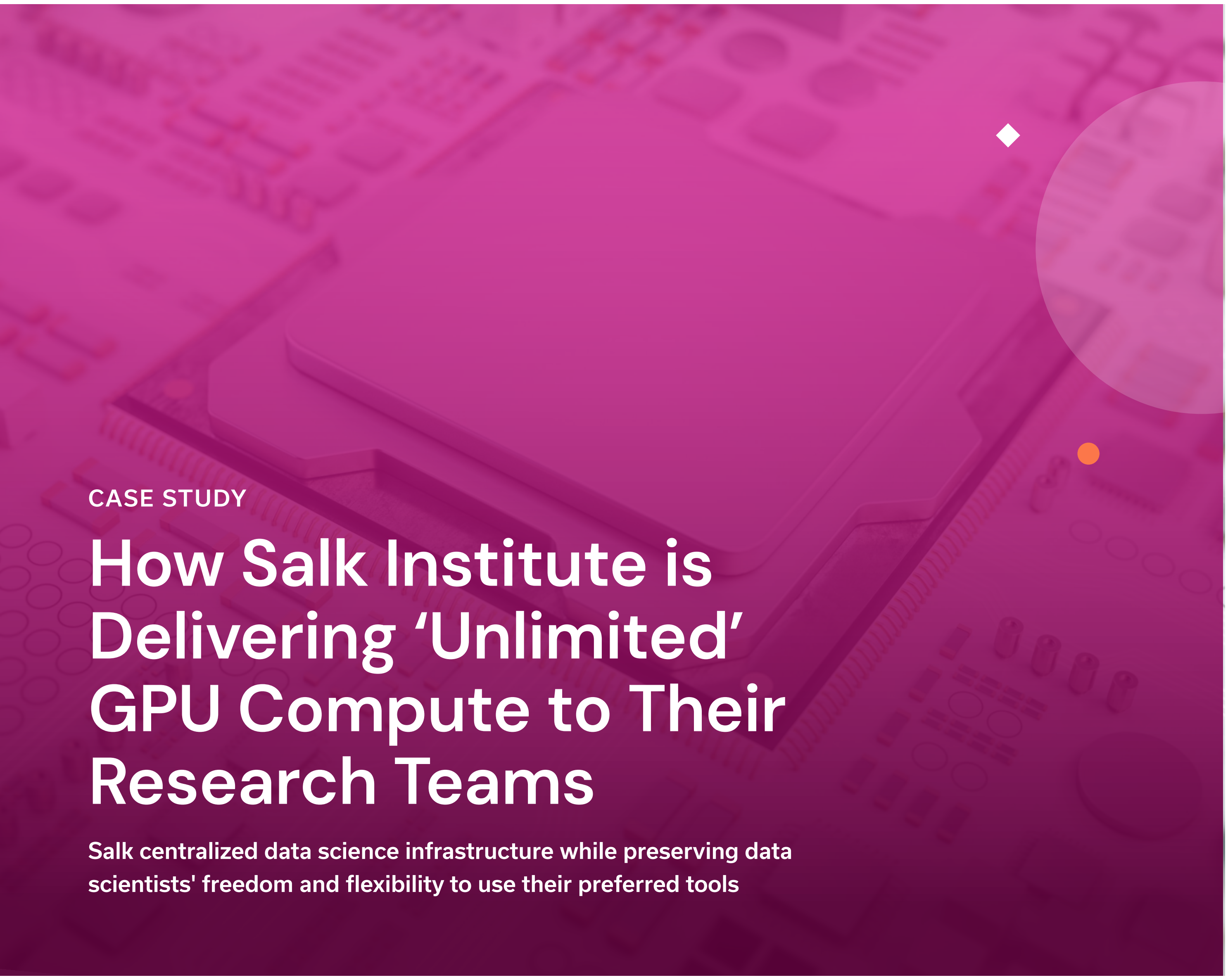


CASE STUDY

How Salk Institute is Delivering 'Unlimited' GPU Compute to Their Research Teams

Salk centralized data science infrastructure while preserving data scientists' freedom and flexibility to use their preferred tools

Institution

The Salk Institute for Biological Studies is one of the world's preeminent research institutions, where scientists are using artificial intelligence to make groundbreaking contributions in cancer and aging research, Alzheimer's, diabetes, and cardiovascular disorders

Challenge

The Salk Institute came to Run:ai with many of the same challenges that universities and research institutions share when it comes to efficient allocation of GPU resources. Siloed teams often purchase their own hardware, leaving IT without visibility into compute utilization or control of these independent systems. As teams begin to scale, the use of static quotas makes it difficult to manage researchers' variable compute demands, leading to frustrated researchers and limited ROI on the hardware. Anticipating these growing pains, Salk sought to centralize all of their GPU compute, for visibility into workloads and utilization metrics while ensuring fair allocation between all researchers.

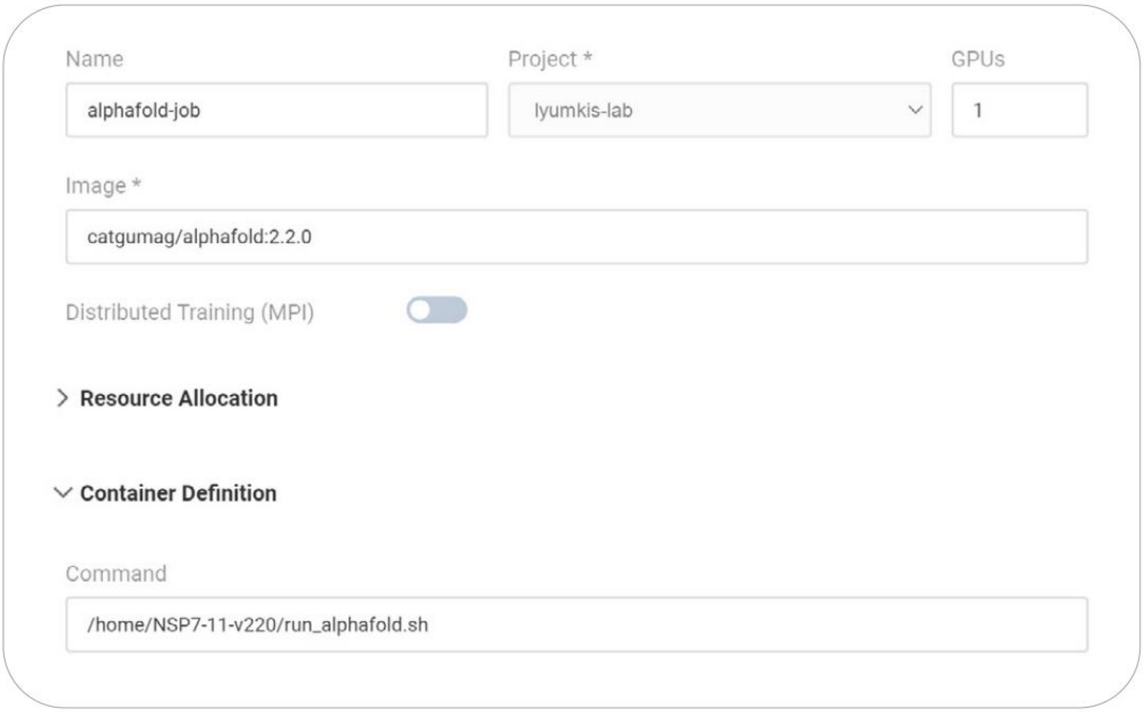
Solution – Advanced Scheduling With Run:ai

Run:ai unifies underlying compute hardware and infrastructure to drive changes that benefit IT as well as data scientists and researchers. Run:ai uses Kubernetes as its backbone to effectively orchestrate pooled GPU compute. By pooling GPU resources, teams of data scientists and researchers unlock significantly more compute power to run their workloads, limiting the time that GPUs sit idle.

To ensure fair access to GPU resources, Run:ai leveraged a scheduling concept called 'projects' to provide guaranteed quotas to the research teams. Salk was able to create multiple projects, enabling multiple different lab groups at the university to all have GPU access. In this way, the lab groups always have preferential access to their guaranteed quota of GPUs, and they can also burst beyond their quota and utilize additional GPUs that are sitting idle within the cluster. Run:ai also has capabilities around Node Affinity to allow for context-based scheduling, specifying that certain workloads run on specific GPU types, or affording preferential access to teams that have contributed their hardware to the cluster to run their workloads on their own hardware.

With enterprise-ready features such as single sign on (SSO), Run:ai helps IT quickly onboard and provision GPU resources for new researchers. The platform also delivers historical metrics which help IT understand the performance of the cluster over time, giving actionable data for capacity planning or building a business case for purchase of additional hardware.

Finally, the extensibility of the platform unlocks the potential for many workloads to run in parallel on a single cluster. Researchers at Salk are able to quickly spin up and launch GPU-accelerated Jupyter notebooks which creates a simple development environment for their AI models. Additionally, Run:ai enables Salk to leverage advanced research tools including Cryosparc and Alphafold with no integration hassles, as seen in the images below.



Name: alphafold-job
Project *: lyumkis-lab
GPUs: 1
Image *: catgumag/alphafold:2.2.0
Distributed Training (MPI): off

Command: /home/NSP7-11-v220/run_alphafold.sh

Figure 1. Salk's Jupyter notebook for Alphafold and the Run:ai job submission form.
They use the Jupyter notebook to create a bash script with all of the parameters that interest them.