



# Akash x DeepSquare

## Integration Whitepaper

### About Akash

*Akash Network is a decentralized, open-source cloud computing platform built on blockchain technology. It creates a peer-to-peer marketplace for cloud computing resources, allowing users to rent out unused computing power. Powered by the Akash Token (AKT), the platform operates on a decentralized network, ensuring greater security, flexibility, and scalability. Offering a range of cloud services, including computing power, storage, and database management, Akash Network aims to create a more efficient, secure, and decentralized cloud computing ecosystem, providing an alternative to traditional cloud service providers.*

### About DeepSquare

*DeepSquare presents a novel **Rent HPC by the Minutes** solution designed to meet the growing demand for computing power by businesses, researchers, and engineers.*

*Our vision is to deploy worldwide an Open Source standardized system, named Cluster Factory, capable of distributing data processing, for HPC and general purpose, across a network of local Cloud providers (DeepSquare Grid) by automating, optimizing and simplifying electronic interchange of information and workloads, with the objective to create a fair trading environment, which will reward suppliers' work, and provide a cost-effective and transparent data processing model to end customers.*

*The platform is designed with environmental sustainability, security, stability, and regulatory compliance in mind, making it an attractive choice for businesses in need of high-performance computing resources.*

## General Motivation and Objective

### [High-End Computing on Akash Distributed System]

*In Akash currently it is very convenient to rent a single or few GPUs from one provider. However, to rent tenths or hundreds of these GPUs, and orchestrate them to perform very complex workloads in coordinated fashion is still very hard and requires a huge amount of expertise and developer hours. The battle tested DeepSquare decentralized HPC(High-Performance Computing) system will help Akash alleviate this issue.*

### [Idle Resources Utilization]

*Task that usually only can run on High Performance Computing, usually are not time critical but requires to fill in multiple numbers of compute resources, SLURM job management system allows for job/task to wait and queue whenever resource not available and fill in all the empty resources as soon as it become available*

### [Cost & Complexity]

*Driving down cost and complexity of HPC usage and operation. Driving down cost and complexity enabling more adoption of parallel computation for retail users and smaller enterprise/organization use-cases. through the HPC-as-a-Service model. Currently smaller teams/startups may have the same computational need to solve big-problems, but don't have the resources to deploy HPC at the enterprise scale. Affordable decentralized HPC provides a solution to this situation. Moreover DeepSquare tech stack integration into Akash Network would allow for running parallel jobs and comprehensive task monitoring across the Akash Network ecosystem to be more manageable and efficient.*

*Below are the identified Akash Network challenges DeepSquare solution can Improve on :*

#### 1. Resource Management and Provider Stability:

- **Issue:** Ensuring deployments continue running during ISP (Internet Service Providers) dropouts and accurate API reflection of available resources.
- **DeepSquare Solution:** DeepSquare's architecture supports resilient deployment and robust resource tracking through its decentralized grid and reliable infrastructure monitoring, ensuring stability and accurate resource reporting.

## 2. Provider Bidding Criteria and Issues:

- **Issue:** Akash providers are not bidding on certain SDLs, specifically related to GPU requirements, while other providers can.
- **DeepSquare Solution:** DeepSquare's decentralized and transparent resource management can help in accurately matching provider capabilities with SDL requirements. By leveraging its open-source infrastructure and real-time resource tracking, DeepSquare can ensure more precise bidding and resource allocation.

## 3. Hardware and Resource Verification:

- **Issue:** Providers may have additional requirements like IP leases and persistent storage that prevent bidding.
- **DeepSquare Solution:** DeepSquare's infrastructure can integrate robust hardware and resource verification mechanisms, ensuring all providers meet the necessary criteria for SDLs before bidding. This can minimize mismatches and enhance bidding accuracy.

## 4. Direct Bidding Queries:

- **Issue:** Direct bidding using `dseq` is questioned due to the necessity of post-bid hardware checks.
- **DeepSquare Solution:** DeepSquare's comprehensive meta-scheduler optimizes job routing and resource checks pre-bid, enabling efficient direct bidding without compromising on resource verification.

## 5. SSL/TLS Certificate Deployment:

- **Issue:** Complexity in deploying SSL/TLS certificates and managing DNS, with a preference for Cloudflare over self-hosting.
- **DeepSquare Solution:** DeepSquare's open-source platform can include streamlined guides and integrated solutions for SSL/TLS deployment, reducing the complexity and providing clear DNS management processes.

DeepSquare's decentralized, transparent, and efficient HPC platform, coupled with its resource management solutions, can address many of the issues currently faced by the Akash Network community.

**By integrating these solutions, DeepSquare can provide enhanced bidding accuracy, resource verification, stability, and financial management, ultimately improving the overall user experience.**

## **Integration Roadmap, Milestones, and Deliverables**

### **Phase 1 DeepSquare to Integrate as Akash Provider and do RnD**

- Setup and Integrate Akash Provider Settings on DeepSquare HPC Cluster
- Preliminary study and workflow design for the DeepSquare Integration Module
- Research and test on the Compatibility and Performances of the Integrated DeepSquare x Akash System.
- Initiate the first two use-cases for the DeepSquare x Akash :
  1. “Large-scale human biomechanics and exoskeleton controller simulations”, In collaboration with The University of Fribourg in Switzerland
  2. “Massively Parallel Molecular Dynamics on the Akash Network Decentralized HPC Enabled Through DeepSquare: A Benchmarking Study”, In collaboration with DreamBrook Labs Pte. Ltd. Singapore.

### **Phase 2 Development of DeepSquare Integration Module for existing Akash Providers**

- Development of DeepSquare Deployment Module for Akash Providers.
- Tutorial and Tech Training on DeepSquare x Akash Integrated System.
- Providers Internal Testnet Recruitment mini-campaign

### **Phase 3 DeepSquare x Akash Blockchain Functionality Integration**

- Consideration of DeepSquare facility utilization via Akash Console interface.
- Message sync between DeepSquare and Akash Console interface, to allow displaying DeepSquare job info directly on Akash Console of the lease owner
- Tokenomics design/discussions [may as well do some tokenomics modeling]
- Reach out and promote to HPC user in Academia / DeSci ecosystem and/or Industry (for climate change modeling, advanced material simulation, for drug design, and more)

## Budget Request, Resource Requirements, and Disbursement Policy.

Total Budget Request of \*\*\$160,000\*\*, Composed of \$50,000 / Phase ( with different composition of Dev & Incentive allocation ) + \$10,000 extra reserve for volatility and liquidity buffer.

The engineering development budget is to be allocated for various tasks totalling up-to xxx /man hours over the whole project duration. While the rest of the budget will be for technical RnD purposes and community incentives. Details are given within the budget allocation table of the Integration Whitepaper. Project budget will be disbursed through the Multisig AKT wallet with the address : ...

### Phase 1 (08/2024 - 12/2024)

Item	Cost	Note
System Integration and Maintenance	25000	Rate : Developer Cost [System Architecture Dev. , Network Dev. ,Application Dev., Full-Stack Dev ] = 45 USD/hour  Community Management / Marketing = 30 USD/hour
2 DevOps / Cluster Supervisor	10000	1 DevOps specialised on Akash ( supervisor) 1 DevOps full time skilled on DeepSquare ( supervisor.) Total contributionhours = [xx Hours],
RnD Budget	10000	To produce a research report on integrating DeepSquare Tech Stack with Akash. Including Hourly Tech Consultation with Experienced Akash Provider, DeepSquare, and External Subject Experts. Budget to cover both Phase I and II.
Integration Team Compensation	5000	Will be detailed on the upcoming SoW for contributions, but overall budget estimate would need to be given here.
<b>Total</b>	(Estimate) ~ \$50,000	30,000CHF (~\$ 33,853USD) + RnD fee (expecting ~\$15,000-20,000)

**Phase 2 (01/2025 - 04/2025)**

Item	Cost	Note
<b>Total</b>	(Estimate) ~ \$50,000	Phase 2 budget: 50,000 CHF (~\$56,422 USD) + Provider incentive campaign for those participating (amount tbd, expect 32-48 GPUs will be nice, maybe about \$500 each GPU installed, unspent will be returned, + campaign administration fee maybe about ~8k , total of up to \$32,000 )

**Phase 3 (05/2025 - 09/2025)**

Item	Cost	Note
<b>Total</b>	(Estimate) ~ \$50,000	Phase 3 budget: 20,000CHF (~\$22,568 USD) + Tokenomics Research and Operation Budget

## Integration Team: Role and Detailed Responsibilities

### Akash

**Domi:** (1) Coordination between DeepSquare x Akash ; (2) Learn and assist with setting up / maintain part of the DeepSquare component deployed on Akash Network ; (3) Work on Use-Cases level development and system testing for HPC Workflow Deployment.

**Phase 2 and 3 Campaign Manager :** *Community campaign (recruiting provider, cluster manager, recruiting users), will be recruited from Akash Insider / Vanguard community as basic infrastructure of the integration has been worked out.*

### DeepSquare

**Luca:** 1) Coordinate DeepSquare internal resources to deliver project deadlines 2) Shape commercial proposal for partnership with Akash 3) Coordinate efforts/engagements with Domi

**Florin:** 1) Coordinate PoC development for August-September deadline 2) Develop value proposition with Luca 3) Identify with Luca Budget allocation

**Charley:** 1) Go to person for internal documentation ( technical, marketing, Financial)

**Lillo:** 1) Infrastructure HW expert 2) Integration resource

**Marc:** 1) Cluster Factory deployment/installation 2) Integration resource

**Maya:** 1) Marketing 2) Community communication

### Contact Person:

Florin, [florin@deepsquare.io](mailto:florin@deepsquare.io) (PoC project coordinator)

Charly, [charly@deepsquare.io](mailto:charly@deepsquare.io) (Collateral and documentation coordinator)

Luca, [luca@deepsquare.io](mailto:luca@deepsquare.io) (General DeepSquare Akash Project lead coordinator)

Dominikus Brian : [domi@dreambrooklabs.com](mailto:domi@dreambrooklabs.com) [Akash Insider, DreamBrook Labs] spot for [Akash Network -- Core Team / Provider Representative]

## DeepSquare Tech Stack and Core Capabilities

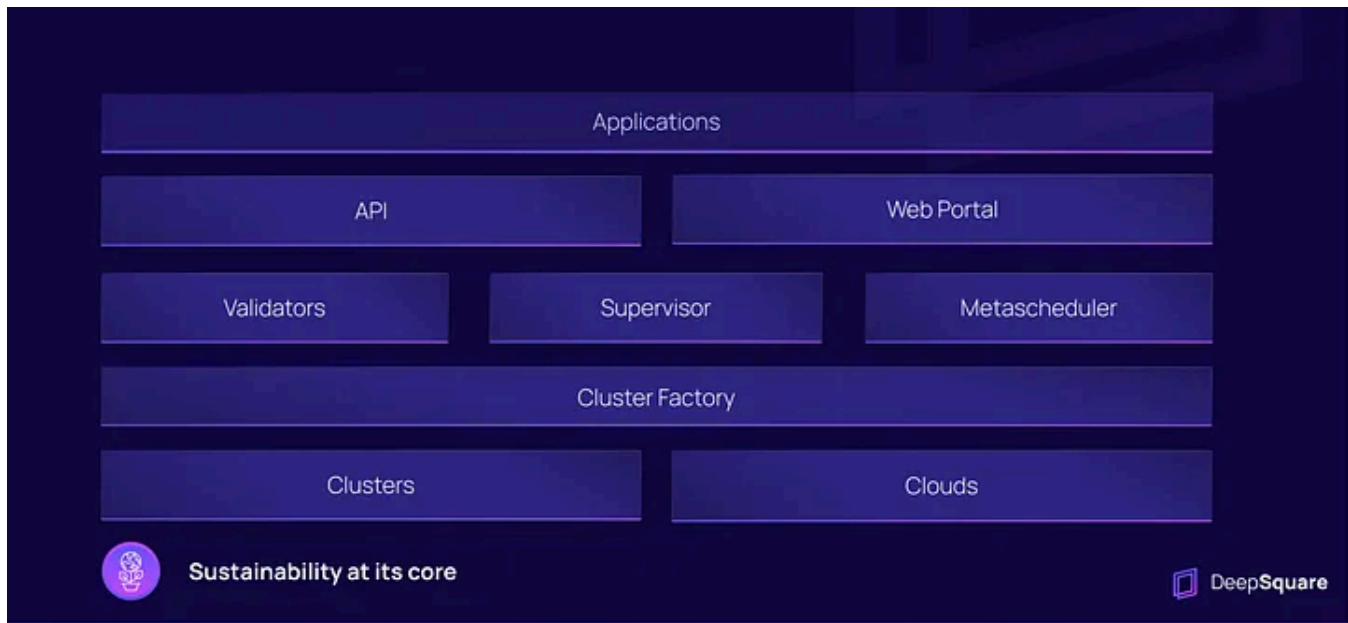


Figure 1. Architectural Overview of the DeepSquare Grid.

**DeepSquare Grid** aggregates the unique capabilities of supercomputers from diverse compute providers worldwide into a unified computational infrastructure. Our primary aim is to democratize HPC, and the Grid stands as a powerful embodiment of this commitment. DeepSquare operates with a user-centric approach to resource allocation. Users specify the computational requirements for their workloads, and DeepSquare, through its Meta-scheduling process, matches these workloads to the most appropriate compute provider available on the grid.

**ClusterFactory (Open Source)**, the Kubernetes-based infrastructure orchestrator together with the community. ClusterFactory brings together best-in-class solutions from the HPC, Cloud, and DevOps industries to manage a cluster in a declarative way in combination with the GitOps practice.

ClusterFactory makes the process of deploying a full fledged HPC cluster and joining the DeepSquare Grid fast and easy. We believe that flexibility, repeatability, availability and ease of use should be prioritized for managing and scaling HPC clusters.



*ClusterFactory has been developed to be:*

- *Performance-oriented: Integrates a key-in-hand HPC stack including Slurm, MPI, DFS, etc.*
- *Highly configurable: With Helm, all configuration is done in a single `values.yaml` file.*
- *Repeatable: With Argo CD following GitOps practices, all states are specified declaratively and saved in a Git repository.*
- *Highly available: With Kubernetes, container scheduling is automatically ensured and easy to set up.*
- *Simple: A single descriptive YAML per application, with Argo CD to automatically updates the application.*
- *Long-term maintainability: Easy to deploy, update, backup and restore with KOs.*

### [DeepSquare Metascheduling capabilities](#)

*Optimizing job distribution with meta-scheduling*

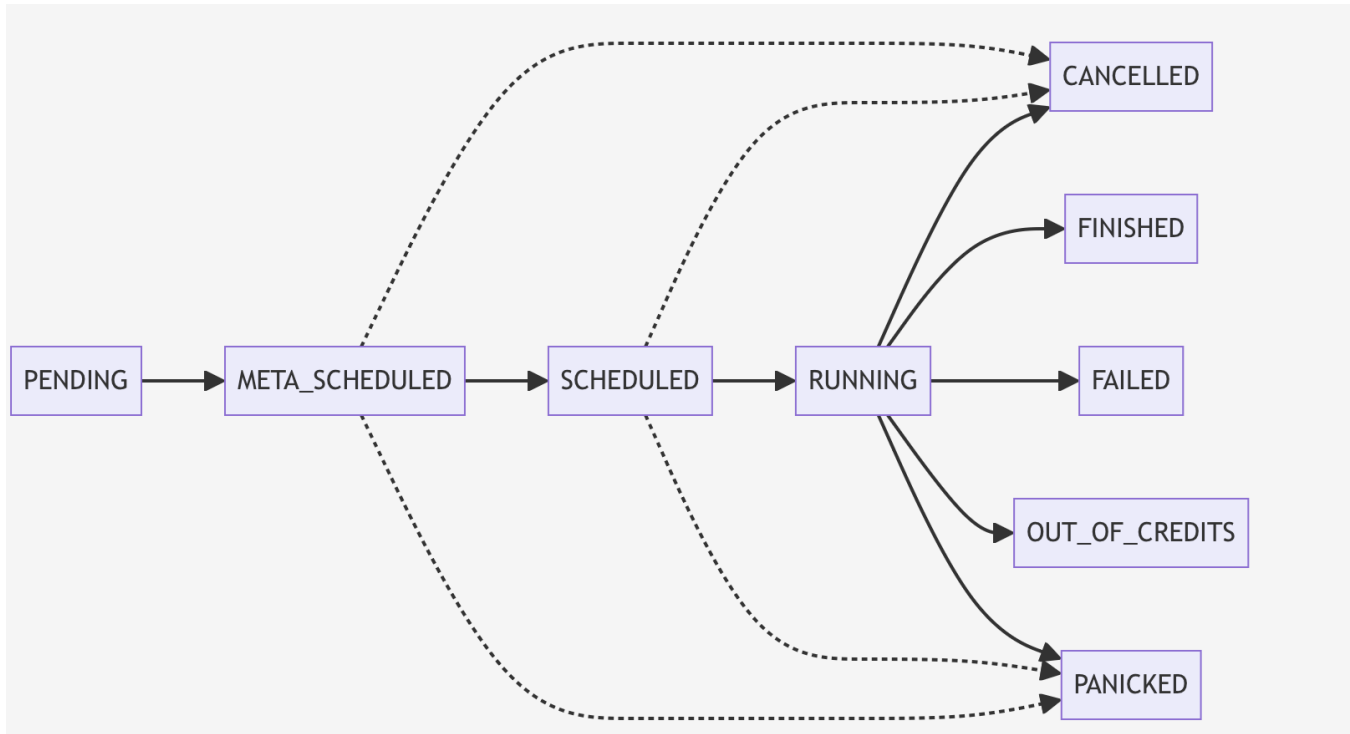
*Meta-scheduling is a strategic technique designed to optimize the scheduling of tasks across multiple distributed computing resources. It operates by employing a higher-level scheduler to oversee lower-level schedulers, which handle the allocation of resources and scheduling of jobs on individual machines or clusters. This approach enhances the efficiency of job distribution and resource utilization in complex computing environments.*

*Meta-scheduling brings about significant advantages in optimizing distributed computing environments. By coordinating task allocation across multiple machines or clusters, it improves resource utilization, minimizing idle time and enhancing overall system efficiency. The higher-level oversight provided by the meta-scheduler enables dynamic adaptation to varying workloads, ensuring effective load balancing and responsiveness to changing computational demands. Lastly, the technique enhances fault tolerance by intelligently redistributing tasks in the event of node failures, contributing to increased system reliability in complex computing environments.*

*Predictive job life cycle*

*In the DeepSquare ecosystem, each job is assigned a status that represents its current state. A job is treated with a Finite State Machine, which allows predictive behavior in the whole system.*

The following are the different types of job statuses:



- **PENDING:** A job has been submitted by a user and is awaiting meta-scheduling.
- **META\_SCHEDULED:** The job is assigned to a cluster.
- **SCHEDULED:** The job has been queued internally by the cluster.
- **RUNNING:** The job is being executed.
- **FINISHED:** The job has ended successfully.
- **FAILED:** The job has ended with a non-zero error code.
- **OUT\_OF\_CREDITS:** The job has ended due to the time limit/credit allocation.
- **CANCELED:** The job has been canceled by the user.
- **PANICKED:** The job is in an undefined behavior and has been killed.

### DeepSquare Validators and Token System

The platform utilizes two tokens, GFL and DPS, to standardize computing resources and monetise them for industrial clients and Grid Partners. The GFL token ensures optimal allocation of computing power, while the DPS token allows businesses to access computing

resources without owning the infrastructure. The platform's halving mechanism adjusts the token supply, ensuring the sustainability and stability of the ecosystem. Over the first 10 years, Grid Partners are rewarded through inflation, enabling DeepSquare to offer competitive pricing to clients.

## **Requirements and recommendations for System set up**

**All documentation on how to deploy cluster factory are here:**

<https://docs.clusterfactory.io/docs/overview/welcome>

### **Node requirements**

All nodes should be accessible via SSH.

All nodes should have a Linux distribution with:

- The Linux kernel version is higher than 3.10.
- An init system based on SystemD or OpenRC.

ClusterFactory has been fully tested on Rocky Linux which is our recommended OS.

### **Required utilities**

- [cfctl](#), for deployment, backing up, and upgrading of the Kubernetes cluster.
- [kubect1](#), for managing your Kubernetes cluster.
- [kubesea1](#), for encrypting the secrets.
- [helm](#), for Helm chart template.

There is a script inside the [scripts](#) directory to install and set up a working environment.

Just run:

```
../scripts/setup-env
```

The binaries are stored inside the [bin](#) directory and the [PATH](#) is automatically set.

## **Recommended tools**

We recommend:

- [VSCode](#). Any IDE with YAML support is a good alternative.
- [Lens](#) to manage your Kubernetes cluster.

## **Recommended documentation**

- [Kubernetes documentation](#)
- [cfctl.yaml API reference](#)
- [Argo CD declarative setup](#)

Before using ClusterFactory, it is strongly advised to have a comprehensive understanding of how Kubernetes operates, specifically with regard to storage and network management using features such as PersistentVolume, StorageClass, Service, Ingresses, LoadBalancer, and more."

To try a "mini" version of Kubernetes we recommend k0s or minikube.

## **Two different scenarios for Cluster Factory deployment**

Assuming we are able to share the same OS image on the compute nodes with all required tools for both DeepSquare and Qarnot.

List of tools needed on the compute node's OS image:

- Linux Kernel > 5.0 (any OS glibc based)
- NVIDIA driver
- [Enroot](#) hardened and [Apptainer](#) (software for containers)
- CVMFS client
- Slurm daemon, MUNGE (auth), SSSD (identity)
- Various software for users:
  - curl wget bash tar unzip xz screen squashfs-tools cpio

- *wireguard-tools*
- *pmix4* (compiled with *hwloc* ? maybe *ucx* too ?)
- *s3cmd*, (*s4cmd* ?) and *s5cmd*
- *spank-provider* (RPM: <https://yum.deepsquare.run/yum.repo>, if DEB we'll see later)
- *slirp4netns*
- *grepcidr*
- *git-lfs*
- (maybe more...)
- *Various postbootscripts:*
  - *CVMFS mount*
  - *Install Grid various software* (*bore*, *logger*)
  - *SSSD* (LDAP client) configuration
  - *LDAP Certificate Authority* configuration
- *Storage layout:*
  - *For users:*
    - */opt/cache/shared* (network shared, perms 777 (world))
    - */opt/cache/persistent* (network shared, perms 777 (world))
    - */opt/cache/world-tmp* (network shared, perms 777 (world))
    - */opt/cache/disk/tmp* (on disk, perms 777 (world))
    - */opt/cache/disk/world-tmp* (on disk, perms 777 (world))
  - *In order to cache container images:*
    - *{/mnt/disk}/scratch/tmp* (on disk, perms 777 (world)) (path can be configured)

## **1. Off site control plane (Preferred scenario, simpler)**

- *VPN* (*Wireguard*, server-client setup) on each compute node or on 1 router (dedicated VM)

→ *In this scenario, all the DeepSquare required services are hosted on the cloud*

## **2. On site control plane**

- *Forwarded ports:* 4789/udp (*Kubernetes tunnel/VXLAN*), 8132/tcp (*Konnectivity, kubernetes worker-controller*), 10250/tcp (*Kubelet/Worker API*)
  - Or use *Wireguard VPN*.

- *K8s with the following services*
  - *Slurm controller*
  - *LDAP*
  - *Traefik*
  - *CoreDNS*
  - *SLURM Login node*
  - *Supervisor + LDAP connector (Connectors to DeepSquare Grid)*
  - *...*

## **2. Developer & System Management Work - Developer works**

- **System Architecture Dev-**

*DeepSquare's global architecture is composed of three layers:*

- *The User Applications*
- *The Meta-Scheduler Smart-Contracts*
- *The Clusters*

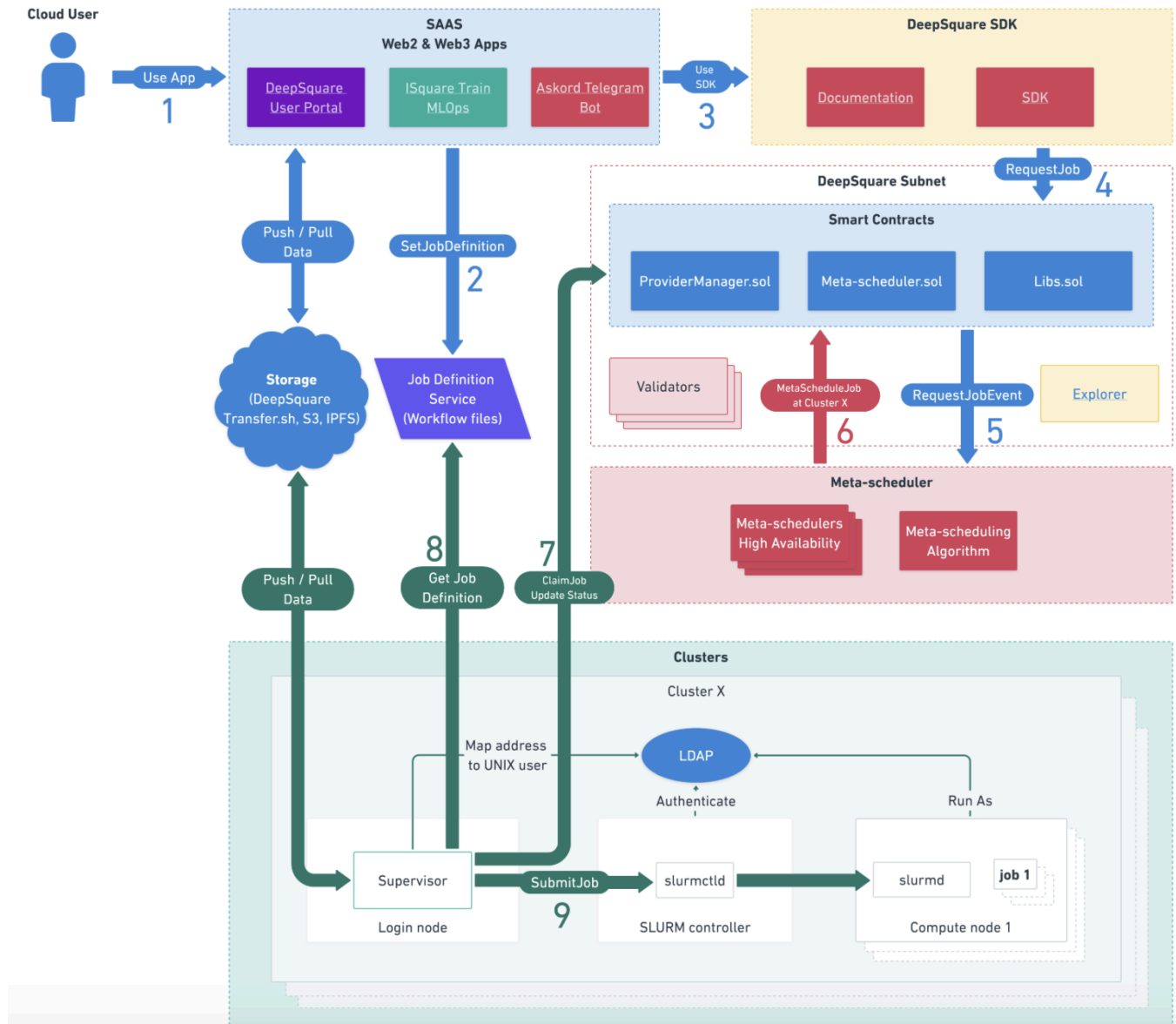


Figure 2. Modular Workflow Pipeline across the DeepSquare Grid Tech Architecture.

you can click on this link for more details:

<https://docs.deepsquare.run/deepsquare-grid/learn/architecture>

- Network Dev. ★
- Application Dev.

We have developed a simple Application to launch workflows, more details here <https://docs.deepsquare.run/workflow/quick-start>

- **ClusterFactory Supervisor Role**

The Supervisor is the link between DeepSquare and the Infrastructure Provider. This guide will help you understand how the Supervisor works and how you can edit its behavior.

## **The Dependencies**

The Supervisor depends on three components:

- The meta-scheduler smart-contract
- The sbatch service (Job Definition Service)
- The SLURM Login

To link the supervisor to the meta-scheduler smart contract, you can click on this link for more details:

<https://docs.deepsquare.run/deepsquare-grid/learn/supervisor>

## **Incentive Program/Package for Recruiting Existing Akash Providers to join the DeepSquare Integration Internal Testnet**

While in the beginning we will start with testing the setup, in the long term Akash Provider's that satisfy the minimum criteria for providing DeepSquare Decentralized HPC support could earn extra earning, simply by integrating the DeepSquare add-ons on top of their existing Akash Provider setup.

To initiate this process, we would need to incentivize the Akash Providers that decided to join and participate in the testnet integration. The program will be available in two modes: Active and Passive.

### **Active Mode:**

- Provider provides a number of dedicated resources AND actively contributes in the setup of DeepSquare add-ons on Akash and maintenance of the integrated system. In



*total we estimate extra commitment of x hours/month will be required to ensure smooth operation of the integrated system.*

*Passive Mode:*

- *Providers simply provide a number of dedicated resources WHILE, the setup of DeepSquare add-ons on Akash and maintenance of the integrated system will be performed by dedicated ClusterFactory Supervisor on an as-needed basis. The Provider should be providing the required access for the Supervisor to the dedicated resources.*

*Ideal Target Providers to join the initiative*

*From the 5-6 possible provider partners (shown below) the test ecosystem can amass about more than 100 GPUs and 1000CPUs for the operation, perhaps starting*

**DeepSquare providers Hardware requirements**

*All relevant hardware is prepared for immersion cooling (fans removed, compatible thermal paste and heat-sinks applied).*

*Configuration excludes immersion cooling tank which can be acquired/rented separately from suppliers such as Submer or GRC.*

- *5x compute nodes (4.879,-€ x 5 = 19.516,-€)*
  - *AMD Ryzen 16 core CPU*
  - *128 DDR4 RAM*
  - *512 GB SSD*
  - *2x 24GB nVIDIA GPU*
- *1x master node (2.150,-€)*
  - *AMD Ryzen 24 core CPU*
  - *256 GB DDR4 RAM*
  - *512 GB SSD*
- *1x storage node (1.912,-€)*
  - *AMD Ryzen 16 core CPU*
  - *128 DDR4 RAM*
  - *2x 3 TB SSD*
- *1x 10 Gb Switch (1.930,-€)*
- *1x PDU (1.047,-€)*
- *Various Cables & Connectors (~500,-€)*

*Currently this section is for internal use. We will decide which Q & A Pairs and supplementary info will actually be in the Integration Whitepaper.*

### **FAQ & Common Concerns**

1. If something goes wrong or down in between the Akash Deployment Lease, and the User Workload ? Who will handle the troubleshooting and/or maintenance ?
2. The owner of all deployment lease under the DeepSquare Network across Akash Providers will be single/multiple accounts owned by the DeepSquare team ?
3. Functionality proposed before for Akash Network, that maybe can be partially covered by DeepSquare tech Stack. We need to check on this feasibility and perhaps mention in the section where we mentioned identified Akash Issue  
<https://github.com/orgs/akash-network/discussions/632>  
<https://github.com/orgs/akash-network/discussions/534>  
<https://github.com/orgs/akash-network/discussions/621>  
<https://github.com/orgs/akash-network/discussions/372>
4. Just in case this is asked, Based on the explained positioning of AKT and DPS , also GFL, will the upcoming launch or future launch of DPS be any influence to Akash Ecosystem ? [we simply can provide simple answer such as No, and they are independent, DPS not used for payment to Akash Provider network, etc. Then post link to yellow paper or something like that]
5. There are also concern from Akash Validators and community , will the inclusion of Akash Provider in serving DeepSquare be causing they might in the future to put less resource on Akash ?

*– Integration Whitepaper ends here –*

***Supplementary Information and Notes [For internal use]***

DeepSquare Sion Cluster HW spec

1 master node

6 compute nodes

Each compute nodes has

- 2x Nvidia 3090 (RTX3090)

- 128GB RAM

- 2 x Xeon 16 cores

- RAM (DDR4 3200 ECC)

- 16 core/32 threads CPUs

- mellanox interconnect for network

-Very fast NVMe storage nodes and 200 Gb RDMA networking

Let's include this somewhere in our Presentation

This is showing the number of daily leases active on Akash Network.

The highlight here is that Spike happened around April. At the time, there's a user that deployed literally thousands of cpu instance leases to do some works that are typical to HPC use-case.