

コンテナ技術を用いた教育情報システムの構築

- 宮平 賢
 - 琉球大学工学部工学科知能情報コース
- 河野 真治
 - 琉球大学工学部

学生が自由に利用できる教育情報システムの構築

- 情報通信技術の普及に伴い様々な学習環境が学生にとって必要となる
- その学習環境として、Virtual MachineやContainerがある
 - 実行には高性能なPCが必要な場合がある
 - クラウドサービスもあるが、無料だと制限がある
- 学生の学習環境として、コストを支払う必要のない環境を提供したい
- 今年度はシステム更新があり、学科サーバにSSDとGPUが搭載された
 - リソースを最大限利用できる教育情報システムが必要となる

これまでの教育情報システムの問題点

- VM貸出サービスの申請方法がわからない
- 貸出サービスが存在することが周知されていなかった
- 旧システムにはGPUが搭載されていなく、研究室ごとの機器、クラウドサービスが利用された
- VMイメージの保存場所として使用されたGFS2はロックマネージャーの影響が大きかった

教育情報システムの構築

目的: 開かれた見通しの良い教育情報システムを構築する

目的を達成するための構成要素

- VMベースからコンテナベースへ移行
- サーバに利用者のログインを許し、GPUなどの利用を促進する
- コンテナによる環境で課題や演習を可能にする
 - 環境構築の手間をかけずに演習を行える
- GFS2を現代的な多重度を持つ分散ファイルシステムに置き換える
- Wordpressを静的HTML生成に置き換える
- システム管理をRedmine/SlackからScrapbox/Mattermostに置き換える

VMベースからコンテナベースへ移行

- 旧システムはVMベースでシステムが構築されていた
- サービスごとにVMがあり、管理に手間が掛かる
- VMベースではサーバに搭載されるGPUを有効活用できない
 - 1つのVMに1台のGPUが必要になる
 - 希望する学生全員に割り当てきれない
- サーバのリソースを効率よく利用できるコンテナへ移行する
- VM貸出サービスは継続する

コンテナ環境の導入

- マルチユーザで利用できるPodman、Singularityを導入する
- Podman
 - Dockerと同じCLIを提供する (`alias docker=podman`)
 - nvidia-dockerの設定を行えばGPUを利用できる
- Singularity
 - ホームディレクトリ、/tmpなどがコンテナにマウントされる
 - コンテナにユーザ権限が引き継がれる
 - コンテナから容易にGPUを利用できる
- コンテナによる環境で課題や演習を可能になる

コンテナエンジンの動作改善

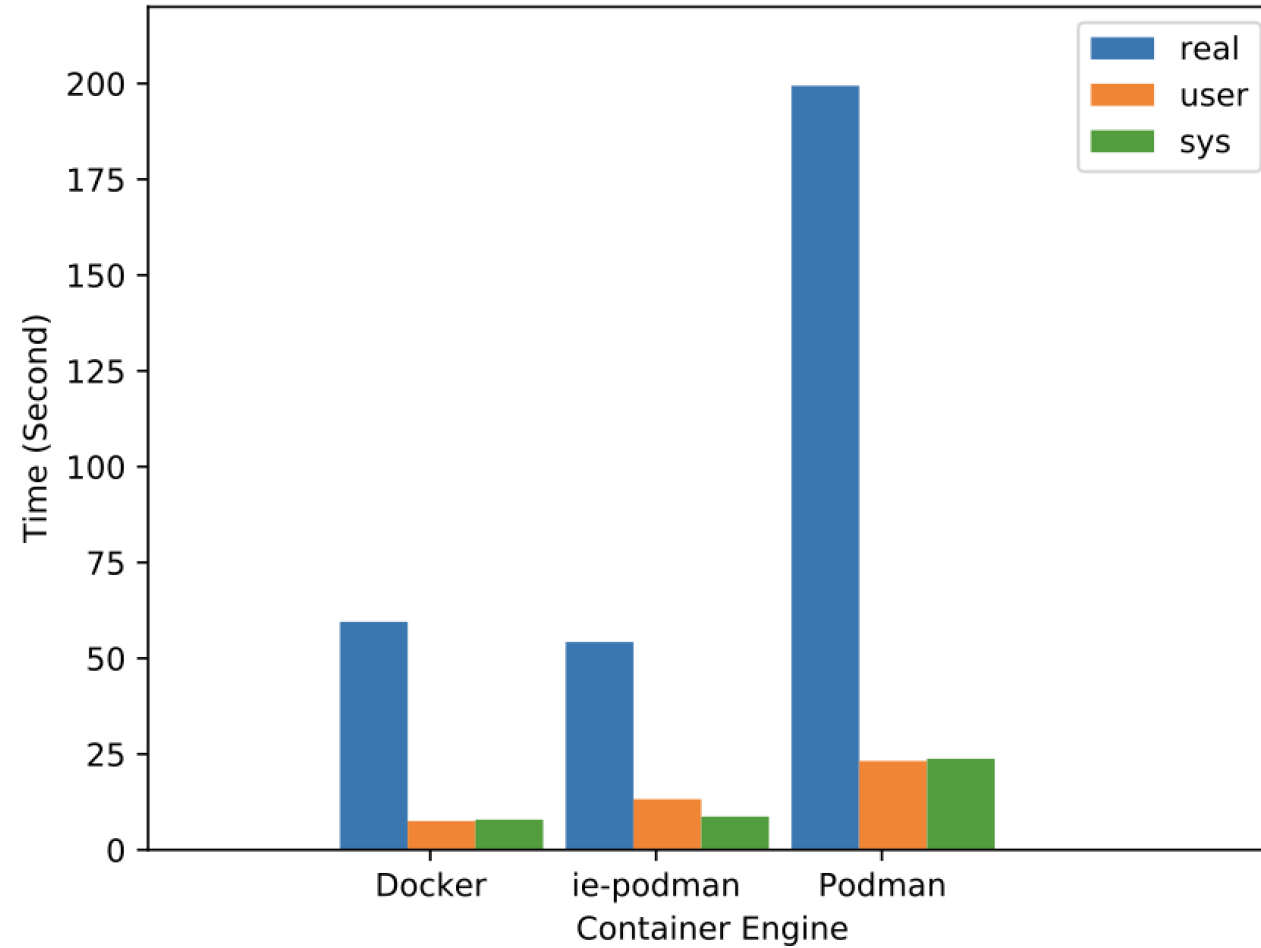
- Podman
 - rootlessではイメージの作成やコンテナの作成が遅い
 - rootlessでは実行できない機能がある
 - IPアドレスの割り当て
- Singularity
 - イメージの作成に時間がかかる
 - ビルド中にエラーが発生すると、一から再開する必要がある

Rootless Podmanの遅さへの対策

- rootfullのPodman wrapperであるie-podmanの導入
- ユーザのUID、GIDを取得し管理を行う
 - 他のユーザのリソースを操作できない
 - コンテナ名、イメージ名をユーザ名で補完する
- SSD上にイメージ等を保存し、高速化を図る
- イメージのBuild速度を比較する
 - Docker
 - ie-podman (Podman rootfull wrapper)
 - Podman (rootless)

Build速度の比較

- Timeの値が小さい方がBuild速度が早い



ie-podmanの機能 1/2

- Podmanのすべての機能を提供するのではなく、一部機能のみを提供する

コマンド	機能
build	Containerfileの指示に従いイメージを作成する
cp	コンテナにファイルを送信する
exec	起動中のコンテナでプロセスを実行する
images	コンテナイメージの一覧を表示する
info	コンテナの情報を表示する
logs	コンテナのlogを表示する
ps	起動中のコンテナの一覧を表示する

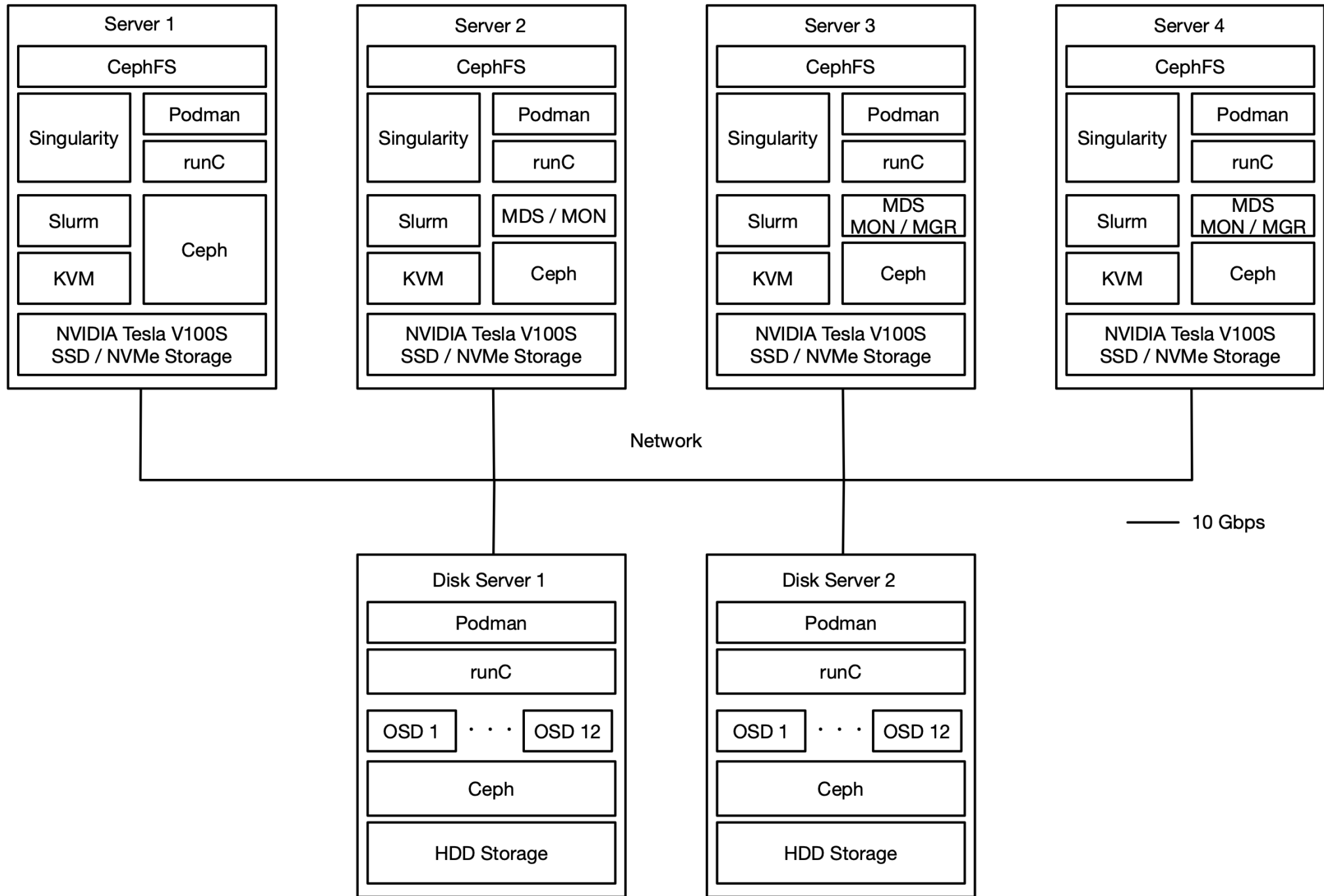
ie-podmanの機能 2/2

- 独自機能を追加
 - registryやsifなどサブコマンド
 - コンテナ作成時に指定できる `--ip` オプション

コマンド	機能
registry	学科のレジストリの操作を行う
rm	コンテナを削除する
run	コンテナを作成する
sif	イメージをsifファイルに変換する
start	コンテナを起動する
stop	コンテナを停止する

ジョブスケジューラとファイルシステムの導入

- Slurmを採用
 - プログラム実行のリソースの管理を行う
 - 利用方針
 - 計算リソースの利用効率を上げる
- Cephを採用
 - 自己修復、自己管理機能を搭載するため信頼性が高い
 - Monitorを3台のサーバで構成することで、最大1台の停止は許容される
 - 柔軟なアクセス方法を提供する
 - Object Gateway
 - ブロックデバイス
 - POSIX互換のファイルシステム

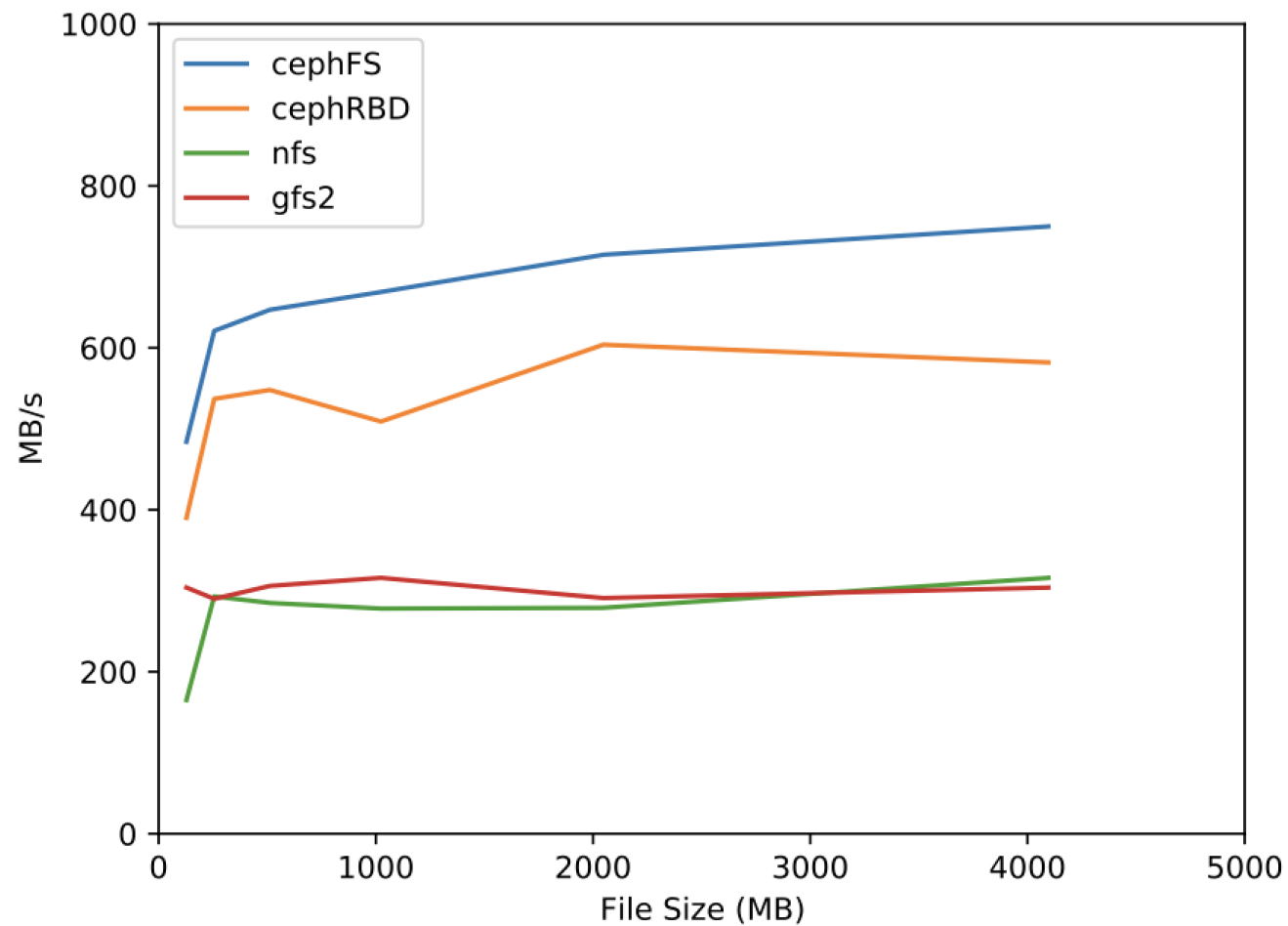


ファイルシステムの比較 1/2

- 実験概要
 - `dd` コマンドを使用し書き込み速度を比較する
- 比較するファイルシステム
 - GFS2
 - 旧システムのVMイメージの保存場所として利用された
 - NFS
 - 旧システムのホームディレクトリとして利用された
 - CephFS
 - 新システムのホームディレクトリとして利用する
 - CephRBD
 - 新システムのVMイメージの保存場所の1つとして利用する

ファイルシステムの比較 2/2

- MB/sの値が大きい方が性能が良い



まとめ

- 今年度のシステム更新で教育情報システムの構築を行った
 - VMベースからコンテナベースへの移行ができた
 - 学生が自由にサーバのリソースを利用できるようになった
- Rootless Podmanの遅さ補うために作成したie-podmanの評価を行った
- Cephと、これまでのGFS2との書き込み速度の比較を行った

今後の課題

- 教育情報システムの周知
 - Jobの投下やリソースの要求方法
 - ie-virsh、ie-podmanの使用方法
- ie-podmanのネットワーク構成の見直し
 - プレフィックス長が24のため、最大254個のIPアドレス
 - コンテナを停止したまま使用されない場合は削除する必要がある
- 基幹サービスすべてのコンテナ移行
 - まだVMで動いている基幹サービスがある
- 監視システムの導入
 - Prometheusによるコンテナの監視