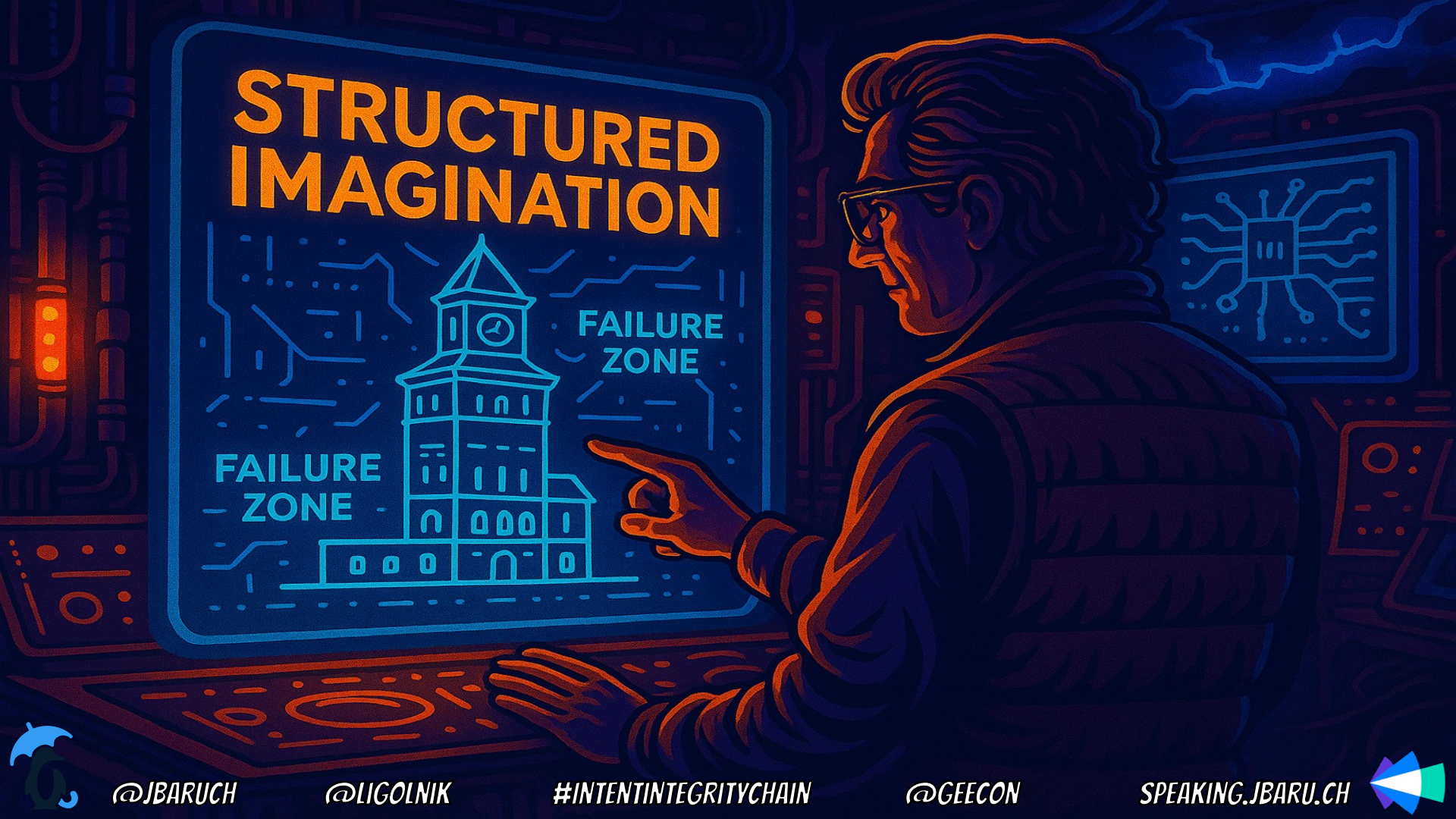



PROMPT
SUCCESSFUL

HE THOUGHT IT WAS OBVIOUS.


INTENT
PROMPT
@JBARUCH
@LIGOLNIK
#INTENTINTEGRITYCHAIN
@GEECON
SPEAKING.JBARU.CH


DEEP DOMAIN MASTERY
AUTO-
CHAIN
DATABASE
SYNC
LATENCY
@JBARUCH
@LIGOLNIK
#INTENTINTEGRITYCHAIN
@GEECON
SPEAKING.JBARU.CH


STRUCTURED IMAGINATION
FAILURE ZONE
FAILURE ZONE

GUARDRAILS FOR SAFETY

@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH

THE ART OF POSSIBLE

DOMAIN MASTERY

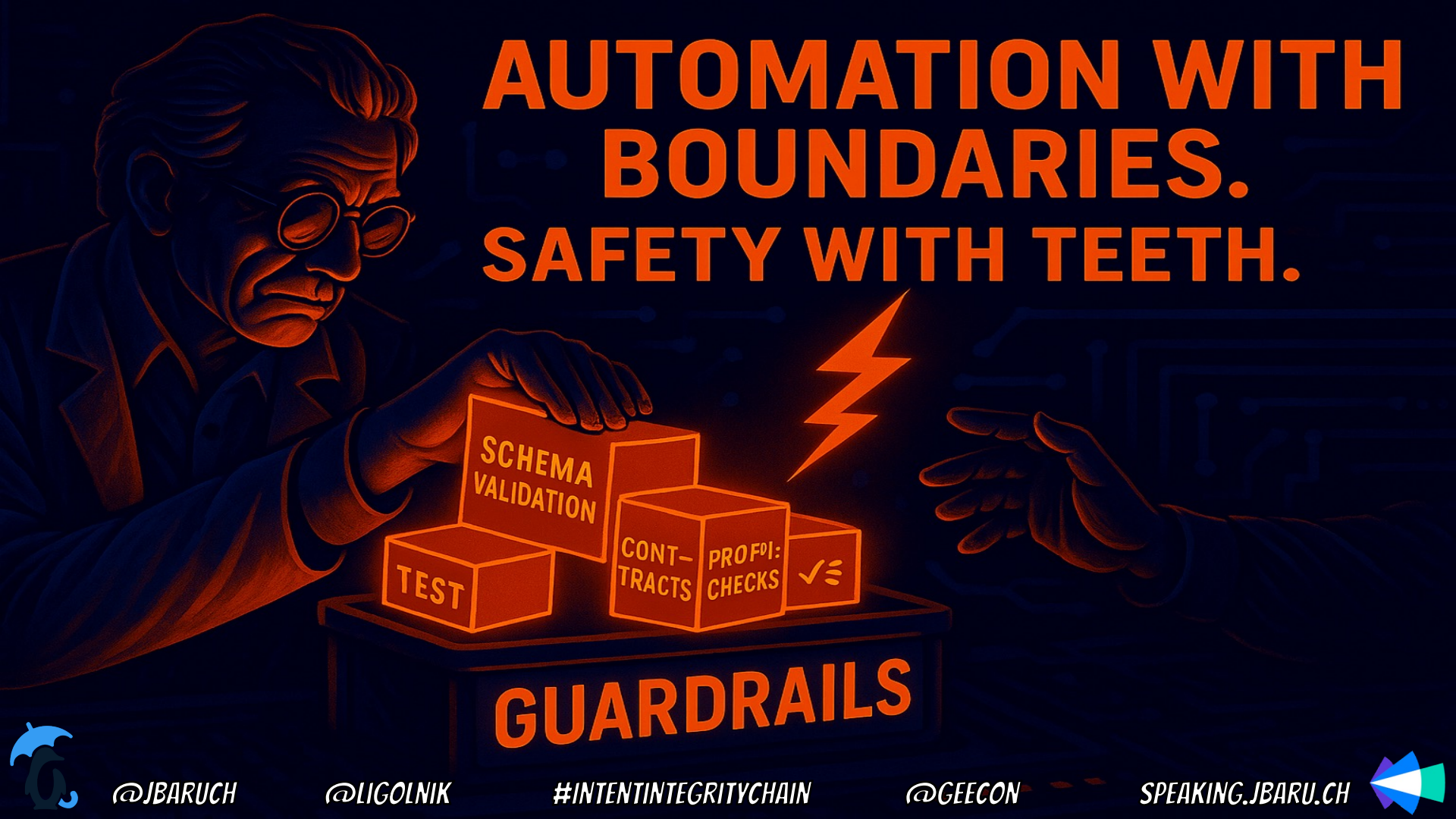
STRUCTURED IMAGINATION

PURPOSEFUL GUARDRAILS

@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH

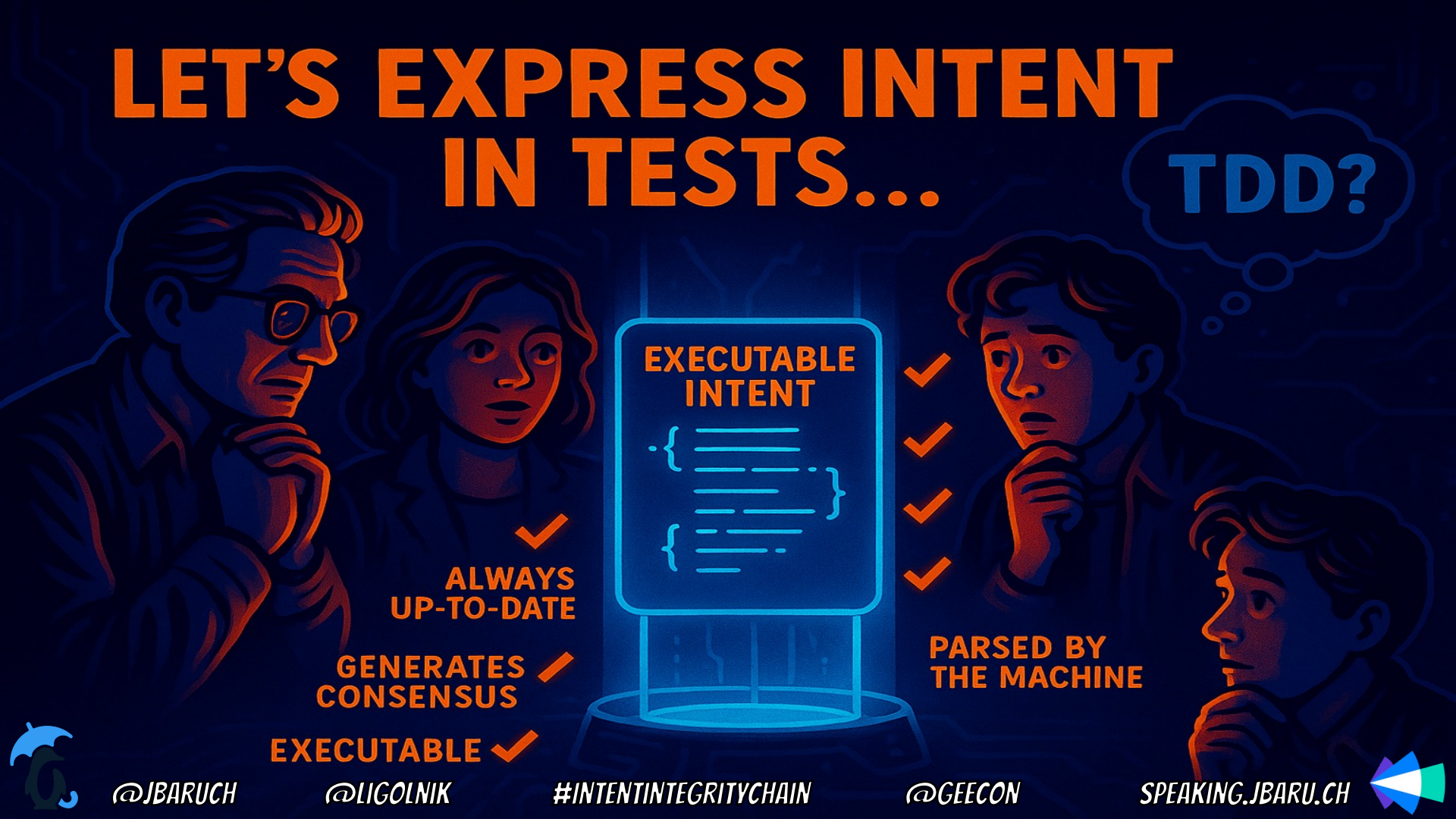
AUTOMATION WITH BOUNDARIES.

SAFETY WITH TEETH.



AI NEEDS STRUCTURE TO BEHAVE

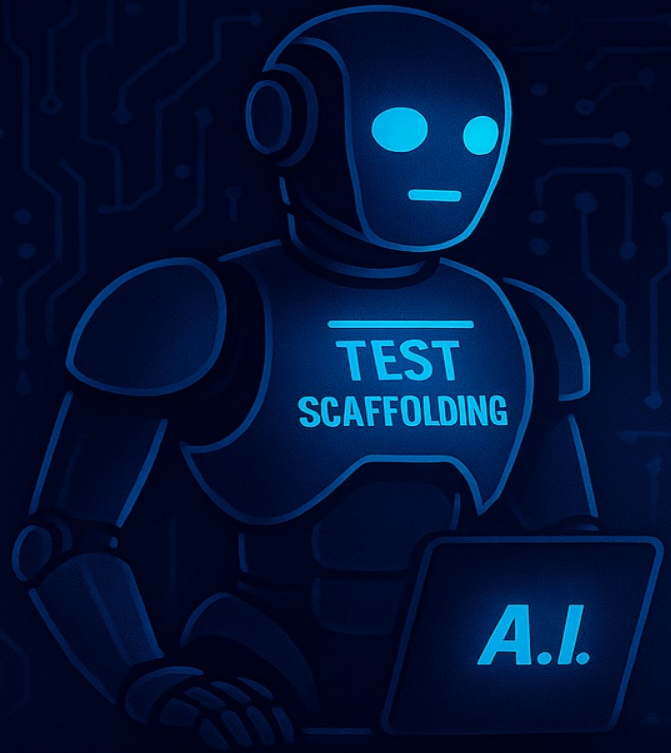
@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH


LET'S EXPRESS INTENT IN TESTS...
TDD?
EXECUTABLE INTENT
ALWAYS UP-TO-DATE
GENERATES CONSENSUS
EXECUTABLE
PARSED BY THE MACHINE
@JBARUCH
@LIGOLNIK
#INTENTINTEGRITYCHAIN
@GEECON
SPEAKING.JBARU.CH

DEVELOPERS ARE BIASED FOR ACTION

TEST-FIRST. WITHOUT TRYING.

DISCIPLINE BECOMES A SIDE EFFECT

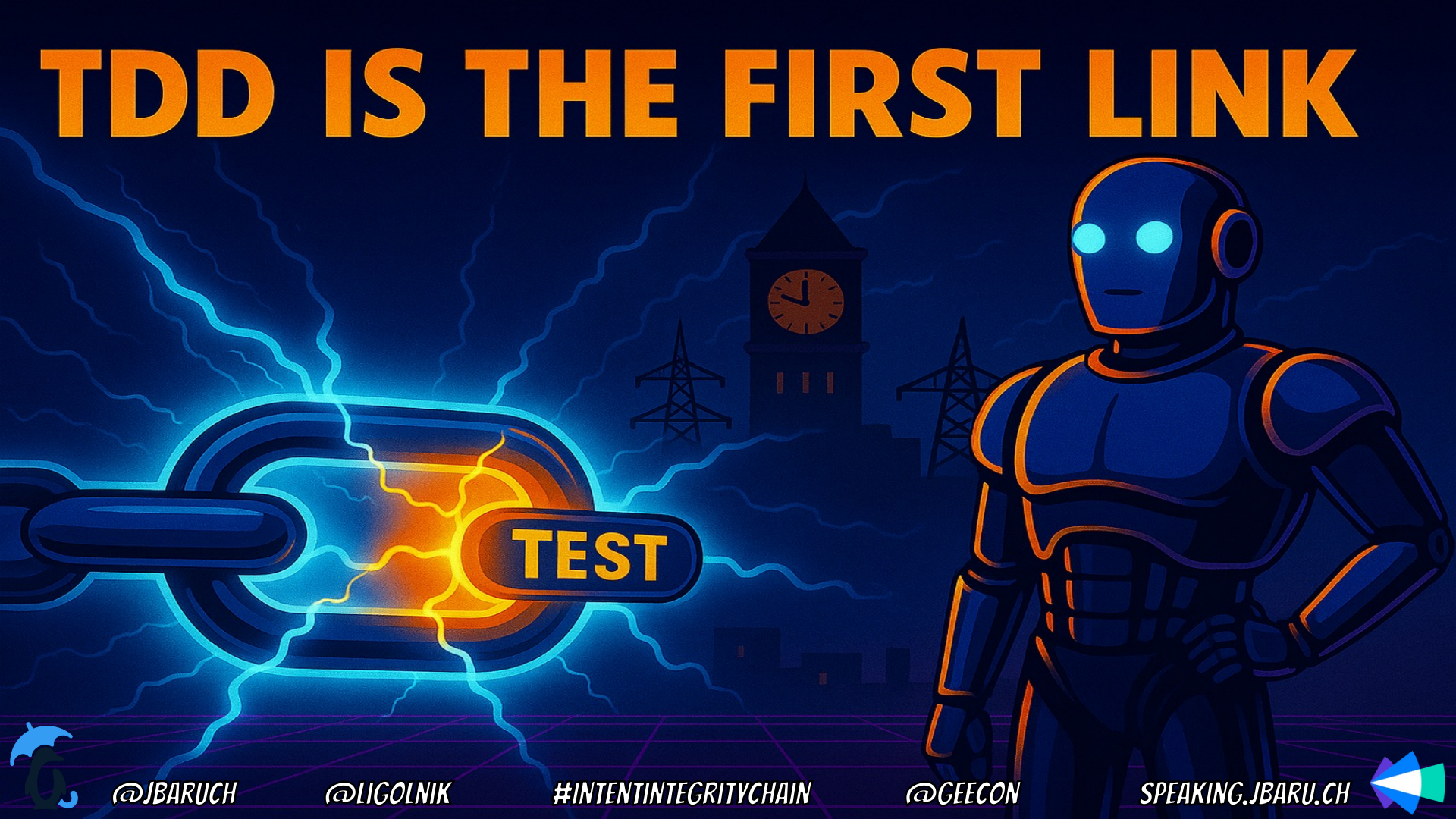


PASS

PASS

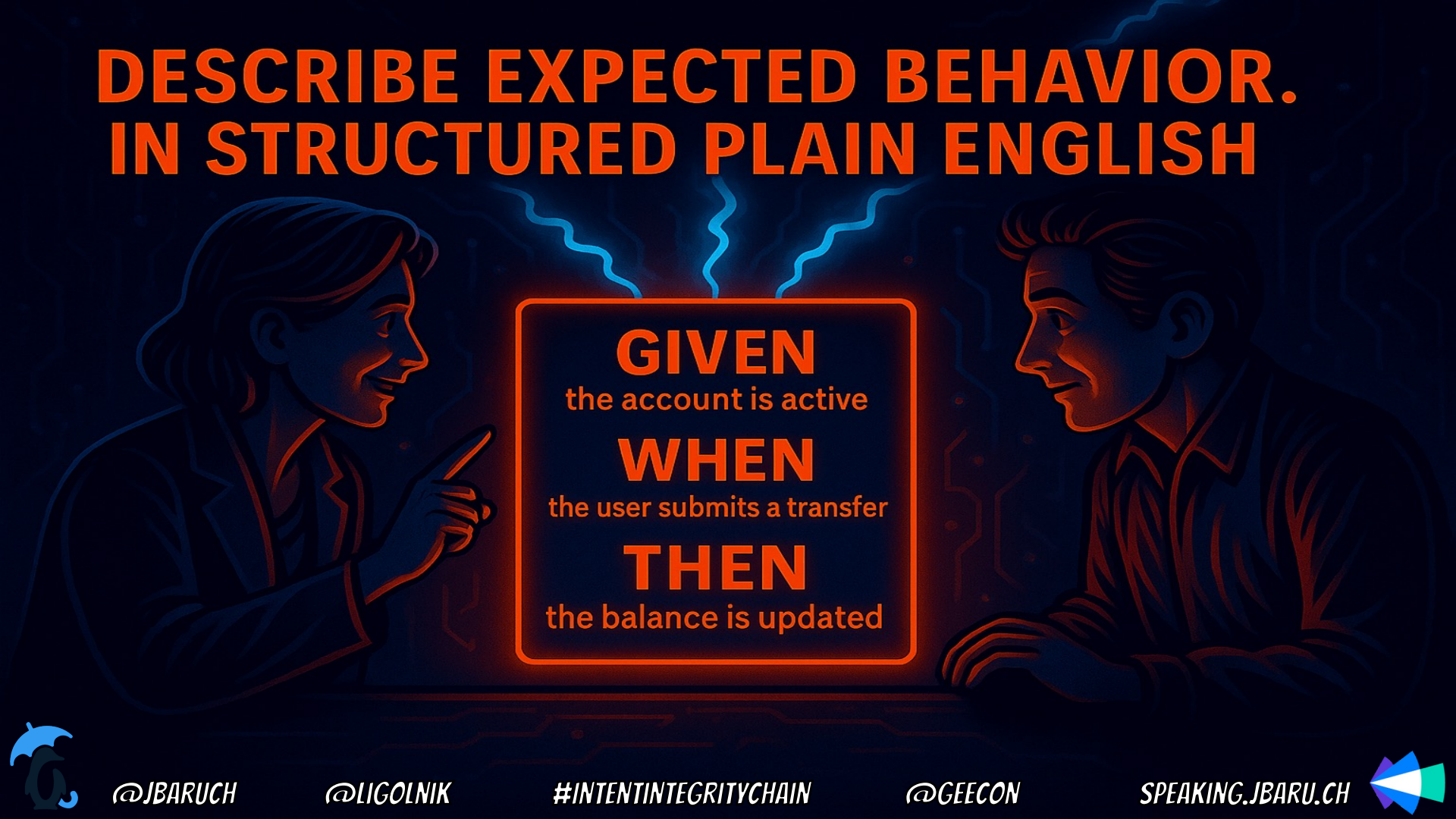
PASS

TDD IS THE FIRST LINK



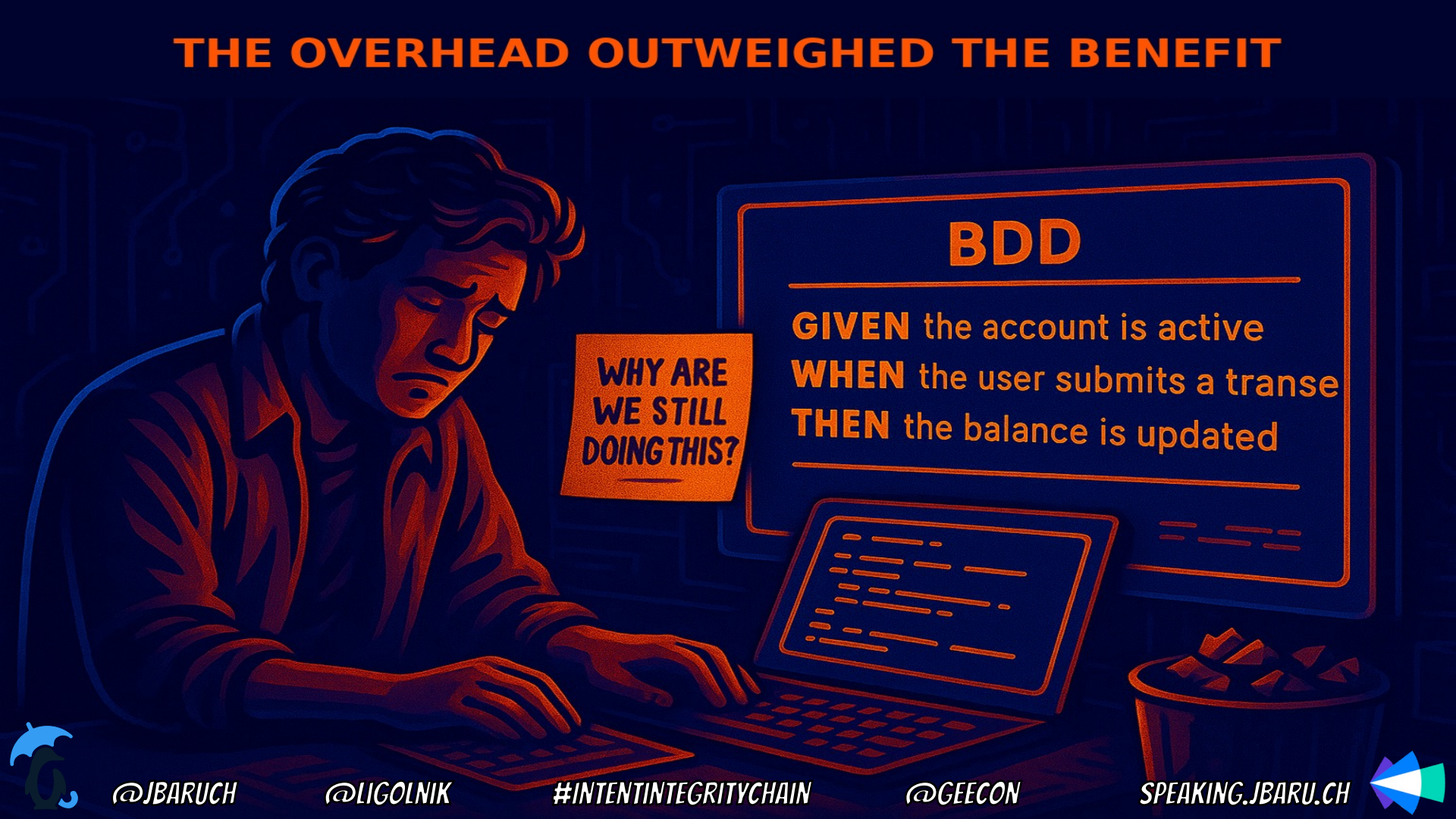
THEY DIDN'T REVIEW THE TESTS... BECAUSE THEY COULDN'T READ THEM

DESCRIBE EXPECTED BEHAVIOR. IN STRUCTURED PLAIN ENGLISH



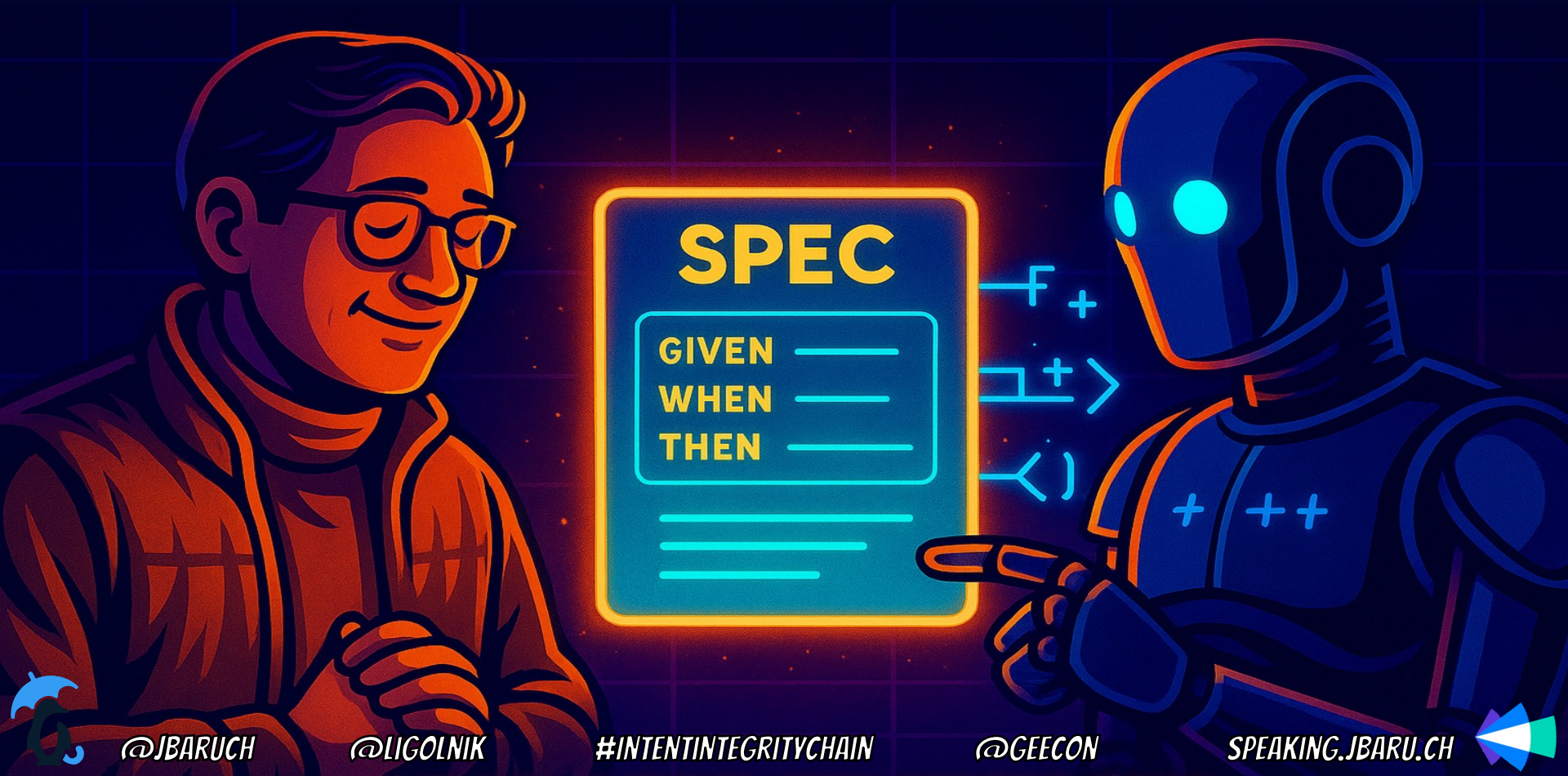
BDD REQUIRED DISCIPLINE AND SYNTAX

THE OVERHEAD OUTWEIGHED THE BENEFIT



WHO NEEDS GHERKIN WHEN YOU CAN VIBECODE?

READABLE BY HUMANS. EXECUTABLE BY MACHINES.



INTENT INTEGRITY CHAIN

PROMPT — SPEC — TEST — CODE

:: PDD BDD TDD

@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH

INTENT INTEGRITY CHAIN

- HUMANS WRITE PROMPT
- MONKEYS GENERATE SPEC
- MACHINE CREATES TESTS
- MONKEYS WRITE CODE

NEVER TRUST A MONKEY

@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH

DETERMINISTIC TEST GENERATION

PROMPT

SPEC

ARTIFACT	CREATED	CAN WE TRUST IT?
PROMPT	👱	N/A
SPEC	🐒	❌
TEST	🤖	✅
CODE	🐒	❌

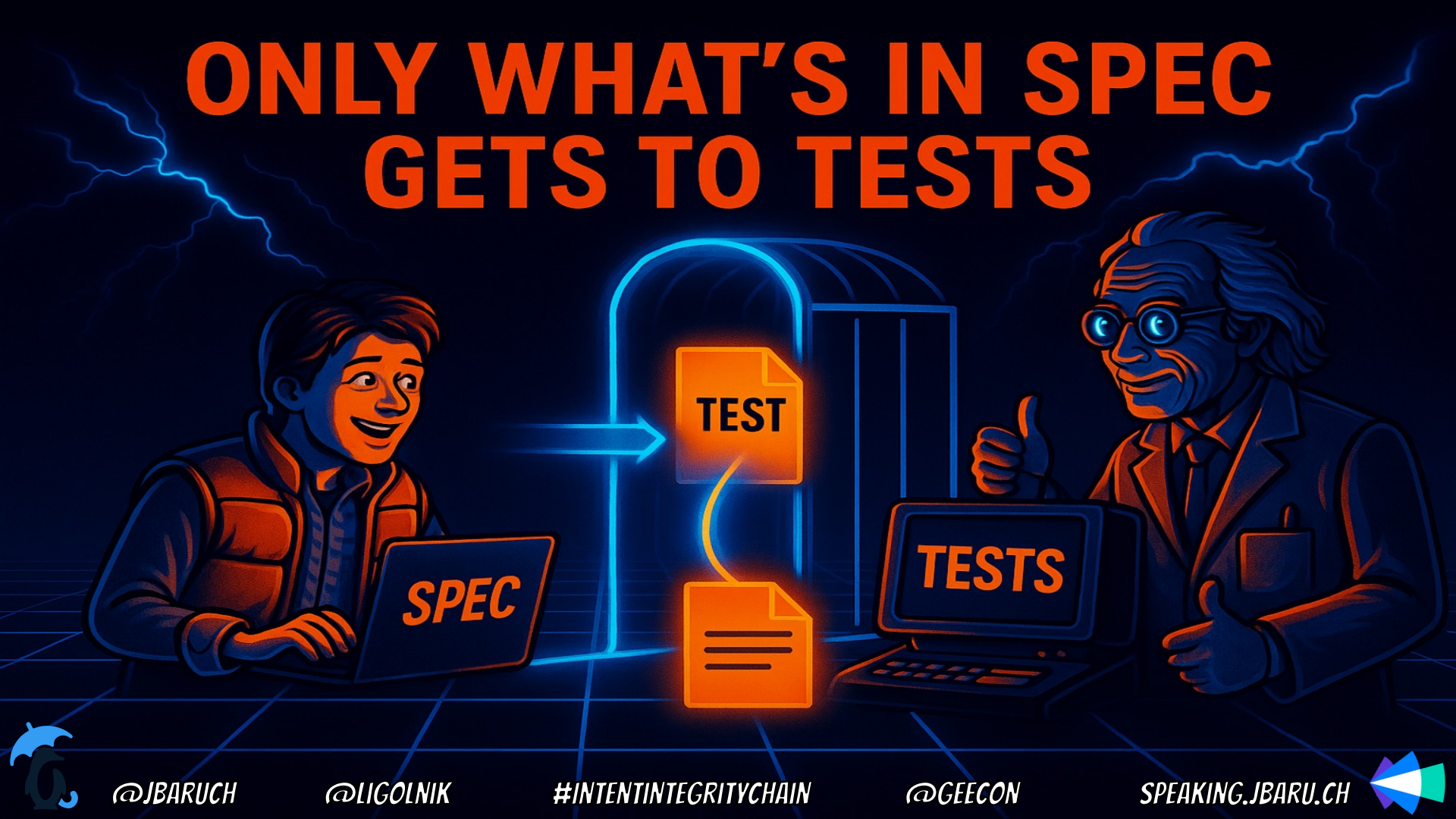
TESTS ARE LOCKED

ARTIFACT	CREATED	VALIDATED
PROMPT	👱	N/A
SPEC	🐒	👱
TEST	🤖	🤖
CODE	🐒	🤖

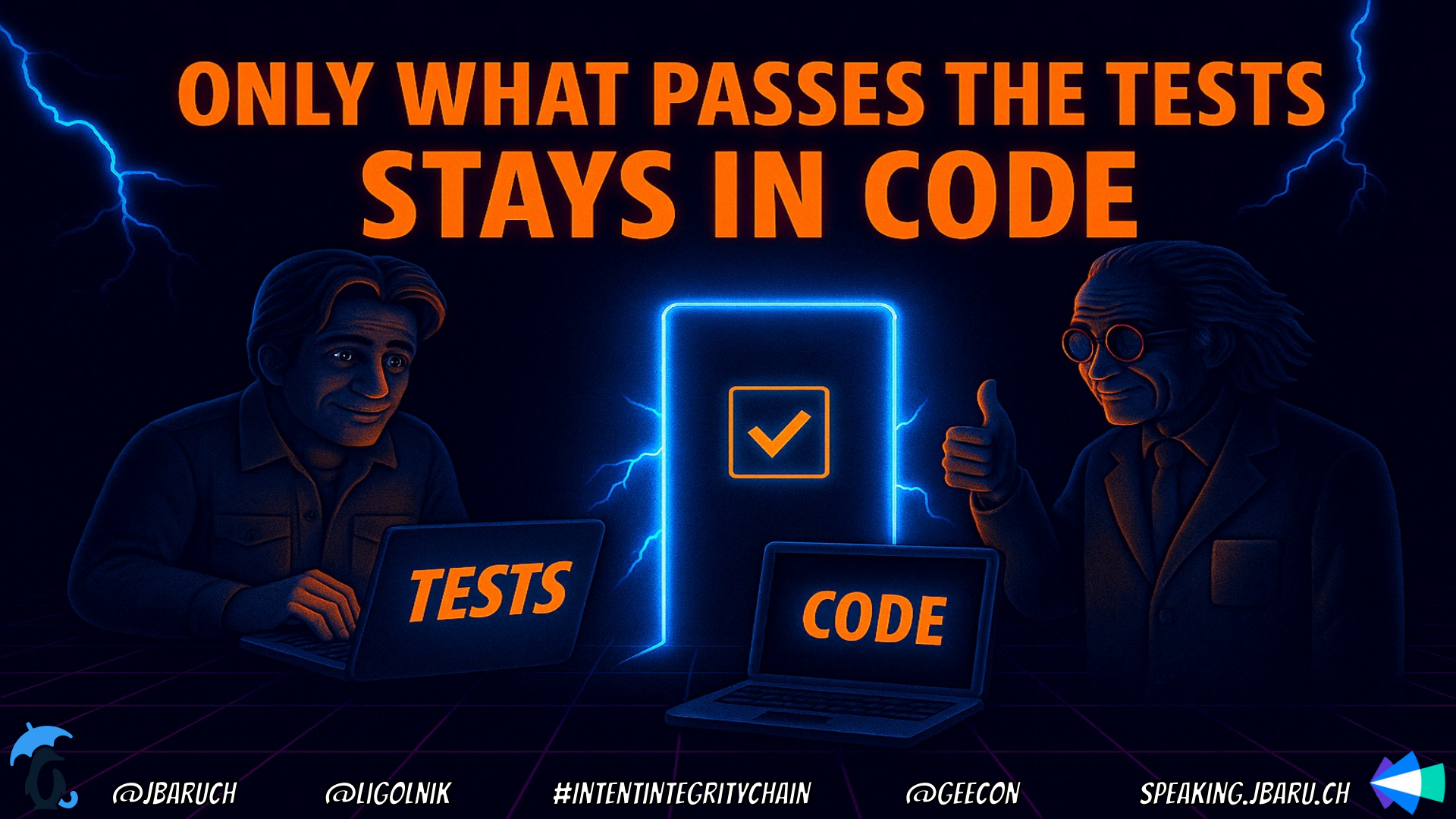
SHARED UNDERSTANDING IS NOW EXECUTABLE

@JBARUCH @LIGOLNIK #INTENTINTEGRITYCHAIN @GEECON SPEAKING.JBARU.CH

ONLY WHAT'S IN SPEC GETS TO TESTS



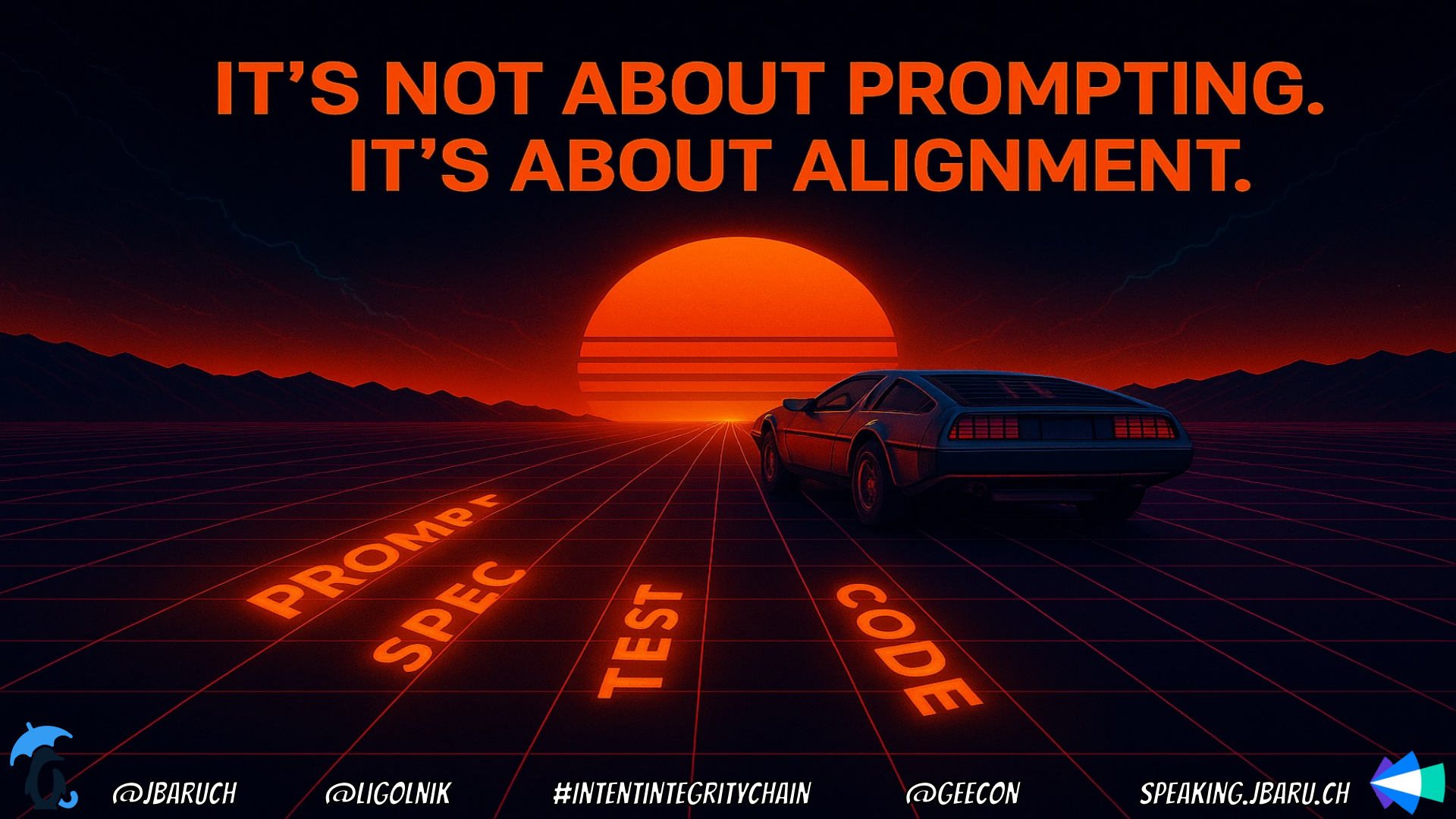
ONLY WHAT PASSES THE TESTS STAYS IN CODE



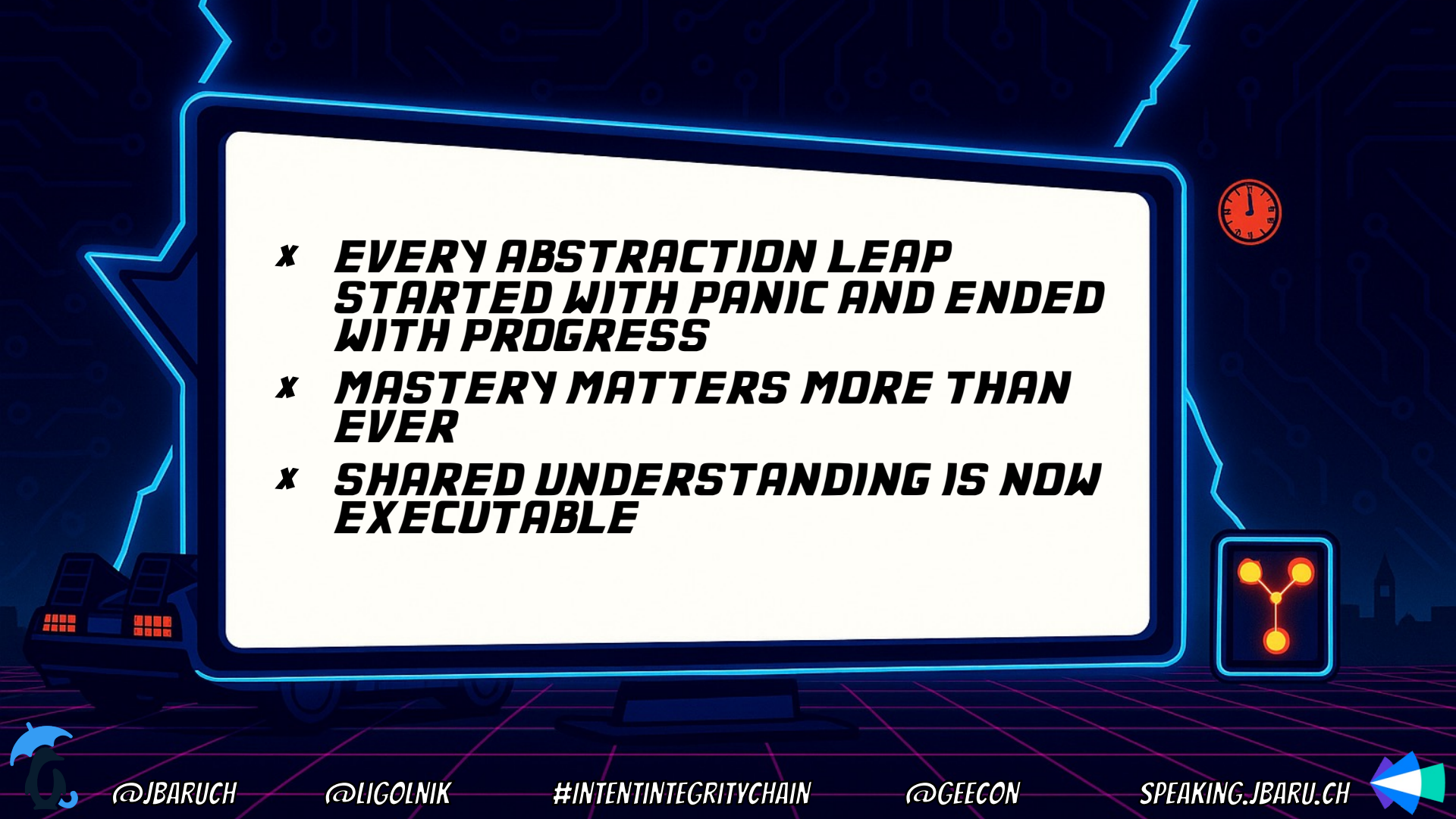
WATCH THE DEMO:

ARTIFACT	CREATED	VALIDATED
PROMPT	👱	N/A
SPEC	🐒	👱
TEST	🤖	🤖
CODE	🐒	🤖


Andrej Karpathy
@karpathy
The hottest new programming language is English
9:14 PM · Jan 24, 2023 · 7.3M Views
1.1K
6.9K
44K
4.2K
@JBARUCH
@LIGOLNIK
#INTENTINTEGRITYCHAIN
@GEECON
SPEAKING.JBARU.CH


IT'S NOT ABOUT PROMPTING.
IT'S ABOUT ALIGNMENT.
PROMPT
SPEC
TEST
CODE
@JBARUCH
@LIGOLNIK
#INTENTINTEGRITYCHAIN
@GEECON
SPEAKING.JBARU.CH

- x EVERY ABSTRACTION LEAP STARTED WITH PANIC AND ENDED WITH PROGRESS
- x MASTERY MATTERS MORE THAN EVER
- x SHARED UNDERSTANDING IS NOW EXECUTABLE



THANK YOU!

- @JBARUCH
- @LIGOLNIK
- #INTENTINTEGRITYCHAIN
- SPEAKING.JBARU.CH