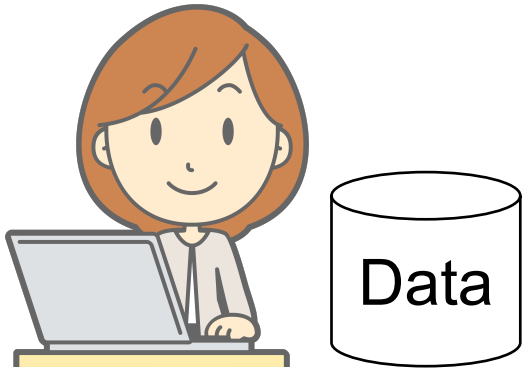
PAUL G. ALLEN SCHOOL
OF COMPUTER SCIENCE & ENGINEERING

SLAOrchestrator: Reducing the Cost of Performance SLAs for Cloud Data Analytics

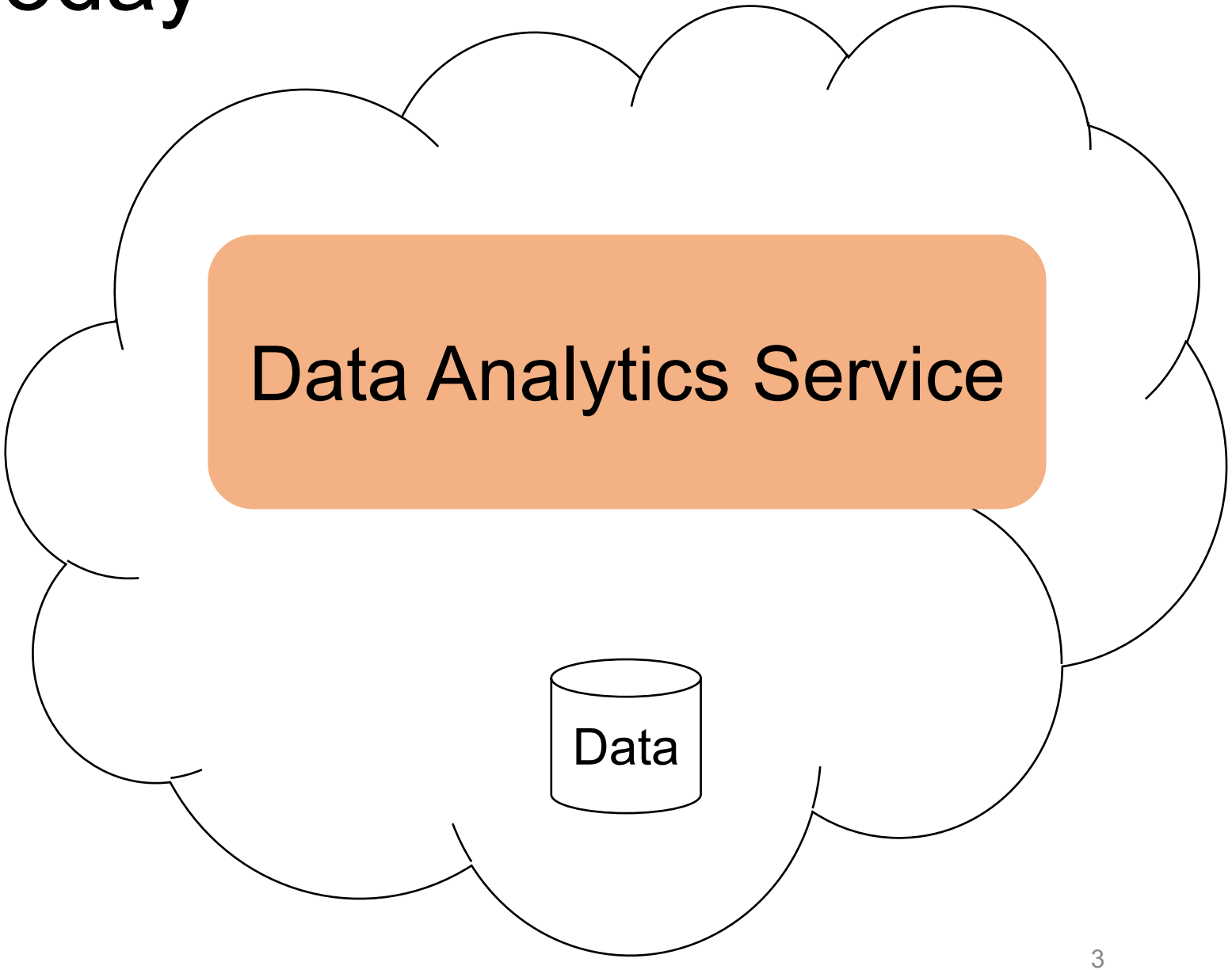
Jennifer Ortiz *(UW)*, Brendan Lee *(now at Spacedust)*, Magdalena Balazinska *(UW)*, Johannes Gehrke *(Microsoft) and* Joseph L. Hellerstein *(eScience Institute at UW)*

Paul G. Allen School of Computer Science & Engineering

University of Washington


Data

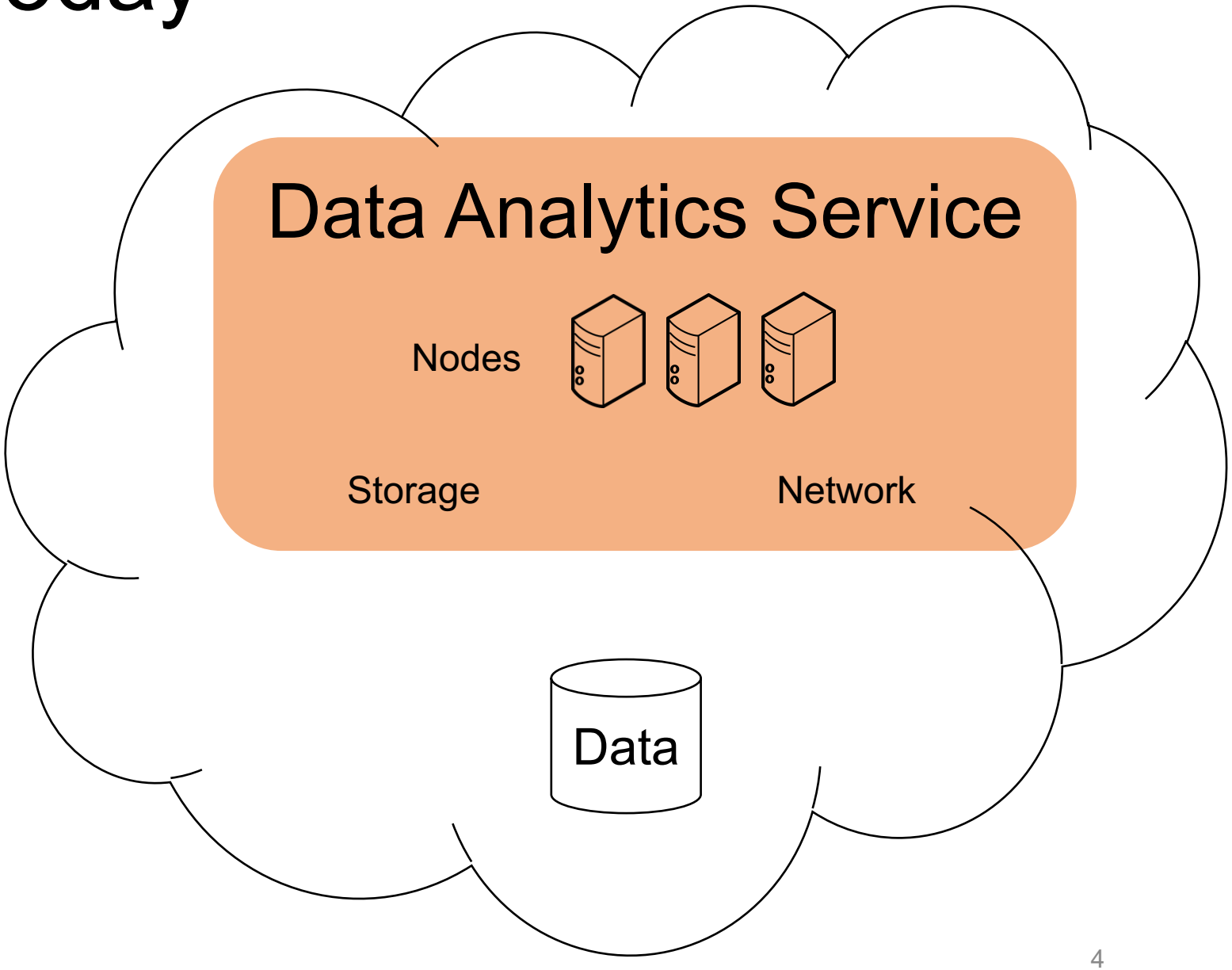
Cloud Services Today



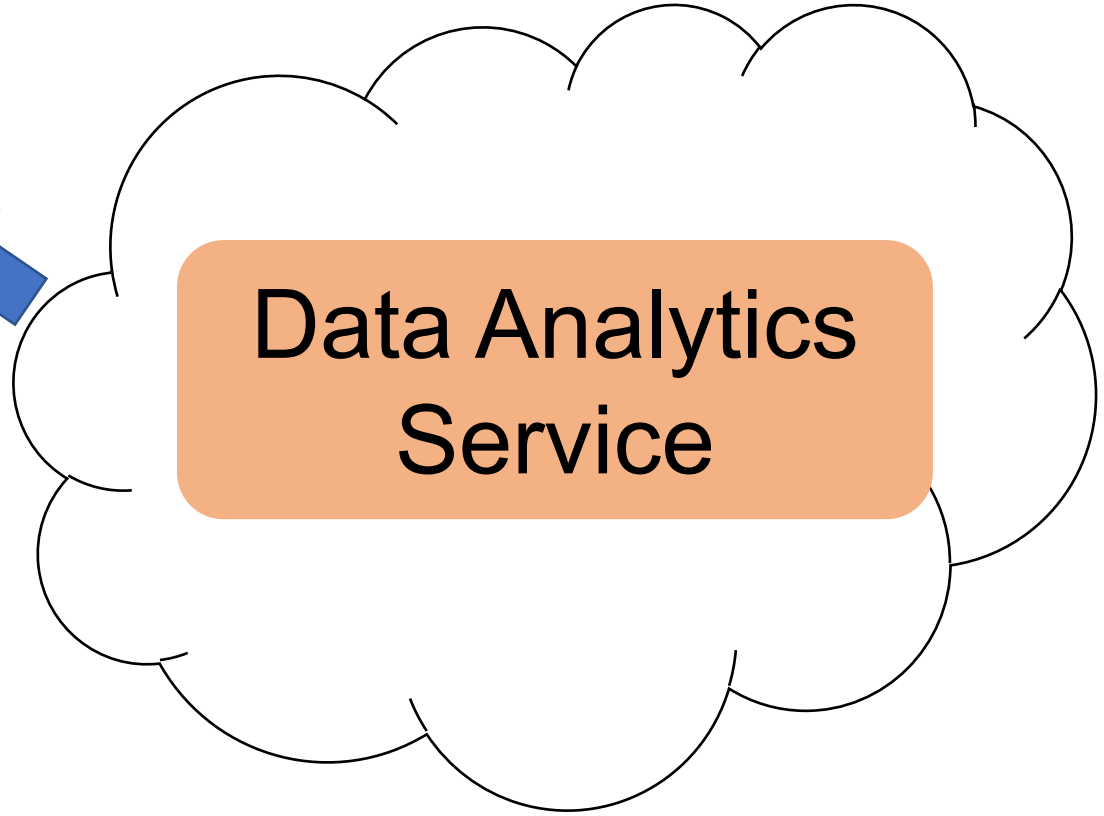
Cloud Services Today

$ How much will it cost me?

How fast will my queries run?



Performance-based Service Level Agreement

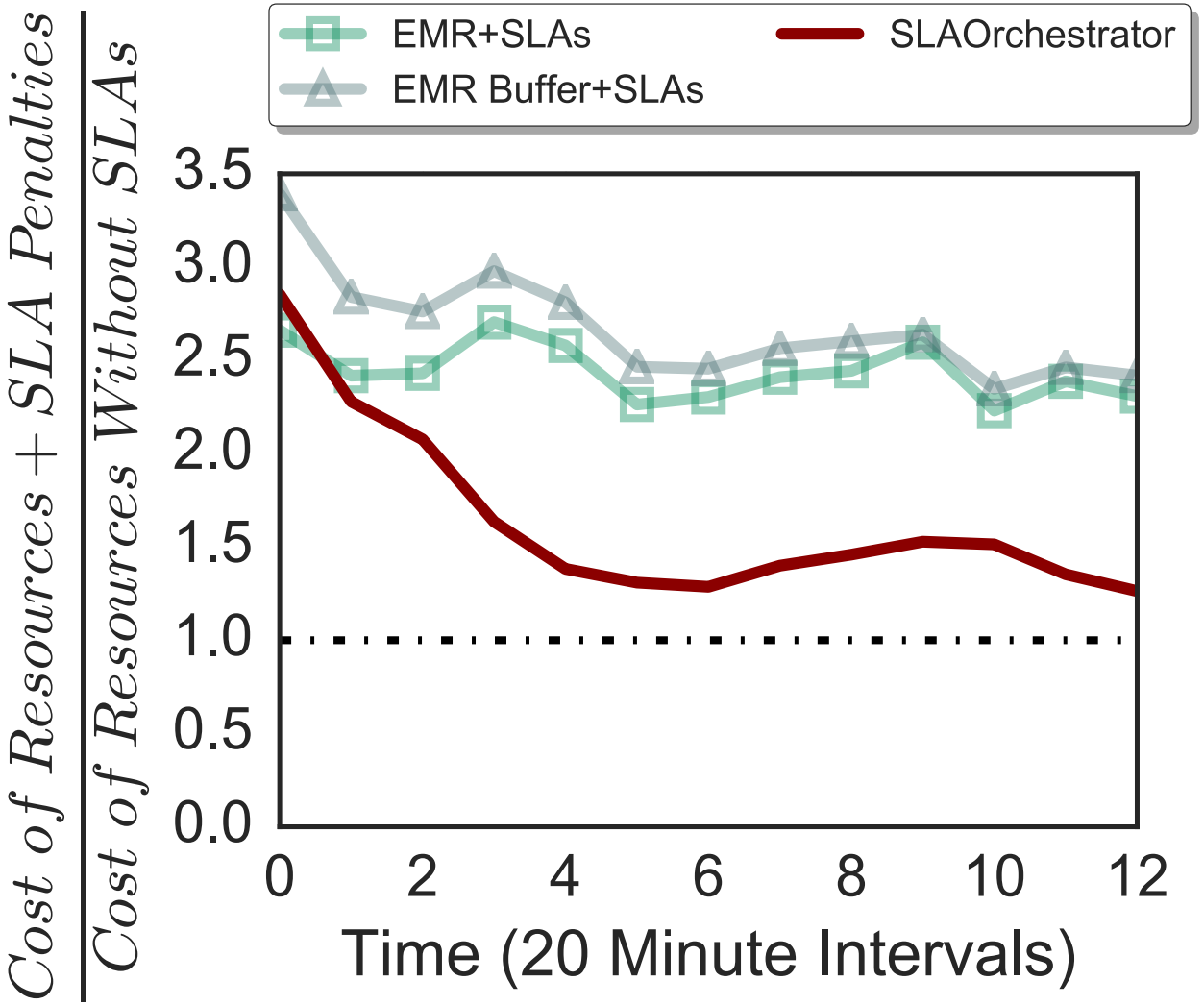


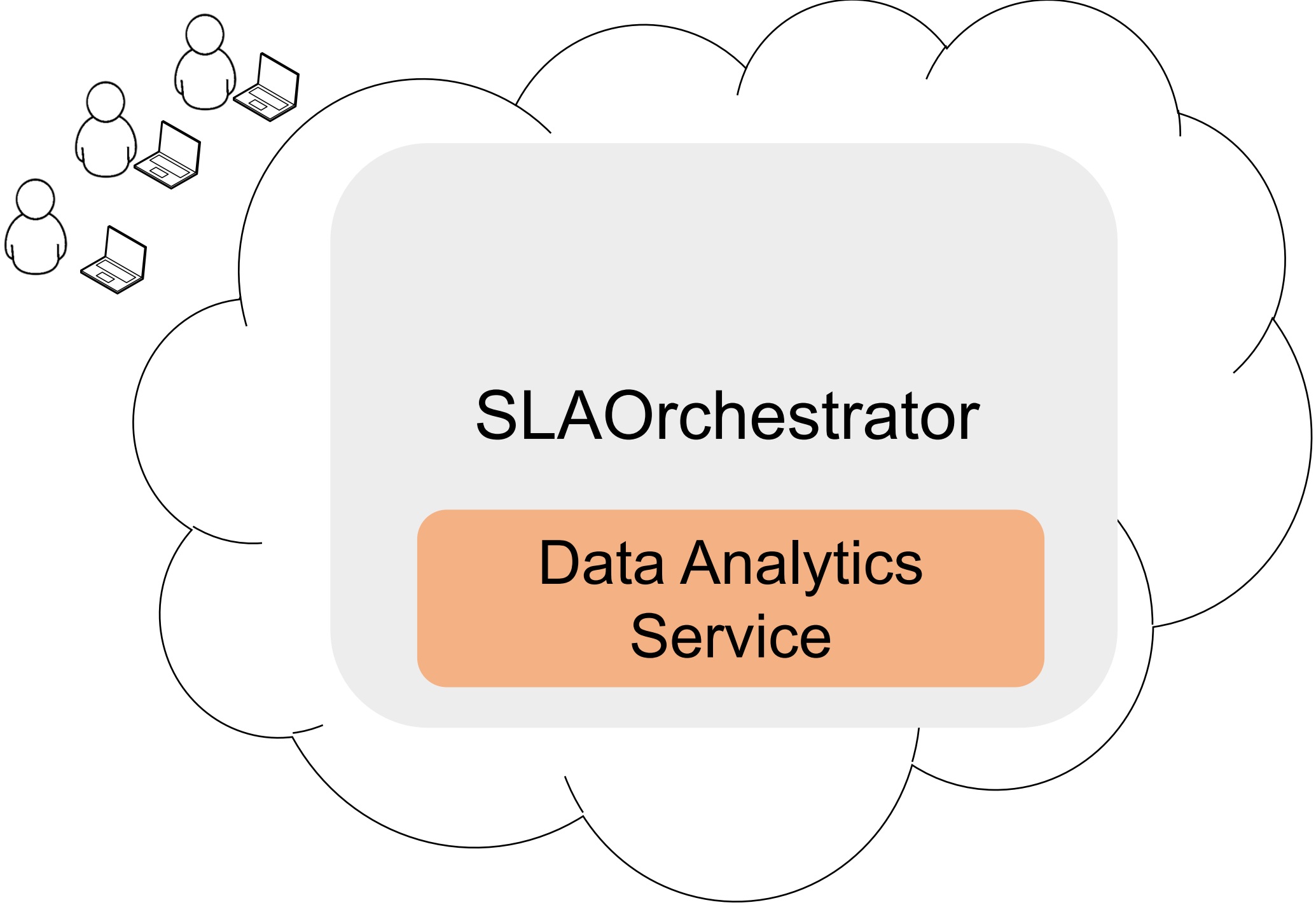
$ How much will it cost me?

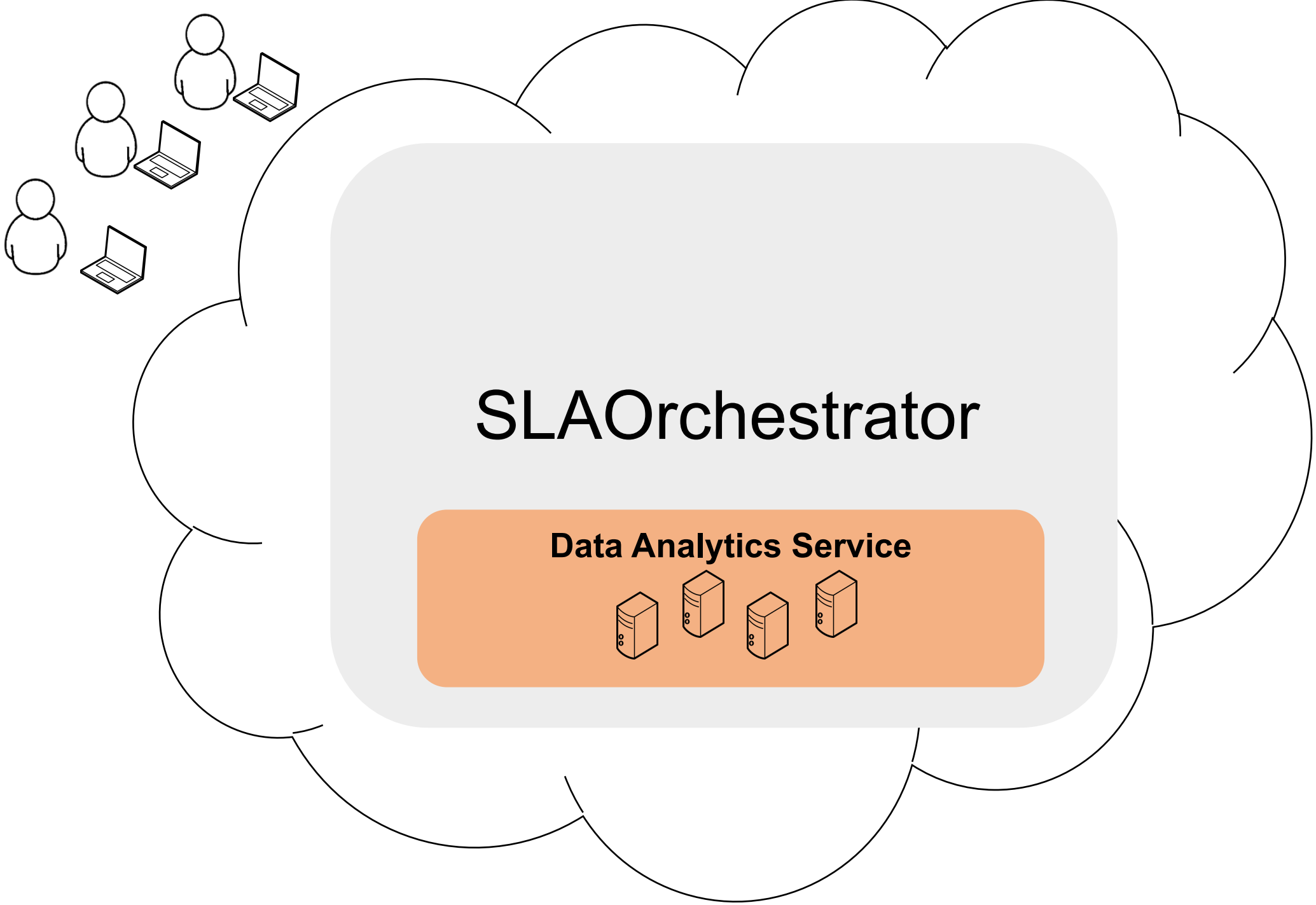
How fast will my queries run?

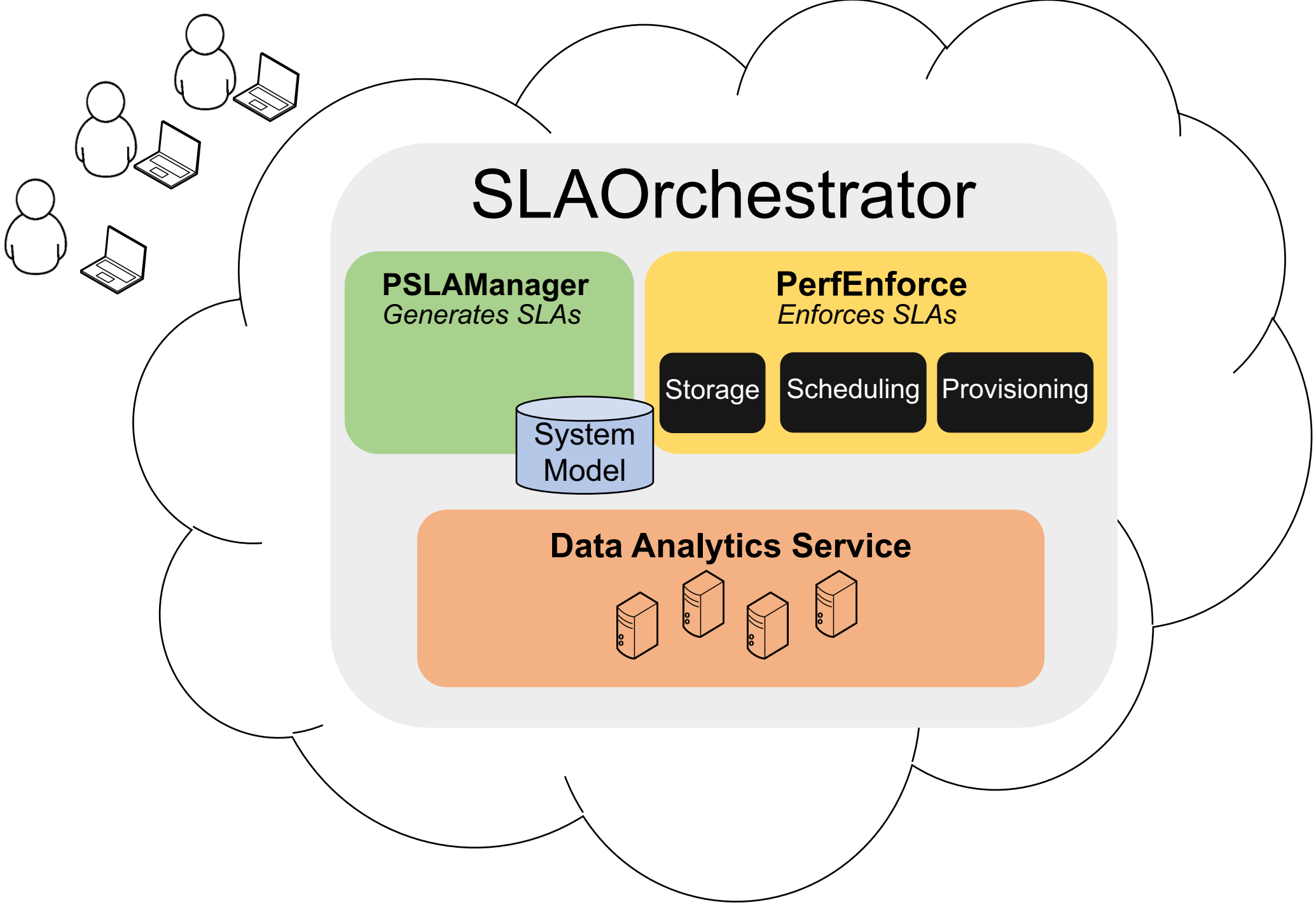
Hard to do cheaply!

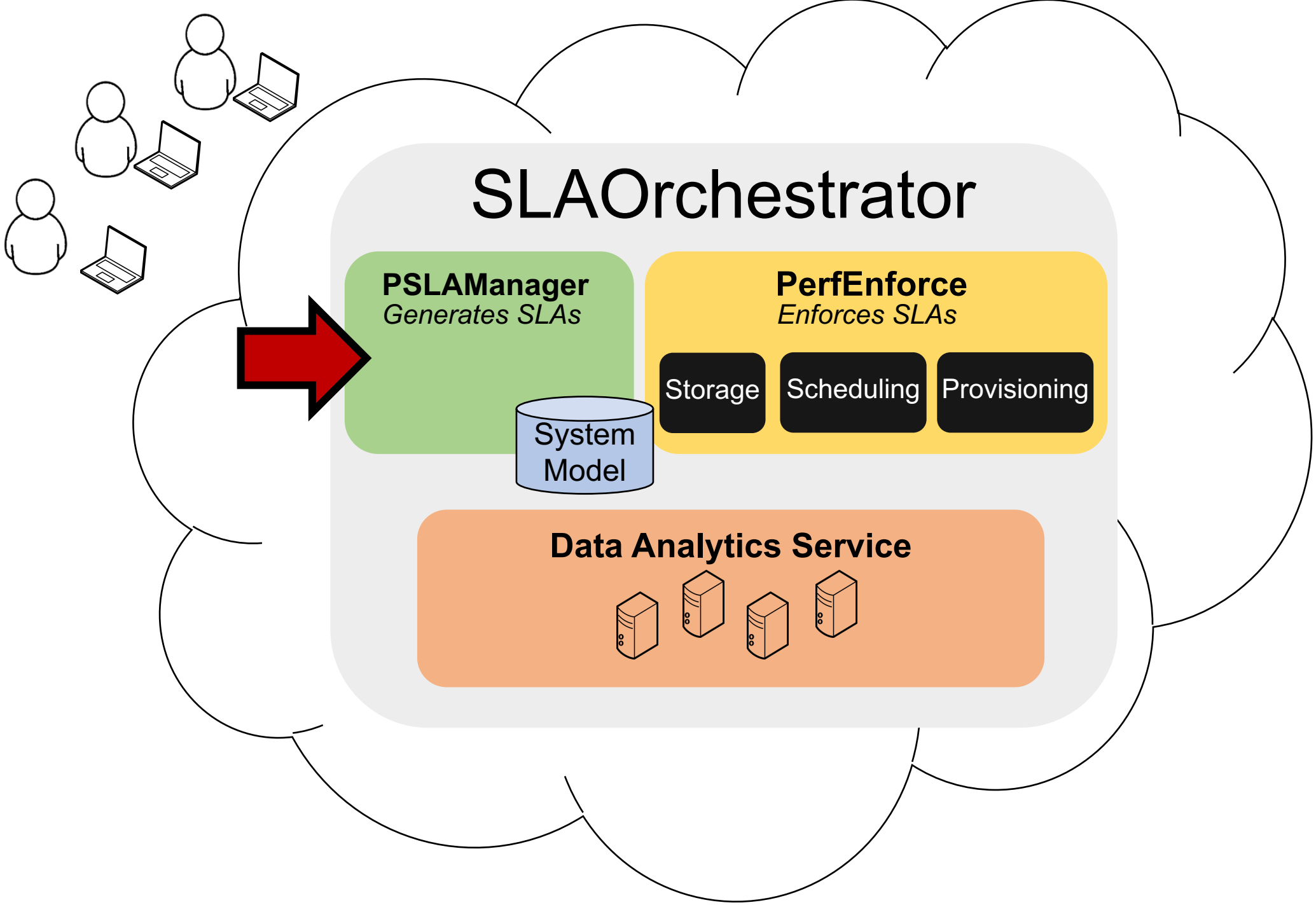
Performance SLAs in a Data Analytics System




SLAOrchestrator
Data Analytics Service


SLAOrchestrator
Data Analytics Service


SLAOrchestrator
PSLAManager
Generates SLAs
PerfEnforce
Enforces SLAs
Storage
Scheduling
Provisioning
System Model
Data Analytics Service


SLAOrchestrator
PSLAManager
Generates SLAs
PerfEnforce
Enforces SLAs
Storage
Scheduling
Provisioning
System Model
Data Analytics Service

Personalized Service Level Agreement (PSLAs)

Service tiers

Tier #1

Query Template	Runtime (seconds)
SELECT (9 ATTR.) FROM (PART) SELECT (9 ATTR.) FROM (CUSTOMER) SELECT (17 ATTR.) FROM (DATE) SELECT (60 ATTR.) FROM (5 TABLES) WHERE 0.1%	10
SELECT (17 ATTR.) FROM (LINEITEM) SELECT (9 ATTR.) FROM (2 TABLES) SELECT (3 ATTR.) FROM (5 TABLES) SELECT (60 ATTR.) FROM (5 TABLES) WHERE 10%	60
SELECT (60 ATTR.) FROM (5 TABLES)	300

Purchase @ $0.16/hour

Tier #2

Query Template	Runtime (seconds)
SELECT (27 ATTR.) FROM (5 TABLES) WHERE 10% SELECT (60 ATTR.) FROM (5 TABLES) WHERE 1%	10
SELECT (11 ATTR.) FROM (2 TABLES) SELECT (9 ATTR.) FROM (5 TABLES)	60

Purchase @ $0.24/hour

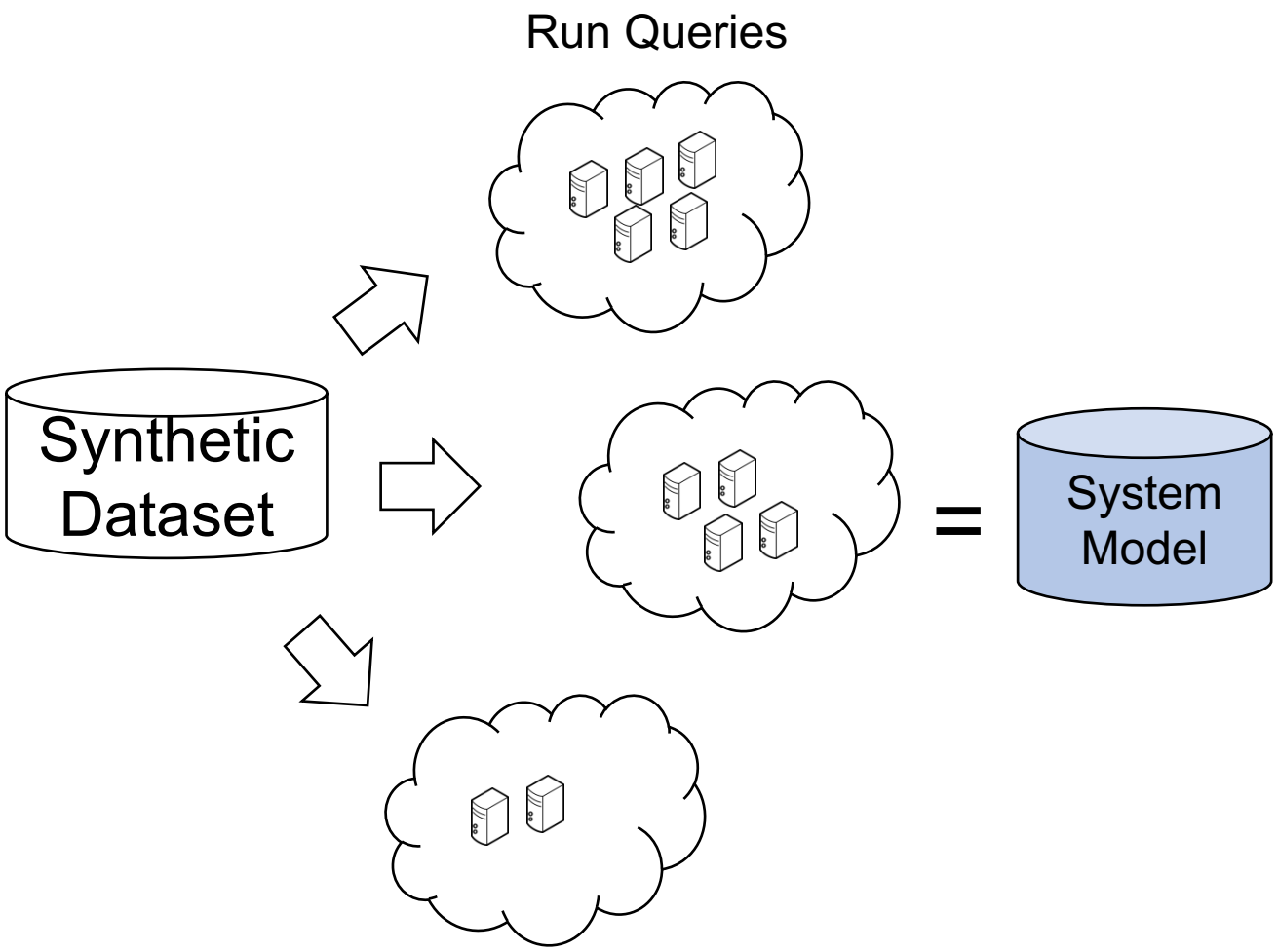
Expected **performance**

Fixed, hourly **price**

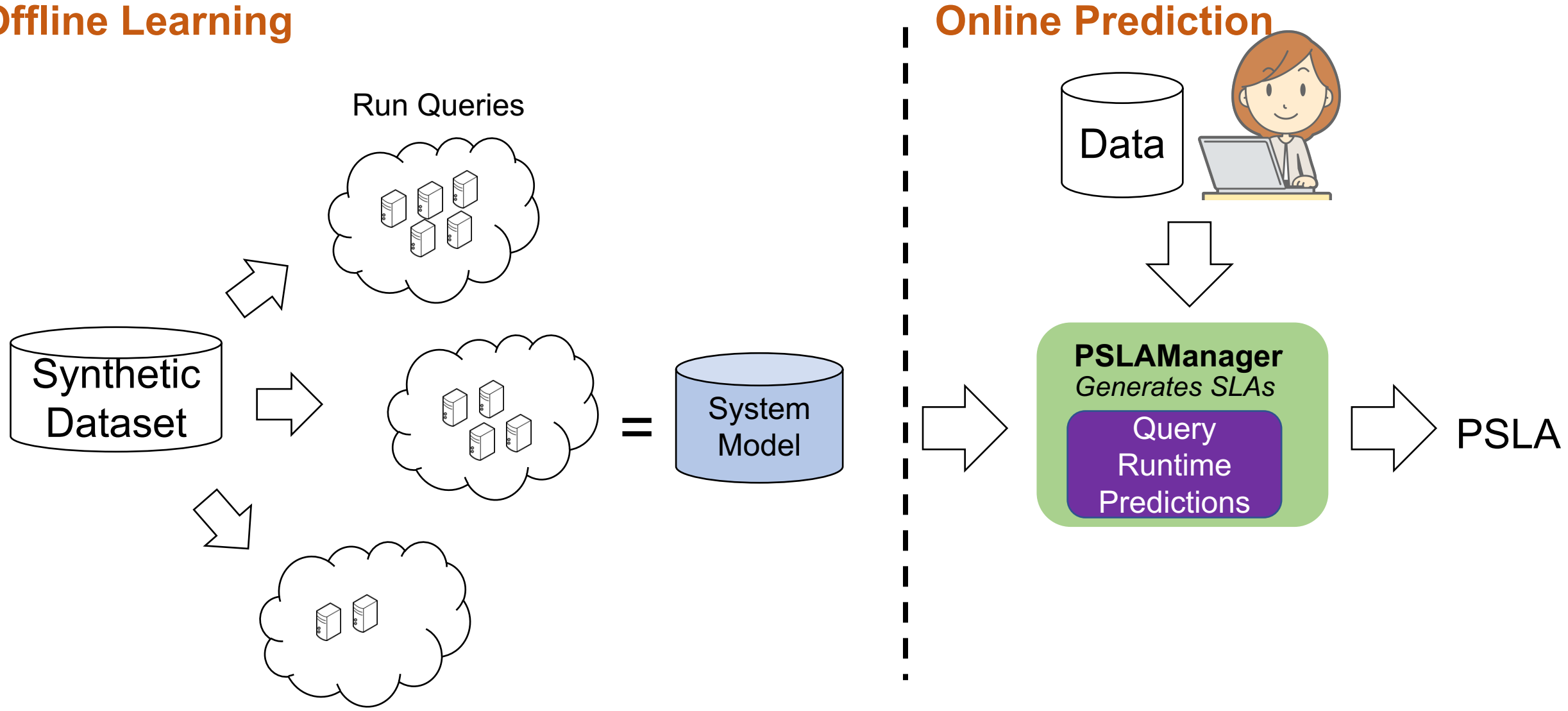
How to generate these SLAs?

PSLAManager: PSLA Generation

Offline Learning



PSLAManager: SLA Generation



PSLAManager Challenges

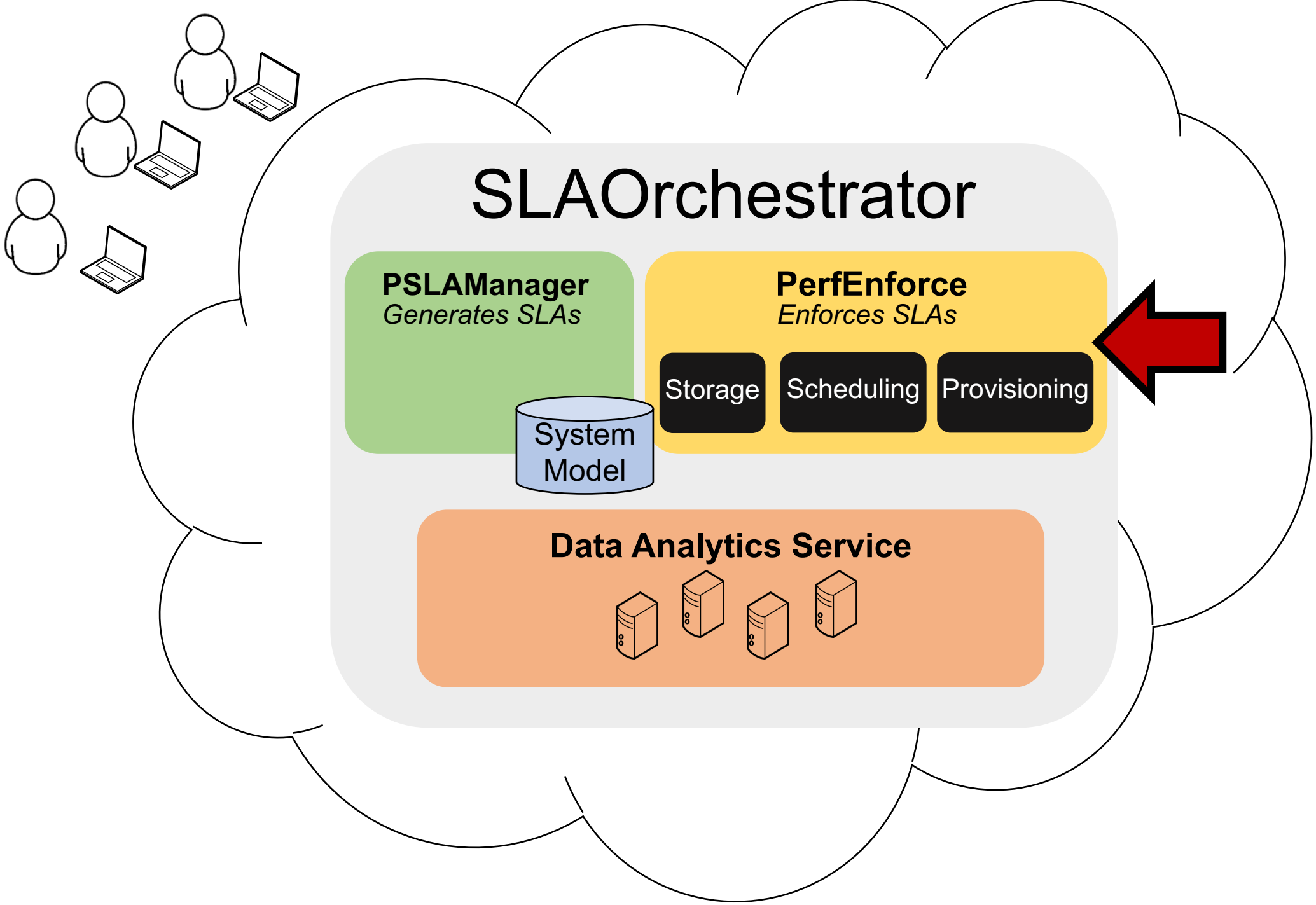
Predictions might be inaccurate

Tier #2

Query Template	Runtime (seconds)
SELECT (27 ATTR.) FROM (5 TABLES) WHERE 10% SELECT (60 ATTR.) FROM (5 TABLES) WHERE 1%	10
SELECT (11 ATTR.) FROM (2 TABLES) SELECT (9 ATTR.) FROM (5 TABLES)	60

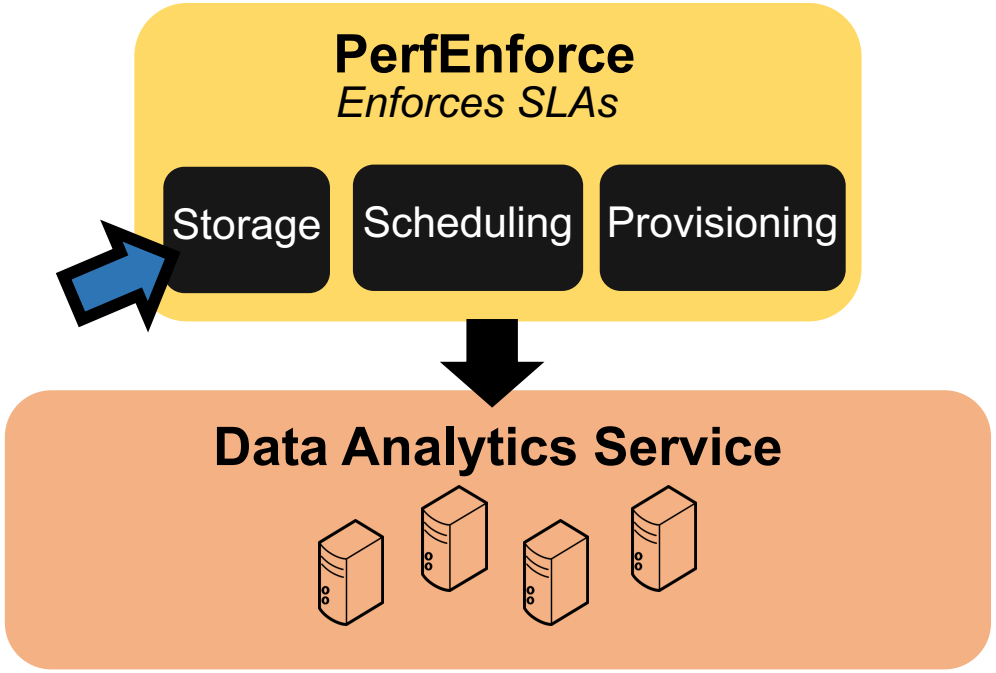
Purchase @ $0.24/hour

How to **scale** the system to enforce guarantee?


SLAOrchestrator
PSLAManager
Generates SLAs
PerfEnforce
Enforces SLAs
Storage
Scheduling
Provisioning
System Model
Data Analytics Service

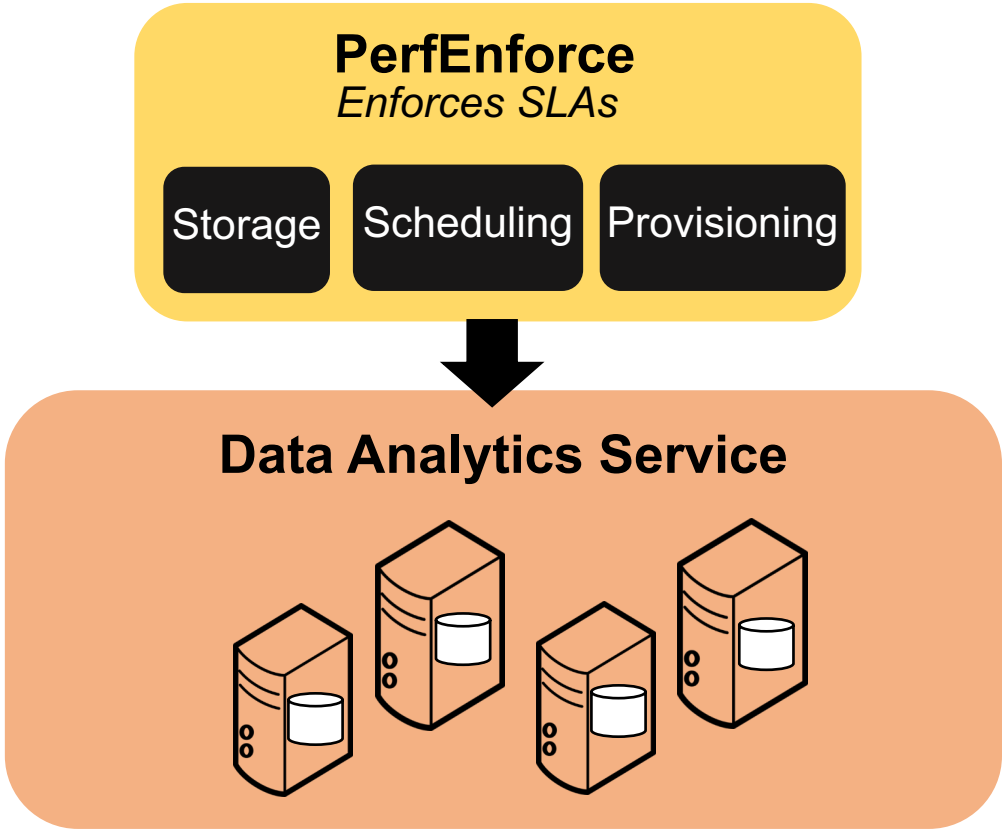
PerfEnforce Challenges

Input : Stream of (*tenant, query, SLA*)



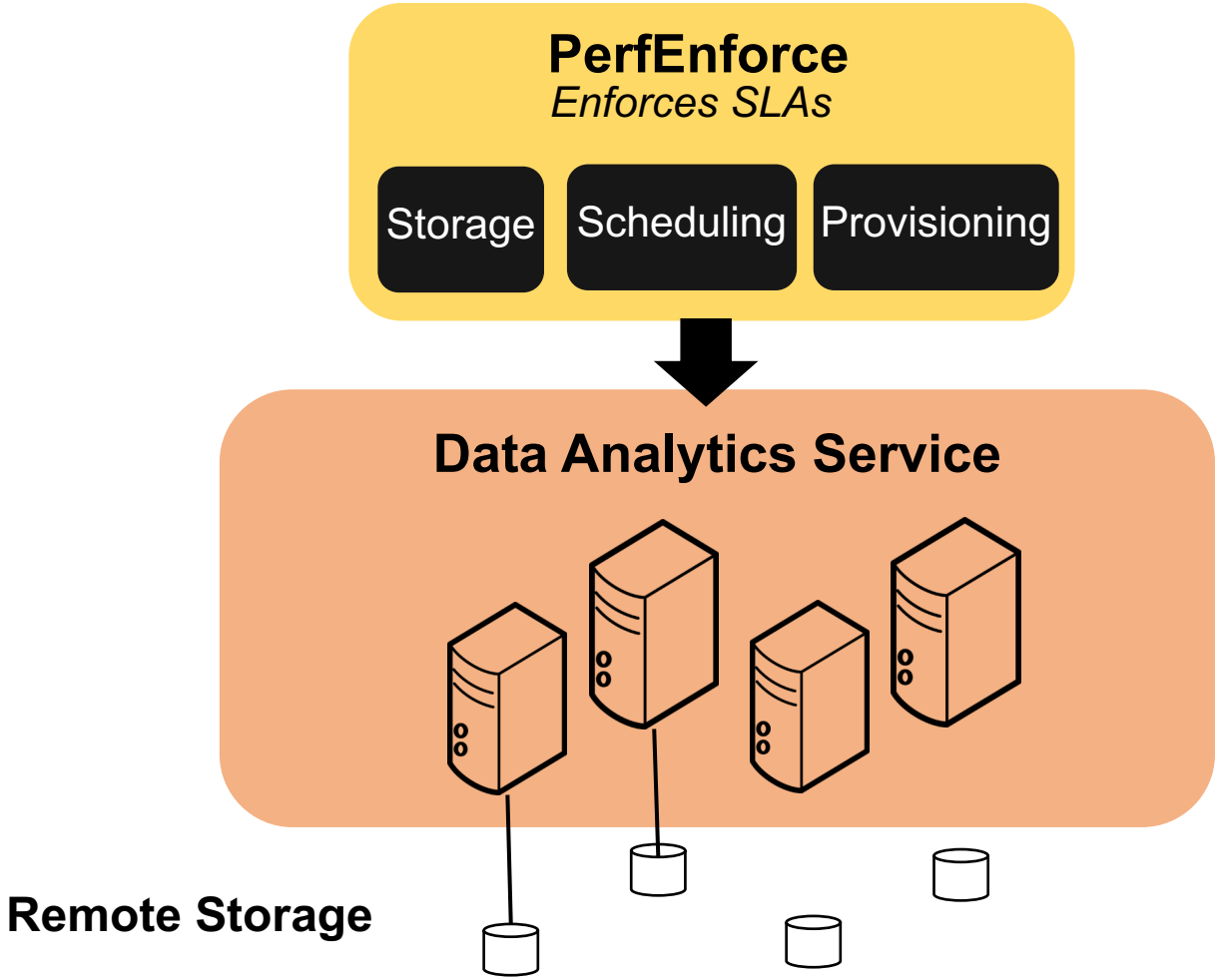
Storage

Input : Stream of (*tenant, query, SLA*)

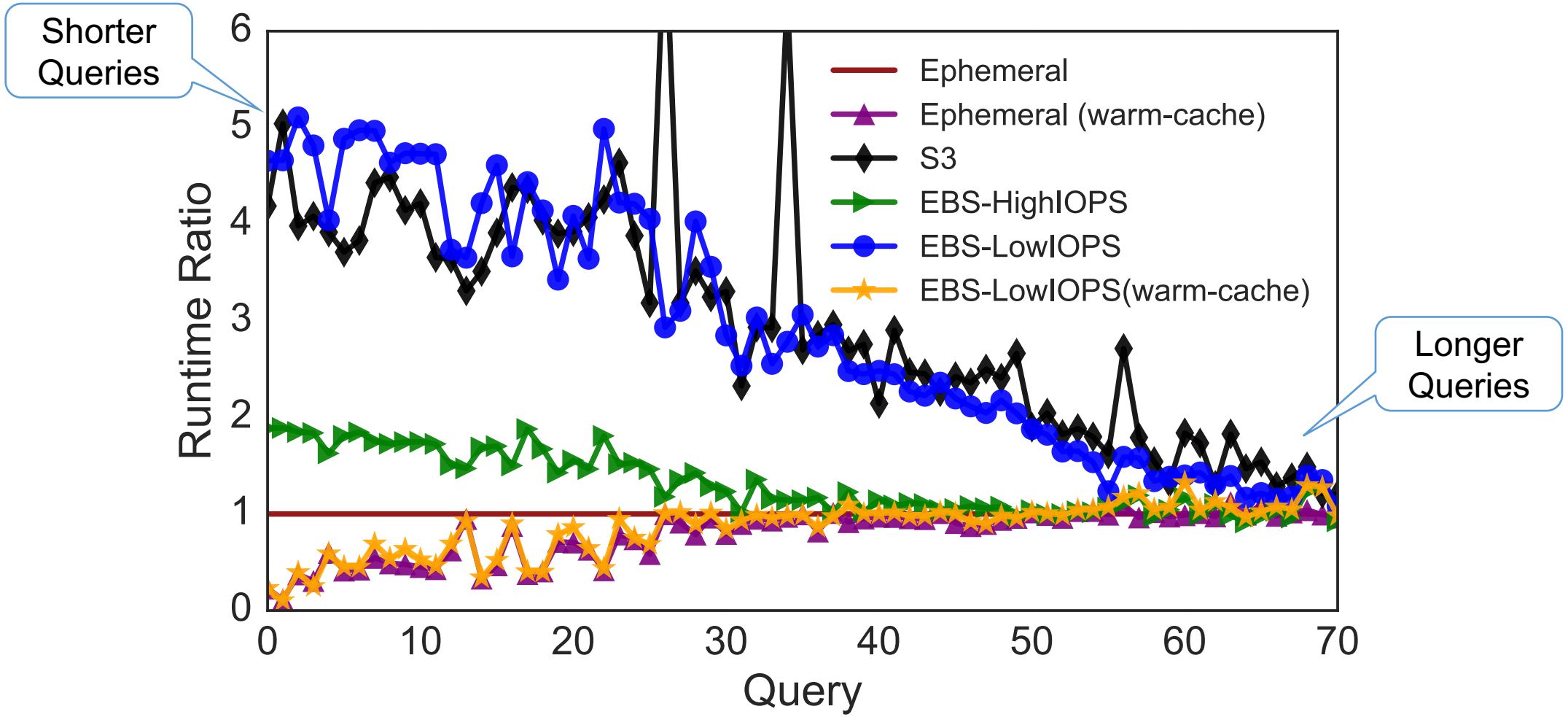


Storage

Input : Stream of (*tenant, query, SLA*)



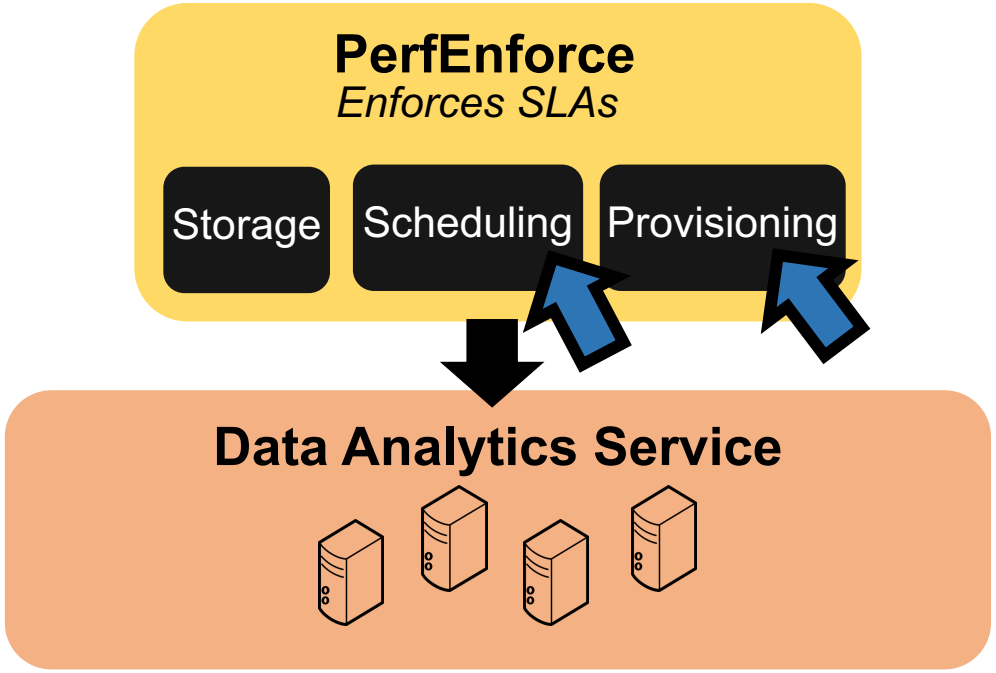
Performance for Networked Storage



32 workers - 100 random queries - 100GB data

PerfEnforce Challenges

Input : Stream of (*tenant, query, SLA*)

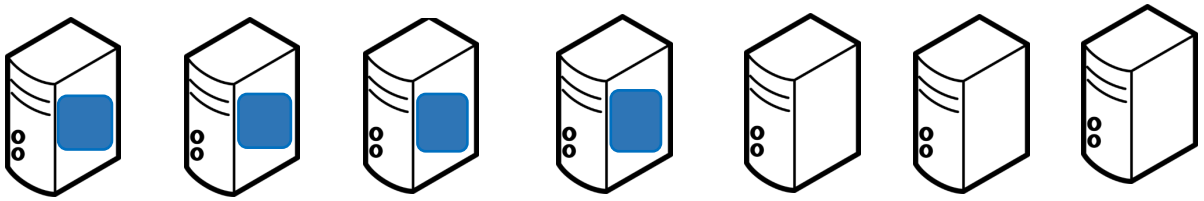


PerfEnforce Optimization Goal

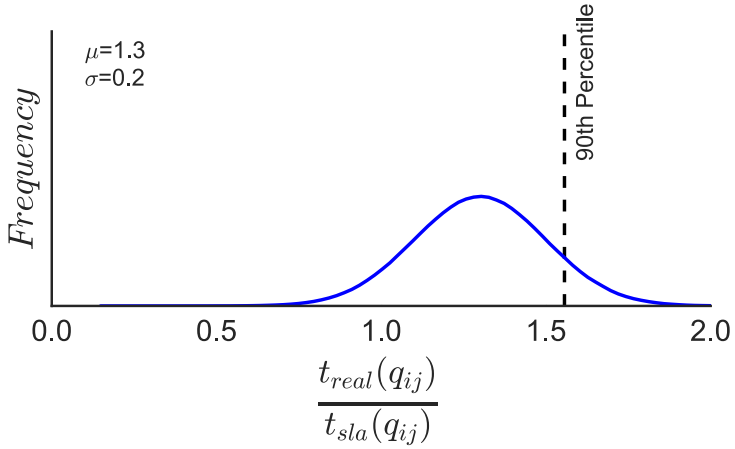
Goal: Minimize cost cluster + cost SLA violations

Scheduling: How many workers to a query?

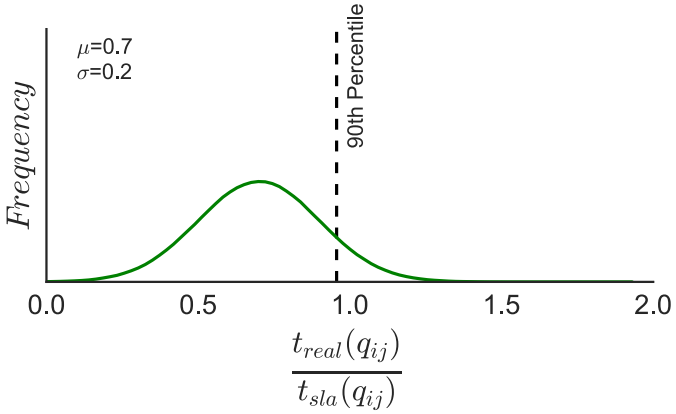
Query Performance Ratio = $\frac{t_{real}(q_{ij})}{t_{sla}(q_{ij})}$ ***= 1.0***



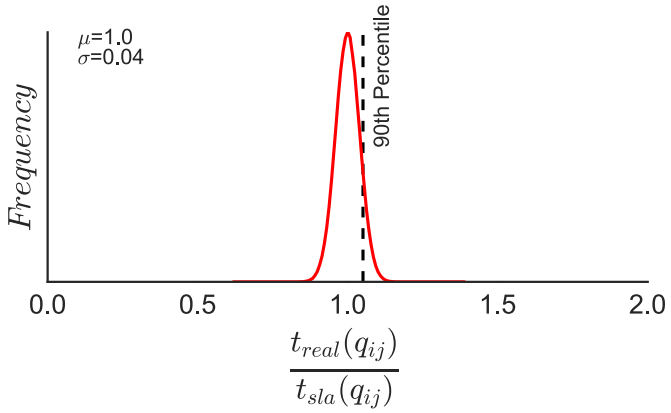
Bad: Violates SLAs



Bad: Wastes resources



Good

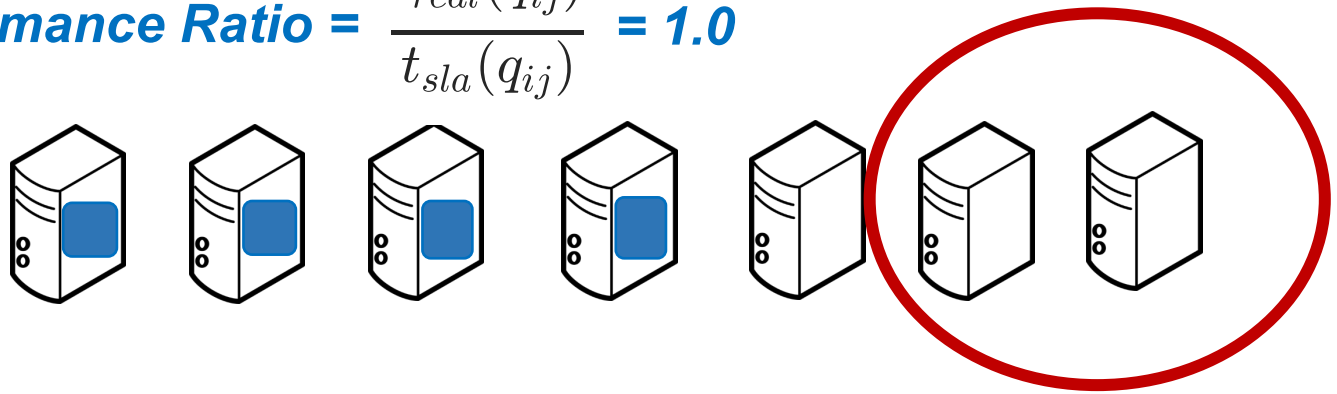


PerfEnforce Optimization Goal

Goal: Minimize cost cluster + cost SLA violations

Scheduling: How many workers to a query?

Query Performance Ratio = $\frac{t_{real}(q_{ij})}{t_{sla}(q_{ij})}$ *= 1.0*



Provisioning: When to turn off VMs?

Query Scheduling Algorithms

Reactive Approaches

Proportional-Integral Control

Multi-armed Bandit

Proactive Approaches

Contextual Multi-armed Bandit

Online Learning

Query Scheduling Algorithms

Reactive Approaches

- Proportional-Integral Control
- Multi-armed Bandit

Proactive Approaches

- Contextual Multi-armed Bandit
- Online Learning

Number of workers

Most recent error

$$u(t+1) = u(0) + \sum_{x=0}^{t} k_i e(x) + k_p e(t)$$

Initial number of workers

Errors accumulated over time

Query Scheduling Algorithms

Reactive Approaches

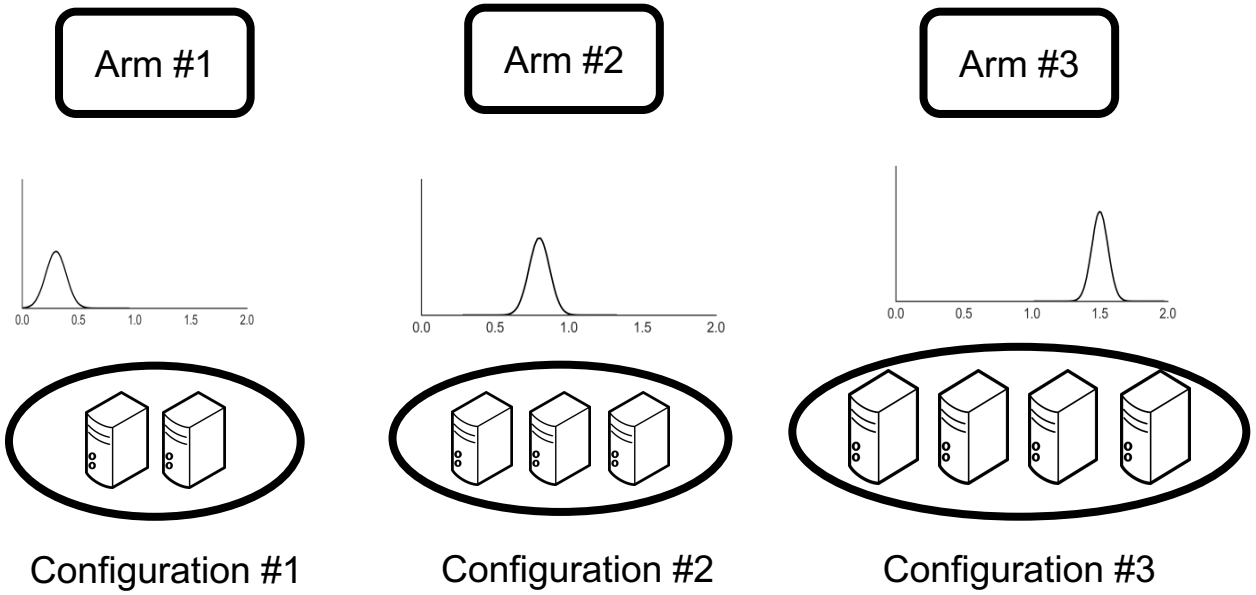
Proportional-Integral Control

Multi-armed Bandit

Proactive Approaches

Contextual Multi-armed Bandit

Online Learning



Query Scheduling Algorithms

Reactive Approaches

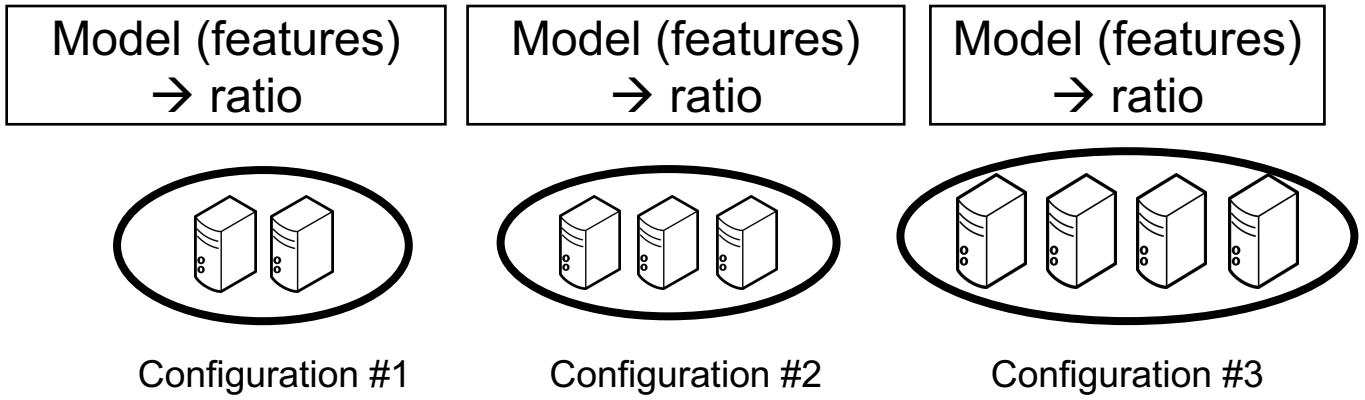
Proportional-Integral Control

Multi-armed Bandit

Proactive Approaches

Contextual Multi-armed Bandit

Online Learning



Query Scheduling Algorithms

Reactive Approaches

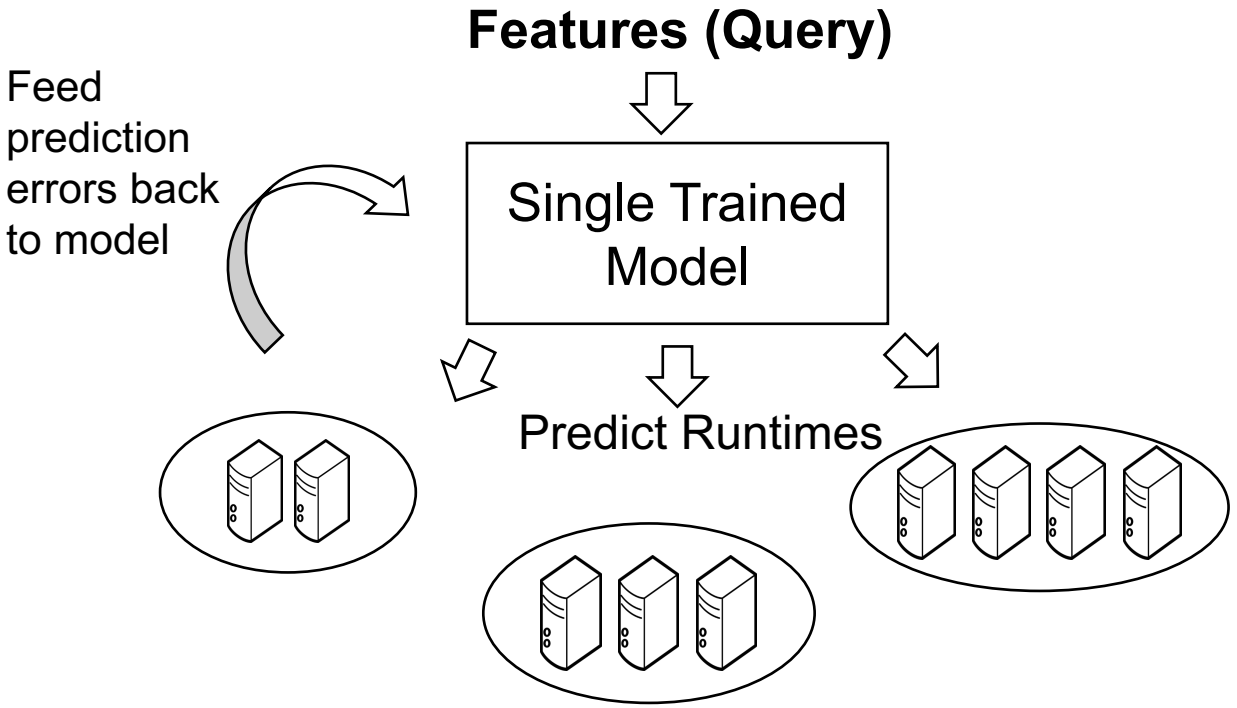
Proportional-Integral Control

Multi-armed Bandit

Proactive Approaches

Contextual Multi-armed Bandit

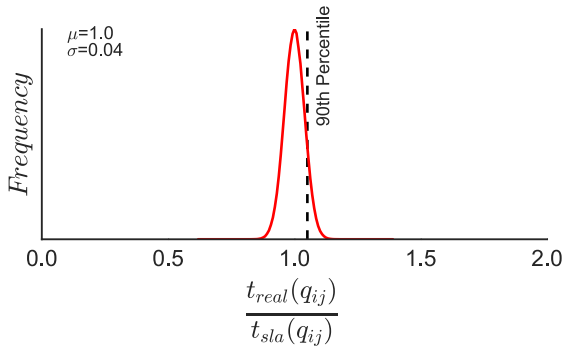
Online Learning



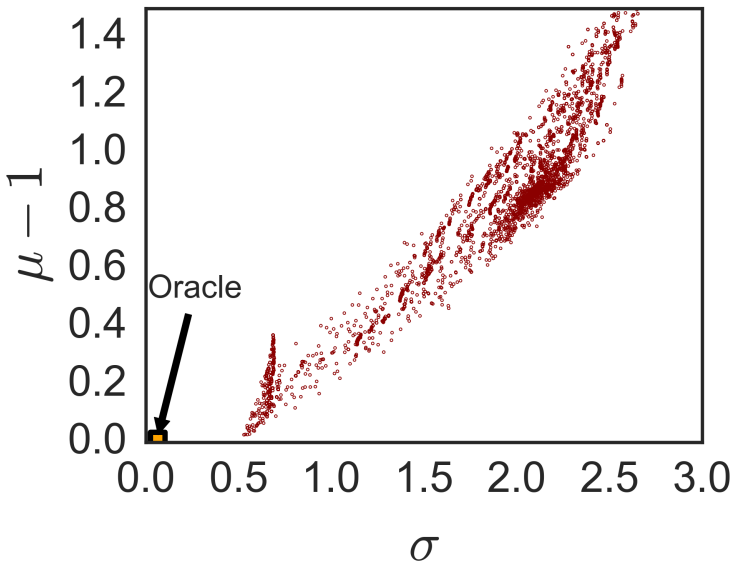
Query Scheduling Results

Amazon EC2 with 4, 8, 12, 16, 20, 24, 28, or 32 VMs – 100 GB –TPC-H Star Schema Benchmark
Each point: One set of configuration parameters and 10 query sessions
SLA generated with PSLAManager

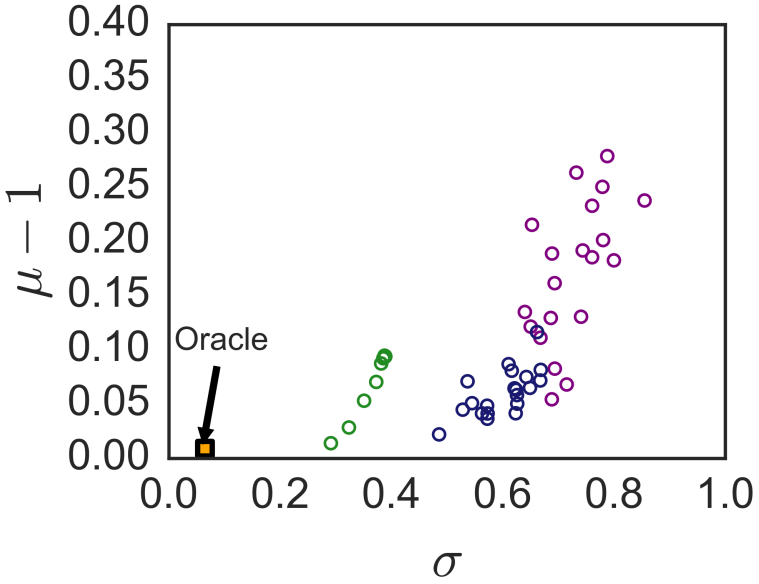
Good Distribution



PI-Control



MAB, CMAB, Online Learning

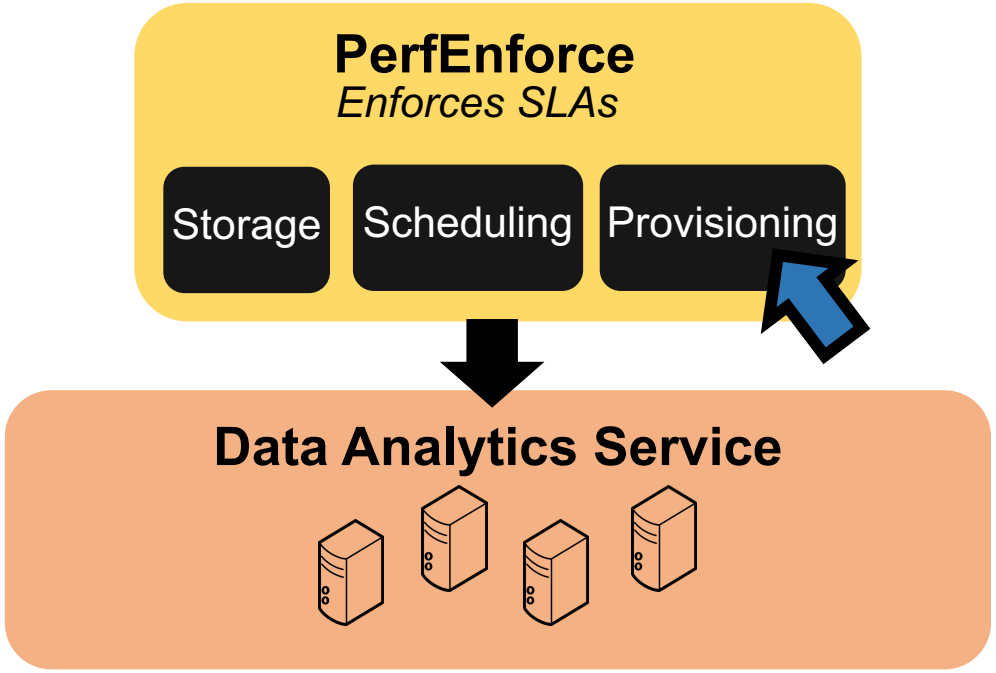


PI-Control MAB CMAB Online Learning

Result: Online learning yields tightest distributions

PerfEnforce Challenges

Input : Stream of (*tenant, query, SLA*)

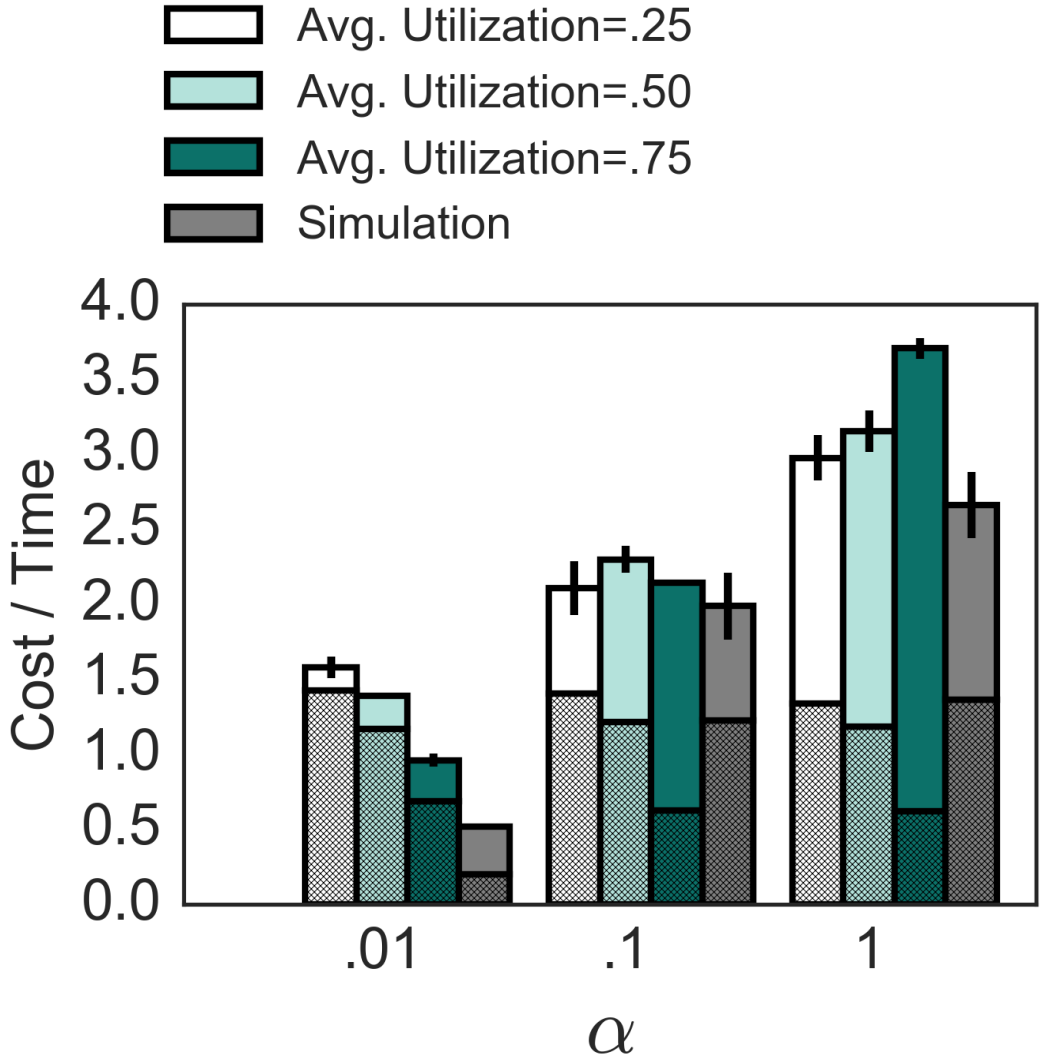


PerfEnforce Resource Provisioning

- Adding and removing VMs takes time
- **Two algorithms to monitor the system**
  - **Resource Utilization**
    - Add/remove VMs to maintain utilization close to set threshold T
  - **Simulation**:
    - Learn past tenant behavior and resize cluster assuming same behavior in next time window

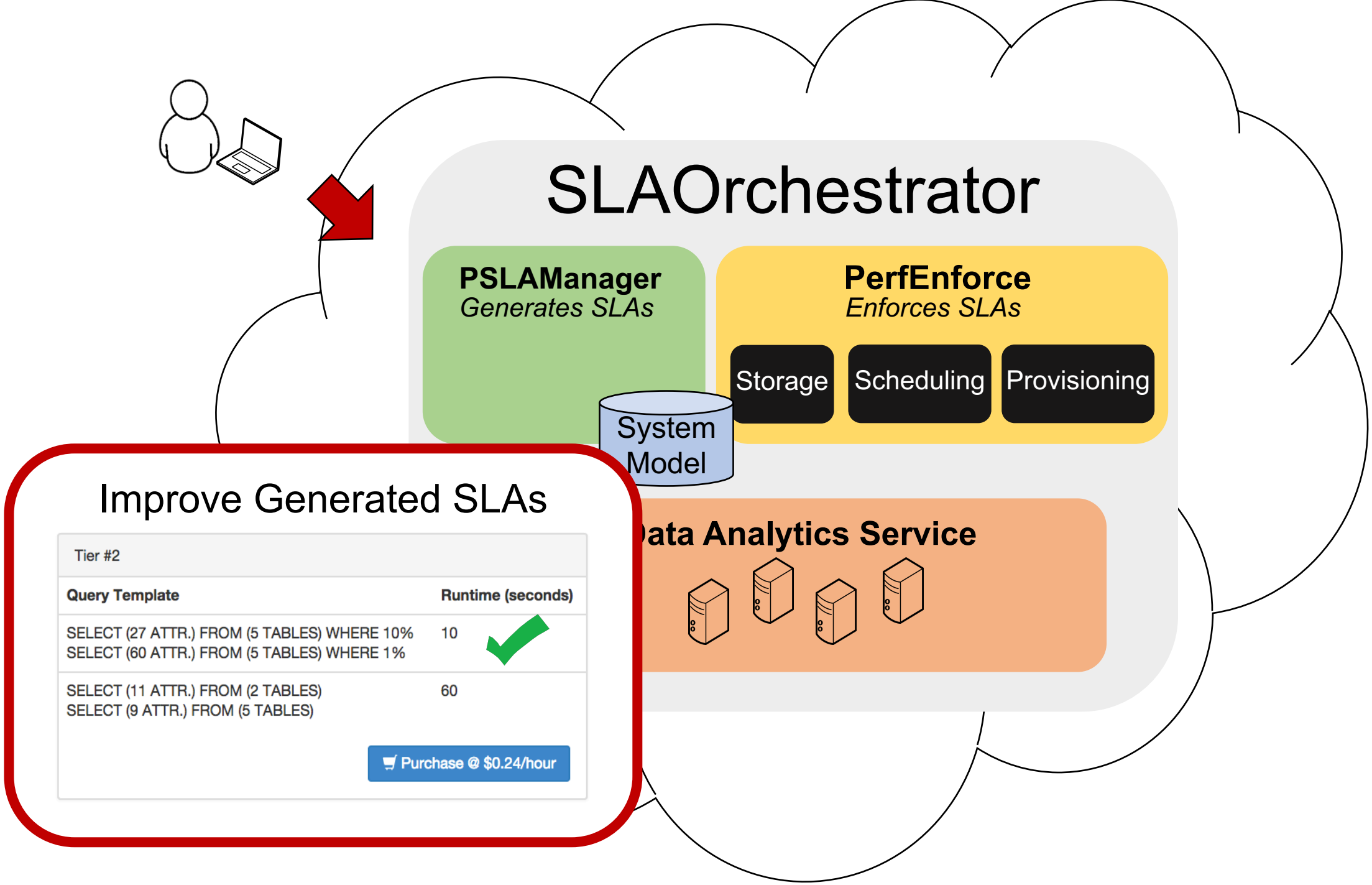
Resource Provisioning Results

100 virtual machines with 10 initial tenants

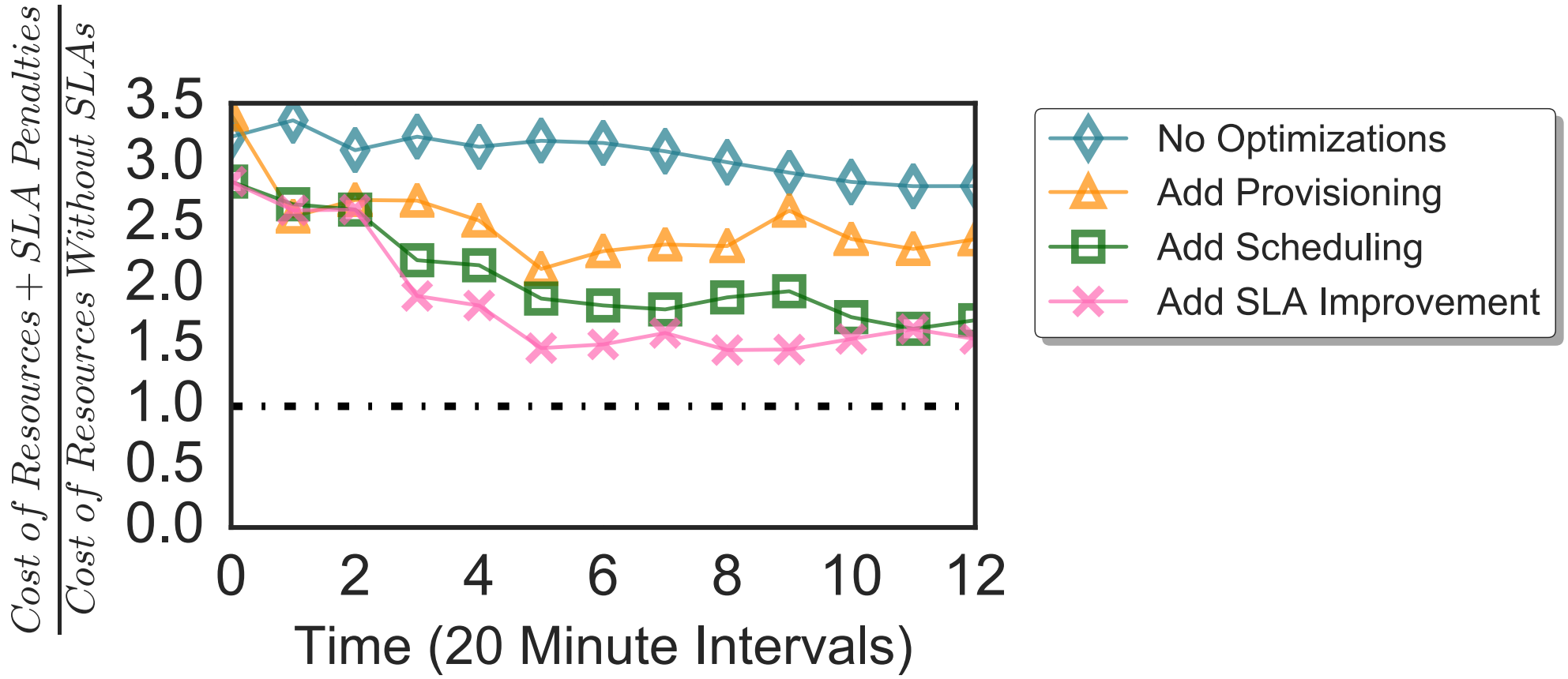


Relative cost of SLA penalties vs VMs

Simulation-based provisioning yields lower costs in all settings


SLAOrchestrator
PSLAManager
Generates SLAs
PerfEnforce
Enforces SLAs
Storage
Scheduling
Provisioning
System Model
ata Analytics Service
Improve Generated SLAs
Tier #2
Query Template
Runtime (seconds)
SELECT (27 ATTR.) FROM (5 TABLES) WHERE 10%
SELECT (60 ATTR.) FROM (5 TABLES) WHERE 1%
10
SELECT (11 ATTR.) FROM (2 TABLES)
SELECT (9 ATTR.) FROM (5 TABLES)
60
Purchase @ $0.24/hour

SLAOrchestrator Optimizations



Conclusion

- SLAOrchestrator reduces the cost of Performance-based SLAs
  - **PSLAManager** generates PSLAs
  - **PerfEnforce** enforces runtimes through scaling

- Source Code Available
  - ***PSLAManager* (SLA Generator)**
    - ***https://github.com/uwdb/PSLAManager***

  - ***PerfEnforce* (Query Scheduler) Prototype available on** Myria
    - ***https://github.com/uwescience/myria***