Unlocking unallocated cloud capacity for long, uninterruptible workloads

Anup Agarwal
anupa@cmu.edu

Shadi Noghabi

Inigo Goiri

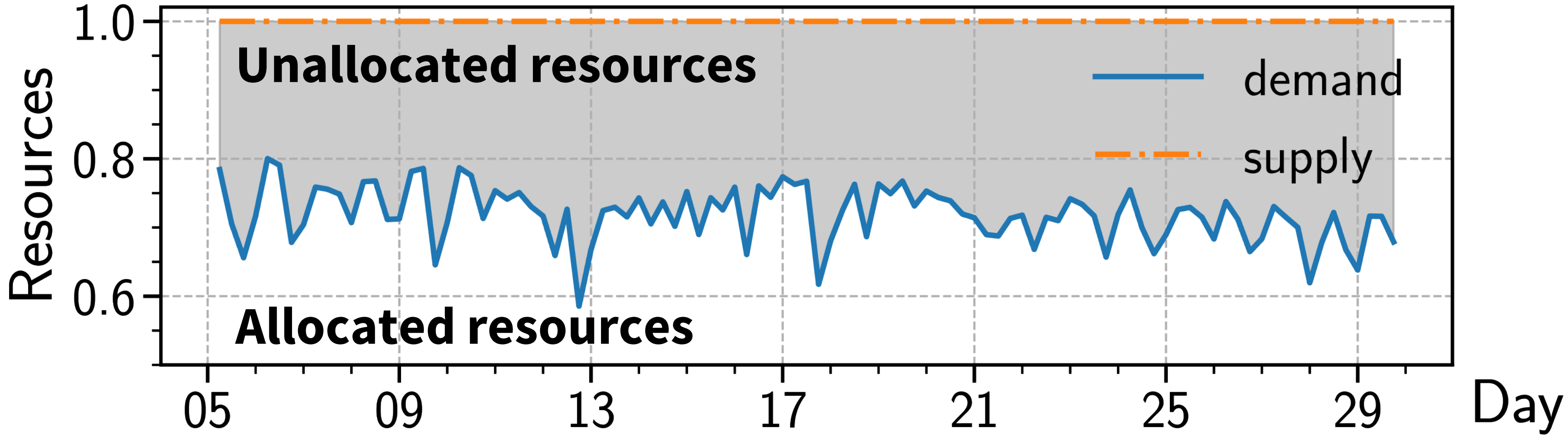
Srinivasan
Seshan

Anirudh
Badam

Carnegie Mellon

Microsoft

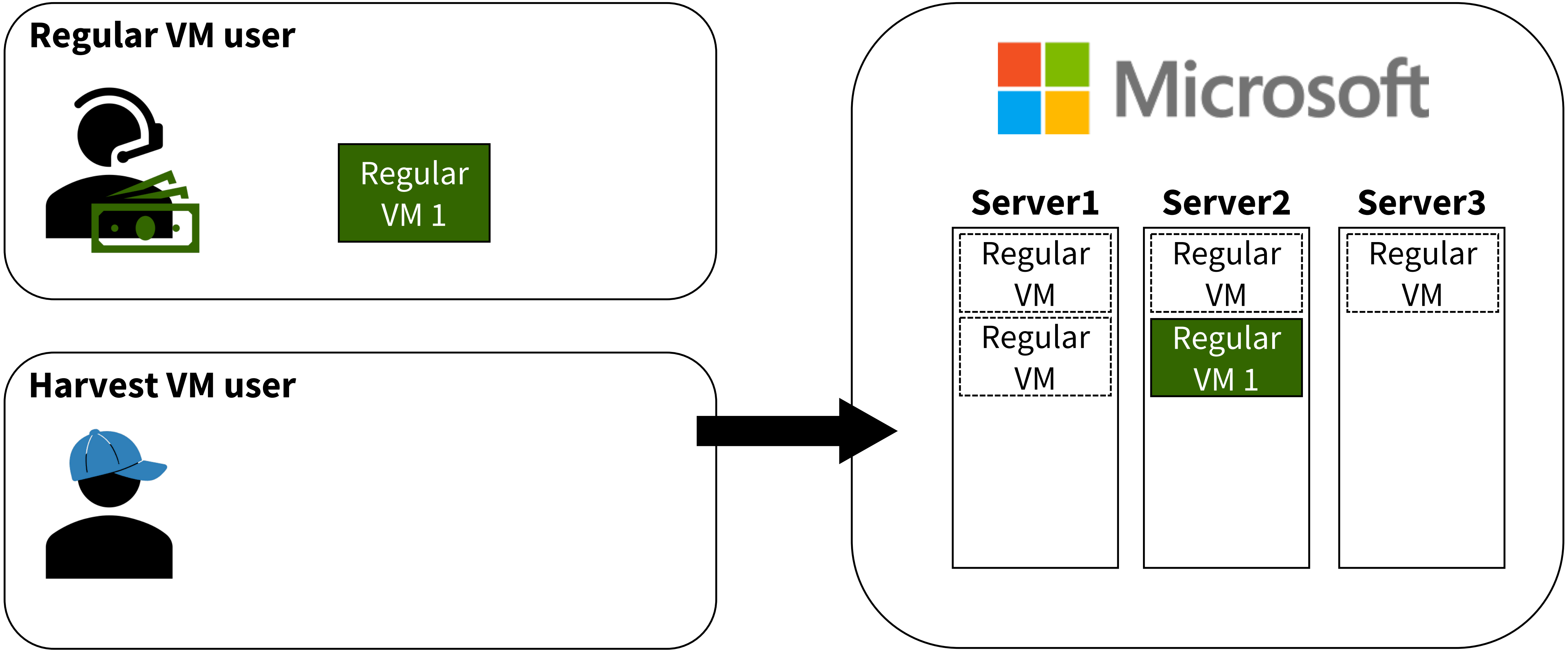
Cloud has unallocated (or unsold) capacity



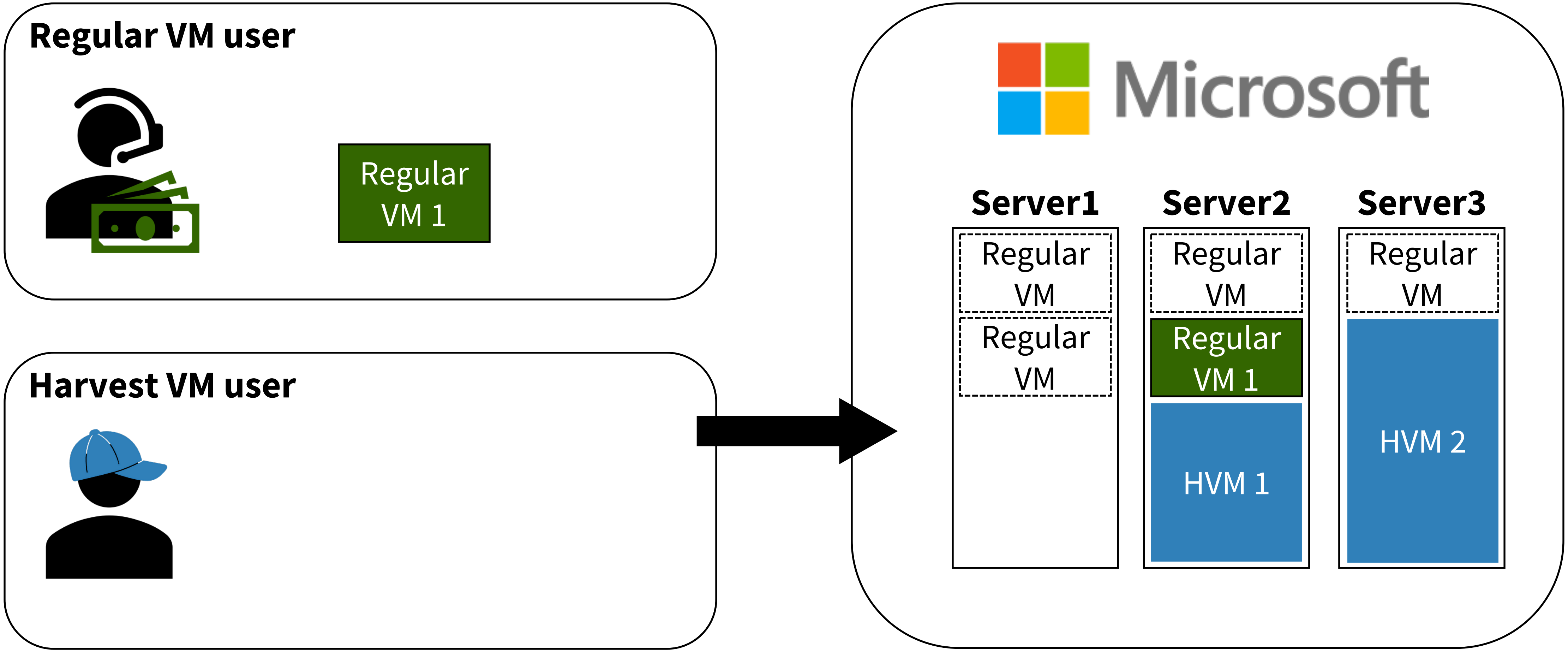
- Illusion of elasticity.
- Provisioned for peak demand
- Variations in demand, VM sizes (fragmentation).

- 20-30% hardware unallocated.
- Spot VMs, oversubscription, serverless…

Harvest Virtual Machines (HVMs) grow/shrink to expose unallocated cloud capacity



Harvest Virtual Machines (HVMs) grow/shrink to expose unallocated cloud capacity



Regular VM user

Regular VM 1

Harvest VM user

Microsoft

Server1

Regular VM

Regular VM

Server2

Regular VM

Regular VM 1

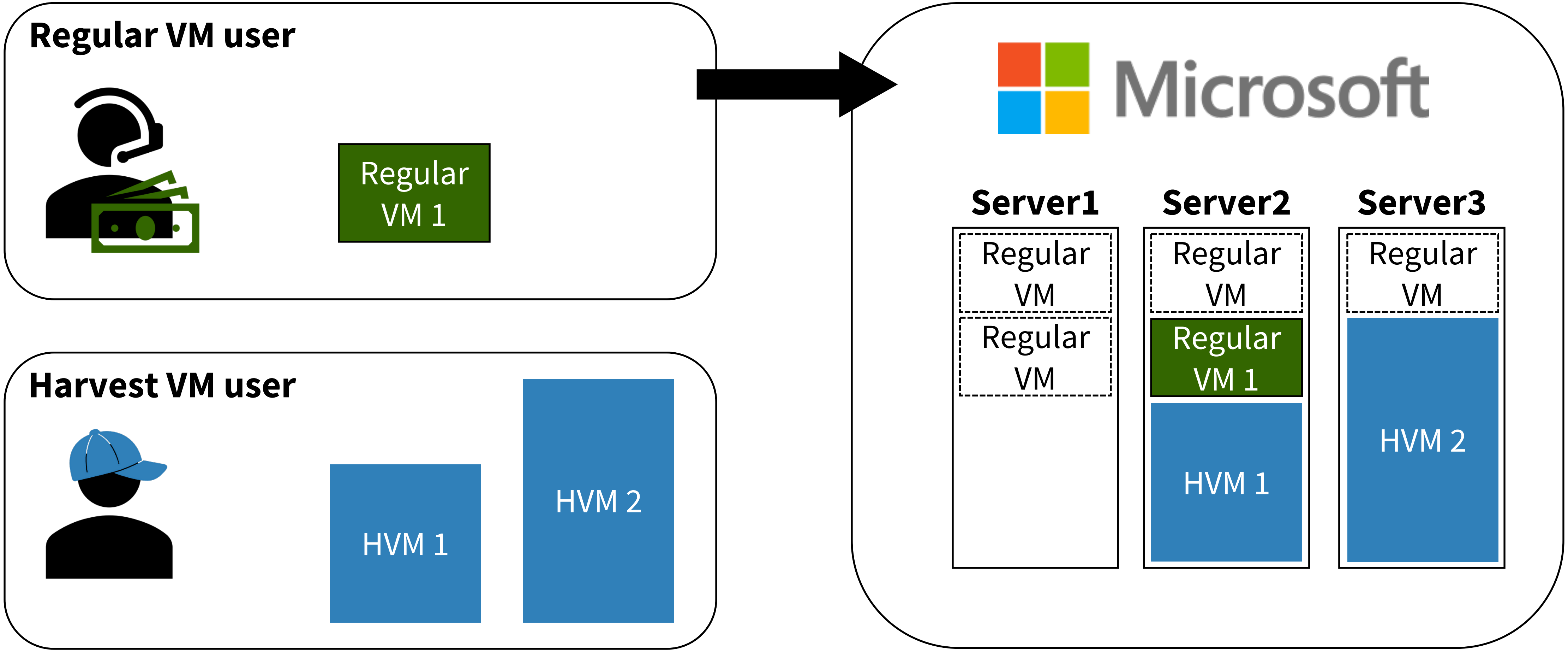
HVM 1

Server3

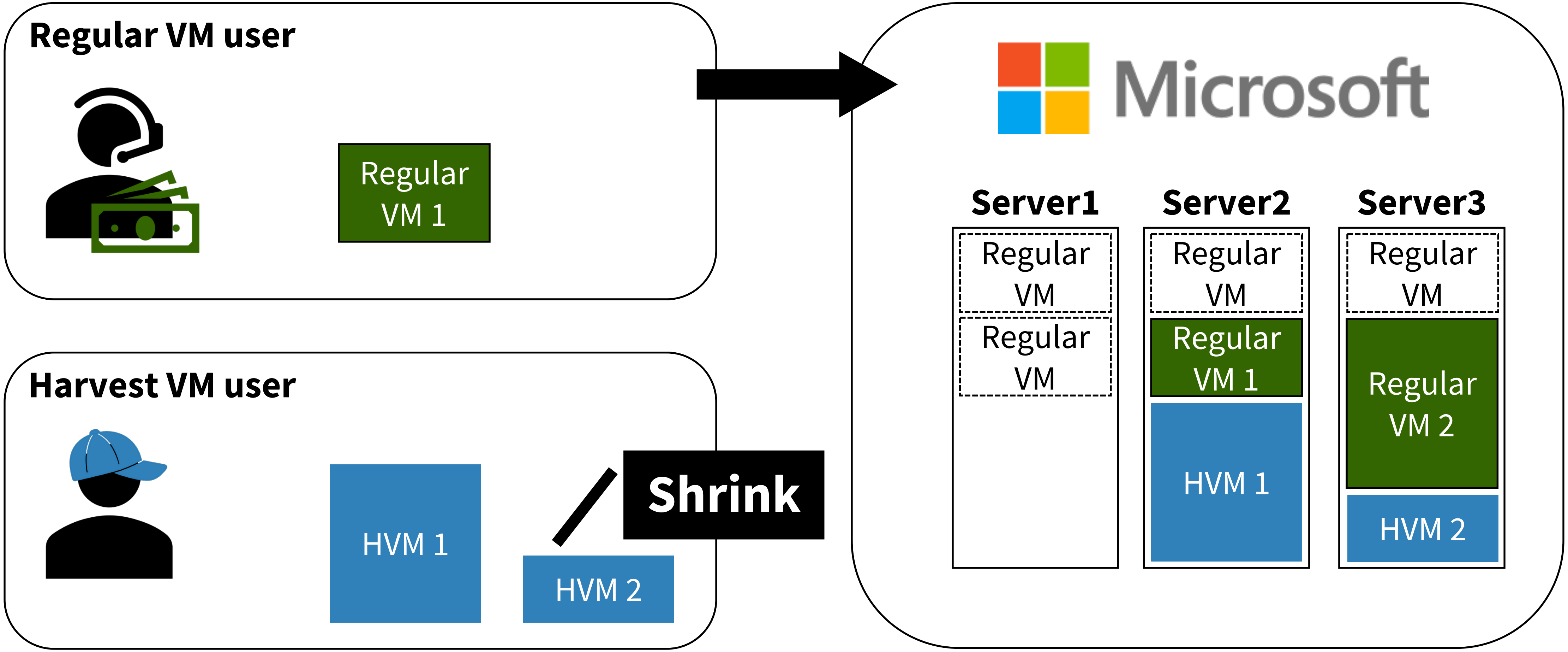
Regular VM

HVM 2

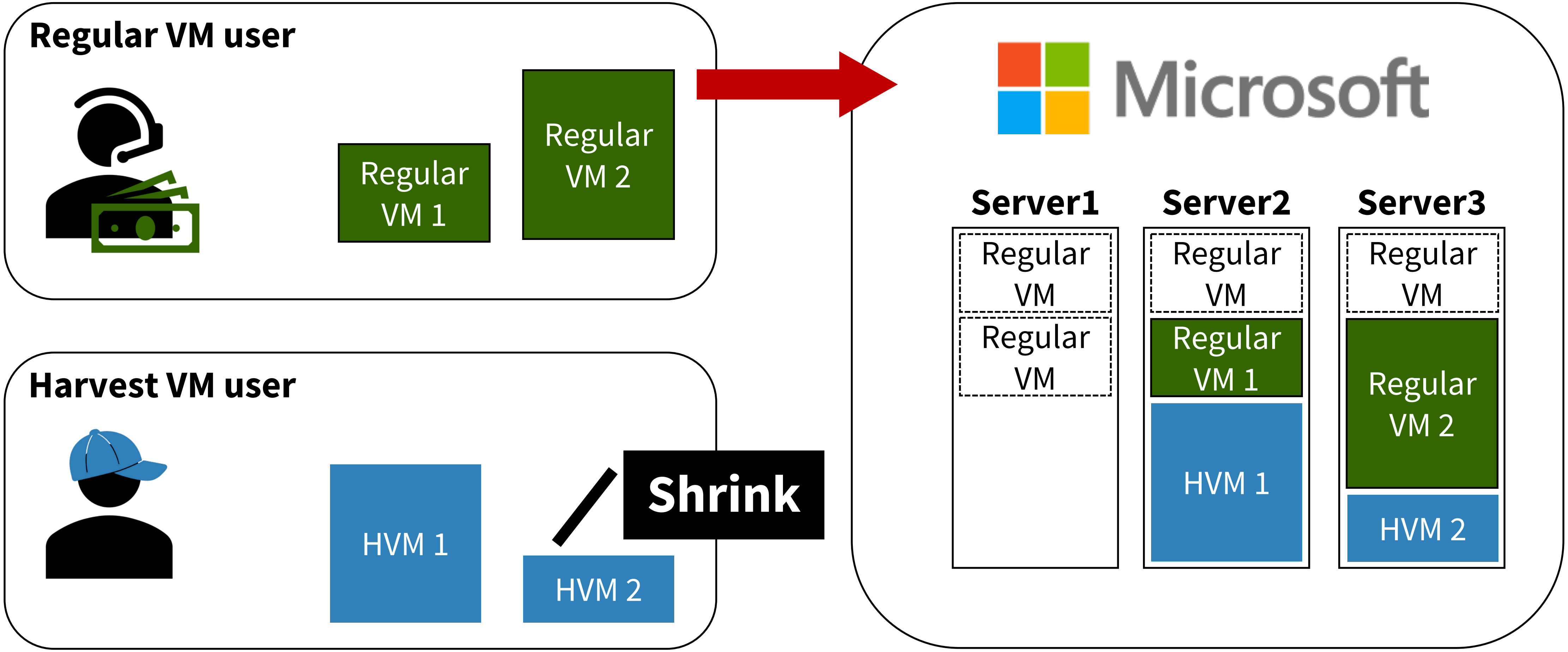
Harvest Virtual Machines (HVMs) grow/shrink to expose unallocated cloud capacity



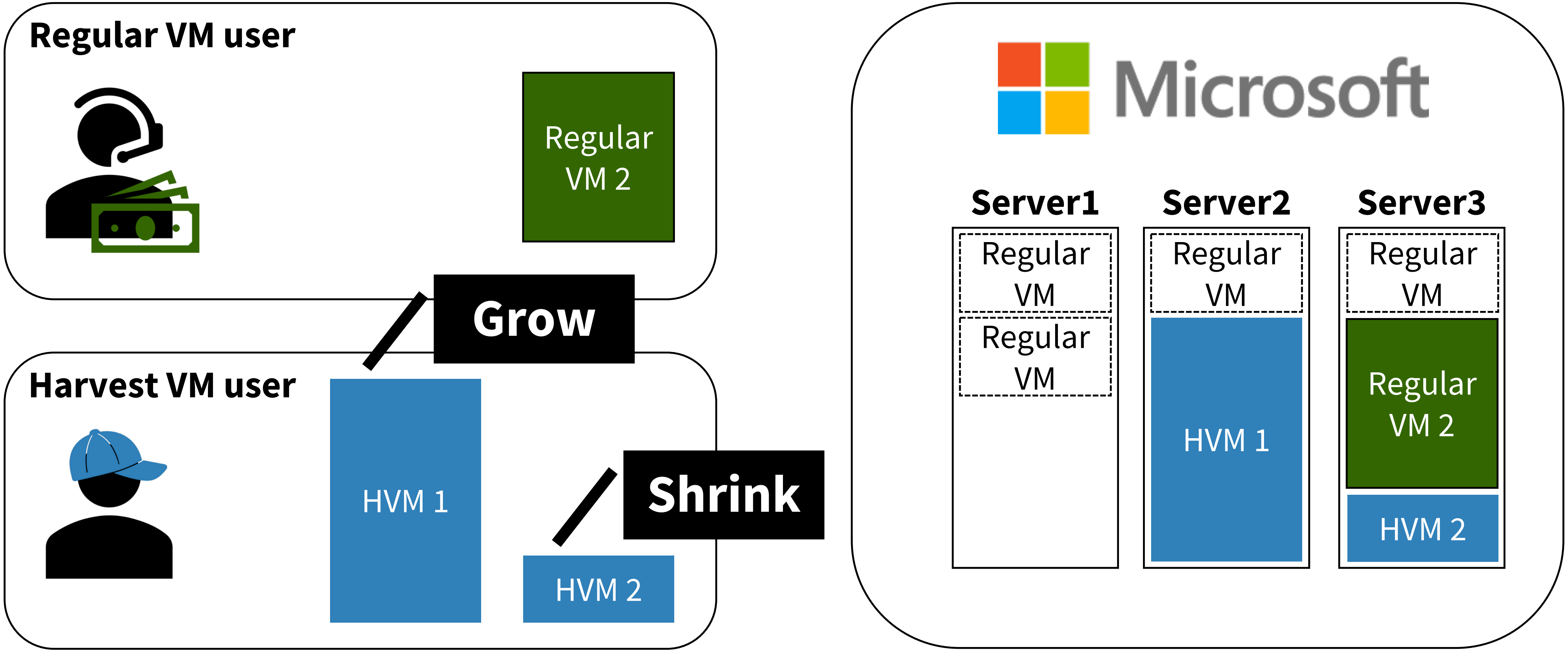
Harvest Virtual Machines (HVMs) grow/shrink to expose unallocated cloud capacity



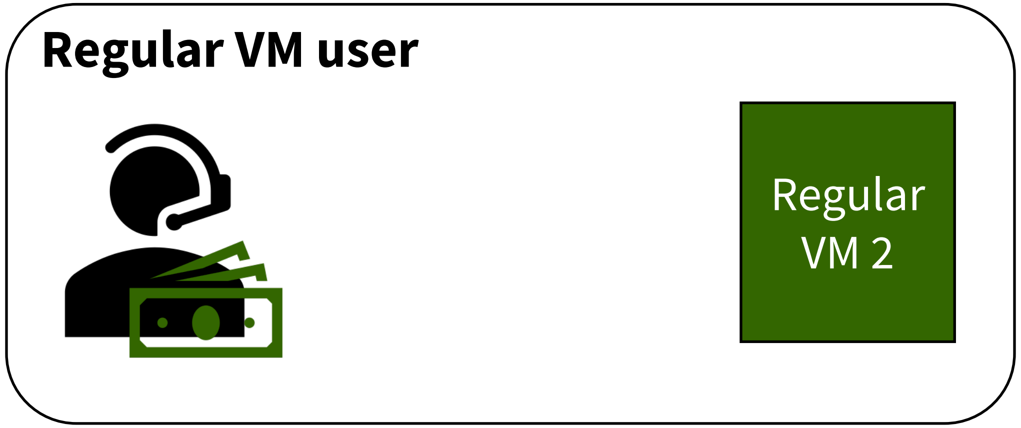
Harvest Virtual Machines (HVMs) grow/shrink to expose unallocated cloud capacity



Harvest Virtual Machines (HVMs) grow/shrink to expose unallocated cloud capacity

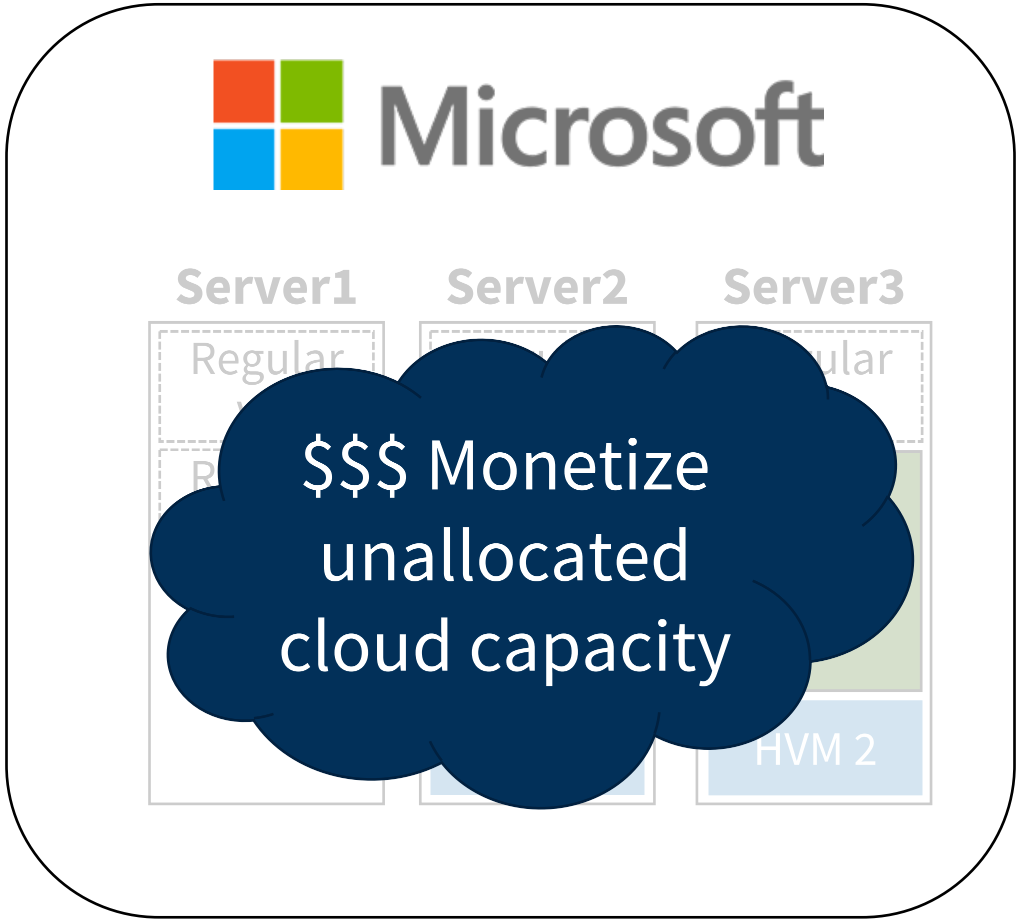


Harvest Virtual Machines (HVMs) grow/shrink to expose unallocated cloud capacity



Harvest VM user
2

Bigger VMs at lower prices



$$$ Monetize unallocated cloud capacity

HVMs overcome capacity and cost bottlenecks of large-scale scientific workloads

- Weather, geospatial simulations
- Genome analysis
- Important to **society** and **cloud providers**. Billions of dollars market.

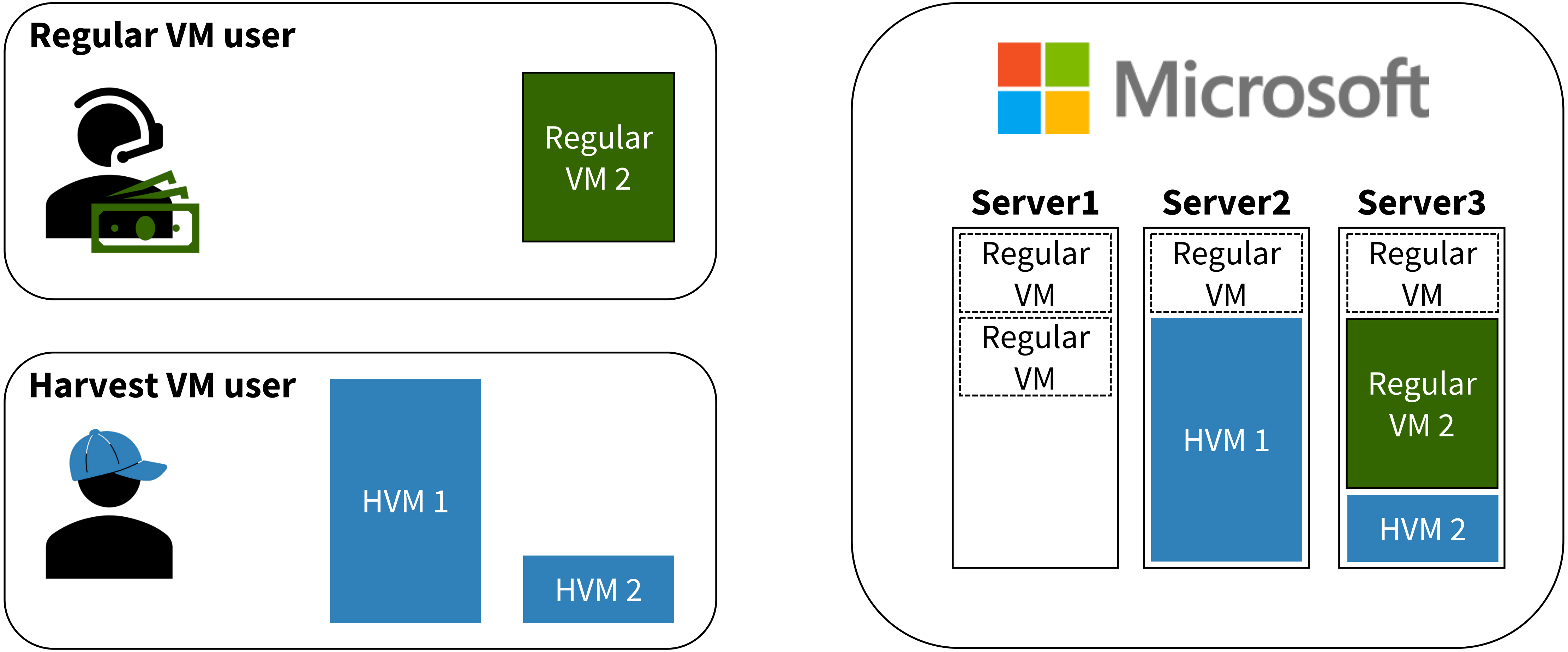
Challenge

Uninterruptible =

Hard to checkpoint/migrate

- Rely on domain specific libraries
- Large working sets & side-effects

Customer’s challenges in using HVMs



Regular VM user

Regular VM 2

Harvest VM user

HVM 1

HVM 2

Microsoft

Server1

Regular VM

Regular VM

Server2

Regular VM

HVM 1

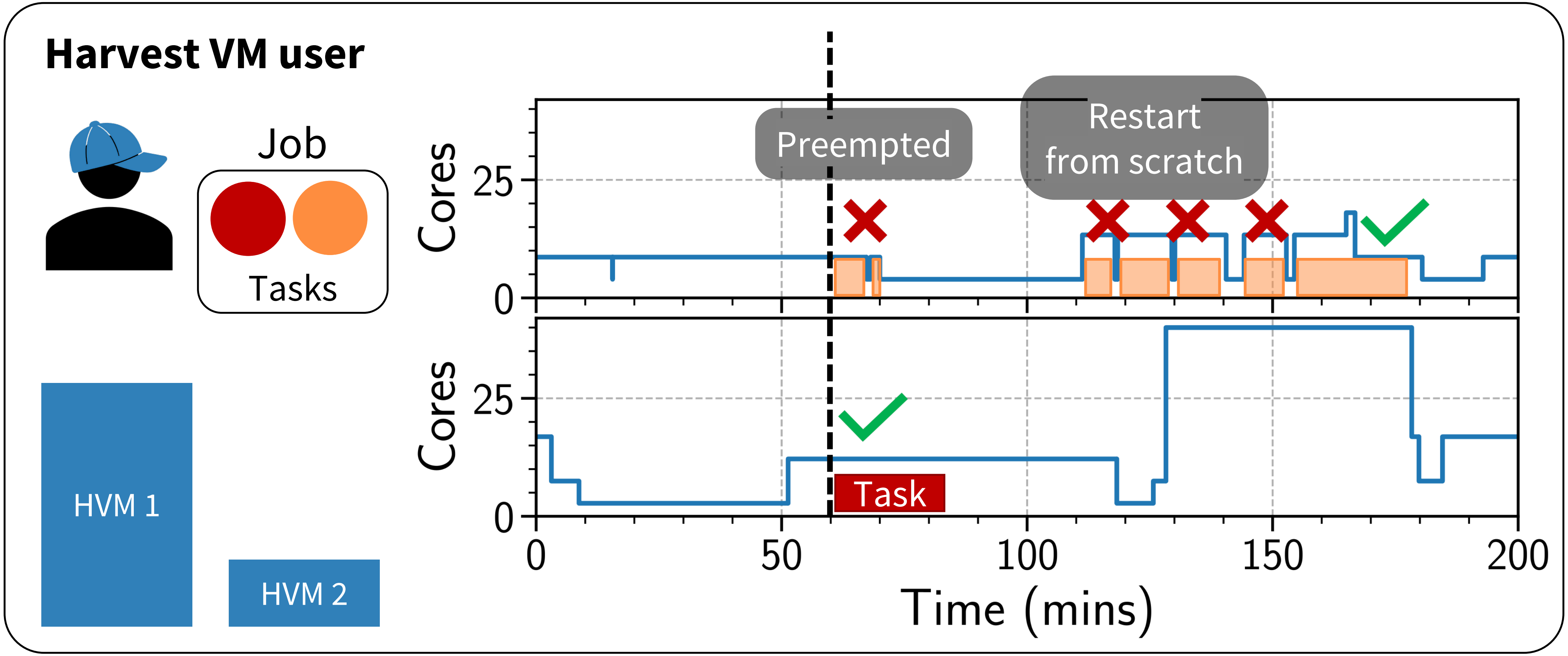
Server3

Regular VM

Regular VM 2

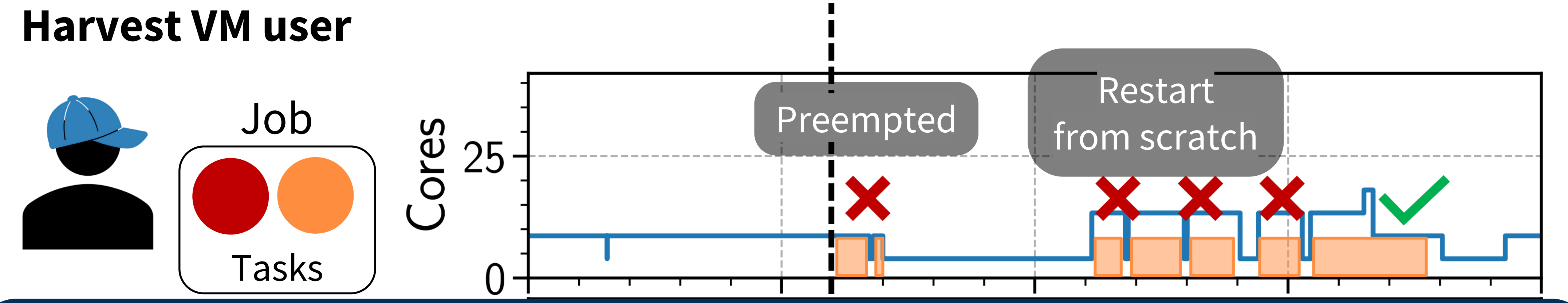
HVM 2

Customer's challenges in using HVMs



HVM resource variations slow down long, uninterruptible workloads by 1.5x

Harvest VM user



Existing techniques are insufficient

- Checkpointing, migration, replication → Prohibitive, Impractical
- Spot VM specific → Do not generalize
- Changes to parallelism changes outputs [Gesall, SIGMOD 17] → Unacceptable

How can customers best use HVMs for long, uninterruptible workloads?

SlackSched Roadmap

1. Characterization to understand the setting

- #1: Harvest VMs
- #2: Workloads

2. Opportunities to use HVMs more efficiently

- #1: Scheduling
- #2: Resource Acquisition

Characterization: Understand interaction between workloads & HVM variations

Harvest VMs

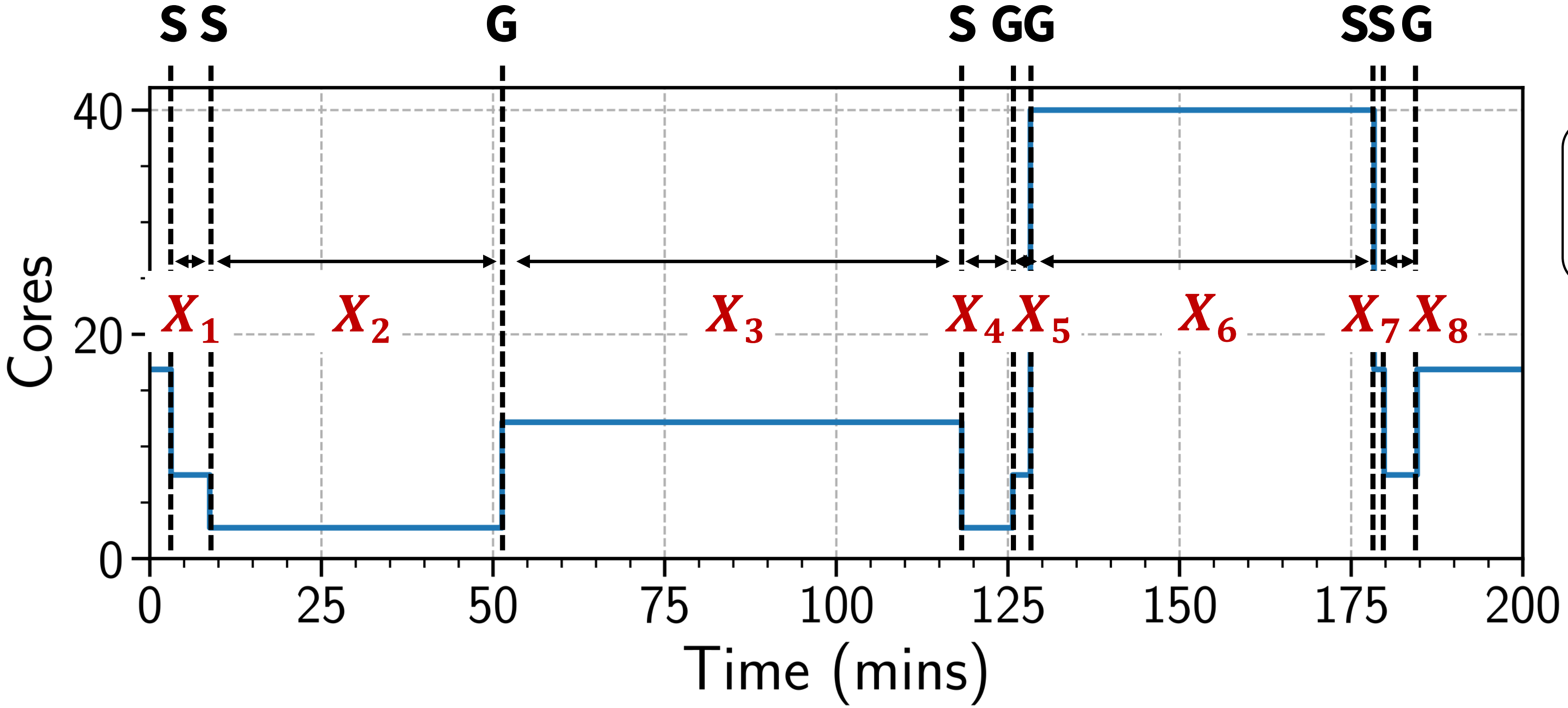
8 production clusters
- Magnitude of changes
- **Inter-change-times**
- Spatial variation
- Temporal variation

Workloads

Genomics, seismic imaging
- Task resources
- **Task runtimes**

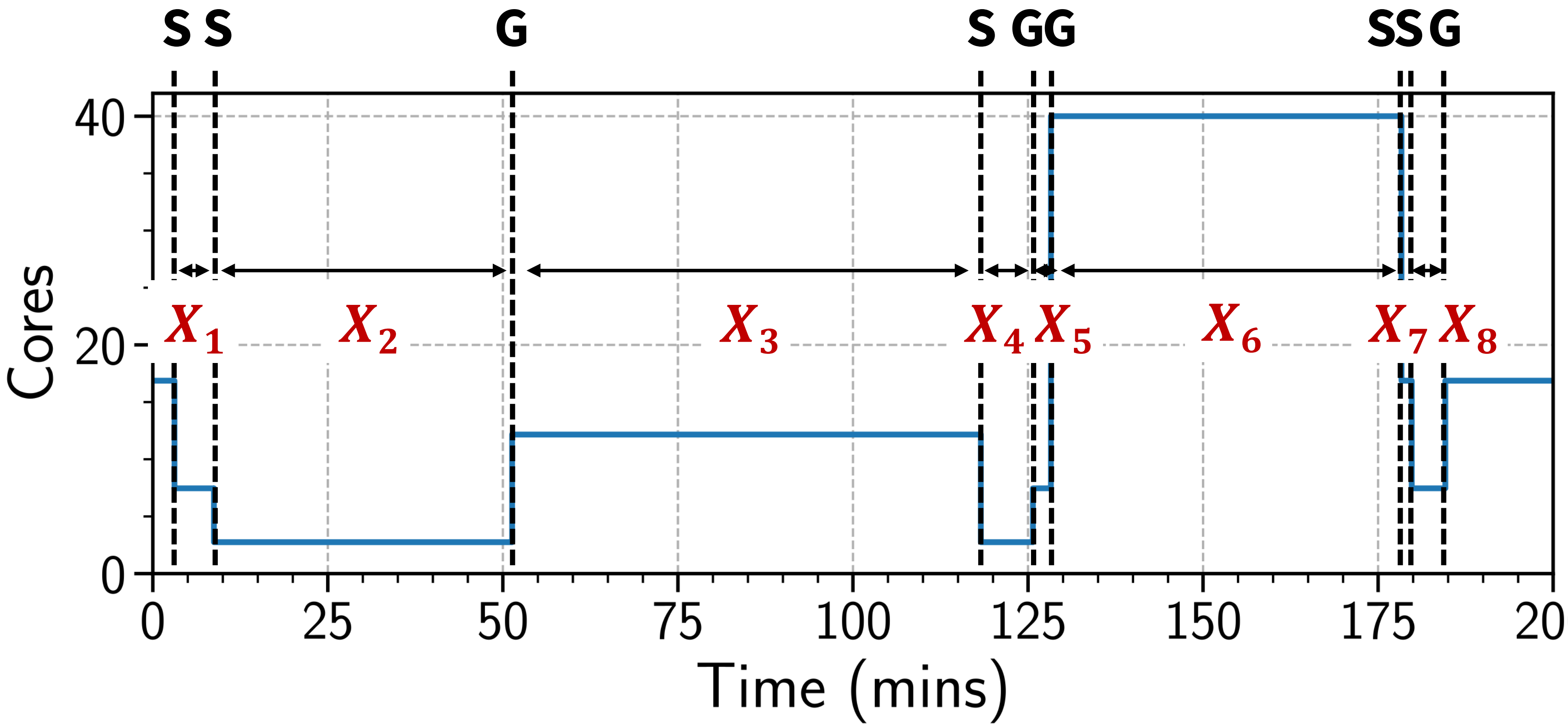
Other characterization in the paper.

Harvest VMs: Inter-change time



Grow: **G**
Shrink: **S**

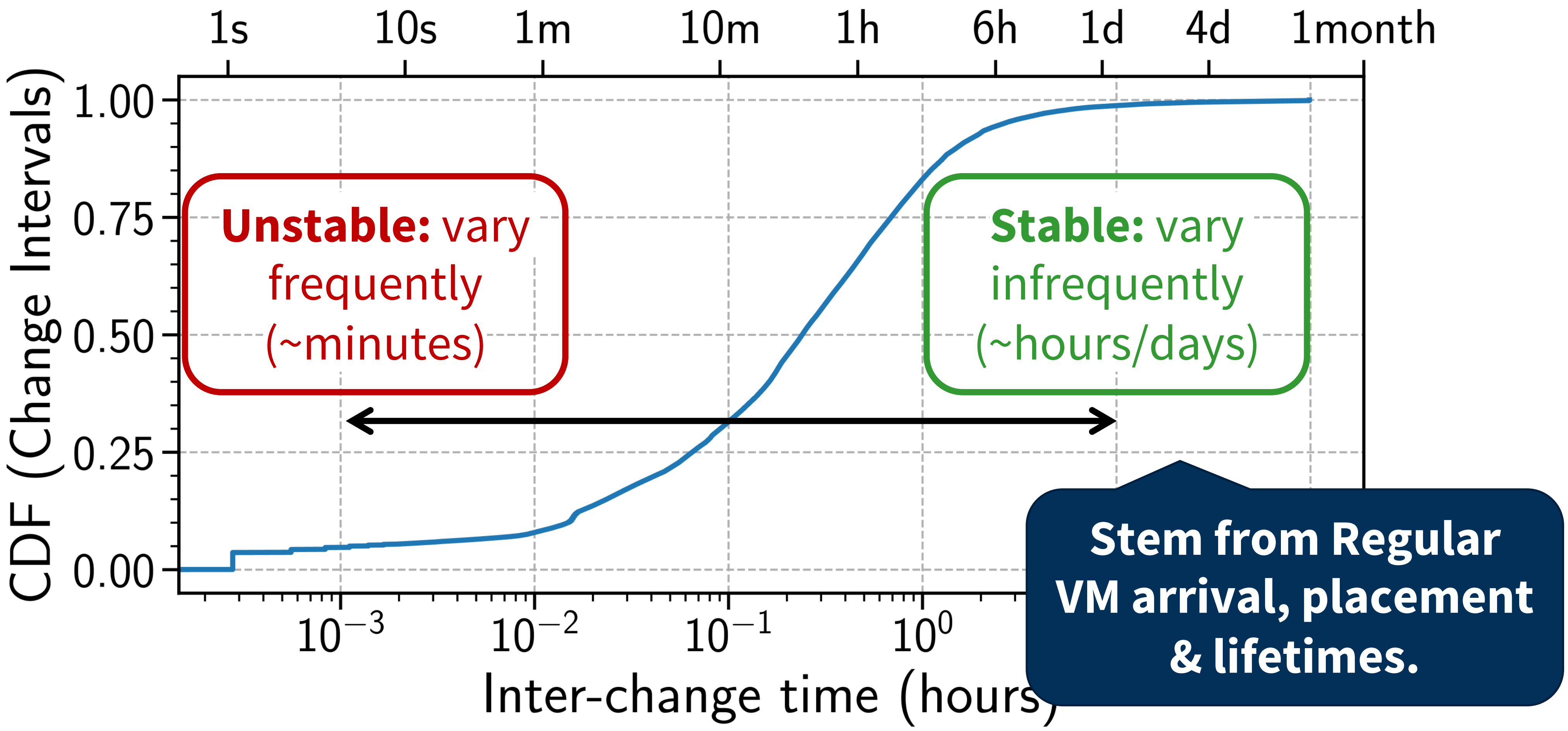
Harvest VMs: Inter-change time



Grow: **G**
Shrink: **S**

Inter-change time distribution (X): X_1 X_2 X_3 X_4 X_5 X_6 X_7 X_8

Obs #1: Inter-change time has high variance. At any time, mix of stable and unstable HVMs.

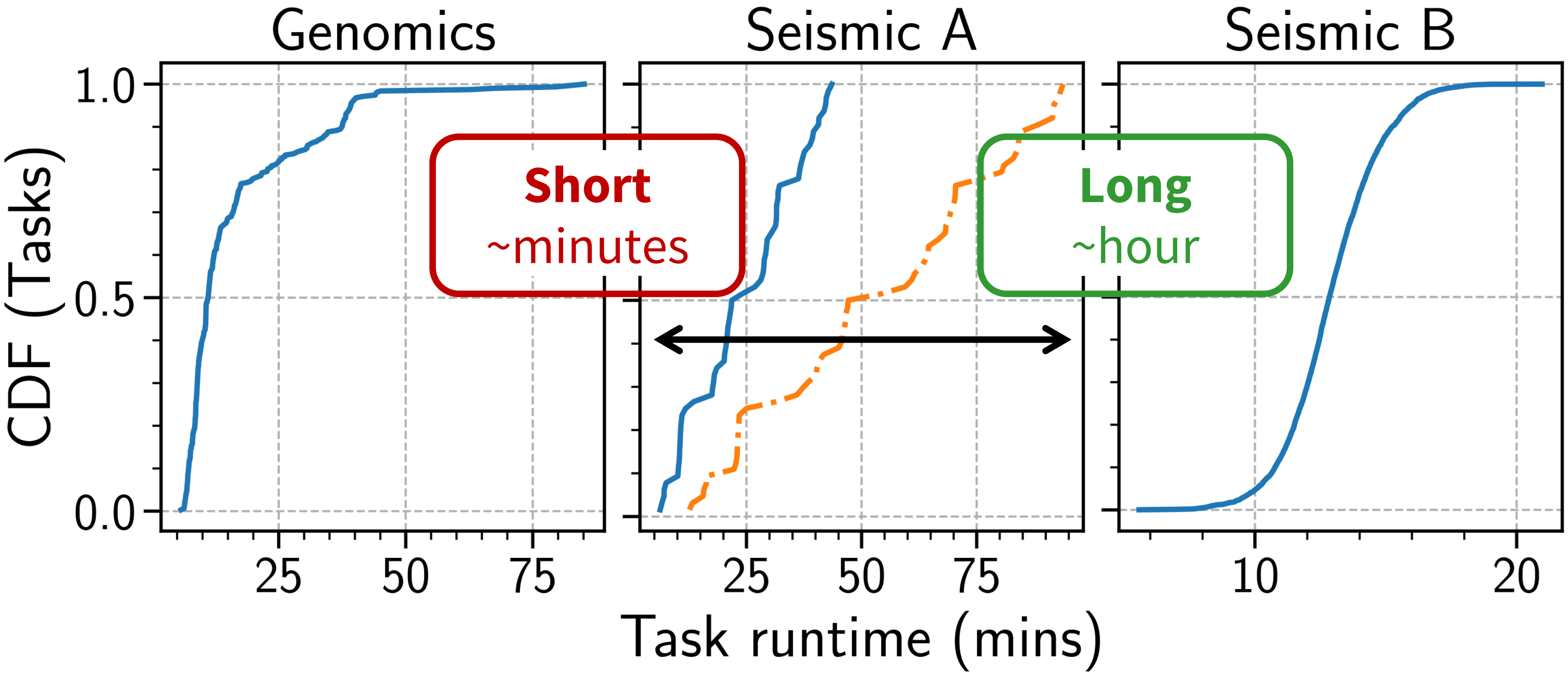


Unstable: vary frequently (~minutes)

Stable: vary infrequently (~hours/days)

Stem from Regular VM arrival, placement & lifetimes.

Obs #2: Task runtimes have high variance. At any time, mix of long and short tasks.



Short
~minutes

Long
~hour

SlackSched Roadmap

1. Characterization to understand the setting

- #1: Harvest VMs → **Mix of stable & unstable HVMs.**
- #2: Workloads → **Mix of short and long tasks.**

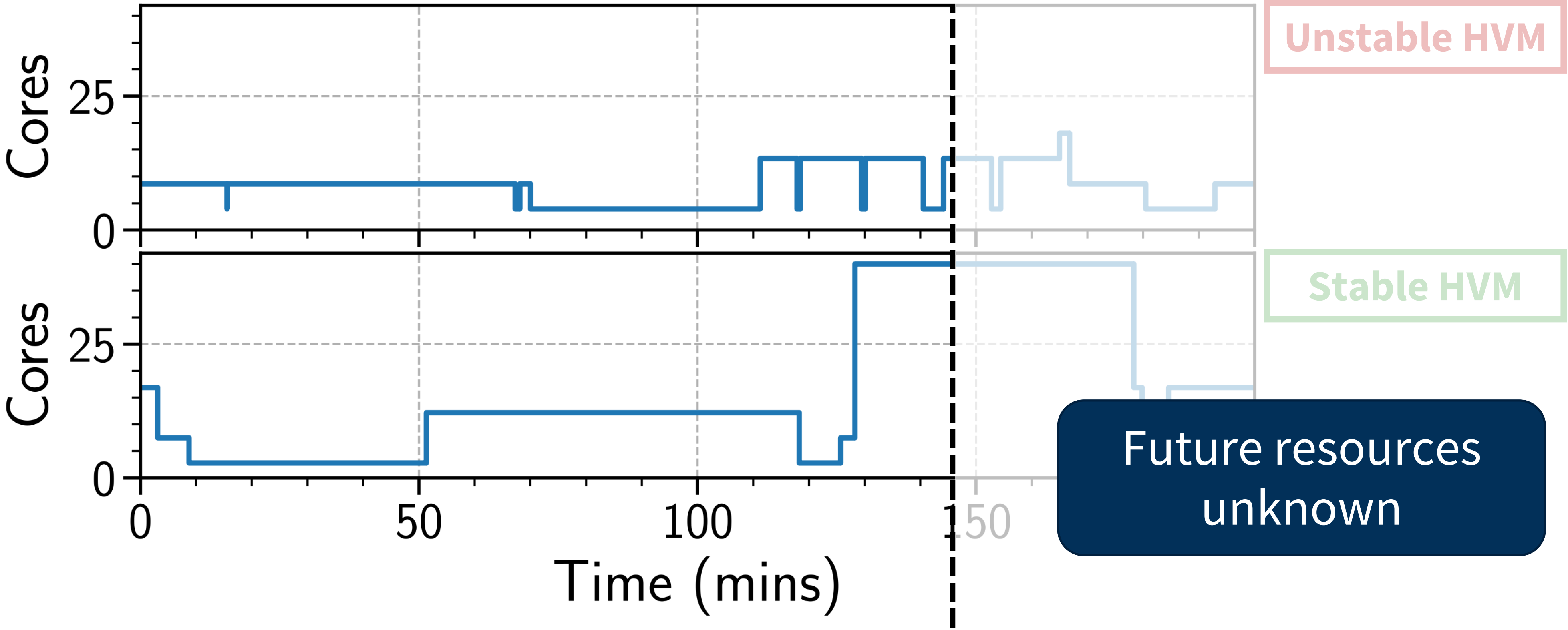
2. Opportunities to use HVMs more efficiently

- #1: Scheduling → **Match longer tasks to more stable HVMs.**
- #2: Resource Acquisition → **Maintain cluster of stable HVMs.**

Challenge. **Predict which HVMs will be stable.**

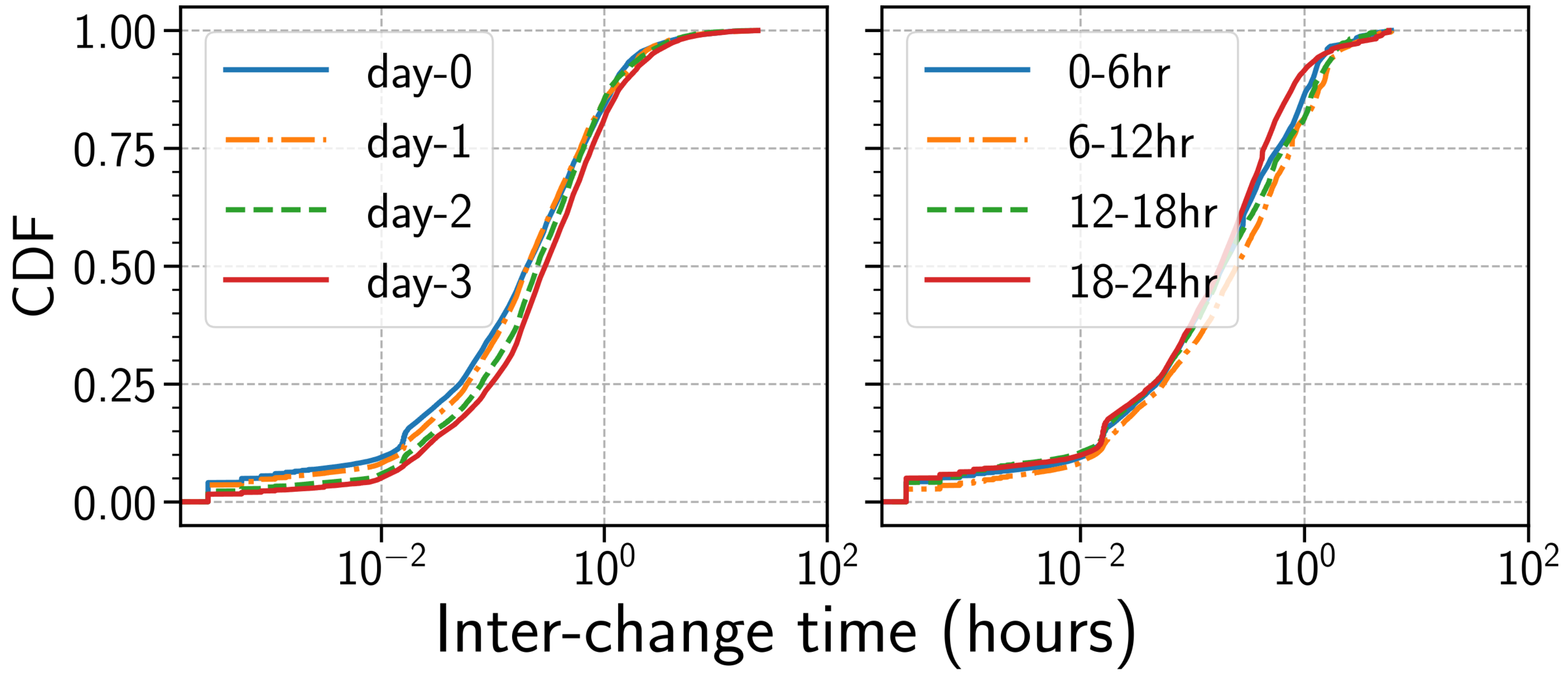
Opportunity #1: Scheduling
Match longer tasks to more stable HVMs

Longer task: 🟩 **Shorter task:** 🟥



Future resources unknown

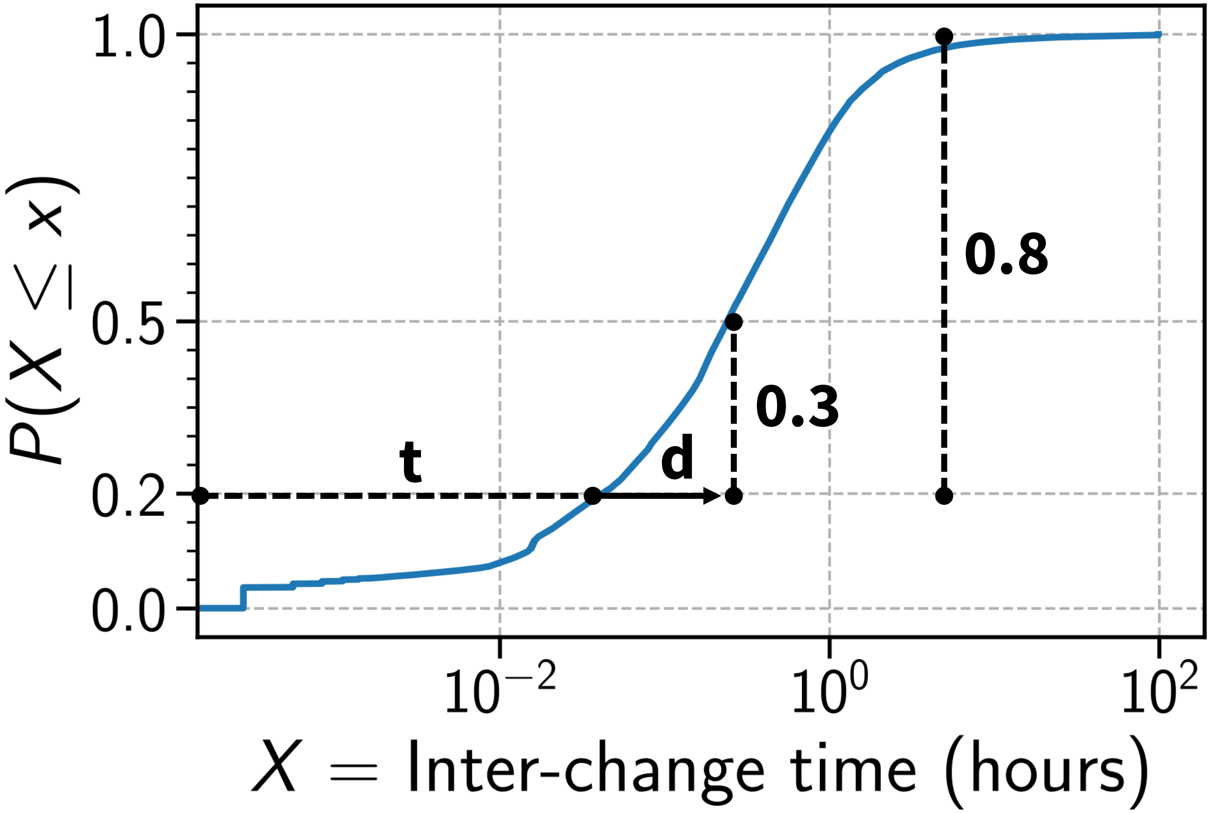
Inter-change time distribution is stationary over time. We know the distribution of future HVM changes.



Using inter-change time distribution to guide scheduling decisions

Conditional probability HVM changes before task (duration d) finishes.

$$P[X - t \leq d \,|X > t] = \frac{0.3}{0.8}$$

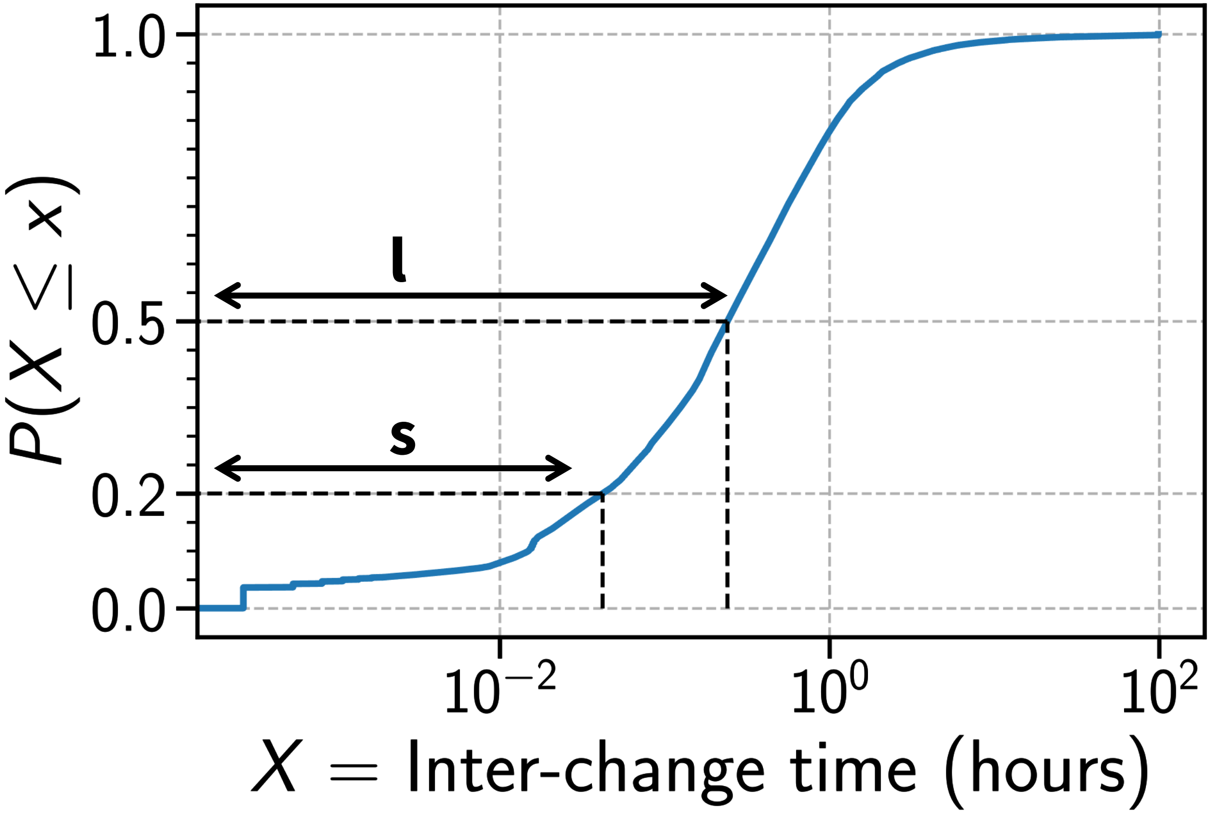


More details in the paper.
Likelihood of task preemptions.
Prob. of shrink. Task runtimes.

Opportunity #2: Resource Acquisition. Periodically deallocate unstable or most recently changed HVMs

The longer an HVM has stayed stable, the longer it will remain stable

$$E[X - l \mid X > l] > E[X - s \mid X > s]$$



SlackSched: Putting it all together

Scheduling

- Match longer tasks to more stable HVMs.
- Minimize likelihood of task preemption.

Resource Acquisition

- Maintain cluster of stable HVMs.
- Deallocate most recently changed HVMs.

Implementation & Evaluation

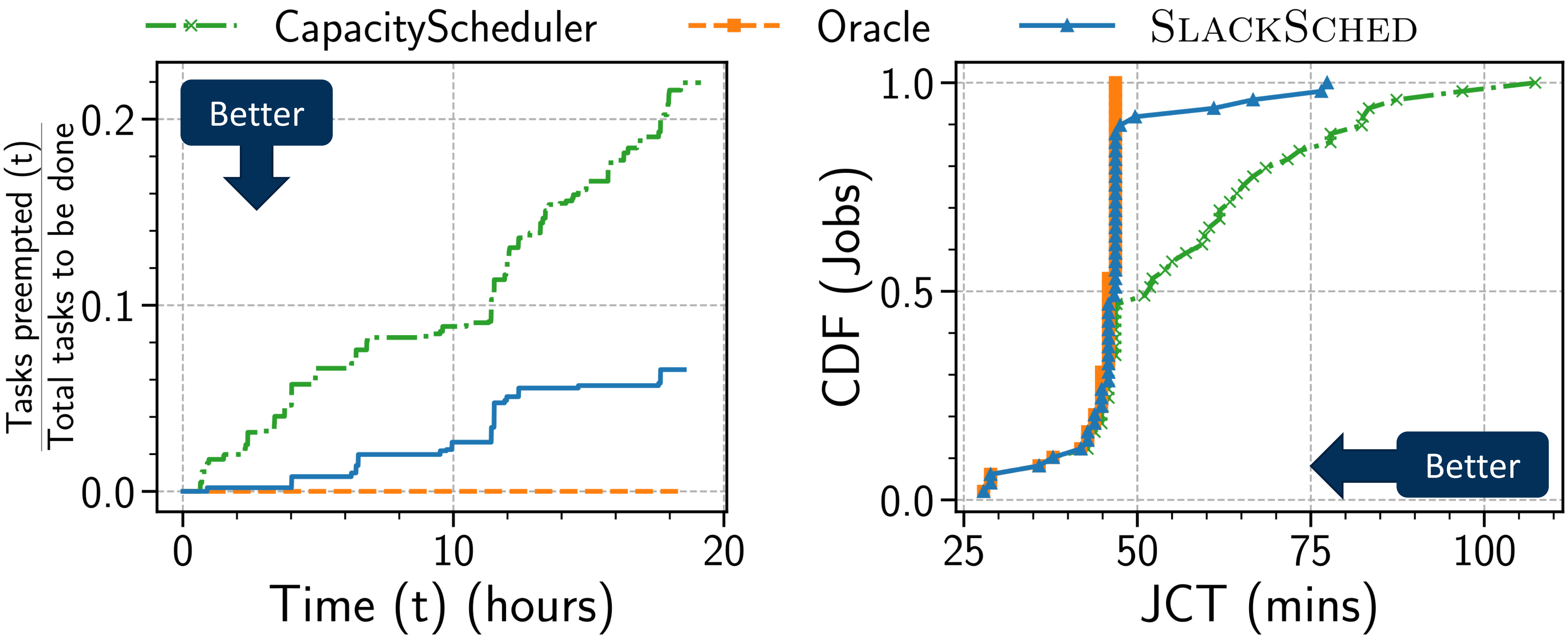
Implement within Hadoop YARN.

Evaluate using:

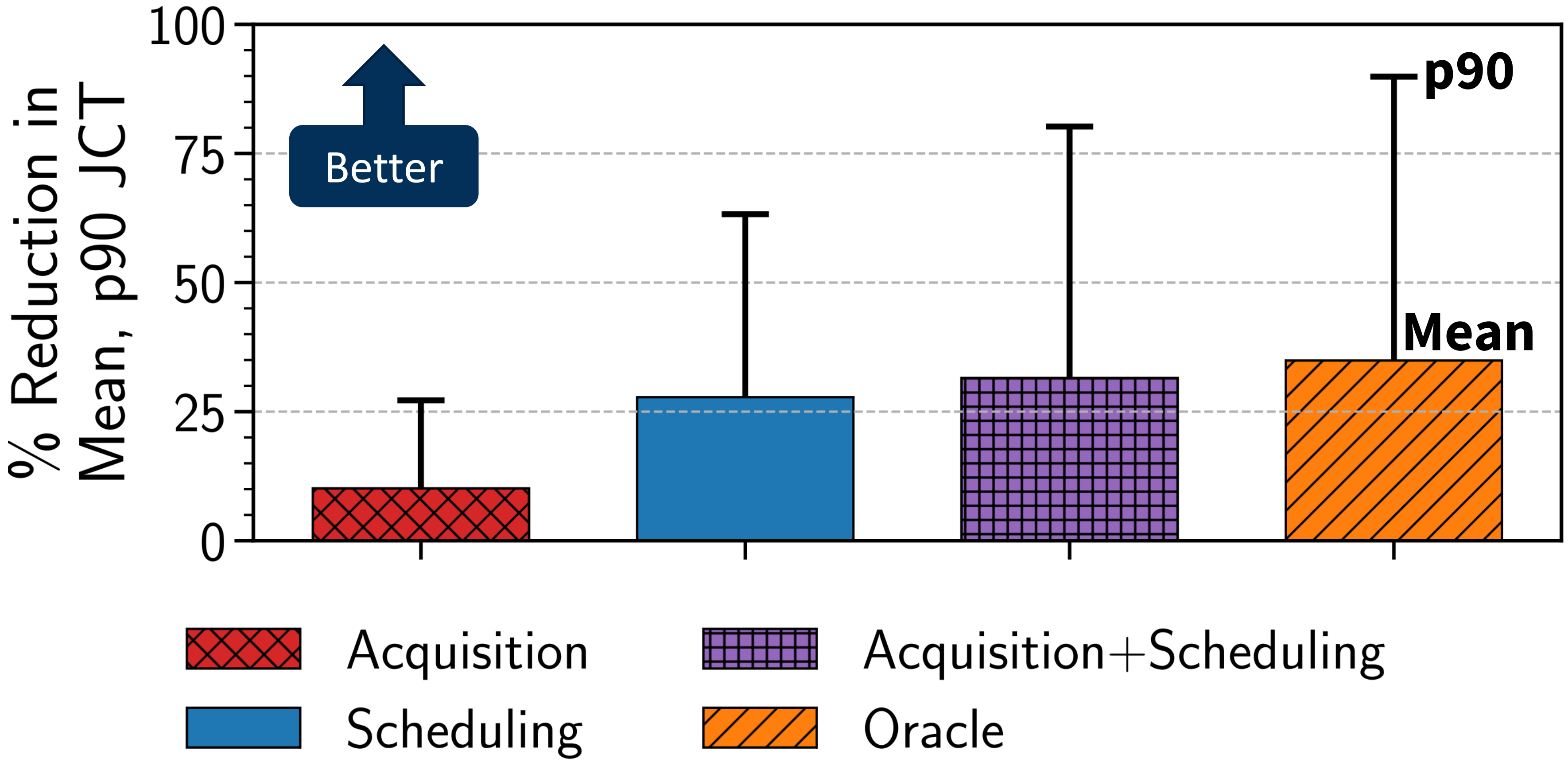
- HVM traces from 8 production clusters (~1000 servers each).
- Workload traces from genomics/seismic domains.

Metrics: Job completion time (JCT) & Cost.

Scheduling reduces mean and p90 JCT by ~27% and 44%



Scheduling and Resource Acquisition complement each other



More evaluation in the paper

Workload parameters

- Task runtime distributions
- Errors in task runtime estimates
- Load, Time varying load

Environment parameters

- HVMs in different clusters
- Different HVM allocation policies
- Comparison with Spot VMs

Empirically compare against checkpointing, migration, replication

Summary: Better use HVMs for long, uninterruptible workloads

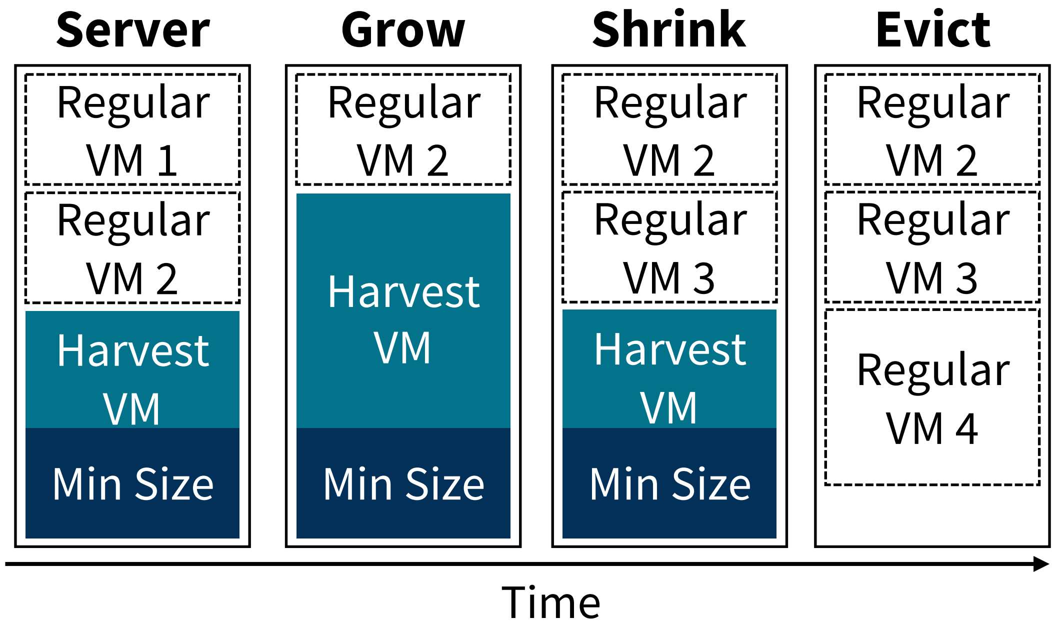
Distribution based predictions to guide

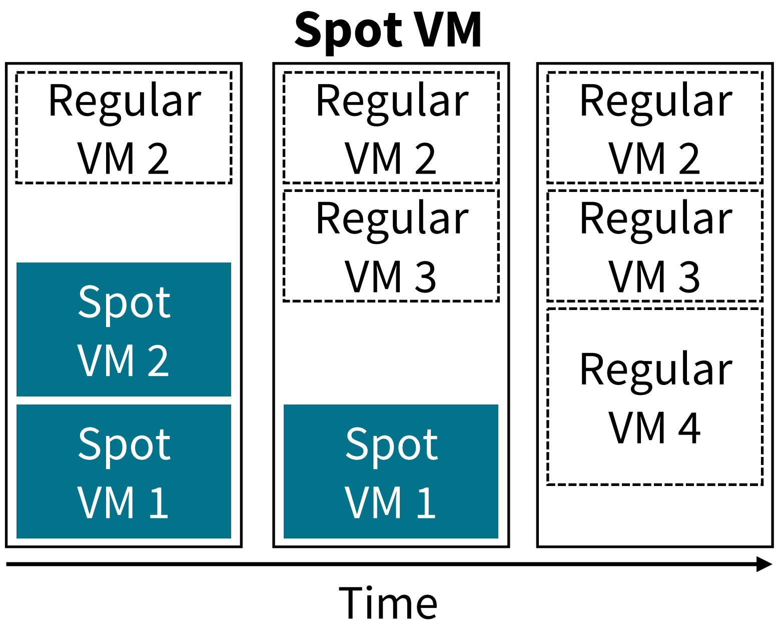
- **Scheduling.** Match tasks to HVMs.
- **Resource Acquisition.** Maintain cluster of stable HVMs.

- ~27% improvement in mean job completion time.
- ~44% improvement in p90 JCT.
- 75% lower cost than regular VMs.

Backup Slides

Harvest VMs vs. Spot VMs

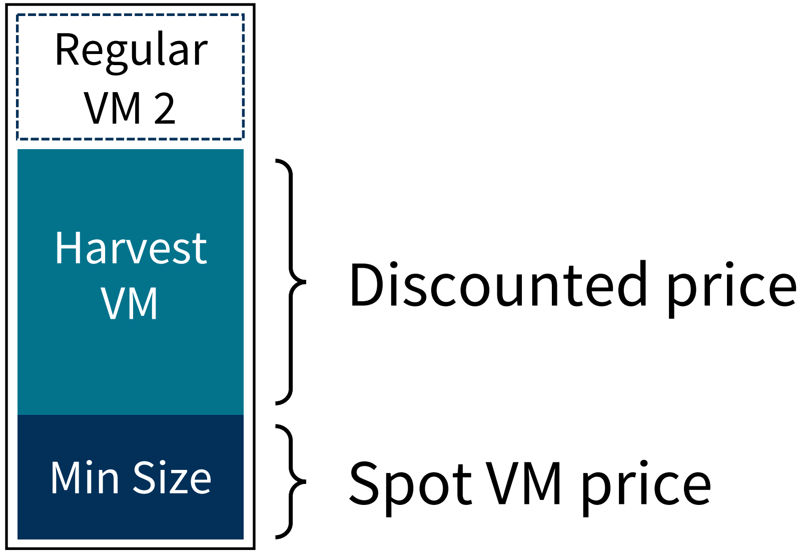




- 3.6x lower eviction rates
- 2.5-7.5x bigger in capacity
- 35-44% cheaper

Harvest VMs vs. Spot VMs

- 3.6x lower eviction rates
- 2.5-7.5x bigger in capacity
- 35%-44% cheaper



- Spot is 48%-88% cheaper than regular