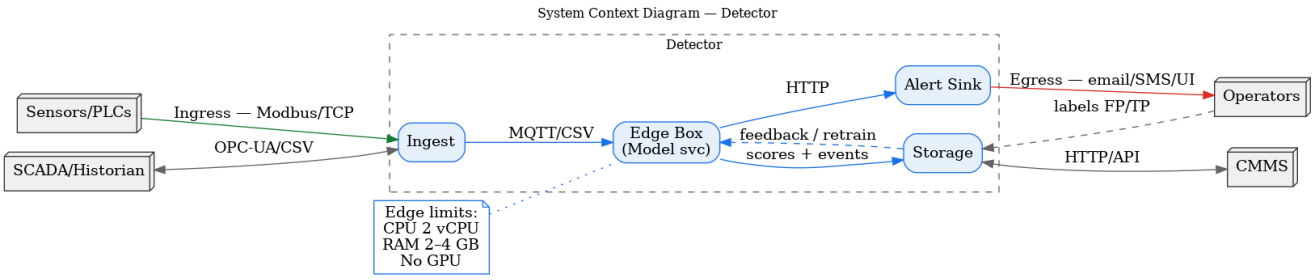
Systems Context Diagram Breakdown



What this picture shows

A small “detector” box sitting on the plant floor. Data goes in once. Alerts come out once. People/tools send feedback back in.

Parts

Outside the detector

- **Sensors/PLCs**: where raw numbers come from.
- **SCADA/Historian**: plant database/visuals that can also feed data.
- **CMMS**: work-order tool.
- **Operators**: humans who see alerts and label them.

Inside the detector

- **Ingest**: first stop for data.
- **Edge Box (Model service)**: runs the lightweight model.
- **Storage**: saves raw slices, scores, alerts, and labels.
- **Alert Sink**: sends notifications.

Flows

- **Ingress**: Sensors/PLCs → Ingest *(Modbus/TCP)*.
- SCADA/Historian ↔ Ingest *(OPC-UA/CSV)*.
- Ingest → Edge Box *(MQTT/CSV)*.
- Edge Box → Alert Sink *(HTTP)*.
- Edge Box ↔ Storage *(scores/events ↔ feedback/retrain)*.

- **Egress**: Alert Sink → Operators *(email/SMS/UI)*.
- Operators → Storage *(labels: real/false)*.
- Storage ↔ CMMS *(HTTP/API)*.

Why it's shaped this way

- Exactly **one door in** for plant data. Easier to secure and test.
- Exactly **one door out** for alerts. Easier to wire to people/tools.
- A **feedback loop** exists to learn and reduce noise over time.

What the model does (simple)

- Reads recent sensor history.
- Computes an **anomaly score** per tick.
- Compares score to a **cutoff**. If over, it alerts.
- Examples: moving average + z-score, IsolationForest, KNN, One-Class SVM, tiny autoencoder, small trees, or a fast linear model.

What we store

- Small raw windows.
- Scores and alerts.
- Operator labels: "real" or "false positive."

What to measure

- **PR-AUC**: higher is better at catching rare faults without spam.
- **Alert latency**: time from fault start to alert.
- **Noise rate**: how many false positives.
- **Drift**: normal behavior shifts over weeks/months.

How it improves

- Watch drift. If behavior shifts, **retrain** and **reset cutoff**.

- Do this on a schedule (e.g., quarterly) or when drift is detected.

Edge limits (design guardrails)

- CPU about **2 vCPU**.
- RAM about **2–4 GB**.
- **No GPU**. Keep models light.