

KEOS
BY POCTEO

BIGTECH AFRICA · TUNISIA · 2026

ZERO HUMANS IN THE LOOP.

Self-healing robots at the edge with local AI inference.

LIVE DEMO

JETSON ORIN NANO SUPER

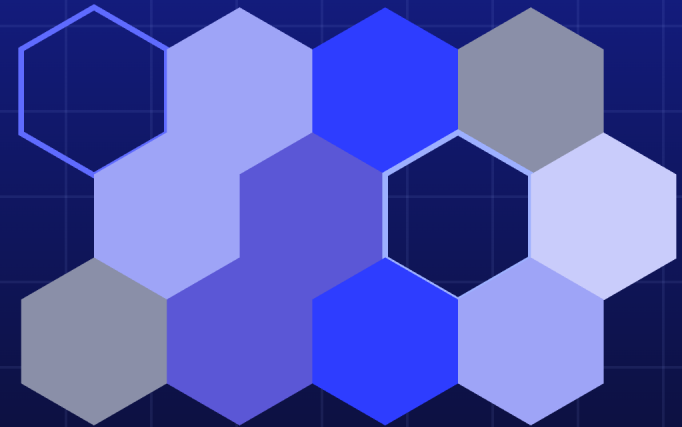
KUBERNETES

EBPF

LOCAL LLM

Walid Dridi

Technical Architect, Edge Infrastructure · Tunisia



pocteo.tn

Industrial activity must always run, whatever the circumstances.



Always-on activity

Lines, ports and plants never stop. Downtime is counted in minutes.



Efficient intervention

Time to recover is what counts. Today most of it is spent getting access.



Proactive, not reactive

Read the symptoms before the failure. Act ahead of the incident.



Huge fleets, few experts

Thousands of devices, a handful of engineers, none standing next to the machine.

KEY POINT

The question is no longer how fast a human can respond. It is how much can be handled before a human is needed.

At 3 am, **nobody** is coming.

A robot fails in a warehouse. No on-call engineer. Often no network. And a restart is not a diagnosis.



No human

The on-call engineer is asleep, 2 000 km away.



No network

No cloud, no VPN, no dashboard.
The site is on its own.



No diagnosis

A reboot hides the cause. It comes back at 3:40.

KEY POINT

Critical edge applications need a responder on the device, not on a pager.

Your best engineer still needs **45 minutes**.

Humans are not slow to think. They are slow to start. At the edge, the start is the whole problem.

A TYPICAL NIGHT • ONE ENGINEER, ONE ROBOT

03:07

Pager fires

If the alert reaches anyone at all.

03:15

Laptop, VPN

Wake up, find the laptop, connect.

03:20

No link

The site is offline. Call someone on site.

03:35

Logs, a guess

One log line, one theory, one restart.

03:50

"Looks fine"

Back to bed. Root cause unknown.

KEY POINT

Waking up, switching context and getting access cost more than the fix itself.

An LLM alone is a new engineer guessing.

THE NEW ENGINEER

Reads one log line and picks a theory.
Has never seen last month's incident.
Acts outside the runbook when stressed.
Cannot prove afterwards what was done.

THE RAW LLM

Fluent and confident on one log line.
No memory of your clusters or your past.
Will happily invent a kubectl command.
No audit, no limits, no undo.

Same failure mode. Fluent, fast, and wrong in ways you cannot audit.

KEY POINT

The difference is not the model. It is the structure around it: flow, context, guardrails, trust, tools, audit.

Human vs AI SRE at the edge

Step	Human on call	KEOS agent on the robot
Detect	Minutes, if the pager reaches anyone	Milliseconds, eBPF inside the kernel
Diagnose	20 to 60 min, from one log line	Seconds, from event + runbooks + live state
Act	Ad hoc, from memory, under stress	Bounded catalog, checked by policy
Verify	"Looks fine", back to bed	30-second watch: healed, or escalate
Learn	A post-mortem, maybe, next week	Every incident calibrates the next one
Available	One engineer for many robots, online	One agent per robot, offline, 24/7
Consistent	Depends on the hour and the person	The same nine steps, every time

KEY POINT

The agent is not smarter than your engineer. It is awake, local, consistent, and already has the context.

KEOS: an operating system **that heals itself.**

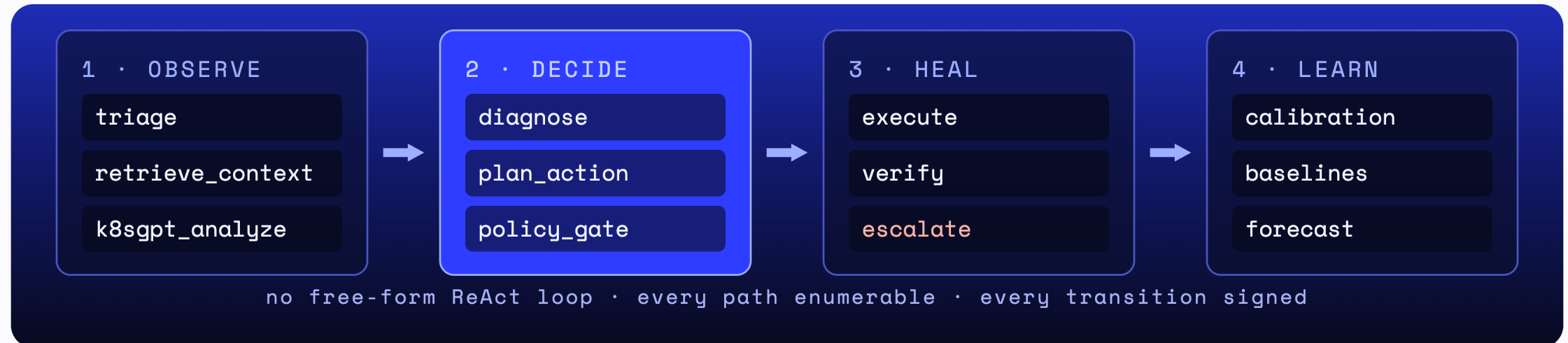
A Kubernetes-based edge OS with the whole AI SRE stack on one NVIDIA Jetson Orin Nano Super.

AGENT	AI SRE agent • a LangGraph state machine: observe, diagnose, act, verify
INTELLIGENCE	Local LLM on the GPU (Ollama • qwen2.5) • embeddings • Qdrant for runbook retrieval
CONTROL	OPA policy gate • MCP server with the action catalog • scoped RBAC • signed audit
OBSERVE	eBPF sensors (Falco • Tetragon) • kubelet events • NATS JetStream event bus
KUBERNETES	Immutable OS • k3s • GitOps • air-gap ready • the robot's apps run here as pods
HARDWARE	NVIDIA Jetson Orin Nano Super • GPU inference on the robot • ARM64

Everything runs in-cluster. No cloud. Nothing leaves the robot.

A closed graph, **not a free agent.**

Every incident walks the same nine steps, in LangGraph. Each step is a pure function and writes one audit line.

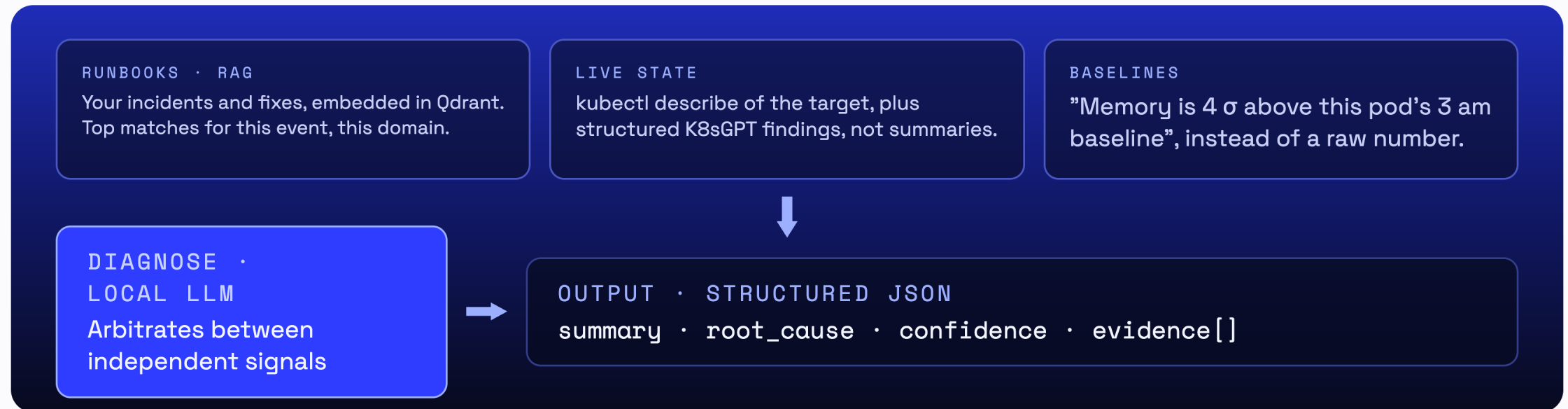


KEY POINT

The LLM decides inside a step. It never decides what the next step is.

Reasoning with **your past**.

A trust score is not enough. The model gets the enterprise context before it sees the event.

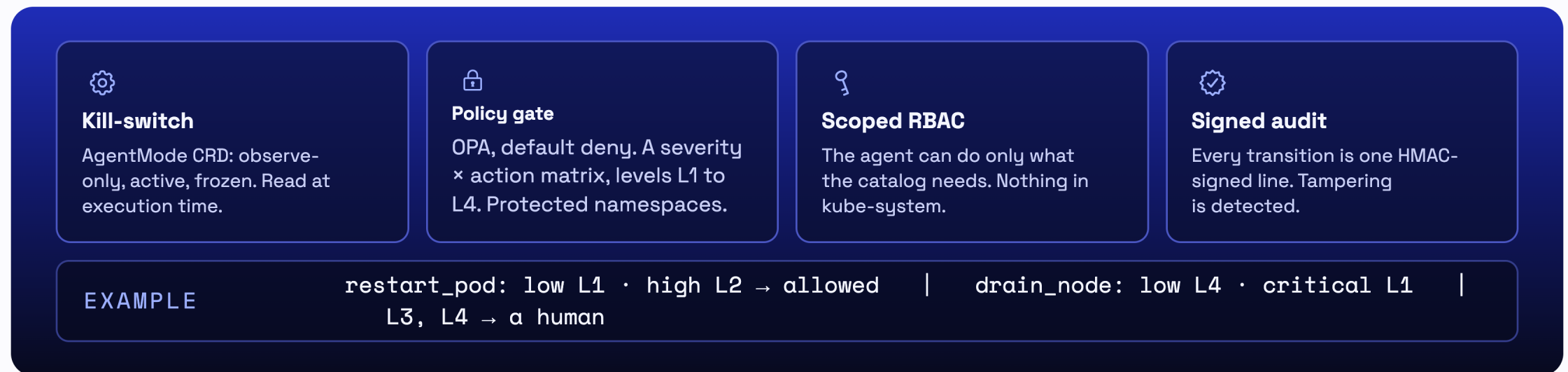


KEY POINT

A small local model with the right context beats a large remote model guessing from one log line.

Four locks on every action.

Autonomy is only safe because four independent controls gate each action. Policy is data, not a prompt.

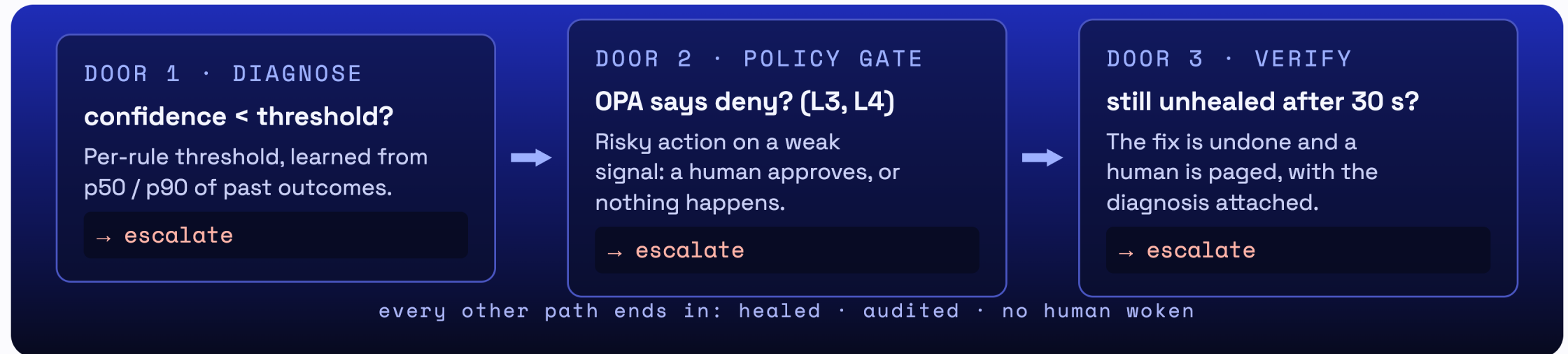


KEY POINT

The matrix is evaluated twice, in the agent and again in the MCP server. They must agree, or it escalates.

A trust score, and three doors to a human.

Confidence is calibrated per rule from graded outcomes. Below the threshold, the agent does not act. It escalates.

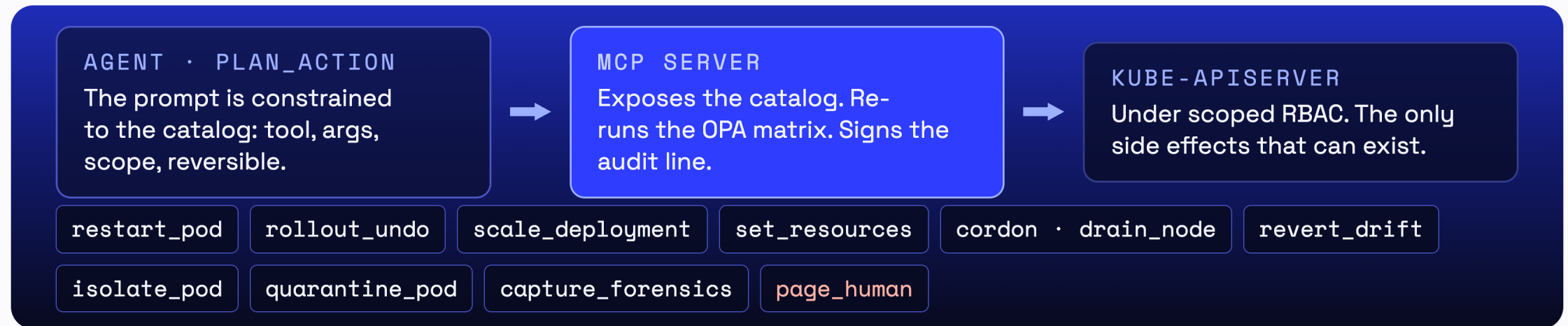


KEY POINT

Human approval is the exception path, by design. Not the default.

MCP: the **only door** to the cluster.

The agent never touches the API server. It calls tools from a catalog, through an MCP server that re-checks the policy.

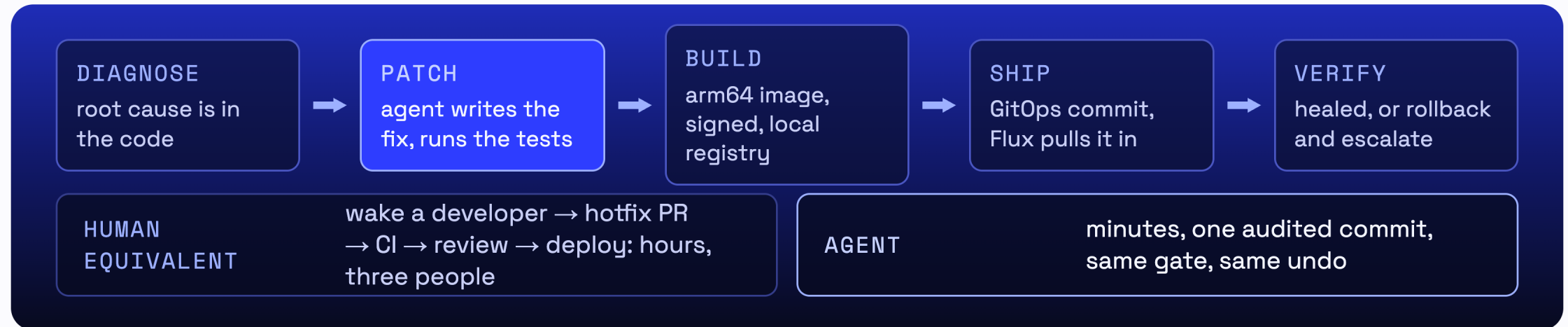


KEY POINT

The prompt can only ask for what the catalog offers. Everything else does not exist. Containment is reversible with one command.

From restart to **rebuild and ship**.

When the fix is code, the loop keeps going: patch, rebuild, sign, ship by GitOps. No ticket, no hands.



KEY POINT

The pipeline already exists: GitOps, signed images, a local registry. The agent is just a new committer, with the same guardrails.

Live: chaos on a robot.

Chaos Mesh injects a failure into the robot's navigation. Watch KEOS detect, diagnose and recover it. No cloud, no reboot, no human.

THE ROBOT ON STAGE

Mobile robot · Jetson Orin Nano Super · KEOS (k3s)

ROS 2 app in four pods:

sensor-pipeline

ai-navigation

motor-driver

keos-agent

Network cable unplugged, on purpose.

WHAT YOU WILL SEE

t + 0 s **Inject** · Chaos Mesh kills ai-navigation, the robot stops

t + 1 s **Detect** · kubelet + eBPF event lands on the local bus

t + 5 s **Diagnose** · runbook match, root cause, confidence above threshold

t + 6 s **Act** · restart_pod, severity high, OPA level L2: allowed, via MCP

t + 40 s **Verify** · healed, robot moves, audit line signed

WATCH FOR

The pipeline view on screen, the confidence number, the policy decision, and the signed audit line at the end.

Humans move up the stack.

Zero humans in the loop does not mean zero humans. It means humans stop being the loop.



Engineering

Write the runbooks the agent retrieves.

Build and extend the action catalog.

Grade outcomes, so trust is earned.



Orchestration

Set the intent for the whole fleet.

Versions, waves, policies by tag, GitOps.

One cockpit for a thousand robots.



Control

Tune the policy matrix, hold the kill-switch.

Approve the rare L3 and L4 actions.

Read the signed audit, not the logs.

The 3 am job is gone. The 9 am job got better.

KEOS
BY POCTEO

BIGTECH AFRICA · TUNISIA · 2026

ZERO HUMANS IN THE LOOP. NOT ZERO HUMANS.

Kubernetes edge OS · AI SRE agent · local inference on NVIDIA Jetson

Walid Dridi

linkedin.com/in/walid-dridi-strivly · strivly.co



POCTEO.TN

Thank you · Questions?