



A High AI-Q[™]
Company



DevOps Transformation for an IoT Analytics Leader

Optimizing business agility and deployment frequency through an automated, containerized Lambda Architecture and a stable, multi-region cloud environment.

Overview

We engineered a modular Lambda Architecture using the LAMP stack, Apache Storm, and Spark to process diverse IoT data streams in real-time.

- Reduced environment setup time from one week to just a few hours by leveraging AWS CloudFormation and immutable infrastructure patterns.
- Established a robust CI/CD pipeline using Docker, Jenkins, and Ansible, enabling seamless code movement from developer stacks to globally distributed production clusters.



Client Profile

This US-based technology provider is at the forefront of the IoT revolution. Their proprietary hardware bridges sensors and tags with enterprise-grade ERP platforms and data warehouses, organizing complex raw data into consumable formats for global industrial leaders.

Challenges: Complexity at Scale

The transition from traditional data management to high-frequency IoT analytics presented several operational hurdles:

- **Deployment Friction:** Moving code through Dev, QA, and Production environments was slow and prone to "works on my machine" inconsistencies.
- **Infrastructure Rigidity:** Setting up new environments for different global centers was a week-long manual process.
- **Reliability Demands:** The system required a "zero-downtime" architecture capable of handling high-volume data streams with no service disruption.
- **Operational Blind Spots:** Developers lacked the access and tools necessary to diagnose production issues quickly, leading to slow recovery times.

QBurst Solution: Immutable Infrastructure & Lambda Orchestration

We implemented a hybrid DevOps model where our tech leads controlled the product master repositories and infrastructure-as-code (IaC) templates. The core of the solution is a Lambda Architecture designed for both speed and accuracy.

The technical strategy focused on automation and high availability:

- **Tiered Environment Strategy:** We used embedded CloudFormation stacks to model infrastructure in layers. This allows developers to launch and destroy their own isolated "mini-stacks" while ensuring Production resources are distributed across multiple AWS Availability Zones.
- **Containerized Continuous Delivery:** Services were launched using Docker containers to eliminate friction. We built a CI/CD pipeline where a GitHub push triggers Jenkins to build a Docker image, push it to a private registry, and execute an Ansible playbook for instant deployment.
- **The Lambda Layers:**
 - **Speed Layer:** Utilized Apache Storm for real-time processing of incoming sensor data.
 - **Batch/Serving Layers:** Leveraged HDFS, HBase, and Spark for long-term storage and DataStax Cassandra for high-speed data serving.
- **Proactive 24/7 Monitoring:** Implemented a unified monitoring suite using CloudWatch, Nagios, PagerDuty, and SemaText to ensure 99.9% uptime and adherence to strict SLAs.

Technical Highlights

- **Immutable Infrastructure:** Containers and IaC templates ensure that the production environment is never "patched"—it is always replaced with a clean, tested version.
- **Real-Time Data Streams:** Engineered to consume and process numerous heterogeneous data sources for downstream analytics.

- **Disaster Recovery Mastery:** Defined and maintained aggressive RTO and RPO targets, ensuring rapid resource re-launch in the event of a regional AWS failure.
- **AI & Data Science Ready:** Integrated Python (Flask & SciKit Learn) to provide advanced analytics and predictive insights on the processed IoT data.

Impact

- **85% Faster Provisioning:** Environment setup time plummeted from one week to a few hours, drastically increasing developer productivity.
- **Accelerated Time-to-Market:** Improved deployment frequency allowed the client to roll out new features and analytics components ahead of the competition.
- **Higher System Stability:** The use of Docker containers provided isolated services, making application upgrades and rollbacks simple and risk-free.
- **Resilient Operations:** Established a more stable operating environment with a significantly lower failure rate and faster mean time to recovery (MTTR).
- **Focus on Innovation:** By automating the "maintenance" aspect of IT, the client's engineering team was able to dedicate more time to adding value through new IoT product features.