THROUGHPUT COMPUTING 2024

Optimizing Cost and Performance Best practices for Efficient HTCondor Workload Deployment in AWS

Sudhi Bhat

He/Him
Principal Specialist SA, Compute
AWS

Agenda

- HTCondor in AWS
- AWS Compute Best practices and tools
- Case Studies

Workload requirements

Compute heavy

Workloads with high performance (HPC) or throughput (HTC) compute requirements

- ***Characteristics:***
- ✓ Job/task duration
- ✓ Task throughput (tasks per second)
- ✓ Scale (no. of CPUs/memory)
- ✓ Framework, language, and platform
- ✓ Ex: Simulations, Risk Modeling

Data heavy

Workloads with high data access, transformation (ETL) or data analytics requirements

- ***Characteristics:***
- ✓ Data transformation (ETL)
- ✓ Streaming
- ✓ Access patterns (no. of connections/file size)
- ✓ Data locality (local or remote access)
- ✓ Ex: Back-testing trading strategies, Genomic Sequencing, Drug Discovery, Research

What's driving this transformation?

Siloed on-premises centralized HPC resource

Skillset shortage

Cost optimization/Sustainability

Cost of hosting/replacing aging physical compute

Customer market signals

Regulatory requirements – FRTB/IFRS17

Increasing market volatility

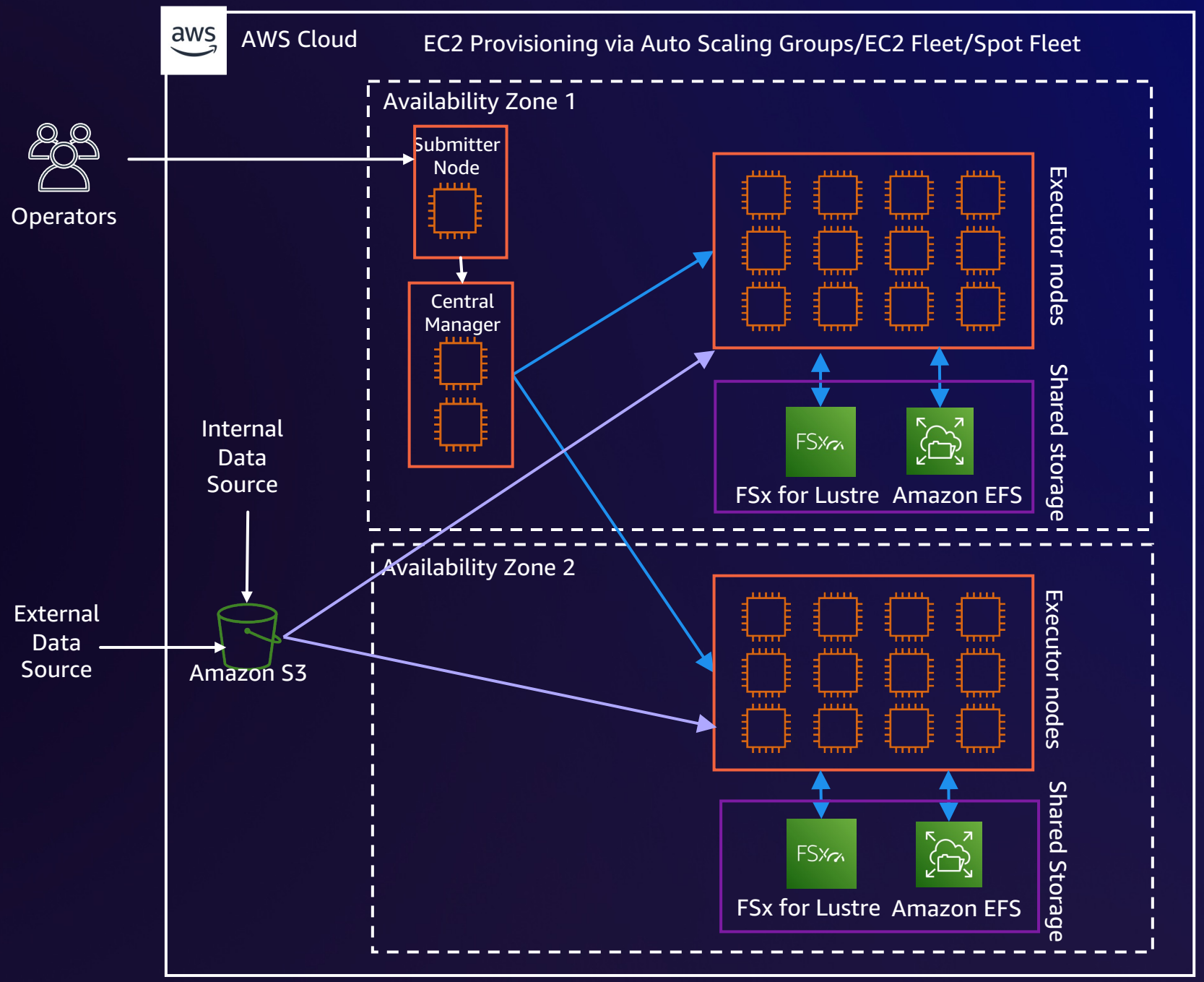
Desire for faster results and on-demand analytics

Competitive advantage on innovation and faster time-to-market

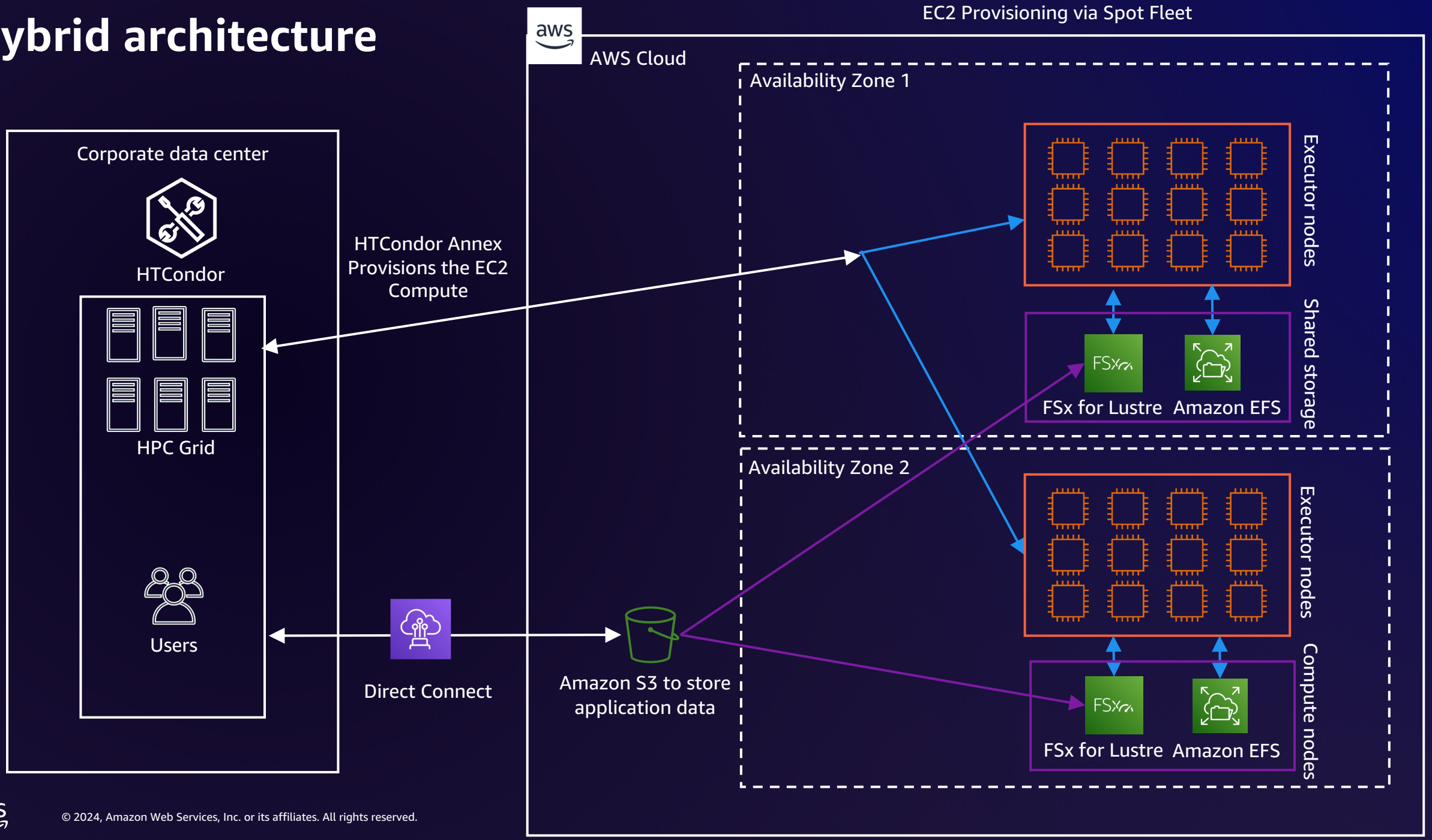
HTCondor in AWS

Primary components:

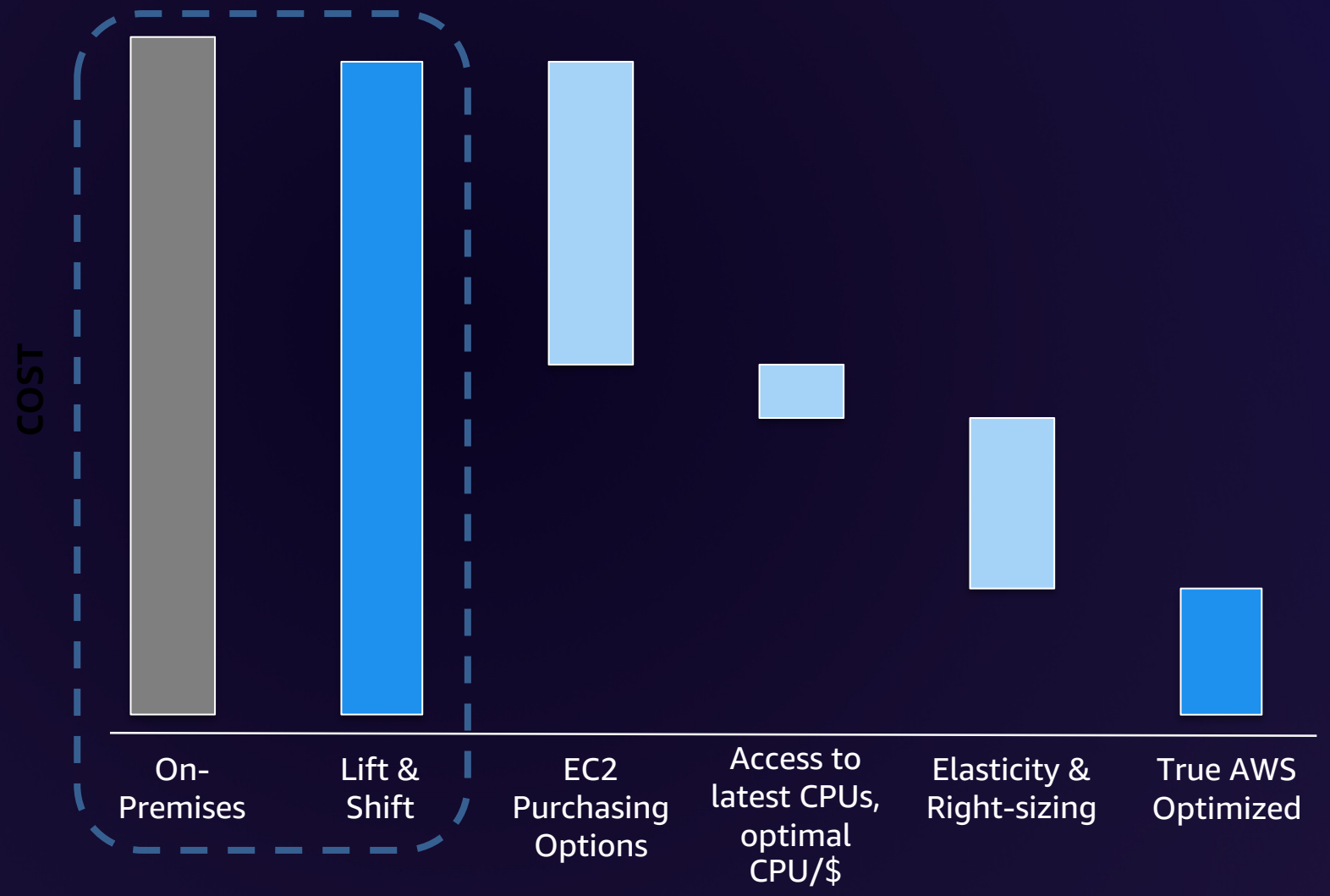
- Compute
- Storage
- Networking



Hybrid architecture



AWS Best Practices for Compute Optimization



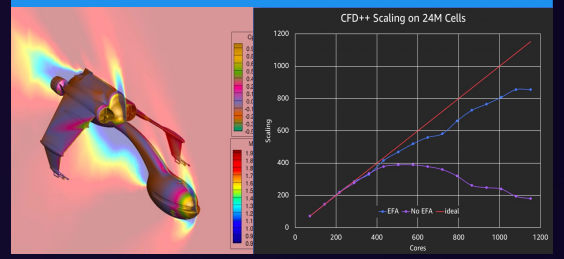
Benefits:

- ✓ **Purchasing Options** – optimize your costs based on your compute needs
- ✓ **CPU/Memory to $** – Selection of instances to match the requirements of different workloads at the best price
- ✓ **Elasticity** – Matching capacity to demand, even where silos exist
- ✓ **Right-Sizing** – Optimize the instance usage by closely tracking relevant metrics

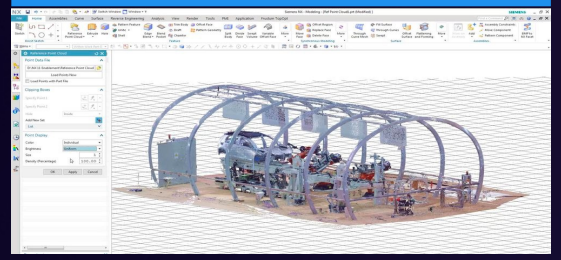
Flexible compute options and purchase models optimize price performance

Flexible compute to maximize performance

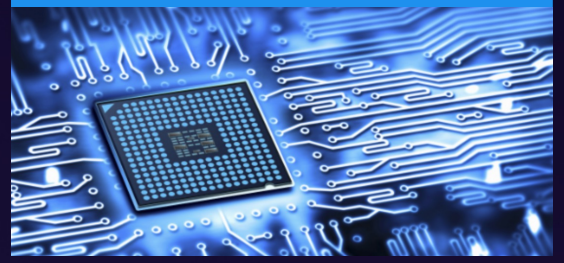
Memory & compute optimized
2.5-3.5Ghz, 2-16GB/core, 100Gbps, EFA



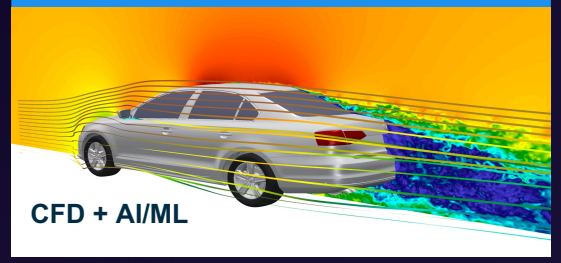
Graphics and rendering
1-8 GPUs, up to 384GB RAM, SSD



High clock speed
4.5Ghz, 192GB RAM, 100Gbps, EFA



Accelerated computing
8 A100 GPUs, 1.1TB RAM, SSD, 400Gbps

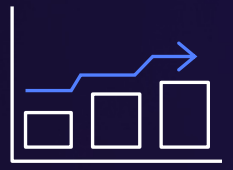


Flexible pricing models to optimize cost

On-Demand

Pay for compute capacity by the second with no long-term commitments.

Savings Plan & Reserved Instances



Make a commitment and to save up to 72% off compute.

Spot Instances

Spare EC2 capacity at savings of up to 90% off On-Demand prices.

EC2 Spot Instances

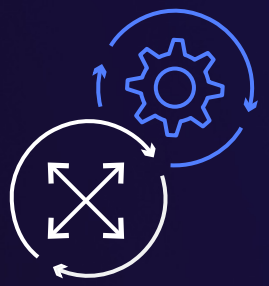
Spot infrastructure

Is **same** as
On-Demand and RIs

Spot pricing

Smooth, infrequent changes
no spikes, more predictable

Up to 90% off (compared to On-Demand pricing)



Interruptions

Happen when On-Demand
instances needs capacity

Diversify

Choose different instance types,
size and AZ in a single fleet

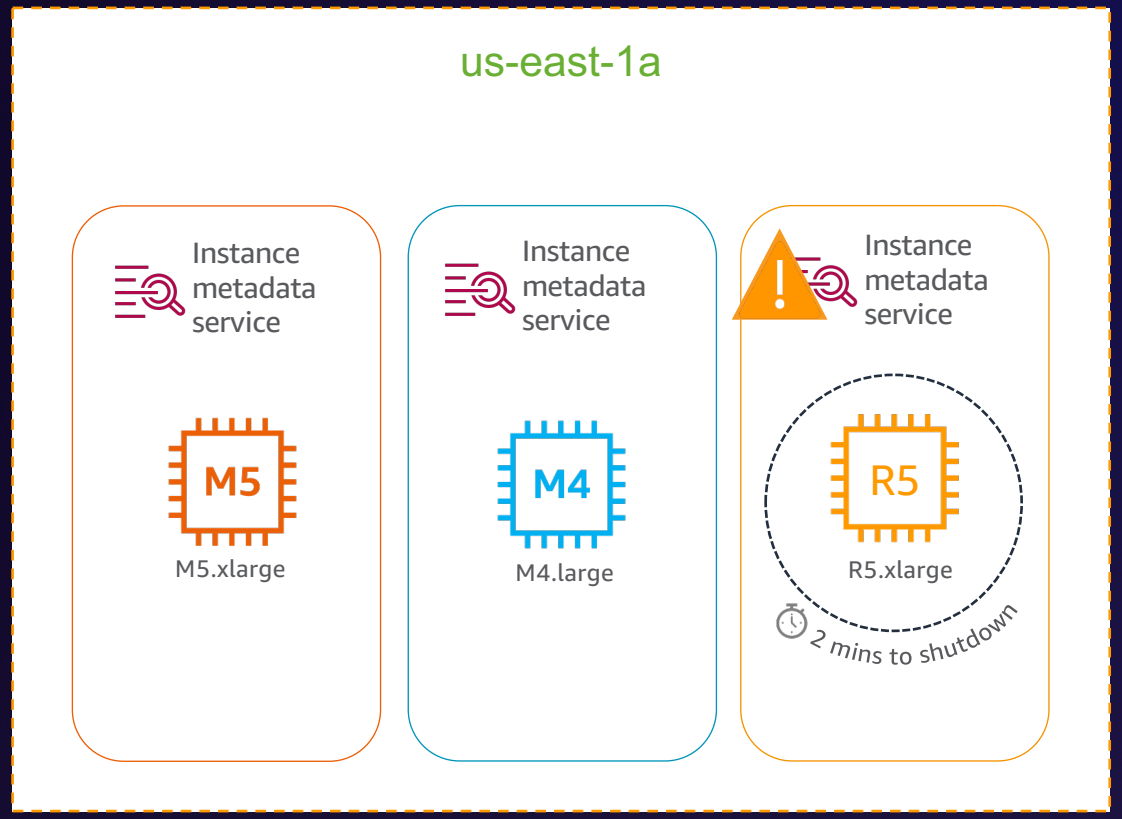
Interruptions Deep Dive

EC2 instance rebalance recommendation (proactive)

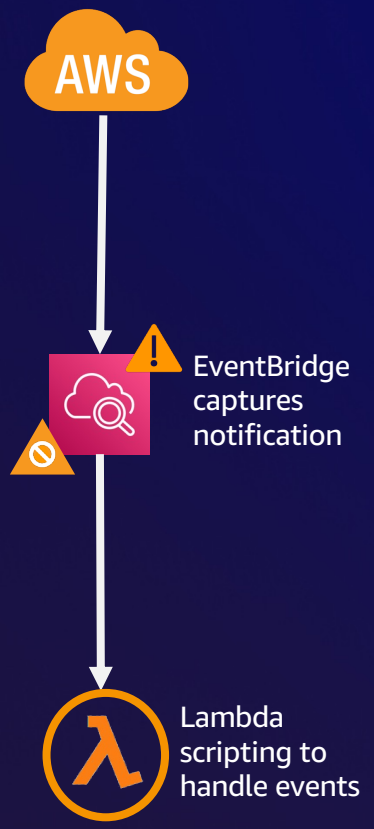
- When an instance is elevated risk of interruption a notification will be sent
- This means the demand for the instance is higher/capacity is lower and is likely to be reclaimed back
- This enables customers to either do their own automation on notification or make use of integrations such as EC2 Auto Scaling

Spot instance termination notice (reactive)

- When an instance will be shut down in 2 minutes at it is required by EC2 (on-demand customer)
- Research shows 2 minutes is plenty of time to pause, hibernate, stop or redistribute workloads
- AWS has DIY recipes or simply make use of integrations to handle it for you



Alternate to Instance metadata service



Diversification is Key

Be diverse and flexible to maintain your target capacity

A Spot Pool (each $x.xx) is a set of unused / spare-capacity instances priced separately based on:

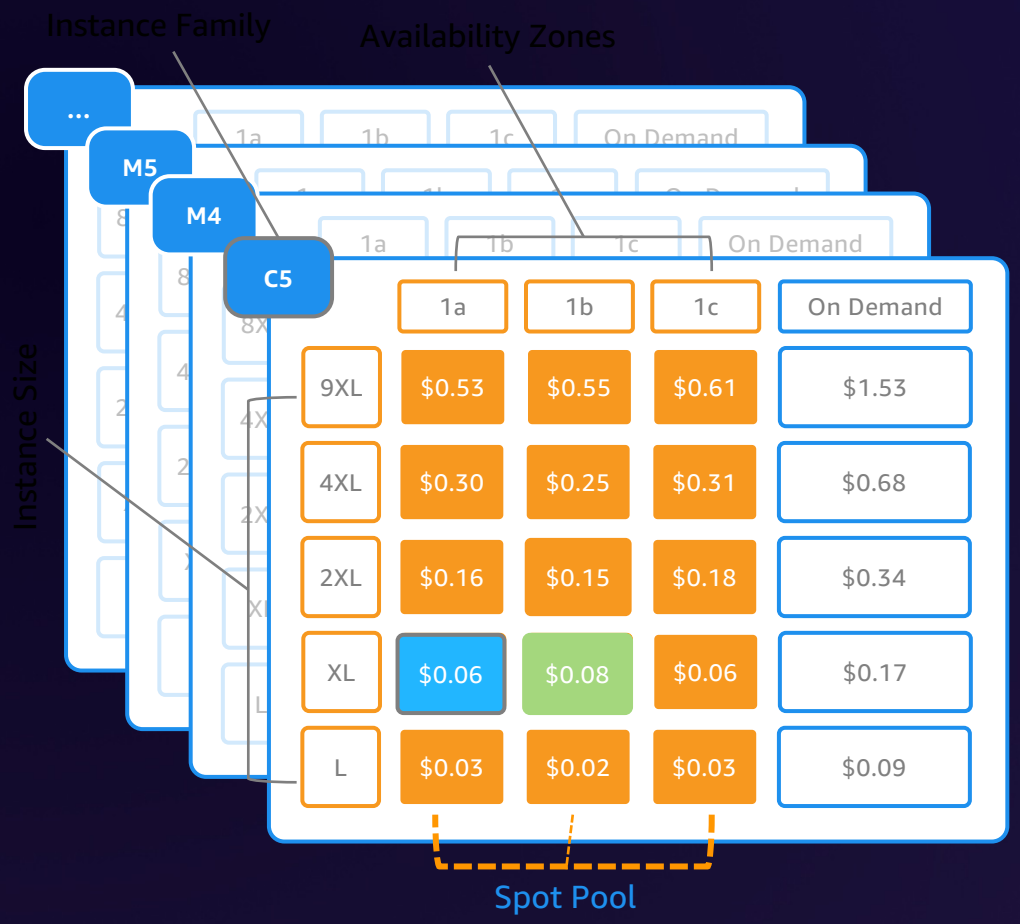
- Instance Family
- Instance Size
- Availability Zone
- Region

Example:

C5.xlarge-1A-DUB

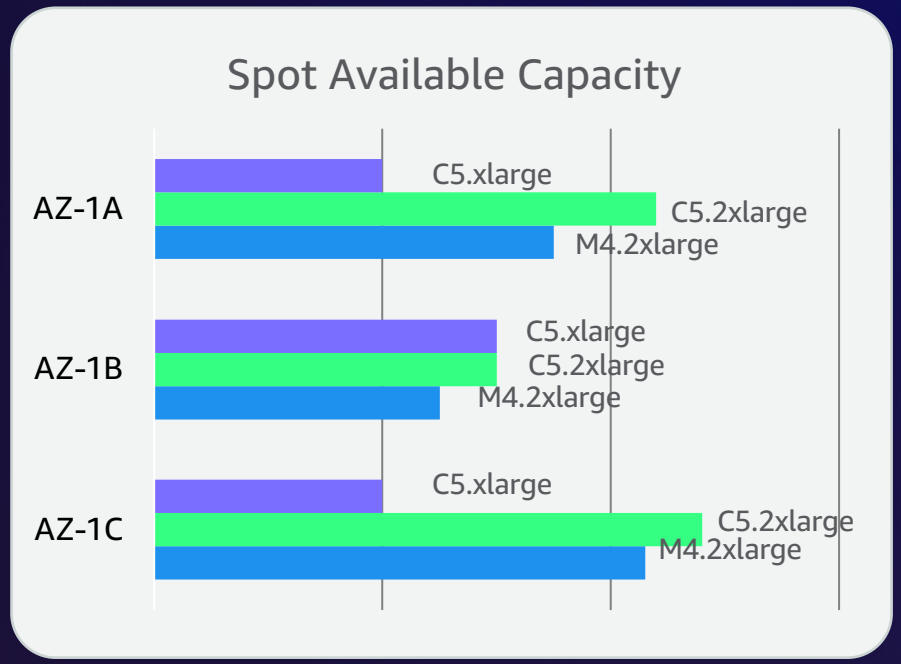
will have different price/capacity than

C5.xlarge-1B-DUB



Mixing Spot Pools is key to ensuring high capacity can be met due to different availability of instance types

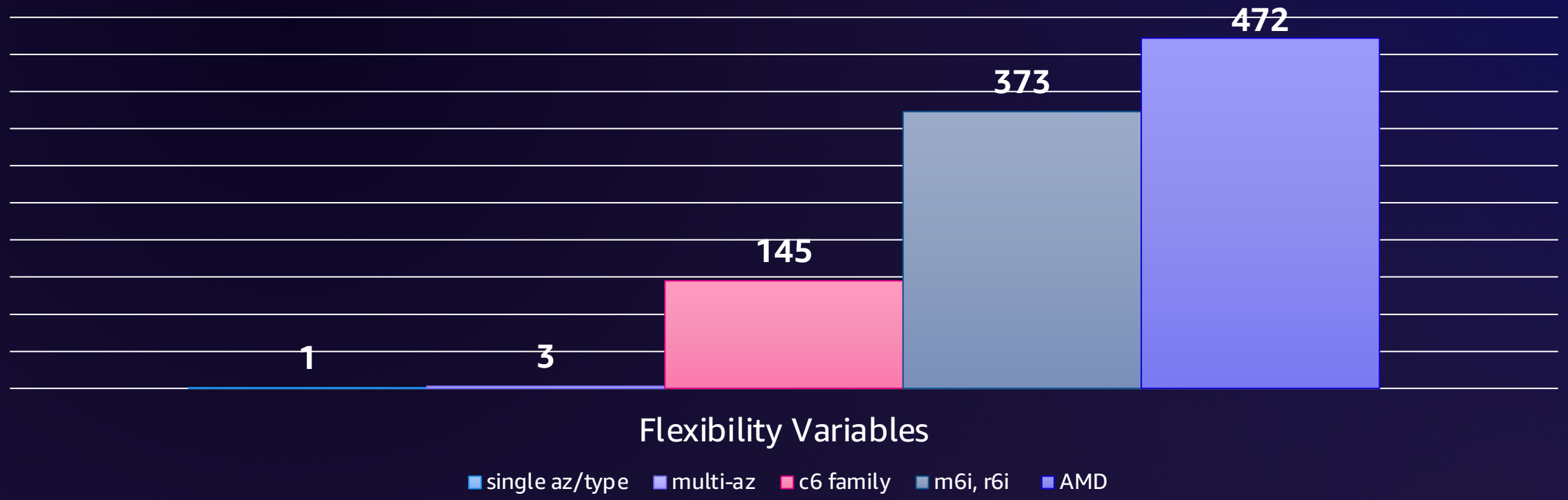
Example:



The Power of Flexibility

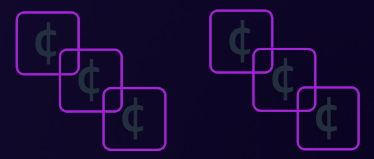
Say you are using c6i.large.

What other capacity pools can you recommend?



EC2 Spot Allocation strategies

Lowest Price



- Select instances and AZs - AWS provisions least cost available Spot instance

Cost Optimization is your primary driver

Diversified

- The Spot Instances are distributed across all Spot capacity pools.

Improve availability of your fleet

Capacity Optimized

- AWS provisions from the pool with the greatest capacity

Interruption minimization is your primary driver

*Recommended

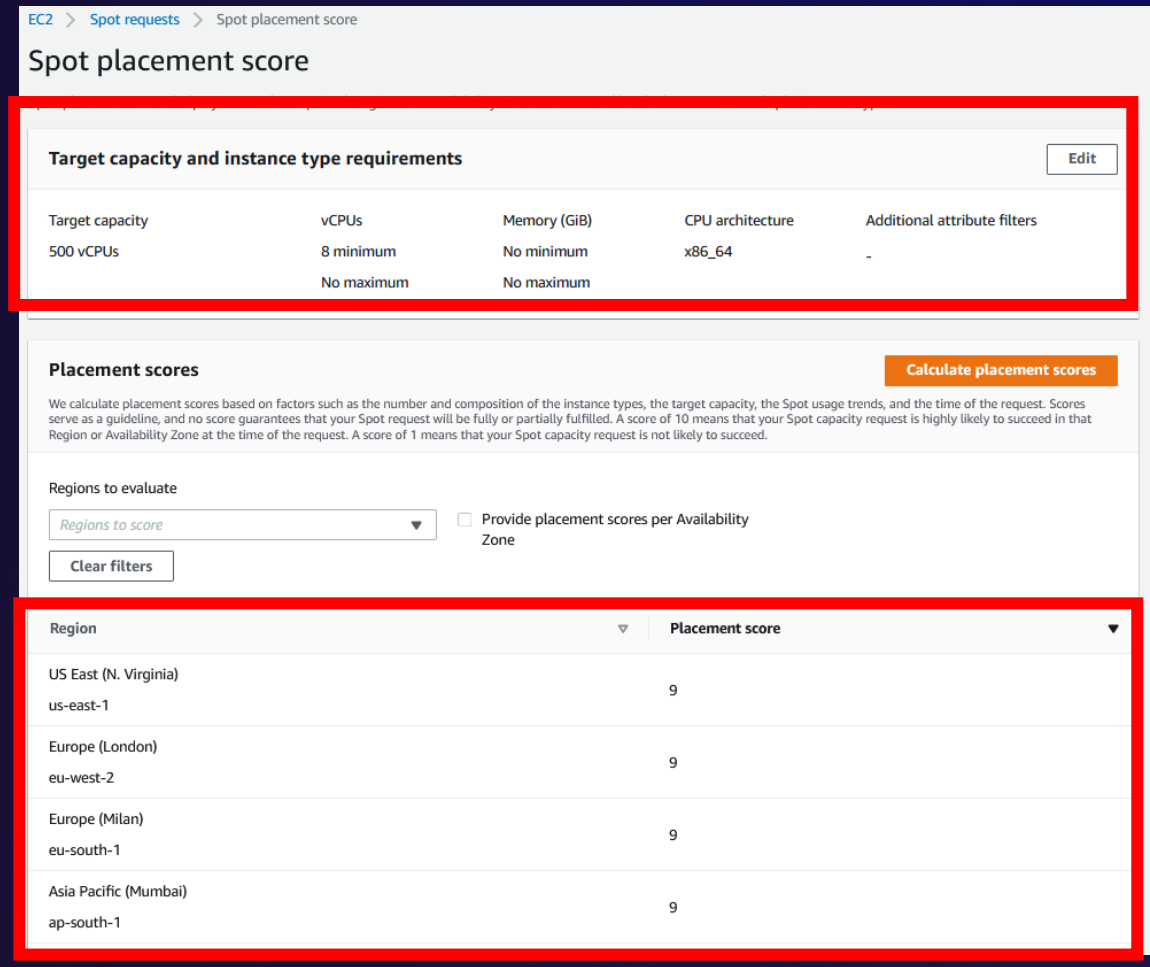
Price-Capacity Optimized ★

- AWS provisions from the pool with the greatest capacity while prioritizing Cost

Interruption minimization is your primary driver while focusing on Cost

Spot Placement Score (SPS)

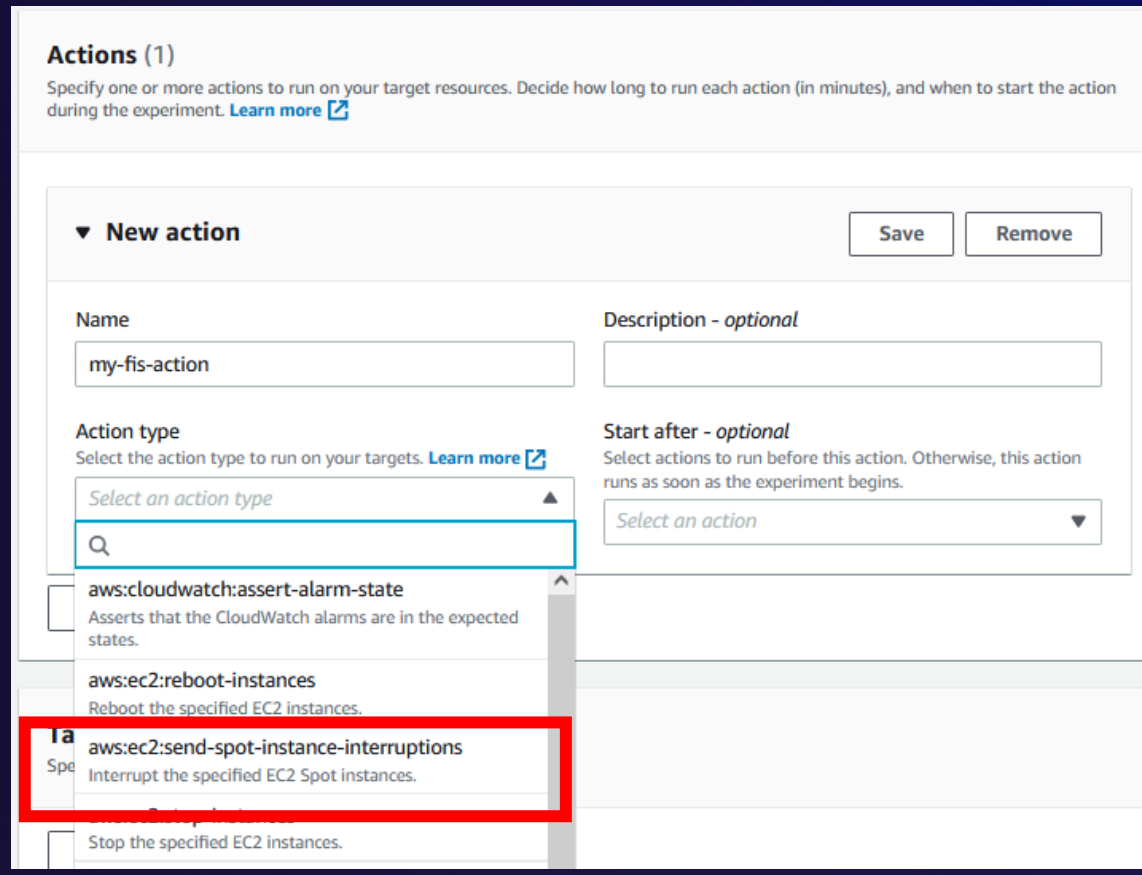
- The ability to indicate which region or AZ is the most likely given criteria when launching Spot Instances at a given point in time
- EC2 capacity changes from moment to moment, so the results will vary with each request, and based on the target capacity
- The target capacity that can be specified for scoring is determined by how many Spot Instances have been launched



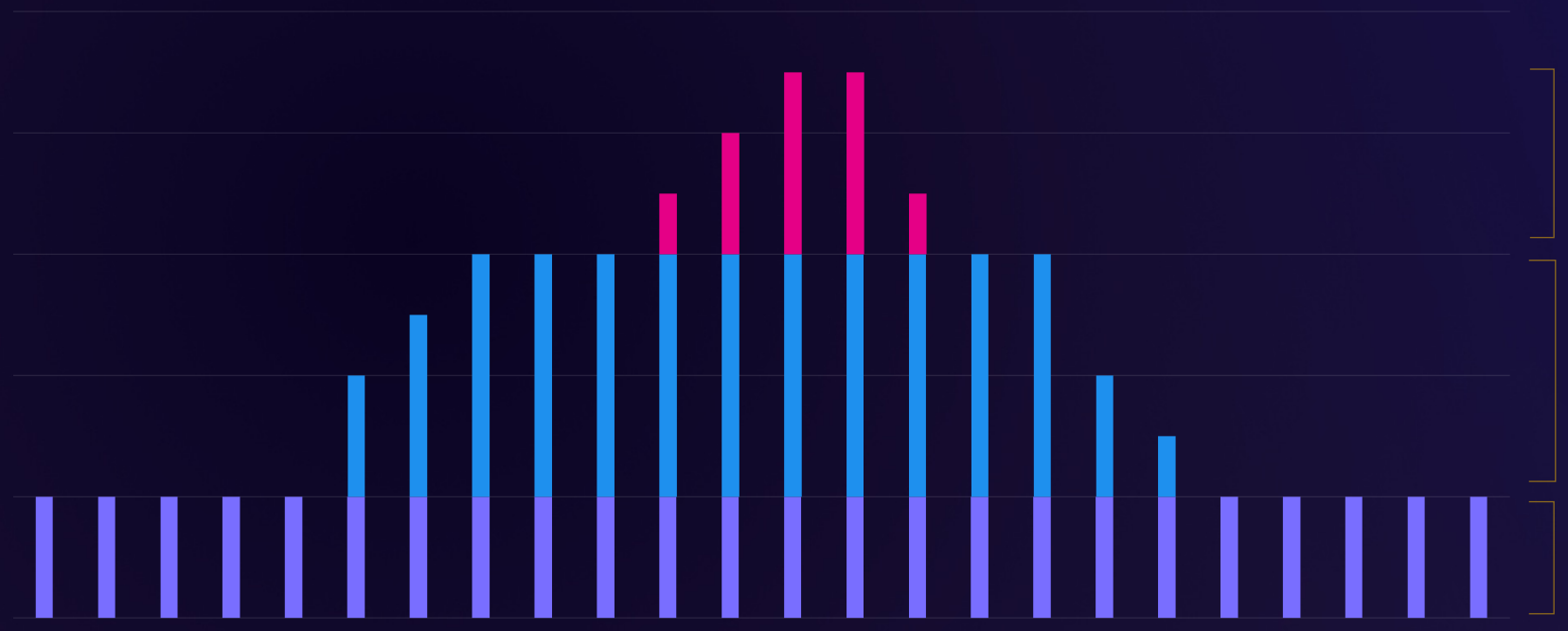
https://aws-solutions-library-samples.github.io/compute/building-a-spot-placement-score-tracker-dashboard-on-aws.html#configuration-settings

Simulation of Spot interruptions with FIS

- Fault Injection Simulator (FIS) – simulates Spot Instance interruptions
- The actual interruption occurs after the rebalance notification and interruption notification are sent



Combine Purchase Options to Optimize costs



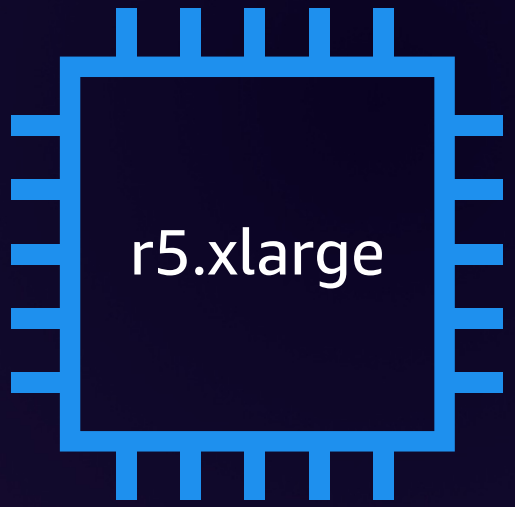
Scale using Spot for fault-tolerant, flexible, stateless workloads

On-Demand, for new or stateful spiky workloads

Use **Savings Plans/RIs for known, steady-state workloads**

Elasticity: Attribute-based instance selection

STOP PICKING INSTANCES! INSTEAD TELL US WHAT YOU ACTUALLY NEED…




```
{
  "ArchitectureTypes": [ "x86_64" ],
  “VirtualizationTypes": [ "hvm" ],
  "InstanceRequirements":
   {
       "VCpuCount": { "Min": 4 },
       "MemoryMiB": { "Min": 32768 },
       "InstanceGenerations": [ "current" ]
    }
}
```

ec2-instance-selector

A discovery CLI tool that can be useful for analysis of instance types you could be flexible with, on the basis of resource criteria

```
$ ec2-instance-selector --vcpus 4 --memory 16 --cpu-architecture x86_64 --gpus-max 0 –o table

Instance Type      VCPUs      Mem (GiB)

-------------      -----      ---------

m4.xlarge          4          16.000

m5.xlarge          4          16.000

m5a.xlarge         4          16.000

m5ad.xlarge        4          16.000

m5dn.xlarge        4          16.000
```

https://github.com/aws/amazon-ec2-instance-selector

On-Demand Capacity Reservations (ODCR)

For steady-state workloads

- Manage capacity and discounts independently
- No commitment required; can be created and canceled as needed
- Reserve capacity basis Availability Zone, Instance Type, Tenancy and Platform/OS
- Capacity held whether or not you run instances
- Share reservations across accounts
- No upfront/ additional charges (charged at equivalent On-demand rate)*

* Billing starts as soon as an ODCR is provisioned in the account

Condor Workloads can run on Arm architecture

HTCondor's ClassAd Mechanism

ClassAds are a flexible mechanism for representing the characteristics and constraints of machines and jobs in the HTCondor system. ClassAds are used extensively in the HTCondor system to represent jobs, resources, submitters and other HTCondor daemons. An understanding of this mechanism is required to harness the full flexibility of the HTCondor system.

A ClassAd is a set of uniquely named expressions. Each named expression is called an attribute. The following shows ten attributes, a portion of an example ClassAd.

```
MyType        = "Machine"
TargetType    = "Job"
Machine       = "froth.cs.wisc.edu"
Arch          = "INTEL"
OpSys         = "LINUX"
Disk          = 35882
Memory        = 128
KeyboardIdle  = 173
LoadAvg       = 0.1000
Requirements  = TARGET.Owner=="smith" || LoadAvg<=0.3 && KeyboardIdle>15*60
```

docker hub | Search Docker Hub ⌘+K | Sign In | Sign up

Explore / htcondor/mini_arm64

htcondor/mini_arm64 ☆0

By htcondor • Updated 13 days ago

IMAGE

Pulls 6

Overview | Tags

Sort by Newest | Filter Tags

TAG

23.8.1-el9

Last pushed **13 days ago** by timtheisen

```
docker pull htcondor/mini_arm64:23.8.1-el9
```
Copy

Digest	OS/ARCH	Compressed Size
20665a317411	linux/arm64/v8	285.02 MB

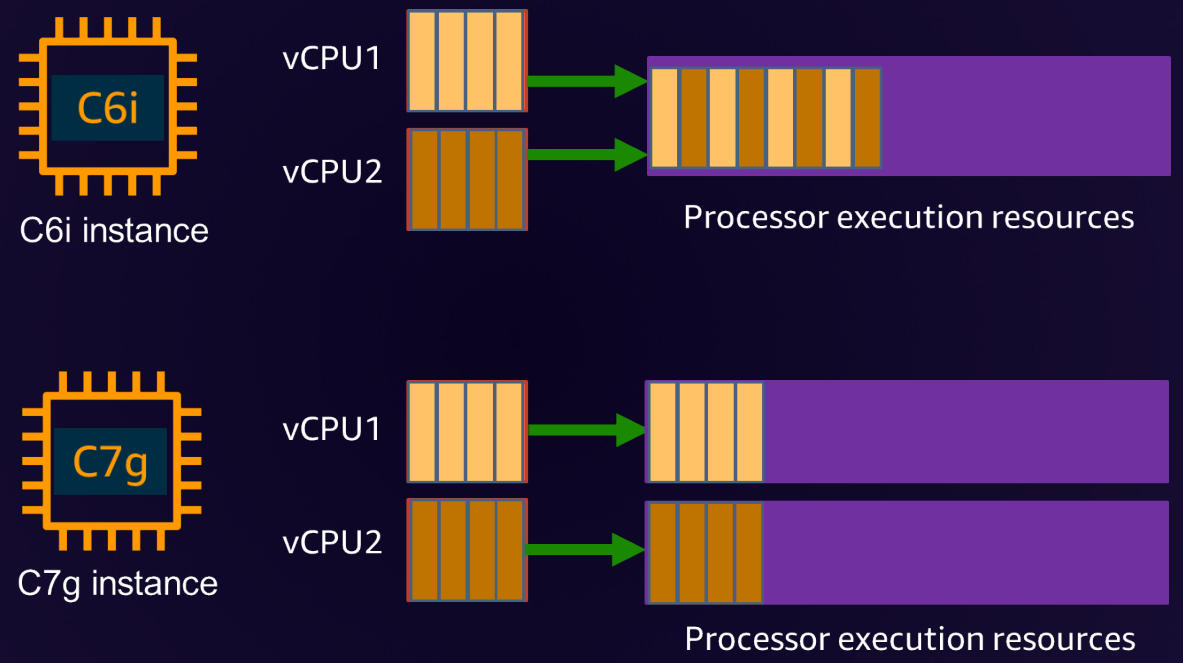
Leverage AWS Graviton

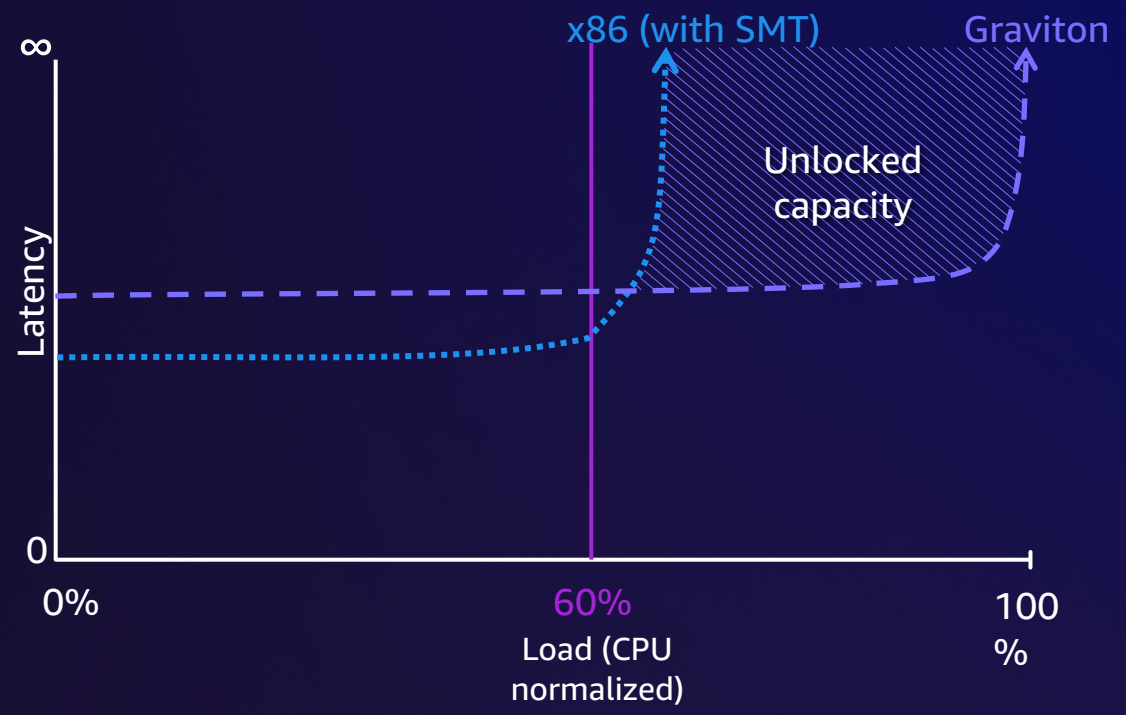
Best price-performance in Amazon EC2 for a broad array of workloads

Costs up to 20% less than comparable EC2 instances

Uses up to 60% less energy than comparable EC2 instances

Graviton (C7g) vs x86 (C6i)





Graviton:

- Every vCPU is a physical core
- No simultaneous multi threading (SMT)

* Graphs are approximations. Actual numbers would depend on workload.

MixedInstancesPolicy API Parameters – Multiple Launch Template Support

```
{
  "AutoScalingGroupName":"my-asg",
  “CapacityRebalance”: True
  "MixedInstancesPolicy":{
    "LaunchTemplate":{
      "LaunchTemplateSpecification":{
        "LaunchTemplateName":"my-launch-template-for-x86",
        "Version":"$Latest"
      },
      "Overrides":[
        {
          "InstanceType":"c6g.large",
          "LaunchTemplateSpecification": {
            "LaunchTemplateName": "my-launch-template-for-arm",
            "Version": "$Latest"
          }
        },
        {
          "InstanceType":"c5.large"
        },
        {
          "InstanceType":"c5a.large"
        }
      ]
    },
    "InstancesDistribution":{
      "OnDemandBaseCapacity": 10,
      "OnDemandPercentageAboveBaseCapacity": 50,
      "SpotAllocationStrategy": ”price-capacity-optimized"
    }
  },
  "MinSize":10,
  "MaxSize":50,
  "DesiredCapacity":15,
  "VPCZoneIdentifier":"subnet-5ea0c127,subnet-6194ea3b,subnet-
  a934b782",
  "Tags":[ ]
}
```

Graviton based EC2 Instance

Intel based EC2 Instance

AMD based EC2 Instance

ML Trends and Capacity Challenges

Incredible growth in new AI-enabled applications and customer experiences

Overwhelming demand for GPU capacity industry-wide

Customers face unpredictable lead times to acquire GPU capacity

EC2 Capacity Blocks

Supports P5.48xlarge, p4d.24xlarge instances

Dynamic pricing based on supply and demand

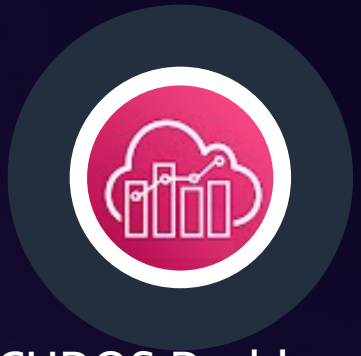
Reserve capacity from 12 hrs to 8 wks in the future

Block duration can range from 1 to 14 days

Cluster size 1, 2, 4, 8, 16, 32, or 64 instances

“If you can’t measure it, you can’t manage it.”

- Peter Drucker



CUDOS Dashboard

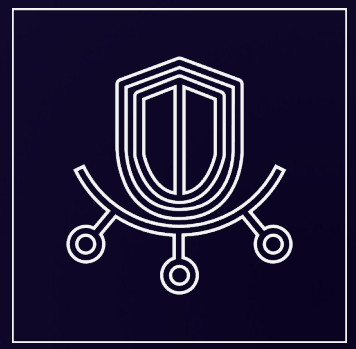
Cost Explorer

AWS X-Ray

Amazon Cloudwatch

Compute Optimizer

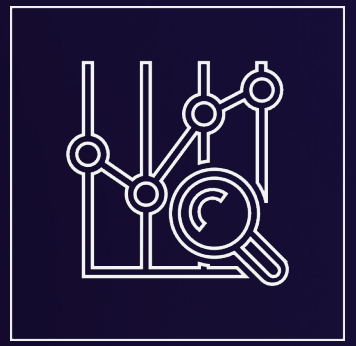
Compute Optimization Tools



AWS Trusted Advisor

High Utilization Amazon EC2 Instances Check

Low Utilization Amazon EC2 Instances Check



AWS Cost Explorer

Rightsizing recommendations

Downsizing within the same

EC2 instance family to save cost

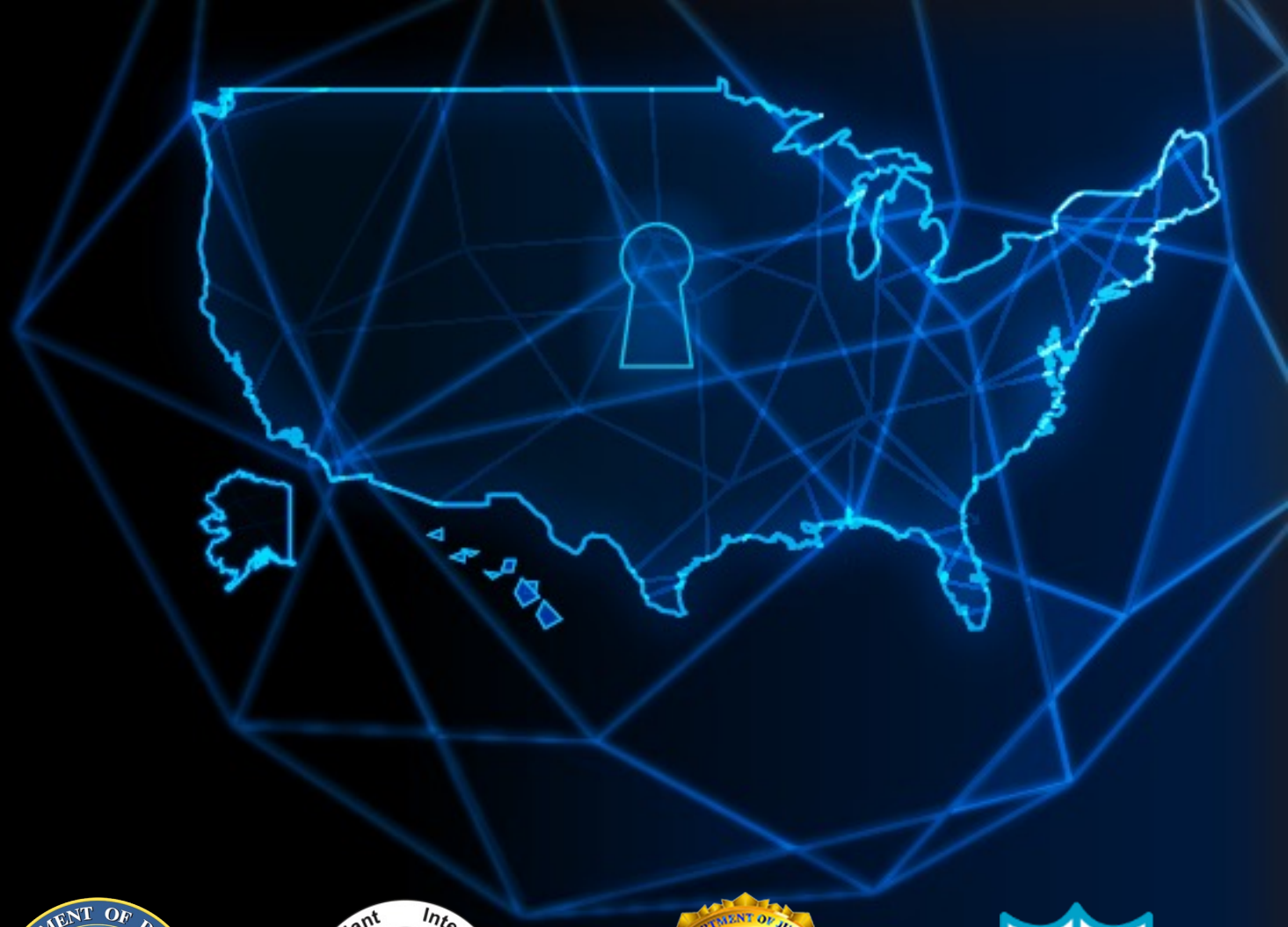
AWS Compute Optimizer

EC2 instance type recommendations for

Standalone EC2 instances and

Auto Scaling groups

AWS GovCloud (US)



Amazon's Regions designed to host sensitive data, regulated workloads, and address the most stringent U.S. government security and compliance requirements.

FedRAMP

Case Studies

Key Facts:

- Usecase: Understanding and measuring how a molecule's magnetic properties interact with different environments
- Tens of thousands of calculations
- Spot Instances as a Compute choice
- HTCondor pool and pushed the jobs into EC2 Spot instances

"From an end-user perspective, a single line was changed in an HTCondor job submission to make this solution available. It worked beautifully, and we were up and running in about two weeks," recalls Dr. Hood.

AWS Public Sector Blog

How researchers at The University of Manchester explore magnetic properties of molecules with the AWS Cloud

by Ray Rogers | on 03 FEB 2020 | in Amazon EC2, Education, Higher education, Public Sector | Permalink | Comments | Share

https://aws.amazon.com/blogs/publicsector/researchers-university-of-manchester-explore-magnetic-properties-cloud/

Use Case:
The IceCube experiment searches for ghost-like massless particles called neutrinos deep within the ice at the South Pole

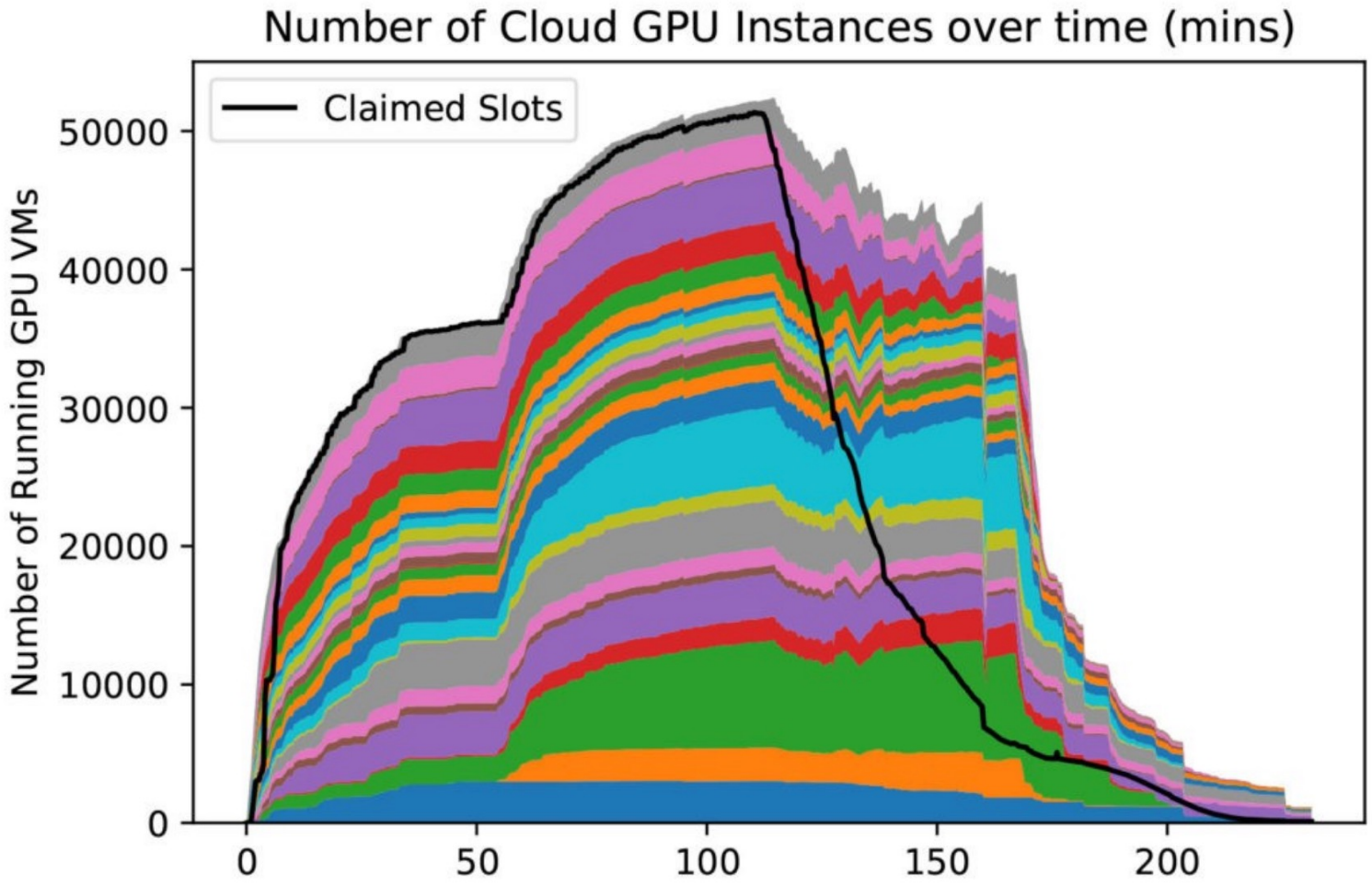
Key Facts:

- 51,500 cloud GPU's
- Both OD and Spot
- Multi-Cloud Setup
- HTCondor was used to integrate all GPUs into a single resource pool

AWS Public Sector Blog

AWS helps researchers study "messages" from the universe

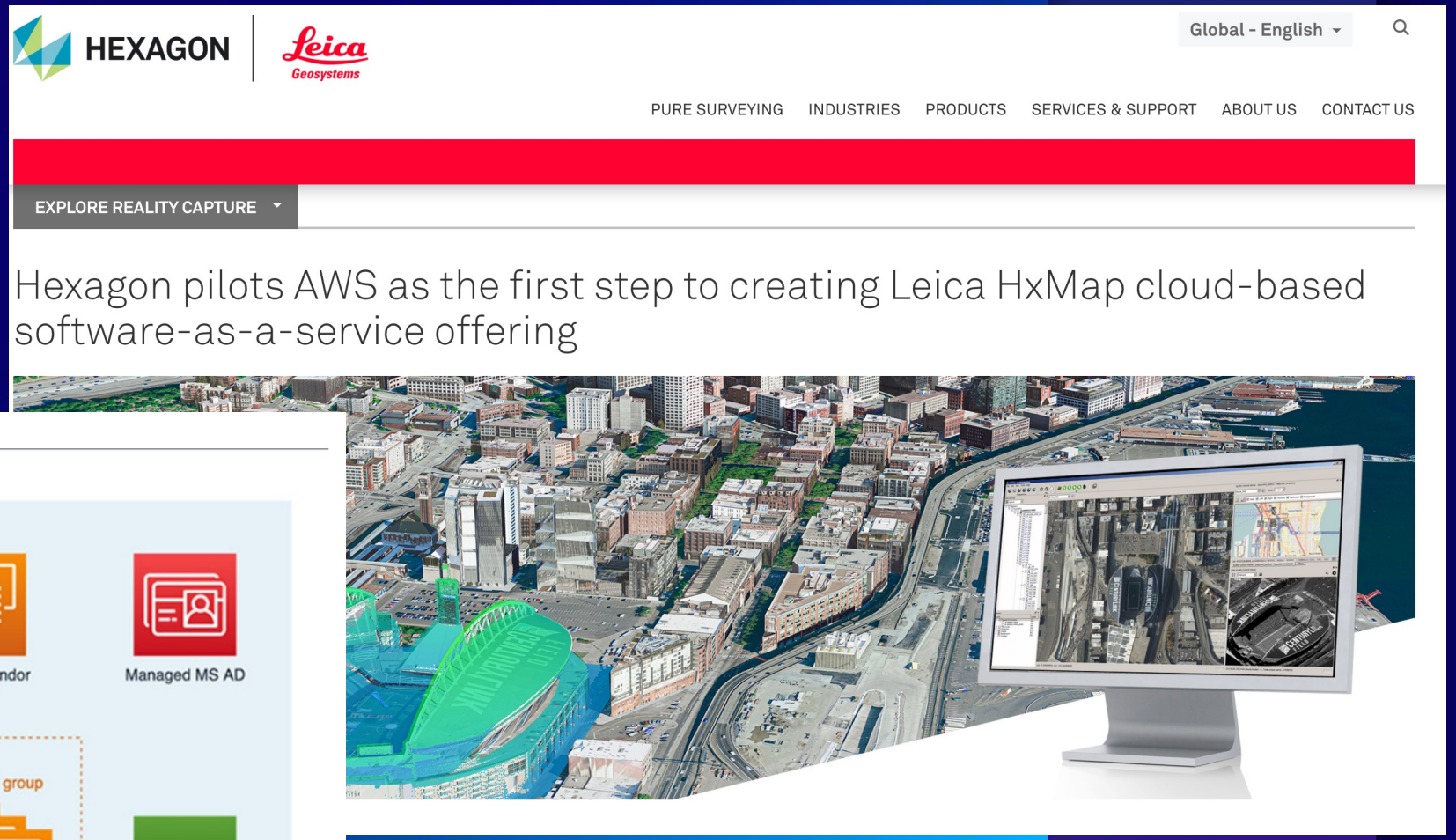
by Sanjay Padhi, Ph.D | on 26 NOV 2019 | in Public Sector, Research | Permalink | Comments | Share

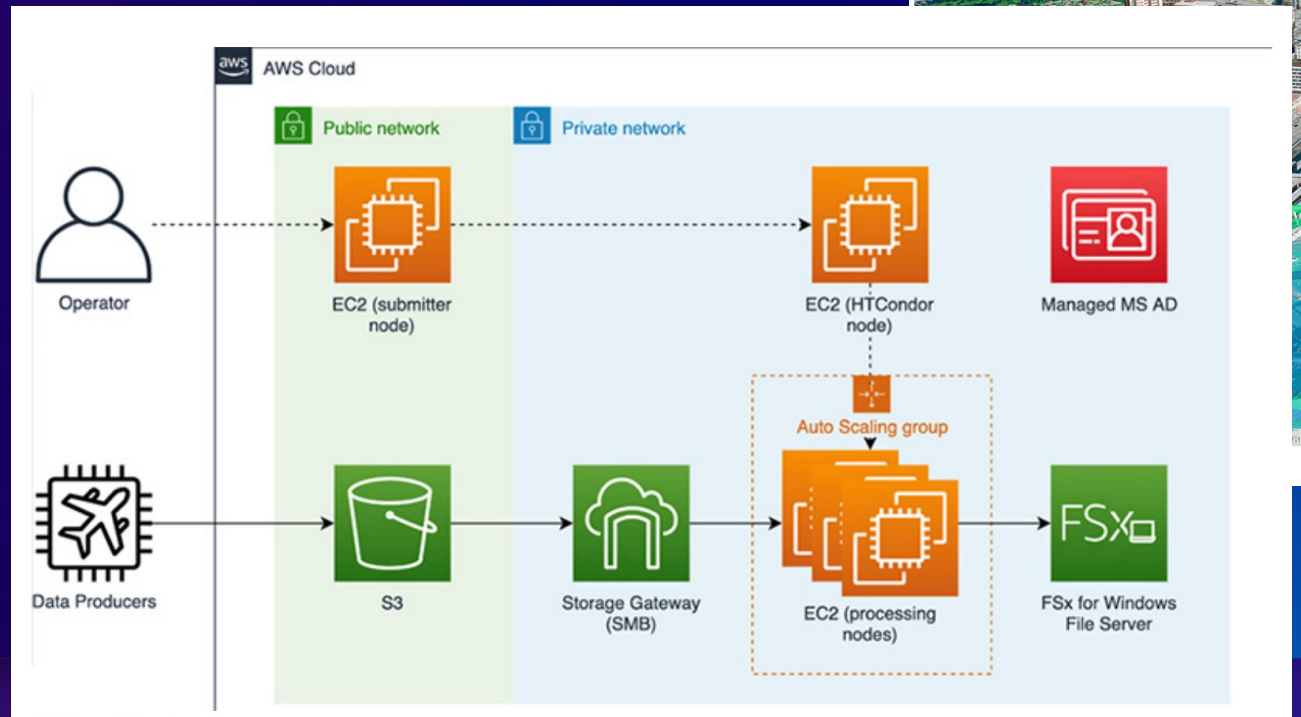


https://aws.amazon.com/blogs/publicsector/aws-helps-researchers-study-messages-from-the-universe/

Case Study: Leica HxMap

Use Case: multi-sensor software platform, that streamlines the processing workflow for all Leica Geosystems airborne sensors.





AWS architecture

https://leica-geosystems.com/case-studies/reality-capture/aws

Summary

- EC2 Purchasing options can be a cost effective way to scale Condor Workloads in AWS
- HTCondor Annex Supports Spot Fleet natively
- Flexibility is key for success with Spot instances
- Leverage Attribute Based Instance Selection for increasing flexibility
- Spot Placement Score can be a great way to determine the optimal EC2 Selections
- Graviton instances can provide cost and performance advantages
- GPU Workloads can benefit from Capacity Blocks Service
- Leverage Compute Optimizer and other tools for continuous right-sizing and optimization

Thank You

sudheenb@amazon.com
https://www.linkedin.com/in/ssbhat/

skillbuilder.aws

Build beyond

Redeem your free 7-day trial of AWS Skill Builder