KubeCon

CloudNativeCon

THE LINUX FOUNDATION

OPEN SOURCE SUMMIT

AI_dev

Open Source GenAI & ML Summit

China 2024

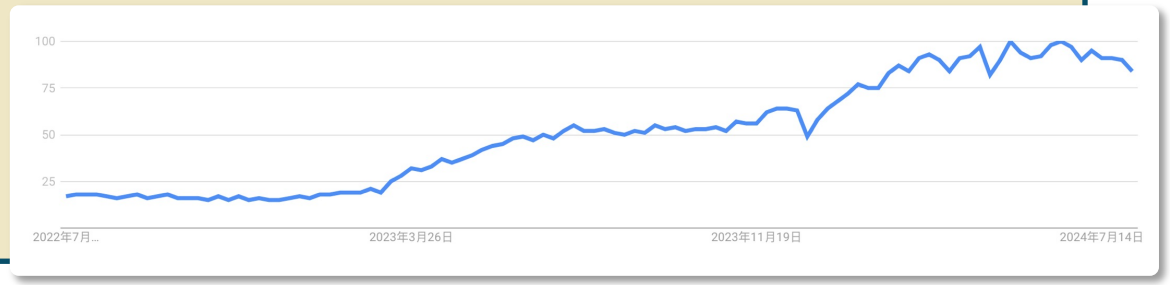
Optimize LLM Workflows with Smart Infrastructure Enhanced by Volcano

Xin Li, Qihoo360
Xuzheng Chang, Huawei Cloud Technologies Co., LTD

Catalog

Background

LLM Keyword Trends

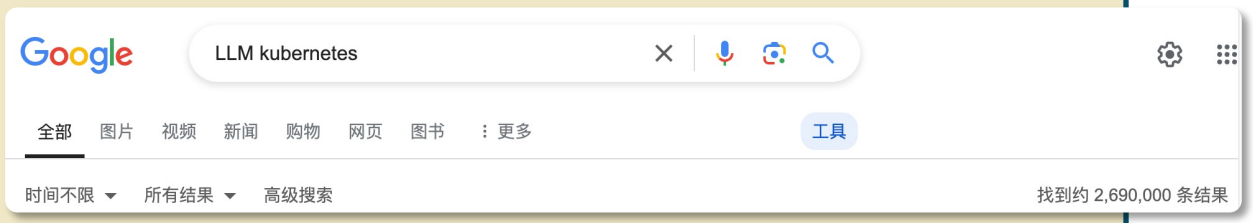


OpenAI Blog Post

Scaling Kubernetes to 2,500 nodes 2018

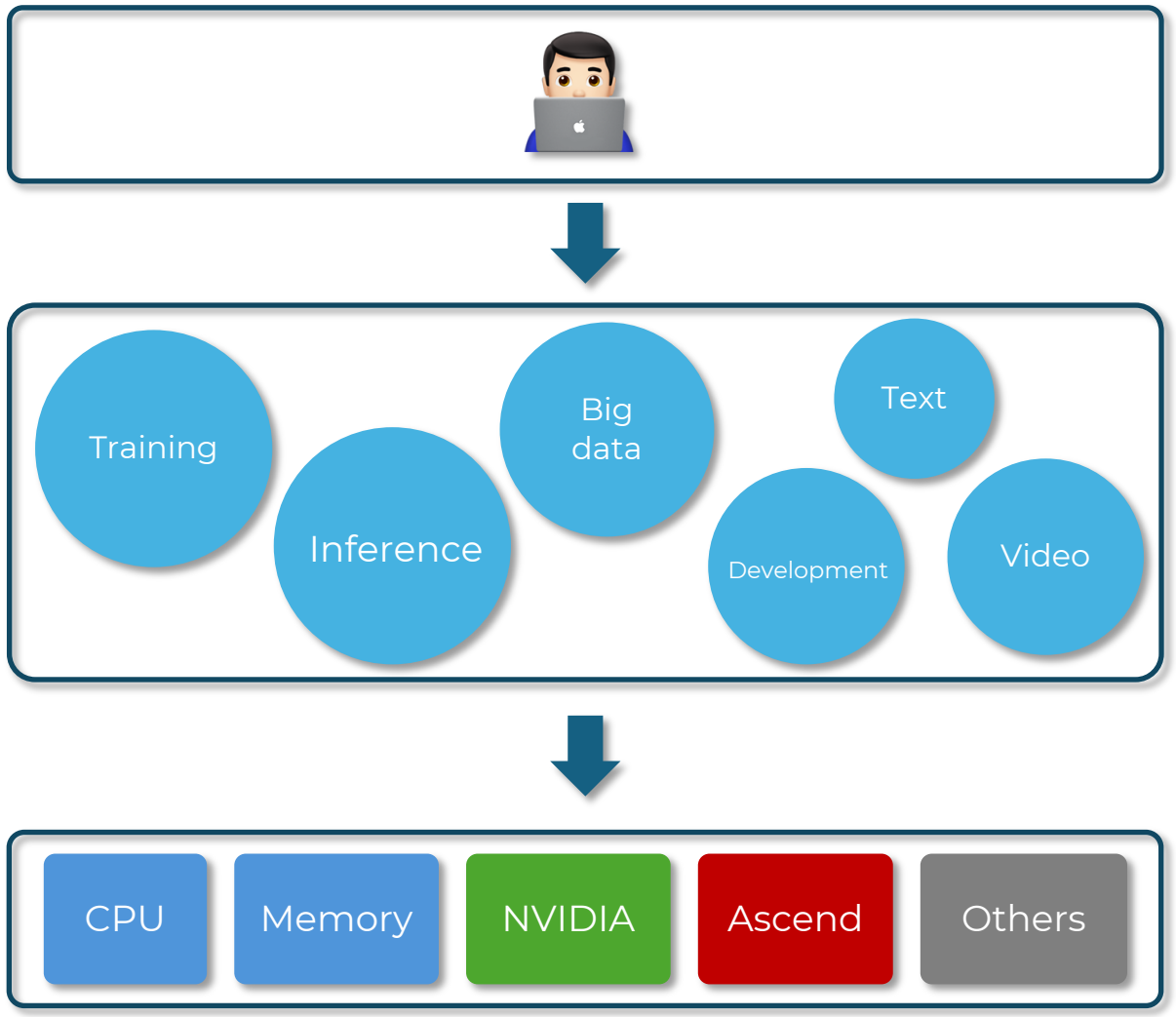
Scaling Kubernetes to 7,500 nodes 2021

Google Search Results



- Starting from 2023, LLM has received more and more attention
- More and more LLM infrastructures using Kubernetes
- Kubernetes' support for LLM is getting better and better

Status



x3000

x6000/M

x1000

- 3000+ users from different departments, 6000+ tasks per month
- 10+ clusters, 1000+ nodes
- Complexity of task types. Training, reasoning, development.
- Resources: 1-200 instances per task, single instance CPU: 1c-200c, GPU: 1-8, memory 20G-2T
- Function: ssh password-free, pod-to-pod communication
- Operation: all instances are scheduled simultaneously
- Complexity of running time. Hours, days, months and days coexist.
- Complexity of computing resources. CPU, GPU, NPU, etc.
- Complexity of network environment. Ethernet, IB, RoCE

Issue

Failure

Efficiency

Usability

Failure

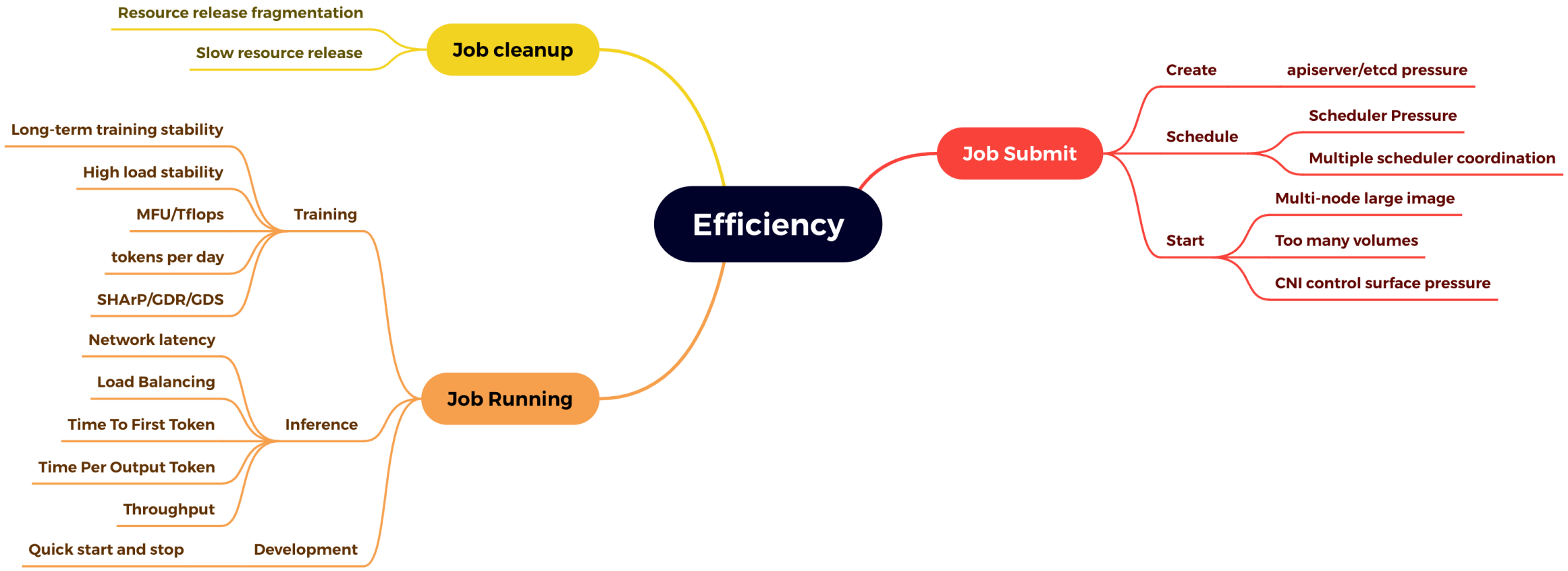
- GPU lost
- ECC error
- GPU failure
- NIC failure
- Data center power outage
- Misoperation
- NAS failure
- Cluster failure
- NVLINK failure
- P2P failure
- Cooling failure
- ……

The Llama 3 Herd of Models

During a 54-day snapshot period of pre-training, we experienced a total of 466 job interruptions. Of these, 47 were planned interruptions due to automated maintenance operations such as firmware upgrades or operator-initiated operations like configuration or dataset updates. The remaining 419 were unexpected interruptions, which are classified in Table 5. Approximately 78% of the unexpected interruptions are attributed to confirmed hardware issues, such as GPU or host component failures, or suspected hardware-related issues like silent data corruption and unplanned individual host maintenance events. GPU issues are the largest category, accounting for 58.7% of all unexpected issues. Despite the large number of failures, significant manual intervention was required only three times during this period, with the rest of issues handled by automation.

https://ai.meta.com/research/publications/the-llama-3-herd-of-models/

Efficiency


Efficiency
Job cleanup
Resource release fragmentation
Slow resource release
Job Running
Training
Long-term training stability
High load stability
MFU/Tflops
tokens per day
SHArP/GDR/GDS
Inference
Network latency
Load Balancing
Time To First Token
Time Per Output Token
Throughput
Development
Quick start and stop
Job Submit
Create
apiserver/etcd pressure
Schedule
Scheduler Pressure
Multiple scheduler coordination
Start
Multi-node large image
Too many volumes
CNI control surface pressure

Usability

Development

- Massive data transfer
- Environment dependency
- Environment preservation
- Multiple IDE integrations
- Tensorboard,
- Grafana
- Observability optimization

Scheduling strategy optimization

- Multi-department resource allocation
- Exclusive resources/public resources
- Task preemption
- Task queuing
- Gang scheduling strategy
- Binpack scheduling strategy

Multiple mission types

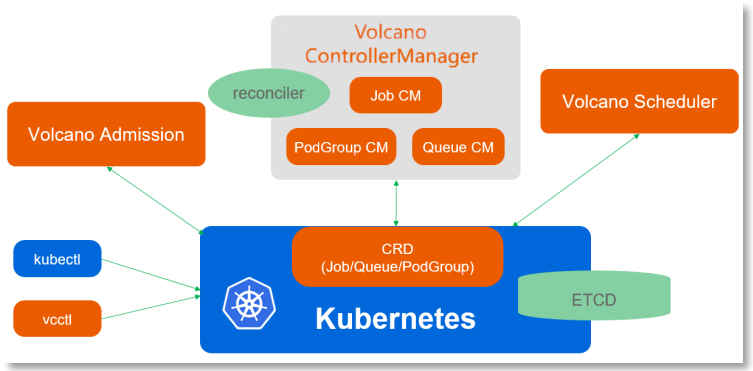
- Megatron-LM
- DeepSpeed
- opensora
- Distributed training tasks
- LLM tasks
- Multimodal tasks
- Data processing

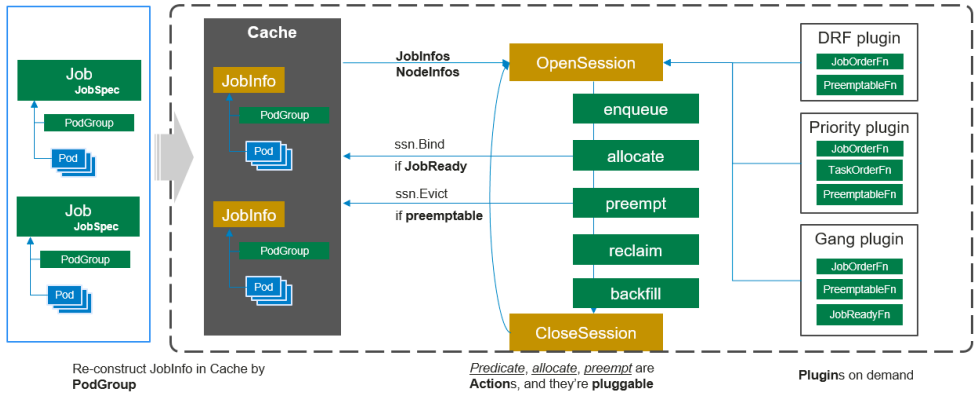
Various hardware

- Single machine single card, single machine multiple cards, multiple machines multiple cards tasks
- NVIDIA
- Ascend
- Pure CPU tasks
- RoCE/IB
- GPU slicing

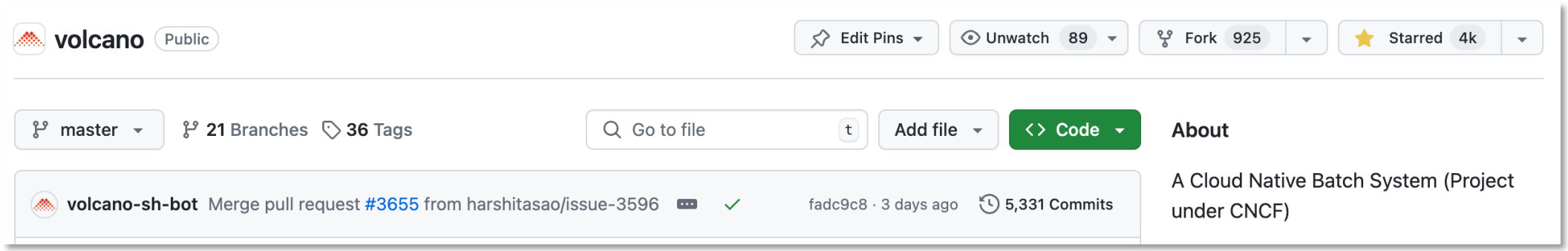
Solutions

Volcano Introduce



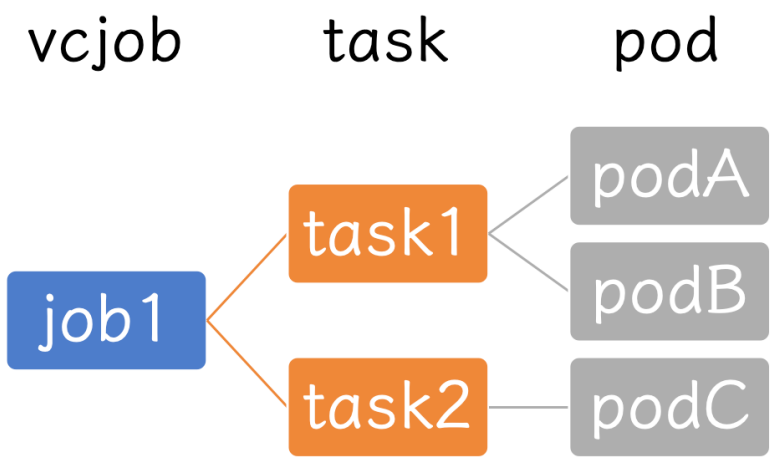


- The volcano project is Huawei open source and CNCF incubation project.
- Establish new resource abstraction according to the characteristics of AI operations. Make up for the shortcomings of Kubernetes native. Support queue, support multiple tasks in a job, and better support batch jobs
- Support multiple computing resources, including but not limited to CPU, GPU, NPU.
- Support multiple training frameworks, tensorflow, pytorch, paddle, etc.
- Support multiple scheduling strategies, and support multiple scheduling strategy combinations, the more important ones are gang, priority, binpack, drf, etc. And compatible with kubernetes native scheduling strategies, such as image perception.



Usability-vcjob

Better Batch Job



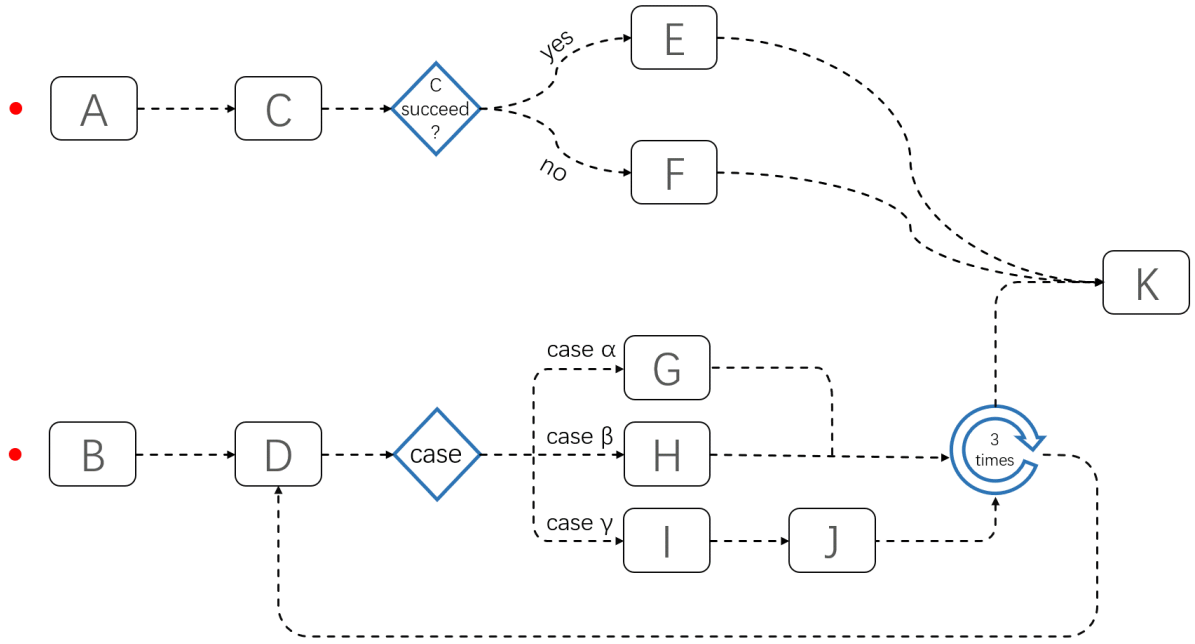
- A vcjob contains multiple tasks, each task is a different role, and a task contains multiple pods.
- task1 and task2 can be compared to ps/worker in tensorflow

Plugins

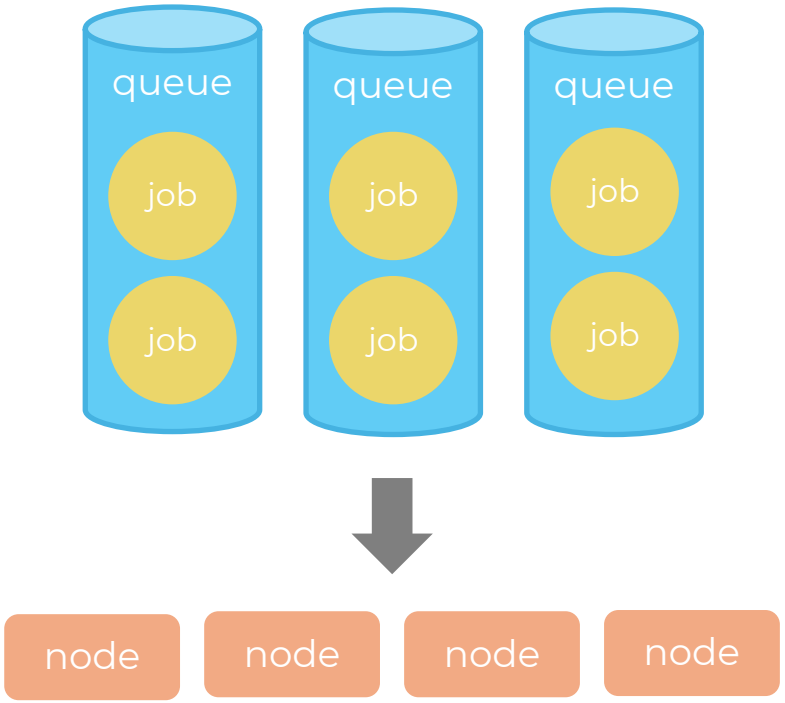
- ssh: pod ssh without password
- env: create environment variables for pod index
- svc: create svc and networkpolicy for vcjob
- Pytorch: enable svc plugin, open port, create environment variables used by pytorch in pod
- Mpi: force enable svc, ssh, open port
- Tensorflow: enable svc plugin, open port, create environment variables used by Tensorflow in pod

Usability-jobflow

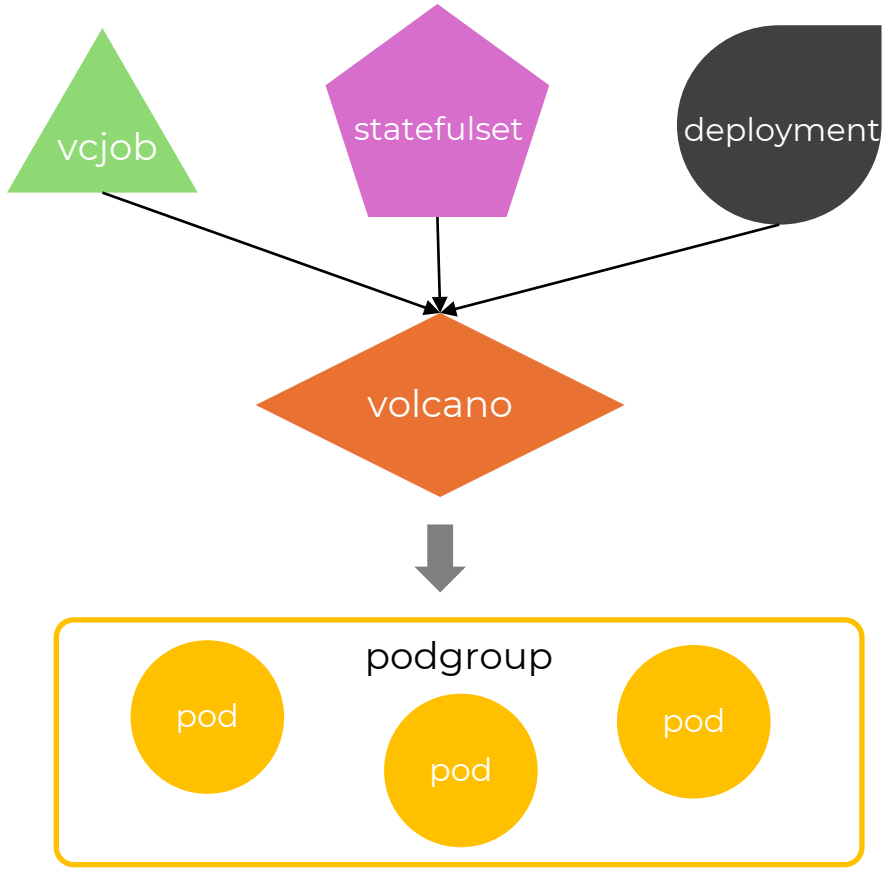
- Support vcjob workflow
- A lightweight alternative to argo workflow
- Supports multiple running conditions (ongoing)



Usability-queue & podgroup



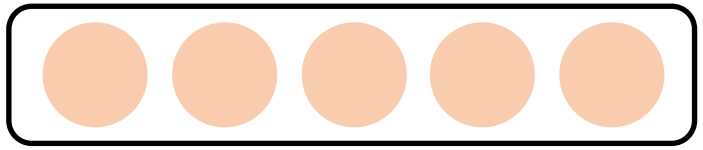
- Multiple queue support
- Job preemption within a queue
- Job preemption between queues
- Queue capacity
- Divide resources by weight
- Hierarchical queue (ongoing)

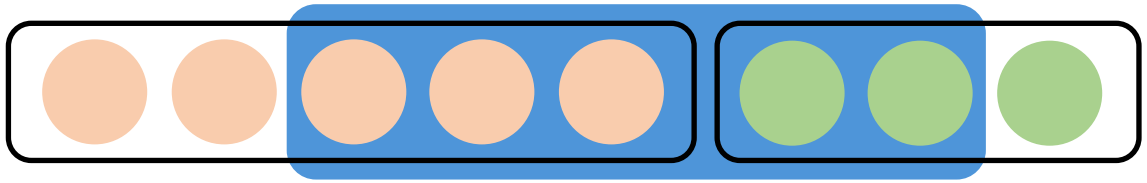


Support converting other Kubernetes workload into podgroups and using volcano scheduling

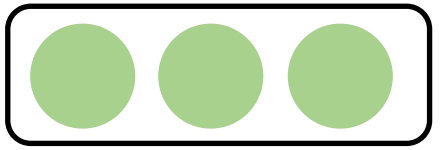
Usability-Scheduling strategy-gang

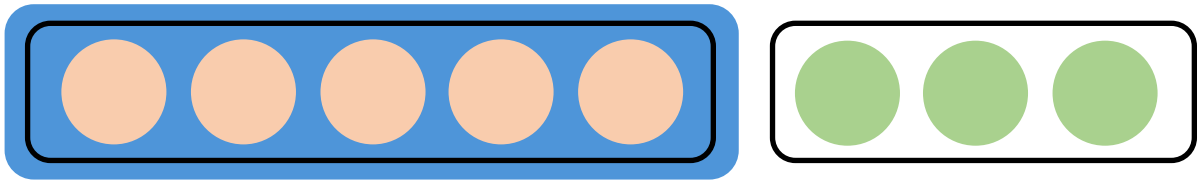
Job A





Job B





Fault Handling -Detection & Repair

Software and hardware failures are inevitable

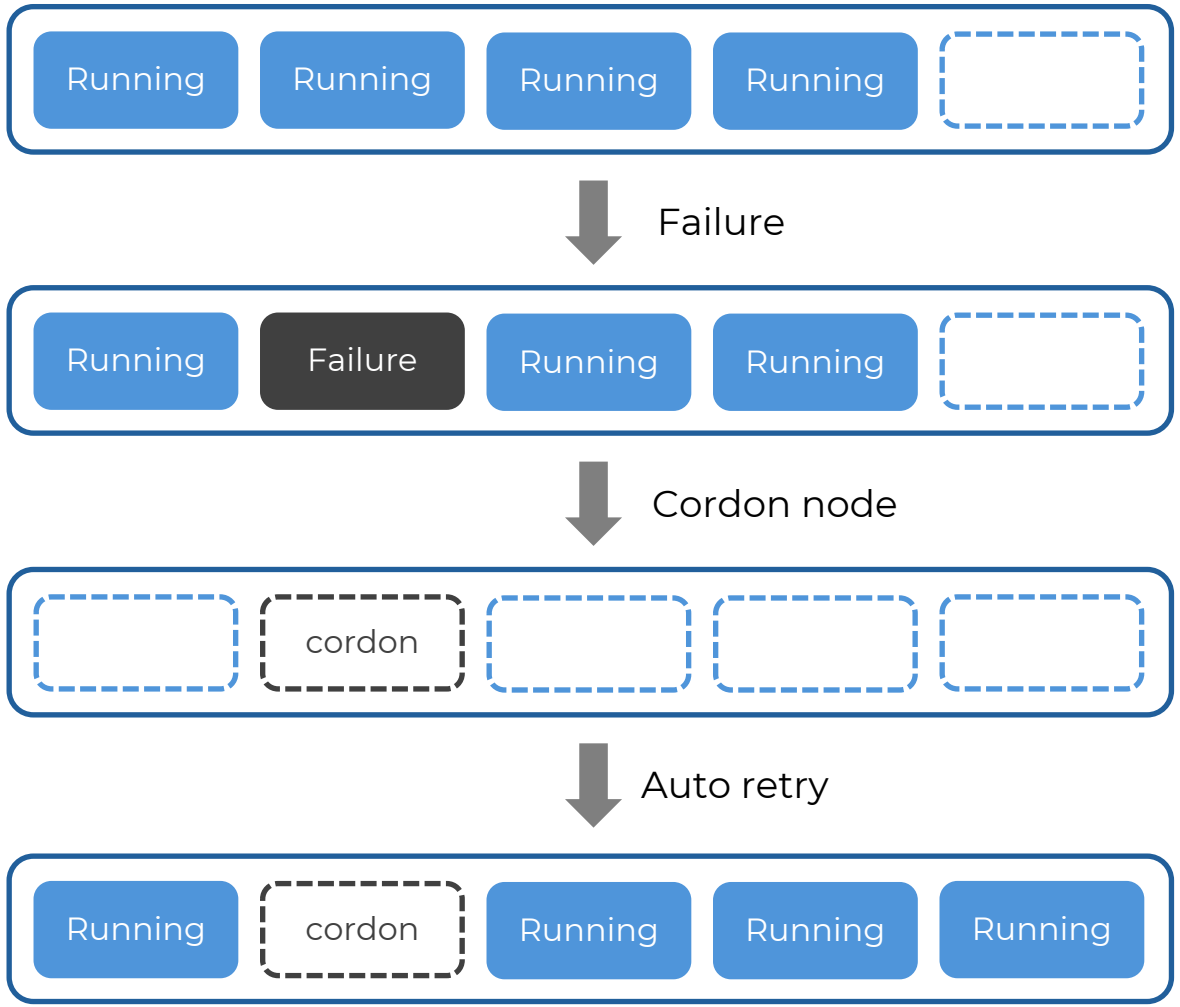
- Auto detect
- Try to repair

Fault Handling-Job Retry

Automatic job retry

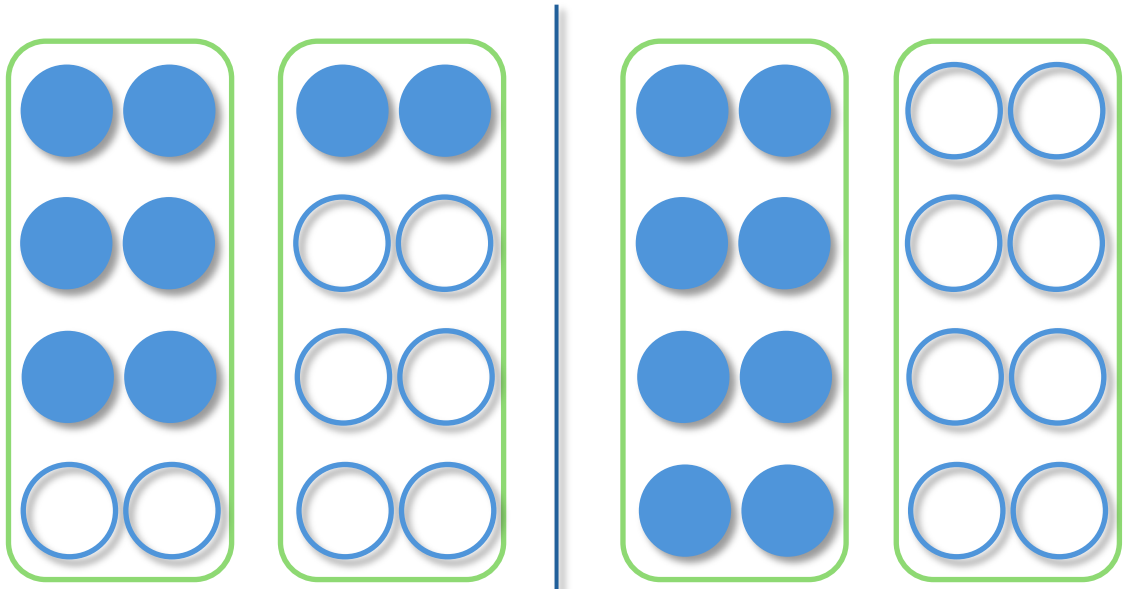
```yaml
apiVersion: batch.volcano.sh/v1alpha1
kind: Job
metadata:
  name: vcjob-demo
spec:
  maxRetry: 3
  minAvailable: 128
  queue: default
  schedulerName: volcano
  tasks:
  - maxRetry: 3
    minAvailable: 1
    name: master
    ...
```

```yaml
policies:
- action: CompleteJob
  event: TaskCompleted
- action: TerminateJob
  event: PodFailed
- action: TerminateJob
  event: PodEvicted
```

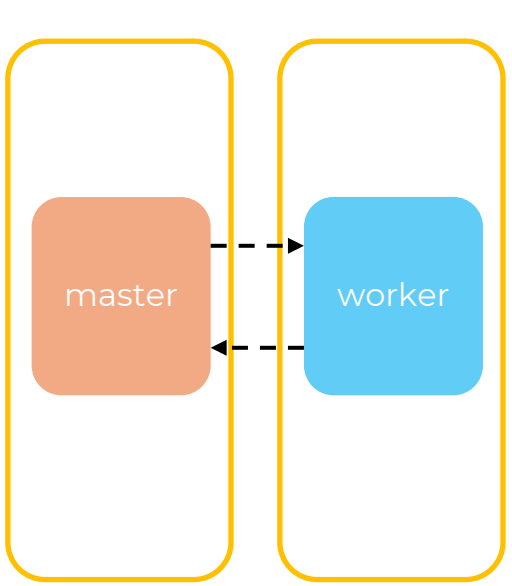


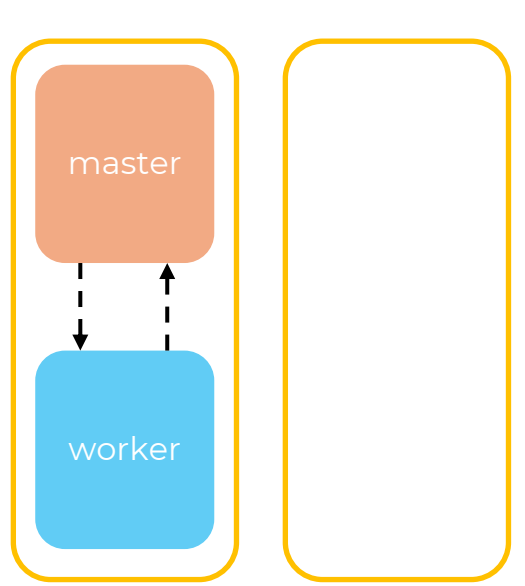
Improve efficiency-binpack&task-topology

binpack



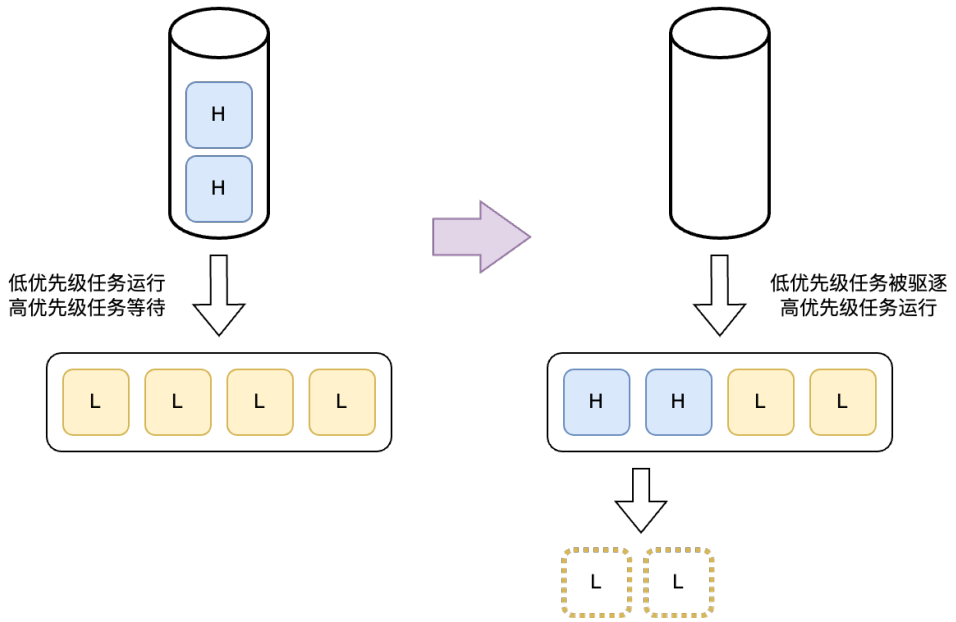
Task-topology



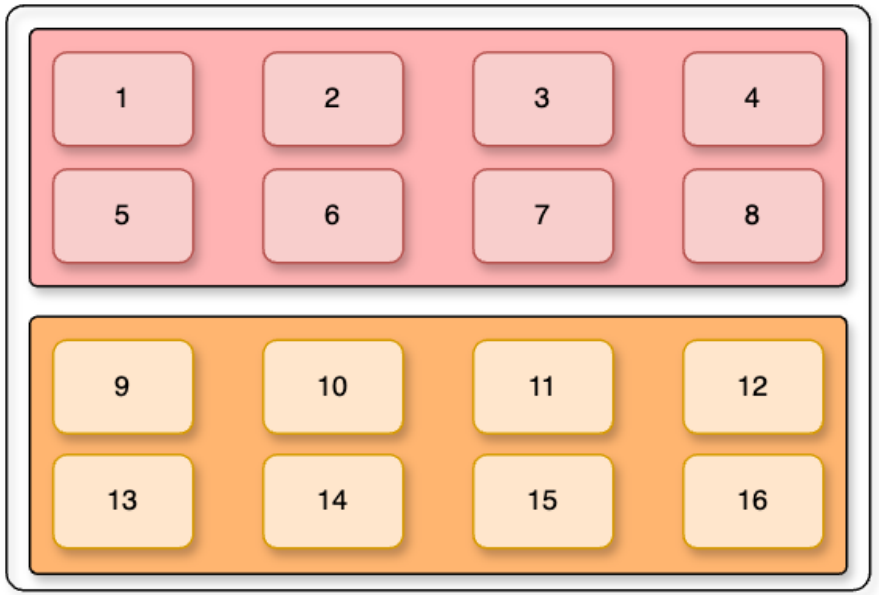


Improve efficiency-preempt&Ascend optimization

preempt



Ascend optimization

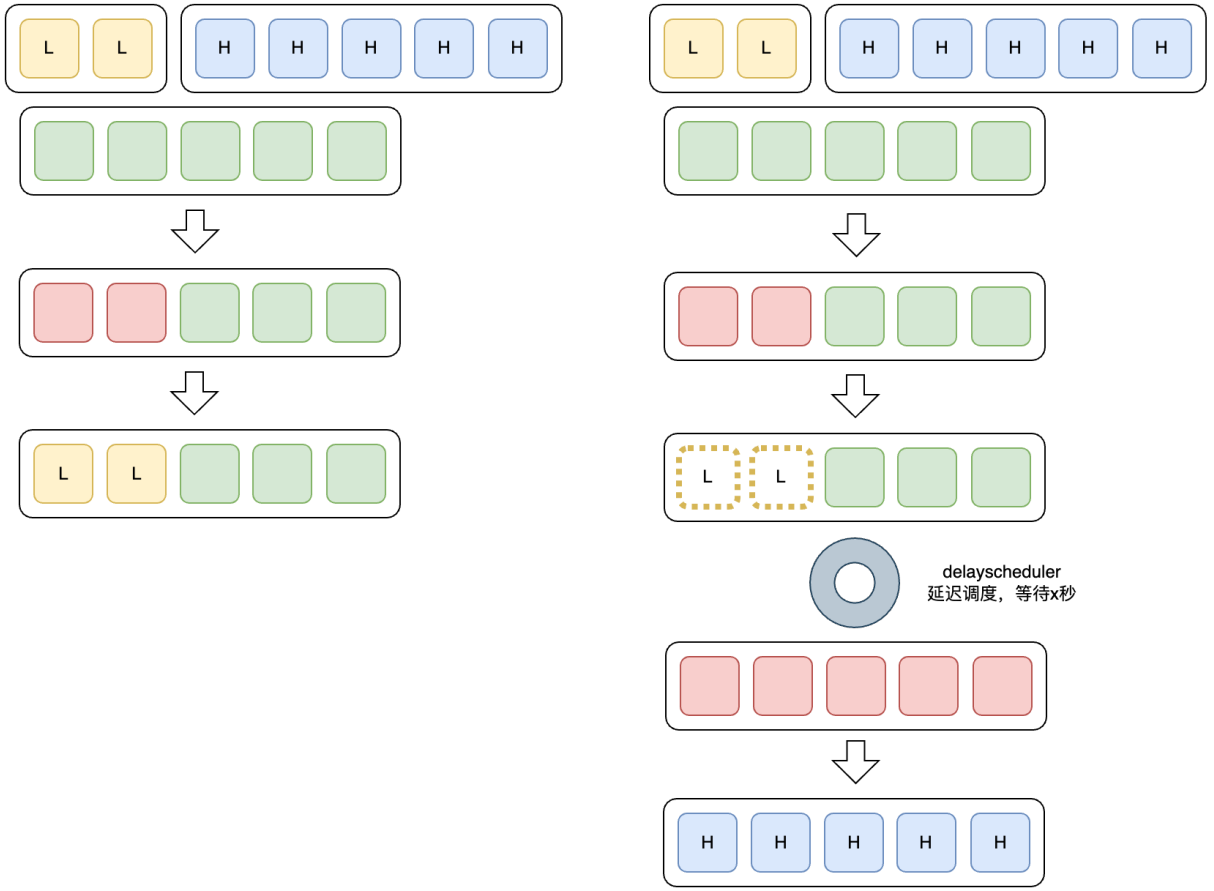


- 16 cards per node
- Eight cards form a ring
- Jobs need to run in the same ring

Improve efficiency-Start&cleanup

Delayed Scheduling

- When Kubernetes releases a Pod, each node releases it at a different time. Small tasks always run before large tasks
- Low-priority tasks wait X seconds before scheduling, giving the node time to clean up




KubeCon
CloudNativeCon
THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
AI_dev
Open Source GenAI & ML Summit
China 2024
Thanks !