CONFIDENTIAL

The NextGen Software Environment

Reliable AI Coding



From fragile AI-code
... to secure and maintainable enterprise software

"Container & logistics" as a spot-on analogy...



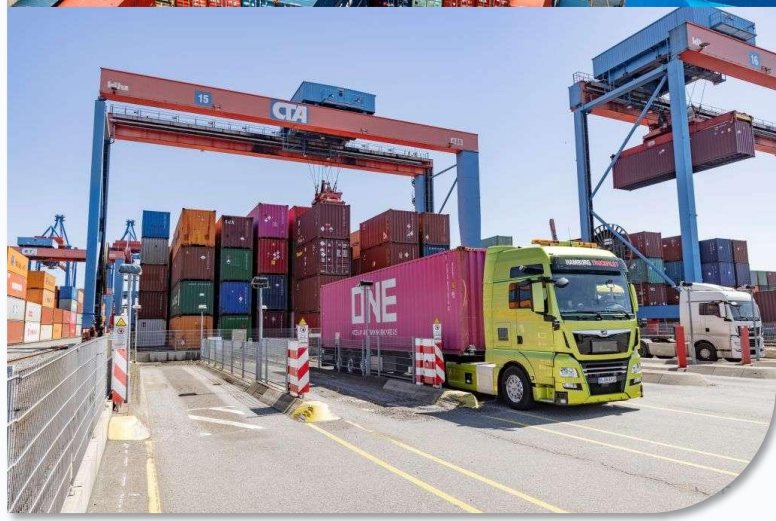
1968

ISO-668

Standardization

- Security
- Efficiency
- Transparency
- Compliance





In AI Software Generation we are in 1968.

PROBLEM

Security

- **+70% to +400% increased chance of security risk [1,2]**
- **4.4 M USD avg. cost of breach [3]**
- **At the same time:**
  - Increased trust in AI vs. Human Code [2,4]
  - Increasing hacking frequency [5]

Efficiency

- **Output explodes, flow does not [6]:**
  - +98 % pull requests,
  - +154 % PR size,
  - +91 % review time,
  - +9 % bugs
- **19 % slower with AI tools, yet believe to be 20 % faster [7]**

Transparency & Maintainability

- **+41 % complexity & +30 % static warnings stay [8]**
- **88 % of devs see AI raising tech debt [9]**
- **A third of developer time already goes to tech debt [10]; flaws live 8.5 months [11]**

Legal Risk

- **2026/27: CRA €15 M / 2.5 %[12],**
- **NIS2 €10 M / 2 % + personal liability[13]**
- **GDPR €20 M / 4 %[14]**
- **42 % of cyber policies already exclude AI-related incidents [15]**

At the moment...

Firms are not increasing speed.
Firms are "taking a mortgage".

So why not:

"Clde, please make it safe, efficient, transparent & compliant"**

?

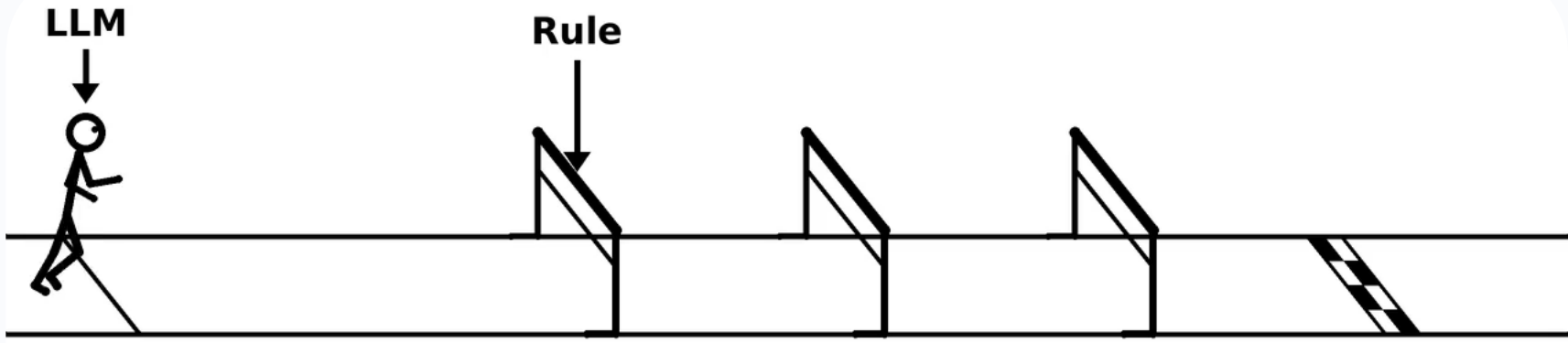
How to ensure rules (hurdles) are satisfied?

The ongoing status-quo

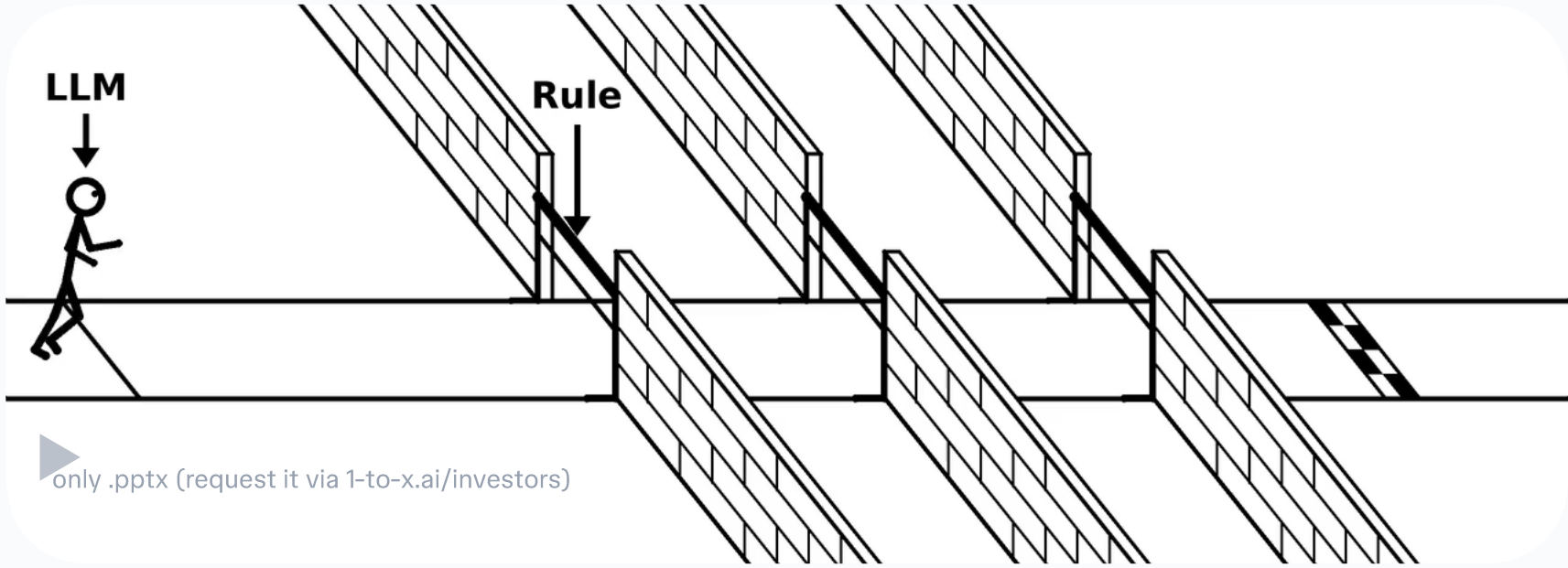
harness.md & abstract conventions

With 1-to-X semi-determinism

provable "correctness" & running-against-a-wall-principle



only .pptx (request it via 1-to-x.ai/investors)



only .pptx (request it via 1-to-x.ai/investors)

SOLUTION

Security

Secure by structure, not by prompt

Efficiency

Provable checks within seconds; Making review loops standardized.

Transparency & Maintainability

Maintaining structure also means stopping complexity from compounding

Compliance

Evidence via transparent logging of relevant actions & decisions.

CONFIDENTIAL

Why doesn't the Silicon Valley do it then?

Cui bono? Chaos is the business model

The Valley sells the generator and the antidote. Every unstructured line of code is a future security seat, review agent, or token.

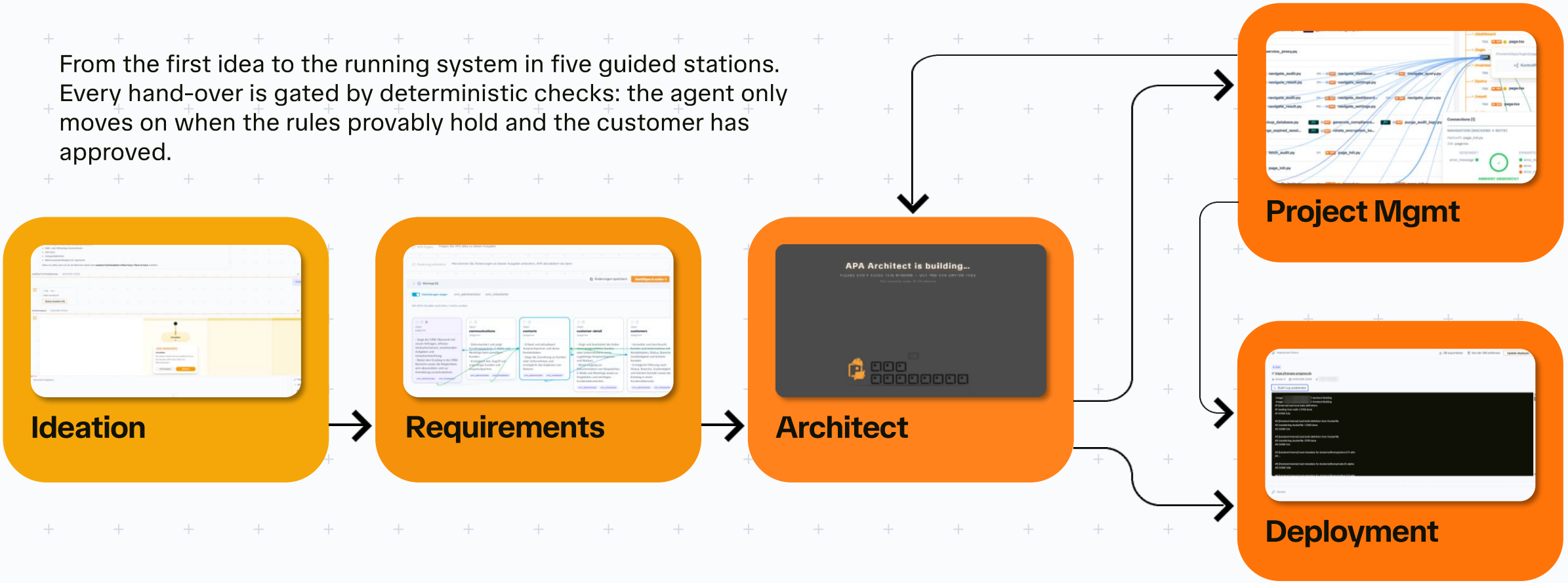
2 years of engineering. Patent pending.

2 years ago, we already saw the pitfalls of generic agent-loops coming up. Now we are ready for companies that neither want a black-box tool or high dependency on Big Tech because they don't understand their products anymore. With us they sit in the driver seat again. Being faster than ever!

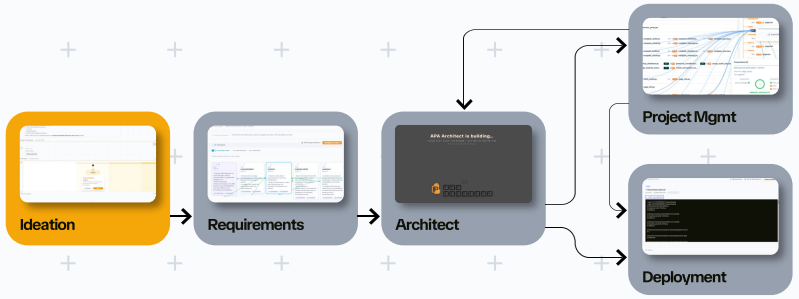
CONFIDENTIAL

One pipeline. One enforced structure.

From the first idea to the running system in five guided stations. Every hand-over is gated by deterministic checks: the agent only moves on when the rules provably hold and the customer has approved.



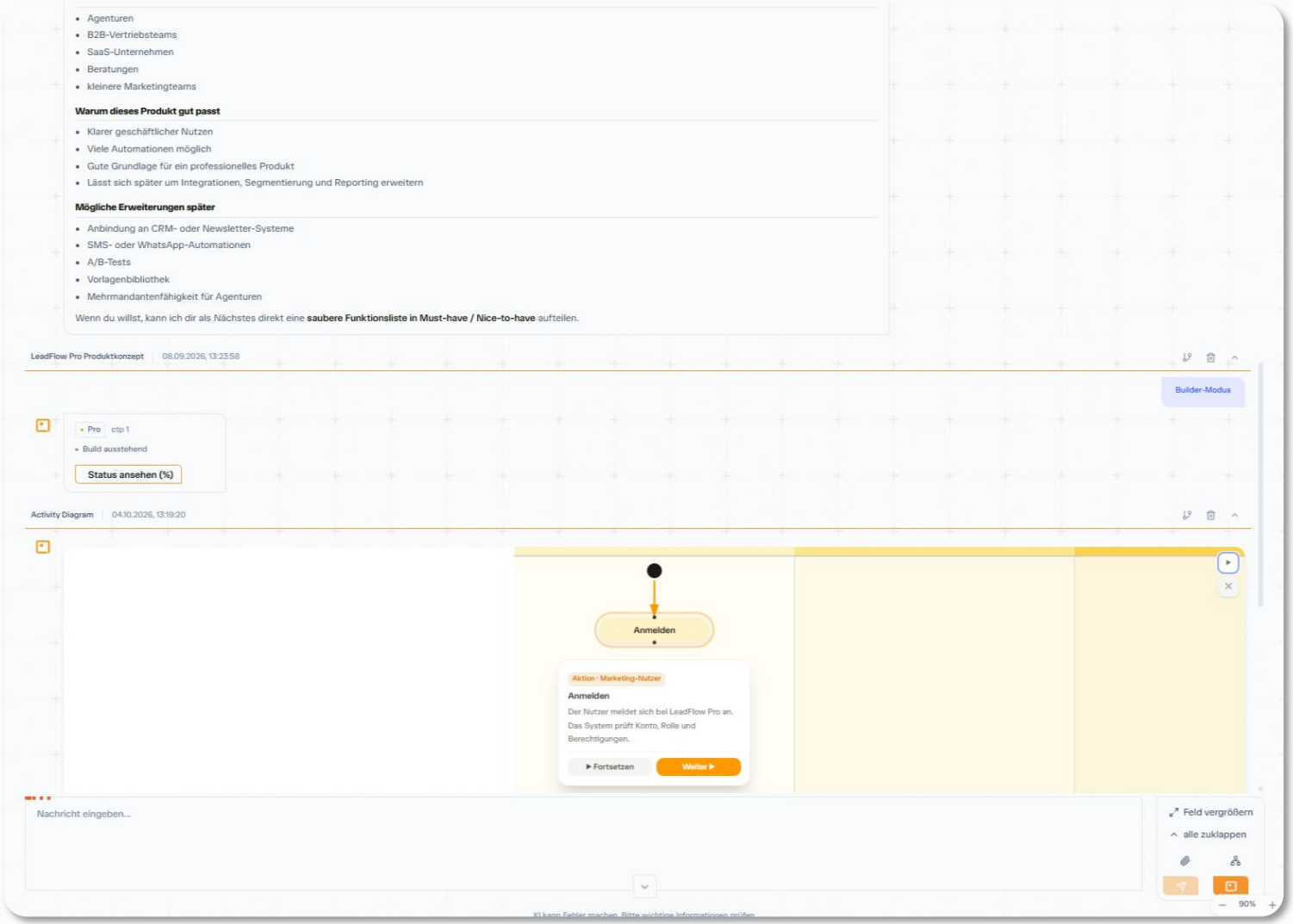
CONFIDENTIAL



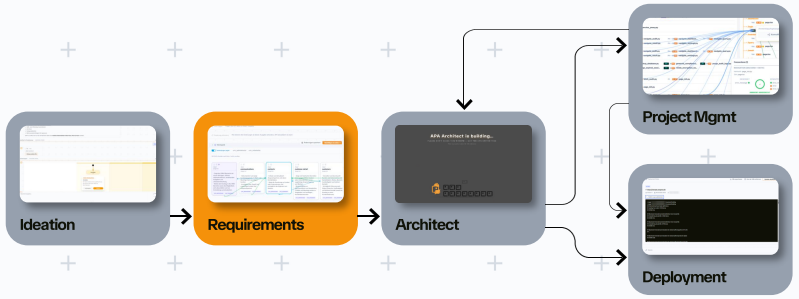
Interactive Ideation

Discuss raw ideas in a highly interactive chat.

- No technical knowledge required
- Explore alternatives side by side
- Your process, visualised as a diagram
- Works with your own documents



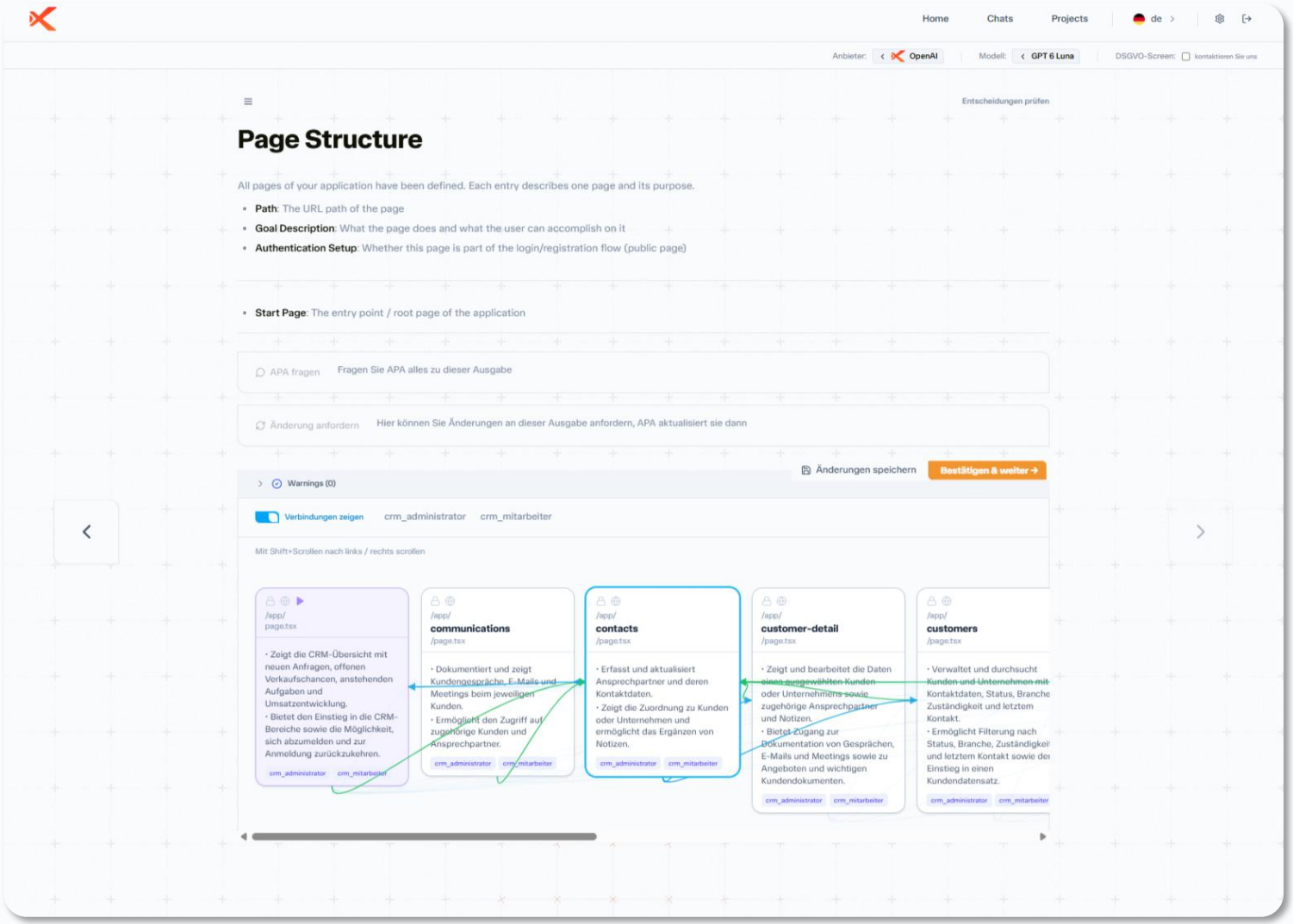
CONFIDENTIAL



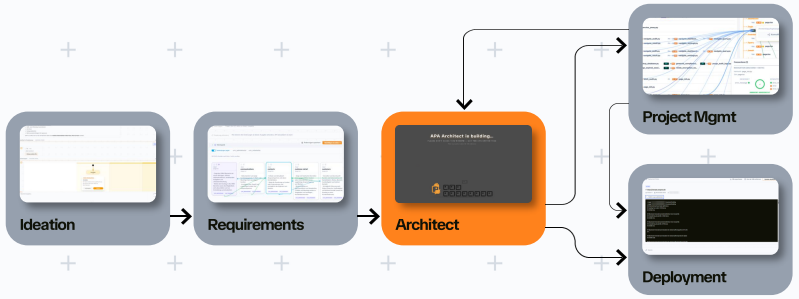
Requirements Analysis

Guided questions turn a rough idea into an approved blueprint.

- Guided questions instead of prompt guessing
- You approve every step
- Change requests at any point
- Your corporate design from day one



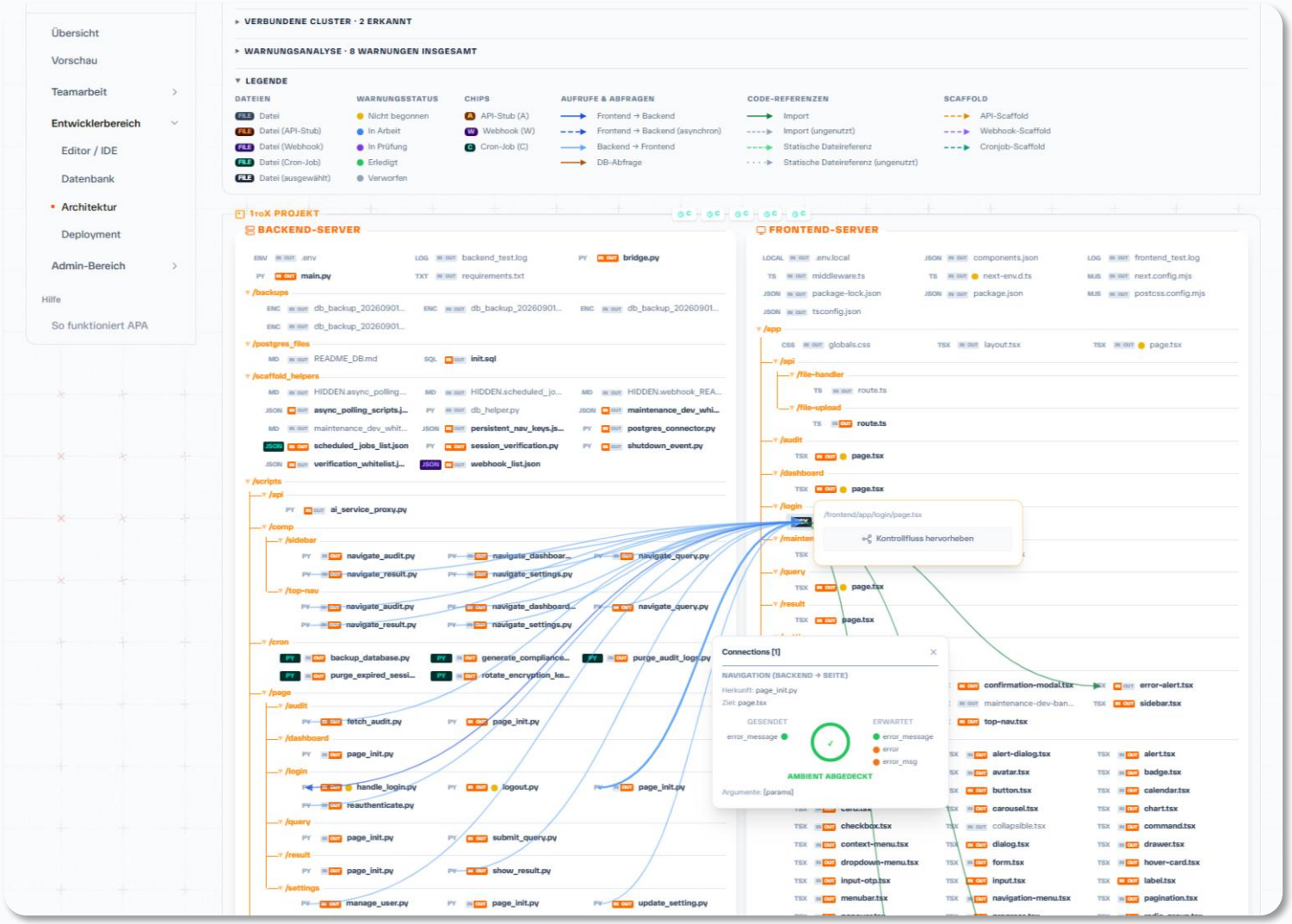
CONFIDENTIAL



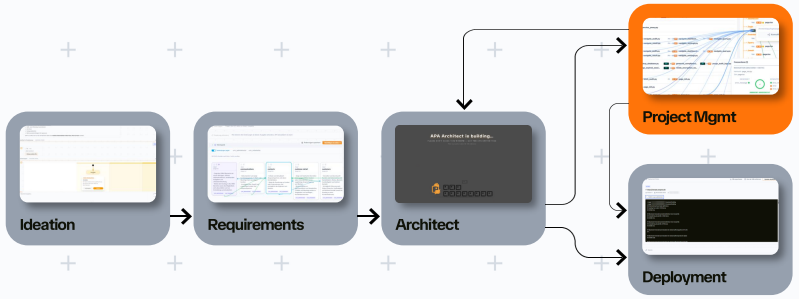
Product Generation

The agent builds against fixed conventions, not against a prompt.

- Secure by structure, not by prompt
- Deterministic checks gate every hand-over
- Runs fully on-prem if required
- Integrates via API into your own pipelines



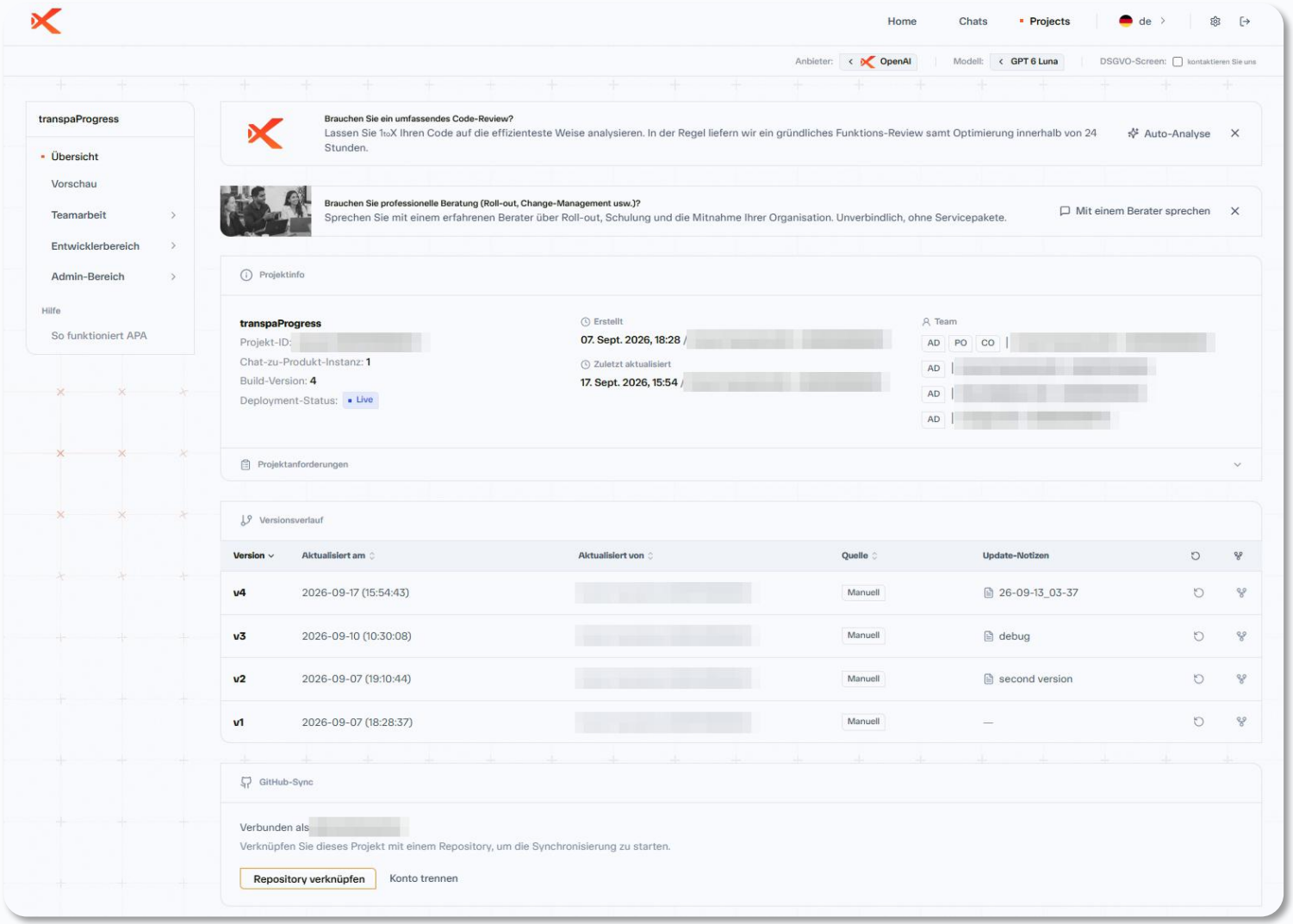
CONFIDENTIAL



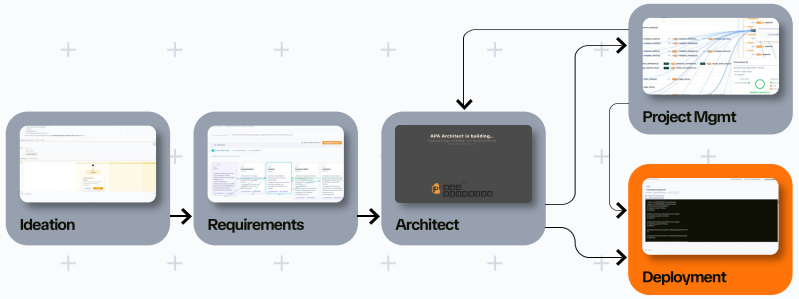
Project Management

Run every project with your team, fully transparent.

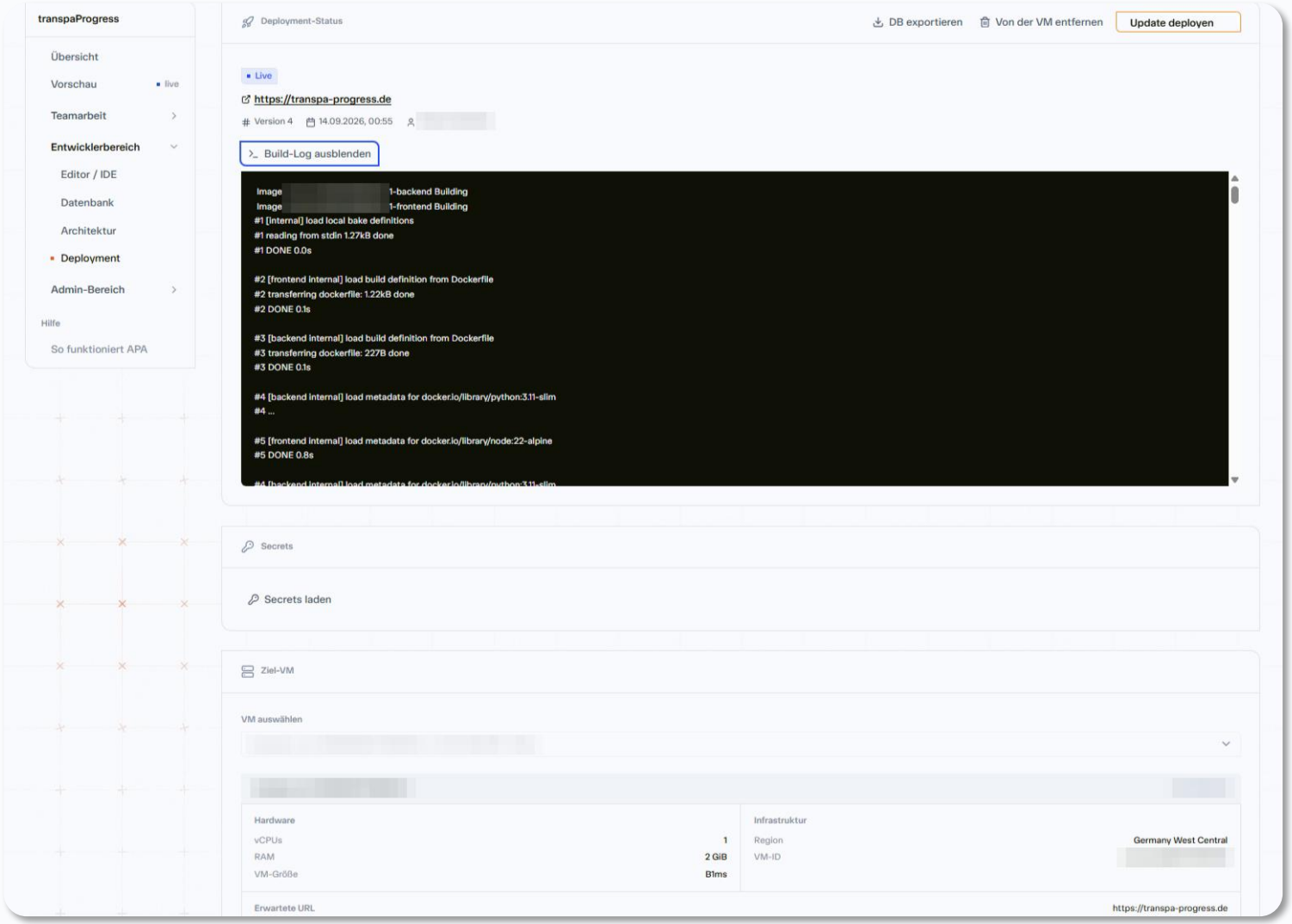
- Full architecture transparency, mapped automatically
- Automatic risk and quality warnings
- Every version saved, rollback anytime
- Built-in IDE with AI assistant and live preview



CONFIDENTIAL



Deploy & Monitor



Go live in one click and stay in control.

- One-click deployment
- Hosted in Germany, ISO 27001 data centres
- Your own domain, SSL included
- Live monitoring, usage and budget control

The calendar is working in our favor: Four curves cross within the next 14 months.

A lot happening today already.

84 %

of developers already use AI coding tools[19]

24 % of production code is AI-generated; a third of companies run on more than 60 %.[16,20]

1 in 5

EU companies had a serious AI-code incident in the last 12 months[16]

Confirmed AI-code CVEs rose from 6 to 35 per month within two months (Q1 2026).[21]

€15 M | 2.5%

CRA fines, plus personal director liability under NIS2[12,13]

From 9 Dec 2026 software is a product under strict liability: “defective” replaces “careless”.[22]

42 %

of cyber policies already exclude AI incidents[15]

Proof of controls for AI code is becoming an underwriting condition at the next renewal.[23]

2024
Our thesis: AI code needs enforced structure, not prompts

Dec 2025
NIS2 in force in Germany: directors liable personally[13]

Aug 2026
AI Act transparency duties apply[24]

Sep 2026
CRA incident reporting: 24 h / 72 h[12]

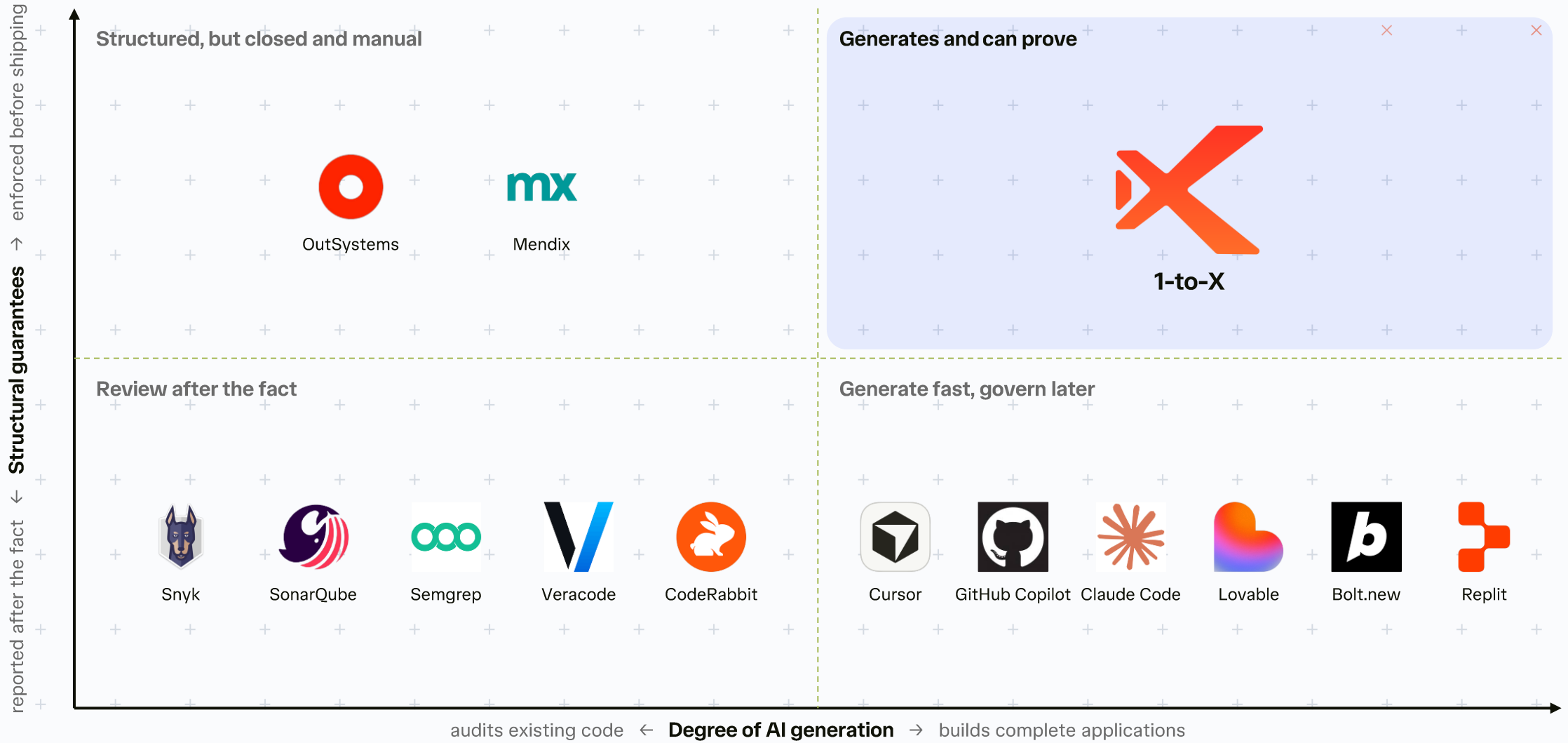
Nov 2026
Slush

9 Dec 2026
Software becomes a product under strict liability[22]

Dec 2027
CRA fully applicable, AI Act high-risk duties[12,24]

COMPETITION

Everyone generates. Nobody guarantees.



Want to see BM, GTM, Traction, ... , more?

→ We do send out the full pitch deck once we received an NDA.