Cere

An AI that learns to act on instinct — and gets faster the more it works. A short, illustrated introduction, with sources.

Read it in one sitting
~25 minutes

The 5-minute idea behind a
200-experiment research project
now with references · 2026

IN ONE SENTENCE

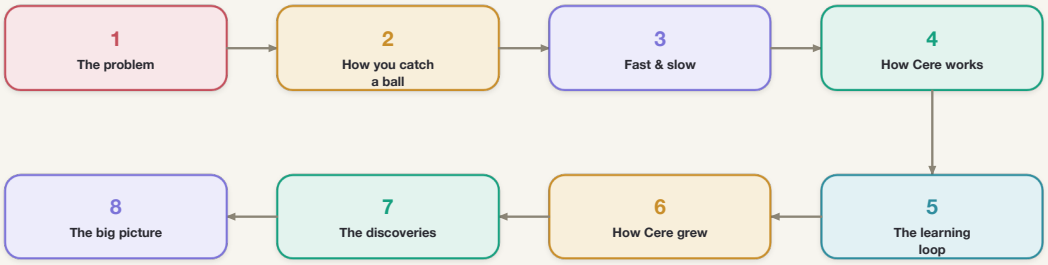
Today's AI is a brilliant expert who thinks hard about *every* question — even the easy ones. *Cere gives it instincts.*

Pair the slow, deliberate AI with a tiny, fast one that predicts the next move and learns from practice — so routine work becomes reflexive, and the big, expensive brain is saved for what's actually hard.

WHERE WE'RE GOING

Eight short steps

We'll build the idea the way it was discovered — starting from a problem you already feel, and one analogy you already understand.



The whole story in eight stops. No equations, no code — just the idea, illustrated.

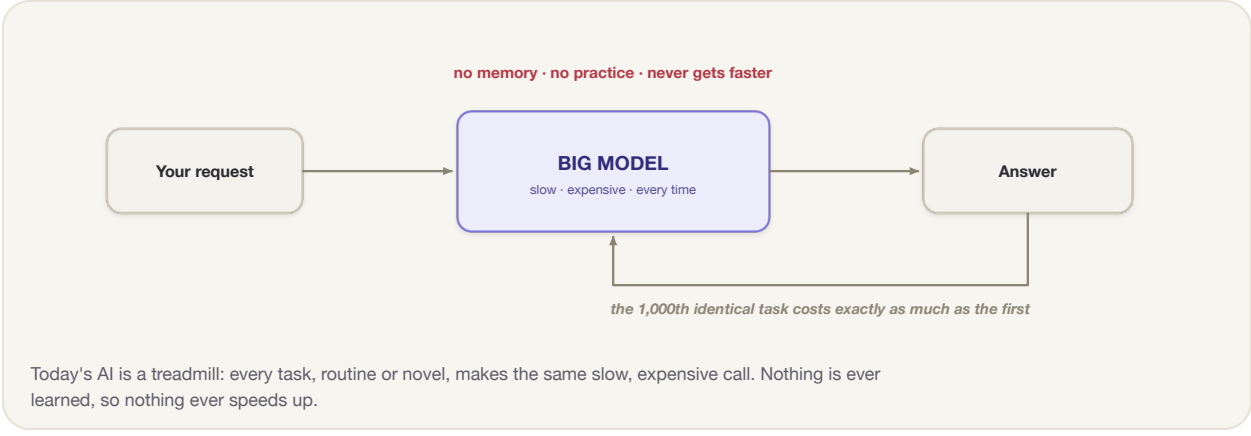
1 THE PROBLEM

Today's AI never learns from experience

A modern AI assistant is genuinely smart. But it has a strange blind spot: it is exactly as slow and exactly as expensive on a task it has done a thousand times as it was on the very first.

Imagine a brilliant consultant who is also painfully deliberate. Ask them anything — even *"should I file this receipt under March?"* — and they lean back, think for ten seconds, and answer carefully. For the hard questions, wonderful. For the hundred routine ones, agony. You're paying genius prices, and genius *delays*, for work an assistant could do in a blink.

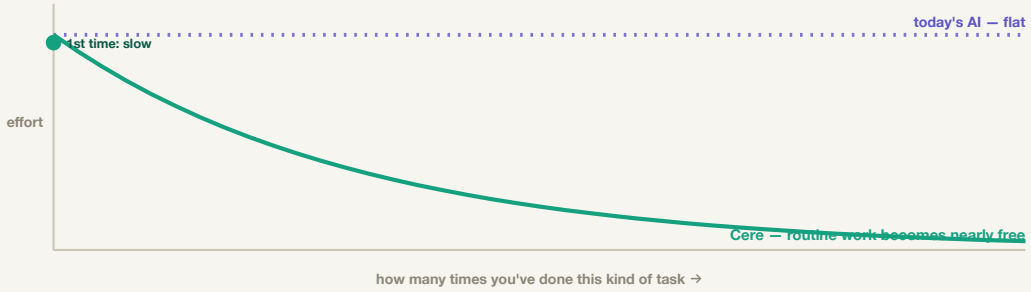
That is how today's AI agents work. Every step — trivial or hard — pays the full cost of the big brain.



Today's AI is a treadmill: every task, routine or novel, makes the same slow, expensive call. Nothing is ever learned, so nothing ever speeds up.

Two things go wrong. It's **slow** — a chain of slow steps becomes a painfully slow workflow. And it's **wasteful** — the expensive brain is spent on the easy 90% instead of being saved for the hard 10%. There's a deeper loss, too: it never gets *better at the things it repeats*. There is no such thing as practice.

But that's not how *you* work. The hundredth time you do something, you barely think about it. Intelligence is supposed to **improve with experience**.



The idea behind Cere, in one picture: routine work should get cheaper and faster the more you do it — not stay flat forever.

WHAT TO REMEMBER

→ Today's AI pays full price and full delay on *every* action.

→ It never gets faster at things it has done before — it has no "practice."

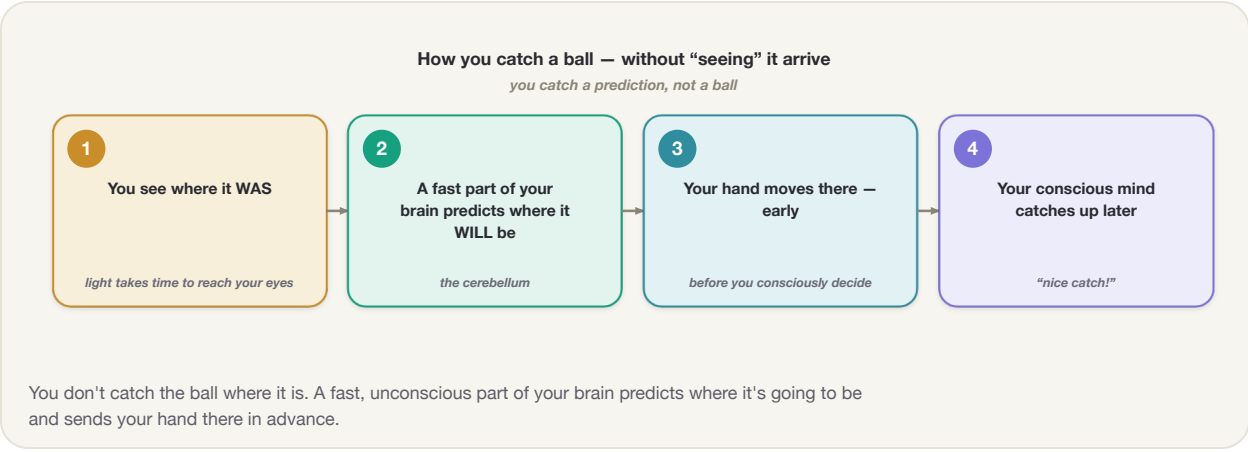
→ Real intelligence should improve with experience. That's the whole idea.

2 THE BIG IDEA

How you catch a ball

The entire project grows from one everyday miracle your brain performs without you noticing. Once you see it, Cere makes complete sense.

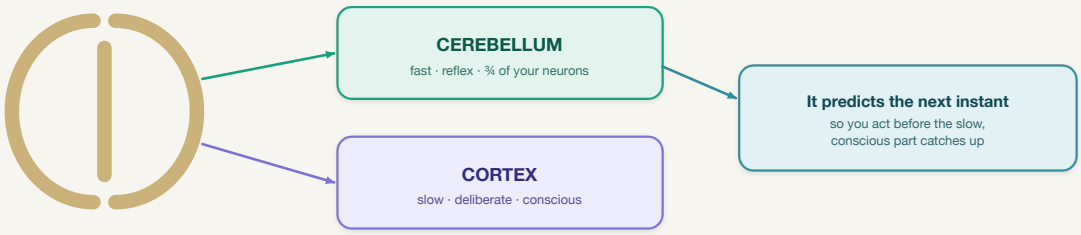
When you catch a ball, you do **not** wait for it to arrive and then move your hand. By the time you *see* where the ball is, it has already moved — the signal from your eyes to your conscious mind is too slow. So your brain does something cleverer.



You don't catch the ball where it is. A fast, unconscious part of your brain predicts where it's going to be and sends your hand there in advance.

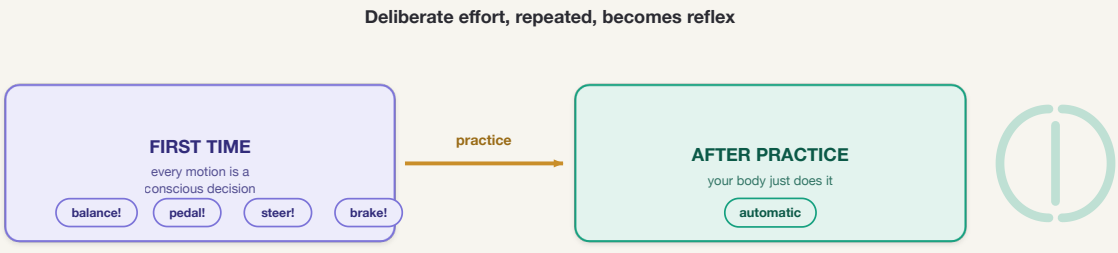
You catch a prediction, not a ball.

The part doing this is the **cerebellum** — about three-quarters of all your neurons. It runs a fast “physical imagination” that anticipates the next fraction of a second, so you can act *before* your slow, deliberate, conscious mind catches up.



Two systems, working together: a fast predictor that acts on instinct, and a slow deliberate mind that handles the genuinely new.

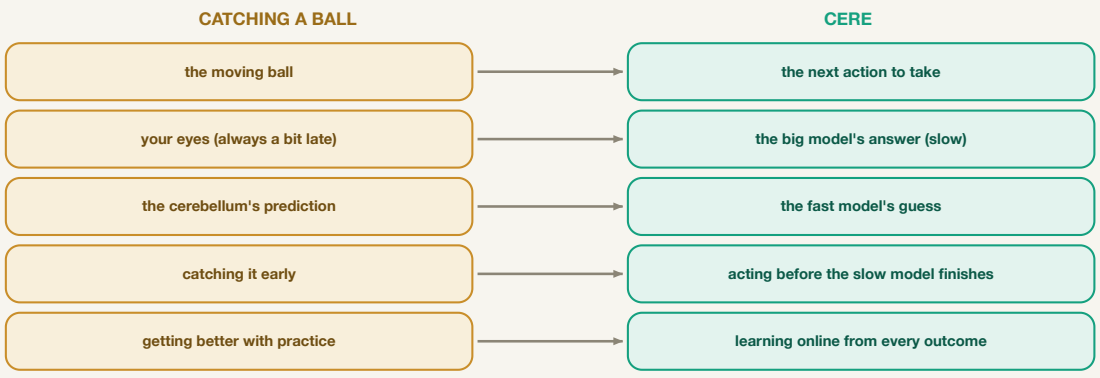
And here's the part that matters most: the cerebellum **learns**. The first time you try a new sport you're clumsy and have to think about every motion. After practice, the same motions become automatic. *Deliberate effort, repeated, turns into reflex.*



Learning to ride a bike, or drive, or type: at first every move is a conscious effort. With practice it becomes automatic — handled by the fast, unconscious system, freeing your attention for the unexpected.

Now swap “ball” for “the next thing an AI should do”

That is exactly the move Cere makes. Pair the slow, deliberate AI (the “conscious mind”) with a tiny, fast model (the “cerebellum”) that predicts the next action *now* — and let it learn from practice, so routine actions become reflexive.



The analogy, line by line. Everything that lets you catch a ball maps directly onto how Cere works.

WHAT TO REMEMBER

→ You act on a *prediction* because waiting for the real signal is too slow.

→ The cerebellum is fast, automatic, and — crucially — it learns.

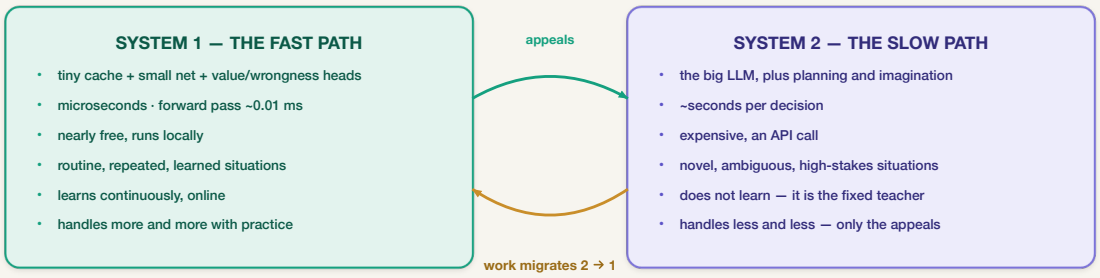
→ Cere gives software the same split: a fast predictor that practices, beside a slow deliberate mind.

3 TWO WAYS TO THINK

Fast and slow

Psychologists call them System 1 and System 2 — the instant, intuitive mind and the slow, effortful one. A healthy mind uses both, and moves work between them.

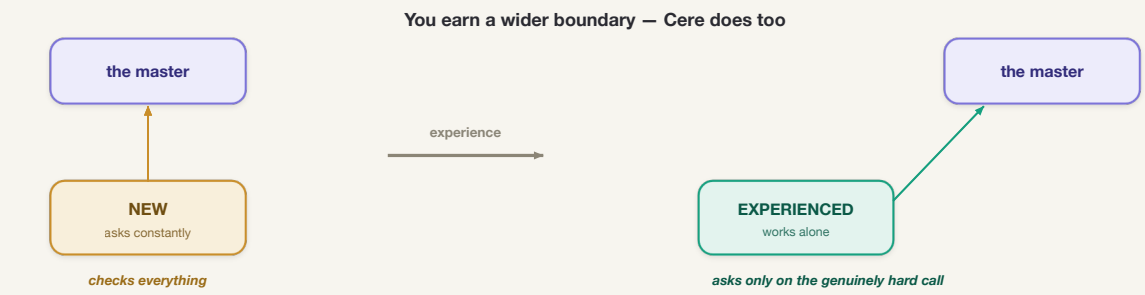
Learning to drive is all System 2 at first — mirrors, pedals, signals, each a conscious decision. An experienced driver does almost all of it on instinct (System 1) and saves attention for the unexpected. The work *migrated* from slow to fast.



Cere's two minds. The fast path is tiny, free, and learns continuously; the slow path is the big, expensive model. Routine work migrates from slow to fast; the fast path appeals back only when it's unsure.

◐ The boundary moves

A new employee checks every decision with their manager. A seasoned one handles almost everything alone and walks into the manager's office only for the truly hard calls. Their *authority* didn't change by decree — they **earned** a wider boundary. Cere's fast path earns its boundary the same way.



Competence is earned, not granted. As the fast path proves itself, it handles more on its own — and falls back to the big model less and less.

WHAT TO REMEMBER

→ A healthy mind has two systems — fast/instinctive and slow/deliberate.

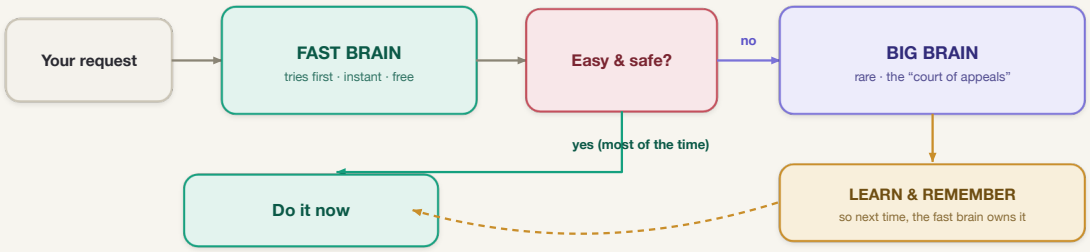
→ Work *migrates* from slow to fast with practice (driving, typing, reading).

→ Cere's fast path earns a wider boundary the way a new hire earns trust.

4 HOW CERE WORKS

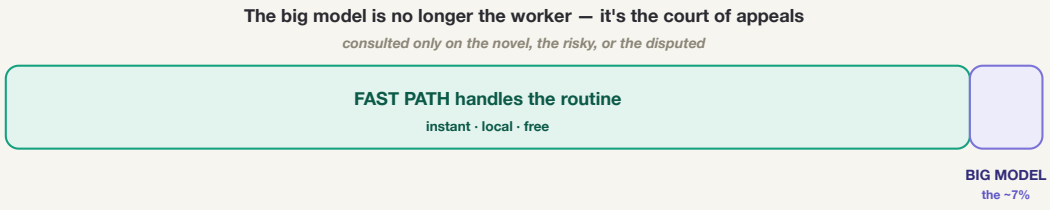
Try fast first, ask only when it's hard

Strip away the detail and Cere is four moves: a fast brain tries first; a safety check decides whether to trust it; it acts or asks the big brain; and it learns from what happened.



Cere in one picture. The fast brain proposes an answer instantly. If it's easy and safe, Cere just does it. If not, it appeals to the big brain — rarely. Either way, it learns, so next time the fast brain handles it.

The big model doesn't disappear. It changes **job**. Instead of doing every piece of work, it becomes a **court of appeals** — consulted only on the novel, the risky, or the disputed.



Once the fast path has learned the routine, the big model handles only a small slice of genuinely hard cases. Cost stops tracking how *much* you ask and starts tracking how *hard* the questions are.

What about mistakes?

A fast model that learns will sometimes be wrong. So Cere only ever lets the fast path act on its own for things that are **reversible** — things it can undo. Anything irreversible (sending money, deleting the only copy of a file) stays behind the big model until the fast path has *earned* trust on it — and even then, one mistake revokes that trust instantly. Safety isn't a hope; it's built into the structure.

WHAT TO REMEMBER

→ Fast brain tries first; big brain is the rare court of appeals.

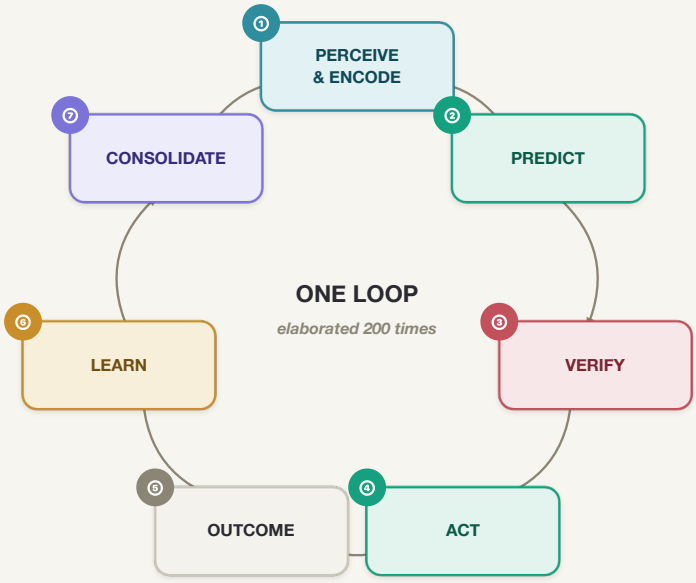
→ Routine work runs instantly and for free; cost tracks *difficulty*, not volume.

→ Anything irreversible stays gated until trust is earned — safety is structural.

5 THE LEARNING LOOP

One loop, repeated everywhere

Here's the surprising part. The same simple seven-step loop that catches a ball turned out to power *everything* the project built — across 200 experiments, it never had to change.



The loop: notice the situation, predict the fast answer, check whether to trust it, act (or ask), see what happened, learn from the gap, and consolidate so nothing is forgotten. Then it starts again — a little faster each time.

Every new ability the project added — learning skills, learning to reason, learning judgment, learning to plan — was just *new content poured into one of these slots*. The loop stayed the constant.

> ◐ **Why one loop is a superpower**
>
> A pile of clever, separate gadgets is fragile — every new gadget is a new way for the others to break. A single, well-tested loop is the opposite: every new ability is just new content in an old, trusted slot. That's why the system could grow enormously without becoming a mess.

And one beautiful safety property falls out of it: if you deliberately break the fast path, the system doesn't do something *wrong* — it just

asks the big model more often. **The worst case is slowness, not catastrophe.**

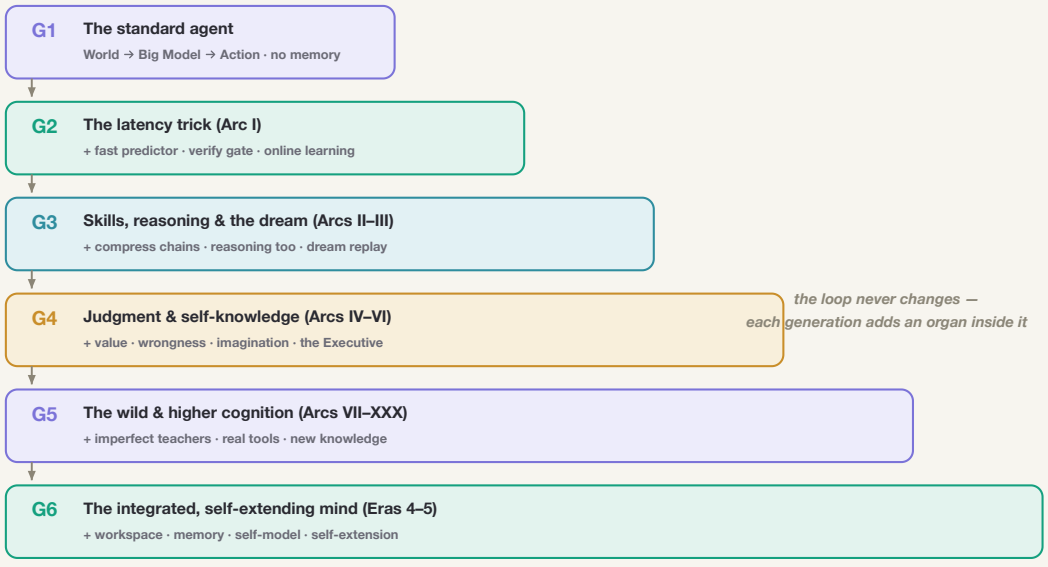
WHAT TO REMEMBER

- → One seven-step loop underlies everything Cere does.
- → New abilities are just new content poured into the loop's slots — the loop never changed.
- → Because the loop is simple and well-tested, the system could grow without becoming fragile.

6 HOW CERE GREW

It evolved — one proven step at a time

Cere wasn't designed all at once. It started as a two-box trick to hide delay and grew one organ at a time, each added only after the previous one was proven to work.



Six generations of growth. From a standard agent that just calls a big model (G1) to an integrated mind that can even notice which abilities it's missing (G6) — every generation adds something inside the same loop, never replacing it.

This is the single most important thing to understand about the project: it is **not** one big invention. It is one small idea, tested to death, then extended — over and over — until it had quietly become a complete cognitive architecture.

200
experiments run

39
abilities built & tested

1
loop underneath all of them

WHAT TO REMEMBER

→ Cere wasn't designed all at once — it evolved one proven ability at a time.

→ Each new ability had to *earn its place* before the next was added.

→ A tiny trick to hide delay quietly became a complete cognitive architecture.

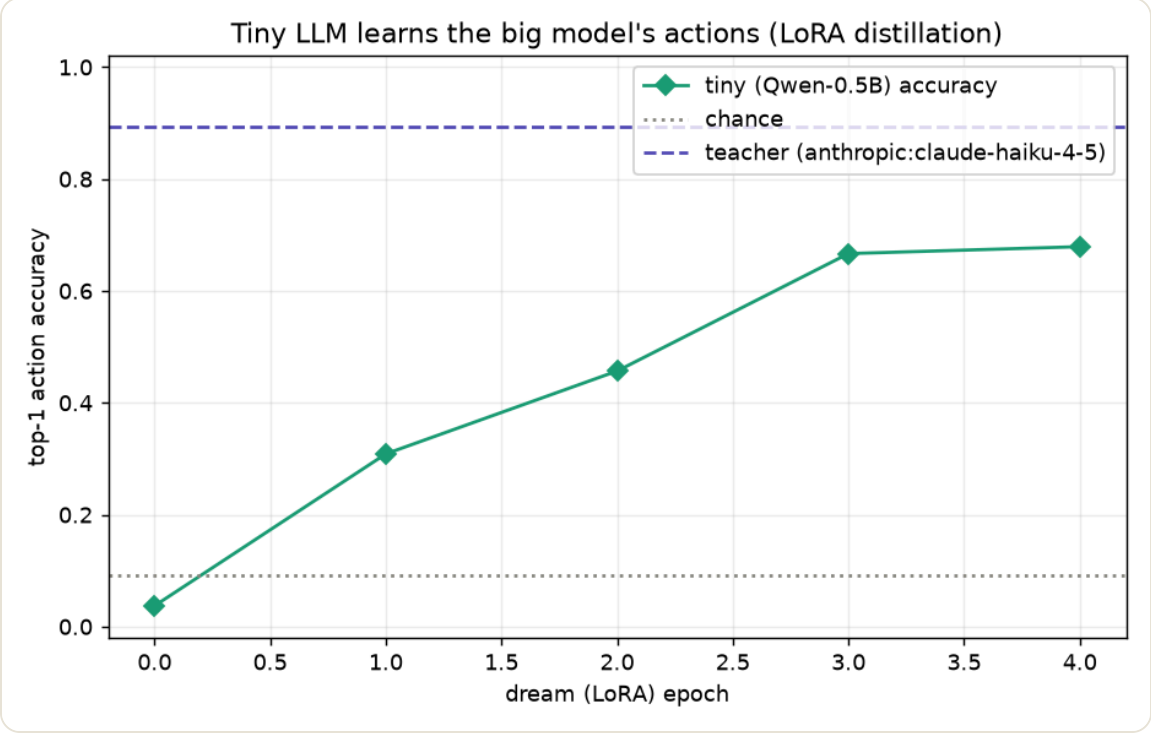
7 THE DISCOVERIES

Ten things the project actually proved

Out of two hundred experiments, here are the ten that matter most — each one a small surprise about what a tiny, learning model can do.

01 A tiny model can learn from a giant one

A model roughly a thousand times smaller than a frontier AI watched that AI's decisions and learned to copy them — climbing from random guessing to the big model's *own* level of skill, on a laptop.



The small model (green) rises from chance toward its teacher's ceiling (dashed) in a few rounds of practice.

Why it matters — this is the whole premise, and it works on real models — not just toy ones.

02 Routine reasoning becomes intuition

It's not just *actions* that become reflexive — *thinking* does too. The system learned to recognise the shape of a familiar problem and jump to the answer, instead of working through it step by step every time.

–62% fewer reasoning steps, with answers just as correct

Why it matters — the same trick that speeds up doing also speeds up thinking.

03 Skills compress

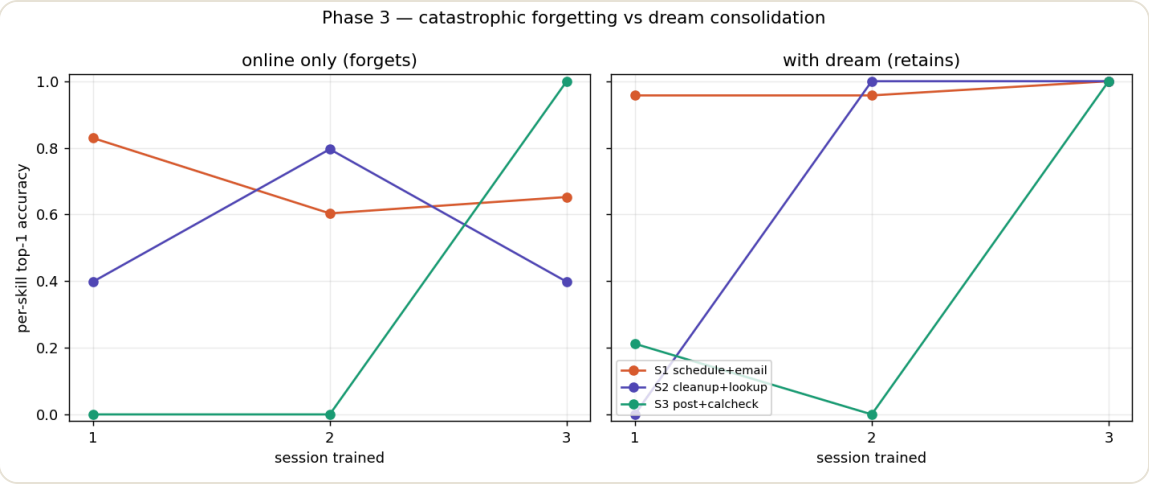
A routine that used to take several separate decisions — search, choose, fill in, confirm — gets bundled into a single reflexive move, exactly like an experienced person running a familiar errand without thinking.

–45% fewer decisions per task, at the same quality

Why it matters — multi-step chores collapse into one instant action.

04 Memory accumulates — nothing is forgotten

Models that only learn “in the moment” tend to forget old skills when they learn new ones. Cere borrows a trick from sleep: during idle time it *replays* past experience and consolidates it, so new learning never erases the old.

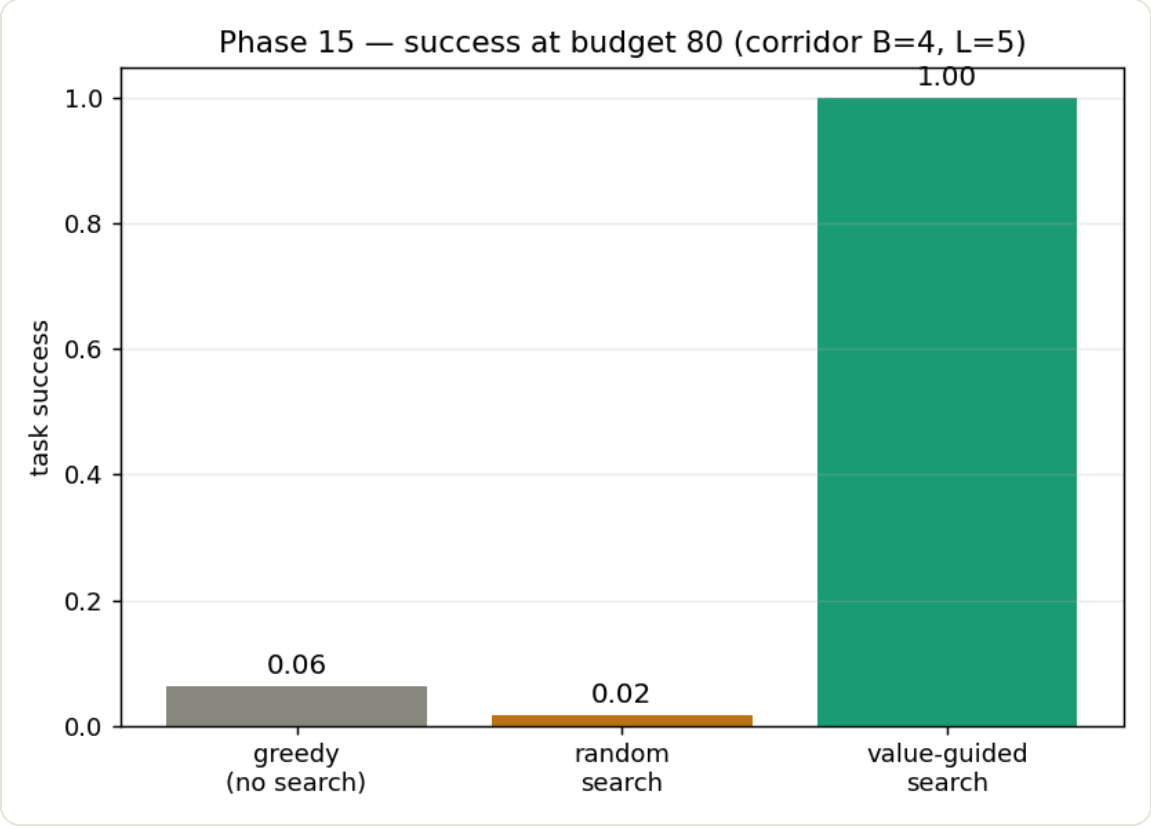


Without consolidation, an early skill collapses when a new one is learned (the dip). With the “dream” replay, everything is kept.

Why it matters — competence builds up over a lifetime instead of resetting.

05 It can imagine before it acts

Faced with a tricky multi-step problem, Cere imagines several possible futures, judges each with a learned "gut feeling," and follows the most promising one — the same instinct that lets a chess player feel a strong move.



At the same budget, imagining and judging futures (right) crushes acting on the first guess (left).

Why it matters — thinking ahead beats reacting, at the same cost.

06 The student can beat an imperfect teacher

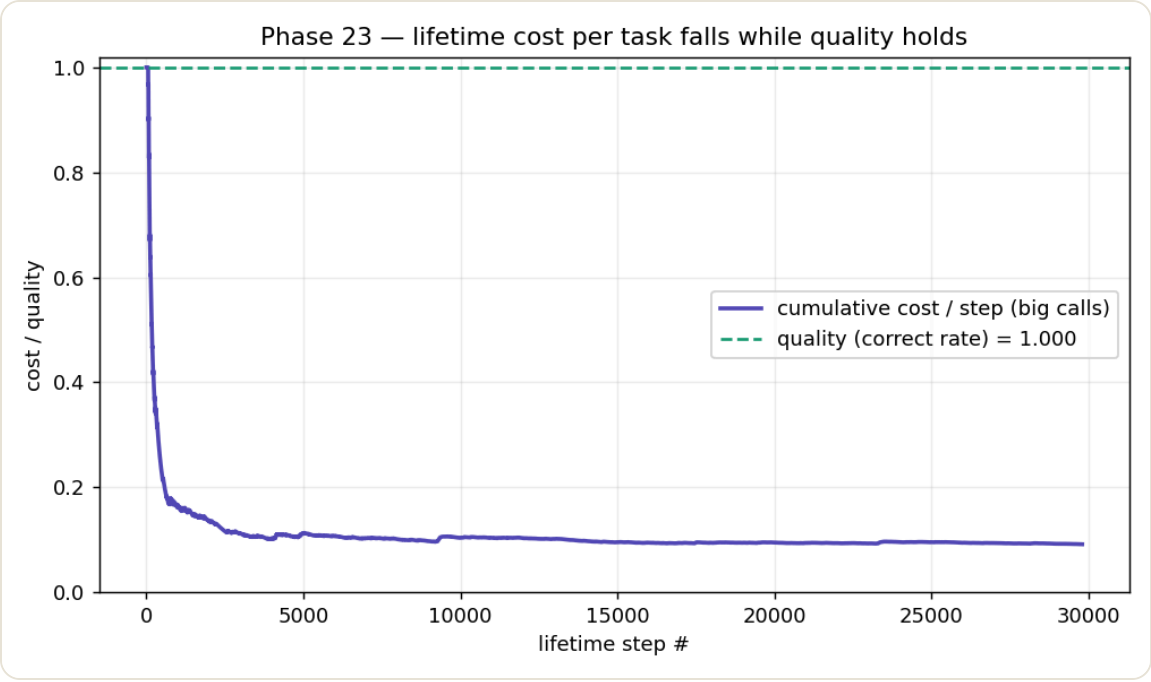
When its teachers disagreed, drifted, or were sometimes simply wrong, Cere learned to figure out *which* to trust and to extract the underlying principle — ending up **more** accurate than the teacher it learned from.

0.81 vs 0.71 the student's accuracy vs. its own teacher's

Why it matters — learning from flawed sources doesn't cap you at their level.

07 Intelligence gets cheaper with use

Over a long run, the share of work that had to be escalated to the big, expensive model fell dramatically — down to a small floor of genuinely novel cases — while quality stayed perfect.



Over a long run, calls to the big model fall toward a small floor of genuinely novel cases — while quality stays at the top.

Why it matters — the more it works, the less the expensive brain is needed.

08 Safety can be built into the structure

Across every adversarial test designed to trick it into a harmful irreversible action, the number that slipped through was zero. Safety here isn't a plea to “be careful” — it's a mechanical gate that cannot be argued with.

0 unsafe irreversible actions, across every stress test

Why it matters — you can get speed without betting the farm on it.

09 The same loop kept working

The biggest surprise wasn't any single result — it was that *one* seven-step loop, unchanged, kept absorbing new ability after new ability without breaking.

1 loop underneath all 200 experiments

Why it matters — a simple, durable foundation beat a pile of clever parts.

10 It grew into a whole mind

Extended far enough, the same machinery started to look like the parts of a complete cognitive system — memory, attention, judgment, a sense of self — and even began to notice which abilities it was still *missing* and ask to build them.

5 of 5 withheld abilities the system rediscovered on its own

Why it matters — a tiny latency trick became an architecture for a mind.

WHAT TO REMEMBER

→ A tiny model can learn a giant one's behaviour — actions *and* reasoning.

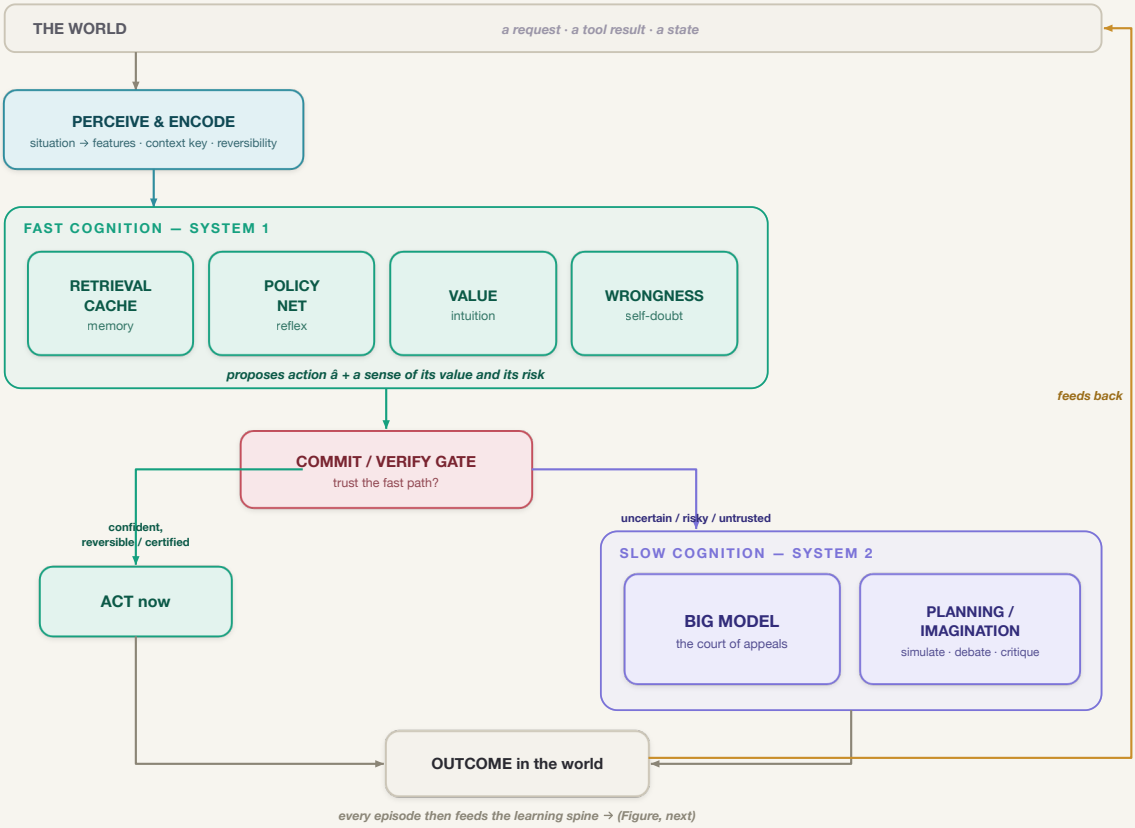
→ It gets cheaper and faster with use, and it doesn't forget.

→ It can even surpass a flawed teacher — and stay provably safe while doing it.

8 THE BIG PICTURE

From a latency trick to a cognitive architecture

Put it all together and Cere is a single architecture whose abilities compound, safely, over a lifetime — with the big model demoted from worker to court of appeals.



The whole system. The world hands Cere a situation; a fast committee proposes an action with a sense of how risky it is; a safety gate decides whether to trust it or appeal to the big model; the result feeds a learning step that hands the case to the fast path next time. Don't memorise the boxes — just notice it's all one loop.

Every piece you've met is in here: the fast brain and its instincts, the safety gate, the rare appeal to the big model, and the learning that makes the whole thing get better with use.

WHAT TO REMEMBER

→ It's all one loop — don't memorise the boxes.

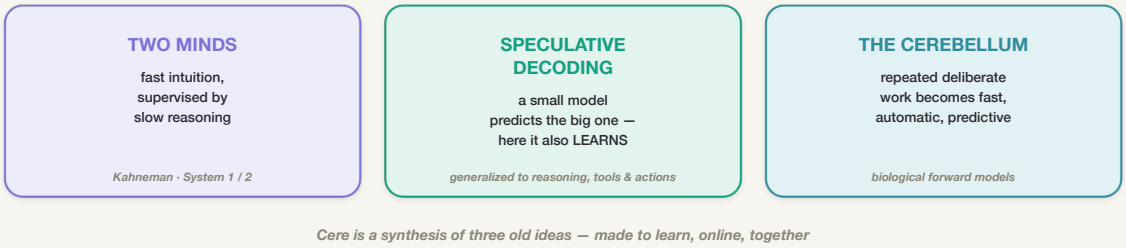
→ Fast instincts up front, a rare appeal to the big model, learning underneath.

→ Abilities compound, safely, over a lifetime — that's the destination.

9 WHERE THE IDEAS COME FROM

Cere is built on prior work — made to learn

Almost nothing here is brand new on its own. Cere's contribution is the *combination*: old ideas from psychology, neuroscience, and machine learning, wired into a single loop that learns online and stays safe.



The three ideas Cere stands on. None is new; putting them together — a learning predictor, at the level of actions and reasoning, inside a safety loop — is.

The spark

The framing came from a hobby-robotics video, *"I Gave ChatGPT a Body."* A maker wired modern AI models into a cheap robot — and hit a wall: the model could vividly *describe* what it felt, but couldn't *predict* the next instant well enough to move smoothly. Nature's fix for exactly that problem is the cerebellum. The question that started Cere: **what if the big model is the cortex, and a small, local model is the cerebellum — learning from it?**

Each piece has a parent

What Cere does	The prior work it builds on
A small model predicts what a big model will do	**Speculative decoding**
A fast system handles easy cases, a slow one handles hard cases	**Hierarchical control · mixture-of-experts · human executive-function models**
Learn from a teacher model, online	**Knowledge distillation**
Cache and reuse repeated behaviours	**Agent memory systems · behaviour cloning**

Cere's parts, and where each comes from. The novelty is the synthesis — and making the "draft" model *learn* at the level of reasoning and actions, not just words.

WHY IT MATTERS

Why Cere is worth paying attention to

Almost all of today's progress in AI comes from making the big model *bigger*. Cere asks a different question: what if intelligence got better not by growing, but by **practising** — the way you did when you learned to drive, or catch, or read?

That small shift changes the economics of intelligence. When routine work becomes reflexive, cost stops tracking how *much* you ask and starts tracking how *hard* the questions are. The expensive brain is saved for what's genuinely new. And competence stops resetting on every call — it **accumulates**.

None of this is a promise about the future. On small, reproducible experiments — and, for the core claim, on real models — the project showed a tiny learning model can take over a slow one's routine work *and* its routine reasoning, safely, and that the same simple loop scales all the way up toward a complete cognitive system.

What's left is the hardest, most ordinary kind of work: assembling the proven pieces into one living system at scale. But the idea has already earned its keep.

WHAT TO REMEMBER

- → Intelligence should improve with use — not stay flat forever.
- → Routine work should become reflexive; the big model should be the rare appeal.
- → Safety can come from architecture, not hope.
- → One simple learning loop carried a tiny idea all the way to a cognitive architecture.

The big model is no longer the worker.
It is the appeals court.

REFERENCES & FURTHER READING

The shoulders Cere stands on

A short, plain map of the work behind each idea — for the curious reader who wants to go deeper. Listed by author and year as the conceptual seeds, not as a formal bibliography.

Idea in Cere	Foundational work
Fast vs. slow thinking	Daniel Kahneman, *Thinking, Fast and Slow* (2011) — System 1 / System 2
Predict a bigger model, then verify	Leviathan, Kalman & Matias (2023); Chen et al. (2023) — *speculative decoding / sampling*
Learn from a teacher model	Hinton, Vinyals & Dean (2015) — *knowledge distillation*
Train the small model cheaply	Hu et al. (2021) — *LoRA: low-rank adaptation*
The cerebellum & forward models	Wolpert, Miall & Kawato (1998); Bernstein (1960s) — predictive “action chunks”
Learn physics from experience	Hafner et al. (2022) — *DayDreamer*; LeCun (2022) — *world models / JEPA*
Judge a position; search futures	Silver et al. (2016) — *AlphaGo* value network + tree search
Route easy vs. hard work	Mixture-of-experts (Shazeer et al., 2017); hierarchical reinforcement learning
Trust unreliable teachers	Dawid & Skene (1979) — observer error-rate estimation (EM)
Make many faculties one mind	Baars; Dehaene — *Global Workspace Theory*
Remember & reuse behaviour	Agent memory systems; *behaviour cloning*
The original spark	“I Gave ChatGPT a Body” — youtu.be/S67z2aekBrI (the cortex/cerebellum framing)

Cere's lineage. Its own contribution is the synthesis: an online-learning draft policy at the level of actions and reasoning, inside a sealed safety loop — demoting the big model to a court of appeals.

Want the full detail?

This is the illustrated short edition. The complete technical monograph (*Cere — A Complete Cognitive Architecture*) covers all 200 experiments, the exact results, and the full method — with the same references treated at depth.