



NPCW · PLENARY · 2026

Agent-Oriented System Control

Adaptive agents, digital twins, and physics-informed inference
for process control, monitoring, and predictive maintenance

Soheil Mansouri, PhD · Co-Founder, Board Director & Chief Scientist

SiC Systems — Sense, Infer & Control · Technical University of Denmark · ssm@si-c.io

Nordic Process Control Workshop · August 2026

LLMs alone can't regulate physical systems

Not decision models

Token prediction $\neq$ constrained decision-making. No native notion of feasibility, operating envelopes, or specs.

Not reproducible

Variable outputs across **identical queries** — engineers operate from established baselines and validated procedures. Control demands both.

No physical grounding

No VLE, kinetics, or transport behind the words. The plant pushes back; text doesn't.

No workflow discipline

Real-time regulation needs fixed process logic and deterministic, physics-based description — LLM reasoning must be **embedded in structured agent architectures**, not left free-running.

...and yet reasoning, decomposition, and tool use are exactly what our control rooms lack. **The question is how to harness them safely.**

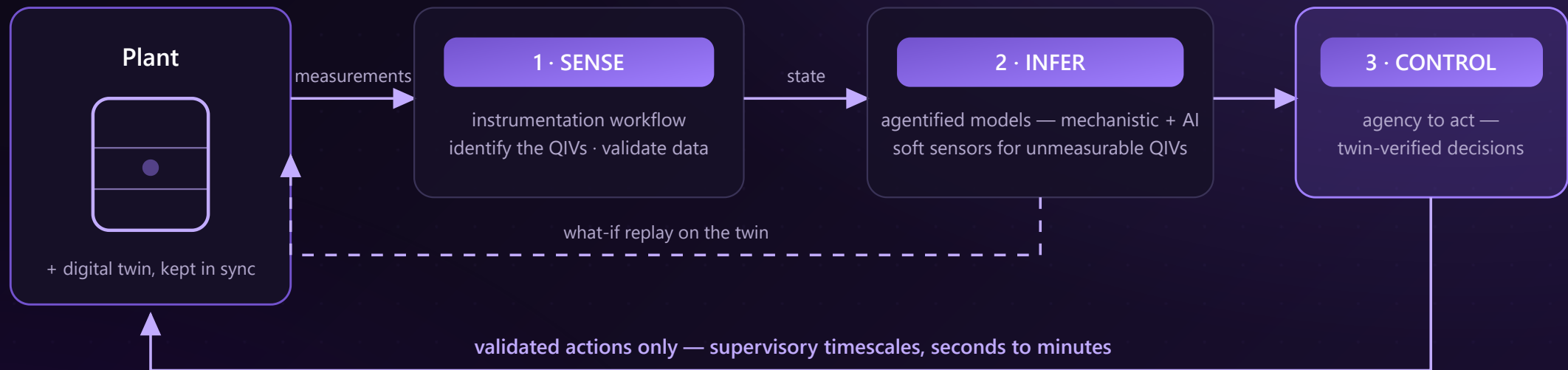
From **AAOSA** to agent-oriented system control

The agent network that once routed language now routes control decisions.



Not new control algorithms — **an orchestration layer** that unifies mechanistic twins, hybrid learning, and task-specialized agents **without replacing MPC or established control.**

Sense · Infer · Control — the three-stage loop



If a QIV can't be measured directly, **stage 2 builds the sensor** — hybrid mechanistic + data-driven models that generate it from what's easy to measure.

Where agents sit in the **control hierarchy**

Planning & scheduling

days – weeks

Agents translate business targets into feasible operating plans — e.g. feed selection and blending informed by the margin-vs-energy record the loop accumulates.

Real-time optimization

hours

Agents watch for drift and re-pose the optimization — detect deviation from reference, decompose its drivers, decide when re-optimizing is worth triggering.

Supervisory control · MPC

minutes

Agents propose set-point and constraint updates — every candidate is verified on the digital twin before hand-off; on communication loss, **the system falls back to safe baseline policies.**

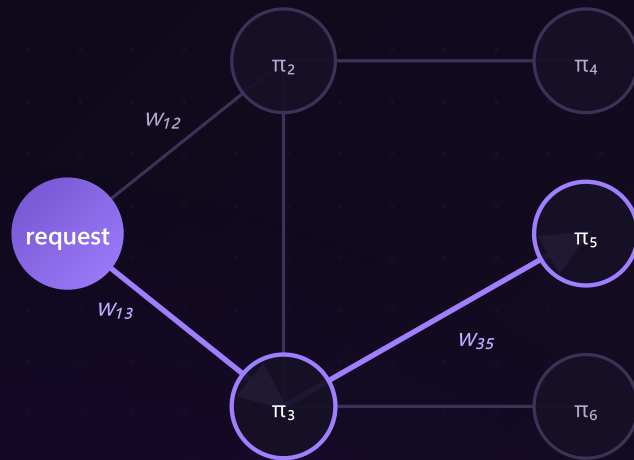
Regulatory control · SIS

seconds — untouched

Deterministic, certified, and off-limits. Agents never write to valves; hard real-time stays with PID and the safety system, exactly as certified.

The mesh is a **coordination layer across time-scales** — it extends the pyramid, it does not replace it.

Coordination — who handles what



agents = nodes · links carry cost w_{ij} · policies π_j estimate responsibility

$$G = (V, E) \quad (1)$$

The system is a directed graph — sensor, simulator, and executive agents as nodes; communication links with cost w_{ij} (latency, reliability).

$$P^* = \arg \min \Sigma w_{ij} + \lambda \cdot \text{uncertainty} \quad (2)$$

Responsibility discovery: a request propagates recursively; each agent scores it with policy π_j , and the handling path minimizes cost plus uncertainty. Routing effort scales with the agents involved — **no global enumeration.**

Descended from AAOSA's "is-this-yours?" recursion — reinterpreted for process monitoring, twin simulation, and distributed control.

Mansouri et al., *Curr Opin Chem Eng* 2026, 52:101253 — formulation illustrates the decomposition, not formal complexity guarantees

Decisions — local solves, global coordination

$$\min C = \sum_j c_j x_j + \sum_m d_m y_m \quad \text{s.t.} \quad \sum_j a_{jm} x_j \leq b_m \quad (3)$$

The plant-wide problem: global cost under capacity and timing constraints — scheduling, allocation, set-point selection.

$$\mu^{t+1} = \max(0, \mu^t + \alpha \cdot \text{constraint violation}) \quad (4)$$

Decentralized: each agent solves its local min C_i ; shared constraints are priced by multipliers updated at supervisory timescales — seconds to minutes.

$$V^*(s) = \max_a [R(s, a) + \gamma \mathbb{E} V^*(s')] \quad (5)$$

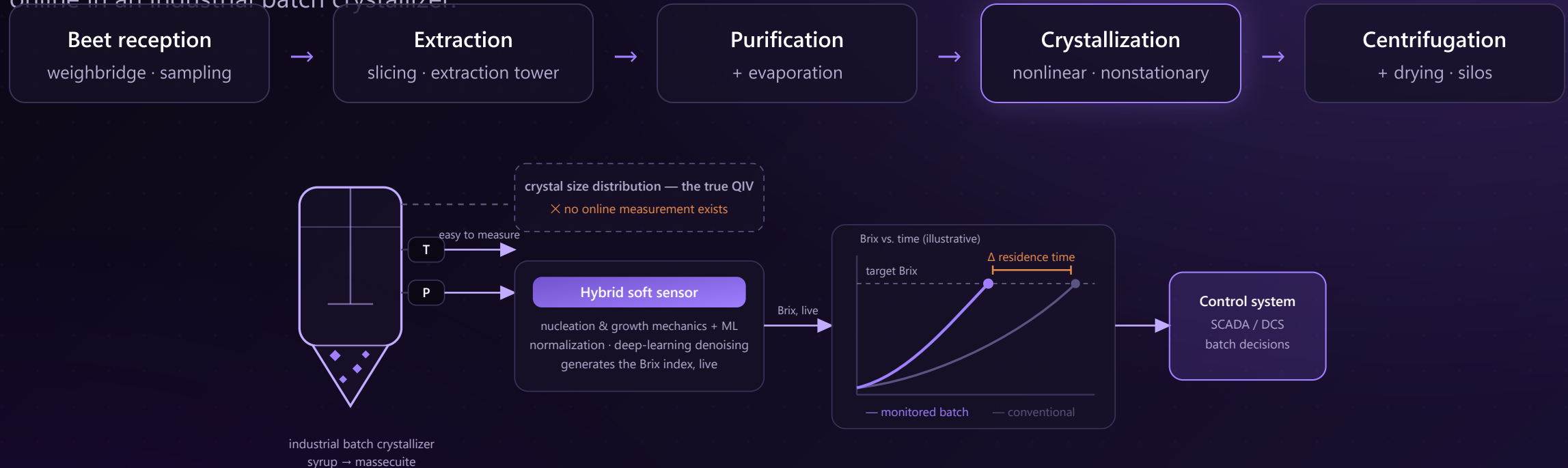
Uncertainty as an MDP: states include twin simulations; agents approximate V^* with learned policies, and **the digital twin supplies fast rollouts** for what-if evaluation before actuation.

Centralized methods scale exponentially and falter under change — **decomposition cuts per-agent complexity** while approximating the coordinated solution.

Mansouri et al., *Curr Opin Chem Eng* 2026, 52:101253 — illustrative formulation, not a convergence proof

Virtual sensing — the crystallizer that reports its own Brix

Industrial sugar production (with Nordzucker & DTU): crystal size distribution is the quality-critical variable — and it can't be measured online in an industrial batch crystallizer.



batch residence time ↓

energy ↓

CO₂ ↓

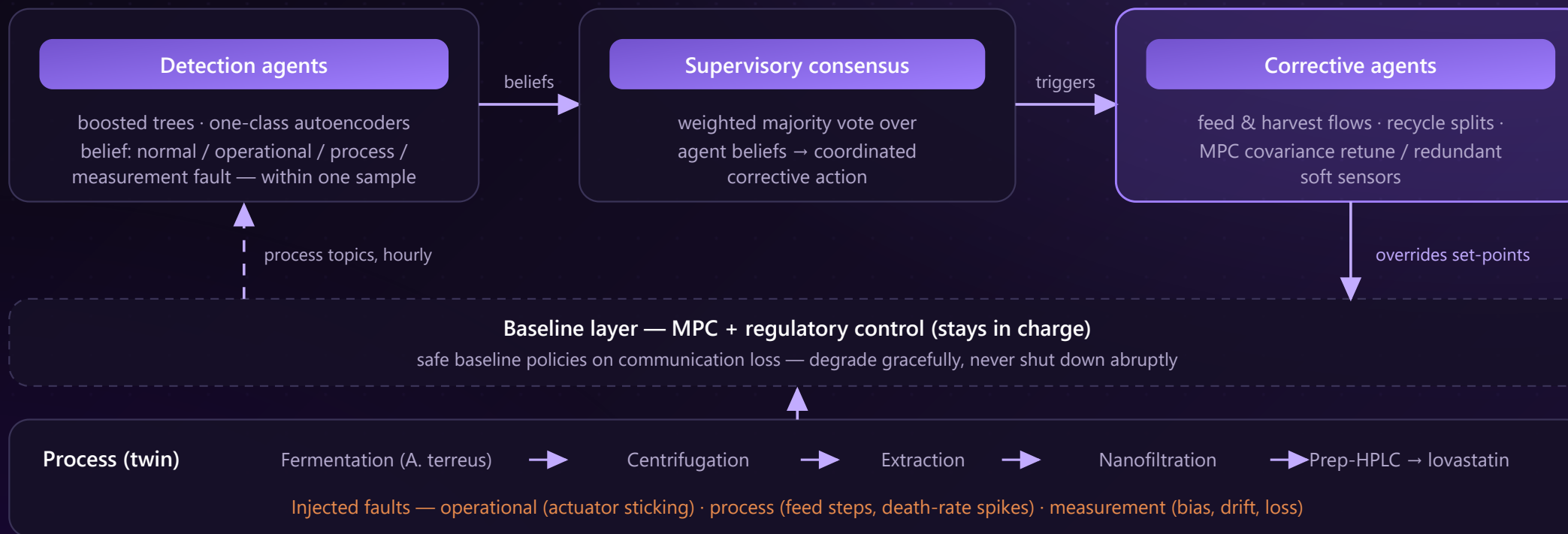
capacity ↑

off-spec remelt loops avoided

Same pattern trackside: generative “virtual sensors” infer tire slip angle from telemetry at the edge — **~100 models inferring live during a race.**

Closed-loop agentic control — self-healing bioprocess

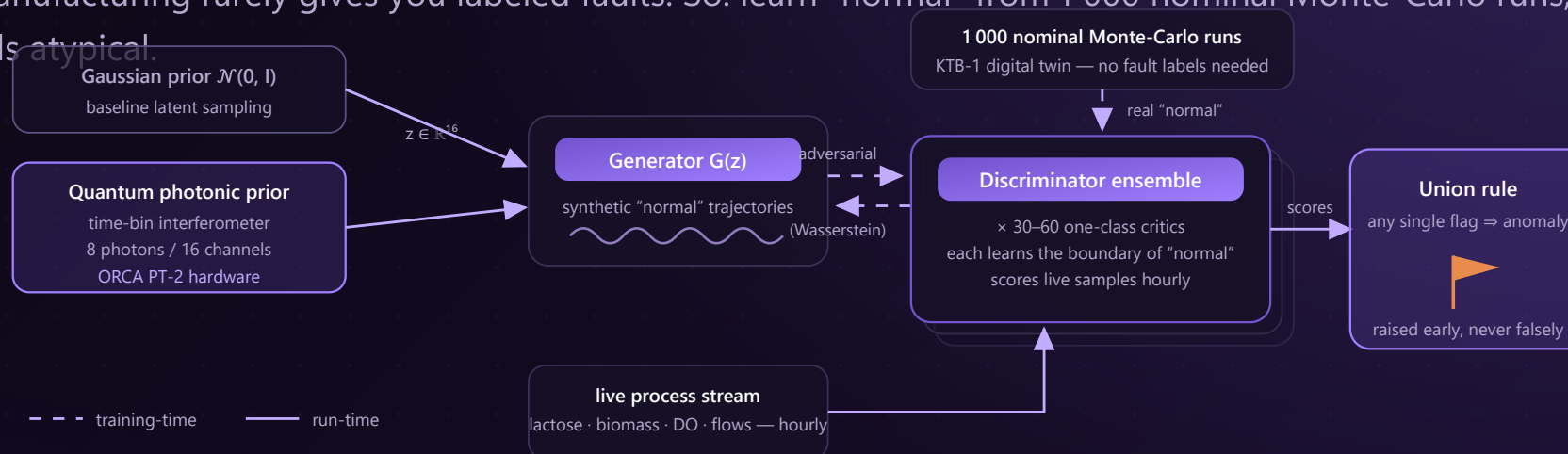
KTB-1: an end-to-end digital twin of continuous lovastatin manufacturing — 38 states (20 measured or soft-sensed), sampled hourly, with realistic sensor and actuator noise.



Result: across 500 Monte-Carlo runs per fault scenario, the overlay **significantly cut fault-induced deviation duration vs. MPC-only** — faster return to nominal. Simulation-based proof-of-concept; experimental validation is the stated next step.

Early anomaly detection — trained on normal only

Continuous biomanufacturing rarely gives you labeled faults. So: learn “normal” from 1 000 nominal Monte-Carlo runs, and flag anything the ensemble finds atypical.



> 50 % true positives at zero false alarms

detection rate at FPR = 0 — deviations flagged early without ever crying wolf in the twin environment

Gaussian prior	0.350 ± .019
Quantum — hardware	0.412 ± .019
Quantum — simulated	0.429 ± .009

AUC, inverted one-class scoring — ranking quality in the low-false-positive regime, where early detection lives

Benchmark: a **1.5 % lactose step at hour 10**, flagged early. Discriminator agents broadcast anomaly likelihoods into the AAOSC loop — **supervisory agents act before the deviation matures into a shutdown.**

What's **measured** so far

Published — simulation studies, stated as such

Closed-loop control: agentic overlay significantly cut fault-induced deviation duration vs. MPC-only — 500 Monte-Carlo runs per fault scenario on the KTB-1 twin

Predictive maintenance: >50 % true-positive rate at **zero false alarms**; quantum priors beat Gaussian baselines consistently

Monitoring: virtual Brix sensor in industrial sugar crystallization — **residence time ↓, energy ↓, CO₂ ↓, capacity ↑**

Every number above is **traceable to the published record** — and the honest caveat travels with it: plant-model mismatch, sensor degradation, and long-term drift are the stated next steps.

Three things to take home

- 1 Agents orchestrate — they don't replace control.** AAOSC is a supervisory coordination layer over MPC and regulatory control, not a new control algorithm.
- 2 Safety comes from structure, not scale.** Constraints in the loop, twin-verified actions, safe baseline fallbacks — that is what makes autonomy governable.
- 3 Monitoring, control, and maintenance share one substrate:** digital twins + distributed physics-informed inference. The open problems are control problems — this community's problems.



Thank you.

Soheil Mansouri, PhD · SiC Systems — Sense, Infer & Control

ssm@si-c.io · si-c.io

Boskabadi et al., *Curr Opin Chem Eng* 2025, 48:101150 — doi:10.1016/j.coche.2025.101150

Mansouri et al., *Curr Opin Chem Eng* 2026, 52:101253 — doi:10.1016/j.coche.2026.101253

BACKED BY



PLUGANDPLAY

BACKUP · FOR Q&A

Backup slides

The three pillars · guarantees · open problems · outlook

One substrate, three run-time jobs

Monitoring

Virtual sensing

QIV inference

Data validation

Drift detection

See what the instruments can't — soft sensors generate the unmeasurable.

Control

Supervisory set-points

Twin-verified overrides

MPC retuning

Act on it — corrective agents override set-points, never the safety layer.

Predictive maintenance

Early anomaly detection

Fault diagnosis

Intervention scheduling

Catch it early — flag deviations before they become shutdowns.

All three run on the **same digital twins and the same distributed inference** — one case study for each in the main deck.

Physics in the loop = governable autonomy

Safe fallback by design

On communication disruption or abnormal conditions, the system **defaults to safe baseline control policies** — stable operation, never abrupt shutdown. The agent layer degrades; the plant doesn't.

Constraint verification

Every candidate action is checked against balances, envelopes, and safety limits before it leaves the sandbox — data validation rejects physically unreasonable values first.

Simulation before actuation

Candidate actions run on the digital twin; only validated actions reach operations. Agents test hypotheses in simulation, not on the plant.

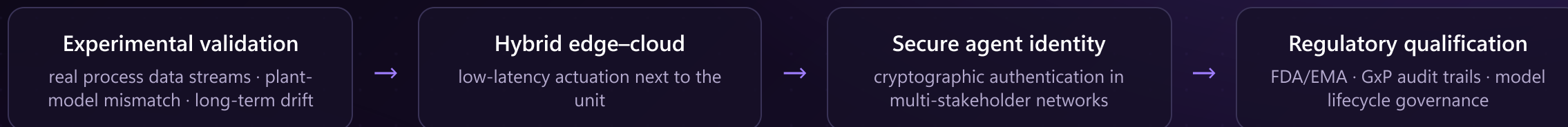
Policies as safety cases

Agent capabilities are explicit and auditable — what data, which tools, which actuators, under whose approval. Human-in-the-loop oversight is preserved.

What we need from **this community**

- **Guarantees for agentic loops** — stability and constraint satisfaction when a planner sits inside the control hierarchy.
- **Hybrid model fidelity management** — when to trust the surrogate, when to escalate to rigorous models.
- **Benchmarks** — open, physics-grounded tasks for monitoring and control agents; today everyone grades their own homework.
- **Human-in-the-loop operations** — approval interfaces and explanations operators can actually audit.
- **Qualification & safety cases** — certifying an agent mesh the way we certify an SIS; FDA/EMA-grade audit trails and lifecycle governance for regulated production.

From proof-of-concept to plant floor



Also ahead: multi-logic decision ensembles within agents, and quantum-accelerated inference where the search space demands it.

The interesting problems ahead are **control problems** — which is why I'm here.