

⌘ PRODACITY 2026 · DAY 2 · NASHVILLE

The Leader's Guide to AI in Production

Russ Miles

Principal Engineer, TechTalk



✦ WHERE THIS COMES FROM

I help engineering teams build a working relationship with AI agents.

Not model selection. Not tooling procurement. The daily practice of humans and agents sharing one codebase without one of them quietly ruining it.

HABITAT THINKING SERIES

Everything that follows comes out of that practice — and out of a story from 1861.



The Sovereign Engineer

[leanpub.com/
thesovereignengineer](https://leanpub.com/thesovereignengineer)



The Sovereign Apprentice

[leanpub.com/
thesovereignapprentice](https://leanpub.com/thesovereignapprentice)

01

Agentic software engineering

Agents as participants in the work — not tools you pick up and put down.

02

Habitat engineering

You don't govern an agent by instructing it. You govern it by shaping what its environment permits.

This one isn't about engineers. It's about leadership.

QUESTION ONE

What does it mean to be a great leader when there is more than one cognition in the room?

Hold both. Every slide after this is one attempt at an answer — from a man who faced the same question in 1861.

Not because the engineering changed. Because there is now more than one cognition in the room — and your codebase is being thought about by something that isn't on your team.

QUESTION TWO

So what, then, is the role of the leader?

Reviewing the work was the answer for a century. It no longer scales to the volume being produced.

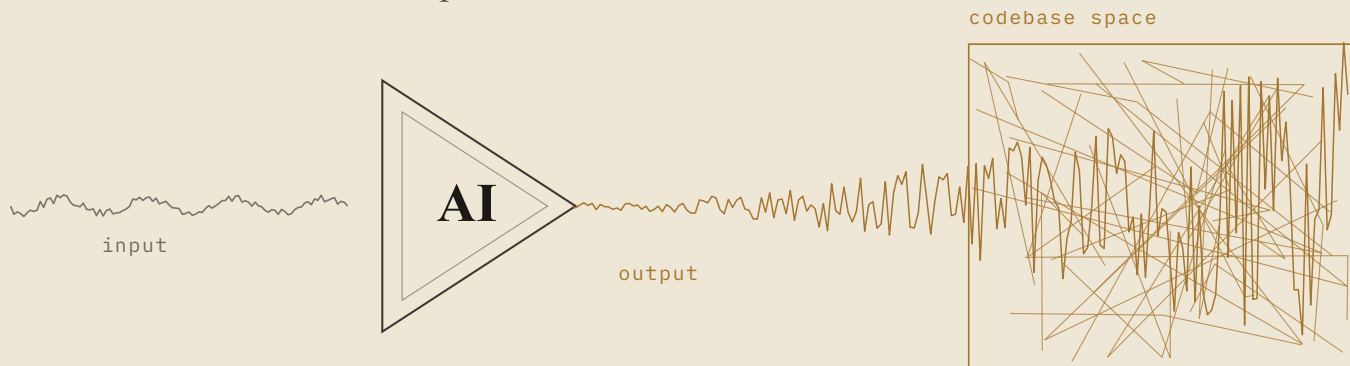
*Every mandate
says the same word.*

Faster.

Every generated line you put live is attack surface. Volume is not a KPI in a DevSecOps shop.

AI is an amplifier.

AI does not fix a broken process. It accelerates it.



No clear constraints in the environment, and what you amplify is *drift* — technical and architectural, at unprecedented scale.

DORA 2025 · 150-developer controlled study

And what it amplifies *from* is the repository it reads. **Bad code gives the agent bad context** — it doesn't see debt, it sees precedent. First time the computers care about code quality, so quality isn't only an output you hope for: it is part of the habitat.

9.5 code health to hold vs 5.15 industry avg
AI breakage low Tornhill, 2026

Meanwhile, the codebase, the understanding, even the intent, *drifts*.

Three layers erode independently. Most organisations measure exactly one of them.

IN THE CODE

Technical debt

Shortcuts, weak abstractions, missing tests. Thirty years of vocabulary — and the layer agents often improve.

IN YOUR PEOPLE

Cognitive debt

Shared understanding erodes, because no engineer had to build the mental model that produced the artifact.

IN YOUR ARTIFACTS

Intent debt

The why — goals, constraints, decisions — never externalised where a teammate or an agent can consult it.

In AI-assisted work the last two may matter *more* than the first — so the code gets cleaner while the understanding and the intent rot.

Margaret-Anne Storey, 2026
“From Technical Debt to Cognitive and Intent Debt” · ACM Queue

Your fastest collaborator is an amnesiac.

Tireless, fluent, generating faster than anyone can read — and every session starts at zero.

what it carries forward



AND NO
BETTER MODEL
FIXES THESE

memory
none beyond the context window

evaluation
none intrinsic

metacognition
no “something feels off”

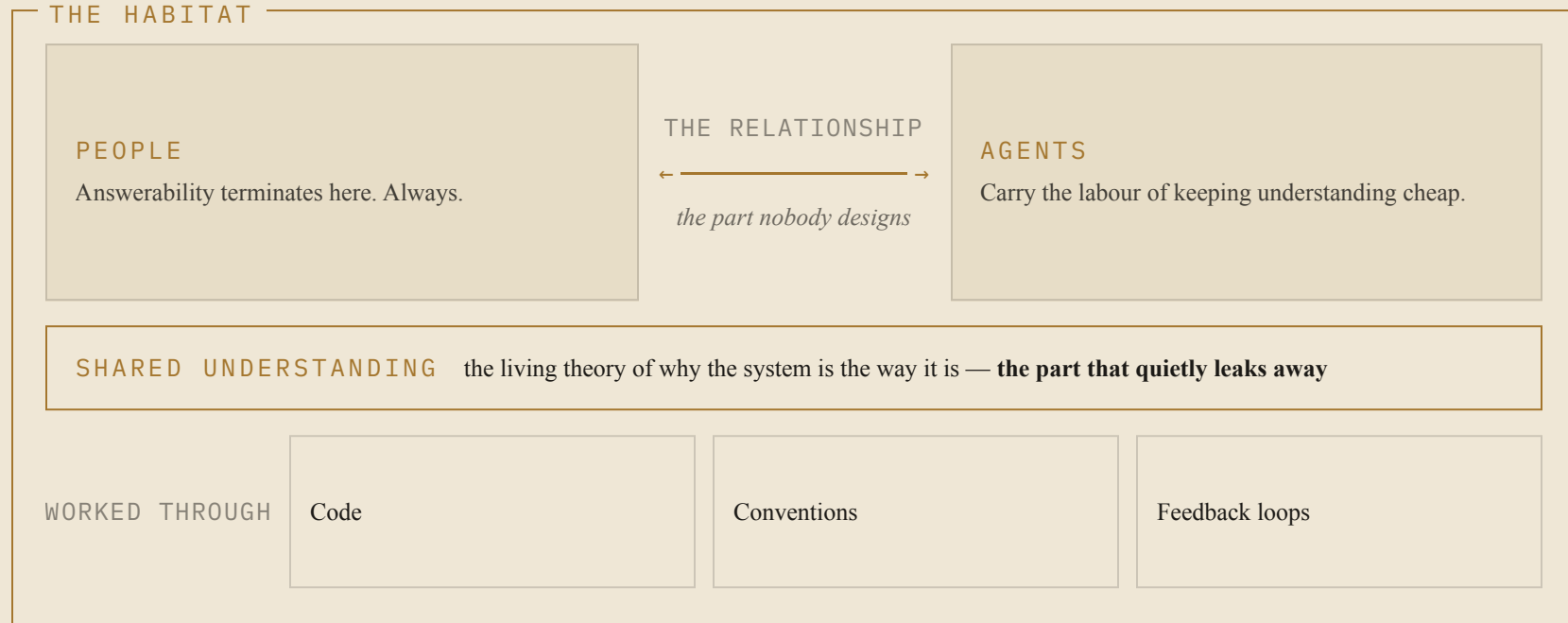
It doesn't remember your conventions, your last surprise, or why last Tuesday's decision went the way it did.

And it never says “*I don't remember.*” It reconstructs — confidently. That's how drift gets in.

So the habitat holds the memory. Continuity is environmental now, not personal.

The habitat is the whole room, not the repository.

Code is one artefact in it. Design for the room, and the code takes care of more than you expect. Design only for the code, and the room slowly stops being one.



Which gives you the oldest question habitability ever posed. *Is this a good place to live?*

**Five levels. Fourteen
dimensions. One question.**

What does your habitat actually *deliver* — not how clever is the model.
The verb is the finding.

L1	L2	L3	L4	L5
Dictating	Commanding	Regulating	Orchestrating	Supervising
<i>“I am running it”</i>	<i>“I am inspecting”</i>	<i>“I am standardising”</i>	<i>“I am specifying”</i>	<i>“I am certifying”</i>
ad-hoc prompts · safe runtime	prompts & commands saved	harness engineered · constitutional	workflow defined · policy-as-code	workflow automated · evidence

At **L3** the habitat — not the human — starts holding the constraints.
Each level is the floor of the next.

Agentic Experience Habitat Maturity Model
coherence is the signal, not the level

The model was never the risk. The collaboration space is.

The assumption underneath that room: we aren't tuning one cognition. We're designing a place where *two very different kinds* can each do what they're good at. A method tells you what to do on Monday; this changes **what the object of engineering is** — and so what counts as a problem, a success, and a fix. Here is what such a place values, and the principles it runs on.

WHAT IT VALUES

The whole cognition in the room

over the code on the screen

Sustained human understanding

over the throughput of the place

The living theory the inhabitants hold

over the artefacts that embody it

Designing the collaboration

over directing the tool

Orientation that outlives the tools

over tactics tuned to today's models

Judgment that compounds

over volume that impresses

Accountability that rests with people

over assurance manufactured by the place

THE PRINCIPLES IT RUNS ON

01 **The habitat is the whole room**

not the repository

02 **Optimise for orientation that outlives the tools**

tactics are rented; orientation is owned

03 **Understanding is a deliverable**

not a bottleneck

04 **Agents keep understanding cheap**

people stay accountable

05 **Protect the friction that makes minds update**

or you hide the debt, not reduce it

06 **Tend the place deliberately, and often**

examine the place, not just the output

There's real value in the right-hand items — a habitat that neglects them is no habitat at all. *We value the left more.* The unit of analysis moved from the artefact to the place: under the old frame most of what follows reads as overhead, *under this one it is the work.*

The Habitat Thinking Manifesto
github.com/Habitat-Thinking/manifesto

Comprehension has decoupled from delivery.

This is the anomaly — the case the old frame cannot absorb. A team ships faster while growing more ignorant of what it ships. That's what forces a frame to change.

READ UNDER PRODUCTIVITY

Success.

Shipping faster. Every number up and to the right.

READ UNDER HABITAT THINKING

Failure.

The same team grew more ignorant of what it ships. One frame can't absorb that. Ours starts there.

NON-PRESCRIPTIVE BY DESIGN

It tells you what to attend to, not what to do on Monday. A compass, not walking directions.

EXEMPLARS COME FROM

the manifesto and the agentic engineering practice

PUZZLES COME FROM

the engagement ladder and the maturity model

A paradigm you can only assent to is a poster. *One with puzzles is something your teams can practise.*

There's no playbook for the leader.

Nobody has run an engineering organisation through this transition and written up what your job becomes.

THE QUESTIONS NOBODY HAS ANSWERED FOR YOU

What do you review, when the volume outruns reading?
What do you measure, when lines shipped means nothing?
Who is accountable when the work was co-authored? And
how do you keep judgment growing in people who are no
longer doing the first draft?

So who can we learn from?

95%

of GenAI pilots, no measurable return — MIT, 2025.
Not the models: a **learning gap**. *They didn't lack
intelligence. They lacked a habitat.*

FOR
ENGINEERS

Prompting guides, tool comparisons,
benchmarks, a thousand talks.

FOR YOU

Your peers improvising, and vendors
selling.

SO

Write one from scratch, in production, at
speed — or borrow from someone who
already did.

⌘ LOOKING FOR HELP FROM A LEADER OF THE PAST

Lincoln, the great... early adopter.

*Not the marble Great
Emancipator. The man who
moved into the machine room.*

Nobody in 1861 had a playbook for leading through a technology that collapsed the distance between a decision and its consequence. He wrote one in practice, under maximum pressure.

PATENT 6,469

The only US president to hold one — a device for lifting stuck boats over shoals.

FIELD TRIALS

He tested new weapons himself rather than accept the Ordnance Bureau's verdict.

TELEGRAPH

He didn't get briefed on the new machine. He sat in the room where it spoke.





⊕ ACT II — FIVE DISPATCHES FROM LINCOLN

Q1 · MORE THAN ONE COGNITION

DISPATCH ONE · MASTER THE MEDIUM

1861: the machine speaks back. Live in the telegraph office.

He didn't ask for summaries of the new medium. He read the traffic himself. He inhabited it.

Your telegraph office is the repo. Open a session. Read a diff. Watch an agent work.

Forty-five minutes, sat with a team.

You don't need to touch the repo. You need to watch how your people and their agents actually work together.

If the answer to the third one is no, you are funding Level 5 and running Level 1.

SIT IN

One real ticket, one real team. Not a demo, not a sandbox — work they'd have done anyway.

HAVE THEM NARRATE

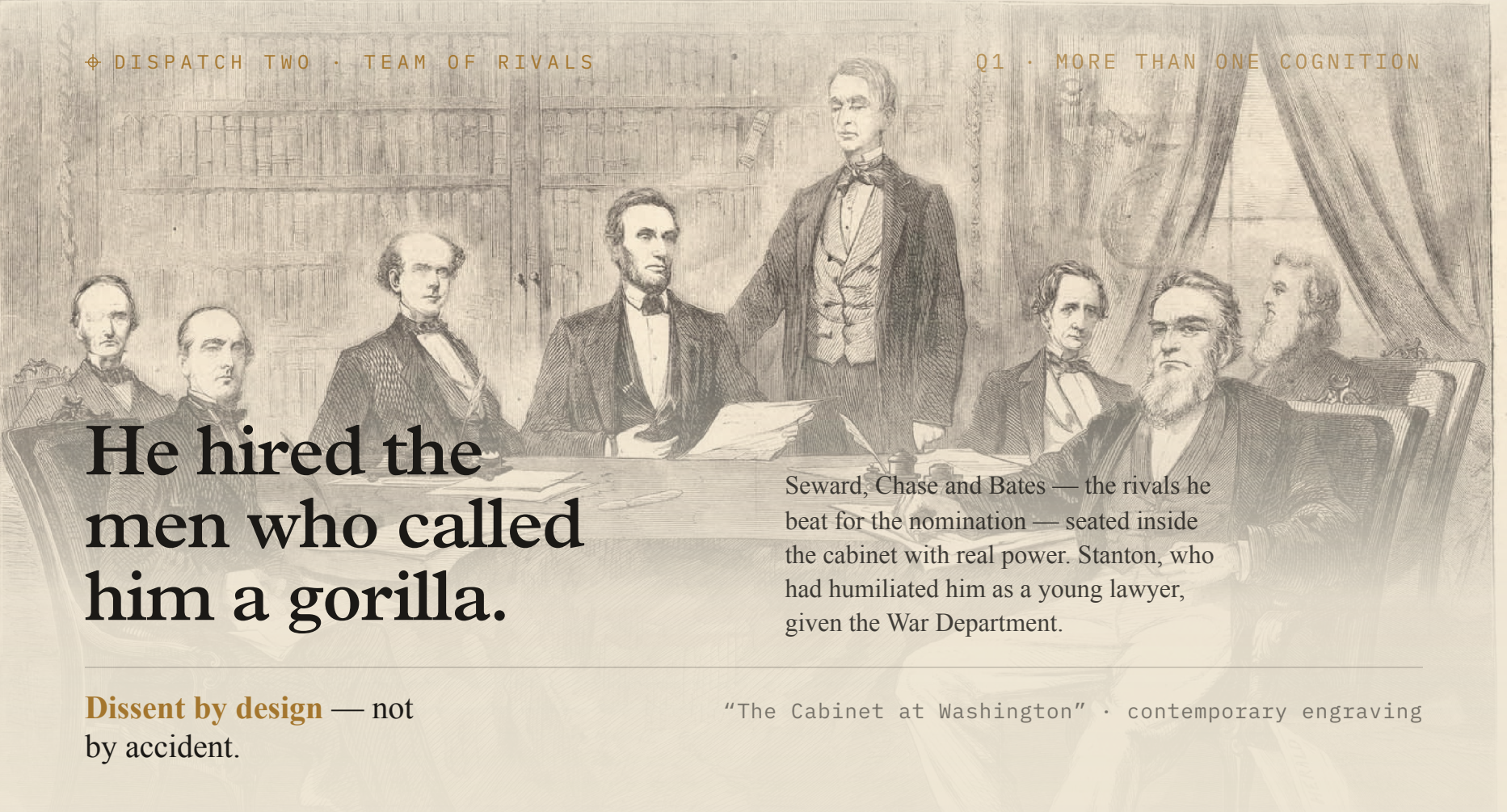
What they asked for, what the agent assumed, what they had to catch. Especially where it went wrong.

THE TEST

Can anyone in the room say what the agent was **forbidden** from doing last week?

✦ DISPATCH TWO • TEAM OF RIVALS

Q1 • MORE THAN ONE COGNITION



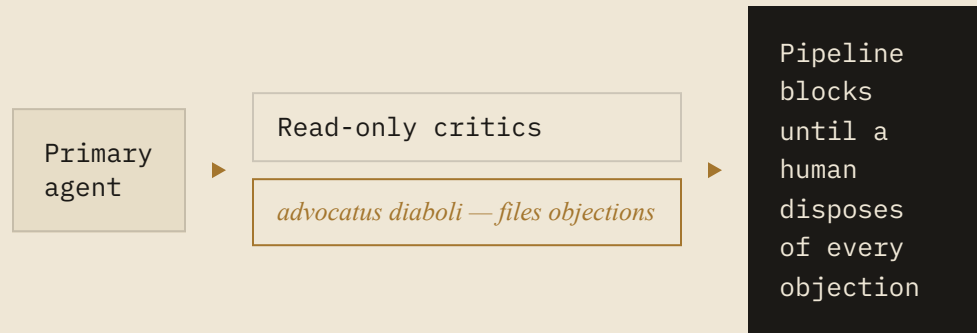
He hired the men who called him a gorilla.

Seward, Chase and Bates — the rivals he beat for the nomination — seated inside the cabinet with real power. Stanton, who had humiliated him as a young lawyer, given the War Department.

Dissent by design — not
by accident.

"The Cabinet at Washington" • contemporary engraving

Give the machine rivals.



Sentinels never sleep.

ENFORCER

verifies every constraint

AUDITOR

checks the harness against reality

GARBAGE COLLECTOR

sweeps entropy on schedule

WARDEN

watches the tired human doing the
approving

“*The harness slows us down.*”

FRICTION THAT PAYS

It names what it protects

It fails fast, loudly, at edit time

A **mechanism** enforces it — not a meeting

It stops something you could not unpick later

*Brakes. You buy the delay back
with interest.*

They’re right. It *is* friction.

FRICTION THAT LEAKS

A rule nobody can defend

A check whose owner has left

Slow because nobody optimised it

An approval that spreads blame, not catches faults

*A dragging handbrake.
Determinacy debt.*

From inside the sprint, both columns feel the same.

Your job isn’t to defend the harness.
It’s to delete the right-hand column in public.

Festina lente.

Make haste slowly.
Augustus, via Suetonius

Brakes are not what stops a car. Brakes are what let it be driven fast.

Take them off and the car does not go quicker. It goes slower — because the only survivable speed is the one you can still come out of. Nobody brakes late on a road they cannot see.

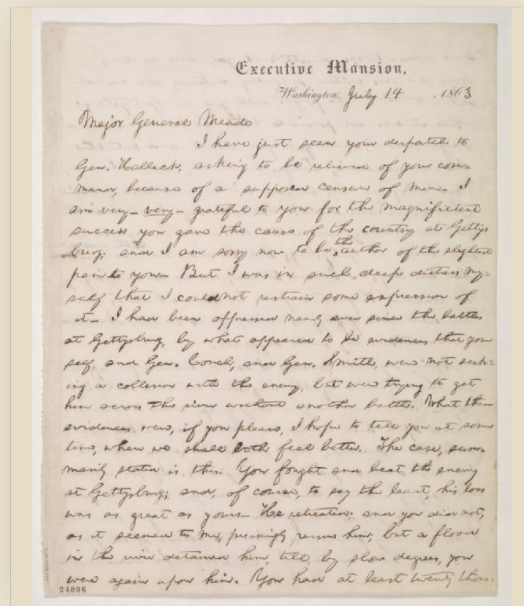
The haste is the goal. The slowness is the instrument that makes the haste survivable.

The harness is not the tax on speed. It is the *permission* to have it — which is why the causality runs the other way than leaders assume.

You don't go fast and add controls to stay safe.
You add the controls and that buys the speed.

He wrote the letter. He never sent it.

The value was in the
writing. **The wisdom was
in the pause.**



Executive
Mansion, 14
July 1863

endorsement:
"never sent, or
signed"

Every AI draft is a hot letter.

*Not whether it reads well — whether
it survives the pause.*

No theory, no merge.

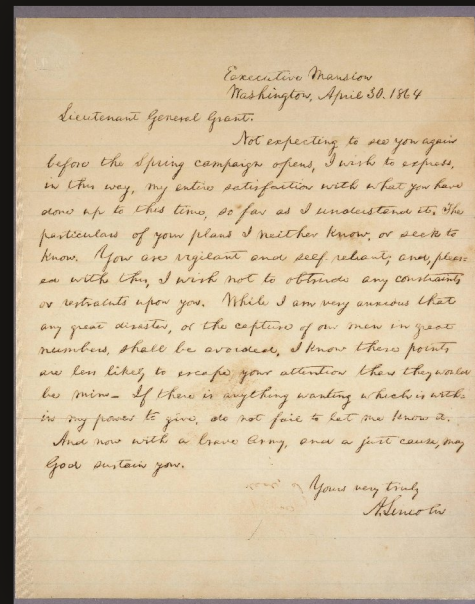
Naur, 1985 — programming is theory building. The code was never the whole value; the understanding was. If no human holds the theory of a change, it doesn't ship.

One line, and it's *yours* to set: the difference between AI as an amplifier and AI as a substitute.



“
*The particulars
of your plans I
neither know,
nor seek to know.*

Specification leadership – intent without micromanagement.



Lincoln to Lt. Gen. Grant
· Executive Mansion, 30
April 1864

Write orders the habitat can enforce.

ADVISORY

Warn

at edit time

STRICT

Block

at merge time

INVESTIGATIVE

Sweep

on schedule

HARNESS.md the source of truth — non-negotiables and the definitions of all three loops.

AGENTS.md a control surface downstream of it — conventions the agents read, never the authority.

Intent that outlives the session, the contractor, the PCS cycle.

Three lines from a real one.

Advisory warns. Strict blocks.
Investigative sweeps. The third
one is why the file is alive and
not a wiki page.

*Recorded is not enforced. The
arrow is the whole difference.*

```
# HARNESS.md · [[service name]]
```

ADVISORY

```
[[convention the agent gets wrong at  
edit time]]
```

STRICT

```
[[non-negotiable that blocks the merge]]  
→ enforced: [[the check that enforces  
it]]
```

INVESTIGATIVE

```
[[what the scheduled sweep looks for]]  
→ last sweep: [[date]] · [[n]] findings
```

When the assessor asks, “*the AI wrote it*” is not an answer.

after Matthew Skelton, *Team Topologies* · Aug 2026

They did not prompt harder.

They built the gate.

THE GATE IN

Code below a fixed health threshold is not handed to an agent at all.

THE GATE OUT

A commit-time check rejects agent output that drops the score; the work loops back to refactor.

THE CONTEXT

Quality signals fed into the agent's loop *as it writes* — not a review after.

6.0 → 9.75

hotspot code health,
twelve months

~70%

of output now
AI-assisted

96%

build success rate,
AI-assisted commits

138 → 44 min

P90 lead time to deploy,
elite frequency held

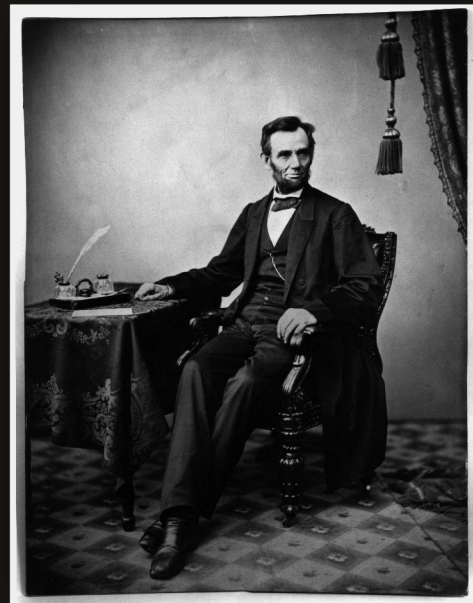
loveholidays
engineering,
AI diary #6-7,
Mar 2026

⌘ DISPATCH FIVE · TO GRANT, JUL 1863

Q1 · MORE THAN ONE COGNITION

“
*You were
right, and I
was wrong.*

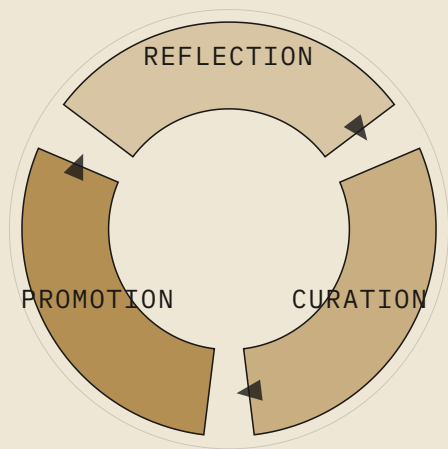
Owned publicly. Never wasted on blame.



Lincoln at the writing
table, 1863 – the letter
to Grant went out over
this desk

Let surprises leave scar tissue in the repo.

How a habitat gets smarter: the system learns from human corrections, so it can't repeat the mistake.



01 Reflection

Capture the surprise or edge case after an AI session.

02 Curation

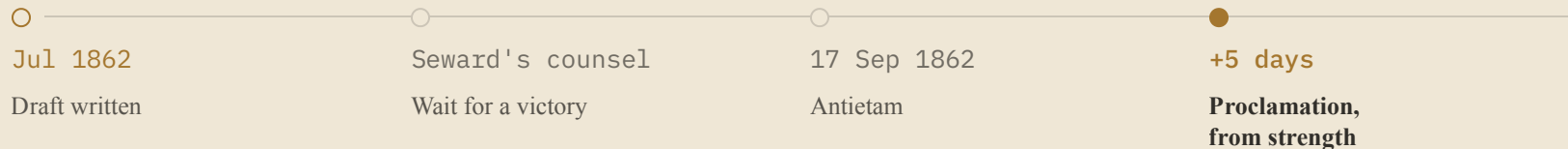
Spot the recurring pattern in AI failures over time — human work.

03 Promotion

Upgrade the reflection into a hard constraint or context rule.

Blameless, but never memoryless.

Right move, wrong moment, still fails.



Habitat before altitude
— 1862 edition.

Seward's counsel, in effect: don't borrow altitude the moment can't carry.

Don't skip levels.
Take a fix.

This measures what a team's habitat *delivers* — not how clever the model is. The verb is the finding.

	L1 Dictating I am running	L2 Commanding I am inspecting	L3 Regulating I am standardising	L4 Orchestrating I am specifying	L5 Supervising I am certifying
AGENT INPUT	Short ad-hoc prompts	Larger prompts, saved commands	Plans co-authored with an agent	Iteratively refined specs	Specs + observable metrics
WORKFLOW	Safe runtime, generic	Prompts & commands saved	Harness engineered	Workflow defined	Workflow automated
OPERATING MODEL	Chat with agent	Prompt engineering	Humans drive & verify	Humans in the loop	Humans certify
WORK PATTERN	Partial task	Small task	e2e development	Semi-autonomous epics	Mostly autonomous
TESTING	Manual	Asserting	Verifying	Validating	Assuring
GOVERNANCE	Trust-based, ambient	Conventional	Constitutional	Policy-as-code	Continuous certification

Most teams sit at L1–L2. At L3 the habitat — not the human — starts holding the constraints.

habitat build gap = cognitive level – habitat mean
Thinking at L4 on an L2 workflow is a positive gap: ambition outpacing enablement.
Coherence is the signal, not the level.

Capability you can measure precisely. Value only approximately.

Lines of code, function points, velocity, now acceptance rate — every era picks a proxy and watches it get gamed. The fix isn't a better metric. It's refusing to collapse three layers into one number.

Capability & flow weeks · team-owned	Lead time · deploy frequency · batch size · review latency · CI reliability · rework rate · the verification tax — time spent checking generated work	capability
Reliability & cost quarters	Change failure rate · recovery time · rework as a share of capacity · retention (a first-order ROI line now, not a soft one)	capability
Business value quarters to years · product-owned	Value realised against a <i>stated hypothesis</i> · cost to serve · % of shipped features that met their benefit case	value

The four failure modes, every time: output not outcome · attribution ignored · Goodhart the moment it touches budget · costs certain, benefits deferred.
DORA · SPACE (2021) · DevEx (2023) · DX Core 4 (2024) · Kersten, *Flow Framework* · Reinertsen, *Cost of Delay*

Rigorous about capability.
Honest that value is a hypothesis under test.

Your metrics are a job description.

CONSTRUCTION METRICS

~~Lines of code generated~~ — attack surface added

Commits, PRs, “AI-assisted %”

Velocity of output

The leader-level form of the first misconception.

Team Topologies drew construction boundaries. The question now is stewardship: who is answerable.

STEWARDSHIP METRICS

Can a **named team** account for what the agents did?

Is there a traceable story of decisions, cost and controls?

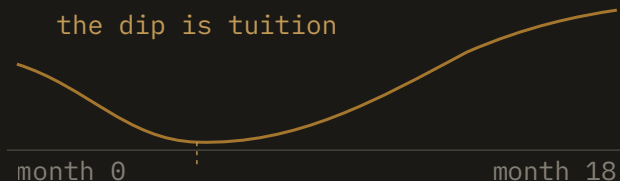
Is the value flow healthier than last quarter?

The harness stops being a cost. It becomes the evidence.

Measure the stewardship of the flow, not the volume of the build.

Framing after Matthew Skelton, “Reframing the AI-native SDLC in terms of stewardship boundaries,” Aug 2026

Budget for the dip.



Get CFO agreement on the dip *in advance*, or funding gets pulled at month four when the numbers look worse.

Reinvest capacity, don't cut heads. The return is reclaimed rework.

System, not tool. Otherwise you buy local pockets lost in downstream chaos.

FOUR RULES I'D HOLD YOU TO

- 01 **Counterbalance speed with quality**
published together, always — a speed number alone is a licence to break things
- 02 **No metric crosses the individual boundary**
team or value stream only; the difference between measurement and surveillance
- 03 **State the value hypothesis first**
and review it after — otherwise you're measuring activity, not value
- 04 **Audit the business case retrospectively**
highest leverage, lowest adoption practice in the field

If you take one thing: never publish a speed number on its own.

DORA, *ROI of AI-assisted Software Development* v2026.1
DORA 2025 · DX Core 4, revisited June 2026

Five dispatches, one artifact.

Telegraph office	→ you, close to the action
------------------	----------------------------

Team of rivals	→ adversarial review · standing sentinels
----------------	---

The hot letter	→ the trust-protocol gate
----------------	---------------------------

Lincoln's orders	→ the HARNESS.md
------------------	------------------

The owned failure	→ GC rules that encode your scars
-------------------	-----------------------------------

Your first moves on “Monday”: assess, send, read.

ASSESS

Run it across the portfolio;
habitat/workflow gaps on one page.

SEND

A pair to build the first harness — into
Thursday's room.

READ

One HARNESS.md a week, yourself, in
the telegraph office.

⌕ BACK TO THE TWO QUESTIONS

QUESTION ONE

What does it mean to be a great leader when there is more than one cognition in the room?

You stop being the best reviewer in the room.

Judgment doesn't scale by reading more diffs. It scales by being encoded — constraints, harnesses, sentinels, orders the environment can enforce.

QUESTION TWO

So what, then, is the role of the leader?

Habitat engineer.

You shape the place the work happens — and keep reshaping it as the scars accumulate.

Lincoln never out-wrote his generals. He built a telegraph office, gave the machine rivals, wrote orders the field could enforce — and never sent the hot letter.

“

*We can't just hope
for a brighter day,
we have to work for a
brighter day. Love too
often gets buried in a
world of hurt and fear.*

Dolly Parton

**The habitat is
never finished.**

Not the harness, not the sentinels, not the scar tissue. There is no version of this where you declare it done and go back to reviewing diffs.

A thriving one is worked for, never hoped for — and the work isn't the agents' to take, or your engineers' alone. It belongs to whoever can decide what the environment permits.

That's you.

You cultivate the habitat.

Nobody else in your organisation can do this.

WIRE THE AMPLIFIERS

Purpose, focus, alignment. The mission is the strongest amplifier in the room — and it costs you nothing.

REMOVE THE DRAINS

Threat, metric worship, distraction. Every one of them is a decision someone above the code made.

READ THE CHANNEL

Gap-delta, drift, GC findings. Measure the habitat — never “lines of AI code accepted.”

Engineers can build a harness.
Only you can make it non-negotiable.

Engineers can spot drift.
Only you can fund the sentinels.

Engineers can feel the incoherence. Only you can close the gap.

The habitat evolves, or it rots. Platforming it is your key responsibility — and right now, probably your biggest.

Assess the gap. Build the harness. Keep it living.

assessment github.com/techtalk/ai-readiness-assessment

writing softwareenchantidion.com · The Sovereign Engineer · The Sovereign Apprentice

Thursday Agentic Engineering workshop

free book a session – run the assessment on your own org, with me
hour

find me russell.miles@techtalk.at · russmiles.com