CONFIDENTIAL

Gfarm ワークショップ 2025

「HPCシステム構築・運用の変革について」

2025年12月18日　宮古島

Pacific Teck Japan 合同会社 Senior Engineer 森本 賢治

背景

- 数年前まで「AIのワークロードをHPC環境で動かすにはどうすればよいか？」がホットなトピックだった。
- 現在「AI環境でHPCワークロードを動かすには？」が次のトピックになりつつある。
- HPCシステムに従来とは異なる利用方法やあり方が求められることが増え、衝突が起きている。
- これまでHPCとは縁のなかった界隈でHPC/AIシステム構築の需要が高まる中、ベストプラクティスが定まらず、混乱が続いている。
- 技術や要件の旬が過ぎるのがとにかく速い。

OSSをサポートするとは？

1. 挙動が開示されたソフトウエアを自己責任で使うセルフサポート。
2. 使い方に通じた者が、第三者にコンサルをするテクニカルサポート。
3. 開発者との契約に基づき、挙動と品質に対して責任を持つ、動作保証サポート。
4. 販売しているソフトウエアのソースコードが開示されているもの。

従来 Pacific Teck のビジネスモデル的には、ソフトウエアの品質保証ができる 3, 4 のみを対象として、「**責任**」と「**知見**」の「**対価**」が渡るようにしている。

しかし、いわゆる＊＊Foundation などコミュニティが開発主体となっていて、有償での品質保証は企業体によるものではなく「替えが効かない唯一性」「広く利用されている実績」「高頻度の更新」「それを利用しているビジネス経済圏」により支えられているプロダクトが増えてきた。

一方で、責任を取ることのコストを市場から供給できずに開発者が更新を止めてしまったり、OSSであることをやめてしまうプロダクトも出てきた。サポート体制をどう維持するのかを再考していく必要がある。

HPCクラスターマネジメント

HPC CLUSTER MANAGEMENT

HPCクラスターの構造

Remote Management Services
Monitoring, Updates, Security, Backup, User Support

Scheduler (Slurm/PBS/SGE) | File System (Beegfs/Lustre/NFS) | openHPC

Cluster Management — trinityX

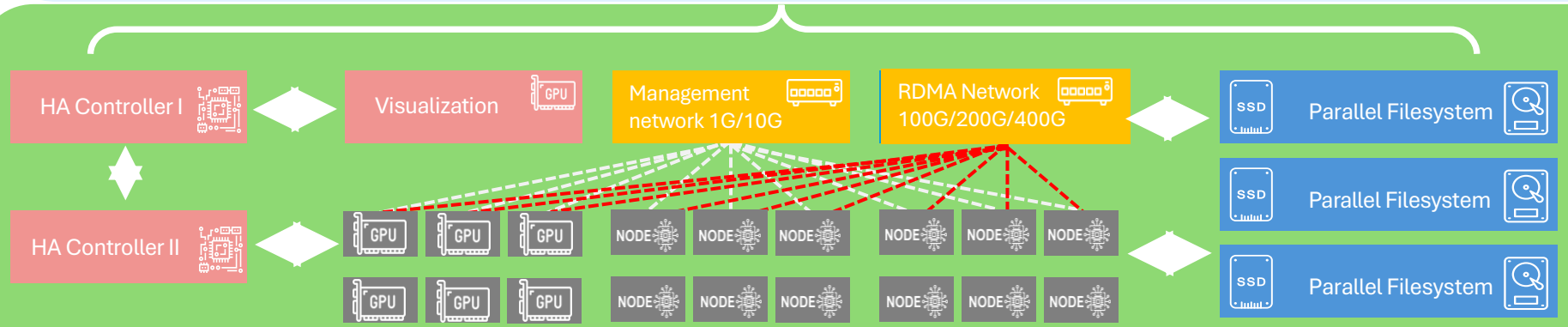
Configuration Management | Monitoring & Alarms | Node Management

Environment Modules | Security & Node Logs | HW/IPMI Management

Linux OS (Rocky / Alma / Ubuntu / Red Hat)

様々なコンポーネント群を精巧に組み合わせてシステムを構築
基本はどれも似た構成。

事前構築済みのベストプラクティスが欲しい。



ハードウエア群
管理ノード
計算ノード
ストレージ
ネットワーク
…

trinityX

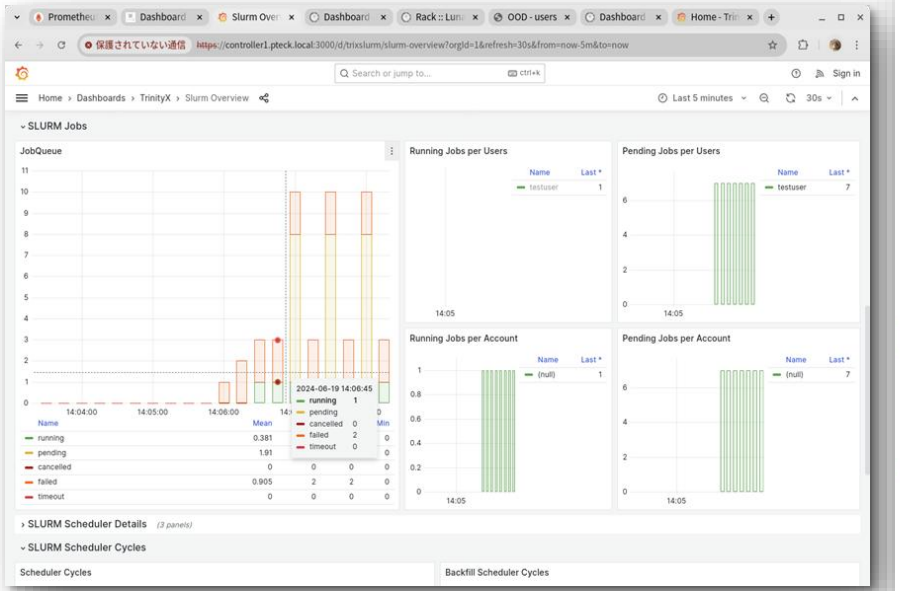
clustervision

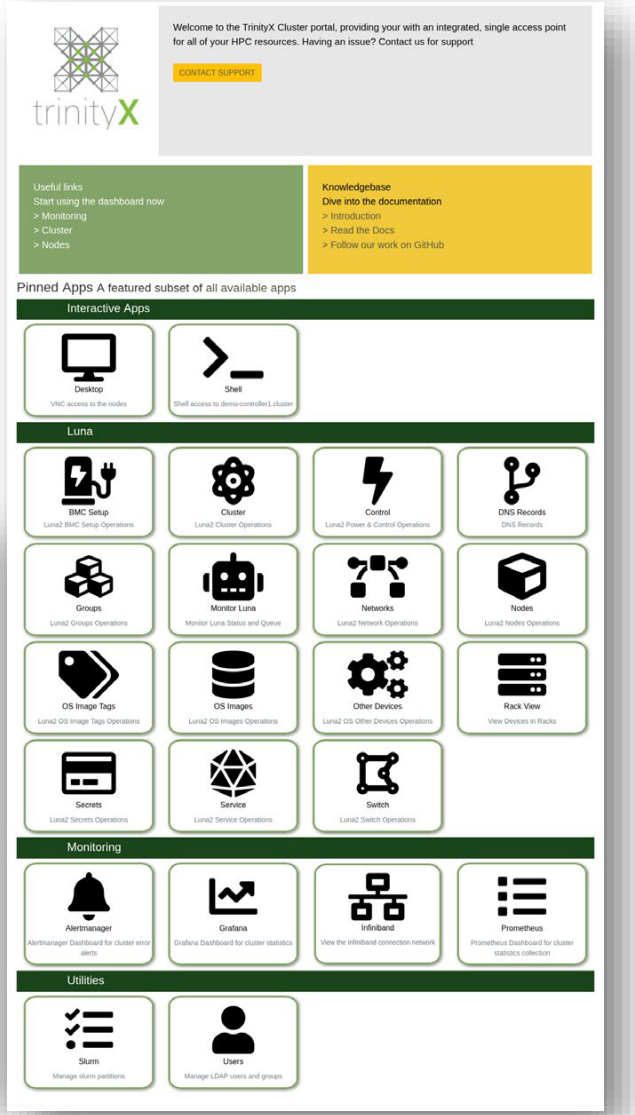
OPEN OnDemand

ANSIBLE

Grafana

Prometheus





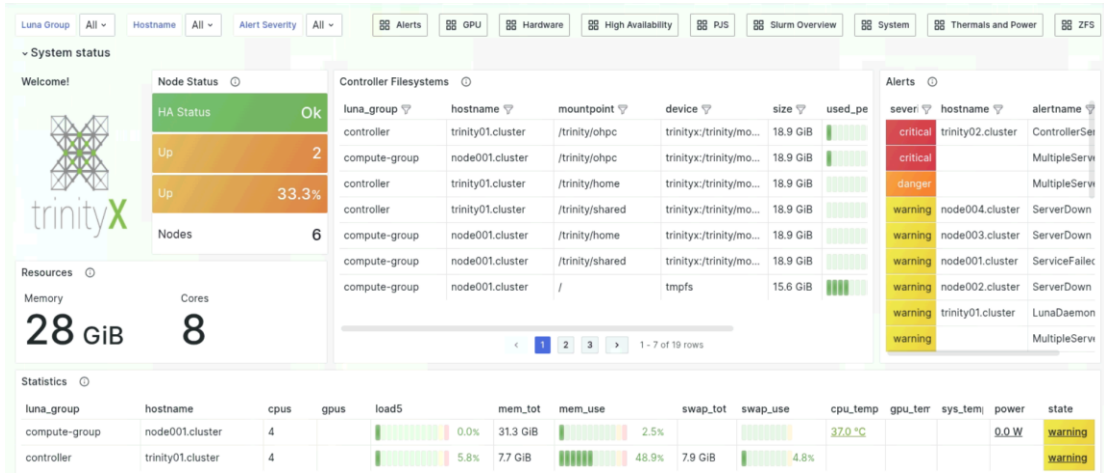
- 実績あるOSSを中心に設計されたオープン志向のHPCクラスター構築・管理ツール。
- Open OnDemandの上にユーザーポータルのみならず、管理UIを統合。
- GPUモニタリング、ジョブ実行状況分析など、安定運用に寄与する様々な管理画面を用意。
- OSイメージをBitTorrent展開可能。クラスター全体のOSプロビジョニング・更新が円滑に。
- ClusterVision社はHPC環境の構築・運用を請け負うサービス企業。自社利用のツールを公開。

Per Job Statistics - ジョブ単位でのリソース利用解析

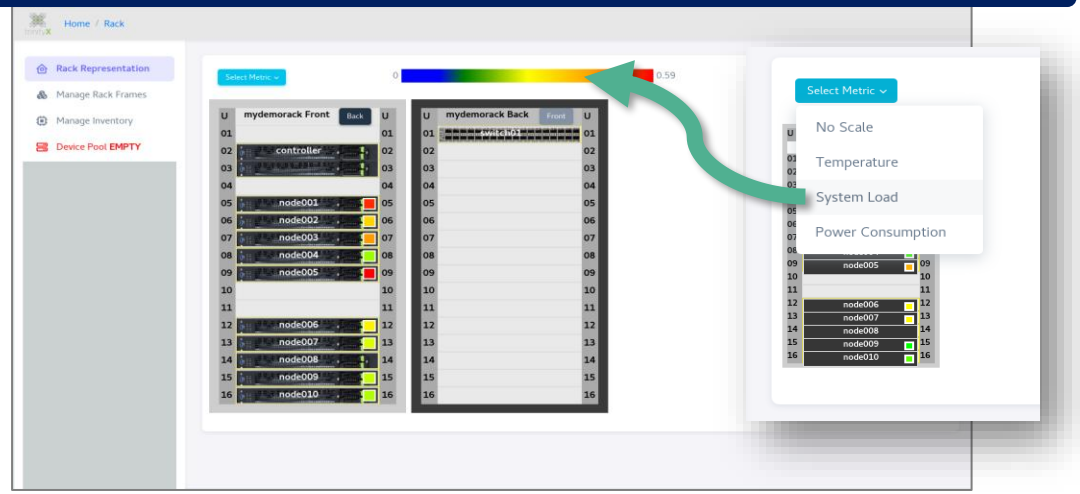
- ノードごとの負荷ではなく、個々のジョブがどうリソースを使用していたかを可視化。
- スケジューラと連携して各アプリケーションのリソース利用を自動記録。実行後に詳細な分析が可能。
- トラブルシューティング、スケジューリング最適化 や リソース計画立案 にも貢献

直感的なモニタリングUI

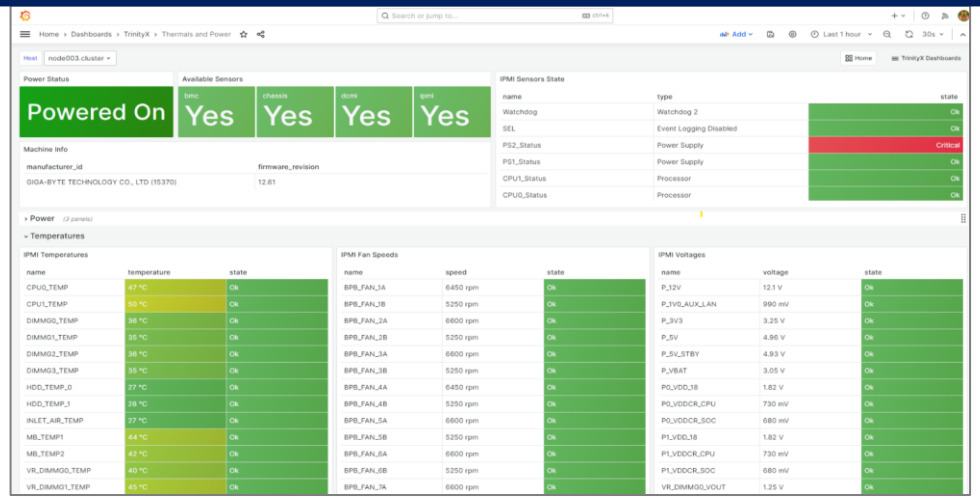
Grafana のホームダッシュボード



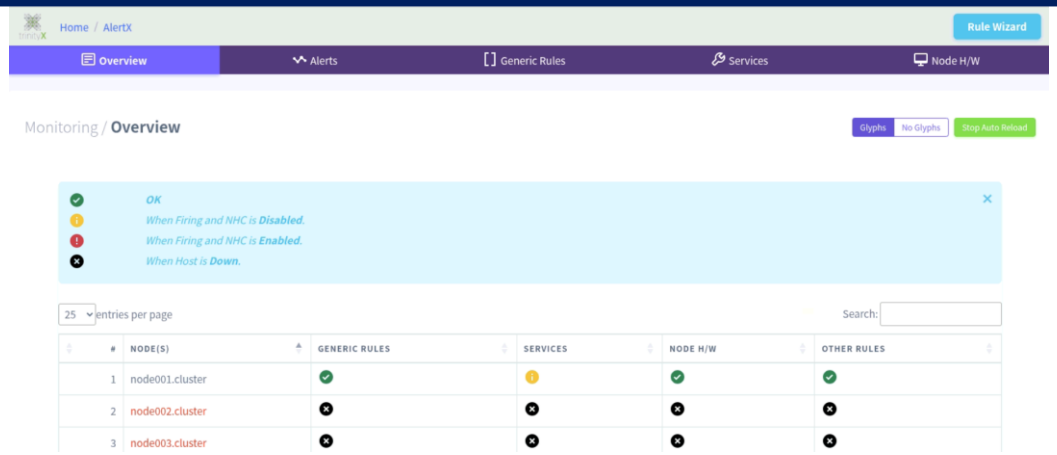
ラックビューでのヒートマップ表示



Grafana によるセンサーデータのダッシュボード

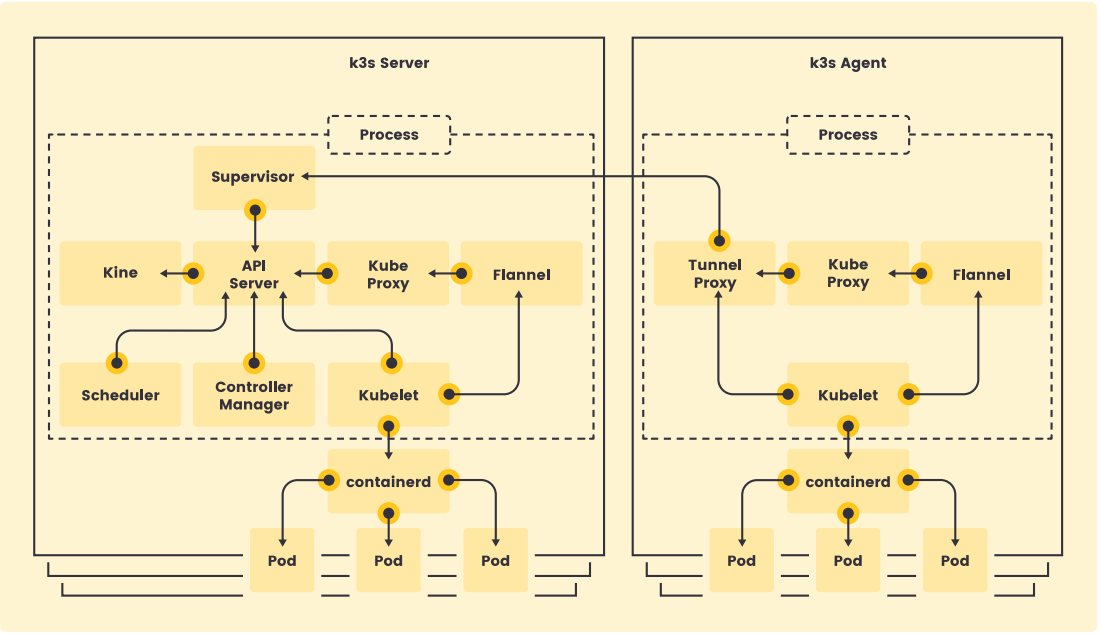


障害管理を行うAlertX のホーム画面



Kubernetes への対応

- 軽量版 Kubernetes である K3s ノードイメージの作成とデプロイに対応。
- OSイメージの選択で、Slurmノードと K3sノードを混在させたクラスター運用が可能。



TrinityX

- https://github.com/clustervision/trinityx からGPLv3 で配布
- 開発元の ClusterVision社とのBack to Back 契約によるテクニカルサポート。
- カスタム対応など

Developer

clustervision

オランダ拠点のClusterVisionは、HPC/AIに特化した企業としてクラスタ管理基盤「TrinityX」を開発・提供。Bright Cluster Manager(現NVIDIA)を開発した企業。

共有ストレージ

Shared Storage System

BeeGFS 8.x

８年ぶりにメジャーバージョンアップしたBeeGFS 8 はソースコードから刷新。
基本仕様はそのままに新機能を追加。データマネジメントツールの充実させた。
開発は ThinkparQ 社によるものでコミュニティによる開発ではないが、ソースは Github にて公開されている OSS 。

Remote Storage Targets(RST)

実データ保存先ストレージターゲットにS3を指定可能。

Watch

イベントリスナーAPIが実装され、メタデータのパフォーマンスを犠牲にせずにIndexingの利便性を強化。

Copy

クライアントノードを束ねてインクリメンタルなファイルコピーを行う強力なデータステージングツール。

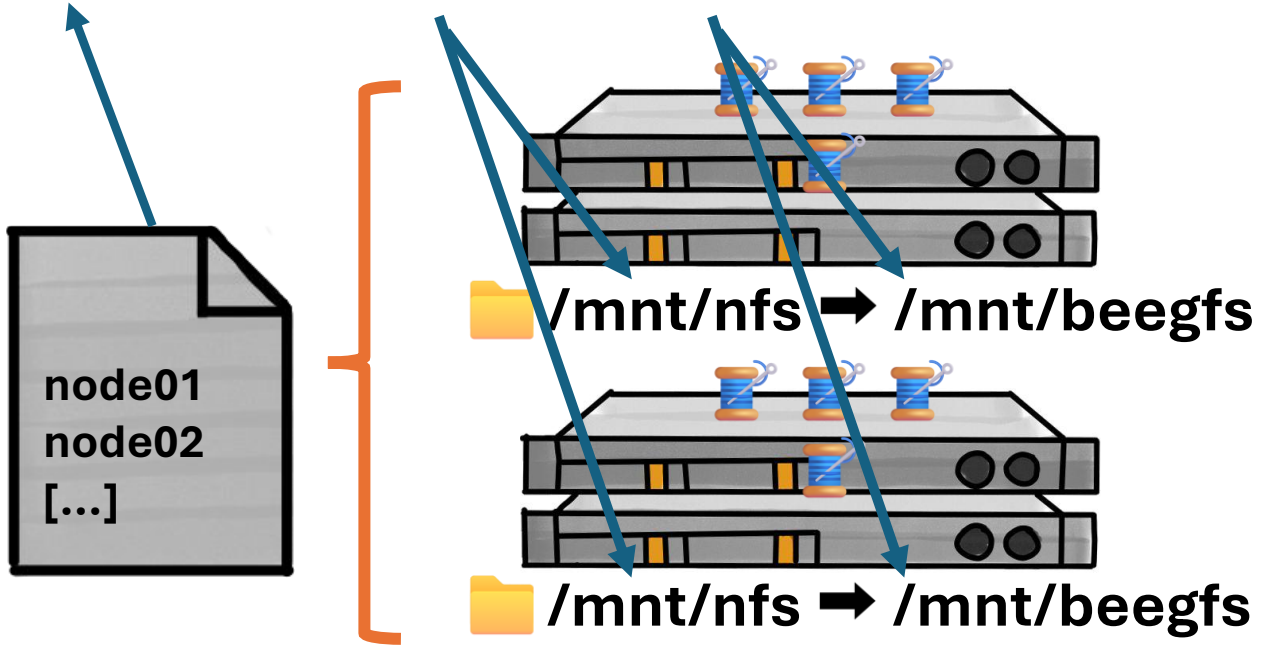
Utility

従来機能ごとに異なるコマンドが用意されていたが、beegfsコマンド１つに機能を集約。

Copy: Like cp, but fast

- ワイルドカードや標準出力からの入力による自在なパス指定でのコピーが可能。
- rsync 同様の差分コピーに対応。
  - データ移動をしない差分表示モード(Dryrun) あり。
- マルチスレッド・マルチノードでのコピーが可能。
  - 巨大なファイルは分割コピーを実施（デフォルトの分割サイズ＝1GB）

```
# beegfs copy -m machinefile.txt -t 4 /mnt/nfs/ /mnt/beegfs
```



Index: A queryable catalogue of your data

- ファイルシステム全体、特定ディレクトリに対するインクリメンタルスキャンが可能。
- パイプ処理を繋ぎ易い入出力。

🔍 Find - GNU find互換のオプション体系
例） 365日アクセスがないファイルのうち最大10件を検索。

```
# beegfs index find --num-results=10 --largest --type=f --atime=+365
```

📝 Query - SQL クエリ
例） logsディレクトリ以下にある全ファイルのBeeGFSエントリ情報。

```
# beegfs index query -l logs/ -s "select * from entries“
```

📊 Stats - ファイルシステム統計情報・メトリック表示
例）ディレクトリ階層毎のファイル数カウント

```
# beegfs index stats files-per-level
```

🧩 他の機能（Copy 等）へパイプで直に接続。

```
# beegfs index [...] | beegfs copy -m machinefile.txt - /mnt/archive/
```

Pools: Data tiering inside BeeGFS

異なる構成のストレージターゲット間のティアリングを実現する StoragePool 機能と連携。

- プールにターゲットを紐づけてグループ化（SSDベース・HDDベース等）
- ディレクトリにストレージプールをアサインし、デフォルトの保存先を指定。
  - ファイルごとに保存先プールを指定・移動でき、単一ディレクトリ内に混在が可能。
- beegfs entry migrate コマンドでプール間を移動可能。
  - ファイルシステム上だけでなく、サーバーノードやターゲットを指定しての移動が可能。

🧩 Pools+Index: プール 1 (fast)上で1年間アクセスがない全ファイルをプール 3 (archive)へ。

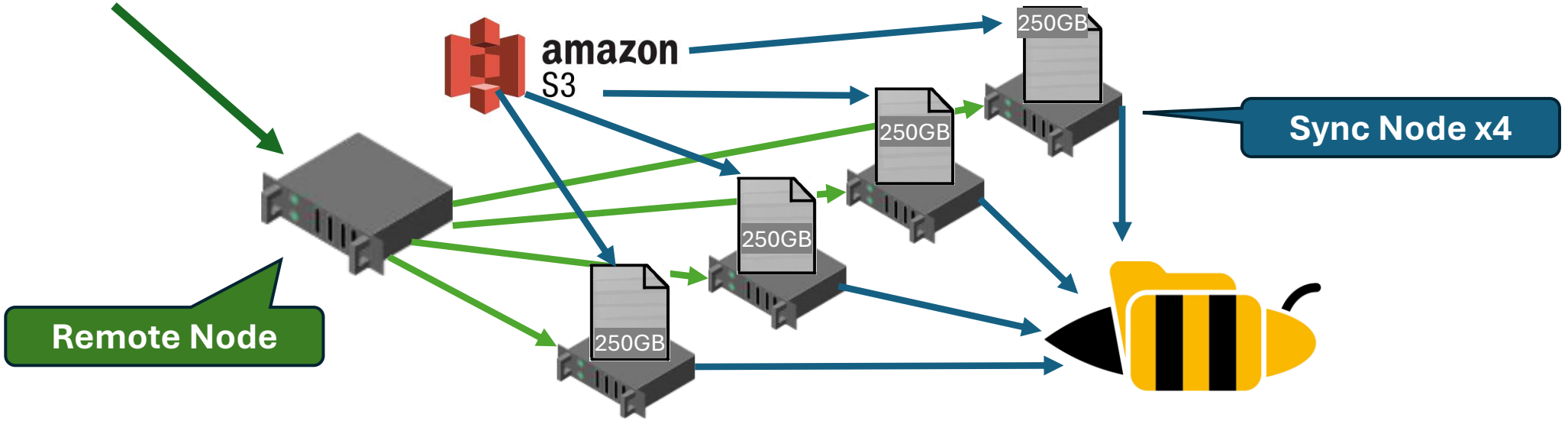
```
# beegfs index find --type=f --atime=+365 ¥
        | beegfs entry migrate --from-pools=s:1 --pool=s:3 --verbose -
```

```
PATH                                          STATUS          ORIGINAL_IDS   ERRORS
/2024-backup/run_2022_04_01/logs_stdout.log   migrated file   [1]            none
/2024-backup/run_2022_04_01/output_0001.dat   migrated file   [1]            none
[...]
Summary: {MigrationStatusUnknown:0 MigrationErrors:0 MigrationNotSupported:0
MigrationSkippedDirs:0
MigrationNotNeeded:0 MigrationNeeded:0 MigratedFiles:15 MigrationUpdatedDirs:0}
```

Remote Storage Targets: S3 Syncing

- S3とBeeGFSの双方向の同期
  - S3からオブジェクトをBeeGFS に Pull
  - BeeGFSからファイルをS3バケットにPush
- 高速な処理
  - 複数のSyncノードを用いて並列転送
  - 余分なリクエストを避けるためローカルにキャッシュ
- シンプルなコマンド

```
# beegfs remote pull --remote-target=1 --remote-path=hello /mnt/beegfs/world
```



BeeGFS

並列分散ファイルシステム

- HPC/AI 向けの並列分散（POSIX）ファイルシステム
- 多数ノードから並列的にストレージにアクセスできる
- HPC、生成AI学習、ライフサイエンス、メディアレンダリング、データ分析などユースケースは多岐に渡る
- 任意のHW上にストレージを構築できるSDS(ソフトウェアディファインドストレージ)

Developer

ドイツのFraunhofer研究機関で生まれた。
BeeGFSの継続開発と商用サポートを担当。
BeeGFS/BeeOND の開発・保守、エンタープライズ向けサブスクリプション（サポート/トレーニング/長期保守）を提供。

ジョブ管理システム

JOB MANAGEMENT

BatchかKubernetesか

- 過去３０年、HPCクラスターはバッチジョブスケジューラで利用されてきた。
- AIワークロードはもはやK8sインフラの利用が前提に。
- どちらもアプリケーションが必要なリソース要求を受け、それらが衝突しないよう割り当てる点は同じ。
- K8s は必要なリソースを切り出して独立した環境を作り、リソースを使いまわす。標準のスケジューラは特に時間軸に対するスケジューリングが貧弱だが、バッチジョブスケジューラの概念を取り込んだ Kueue の登場で解決？
- 結局 HPC 環境とK8s の共存は不可能？

計算リソースを奪い合う醜い争い


slurm
slinky


PEACE!
PEACE!
COOPERATION!
INTEGRATION!
BATCH
BATCH
HARMONY!

Slinky とは

Slurm の開発元である SchedMD 社による、バッチ環境とK8s の融合を目指すオープンソースプロダクト。

slurm-operator

- 既設のK8s上にSlurm バッチ処理クラスターをhelm で構築するもの。
- 外部認証(sssd)・アカウンティング・REST・Exporter のデプロイに対応。

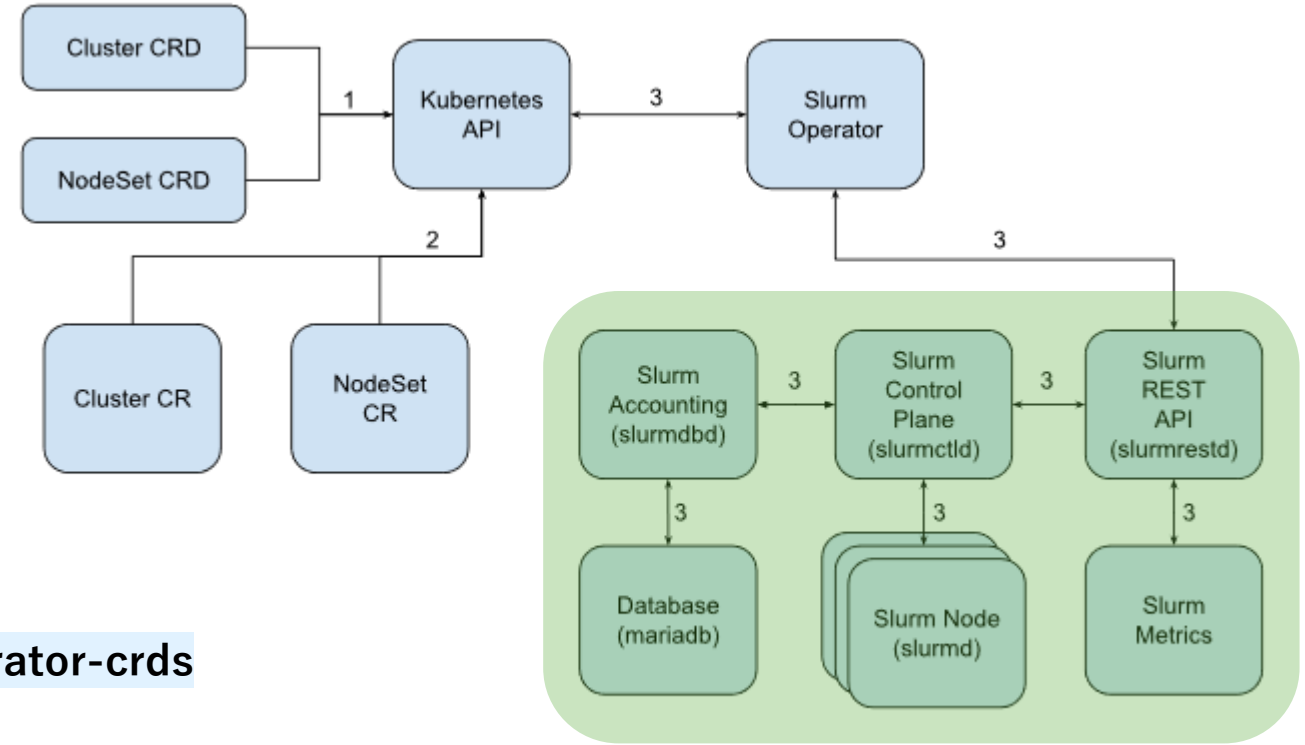
slurm-bridge

- クラスターノードに kubeletとslurmdを同居させ、slurm のポリシーで K8sのリソース割り当てとスケジューリングを行う。
- slurm からは K8s のリソース利用がジョブのように見え、slurmdbdによるアカウンティングに統合できる。

SC25に合わせ v1.0.0 をリリース

Slurm-Operator の構成

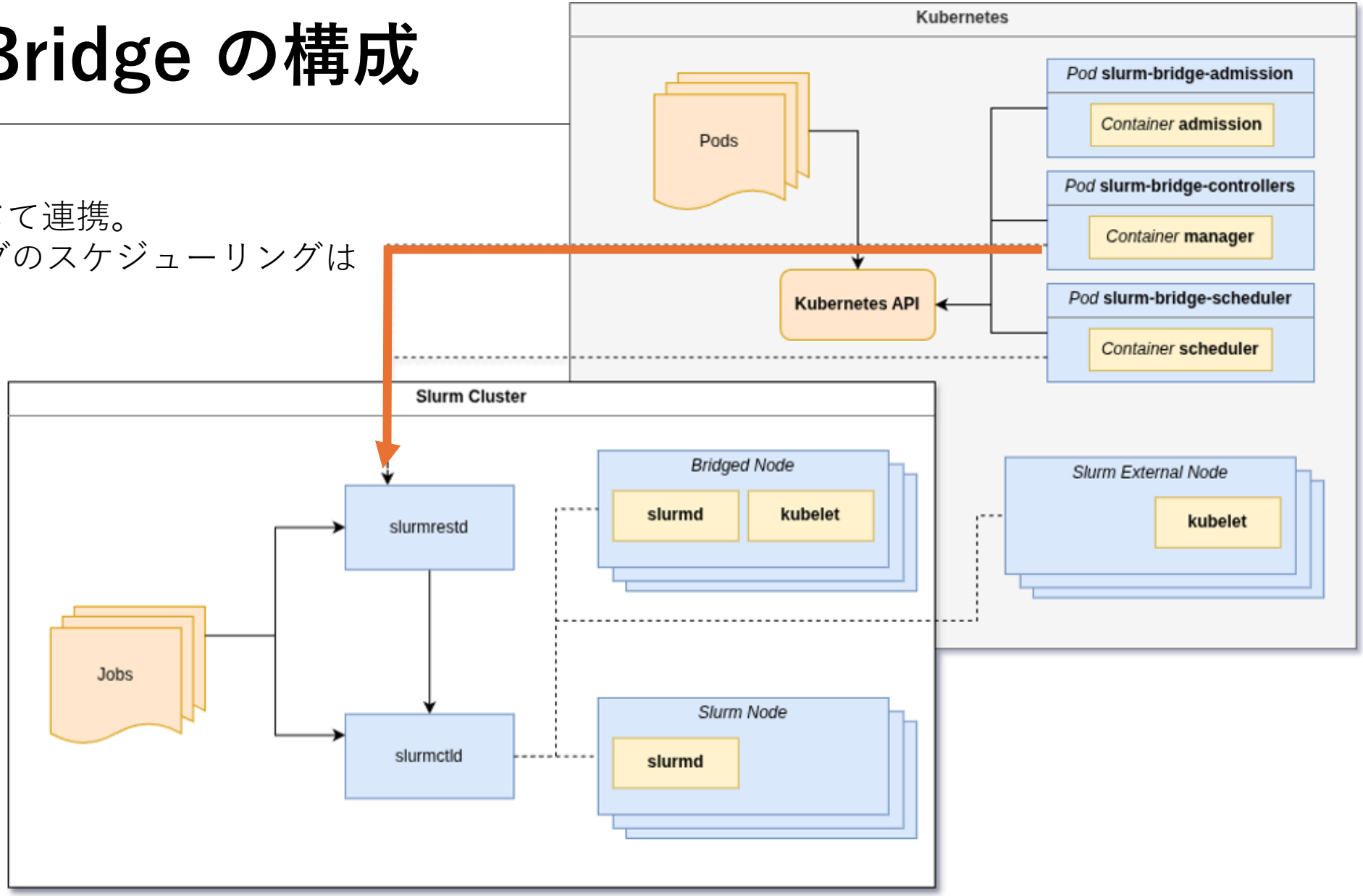
- slurmctld, slurmdbd, slurmrestd, login, worker など、別Podで起動。
- K8s外のコンポーネントと連携可能。




```
# helm install slurm-operator-crds ¥
  oci://ghcr.io/slinkyproject/charts/slurm-operator-crds
# helm install slurm-operator ¥
  oci://ghcr.io/slinkyproject/charts/slurm-operator --namespace=slinky --create-namespace
```

Slurm-Bridge の構成

slurmrestd を通じて連携。
リソースとジョブのスケジューリングは
slurm 側で統合。



Pod/Job info translation

K8s job description

```
apiVersion: batch/v1
kind: Job
metadata:
  name: job-sleep-dra
  namespace: slurm-bridge
  annotations:
    slurmjob.slinky.slurm.net/job-name: job-sleep-dra
spec:
  completions: 1
  parallelism: 1
  template:
    spec:
      schedulerName: slurm-bridge-scheduler
      containers:
        - name: sleep
          image: busybox:stable
          command: [sh, -c, sleep 30]
          resources:
           limits:
              cpu: '1'
              memory: 100Mi
              deviceclass.resource.kubernetes.io/gpu.example.com: 1
```

scontrol show job

```
JobId=1 JobName=job-sleep-dra
   JobState=RUNNING Reason=None Dependency=(null)
   ReqNodeList=kind-worker[5-9] ExcNodeList=(null)
   NodeList=kind-worker5
   BatchHost=kind-worker5
   NumNodes=1 NumCPUs=12 NumTasks=1 CPUs/Task=1 ReqB:S:C:T=0:0:*:*
   ReqTRES=cpu=1,mem=100M,node=1,billing=1
   AllocTRES=cpu=12,mem=100M,node=1,billing=12
   Socks/Node=* NtasksPerN:B:S:C=0:0:*:* CoreSpec=*
   JOB_GRES=gpu:gpu.example.com:8
     Nodes=kind-worker5 CPU_IDs=0-11 Mem=100
     GRES=gpu:gpu.example.com:8(IDX:0-7)
   MinCPUsNode=1 MinMemoryNode=100M MinTmpDiskNode=0
   AdminComment={"pods":["slurm-bridge/job-sleep-dra-fjkp8"]}
   TresPerNode=gres/gpu:gpu.example.com=1
```

Slurm/Slinkyコマーシャルサポート

- Slurm/Slinkyに関する、脆弱性対応を含むサポート
- 開発元によるハンズオントレーニング
- インストールおよびコンサルティングサービス

Developer

SCHEDMD
The Slurm Company

オープンソースのワークロードマネージャーであるSlurmの主な開発企業であり、主にSlurmに関するサービスの開発と提供を目的として2010 年に設立。本社の所在地はアメリカ、ユタ州。

12月15日・・・

NVIDIA

Home AI Data Center Driving Gaming Pro Graphics Robotics Healthcare Startups AI Podcast NVIDIA Life

NVIDIA Acquires Open-Source Workload Management Provider SchedMD

NVIDIA will continue to distribute SchedMD's open-source, vendor-neutral Slurm software, ensuring wide availability for high-performance computing and AI.

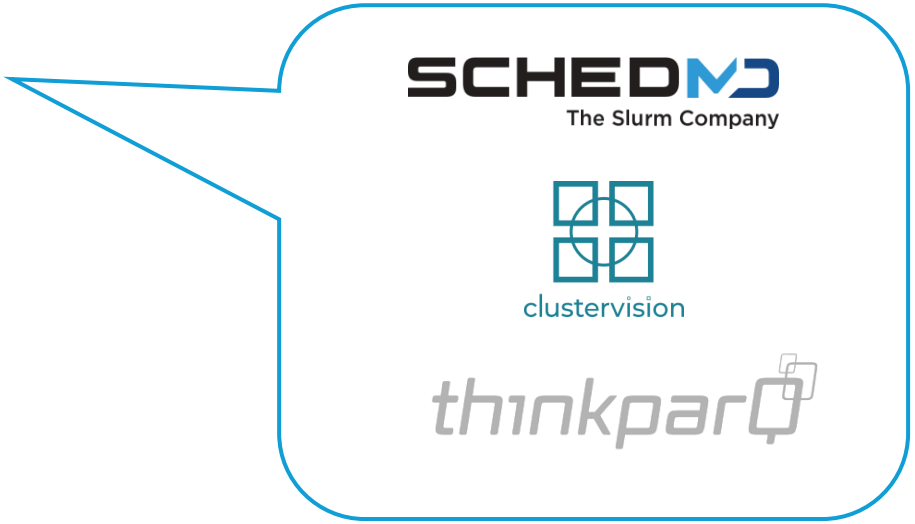
December 15, 2025 by NVIDIA Newsroom

まとめ

- モニタリングまでを統合したHPC環境の構築については、オープンな TrinityX の登場でようやくデファクトスタンダードが得られた感がある。
- もはやHPC環境はクローズドでは済まない。様々なデータソースや外部連携のため、ストレージがシステムのフロントエンドになり、コンピューティングがバックエンドな位置づけになっていくのでは。
- バッチジョブスケジューラは依然としてHPCにおける必須コンポーネント。一方で AI ワークロードをどう取り込んでいくのか、K8s に代表される AI 向けのインフラをどうHPCに活かすのか、試行錯誤が続いている。
- 共存不可能と思われたバッチジョブ環境と Kubernetes 環境の統合について、K3s や Slinky といったソリューションが出てきたことで選択肢が得られつつある。

イベント紹介

- **SCA2026/HPC Asia 2026**
  日程：2026年1月26日～29日
  会場：大阪国際会議場(グランキューブ大阪)
- **Super Computing Japan 2026**
  日程：2026年2月2日～3日
  会場：タワーホール船堀



Thank you!

sales@pacificteck.com