TWENTY-SEVENTH ANNUAL

TestConX™

Archive

DoubleTree by Hilton
Mesa, Arizona
March 1-4, 2026

AutoML-Driven Pipeline for Real-time Semiconductor Test Optimization via Cloud-IoT Integration

Vincent Chu
Advantest

Mesa, Arizona • March 1–4, 2026

ADVANTEST®

Agenda

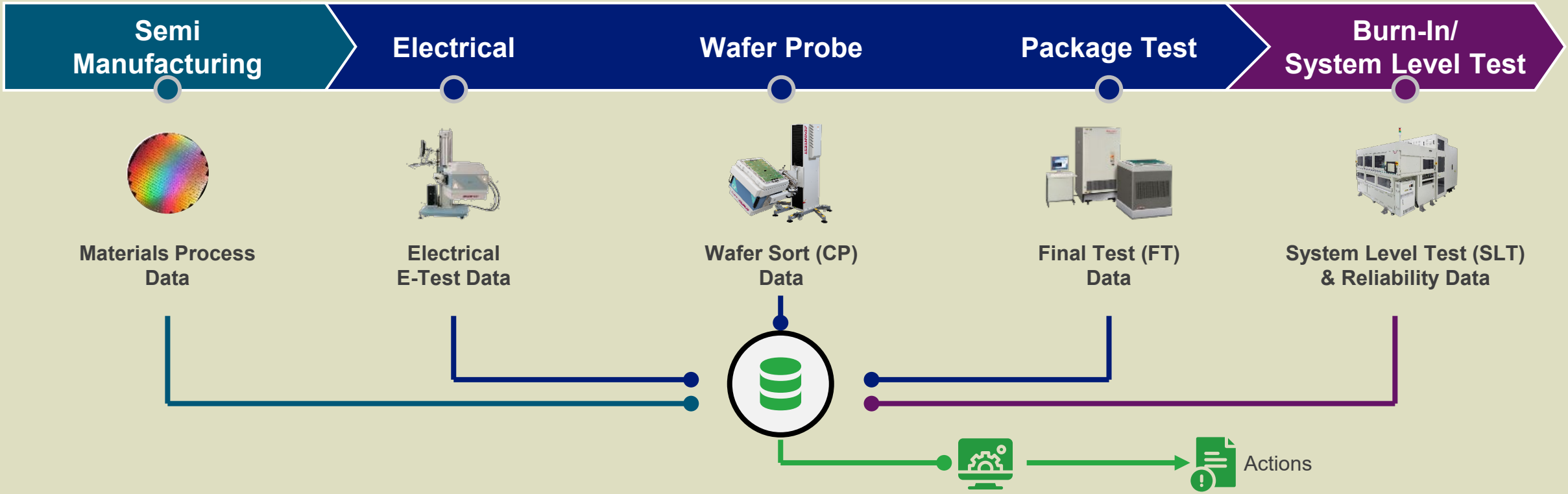
1 Real-time Optimization & Cross-Insertion ML

2 The Production Gap

3 Our Solution: AutoML + Cloud-IoT Pipeline

4 Impact & Key Benefits

Why Real-time Optimization Matters



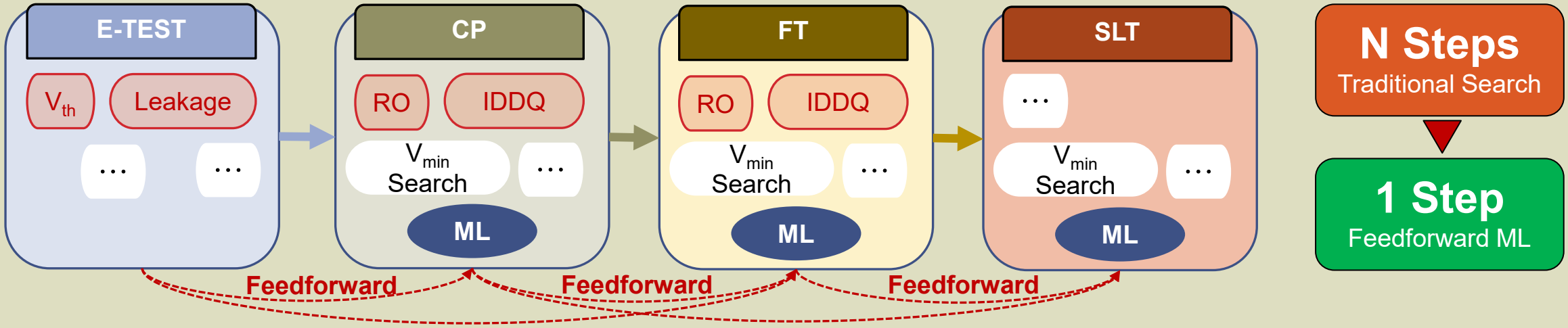
ML models use data from every test stage to reduce test time, improve yield, and catch quality escapes — in real time.

Reduce Test Time: Skip-test, predictive search
Improve Yield: Shift-left, predictive binning, die matching
Improve Quality: Multivariate outlier screening

Cross-Insertion ML: Predicting V_{min} from Prior Test Data

Key insight: Feedforward data from prior insertions predicts V_{min} at CP or FT — replacing N-step search with a single step.

- **Cross-insertion feedforward**: E-Test (V_{th}, Leakage), or CP features (RO, IDDQ) stored per-die, retrieved in real time at CP or FT for inference.
- **Challenge**: Cross-site aggregation + real-time retrieval under tight latency constraints.



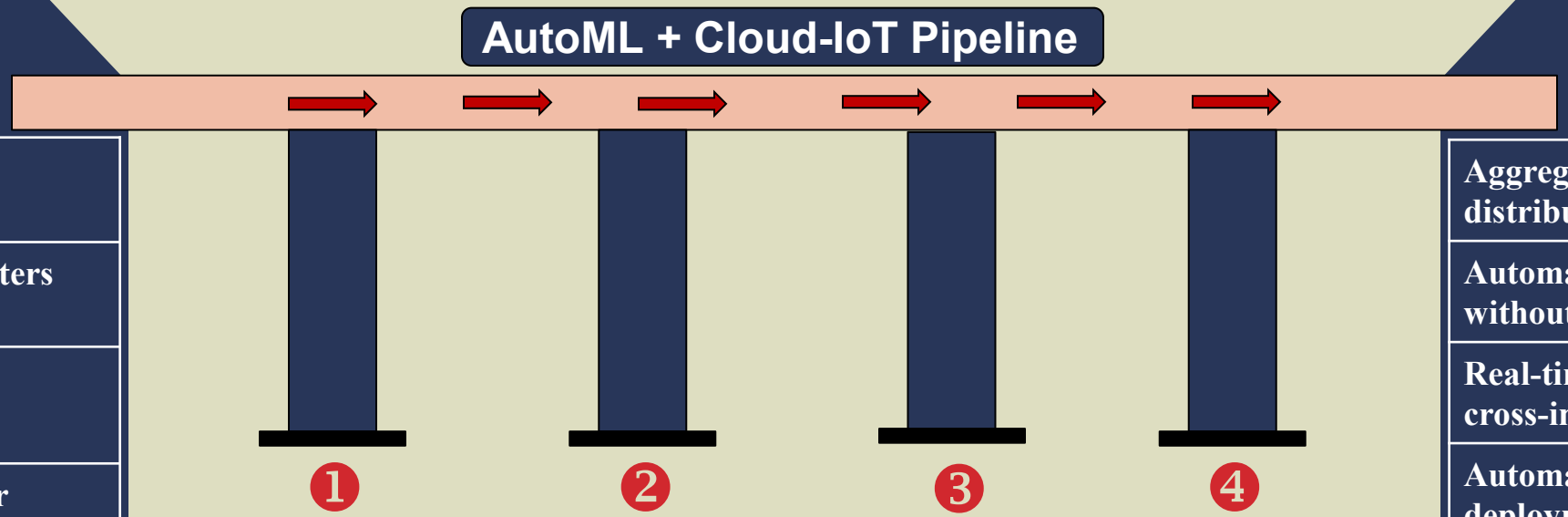
The Production Gap: ML in the Lab vs. ML on the Test Floor

ML in the Lab

Proof-of-Concept

- Train a model on historical data
- Tune hyperparameters manually
- Run inference on a workstation
- Deploy to one tester manually
- Validate once and ship

ML can optimize semiconductor test. The unsolved problem: how do you develop, deploy, and orchestrate ML models across distributed testers and OSATs — without a data science team?



4 innovations: lab proof-of-concept to production floor reality

ML in Production

At Scale. Across OSATs

- Aggregate data across distributed OSAT sites
- Automate model selection without ML expertise
- Real-time inference with cross-insertion features
- Automatic Fleet-wide deployment in OSATs
- Detect drift, retrain, and redeploy without disrupting production

What We Built: End-to-End Results

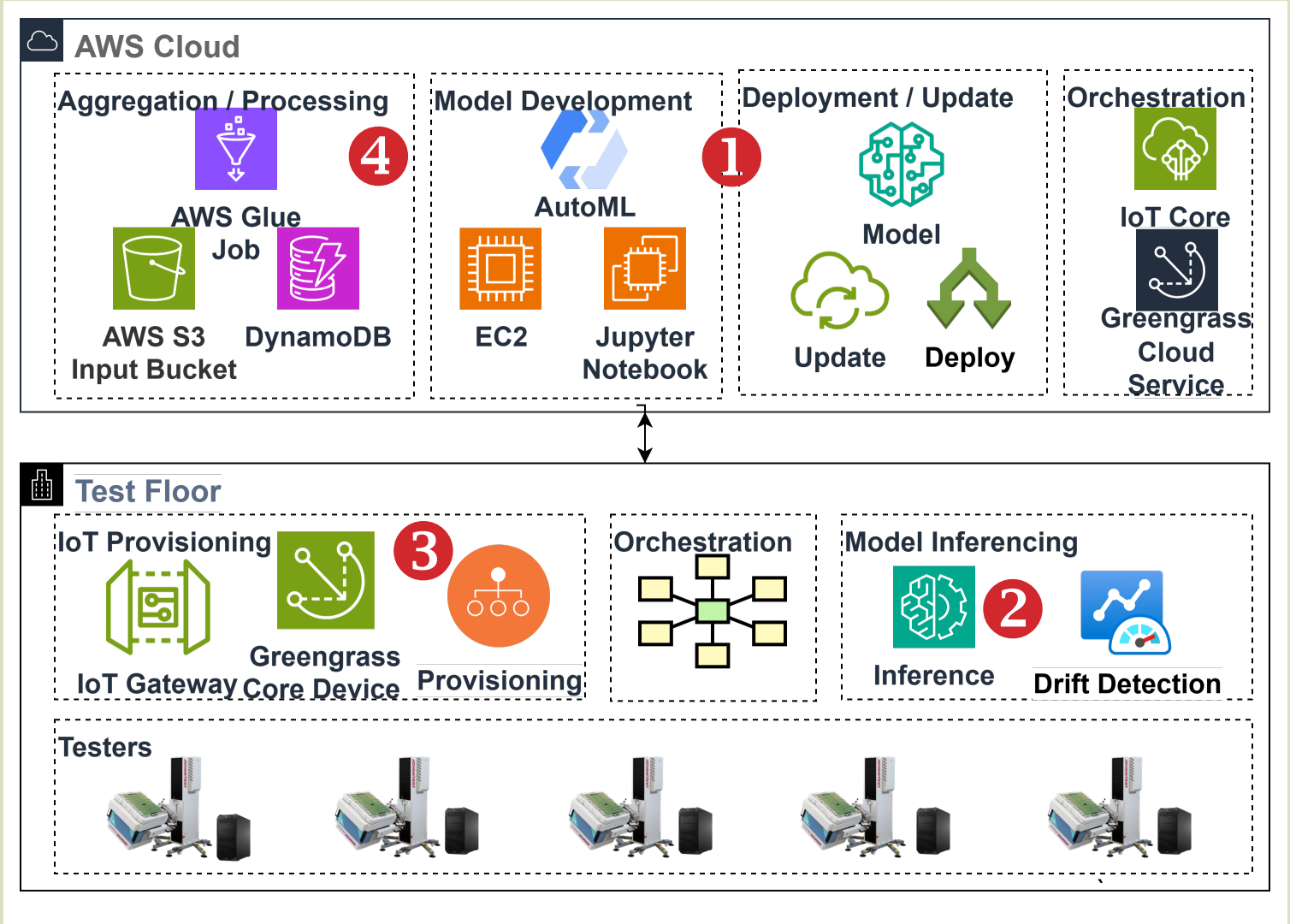
36x	50x	<130ms	0-touch
Faster	Less	Inference	Scaling
• >3 days → <2 hours model development	• ~500 lines → <10 lines of code	• Incl. cloud feature retrieval	• Fleets of testers across OSATs

AutoML removes the ML expertise barrier. **Cloud** removes the infrastructure barrier. **IoT** removes the deployment barrier.
Together: a closed-loop system a test engineering team can operate without data scientists or dedicated IT.

Framework Overview

This pipeline is purpose-built for semiconductor test: STDF ingestion, test-specific feature extraction, cross-insertion data fusion, and edge inference within test-program timing budgets

1. Test-specific AutoML pipeline – develop, deploy and update
2. Real-time edge inference with cross-insertion features
3. Automated OSAT provisioning for fleet-wide deployment
4. Cross-Insertion ML feedforward



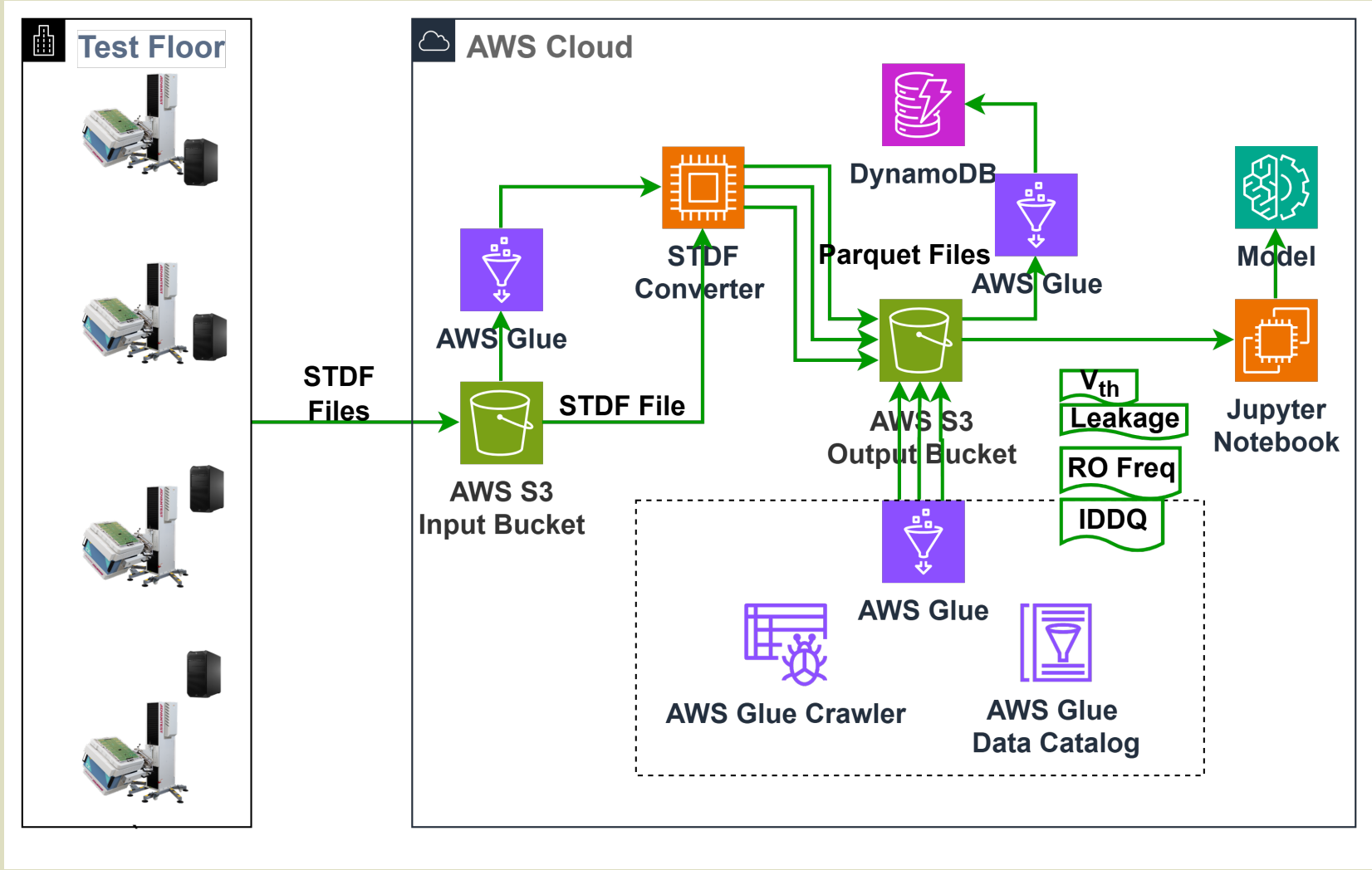
Data Aggregation & ETL

Data Aggregation

- S3 stores STDF and feature data in buckets

Extract, Transform, Load (ETL) Processing

- Glue: STDF → tabular with feature extraction
- Event-driven pipeline, trigger by new data in S3
- DynamoDB stores prior-insertion features (V_{th}, Leakage, or RO, IDDQ) for fast retrieval in CP or FT



AutoML for Semiconductor Test: Results

AutoGluon provides the ML engine — but the pipeline around it is purpose-built for semiconductor test:

- STDF ingestion and parsing (domain-specific data format)
- Test-specific feature extraction
- Cross-insertion feature aggregation: die-level matching across insertions
- Validation against test-program timing constraints

```
from autogluon.tabular import TabularPredictor
predictor = TabularPredictor(label='Vmin').fit(train_data)
predictions = predictor.predict(test_data)
```

Prediction Performance Comparison

Metric	AutoGluon Default	AutoGluon Best_quality	AutoGluon Extreme_quality	LightGBM Default	LightGBM Best parameters
R^2	0.9080853	0.9055413	0.9080536	0.907	0.9083

Training Time (s) Comparison (8GB RAM)

Rows	2000	5000	10000	20000	30000	60000
t3a.large (2 vCPU)	128.76	265.67	606.6	809.69	1341.26	2444.6
c5.xlarge (4 vCPU)	51.73	151.73	316.92	421.88	663.13	765.19

Development Efficiency Comparison

	Manual	AutoML
Development time	>3 days	< 2 hours
Code required	~500 lines	< 10 lines

AutoML matches expert-tuned models with 36× less effort — enabling test engineers to build production ML without data science expertise.

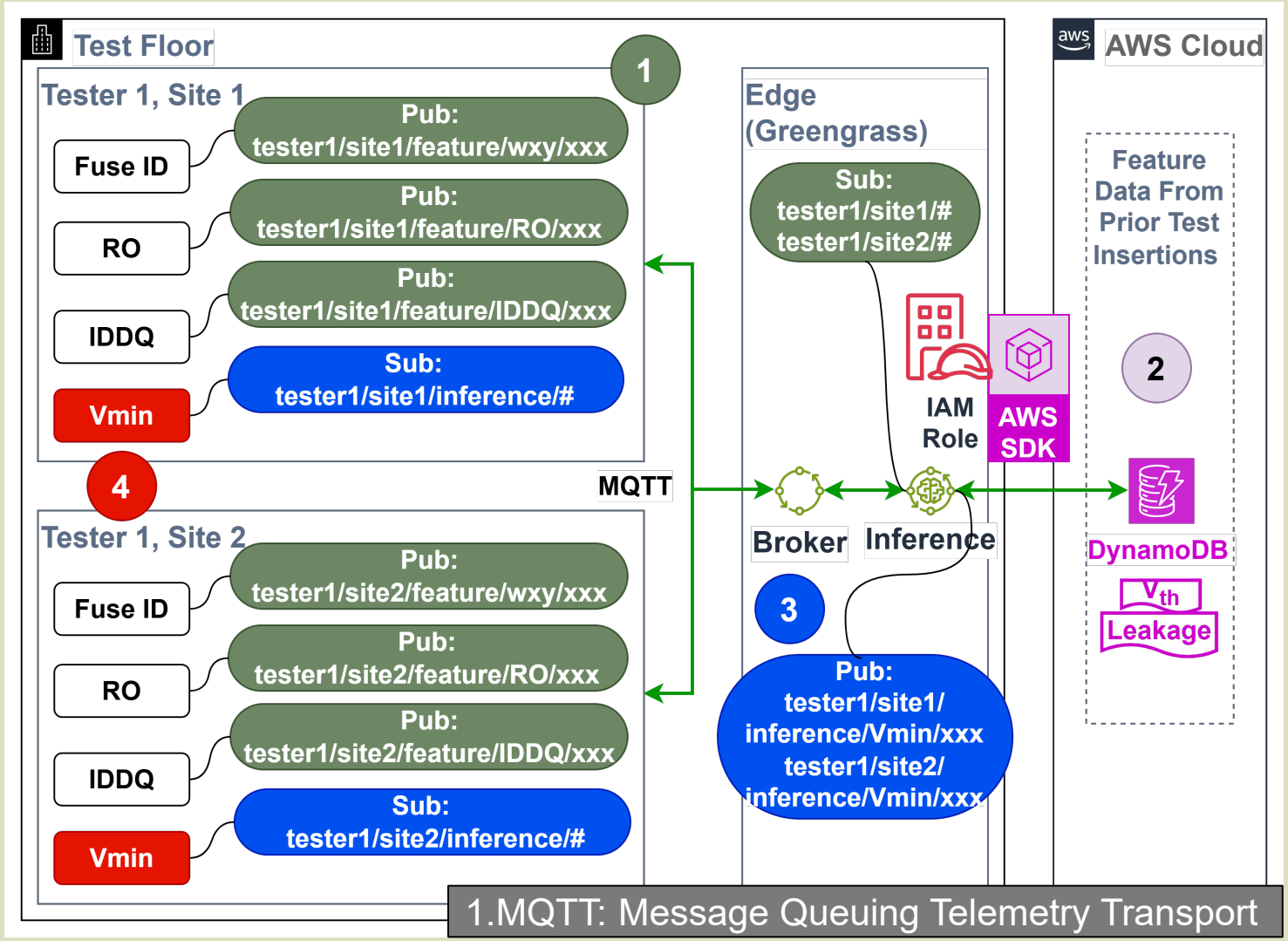
Real-Time Edge Inference in <130ms

During live test execution:

1. Tester publishes RO Frequency and IDDQ to edge via MQTT(1)
2. Edge fetches E-Test V_{th} and Leakage from DynamoDB (~30ms round-trip) via SDK
3. Model fuses current-insertion + prior-insertion features, predicts V_{min} (<100ms)
4. Tester receives prediction, skips N-step search

Total: <130ms end-to-end, within test-program timing budget

Scale: Greengrass Core serves multiple testers and sites simultaneously



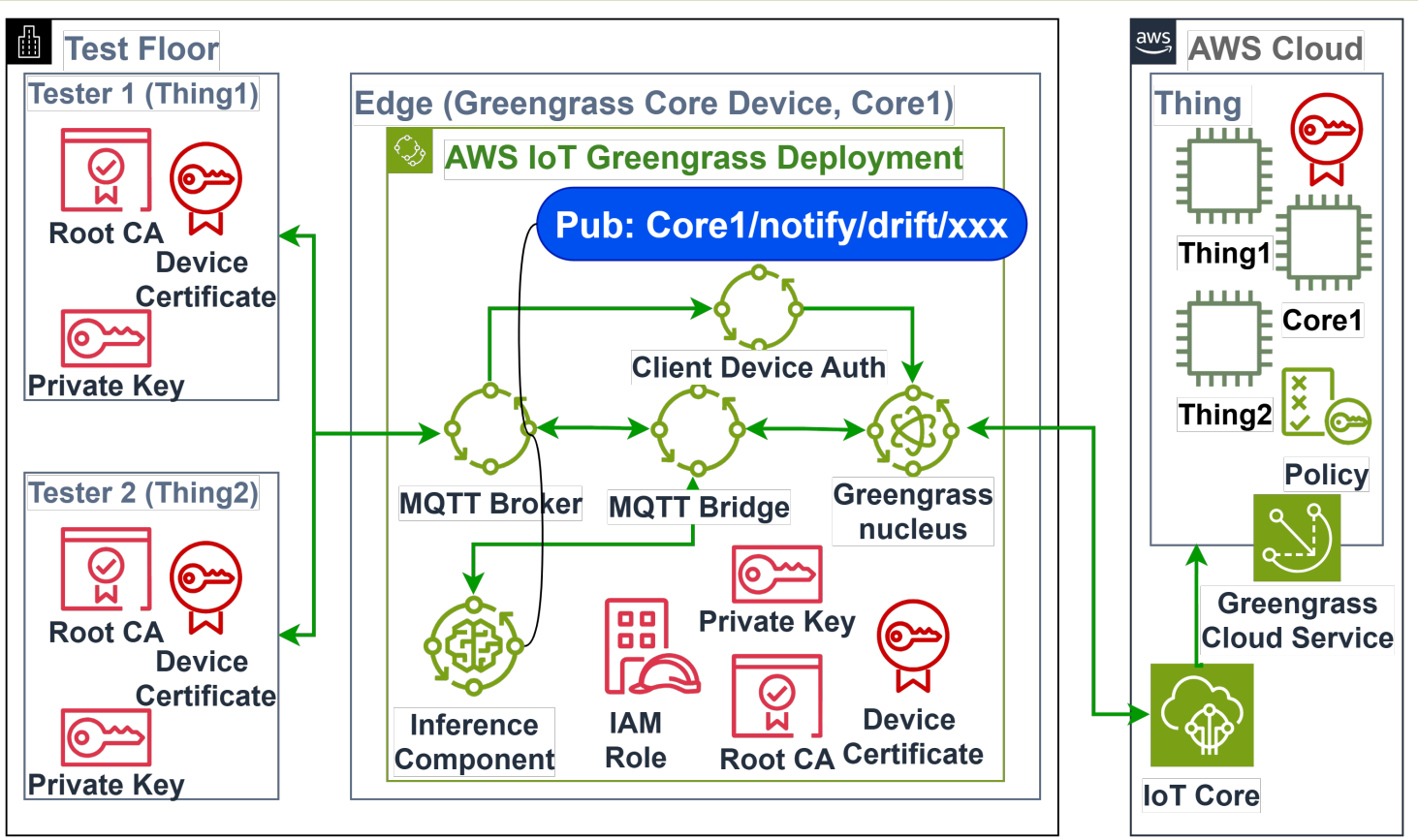
Deployment, Drift Detection & IoT Security

Security:

X.509 certificate-based device identity. IoT policies control per-device topic access. IAM roles for cloud resource access (DynamoDB). mTLS (mutual TLS) for all communication.

Deployment:

Container application, OTA via Greengrass component. Model updates pushed to the fleet without tester-by-tester manual installation.



Drift Detection: Edge monitors prediction accuracy. When degradation exceeds threshold, publishes drift notification to cloud → triggers retraining via AutoML → OTA redeployment.

Bridging the Fabless-OSAT Gap: ML Deployment Without IT Integration

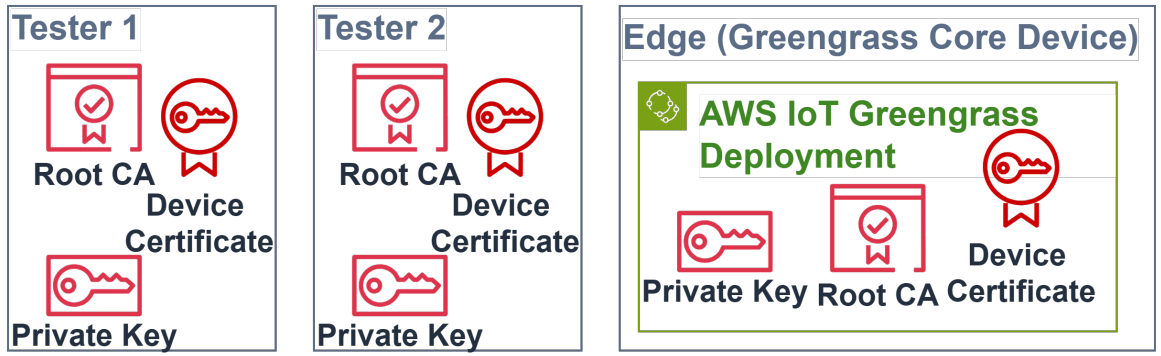
The Problem: Today, when a fabless company wants to deploy ML at an OSAT

- They don't own the testers or the OSAT's network
- Custom IT integration is required at every site
- Security concerns across organizational boundaries
- No standardized way to manage models across multiple OSATs

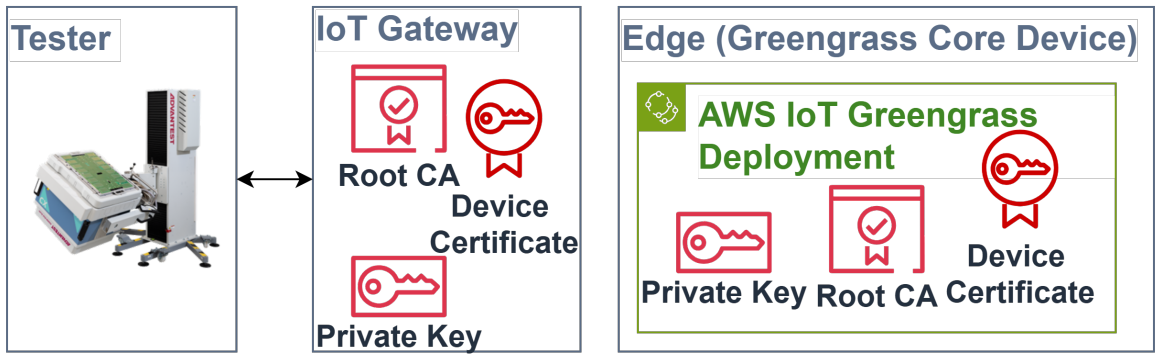
Our Approach: Treat tester as IoT Thing

- Provisioned automatically via fleet templates — zero-touch at scale
- Secured with X.509 certificates — no shared credentials, no OSAT network changes
- Cloud Managed — deploys to hundreds of testers
- Supports IDM, OSAT, and consigned business.

Scenario A: IDM, Fabless Consigned Testers



Scenario B: Multi-tenant OSAT



Fabless ML deployment goes from a multi-month IT project to a configuration task.

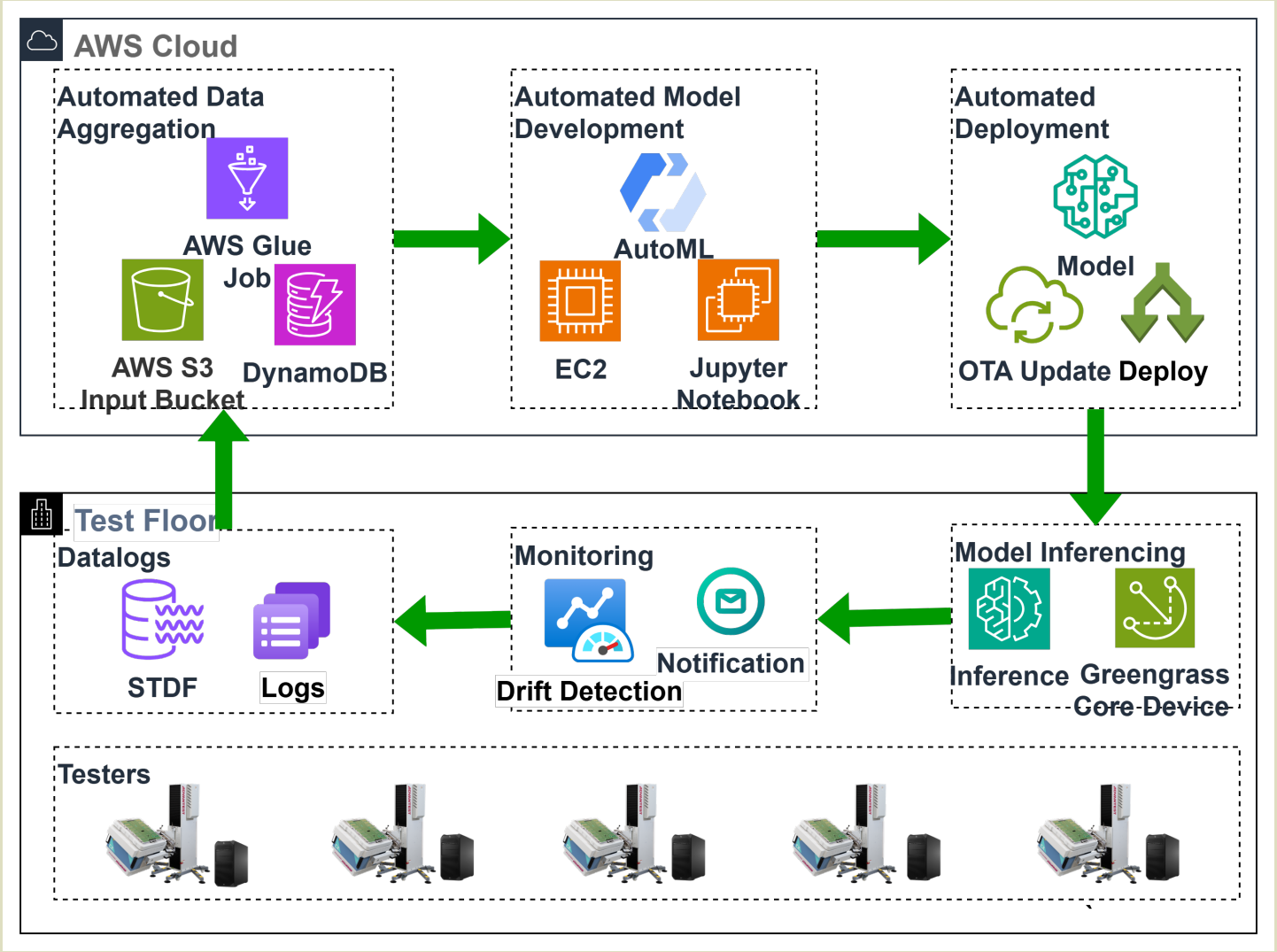
Closed-Loop Automated Pipeline

Closed-Loop ML Pipeline

- Data Aggregation and processing
- Model development
- Deployment
- Drift monitoring
- Update and redeployment

Automation: From new STDF data arriving in S3 to an updated model running at the edge

- S3 event-triggered data processing
- AutoML for model training and tuning
- OTA deployment and redeployment
- Drift detection for retraining and update



Impact: Why AutoML + Cloud-IoT Changes the Game

AutoML removes the ML expertise barrier. Cloud removes the infrastructure barrier. IoT removes the deployment barrier. Neither alone solves the problem. Together, they enable a closed-loop system a test engineering team can operate.

Category	Metric	Evidence
Engineering	Development effort	36× faster; <10 lines of code
	Compute scalability	2× faster training on larger EC2
Operation	Deployment	OTA via Greengrass; zero-touch fleet provisioning
	Lifecycle	Automated drift detection, retraining, redeployment
	Interoperability	Cross-OSAT deployment via standardized IoT
Test Outcome	Test time	V_{min} search: N steps → 1 step
	Latency	<130ms end-to-end inference
	Accuracy	$R^2 > 0.90$, matching expert-tuned models

COPYRIGHT NOTICE

www.testconx.org