変革を支える技術・踏襲を支える技術

2025/02/03 Supercomputing Japan 2025 於: タワーホール船堀

Pacific Teck Japan G.K. Senior Engineer　森本　賢治

Pacific Teck
HPC and Machine Learning Experts

世界中の最先端の次を作る技術にフォーカス

- HPC/AI 環境におけるストレージ/コンテナ/ジョブ管理に習熟
- 日本を拠点にインドを含むアジア太平洋エリア(APAC)に製品を提供
- 英語 / 中国語 / 日本語のグロバールな言語でのサポートが可能
- 各スパコンや民間企業研究所等での採用実績多数

カテゴリー別 取扱製品

- ジョブ管理システム
  - ALTAIR
  - slurm workload manager
- HPC向けコンテナ
  - S PRO
  - S ENTERPRISE
- セキュリティ分析プラットフォーム
  - anchore
- ストレージソフトウェア
  - 並列ファイルシステム
    - BeeGFS®
    - BeeOND®
    - VAST
  - S3 オブジェクトストレージ
    - DATACORE
- S3クラウドストレージサービス
  - wasabi
  - cunoFS unleashing object storage
  - Blocky for Veeam®
- データマネジメント
  - iRODS®
  - TUXERA
- クラスターマネジメントシステム
  - Bright Computing
  - trinityX
- I/O Profiling 開発者用ツール
  - oneAPI
  - linaroforge
  - ALTAIR

スパコンを作るのに必要なミドルウエア群をワンストップで調達
ライセンス販売だけでなく、技術サポートを展開

変革期

Supercomputingは常に変革の最前線では？

- 様々な技術の実験場
- 生まれては消えていく技術や製品を数多く目にしてきた。

今の最先端が次の最先端を生む

- 先行投資と回収のジレンマ
- 研究？実用？

この数年のトピック

- AIと~~量子~~
- AI「で」何を計算したいのか？
- AI「が」何を計算しているのか？

一番変わっているのは

- 「欲しい正解」へのアプローチ方法？
- 「欲しい正解」の定義？

適用範囲の拡大

- 実用上の「欲しい正解」と、アカデミックでの「欲しい正解」の関係。
- 投資と回収のサイクルの短縮化。

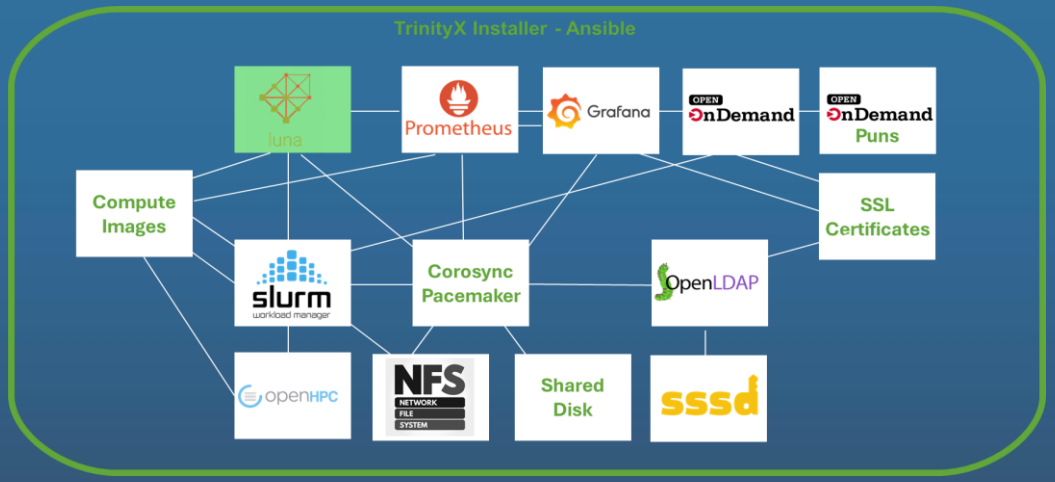
天気予報？
生成AIでイラスト描画？
小説やアニメ作成、
作曲？論文執筆？

システム構築について

Supercomputing環境を構築方法の変革

サーバーを購入するだけではクラスターは作れません。
システム構築・運用には膨大な工数がかかります。

1. 全台にOSインストール、設定
2. GPU・インターコネクタ用ソフトウエア導入
3. 個別サーバーのネットワーク設定
4. 共有ストレージ作成、マウント設定
5. 共有認証基盤（LDAP等）作成、接続設定
6. ジョブスケジューラの導入、キュー設計
7. 並列計算ソフトウエアのインストール、設定
8. アプリケーションのインストール、設定
9. GUI のインストール、設定
10. 監視・モニタリングのインストール、設定
11. ユーザー管理、教育、トラブルシュートなど
12. システムアップデート、セキュリティ対応など



コンポーネントはOSSで揃うが、システムに組み上げることは膨大な工数とノウハウ、経験が必要。もはや力業は限界に近く、自動化するツールが必須。

HPC・AI クラスター（スパコン）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

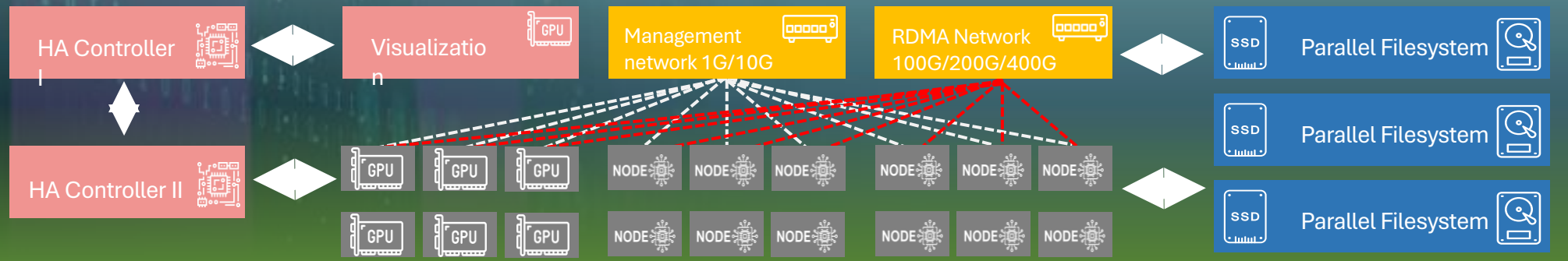
Environment Modules | Security & Node Logs | HW/IPMI Management

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

ここを整備してハードウエアを
統合し、システムを構築・運用

様々なコンポーネント群が
個々のシステムを特徴づけてきた

⇒ 統合管理ツールを使いたい



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
……

柔軟なシステム構築・マネジメント

trinityX

最初に Bright Cluster Manager を作った、ClusterVision社による、
新設計のHPCクラスター構築・管理ツール。

OpenOnDemand の上にユーザーポータルのみならず、管理ツールUIも統合。

充実のGPUモニタリング、ジョブ実行状況分析に寄与する様々な管理画面。

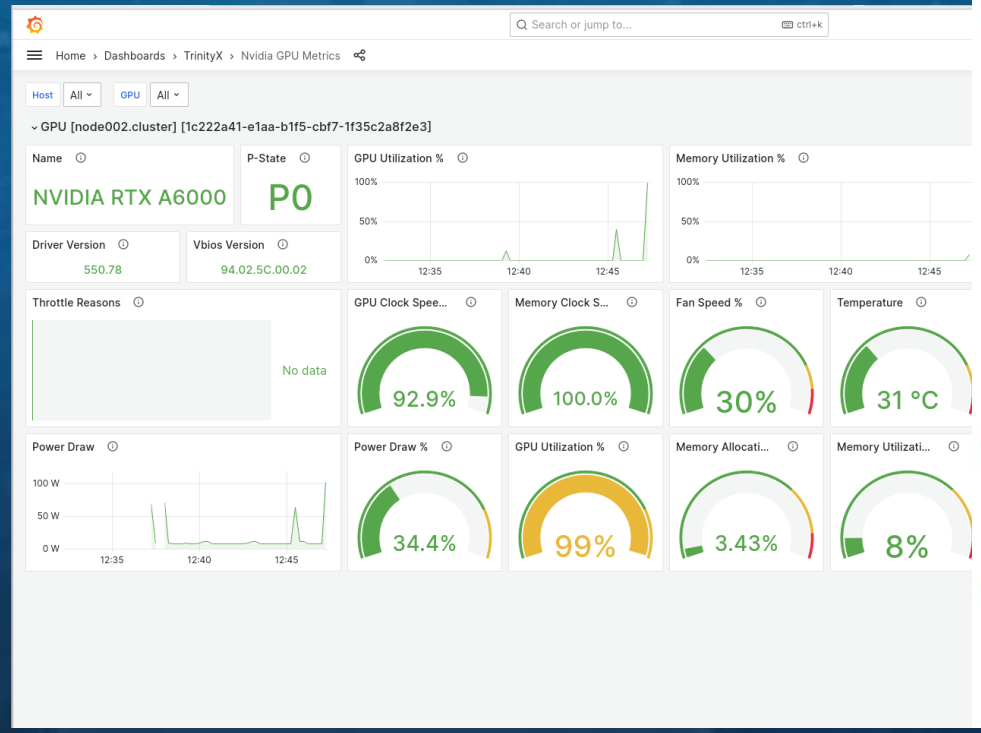
計算ノードOSイメージを数千ノードまで一斉に展開可能。システムのOS更新も円滑に。

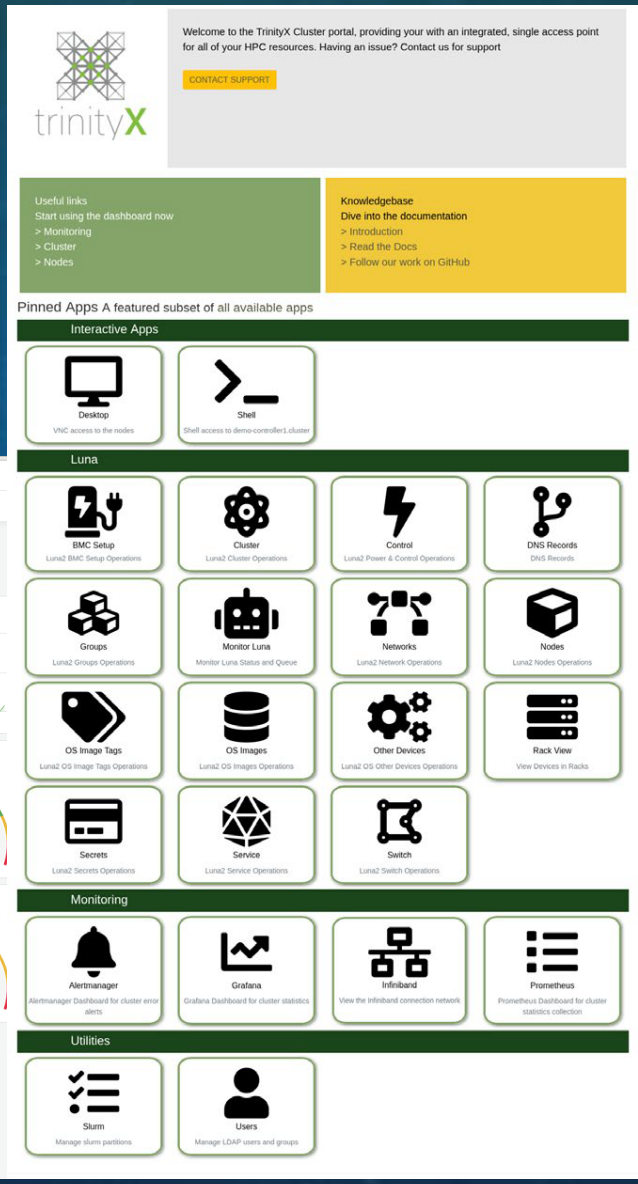
OPEN OnDemand

Grafana

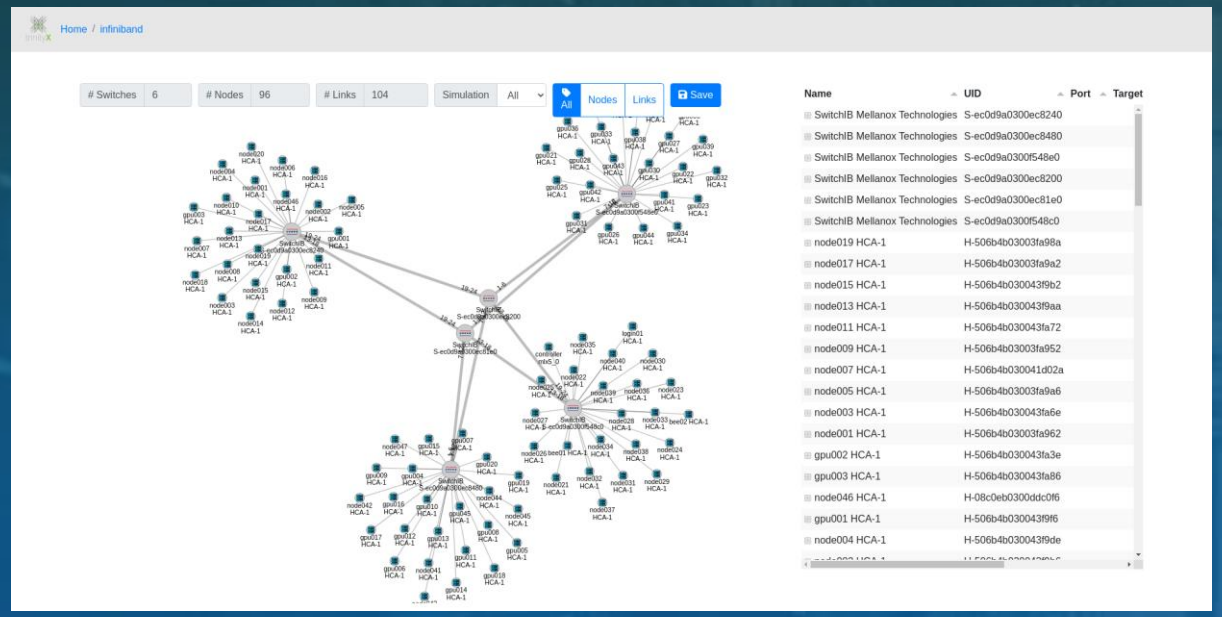
ANSIBLE

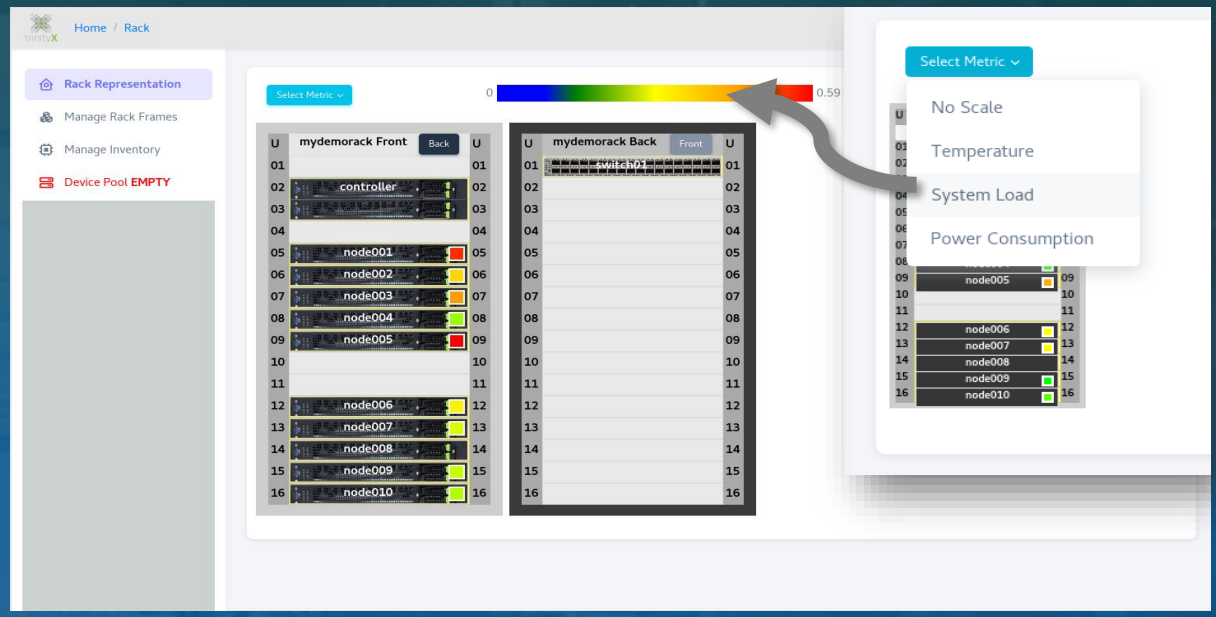
Prometheus

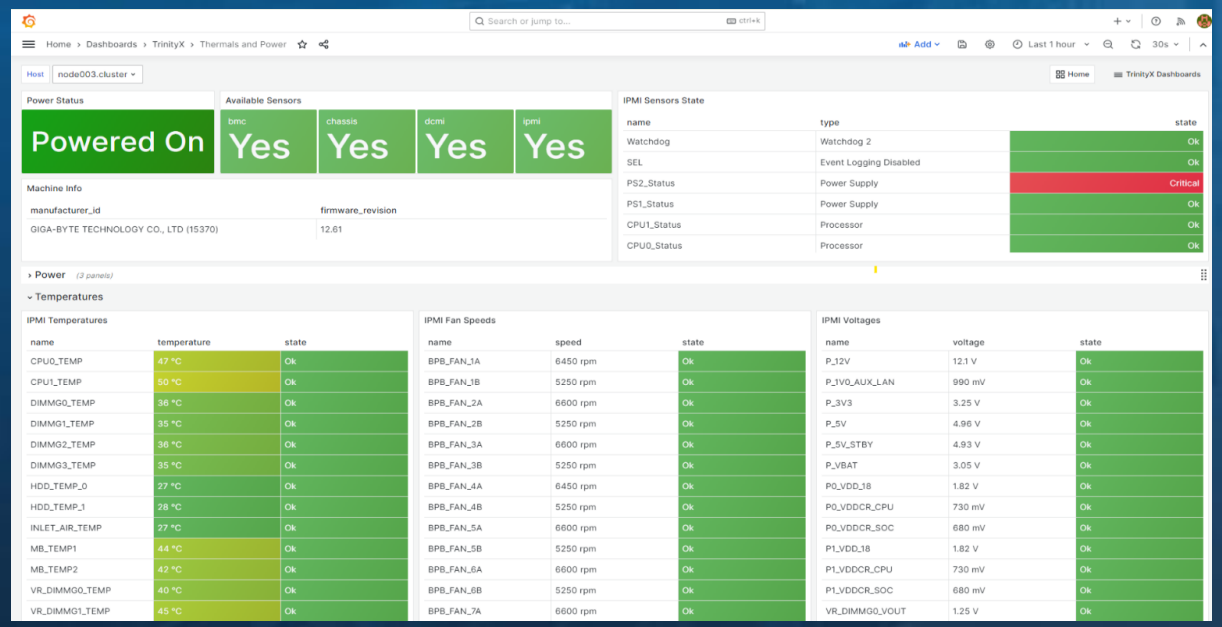


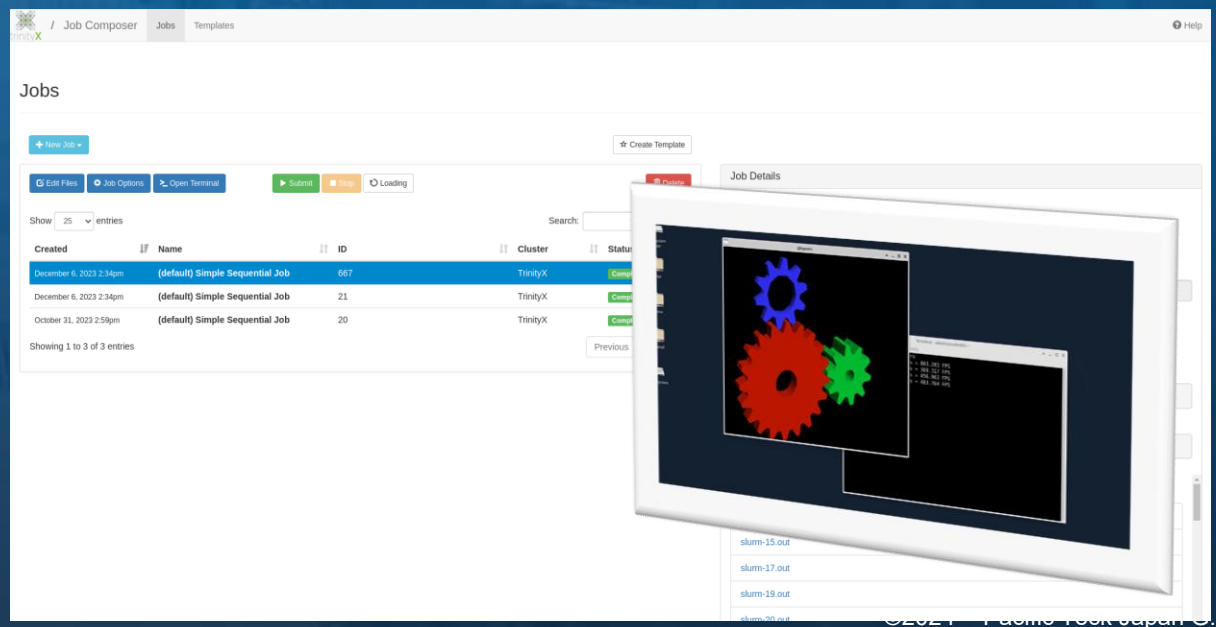


管理者にも利用者にも直感的なUI





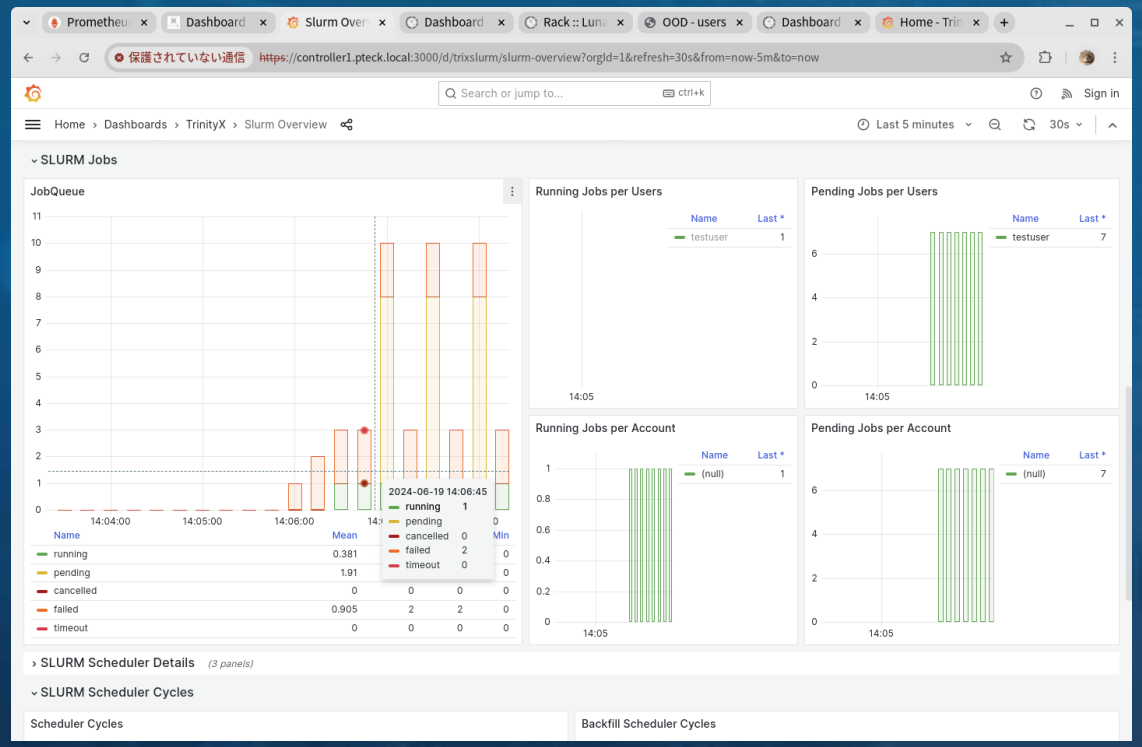


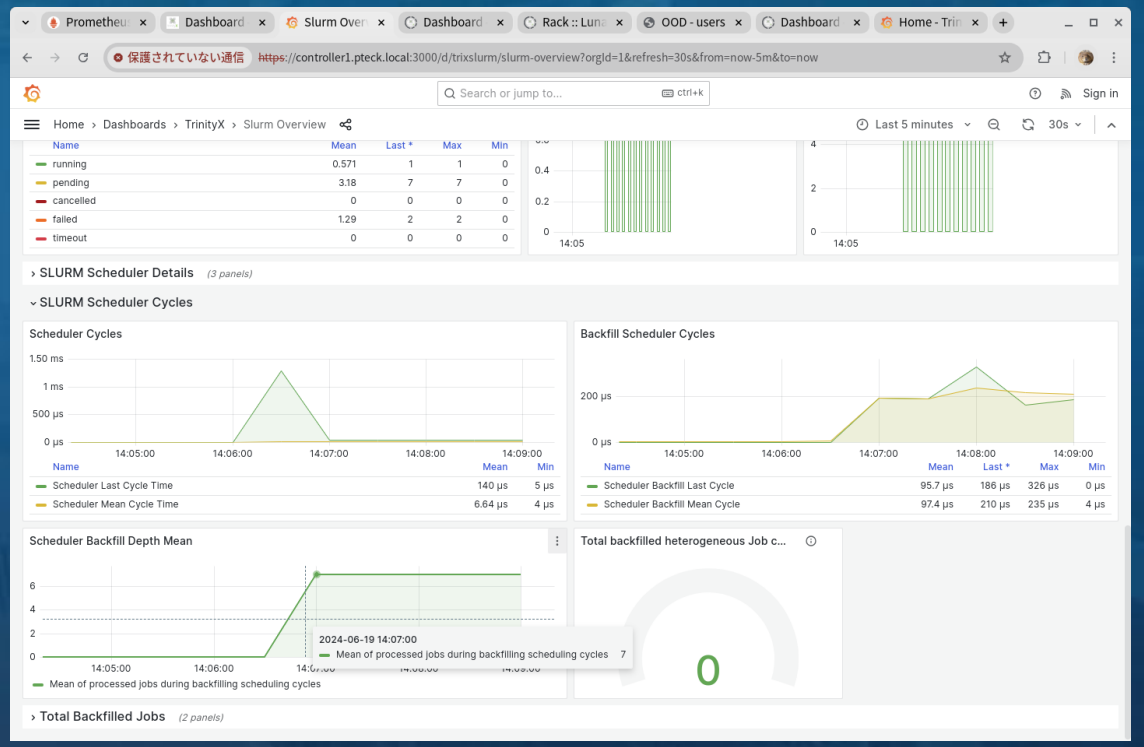


ジョブスケジューラの稼働状況の分析

ジョブの待ち時間等のデータを分析するダッシュボードがデフォルトで作成されており、利用状況に合わせたスケジューリング設定の調整だけでなく、将来的なリソース追加の計画策定にも寄与します。

slurm 設定の構築・検証ツールも付属しているため、GUIから必要な作業が完結します。





アプリケーション実行環境について

アプリケーション環境構築の変革

準備

- ユーザーが使うアプリケーションを事前に募り、必要なランタイムをリストアップして、共存できるようにすり合わせを行う。

導入

- リストアップしたアプリケーション、ランタイムに対し、それぞれ必要なバージョンを並存させ、切り替えて使う機構を用意する。

運用

- アップデートが必要になった際は、管理者に要望を出し、メンテナンスのタイミングで追加してもらう。

- 多様化する用途に合わせ、多彩なアプリが登場。高頻度でアップデートを繰り返す。
- システム全体でそれらの環境をすり合わせるのは、既に現実的でない。
- NGCをはじめ、コンテナを用いて実装をデリバリーする仕組みの充実。
- ユーザーの方が管理者より、アプリの実行環境の詳細に詳しい。

SingularityCE・SingularityPRO・Singularity Enterprise

SingularityCE

- オープンソースで提供され、有償サポートはありません。
- 新機能は最初に実装されますが、セキュリティ対応や修正は最新版にのみです。

SingularityPRO

- Sylabs社による検証・署名済みバイナリを提供します。
- 長期サポート(リリースから2年)・機能追加・プラグイン開発を依頼できます。
- 脆弱性の公開前に対応済みバイナリが提供され、バックポートにも対応します。

Singularity Enterprise

- コンテナランタイムとしてはSingularityPROを提供します。
- Kubernetes上に、RemoteBuilder, KeyStore, Libraryを提供します。
  EKS, GKEもサポート。ただし、Kubernetesの運用はユーザー責任です。
- 自社・自サイトの認証基盤を統合可能です

SingularityにはKubernetesやOpenShiftのようなオーケストレーター実装がありません。
ただし、OCIイメージを取り込んだり、SIFをOCI bundle として見せたり、そこからcrun/runcを使って動作可能。

SIF利用による優位性

Singularityという概念は、「ソフトウエアとしてのSingularity」と「規格としてのSIF」、両方からなります。
コンテナエンジンとしてのSingularityのみならず、SIF自体にも際立った優位性があります。

高い可搬性

- シングルファイルで保存するため、持ち運びが用意。
- 通常のNASでの管理も可能。
- リポジトリに依存せず、電子署名による同一性を保証。

低フットプリント

- 内部はSquashFS。tar+gzipを直接マウントしているイメージ。
- オンメモリでアクセスでき、事前に展開する必要がない。メモリやCPU負荷も極めて小さい。

ハイパフォーマンス

- ファイルアクセスはローカルのオンメモリで行われ、共有ファイルシステムへの、メタデータアクセス負荷を極小化できる。
通信負荷も下げられる。

特に昨今、AIのワークロードがPythonベースで実行されるため、個別環境がユーザーホームディレクトリに展開。共有ファイルシステムに高負荷を生み、大きな問題となっている。
例）PyTorchはimportするだけで1000回もファイルをオープンする。

- ホストの環境とは独立にランタイム環境を固定できる。
- 古い実行環境の保存だけでなく、既存システムが対応できない、最新の環境を試すという、踏襲・変革どちらの方向にも有用。

参考：弊社ブログ
Singularity利用による共有ファイルシステムの負荷軽減

Pros/Cons

Pros

SIFを使うメリットは前項の通り、直感的な可搬性やI/O負荷の低減など、極めて高い。

ファイル名で管理されるため、誤選択やコピーミス等で、再現性トラブルだけでなく、セキュリティリスクとなり得てしまう。

Cons

リポジトリ＋タグ指定による実行で解決

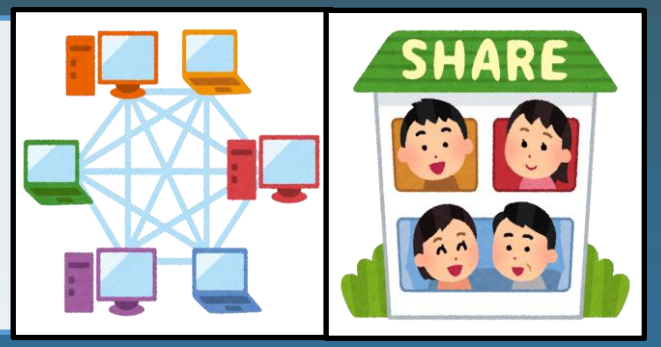
- SIFファイルの直接指定ではなく、リポジトリのイメージを指定する。
- $ singularity exec ./sample.sif appli → ×
- $ singularity exec library://myrepo/smp:2.1 appli → ○
- ローカルにSIFでキャッシュされ、1ファイルのメリットを失わない。

環境の統合と分割

コンテナを利用したリソースの切り出し

リソースの統合と分割

- とにかく巨大な1システムが欲しかったHPC
- 大きくなりすぎたマシンリソースを分割し多目的に利用可能に
- マルチテナントの要求



コンテナによるリソース分割

- dockerとKubernetesの台頭
- バッチジョブスケジューリングとの競合

落としどころ

- Hadoop ブームの時代から試行錯誤が続いている
- TrinityX の K3s（ライトな K8s 実装）対応を予定。
- SchedMD の Slinky による slurm との共存。

まとめ

- これまで培われた蓄積をさらに発展させ、従来を踏襲しつつ、これからの変革に耐えうる環境を作る。そうした流れに有用と思われるプロダクトをご紹介しました。

- 皆様の「欲しい正解」を、もう一度考え直すきっかけとなりましたら幸いです。

Thank you

お問い合わせはこちらまで
sales@pacificteck.com