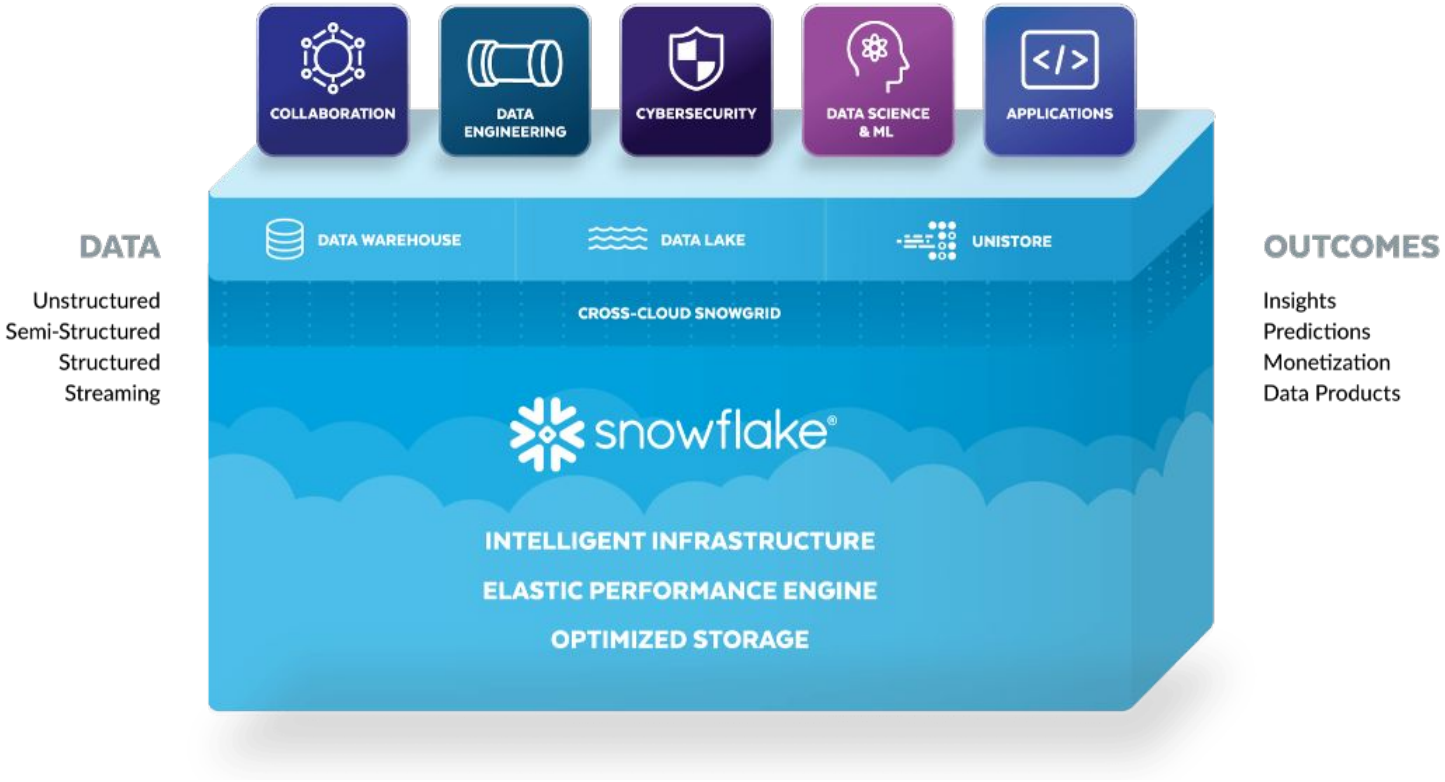
Elastic Cloud Services: Scaling Snowflake's Control Plane

Themis Melissaris[1], Kunal Nabar[1], Rares Radut[1], Samir Rehmtulla[1], Arthur Shi[1], Samartha Chandrashekar[1], and Ioannis Papapanagiotou[2].

[1]Snowflake, Inc., [2]Gemini Trust

Snowflake Data Cloud



DATA

Unstructured
Semi-Structured
Structured
Streaming

OUTCOMES

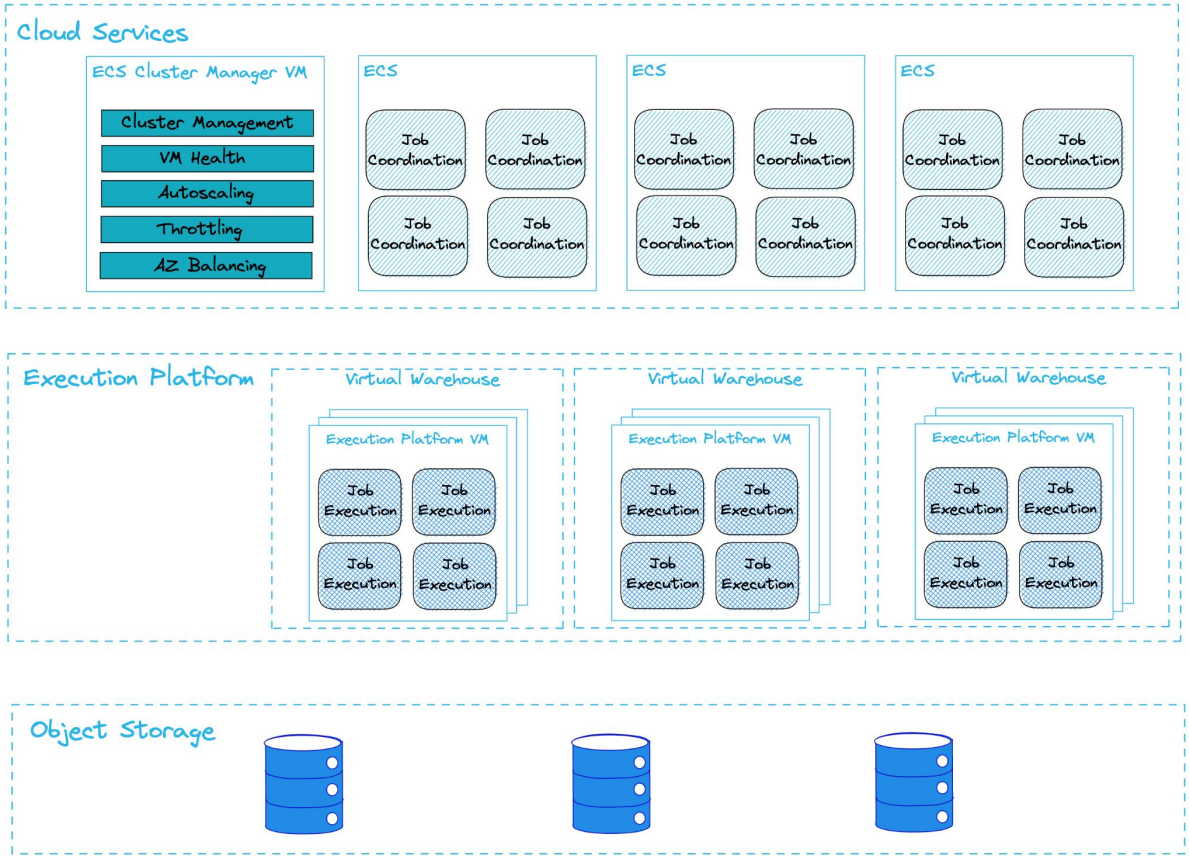
Insights
Predictions
Monetization
Data Products

Snowflake Architecture

Control Plane

Compute Layer

Storage Layer



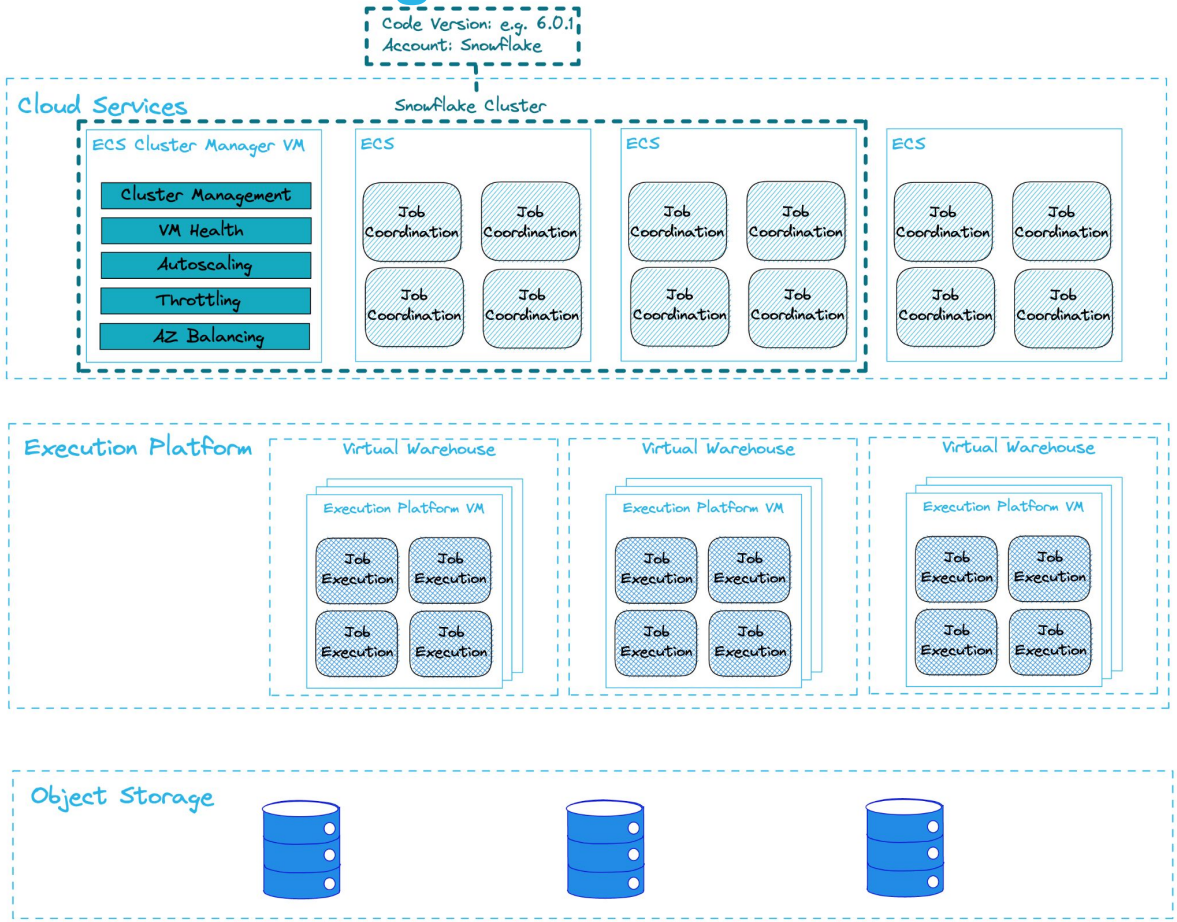
Elastic Cloud Services: Challenges

- Snowflake workloads
  - Workload duration ranges from seconds to days
  - Spiky, unpredictable workload, requirement to provide (ideally) instant capacity
- Snowflake ECS Architecture
  - Self-managed, no knobs system
  - Enables sub systems to evolve independently
    - VM health decoupled from autoscaling
  - Operating on Multiple Cloud Providers
    - Design guided by providing consistent performance and interfaces
  - Operating at Snowflake scale

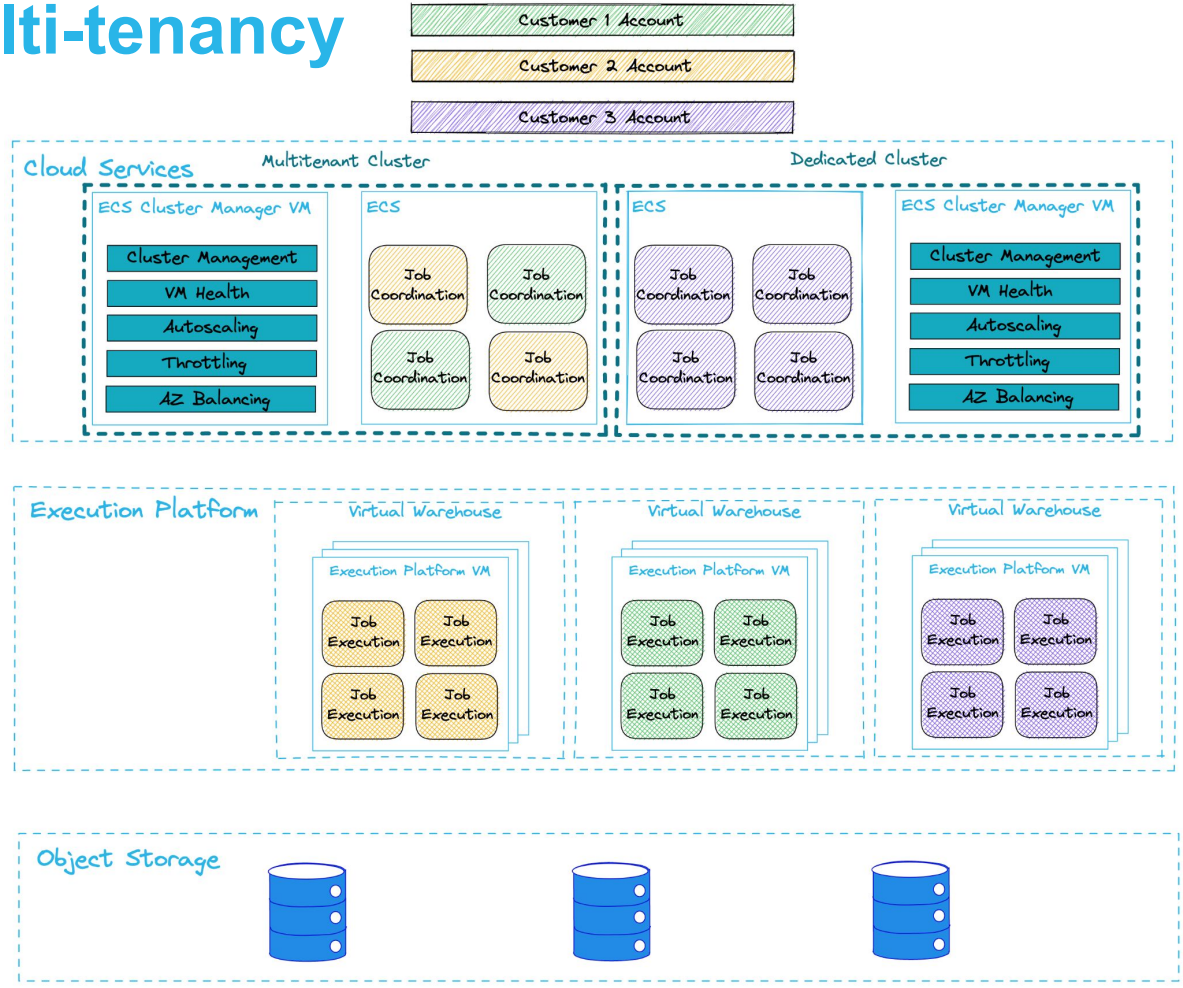
Outline

- Motivation
- Snowflake Architecture & Background
- ECS Cluster Management
  - Resource lifecycle management at scale
  - Automatic code management (paper)
- Balancing across Availability Zones (paper)
  - Cross-cloud and cross-region Replication (paper)
- Autoscaling & Throttling

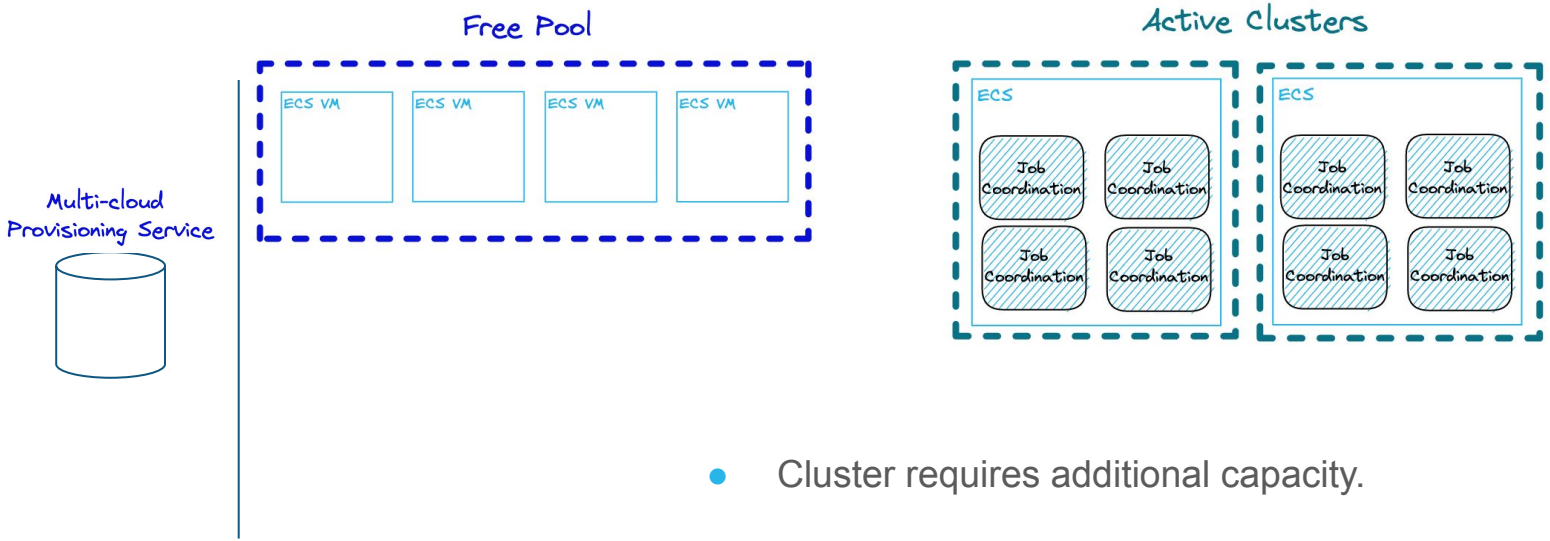
ECS multi-versioning



ECS Multi-tenancy

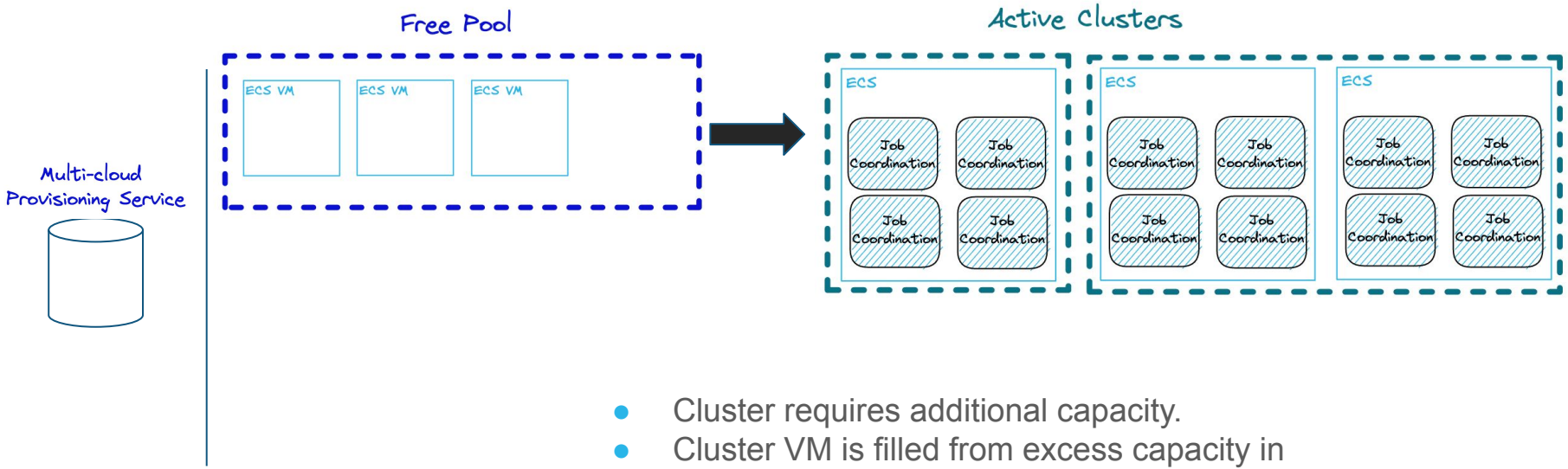


ECS VM provisioning



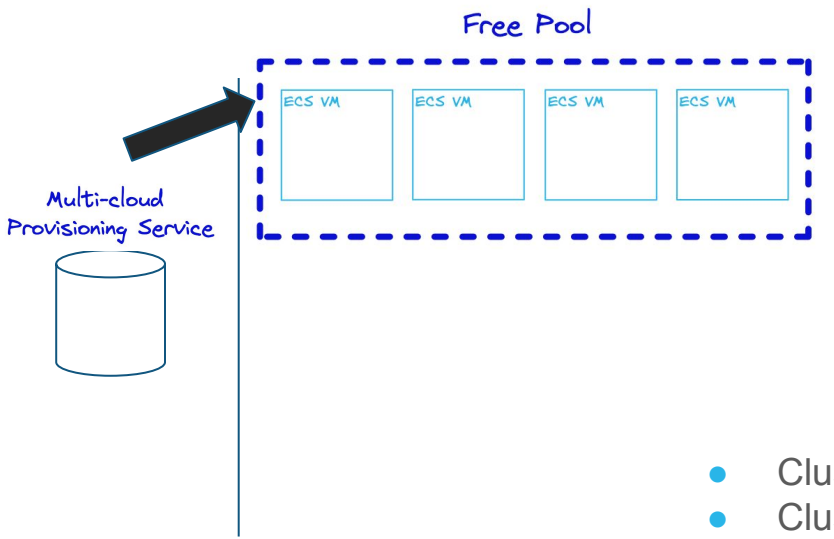
- Cluster requires additional capacity.

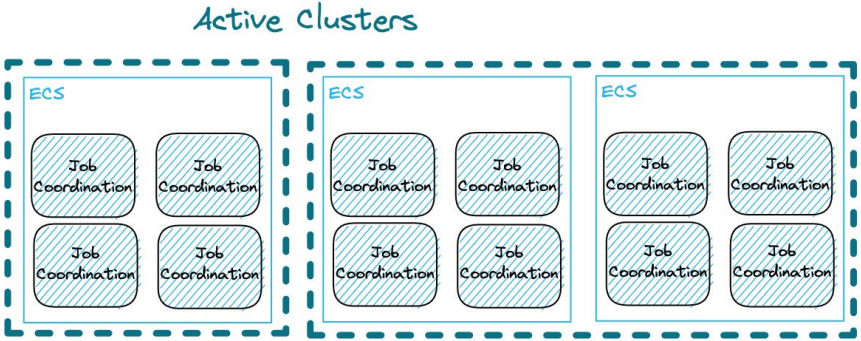
ECS VM provisioning



- Cluster requires additional capacity.
- Cluster VM is filled from excess capacity in the Free Pool.

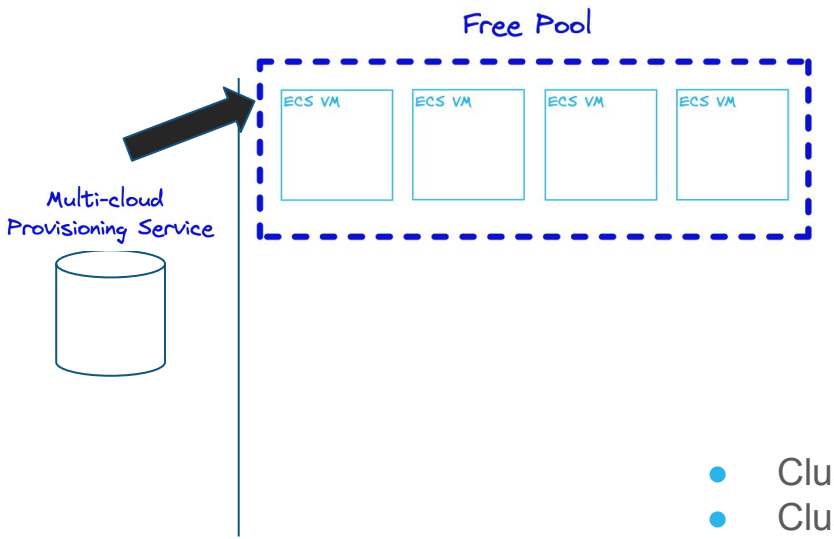
ECS VM provisioning

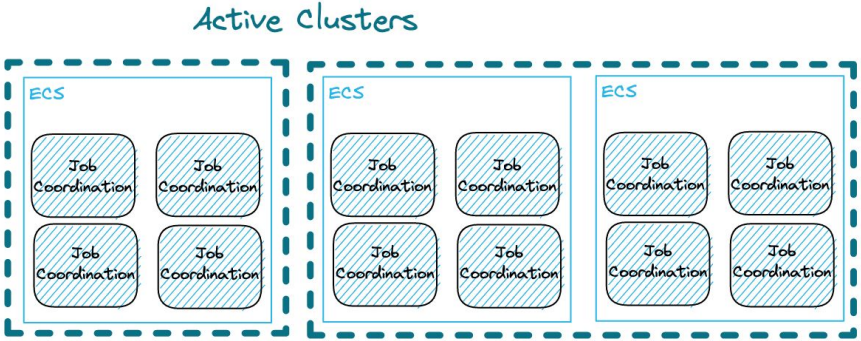




- Cluster requires additional capacity.
- Cluster VM is filled from excess capacity in the Free Pool.
- A new VM is provisioned into the Free Pool.

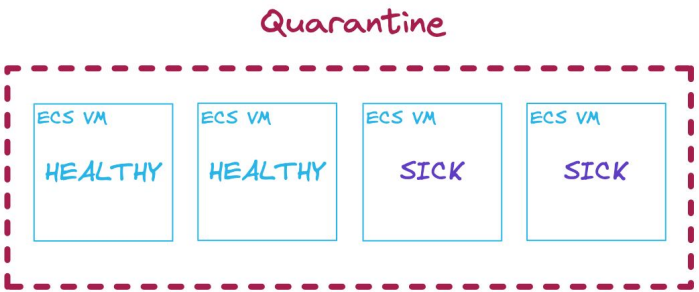
ECS VM provisioning

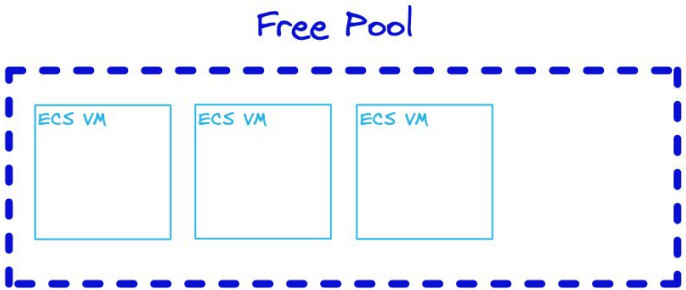


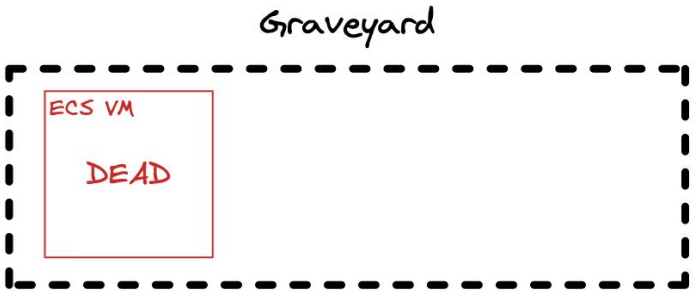


- Cluster requires additional capacity.
- Cluster VM is filled from excess capacity in the Free Pool.
- A new VM is provisioned into the Free Pool.
- Challenge: Expediting VM fulfillment

ECS Lifecycle Management

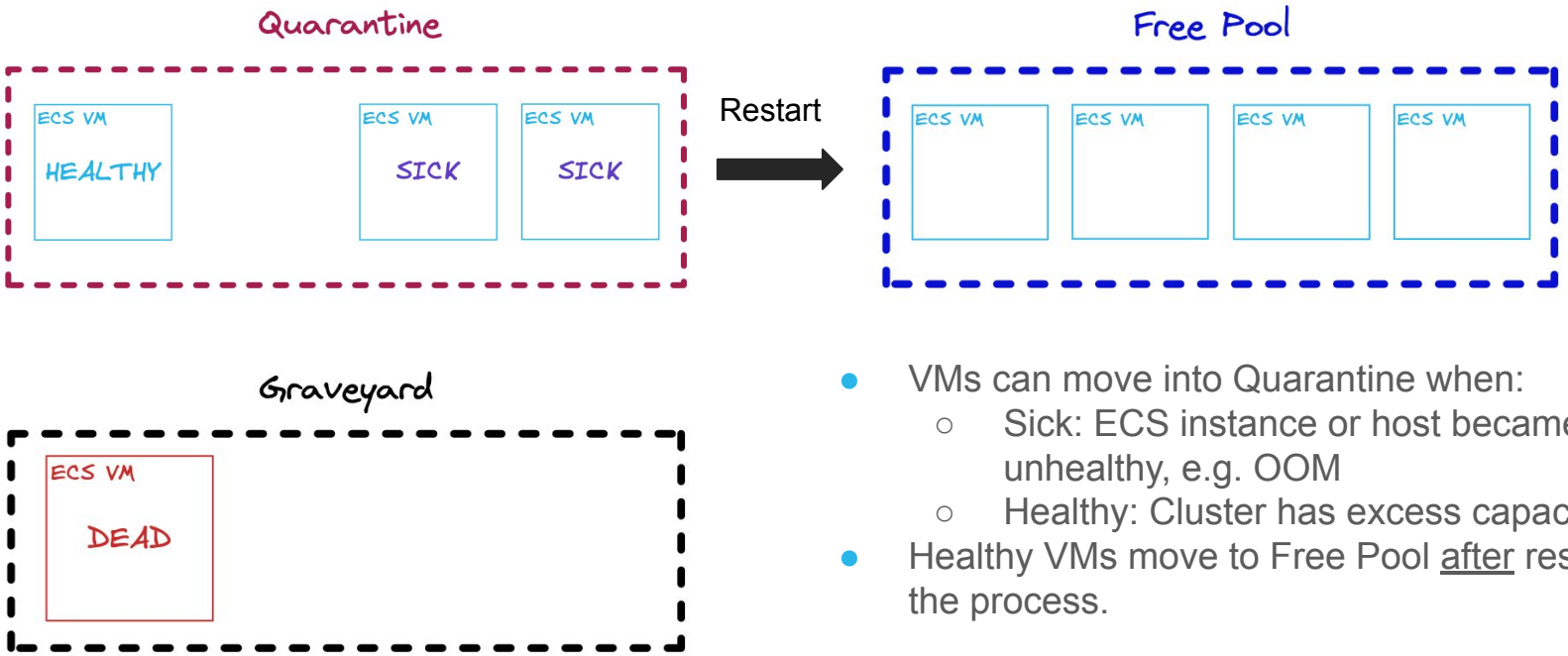






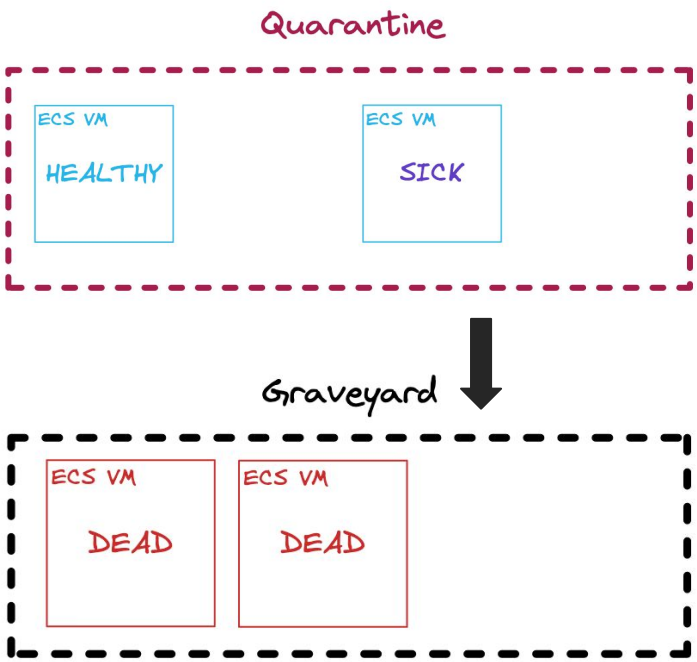
- VMs can move into Quarantine when:
  - Sick: ECS instance or host became unhealthy, e.g. OOM
  - Healthy: Cluster has excess capacity

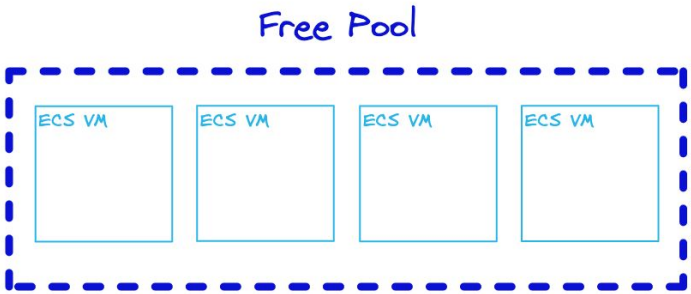
ECS Lifecycle Management



- VMs can move into Quarantine when:
  - Sick: ECS instance or host became unhealthy, e.g. OOM
  - Healthy: Cluster has excess capacity
- Healthy VMs move to Free Pool after restarting the process.

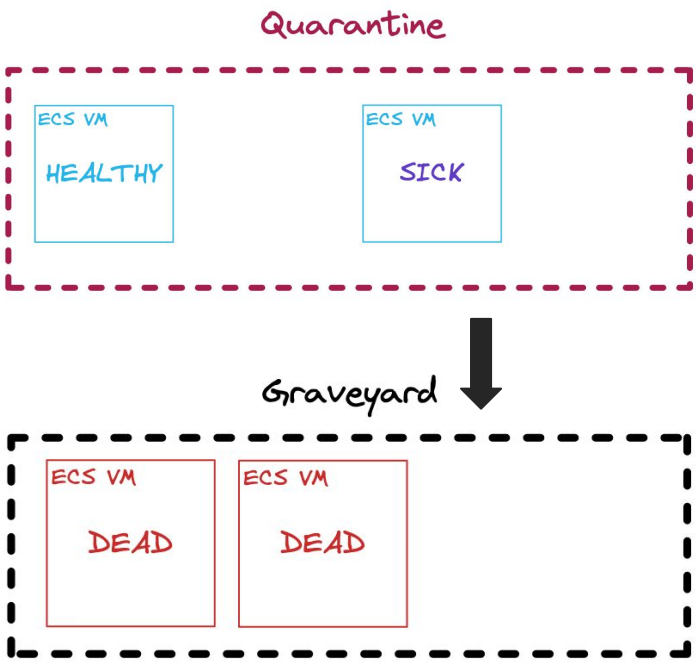
ECS Lifecycle Management

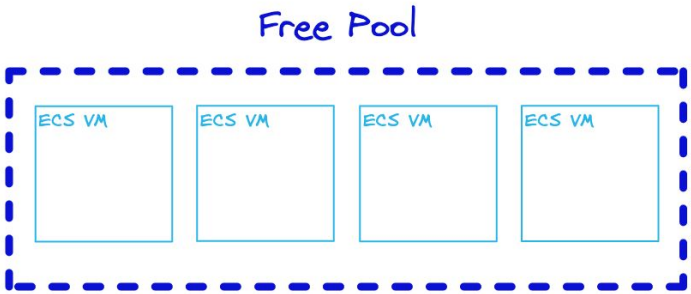




- VMs can move into Quarantine when:
  - Sick: ECS instance or host became unhealthy, e.g. OOM
  - Healthy: Cluster has excess capacity
- Healthy VMs move to Free Pool after restarting the process.
- Sick VMs move to the Graveyard where they are waiting for release to CSPs.

ECS Lifecycle Management





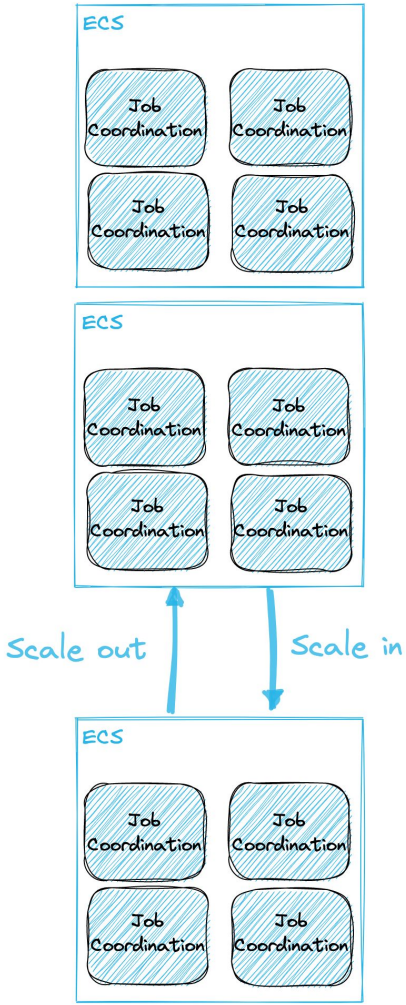
- VMs can move into Quarantine when:
  - Sick: ECS instance or host became unhealthy, e.g. OOM
  - Healthy: Cluster has excess capacity
- Healthy VMs move to Free Pool after restarting the process.
- Sick VMs move to the Graveyard where they are waiting for release to CSPs.
- Challenge: Moving VMs with long running jobs into Quarantine

Outline

- Motivation
- Snowflake Architecture & Background
- ECS Cluster Management
  - Resource lifecycle management at scale
  - Automatic code management (paper)
- Balancing across Availability Zones (paper)
  - Cross-cloud and cross-region Replication (paper)
- Autoscaling & Throttling

ECS Autoscaling

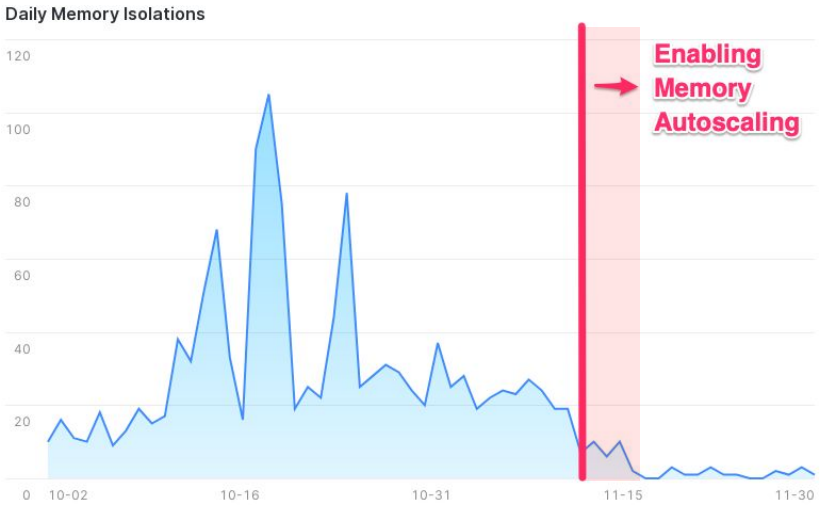
- ECS clusters automatically scale **horizontally** based on aggregated CPU load & throttling rate. Designed to limit cluster oscillations.
- Autoscaling and dynamic throttling work **synergistically**. Scaling will ingest throttling data as signals to increase the capacity.
- Scale **vertically** by changing instance type based on memory, OOMs or Jobs killed or when we need more capacity. ECS autoscaler intelligently chooses instance types based on provider capacity.



ECS Autoscaling

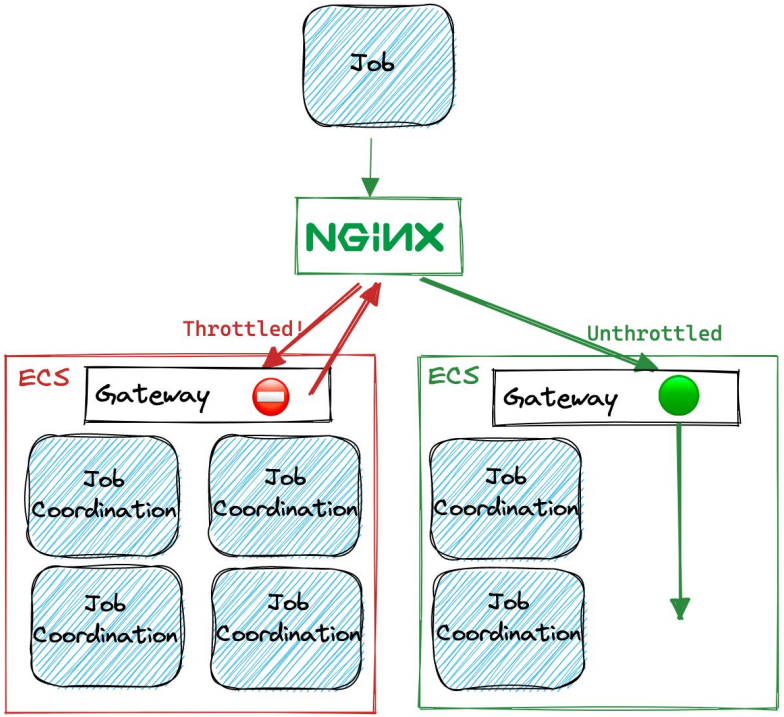
- Reduction in OOMs and memory related issues (jobs killed).
- Working to reduce ECS footprint.
  - Ongoing work in predictively scaling clusters prior to hitting resource limits.

ECS OOMs over time



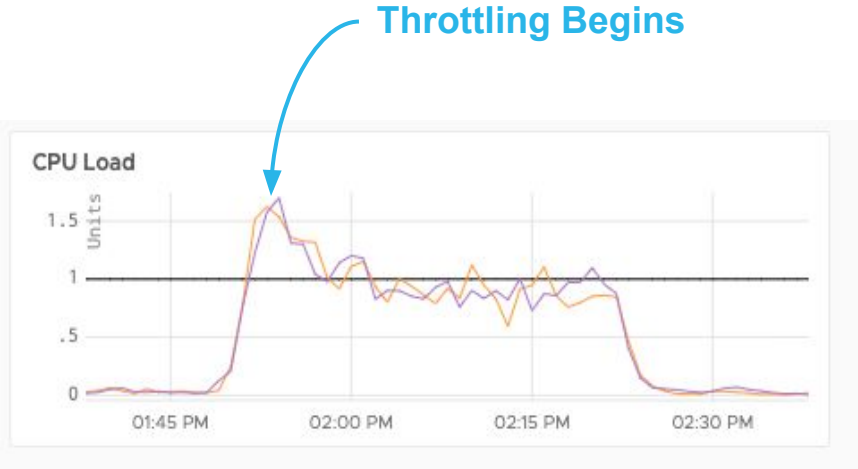
Dynamic Throttling

- Gateways limit ECS backpressure.
- Automatically adjusts ECS gateway size by estimating total allowable jobs after exceeding resource usage thresholds:
  - CPU load
  - Metadata database thread saturation
- General framework that accounts for high variance in job duration.



Dynamic Throttling

- Reduction of **>90%** in cases of overloaded ECS from consistent high CPU load.
- Eliminated jobs waiting due to compilation and reduced percentage of jobs with client retries.



Conclusion

- Presented the design and architecture of Snowflake’s control plane.
- ECS manages Snowflake at scale and is responsible for VM and cluster lifecycle, health management, self-healing automation, account service placement, traffic control, and resource management.
- Extensive results and additional ECS features in the paper.

Thank you!