Freedom *and* Trust

自由と信頼

What AI-Augmented Development Can Learn from the Toyota Production System

AI拡張開発がトヨタ生産方式から学べること



Terry Yin · Odd-e

Tokyo LeSS Conference · 2026

About Me

自己紹介

I coach **LeSS** and technical practices at **Odd-e**.
Odd-eで**LeSS**と技術プラクティスのコーチをしています。

Nearly **30 years** building software, including a decade inside Nokia R&D — and I still program.
ソフトウェア開発は**30年**近く、うち10年はNokiaのR&D。今もプログラムを書いています。

Still programming. Still learning. Now asking what AI should free us from.
今もプログラミングし、学び続けている。いま問うのは、AIが私たちを何から解放すべきか。

Terry Yin · Singapore · terry@odd-e.com

釈迦に説法

Preaching to the Buddha — TPS in Tokyo.
TPSを東京で語る。

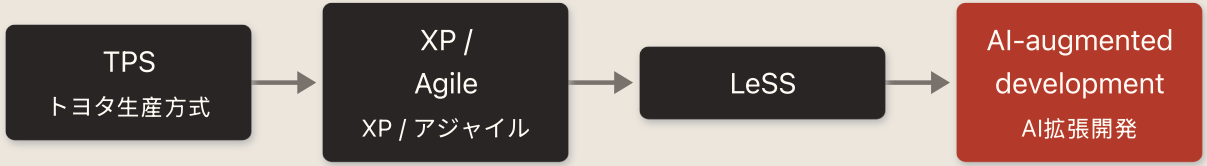
TPS has helped me. Even more with AI.
TPSは役立ってきた。AI時代には、なおさら。

- Software combines **discovery and building**.
  ソフトウェアは、**発見しながら作る**。
- Adapt **TPS thinking** to software.
  TPSの考え方を、ソフトウェアに合わせて活かす。



One lineage of inspiration

ひとつながりのインスピレーションの系譜



TPS inspired **XP** and the Agile movement,
TPSが**XP**とアジャイルムーブメントにインスピレーションを与え、

then **LeSS** —
LeSSにも——

and now, **AI-augmented development**.
そして今、**AI拡張開発**へ。

How do you know if the organization is using AI right?

組織がAIを正しく使えているか、どうすればわかる?

If the teams are more **freed** than **constrained** by what they built.

チームが作ったものに**縛られる**より、**解放されている**なら。

Constrained by what they built

自ら作ったものに縛られる

- Leftover ownership
  オーナーシップの残骸
- Judgment-loaded output presented as finished
  必要な判断がされないままのアウトプットを「完成」としてしまう
- Unable to take the next highest-value item
  次に最も価値の高いアイテムに取りかかれない

Being **constrained** ≠ taking **responsibility**
縛られる ≠ 責任を引き受ける



Judgment-intensive work

判断集約型の仕事

Use judgment to discover. **Encode what you learn.**

判断して見つける。**学びを仕組みにする。**

In a document

文書なら

“The list must not be empty.”

「リストは空にしない」

Someone must interpret it.

誰かが解釈する。

In a test

テストなら

Empty list → failure.

空のリスト → 失敗。

Otherwise, no news.

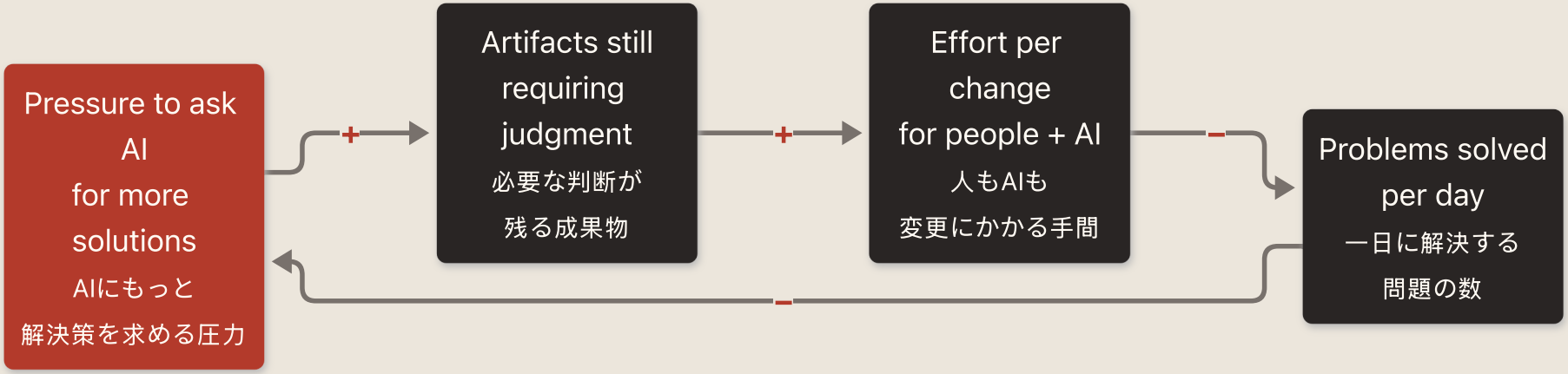
問題なければ、何も起きない。

The next person need not rediscover the rule.

次の人がルールを見つけ直さずに済む。

AI speeds whichever loop you feed

AIは、好循環も悪循環も加速させる



+ increases · – reduces

＋ 増やす · – 減らす

Freedom vs. trust?

自由 VS 信頼?

Give freedom.

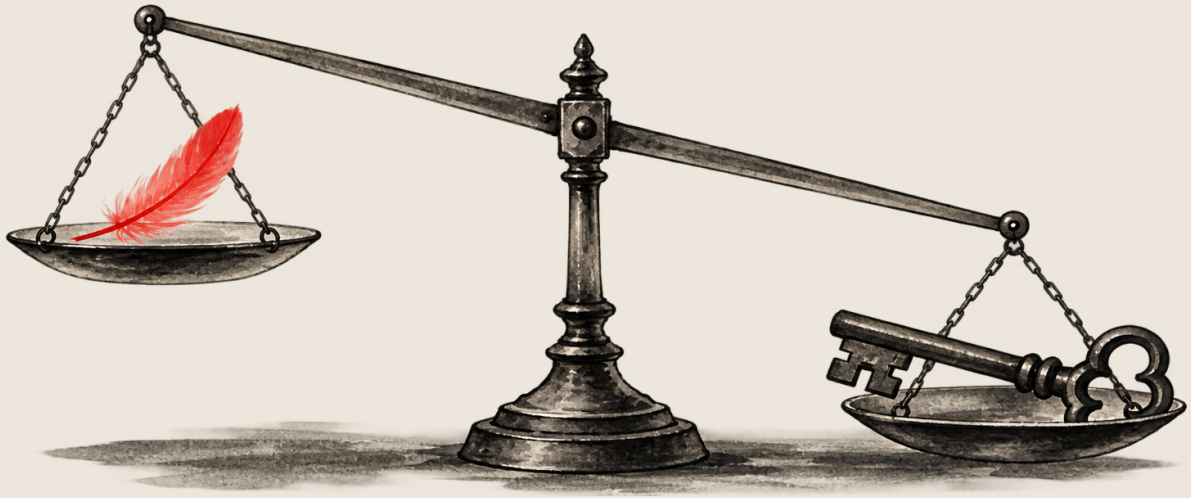
Withhold real work?

自由なら、大事な仕事は任せられない？

Entrust real work.

Control every step?

任せるなら、手順まで管理する？



Why should we have to choose?

両立できないのだろうか。

TPS shows how a system can continually convert learning into constraints that make greater freedom responsible —

and use that freedom to produce the next learning

on which deeper entrustment, and then mutual trust, can rest.

TPSは、学びを制約に変え、その制約によってより大きな自由を責任あるものにする。その自由で次の学びを生み出す。

その学びを土台に、より深く任せられるようになり、やがて相互の信頼が成り立つ。

Two houses, different layers

二つのハウス、異なる階層

The TPS house

TPSハウス



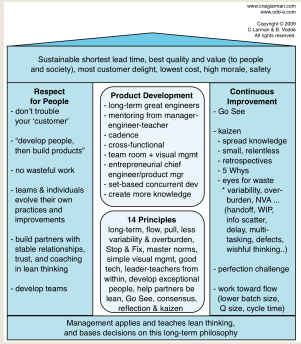
Toyota's **operating system** for making things: **Jidoka** builds quality in by stopping; **Just-in-Time** makes only what is needed.

モノづくりのためのトヨタの**オペレーティングシステム**：**自働化**は止めることで品質を作り込み、**ジャスト・イン・タイム**は必要なものだけを作る。

Commonly taught Cho house; structure after Toyota's 1998 TPS booklet. Original drawing — not a Toyota asset. Pillar meanings: Toyota, "Toyota Production System"

Larman & Vodde's Lean Thinking house

Larman & Vodde のリーン思考ハウス



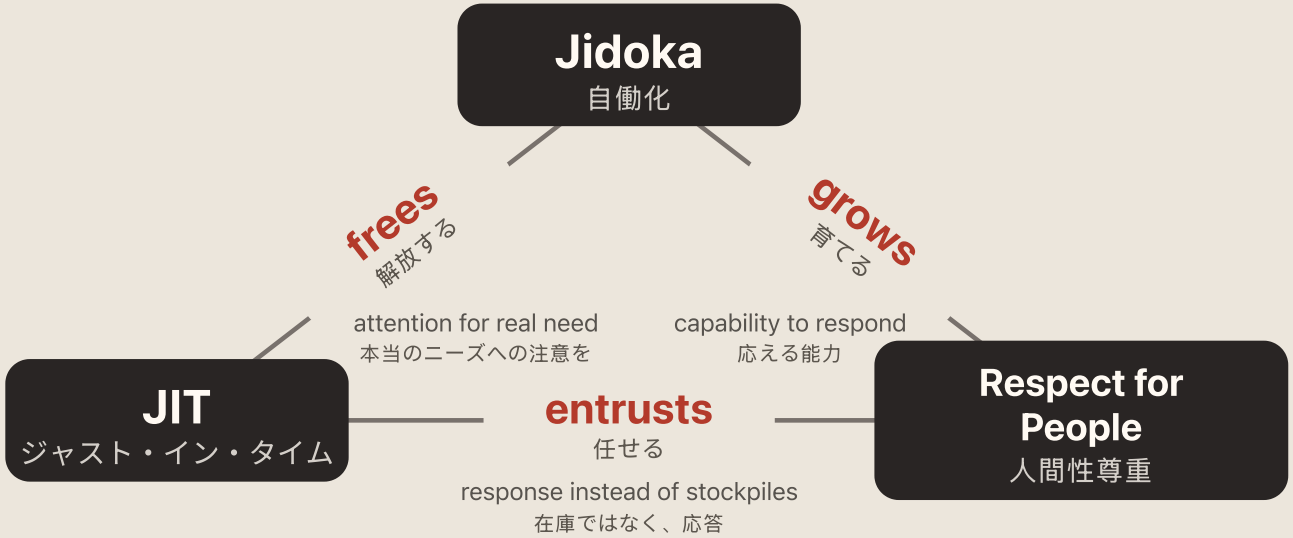
A **Toyota Way / lean-thinking** synthesis: **Respect for People** and **Continuous Improvement**, on managers-as-teachers, toward **perfection**.

トヨタウェイ / リーン思考の統合：教える人としてのマネージャーを土台に、**人間性尊重**と**継続的改善**で**完璧**を目指す。

Larman & Vodde, *Scaling Lean and Agile Development*, Fig. 3.1 (2009)
Creative Commons for presentations via less.works

The triad

三位一体



Technical excellence keeps the shared product and its abnormalities visible soon enough for teams to collaborate just in time.

技術的卓越性が、共有プロダクトの異常を早く見えるようにし、チームがジャスト・イン・タイムで協働できるようにする。

Jidoka preserves knowledge

自働化は知識を保存する

Generation is cheap; **judgment is expensive**.
生成は安く、**判断には多くの労力がかかる**。

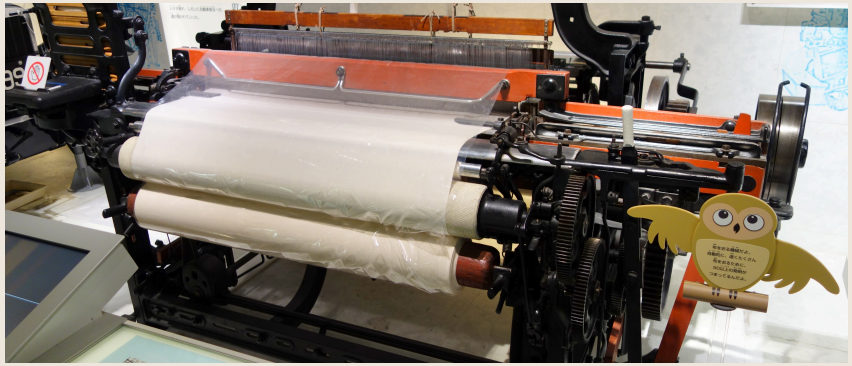
Encode what we already know. Leave people able to **experience** the next problem and **comprehend** the solution.
すでにわかっていることは仕組みに組み込む。
人が次の問題を**体験し**、その解決を**理解できる**ようにしておく。

The loom’s closed stop

織機の「閉じた停止」

Sakichi Toyoda wanted to ease his mother's work.

豊田佐吉は、母の織る仕事を楽にしたかった。



Type G automatic loom, 1924

Photo: Daderot / Wikimedia, CC0

Jidoka / Autonomation

自働化

ninben = person

Automation with a human touch

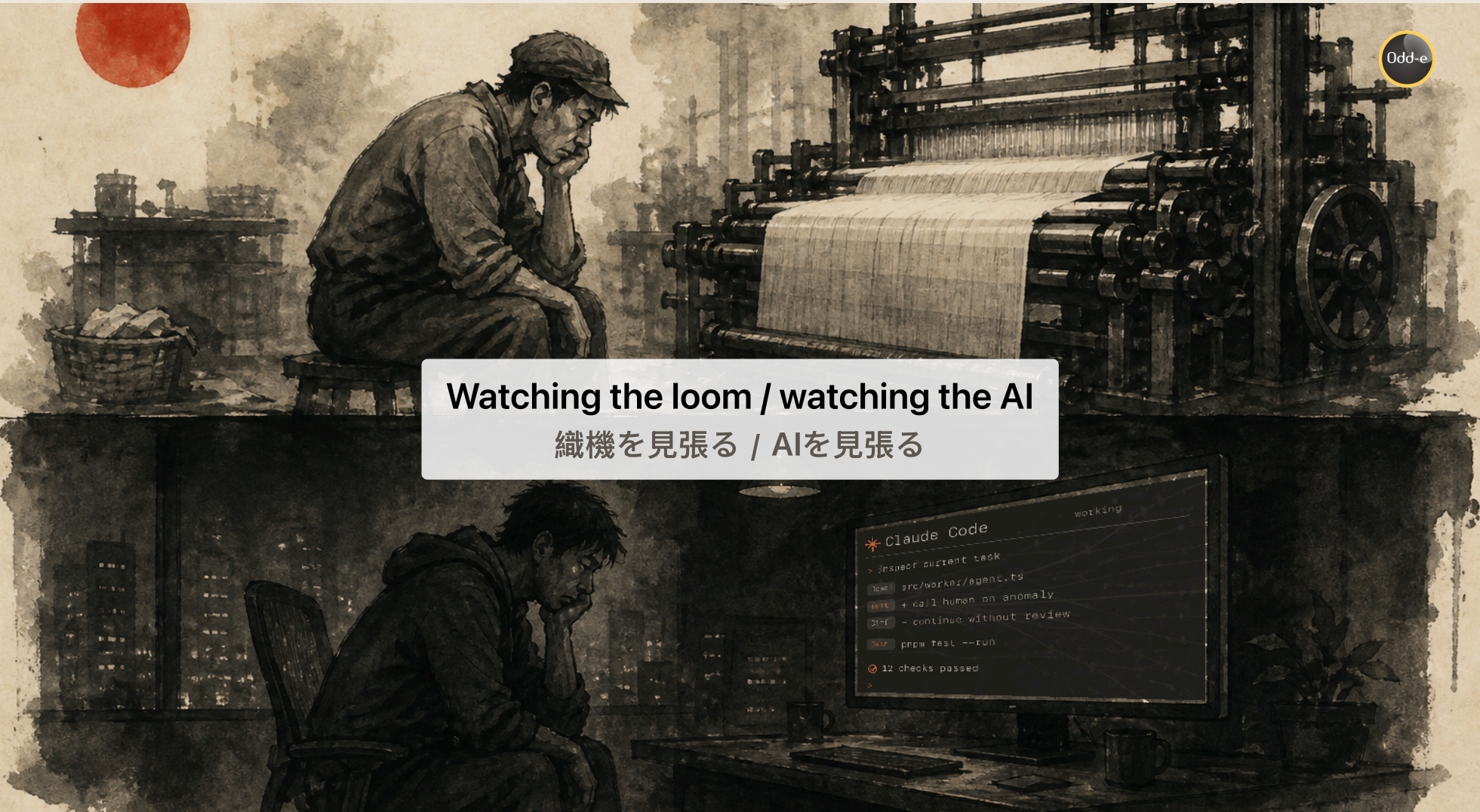
人の知恵を加えた自動化

A broken thread **stops the loom.**

糸が切れると、織機が自ら止まる。

People are **free from watching.**

人は見張りから解放される。


Odd-e
Watching the loom / watching the AI
織機を見張る / AIを見張る
Claude Code
working
> answer current task
Read src/worker/agent.ts
Edit + call human on anomaly
Diff - continue without review
Bash pnpm test --run
12 checks passed
>

Dumb: the rule becomes a stop

単純：判断を停止の仕組みに組み込む

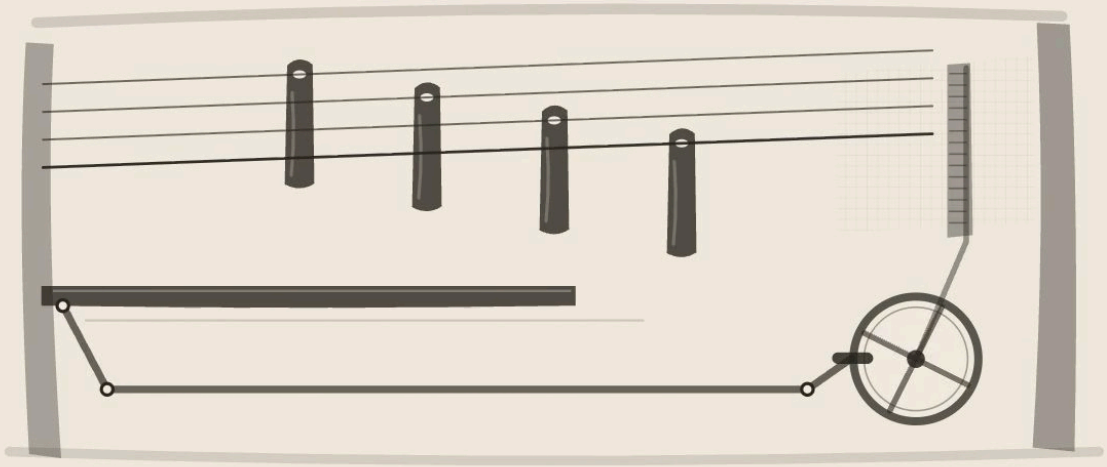




Photo: Christoph Roser
CC BY-SA 4.0

Thread intact: run

糸が張ると動く

Thread broken: stop

糸が切れると止まる

Schematic illustration / 模式図


Odd-e
Called by the stop
停止したら呼ばれる
Claude Code
paused
inspect abnormal result
src/agent/stop.ts
check returned unknown
review required
Waiting for human judgment
context preserved

Smart → dumb → gone

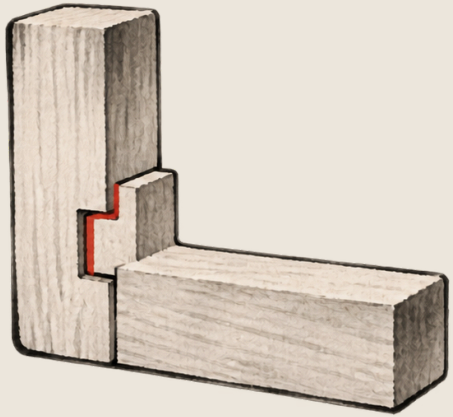
賢い → 単純 → 消える



Smart

Judge while creating

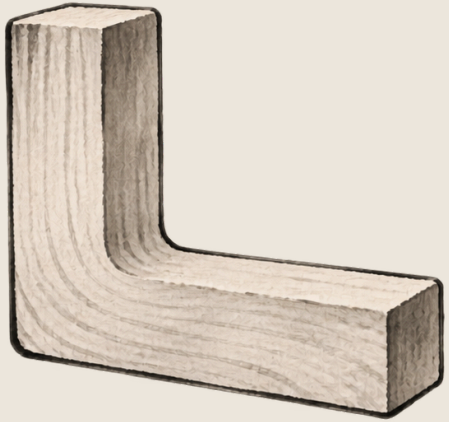
つくるときに判断する



Dumb

The joint checks the fit

形が間違いを防ぐ



Gone

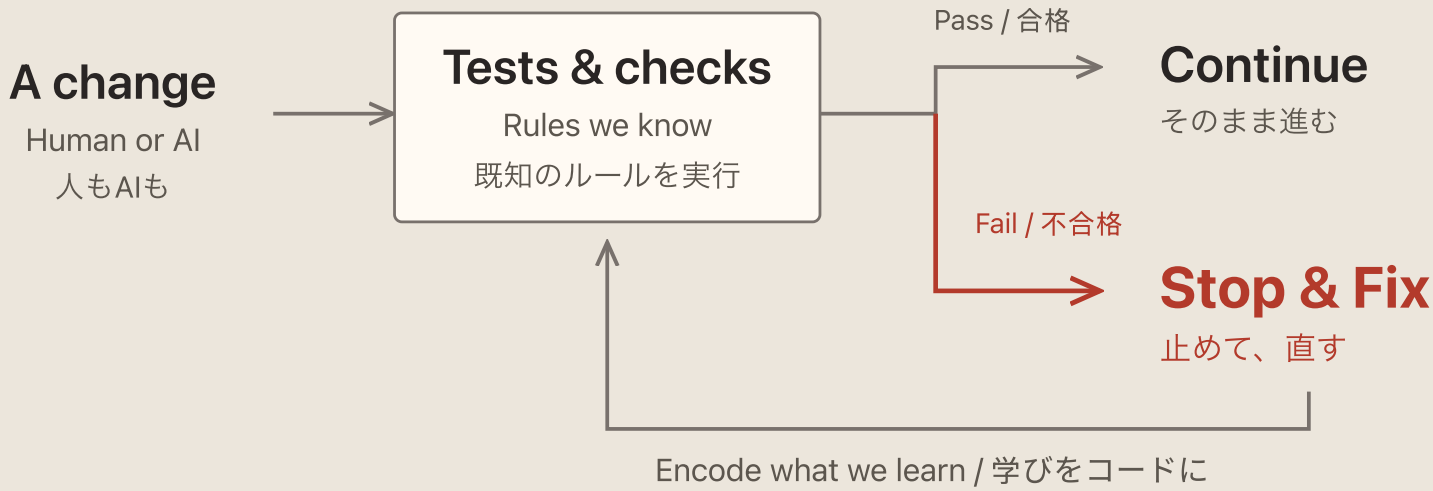
No joint to maintain

接合部も、その保守もなくす

AI-generated illustration

Build the stop into the software

ソフトウェアに「止まる仕組み」を組み込む



Fix the cause. Leave a safeguard.

原因を直す。次を守る仕組みを残す。

Jidoka frees people

自働化は、人を自由にする

Known rules
protect the product.

既知のルールが
プロダクトを守る。



Free to solve
the next real problem.

次の本当の課題に、力を使える。

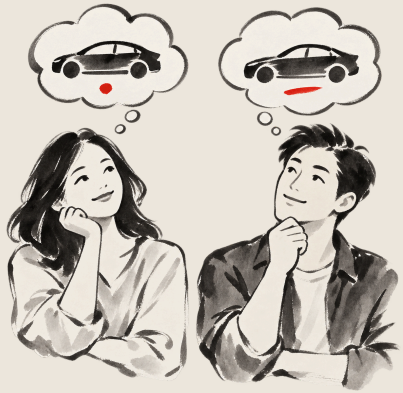


Just-in-Time

ジャスト・イン・タイム

Only what is needed

必要なものを



Order 2 cars

2台の注文

When it is needed

必要なときに

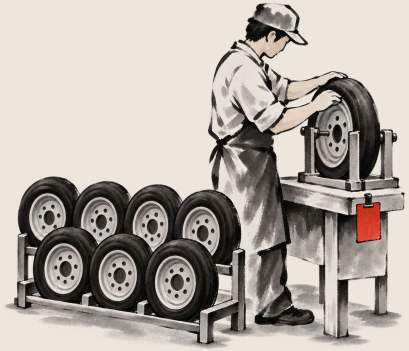


Pull 8 wheels

車輪8個を引き取る

The amount needed

必要な量だけ



Replenish 8 wheels

車輪8個を補充する

Minimum stock. Steady flow.

必要最小限の在庫で、流れをつなぐ。

Toyota: Toyota Production System

JIT entrusts people

JITは、応える仕事を人に任せる

ある物で工夫する

Be resourceful
with what you have.



Confidence comes from the ability to respond.

応える力が、任せる確信になる。

Terry's reading of JIT, inspired by Kazumasa Ebata's teaching

Pull: smaller customer problems

プル：顧客の問題を小さく分ける

“Get home after dinner.”

「食事のあと、家に帰りたい。」

Find the next train

次の電車を調べる

Check the fare

運賃を調べる

Find a step-free route

段差のない経路を探す



One useful outcome, end to end.

ひとつの役立つ結果を、エンドツーエンドで。

Freedom to choose again

次を選び直す自由

“Next train: 22:45.”

「次の電車は22:45。」

“Does that route have stairs?”

「その道、階段はある？」

Next: a step-free route

次は、段差のない経路



Keep completed value. Leave later stories unstarted.

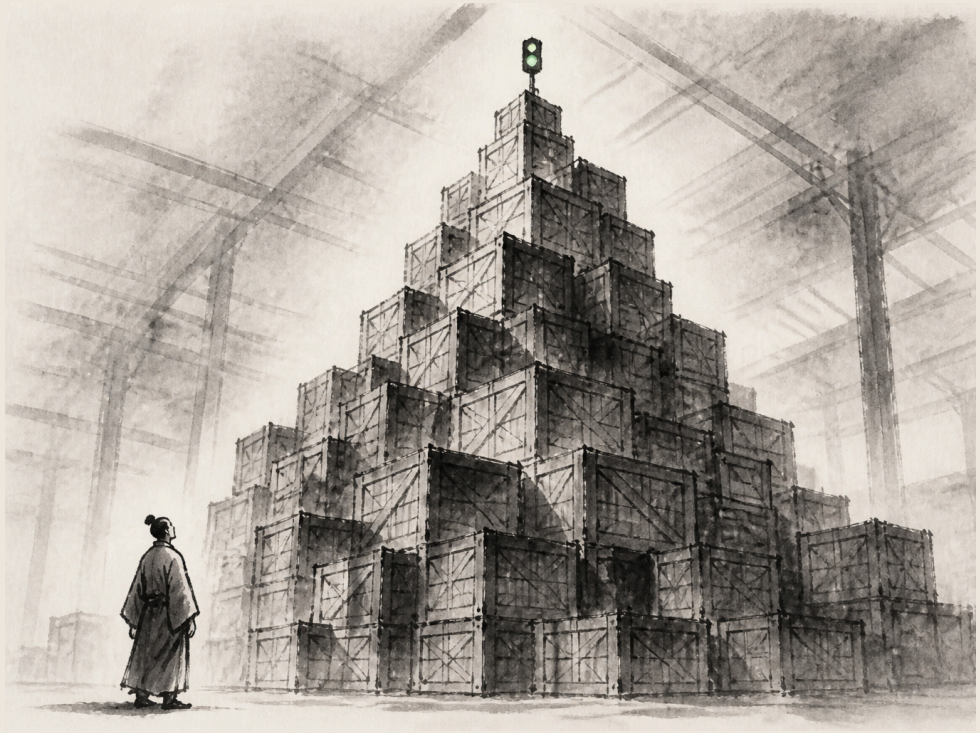
届けた価値は残す。後のストーリーは未着手に。

Continuous integration is a practice, not a system

継続的インテグレーションはシステムではなく、プラクティスだ

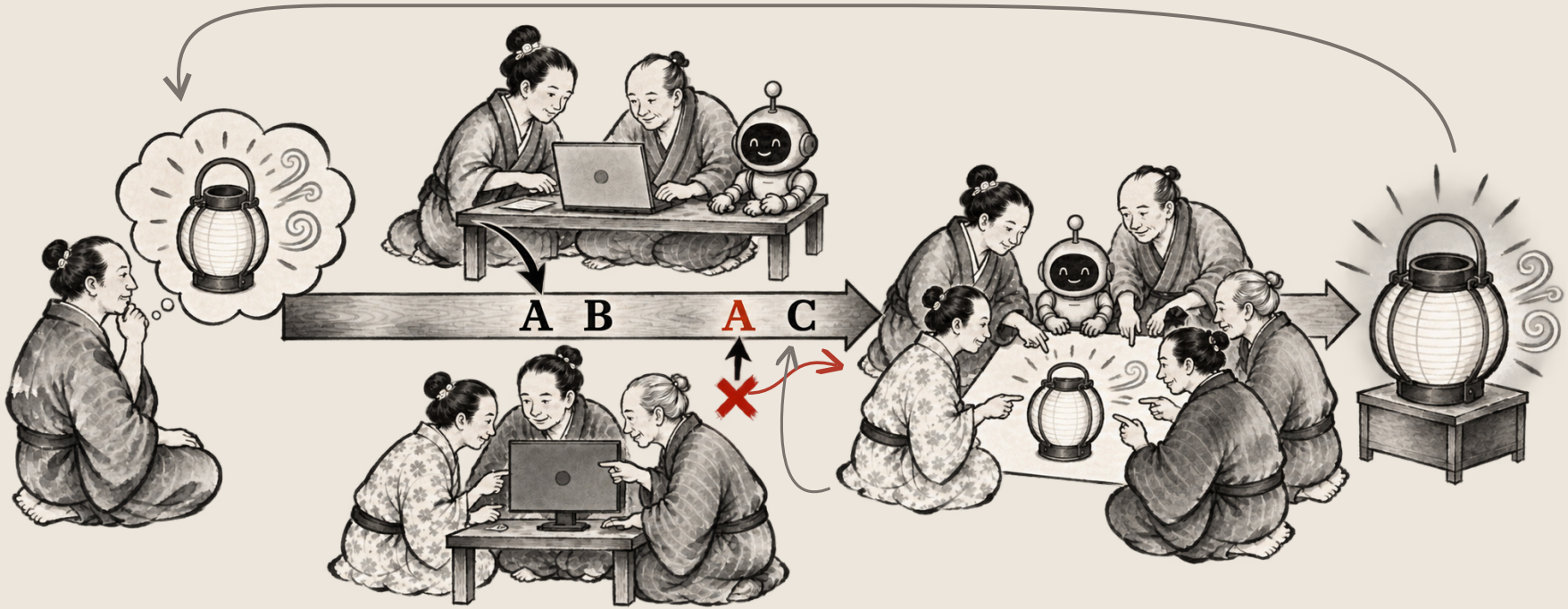
A CI server that integrates unowned branches is a stockpile with a **green light** on it.

誰も所有しないブランチを統合するCIサーバーは、緑のランプがついた在庫にすぎない。



Let the shared product pull collaboration

共有プロダクトに、協働をプルさせる

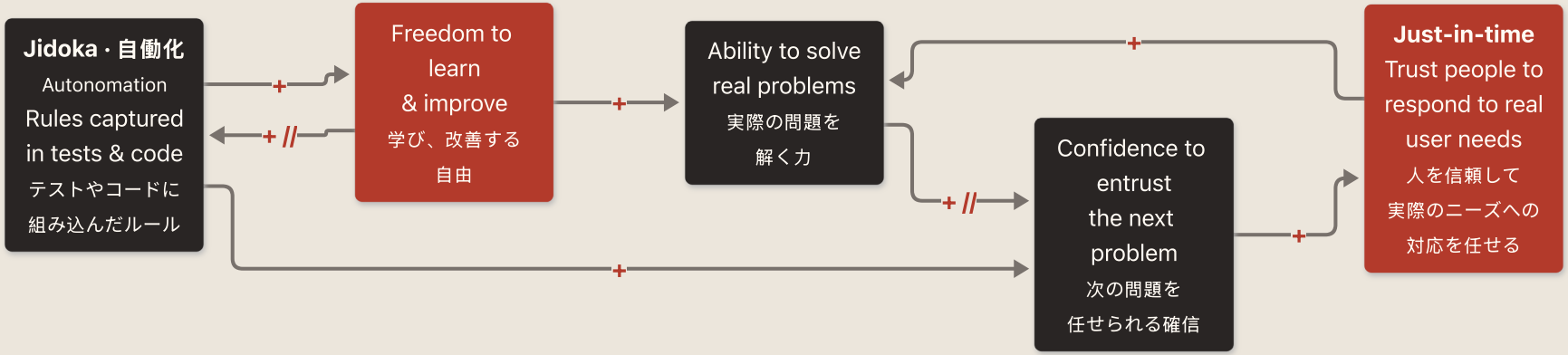


Integrate continuously. Collaborate just in time.

継続的に統合し、必要なときに協働する。

The engine of freedom and trust

自由と信頼のエンジン





+ increases · // takes time

＋ 増やす · // 時間がかかる

AI-generated illustration

Respect for People: making things means making people

人間性尊重：モノづくりは人づくり



Grow people who can think for themselves.

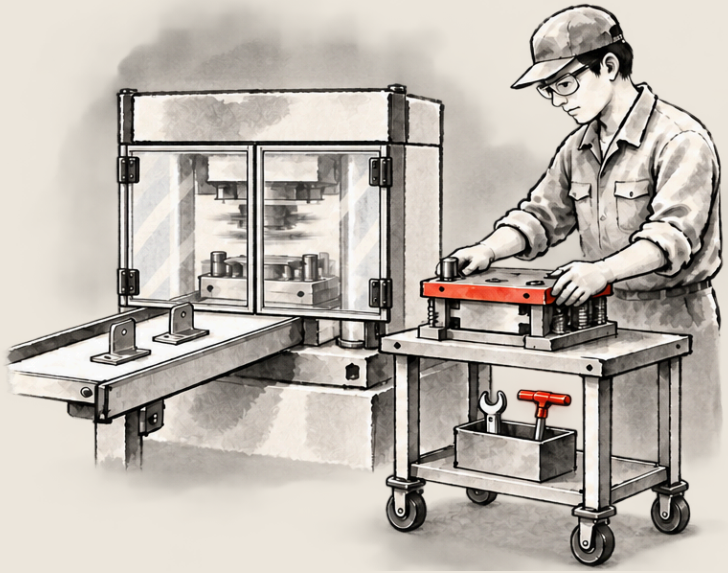
自ら考える人を育てる。

Continuous improvement towards perfection

完璧に向けた継続的改善

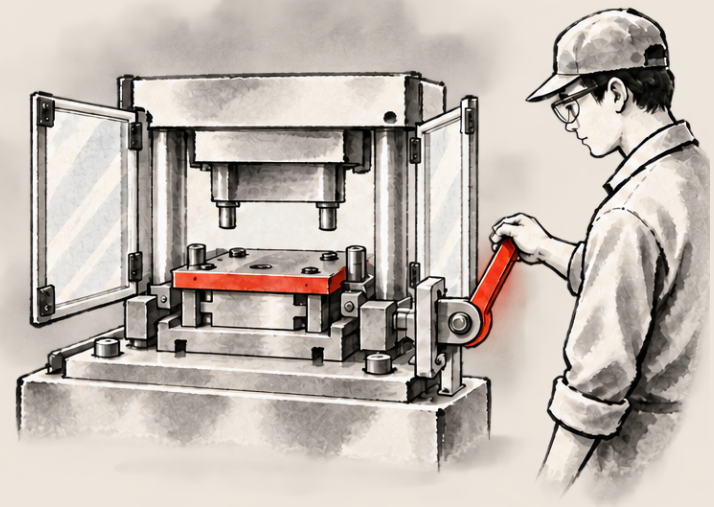
SMED · Prepare while running

稼働中に準備して、停止時間を短く。



OTED · One touch to change

残る段取りを、ワンタッチに。



Keep making the next change cheaper.

次の変更の負担を、減らし続ける。

Go-See may mean entering the AI harness

現地現物 (Go-See) とは、AIハーネスの中に入ることかもしれない

Do it yourself a thousand times.

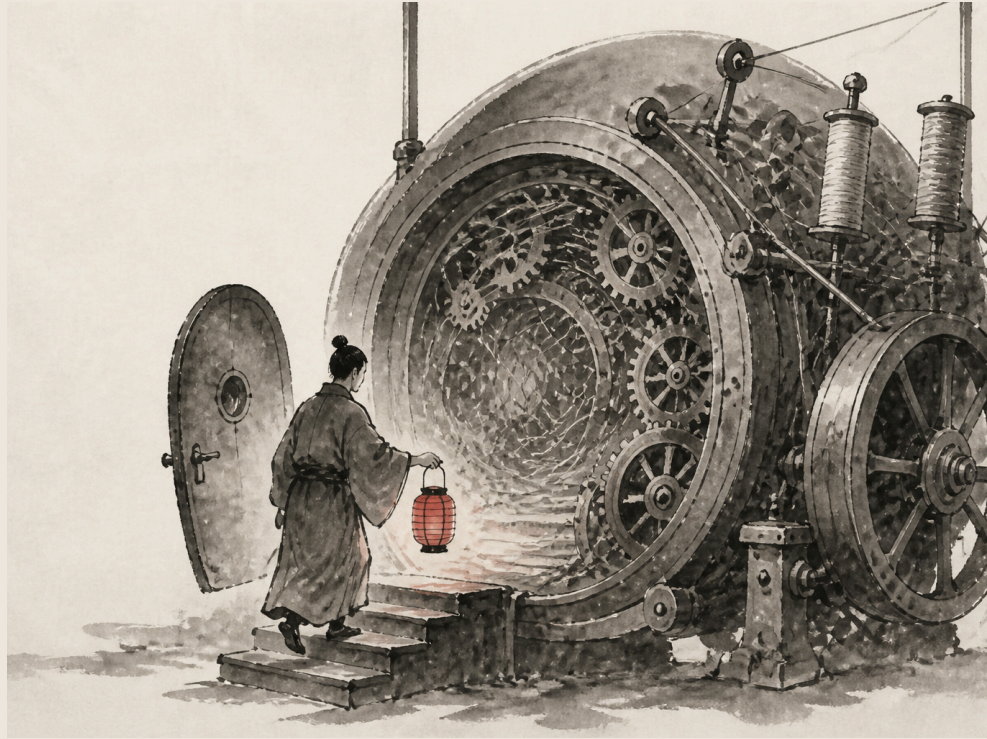
まず、自分で千回やる。

Know the consequences of locking it in.

何を固定するのか、その影響を知る。

Then entrust it to AI.

そのうえで、AIに任せる。



How do you know if the organization is using AI right?

組織がAIを正しく使えているか、どうすればわかる?

More freed than constrained by what they built.

チームが作ったものに、縛られるより解放されている。

Can we trust the team with the next real problem?

次の本当の課題を、チームに任せられるか。

Jidoka

自働化

Learning stays in the product.

学びが仕組みとして残る。

Just-in-Time

ジャスト・イン・タイム

Useful value now. Free to choose what's next.

価値を届け、次を選び直せる。

Respect for People

人間性尊重

People grow the ability to think for themselves.

自ら考える力が育つ。

Takeaways

持ち帰ってほしいこと

1. **Judge AI by freedom and capability.**
   自由と、応える力が増えているか。
2. **Pull small slices of real value.**
   本当のニーズから、小さく届ける。
3. **Smart → dumb → gone.**
   賢い → 単純 → 消える。
4. **Same gates for people and AI.**
   人にもAIにも、同じ基準。
5. **Integrate continuously.**
   Collaborate just in time.
   統合を続け、必要なときに協働する。



Odd-e



Build products that free people.

人を自由にするプロダクトをつくる。

Trust them with the next real problem.

次の本当の課題を、人に任せる。





Thank you

ありがとうございました

Terry Yin · Odd-e · terry@odd-e.com