



EDA Slurm Cluster on AWS

Slurm Users Group 2022

Allan Carter
Senior Specialist Solutions Architect, EDA
09/20/2022

EDA is a Highly Variable, Demanding Workload

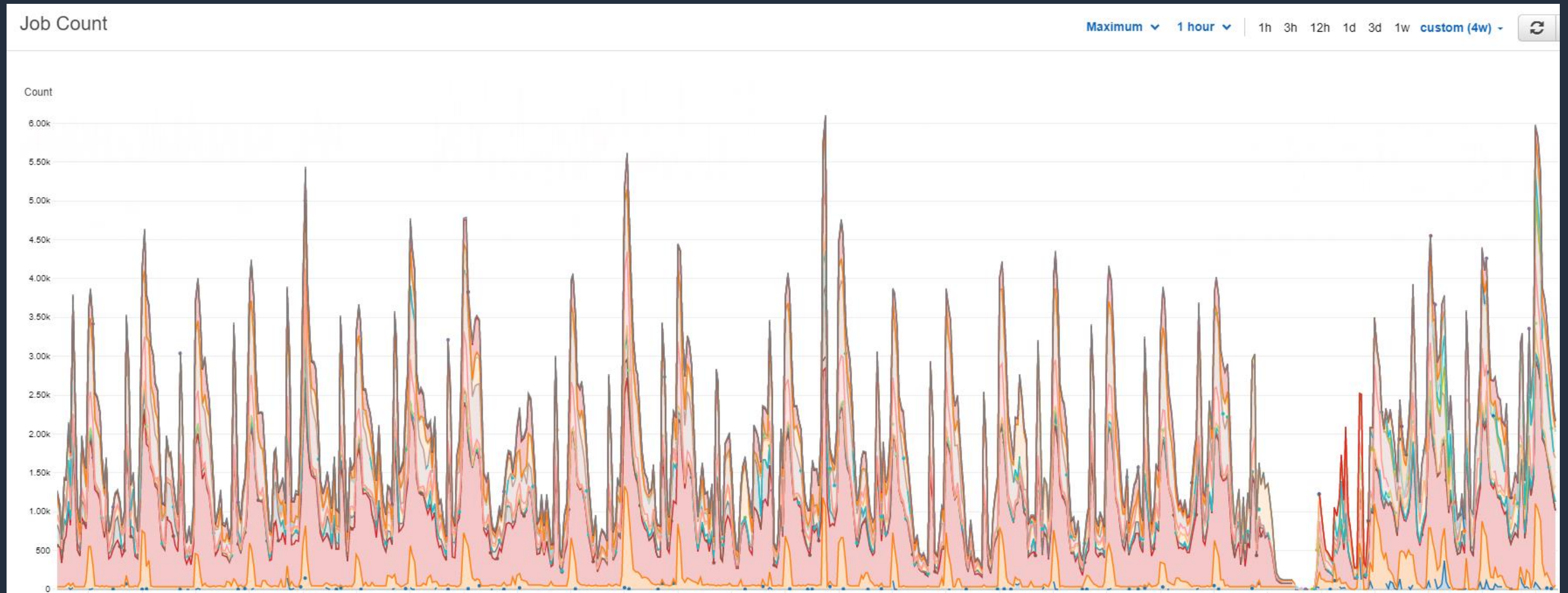


Table of contents

- Annapurna Background
- Key Requirements
- Overview of Slurm Compute Cluster
- Features
- Demo

Annapurna Background

- Annapurna Austin designs custom silicon for the AWS datacenters used in the F1, INF1, and TRN1 instances.
- From 2014 to 2020, Annapurna Austin used SGE and AWS Batch for their compute needs.
- SGE required manual scaling so resources never matched demand
- AWS Batch lacked features required by EDA workloads and simply couldn't run some EDA applications.
- A new solution was needed as the compute requirements grew in volume and complexity.

Key Requirements

- License-aware scheduling
- Fair-share scheduling
- Doesn't require containers
- Automatic scaling
- Instance based (no containers)
- High throughput
- High capacity
- Open source
- Security: minimize external dependencies so can run with minimal internet access

Why not ParallelCluster?

- No fair share
- No license management
- Limited queues
- Limited number of instance types
- No slurmdbd

Overview

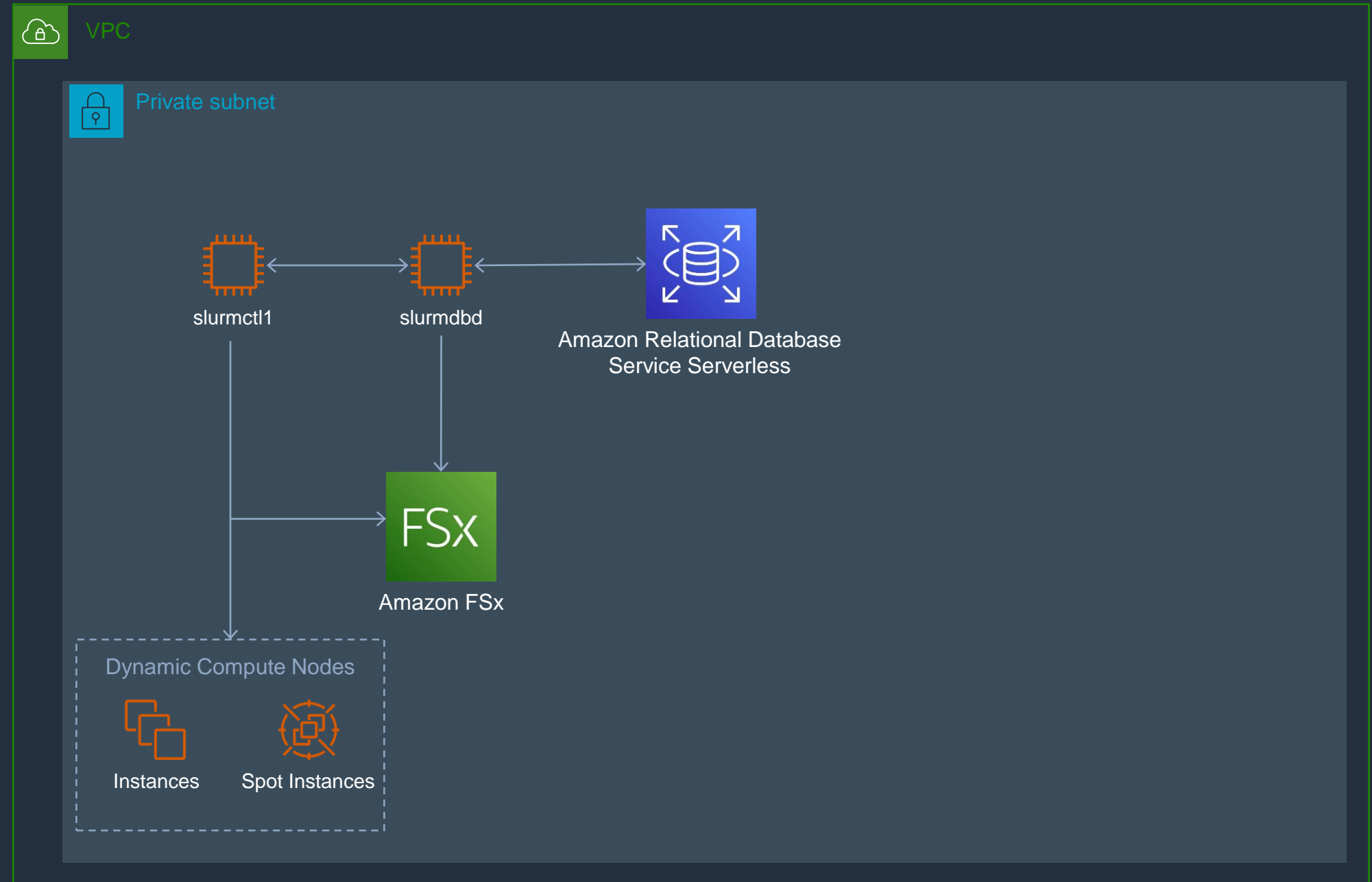
- In late 2019, Annapurna Austin started developing a Slurm plugin for AWS for its engineering and software development compute needs
- It went into production in April of 2020 and they have been using it successfully since then.
- The Annapurna solution has been enhanced and published on Github as [aws-samples/aws-eda-slurm-cluster](https://github.com/aws-samples/aws-eda-slurm-cluster)
- It is NOT an AWS solution or service
- None of the capabilities described are proprietary and can be used by any AWS customer

Github Repository

<https://github.com/aws-samples/aws-eda-slurm-cluster>



Architecture – Single AZ



Architecture – Multi-AZ



Features - Compute

- Instances can run in multiple AZs and regions
- Can manage on-premises servers
- Any EC2 instance type: x86_64, arm64
- RedHat/CentOS 7 & 8, Amazon Linux 2, Alma and Rocky Linux 8
- Spot instances (FIS allows testing spot terminations)
- Under 2 minutes cold start
- Instances stop after idle timeout so can restart faster
- Instances terminate after stopped for a period to save EBS costs
- Minimal IAM roles to allow metrics collection, monitoring, management.
- NVME SSD configured as /tmp and swap.

Features – Slurm Infrastructure

- Fixed Slurm infrastructure runs in 1 subnet/AZ
- Controller and SlurmDBD instances
- RDS serverless database behind SlurmDBD for job and license accounting
- HA for controller (up to 3 controllers)
- Shared file system (EFS, FSxN, FSxZ)
- Scales compute up and down based on demand
- Jobs run in cgroups to limit jobs to requested cores and memory.
- Always on instances can be configured
- On-premises servers can be configured
- Automatically drain and terminate instances running old AMIs.
- Federation (multi-cluster support)

Features - Scheduling

- License accounting
- Fair share allocation
- Prioritized queues for AZs which can span regions
- Scheduling done based on cores and memory
- Jobs can be constrained to available features: OS, CPU architecture, SSD, ondemand/spot, CPU frequency, instance family, instance type, etc.
- Automatic instance selection based on requested features and cost
- Interactive and batch queues
- Job preemption
- Accounting database

Features - Operations

- Completely automated deployment based on configuration using Cloud Development Kit (CDK) and Ansible
- Automatic AMI creation for combinations of OS and CPU architecture
- Detailed CloudWatch metrics and dashboard
- Automatic error notifications
- Can configure multiple clusters in the same VPCs
- Set default number of requested cores and memory

Demo

- How to deploy a cluster
- How to configure your VDE to use the cluster
- How to submit an interactive job
- How to submit a batch job