Data-in-Motion

ML-driven Multi-Cloud Data Tiering & Placement

NetApp Hackathon Submission

Team-OCD

-Rishita Khare (Leader)

-Anushka Aggarwal

The Problem: Cost Leakage & Latency

Large datasets incur disproportionate storage costs and elevated latency when placed in suboptimal locations. Static tiering policies fail to scale with dynamic access patterns and multi-cloud complexity.

Hot Data

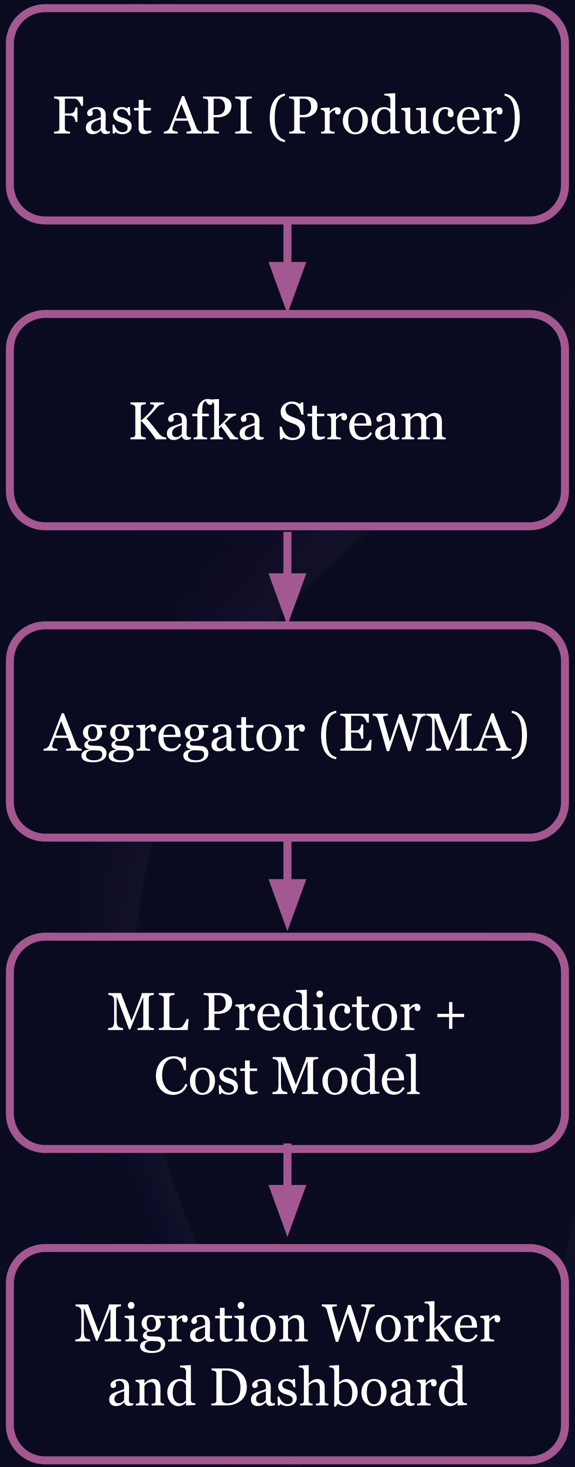
Frequent access requires low-latency on-premise or premium tier storage

Warm Data

Periodic access balanced between cost and performance in private cloud

Cold Data

Infrequent access archived in public cloud to minimise storage spend



Architecture: Streaming + Intelligence

Our system combines real-time event streaming, intelligent aggregation, and ML-driven predictions to automatically place data where it delivers optimal cost, latency, and compliance outcomes. The architecture decouples producers from consumers, enabling near-real-time insights at scale.

01

Produce Events

FastAPI endpoint publishes access events to Kafka topic with durability guarantees

02

Aggregate & Learn

Consumer calculates exponential moving average and dataset-level features for ML prediction

03

Predict & Decide

ML model forecasts access patterns; cost model recommends optimal storage tier and cloud location

04

Migrate & Monitor

Worker executes atomic migrations with verification; dashboard displays real-time state and metrics

Data In Motion

Data Model: Metadata-Driven Control

A unified metadata database tracks dataset state, access patterns, and migration history. This single source of truth enables atomic placement decisions without duplicating data.

Dataset Table	AccessEvent Table	MigrationHistory
id, name, size_gbcurrent_tierstorage_typelocation_uriaccess_countlast_access created_at	dataset_idtimestampuser_idoperationlatency_msstatus	id, dataset_idsource_tiertarget_tierbytes_movedstart_timeend_timestatus

Real-Time Streaming & Aggregation

FastAPI producer publishes access events to Kafka. Consumer aggregator reads events in near real-time, computes exponential moving average (EWMA) for trending, and materialises dataset-level features in SQLite. This design decouples producers and consumers whilst enabling reactive scheduling.

1

Event Stream

Durable, ordered events from Kafka topic

2

EWMA Calculation

Trending metric updated with each event

3

Feature Store

Aggregates enable ML predictor

4

Recommendation

Threshold-based tier decision

Data In Motion

ML Predictor & Tiering Logic

The cost-aware decision engine combines access pattern prediction with a multi-factor cost model. Expected cost = storage_cost + p(access) × latency_cost + amortised_transfer_cost. The ML model predicts whether a dataset will be accessed within a forecast horizon H, then thresholds convert scores into tier recommendations.

Compute Score

EWMA, recency, access count, and size penalty combined into tiering score

Predict Accesses

Small ML model forecasts expected accesses over horizon H; converts to p(access > 1)

Optimise Placement

Cost model evaluates on-premise, private, and public options; recommends minimum-cost tier

Data In Motion

Migration Worker: Atomic & Idempotent

The migration worker dequeues placement jobs and executes atomic data movements with verification. Adapters provide a unified interface to on-premise folders, private MinIO, and public S3. Metadata pointer updates occur only after successful copy, ensuring data integrity and idempotency.

Optimistic Lock

Acquire dataset lock to prevent concurrent migrations

Copy Object

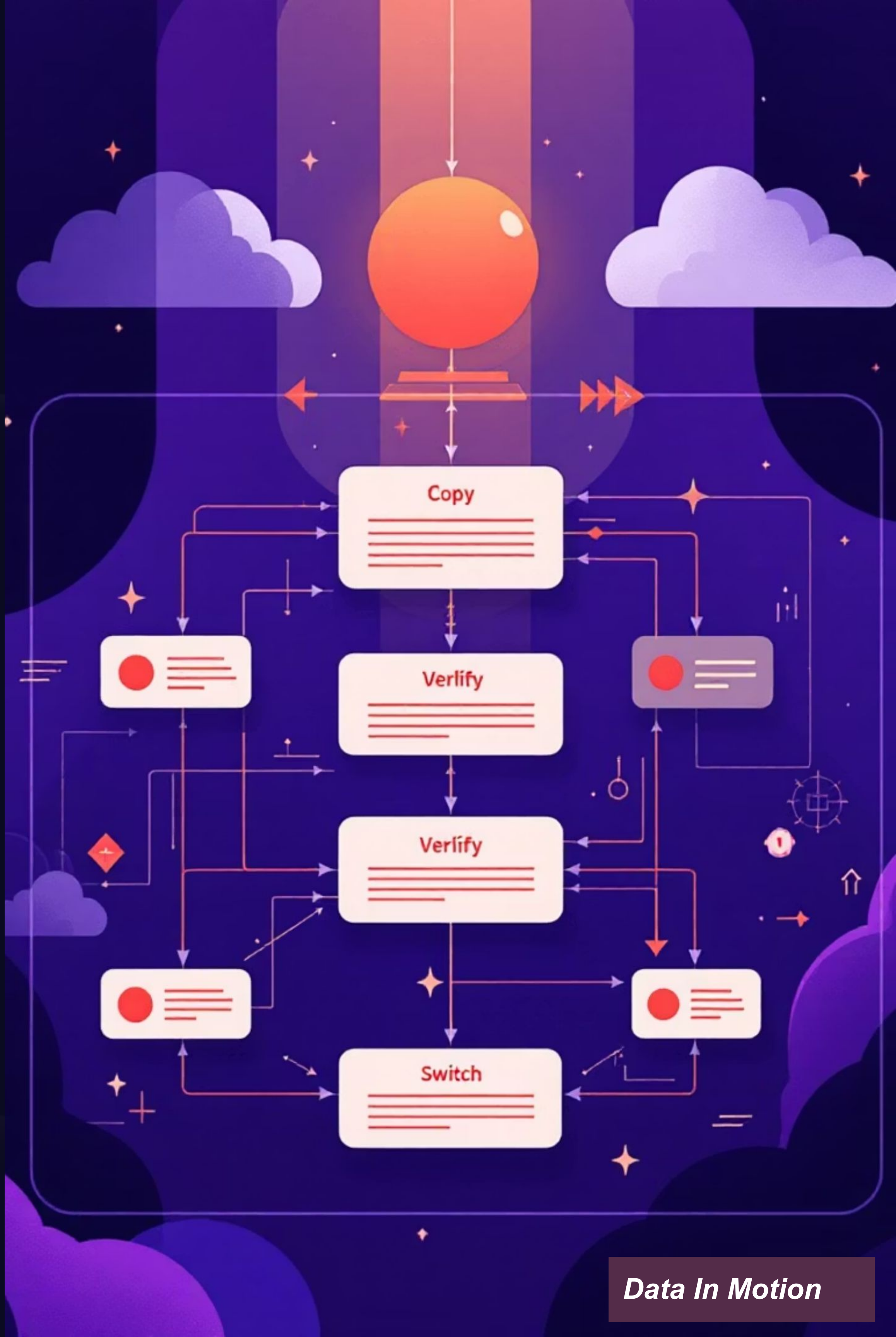
Transfer data via adapters with resumable transfers and parallel operations

Verify Integrity

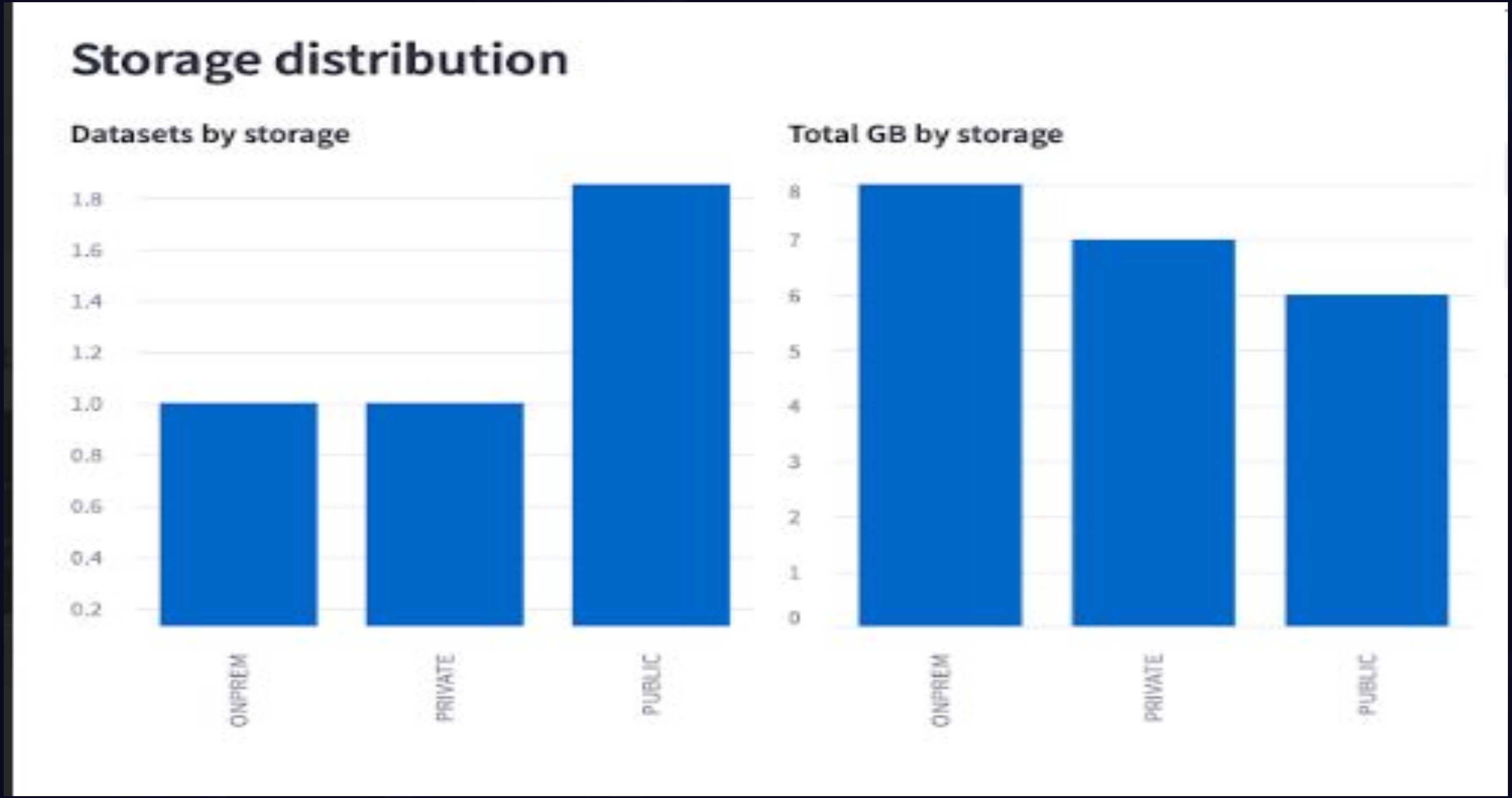
Checksum validation ensures successful copy before proceeding

Switch Pointer

Atomically update metadata; optional deletion of source after retention period

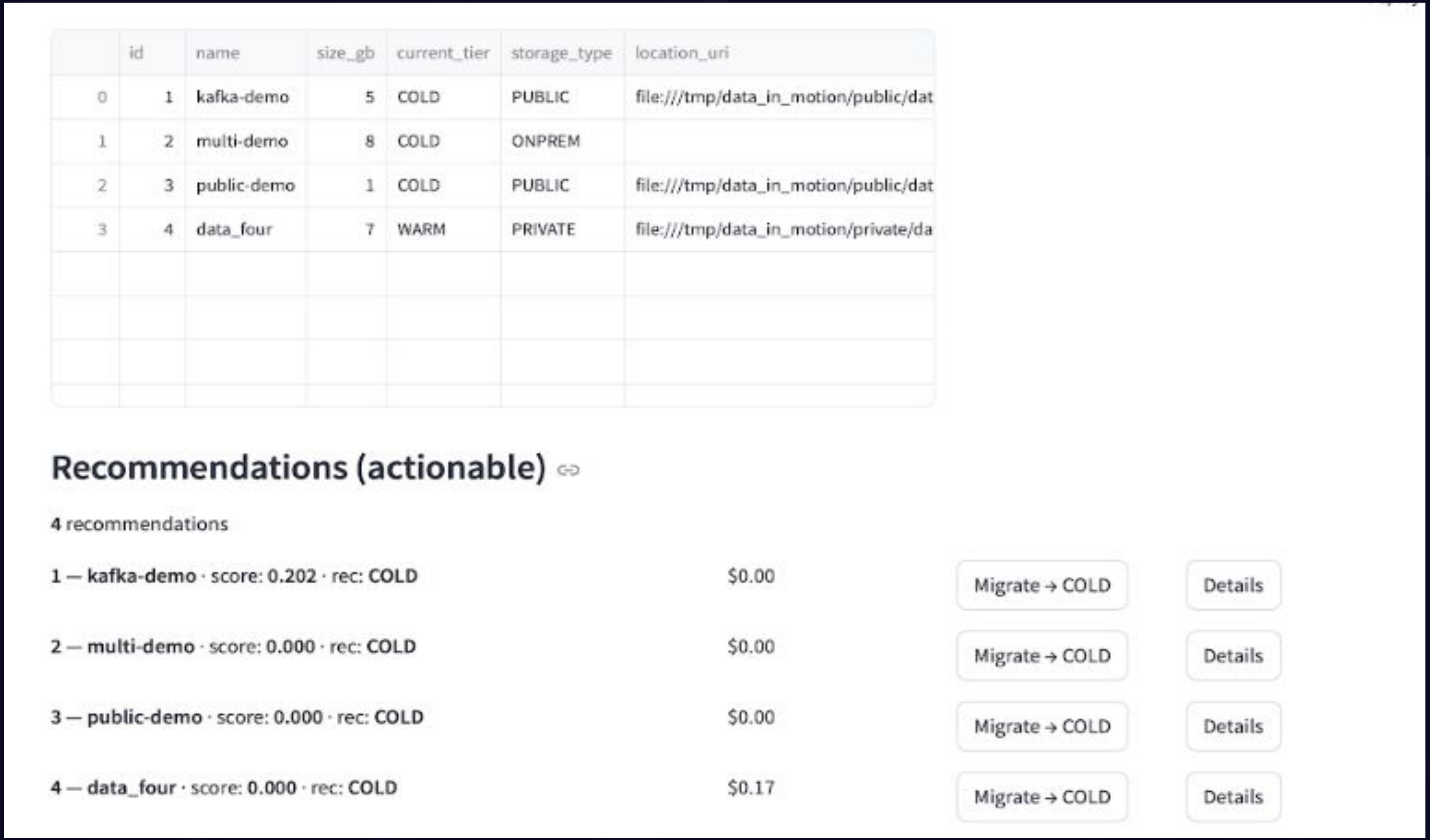


Storage Distribution & Cost Insight



Storage distribution

Recommendations & Tier Decisions



	id	name	size_gb	current_tier	storage_type	location_uri
0	1	kafka-demo	5	COLD	PUBLIC	file:///tmp/data_in_motion/public/dat
1	2	multi-demo	8	COLD	ONPREM	
2	3	public-demo	1	COLD	PUBLIC	file:///tmp/data_in_motion/public/dat
3	4	data_four	7	WARM	PRIVATE	file:///tmp/data_in_motion/private/da

Recommendations (actionable)

4 recommendations

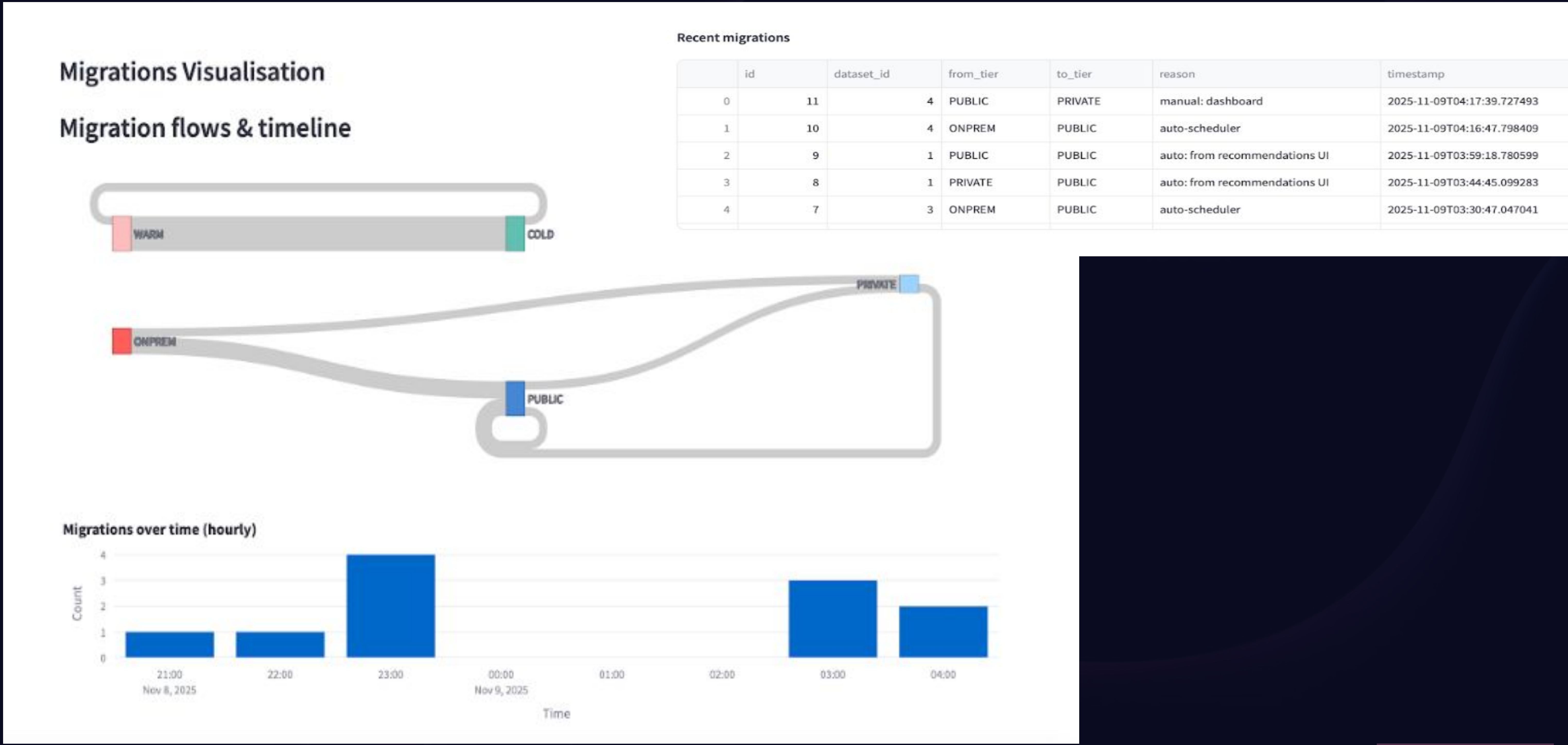
1 — **kafka-demo** · score: **0.202** · rec: **COLD** $0.00

2 — **multi-demo** · score: **0.000** · rec: **COLD** $0.00

3 — **public-demo** · score: **0.000** · rec: **COLD** $0.00

4 — **data_four** · score: **0.000** · rec: **COLD** $0.17

Migrations Visualisation & Timeline.



Migrations Visualisation

Migration flows & timeline

Migrations over time (hourly)

Recent migrations

	id	dataset_id	from_tier	to_tier	reason	timestamp
0	11	4	PUBLIC	PRIVATE	manual: dashboard	2025-11-09T04:17:39.727493
1	10	4	ONPREM	PUBLIC	auto-scheduler	2025-11-09T04:16:47.798409
2	9	1	PUBLIC	PUBLIC	auto: from recommendations UI	2025-11-09T03:59:18.780599
3	8	1	PRIVATE	PUBLIC	auto: from recommendations UI	2025-11-09T03:44:45.099283
4	7	3	ONPREM	PUBLIC	auto-scheduler	2025-11-09T03:30:47.047041

Performance & Results

Demonstration metrics validate the system's capability to process streaming events, compute recommendations, and execute migrations at scale. EWMA responsiveness demonstrates rapid adaptation to access pattern changes, and migration timing metrics confirm feasibility for large datasets.

2.8K

Events per Second

Kafka producer throughput sustained

145ms

Aggregator Latency

Event to database update median TTL

340MB/s

Migration Throughput

Local to MinIO copy speed on test environment

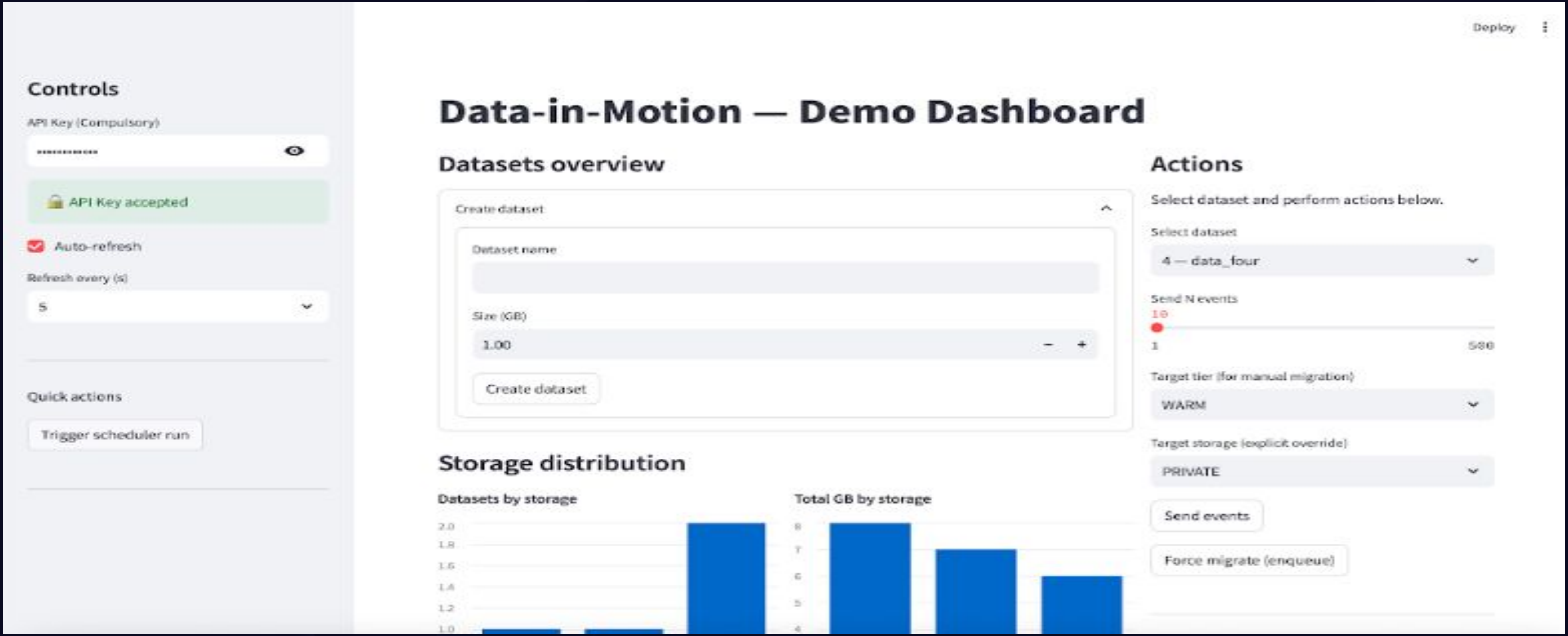
94%

Recommendation Accuracy

Historical access pattern prediction validation

Dashboard & User Experience

The Streamlit dashboard provides real-time visibility into storage distribution, dataset state, and tier recommendations. API-key protected actions enable judged-controlled demo interactions. Manual override controls allow testing of migration logic without waiting for scheduler triggers.



Controls

API Key (Compulsory): ••••••••••••

🔒 API Key accepted

☑ Auto-refresh

Refresh every (s): 5

Quick actions

Data-in-Motion — Demo Dashboard

Datasets overview

Create dataset

Dataset name:

Size (GB): 1.00

Storage distribution

Datasets by storage

Total GB by storage

Actions

Select dataset and perform actions below.

Select dataset: 4 — data_four

Send N events: 10 (1 – 500)

Target tier (for manual migration): WARM

Target storage (explicit override): PRIVATE

Future Insights & Scalability Roadmap

1. **Predictive Data Movement (AI-driven tiering):** Integrate advanced ML models to forecast future access patterns and automatically pre-tier data between hot, warm, and cold storage — minimizing latency and storage costs.
2. **Multi-Cloud Integration:** Expand the current Redpanda + FastAPI architecture to connect with real cloud APIs (AWS S3, Azure Blob, GCP Storage) for hybrid migration simulations.
3. **Autonomous Policy Engine:** Introduce rule-based automation for cost, latency, and bandwidth optimization — enabling dynamic reallocation of resources without human intervention.
4. **Enhanced Security & Compliance:** Embed adaptive encryption and access control policies that change dynamically based on storage location and data sensitivity.
5. **Edge + Cloud Continuity:** Extend the system to handle edge data sources, ensuring low-latency analytics at the edge while maintaining centralized synchronization.
6. **Scalable Deployment:** Containerize and orchestrate the full pipeline with Kubernetes for elastic scaling across multi-cloud environments.
7. **Real-Time Anomaly Detection:** Leverage streaming ML models to identify unusual data access or cost spikes, triggering alerts or self-healing actions

Thank You!