

IT & DATA SCIENCE

# Avoiding Fragmentation in your GPU Cluster using Run:ai



## Table of contents

|  |   |
|--|---|
| Introduction .....                                 | 2 |
| Understanding Fragmentation .....                  | 3 |
| Bin Packing: Addressing GPU Fragmentation .....    | 3 |
| Consolidation: Addressing Node Fragmentation ..... | 4 |
| Applicability to Batch Jobs .....                  | 5 |
| Time Limits for Interactive Workloads .....        | 5 |
| Idle Detection and Termination .....               | 5 |
| Considerations and Recommendations .....           | 5 |
| Conclusion .....                                   | 6 |



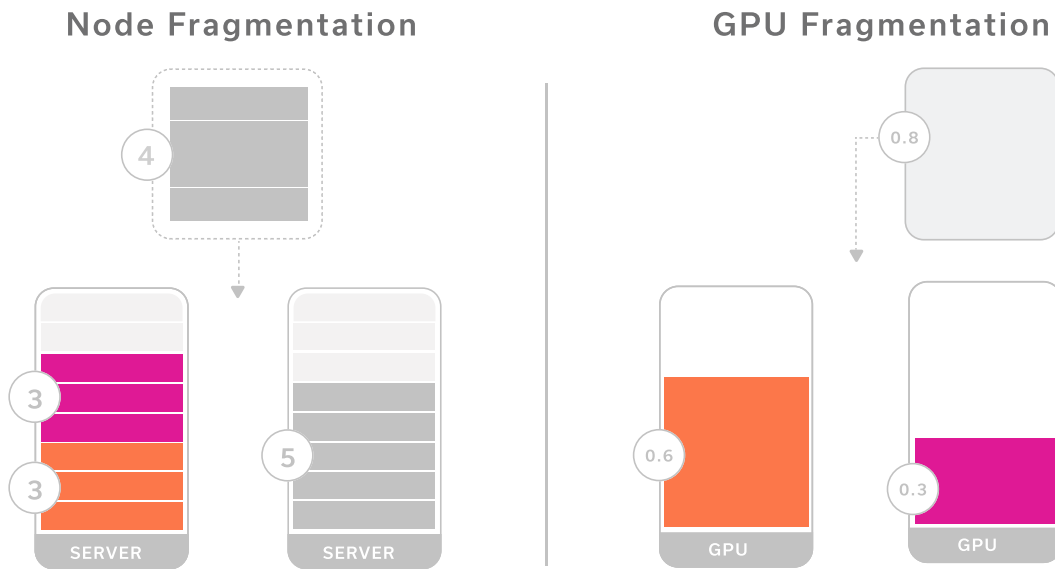
## Introduction

In a dynamic computing environment, managing workloads efficiently is crucial to maximize resource utilization and ensure smooth operations. One challenge that organizations often face is fragmentation, which can lead to inefficient resource allocation and reduced system performance. In this blog post, we will explore two methods, bin packing and consolidation, to address fragmentation in workload management. We will also highlight the difference between interactive workloads and batch jobs, emphasizing the applicability of bin packing and consolidation to batch jobs only.



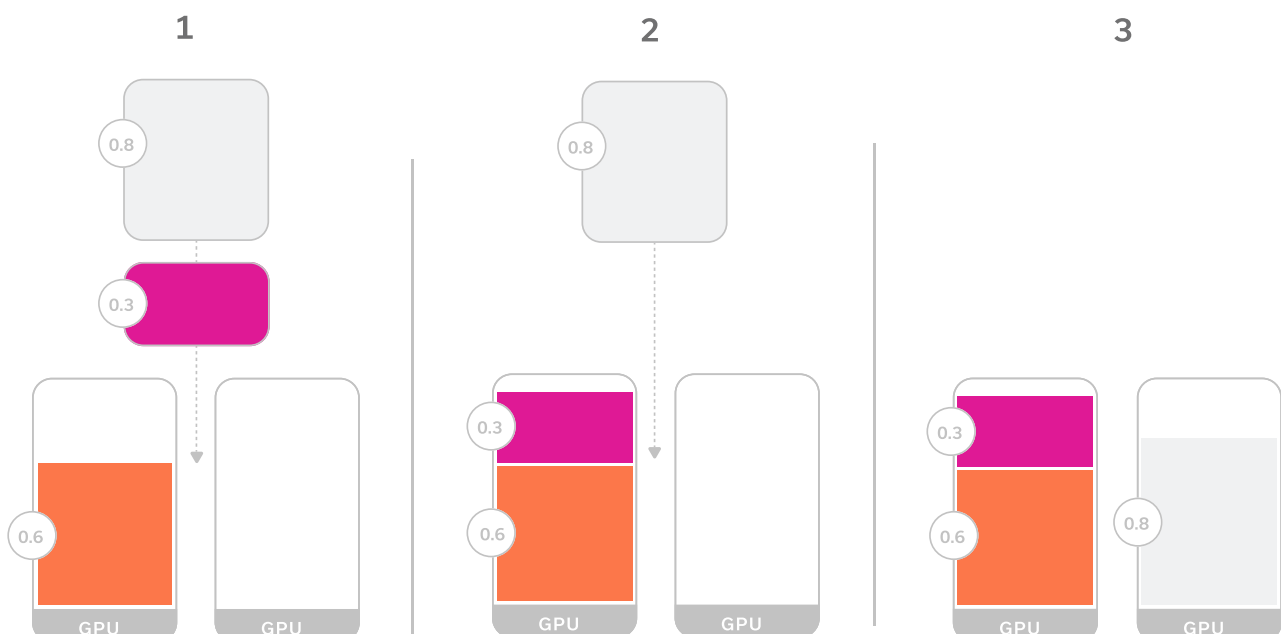
## Understanding Fragmentation

Fragmentation can occur in two forms: node fragmentation and GPU fragmentation. Node fragmentation refers to the situation where valuable resources in the cluster, such as GPUs, are not fully utilized due to the way tasks are allocated. GPU fragmentation arises when tasks allocate fractions of GPUs, leading to suboptimal utilization of available GPU resources.



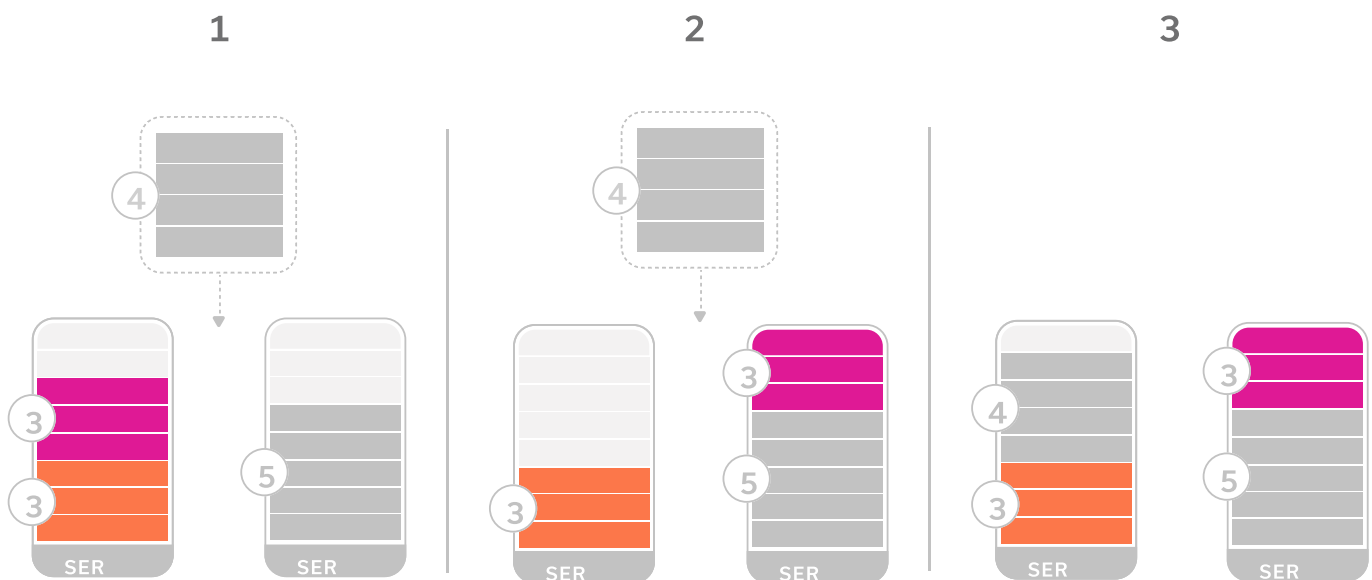
## Bin Packing: Addressing GPU Fragmentation

To combat GPU fragmentation, a technique called bin packing is employed. Bin packing aims to allocate tasks in a way that minimizes unused portions of GPUs. By assigning tasks with smaller GPU requirements to partially allocated GPUs, the available space can be utilized more effectively. This approach ensures that the maximum number of tasks can be accommodated within the available GPU resources.



## Consolidation: Addressing Node Fragmentation

While bin packing helps address GPU fragmentation, it may not always solve node fragmentation. Node fragmentation occurs when a task requires a larger number of GPUs than can be allocated on any single node. In such cases, consolidation comes into play. Consolidation involves identifying available resources on different nodes and preempting tasks from one node to allocate them on another node with sufficient resources. This strategy optimizes resource utilization by making space for new tasks on nodes that would otherwise remain underutilized.



## Applicability to Batch Jobs

It's important to note that bin packing and consolidation techniques are primarily applicable to batch jobs rather than interactive workloads. Batch jobs are typically scheduled and executed in a preemptible manner, meaning they can be interrupted and reallocated if necessary. In contrast, interactive workloads often require continuous and uninterrupted access to resources, making them non-preemptible. As a result, bin packing and consolidation are not applicable to interactive workloads due to their unique requirements.

---

## Time Limits for Interactive Workloads

To ensure that interactive workloads don't contribute to fragmentation, it's recommended to set time limits for their execution. By imposing reasonable time limits, such as 8 to 12 hours, interactive sessions can be terminated automatically, freeing up resources for other tasks. These time limits need to be aligned with the workday hours and strike a balance between allowing sufficient time for interactive work and preventing excessive fragmentation and idleness.

---


## Idle Detection and Termination

Another best practice for managing interactive workloads is to implement idle detection and termination mechanisms. By monitoring GPU usage, the system can detect when a workload is not actively utilizing the GPU for a certain period. In such cases, the workload can be terminated to release resources for other tasks. It's important to consider the specific use case and workload characteristics when setting the idle detection and termination thresholds, as certain tasks may have phases that require intermittent GPU usage.

---

## Considerations and Recommendations

While the suggested time limits and idle detection mechanisms serve as general guidelines, it's essential to tailor them to specific organizational requirements. Some workloads, such as distributed training, may benefit from different configurations and optimizations. For training workloads that involve CPU-intensive processes before transitioning to GPU utilization, longer idle detection periods may be necessary to avoid premature termination. Striking the right balance between termination and resource conservation is key to optimizing system performance.



## Conclusion

Fragmentation can significantly impact resource allocation in GPU clusters, leading to suboptimal utilization and performance. Through techniques like bin packing and consolidation, organizations can mitigate fragmentation and improve resource allocation efficiency. By implementing best practices such as setting time limits for interactive sessions and utilizing idleness detection and termination, clusters can be optimized for productivity and resource utilization. Ultimately, understanding fragmentation and adopting appropriate strategies will lead to enhanced performance and better utilization of GPU resources in clusters.

## About Run:ai

Run:ai is an AI management platform for MLOps, Data Science, and DevOps teams. In addition to helping these teams access and utilize their GPU resources more effectively, it also has a powerful set of features that can abstract infrastructure complexities and simplify the process of training and deploying models. With or without a GPU shortage, Run:ai enables data scientists to focus on innovation without having to worry about resource limitations.

Read more about how Run:ai supports  
data scientists here

[www.run.ai/runai-for-data-science](https://www.run.ai/runai-for-data-science)