AUGELA
The Human AI Interface

THE AI PLAYBOOK FOR BUSINESSES

Training humanoid robots for your company

A humanoid arrives knowing how to walk. It does not know your company. How to correct a robot's mistakes at the source, and turn what your floor learns into permanent company knowledge.



CONTENT

· Why a robot arrives capable, but doesn't know your business
· The three mistakes — and which are yours to fix
· The loop, now for machines with hands
· Corrections vs. discoveries, captured off the floor
· What to do before the first robot arrives

WHY THIS MATTERS NOW

Every deployment has two very different layers.

Humanoid and mobile robots are moving from demos into real operations. Accenture, SAP and Vodafone have piloted humanoids in a Duisburg warehouse. BMW is piloting Hexagon's AEON at its Leipzig plant. Agility Robotics' Digit is operating at GXO. The companies running these pilots are discovering something familiar from AI, only faster and more expensively: the machine may be capable. It still does not know your business.

THE VENDOR'S PROBLEM

The body

Balance, grasping, navigation, perception, physical control. The manufacturer is responsible for making it reliable enough for the intended task. This is what the industry competes on.

YOUR PROBLEM

The job

What "fragile" means in your warehouse, which pallets go to which dock, which exception applies on a Friday. That second layer is where your company-specific knowledge lives.

ONE NOTE BEFORE YOU START

None of this works on a robot straight out of the box. It needs the robot's own operations layer, an SDK, or a systems integrator's build to expose a way to get in. A showcase unit running predefined routines has nothing to build on. If you're evaluating vendors, ask directly whether operational knowledge can be corrected without touching firmware. That's the real product question, not the walking.

BEFORE YOU FIX ANYTHING

"The robot got it wrong" is three problems, not one.

Mixing them up is the reason pilots stall.

A robot did the wrong thing on the floor. Which of the three was it?

CAPABILITY

It physically could not do the thing reliably. Wrong grip, wrong angle, wrong aisle. **Not yours to fix.**

VENDOR TICKET →

KNOWLEDGE

It did exactly what it was told, based on information that was out of date or wrong. Sound reasoning, stale inputs.

FIX THE SOURCE →

RULE

Its information was correct and it still did the wrong thing, because a policy nobody wrote down.

WRITE THE RULE →

EXAMPLE	RETURNS INTAKE FLOOR
Trigger	Sorts by the branded box a parcel arrives in are sometimes wrong, because customers reuse boxes. A return in a printer box is not a printer.
Looks like	A knowledge mistake because the box doesn't match the contents.
Actually is	**A rule mistake.** The label matters more than the box. Everyone on the floor knows it. It was never written down, so the robot doesn't.

THE LOOP

Three steps. Now for machines with hands.

Each step makes the next one work better.
And the third feeds back into the first.

1

Ground the robot

Stop it inferring your business from what it can see. Connect SOPs and site layout, define its scope, decide the fallback.

SOPs · Scope · Fallback

2

Govern what it does

Make it something you can approve, not supervise personally — because you can see and control what happens.

Roles · Guardrails · Audit trail

3

Compound what it learns

Every correction and discovery becomes permanent, shared knowledge — reviewed by a human, placed at the right level.

Flags · Elevation · Escalation

↑ STEP 3 BECOMES KNOWLEDGE FOR STEP 1

STEP 1 – GROUND THE ROBOT

Goal: stop the robot inferring your business from what it can see.

A robot operating on perception alone is improvising a policy from visual evidence, every time, and it will improvise differently tomorrow.

CONNECT	SOPs, site layout, handling exceptions, safety instructions, client rules, usually already written, usually never read after induction week.
DEFINE SCOPE	What the robot is for, and what it is not for, in plain language a supervisor can read. This is the cheapest failure signal you'll get.
SET FALLBACK	Stop, ask, escalate, or route to the manual lane. A robot that infers a rule moves physical objects – worse than one that asks only.

START WITH ONE UNIT, ONE SHIFT, ONE TASK

The rules you discover in the first fortnight will be wrong in interesting ways, and that is the point. Give the machine a small piece of the company, and make sure it can learn that piece without inventing its own version of the truth.

STEP 2 – GOVERN WHAT'S ALLOWED TO DO

Goal: make robots something you can approve, not supervise personally.

On a floor, governance is the thing that lets you leave the room.
Set up before the first unit is commissioned, not after.

ROLES

Picker flags · reviewer diagnoses · tutor decides the rule. Keeping these separate stops a shift supervisor's preference from quietly becoming company policy.

GUARDRAILS

What the robot must always follow, where a human is required, and what actions need approval before they happen.

AUDIT TRAIL

What it was told, how it was used, how confident it was, which human was accountable. "The vendor has the logs" convinces nobody.

The regulatory reason to build this early



The requirements vary, a humanoid sits in several regulated areas such as machinery, workplace safety, AI, and data.
The direction is clear: risk controls, human oversight, accountability.
An insurer, a client, a regulator or a worker on site can ask what the robot knew, what it did, and who could intervene.

STEP 3 – COMPOUND WHAT IT LEARNS

Goal: make every correction and every discovery permanent, and shared.

With robots, the cost of skipping this is unusually visible. You can watch four machines make the same mistake on the same afternoon. The hard part is not the fix. It's the capture.

THE FLOOR	**The flag happens in seconds**. A verbal tap, or the robot's own escalation. Nobody stops to write a considered account.
THE DESK	**The correction happens later** by someone qualified who diagnoses which of the three mistakes it was, and fixes the cause.

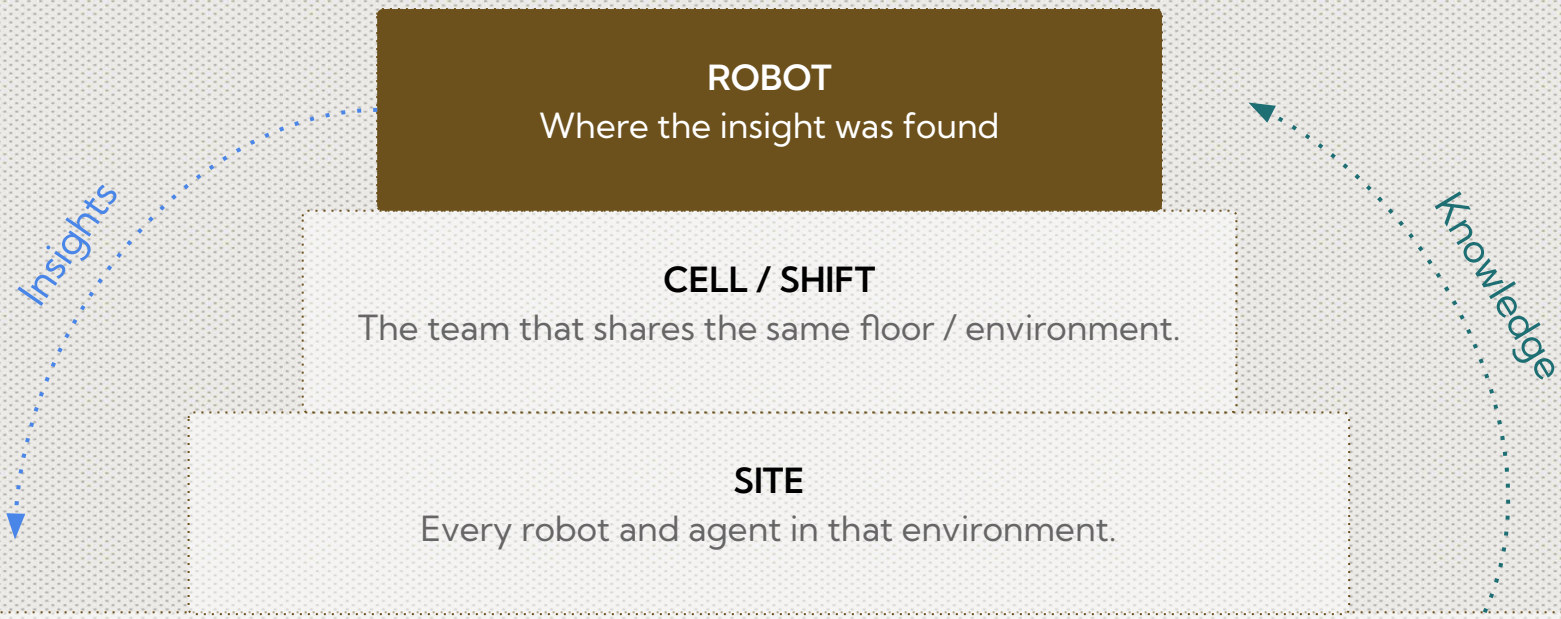
Near-duplicate submissions are caught automatically, so the knowledge base sharpens instead of bloating. And because every elevated insight becomes grounding knowledge, the next answer is better, the next correction is about something more subtle, and the next insight is more valuable.

The fix is the rule, not the incident.
In the returns example: the fix isn't "tell the robot about printer boxes." It's writing down that the label matters more than the box — a rule that was always true and never recorded.

WHY THIS MATTERS MORE WITH ROBOTS

Your environment is not single vendor. Your knowledge shouldn't be either.

An elevated insight moves one level up from where it was found, so the person who found it never has to work out who else needs to know.



Telemetry is not knowledge.
Fleet platforms as InOrbit, NVIDIA Fleet Command, MiR Fleet, tell you where every robot is and what errored. It won't tell you what it knows about your business, or give you a way to correct it.

WHY THIS COMPOUNDS, NOT JUST ACCUMULATES

The second robot should be dramatically cheaper to commission than the first.

Because Step 3 feeds back into Step 1. Every elevated rule becomes grounding knowledge. The next mistake is subtler, the next insight is worth more.

THE STANDARD WAY	THE BETTER WAY
Four robots make the same mistake all week	Flagged once, the rule gets written
A supervisor's workaround lives in their head	Elevated, reviewed, and permanent
Each new unit relearns the site	New units inherit what the site knows
Knowledge lives in a vendor cloud	One knowledge layer across robots and agents
The vendor has the logs	A record of what it knew, chose, and who approved it

If it isn't cheaper:
Ask what the first robot taught you that the second one inherited. If the answer is "nothing," you built a pilot. You did not build organizational memory.



ROLLOUT

Before the first robot arrives

Before commissioning. Write down the ten rules everyone already knows, the unwritten ones. Ask two experienced people to list them, the disagreements are the valuable part.

Week 1 — **Decide who flags, who diagnoses, who decides.** Three names. Confirm the audit trail captures what you'd need.

Week 2 — **Close the loop once, deliberately**. Take one real mistake from floor flag to diagnosis to correction to shared knowledge, with real content.

Week 3 — **Review the escalations, not just the incidents**. Every question the robot asked a human is a gap in what it was given. That list is your roadmap.

THE END STATE

Motion, safety and control stay with the robot's own certified stack from its vendor. You built **the knowledge layer** that sits above it, something that knows your business and humans can correct it.

Talk to us about your case →

- No credit card required
- Flat monthly price for your team
- Compliance and GDPR by design.