GPU Slicing on EKS

This documentation highlights a procedure to leverage GPU slicing on EKS with karpenter as autoscaler.

GPU Slicing or better called GPU sharing is the process of provisioning high intensive workloads to utilise the same GPU.

There are different types of GPU slicing but since the focus is on cost efficiency, time-slicing is recommended here.

Time-slicing involves dividing the GPU into small time intervals, allowing different tasks to use the GPU in these predefined slices. Time-slicing is suitable for environments with multiple tasks that need intermittent GPU access as well as unpredictable GPU demands.

When to consider GPU Time-slicing:

1. Multiple small-scale workloads
2. For development and testing environments
3. Batch processing
4. Real-time analytics
5. Simulations
6. Hybrid workloads
7. Cost efficiency

Procedure to implement GPU Time-slicing

1. Install Nvidia Device Plugin on the EKS: this plugins is essential for exposing GPU resources to kubernetes pods
2. Ensure that Karpenter recognises the GPU resource limit and request
3. Configure Nvidia plugin to allow multiple pods to share a single GPU. This can be achieved using a configmap
4. Define node templates and nodepools in karpenter to specify instance type and the requirements for GPU nodes
5. Integrate with Qovery that simplifies deployment and management of application on EKS with karpenter.