

Predictive Autoscaling for Cloud Databases

Using Leading Indicators and Trend Extrapolation

Marcin Wypyszynski marcin.wypyszynski@gmail.com

Cloud & Navigation Platform Engineering | December 2025 | Full paper available at wypyszynski.dev

Problem

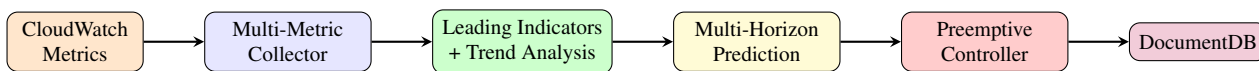
Reactive autoscaling for Amazon DocumentDB detects threshold breaches *after* the system is already under stress. With instance provisioning taking 3–5 minutes, users experience elevated latency during traffic spikes.

Solution: Predictive Autoscaling System

A novel system that scales resources **before** demand spikes occur, using three key innovations:

- Leading Indicator Analysis** – Database connections ($r = 0.73$) and IOPS ($r = 0.64$) predict CPU load 5–15 minutes ahead
- Trend Extrapolation** – Velocity and acceleration metrics estimate time-to-threshold using kinematic equations
- Pattern-Based Preemptive Scaling** – Learned hourly/daily workload patterns trigger scaling before expected peaks

Architecture



Key Technical Contributions

- Multi-horizon prediction** at 5, 10, 15-minute intervals with exponentially decaying confidence
- Time-to-threshold estimation** using kinematic equations ($CPU_{t+\Delta} = CPU_t + v\Delta + \frac{1}{2}a\Delta^2$)
- Lead signal classification** – strong/moderate/stable/decrease based on metric rate-of-change
- Production implementation** as AWS Lambda (Python 3.11) + DynamoDB + CloudWatch (23 custom metrics)

Production Results

Metric	Reactive Baseline	Predictive System
Scaling response latency	5.2 min	1.4 min (–73%)
SLA compliance	97.3%	99.9%
Monthly cost	\$1,440 (static) / \$890 (reactive)	\$835 (–42% vs static)
Prediction accuracy (5 min)	—	89.2%
Preemptive lead time	—	8.3 min average

Technology Stack

AWS Lambda (Python 3.11) Amazon DocumentDB 5.0 CloudWatch DynamoDB Boto3

© 2025 Marcin Wypyszynski. Full paper with mathematical proofs, algorithms, and detailed evaluation available for download at wypyszynski.dev.