# 初心者のためのクラスタ　II

## クラスタの構成要素

有限会社イワタシステムサポート　　岩田　進吉

2026年2月2日(月)

# 初心者のためのクラスタII

クラスタの構成要素

HPCクラスタでのOS

Linux、Windows

ソフトウェア

ジョブ管理、ユーザ管理、データ管理

ジョブ管理

slurm、PBSProfessonal、OpenPBS、LSF、ーーー

SunGridEngine　ー>　UNIVA　ー>　Altair買収？

HPCPack　／　2019が最新でUpdateされている

ユーザ管理、ファイルサーバ

ノード間通信

ハードウェア、ソフトウェア

Infiniband、Ethernet、専用インターコネクト

ＣＰＵの種類と選択

x86、arm、tron、ーーー

# クラスタシステムの構成要素

クラスタは

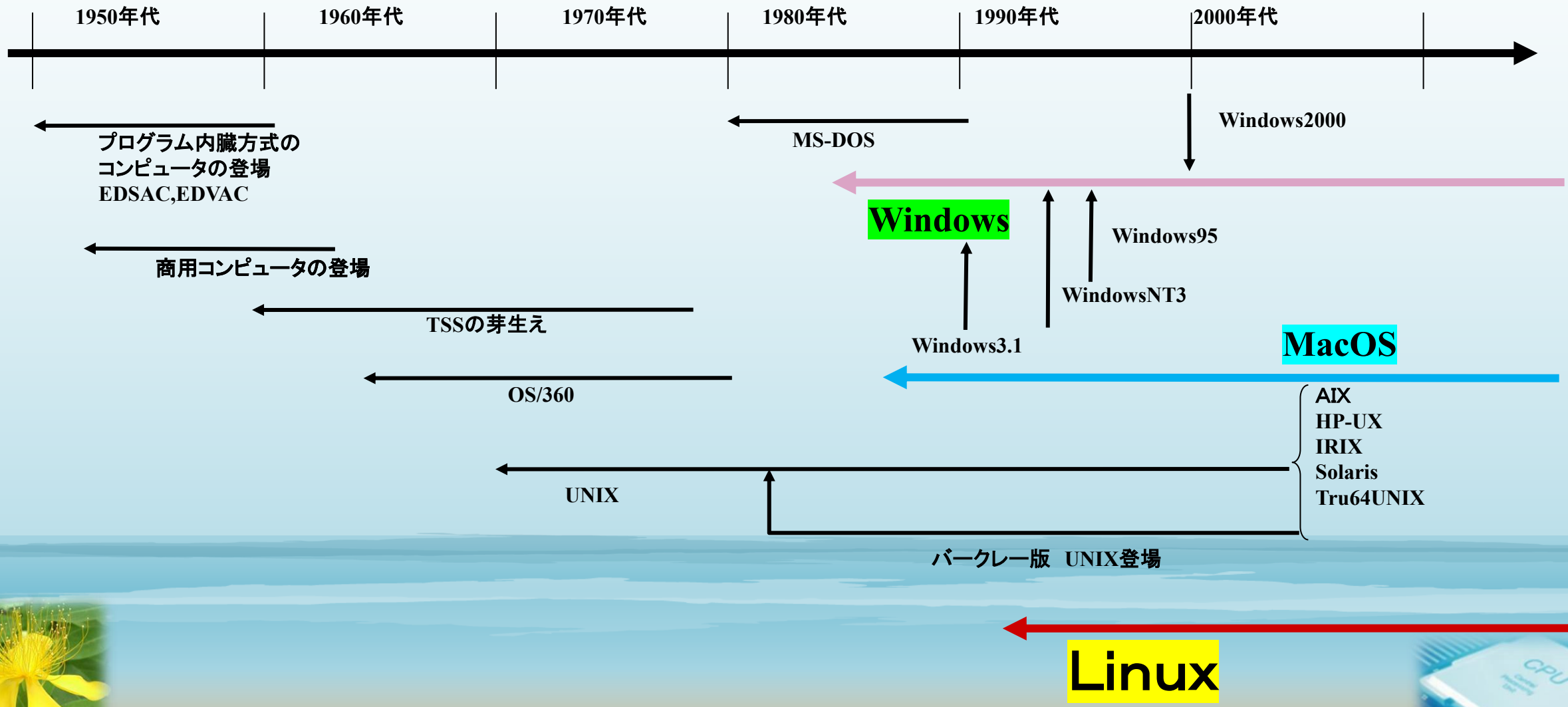
- 高性能コンピュータ（OS、高速メモリ、高速ディスク、ーーーー）
- 共有ファイルサーバー、ユーザ管理／管理ノード
- 高速な通信（現在はInfiniBandが一般的）
- 並列アプリケーション
- ジョブ管理システム（ slurm、PBSProfessonal、LSF、————等）

で構成されており、クラスタはジョブがコンピュータを跨って効率的に実行できる環境を提供します。この各要素についてご説明いたします。

# クラスタの構成要素

## HPCクラスタでのOS

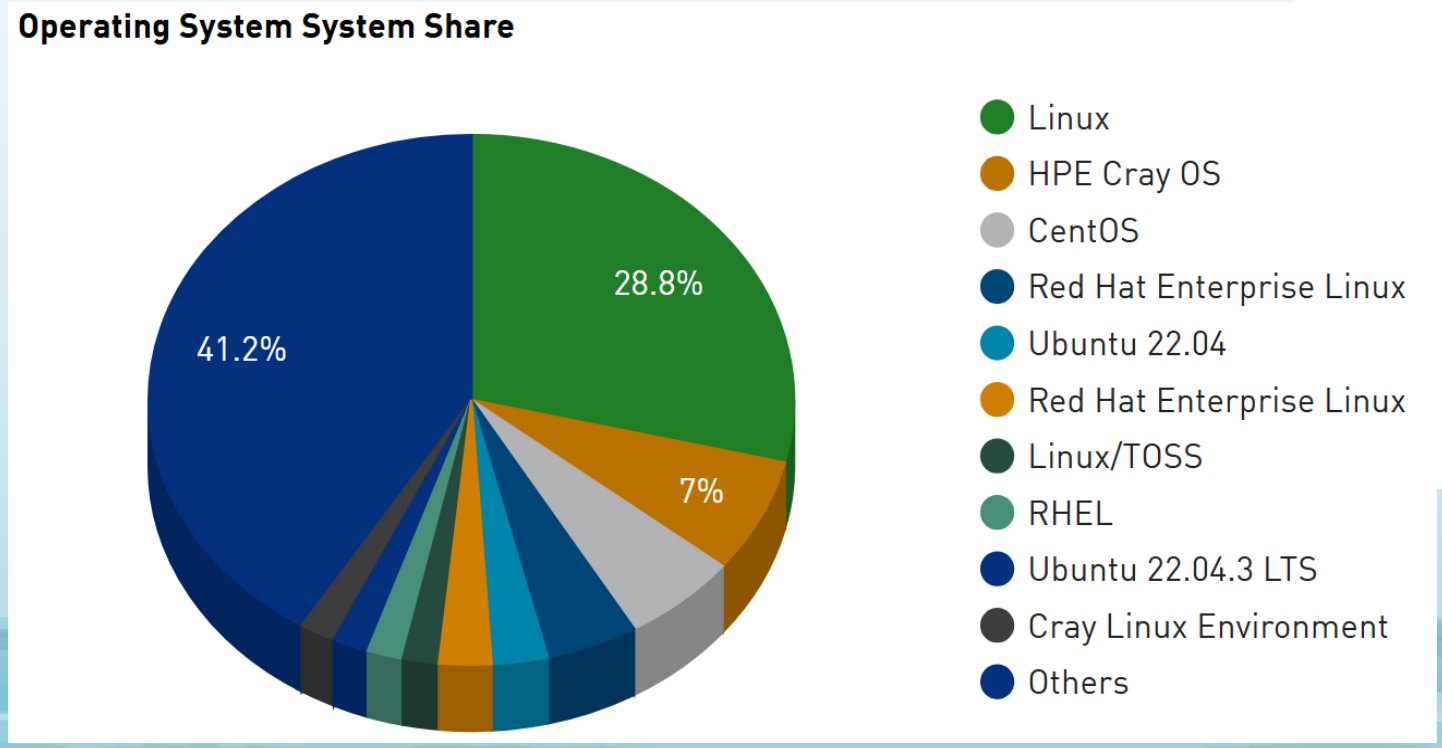
# ＯＳの動向


1950年代
1960年代
1970年代
1980年代
1990年代
2000年代
プログラム内臓方式の
コンピュータの登場
EDSAC,EDVAC
MS-DOS
Windows2000
Windows
Windows95
WindowsNT3
Windows3.1
商用コンピュータの登場
TSSの芽生え
OS/360
MacOS
AIX
HP-UX
IRIX
Solaris
Tru64UNIX
UNIX
バークレー版　UNIX登場
Linux

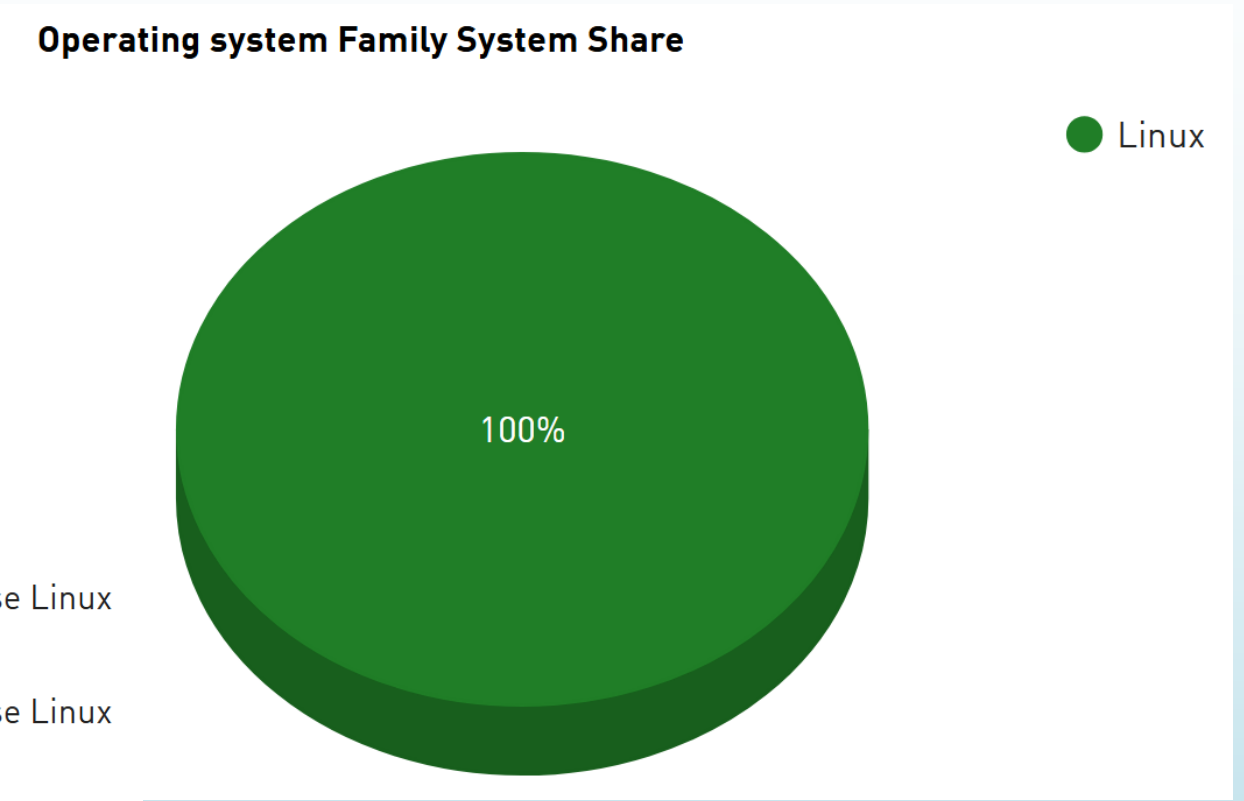
# Top500から見たスーパーコンピュータの動向

https://top500.org/

2025年11月




OS別比較




OSタイプ比較

# クラスタにおける
# ユーザ管理とファイルサーバ

# NISの概要

## NIS (Network Information Service)

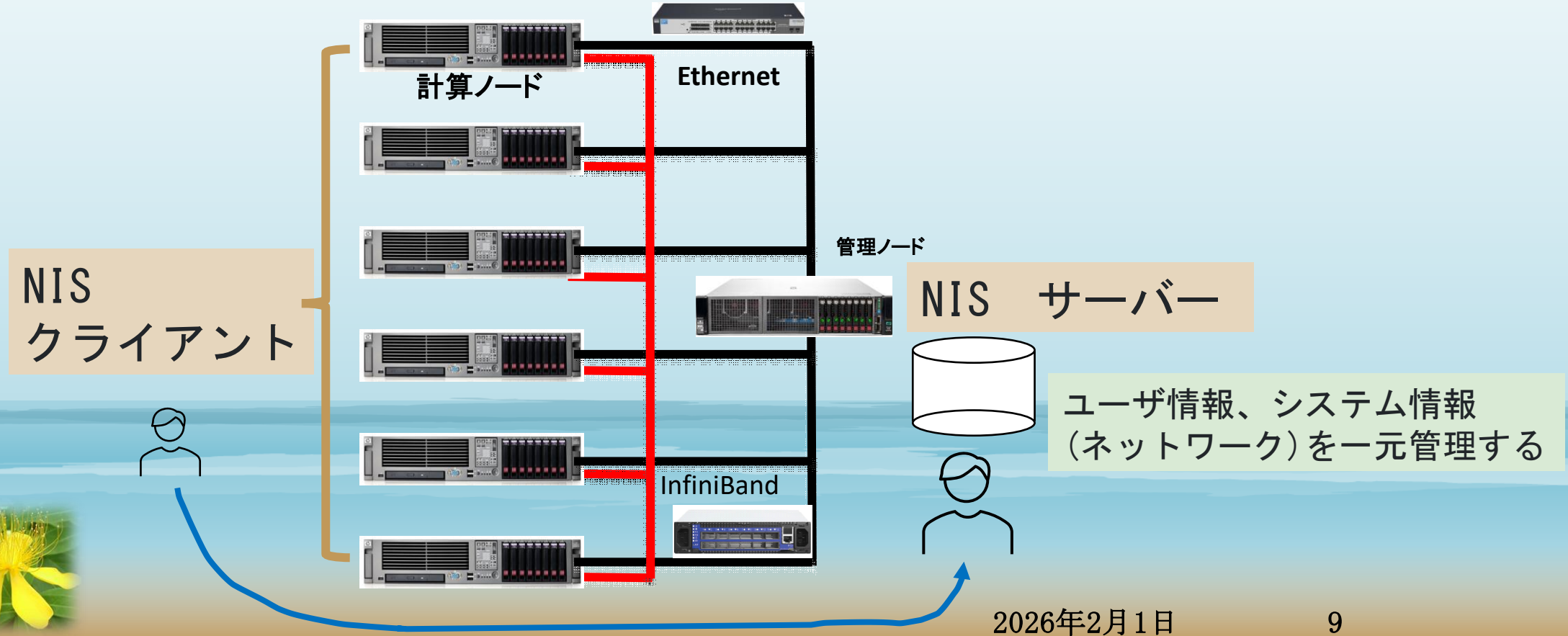
ユーザー情報（ユーザー名やパスワード）をネットワーク上のコンピュータで共有するサービス。

複数のコンピュータを協調して利用するクラスターシステムでは、/homeディレクトリ（個人フォルダ）をNFSでクラスター内のすべてのコンピュータで共有。

ユーザー名やパスワードも１台のマシン（NISサーバ）からパスワード情報を各コンピュータに配布し、同じパスワードですべてのマシンにログインできる環境を作成。これにより分散並列ソフトウェアであるMPIアプリケーションをスムーズに実行することができる。

# NISのイメージ

NIS にはサーバー、クライアントがあり、サーバーでユーザ情報、ノード情報等を管理し、クラスター構成が変わっても、サーバーの情報を更新するだけで、全てのコンピュータに反映されます。



# NFSの概要

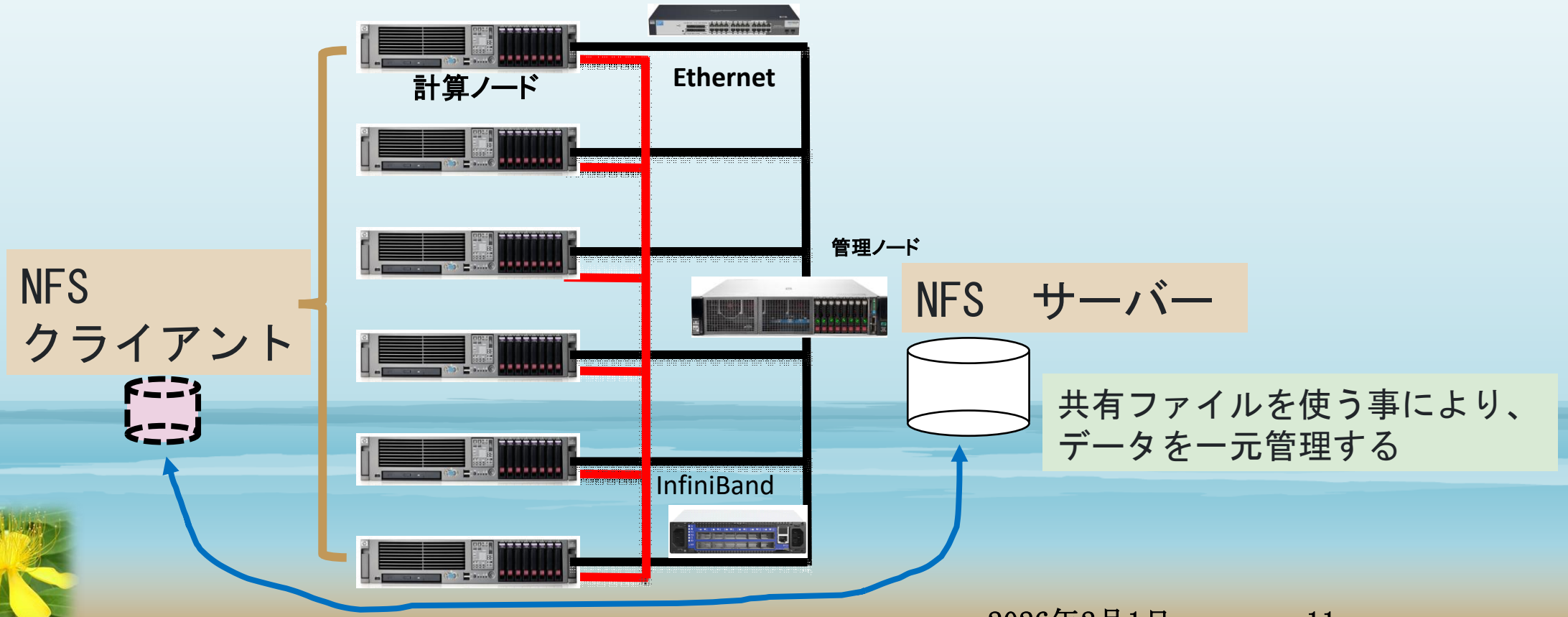
NFS (Network File System)とは、サーバのファイルシステムをネットワーク上のコンピュータで共有するサービスです。

複数のコンピュータを協調して利用するクラスターシステムでは、ファイルサーバに共有ディスクをクラスター内のすべてのコンピュータで共有し、同じデータ領域を使って計算処理を進めます。

これにより分散並列であるMPIアプリケーションをスムーズに実行することができます。

# NFSのイメージ

NFS にはサーバ、クライアントがあり、サーバ側で大容量のディスクを用意してクライアントから利用できるようにします。



# ジョブ管理システム

# クラスタの利用方法

## クラスタ利用な一般的な形態／会話型で使うのは非効率

1. ワークステーションもしくはデスクトップPCでモデル作成、条件設定等を行い、解析計算の前処理を行う。モデルデータは共有ファイルシステムにおいて処理を行う。
2. 前処理が終了したら、利用した前処理アプリケーションから自動的にクラスタシステムにジョブを投げるか、ジョブ実行スクリプトを作成してジョブを投入する。
3. ジョブの終了はメールでジョブ投入者に知らせるか、ジョブ投入利用者の端末に実行状況を表示してわかるようにする。
4. ジョブが終了したら、共有ファイルに結果が書かれているので、その結果を利用して後処理を行う。

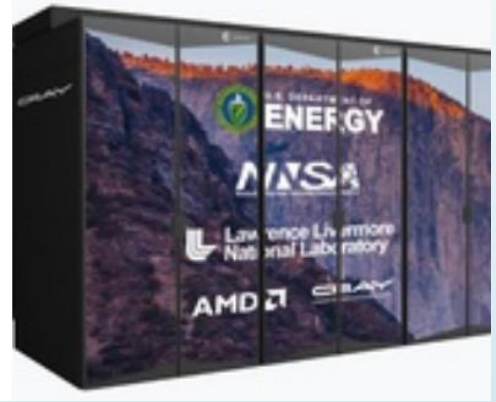
# クラスタの利用イメージ

１．社内でモデル作成
　　データの作成
６．解析結果の評価

２．入力ファイル転送
　　実行スクリプト転送

ネットワーク

クラスタ




３．クラスタに入力してジョブを実行

４．バッチ実行処理

５．結果の転送

ネットワーク

# HPC分野のジョブの特徴とジョブ管理の必要性

クラスタシステムで計算する場合は、CAE解析で計算時間の長いものは数日から数週間もかかるモデルがあります。

この為にはジョブを効率的に実行する必要があり、効率的にジョブを実行するためにジョブ管理ソフトを使います。

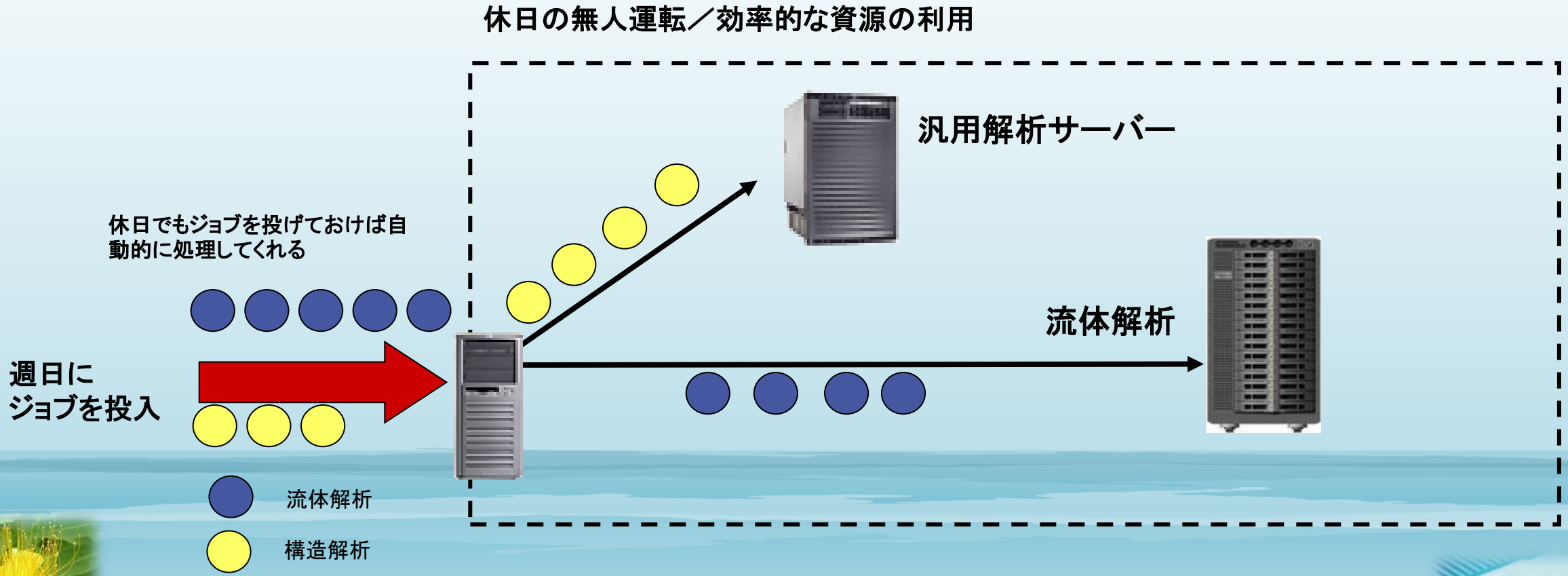
ジョブの投入方法は「ジョブ管理ソフトウェア」として

slurm、PBSProfessonal、－－－

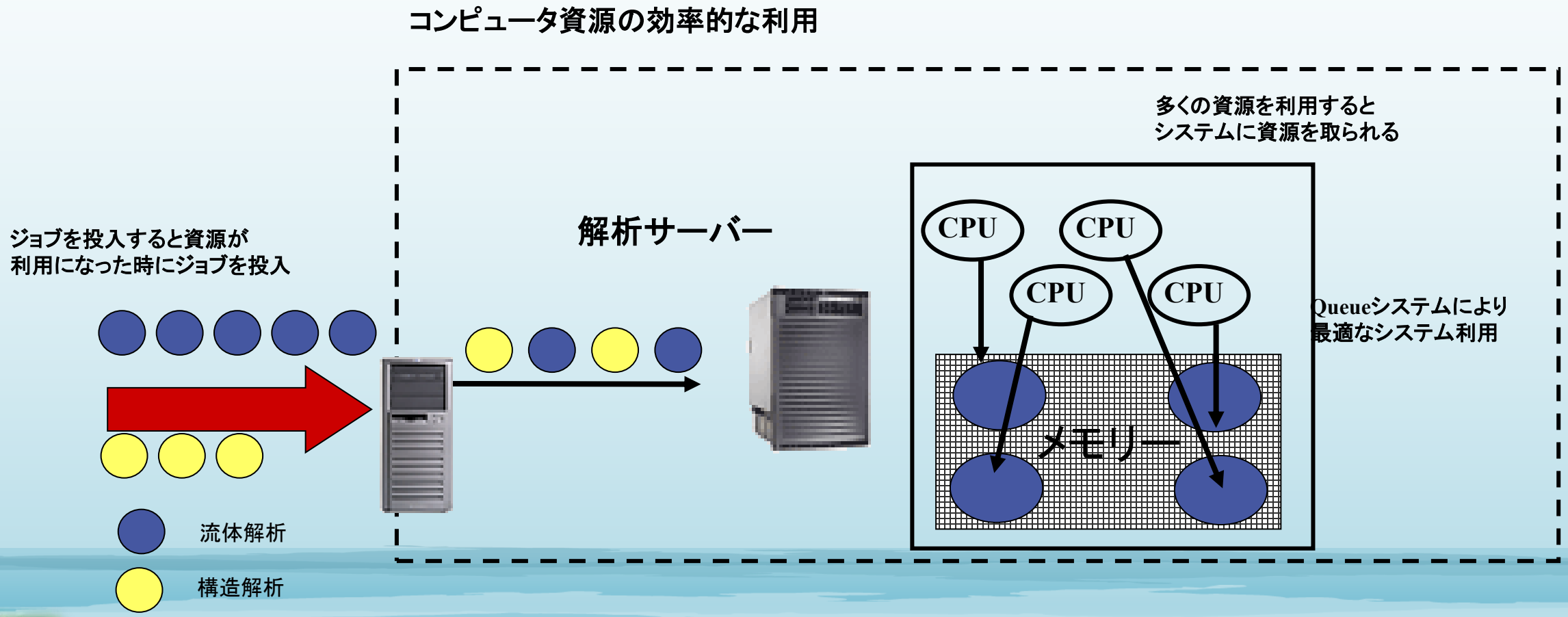
等を利用することになります。

# ジョブ投入システムの主な役割

## 連続的なジョブの投入



# CPU、メモリの効率的な利用



# 優先順位による仕事の効率化

急ぐ処理を先頭に移動

優先順位、利用する資源量の設定

汎用解析サーバー

流体解析

流体解析

構造解析

# ジョブ管理に使われるソフトウェア

- Slurm
  無償のジョブ管理ソフトウェア。米国の研究機関、コンピュータメーカ等で共同開発されたもの。TOP500では60%近くのシステムで使われている。サポートとしては有償でサポートしている会社もある。最新のVersionは25.11.1

- PBSProfessonal
  Altair社が販売している有償のソフトウェア。Linux、Windowsをサポートする

- OpenPBS
  Altair社が出しているPBSProfessonalの無償版／Linux版のみで非商用

- LSF(IBM Spectrum LSF)
  昔、カナダのPlatform社が出していたLSFをIBMが買い取って有償版として出している。多くのLinuxやWindowsシステムでもサポートしている。

- HPCpack
  MicroSoftが出しているWindows　OS向けのジョブ管理システム。2016からはLinuxもサポートするようになってきている。最新版は2019で古くなっている。

- Altair Grid Engine ／　SGE(SunGridEngine)
  Sunが昔作ったGridEngine（ジョブ管理ソフトウェア）。今はAltairが管理、販売。

# slurm

Slurm（Simple Linux Utility for Resource Management）は、LinuxおよびUnix系のカーネルのためのフリーでオープンソースなジョブスケジューラーです。世界中の多くのスーパーコンピューターやコンピュータークラスタで使用されている。Slurmは3つの主要な機能を提供しています。

1. 計算を実行するユーザに対してリソース（コンピューターノード）への排他的・非排他的なアクセスを割り当てる機能。
2. 割り当てられたノードの集合上でのジョブの開始、実行、モニタリング（MPIなどの並列ジョブでよく使用される）を行う機能。
3. 待機中のジョブのキューを管理することで、リソースへの競合を解決する機能。

Slurmは、TOP500の約60%のスーパーコンピューターでワークロードマネージャーとして使用されています。

Slurmは当初、主にローレンス・リバモア国立研究所、SchedMD（Slurm Support and Development）、Linux NetworX、ヒューレット・パッカード、Groupe Bullによる共同開発。開発は2002年に始まり、現在のVersionは20.11.4が最新のVersionになります。

世界中の100人以上の開発者がプロジェクトに貢献しております。多くの巨大なコンピューターセンターの要求を満たす性能を持つ、洗練されたバッチシステムとして進化

開発経緯は、LLNLとしてはジョブ投入システムとしてQuadrics RMSが良いと考えましたが、2002年当時、通信ネットワークのサポートがQuadrics Networkしかサポートしていなかった、そして他のツールは以下の理由で採用が見送られ、新規に開発することになったようです。

1. Portable Batch SysTem(PBS) --- 移植性は高いが、大規模システムに向いていなかった
2. IBM LoadLeveler --- ポータブル性がなく、大規模システムにも向いていなかった。
3. Load Sharing System（LSF） --- 移植性は良く、かなり大規模システムに利用するのにも良かったが、大規模システムに利用しようとすると有償の為高価であった。

以上より、移植性に優れ、拡張性があり、かつ障害に強いシステムとしてSlurmの開発が始まりました。

# Slurmの構成

Slurmの構成

- 管理サーバー(slurmctld)

  ユーザーのジョブを受け取り，実行サーバー slurmd に実行を指示

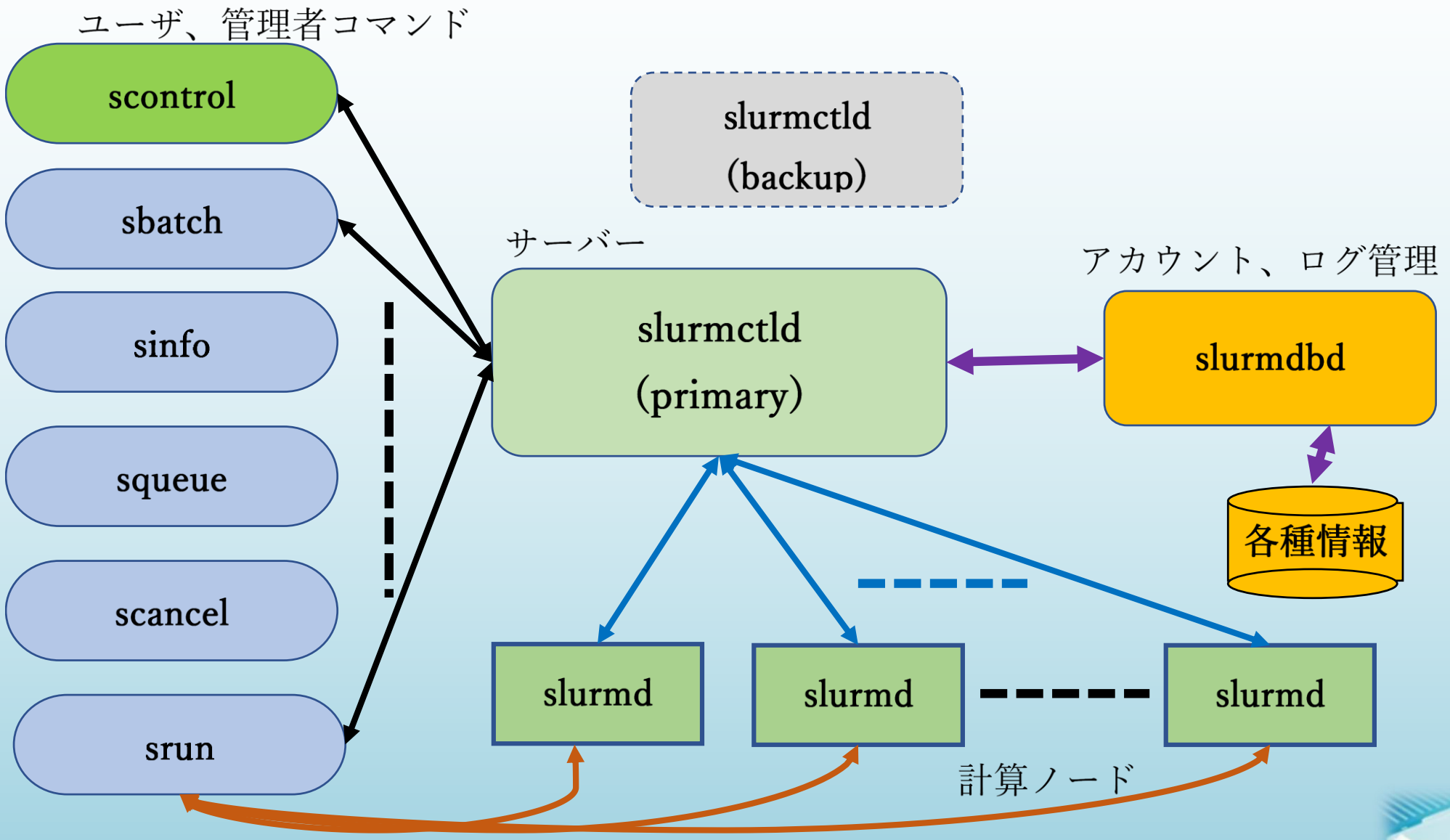
- 実行サーバー（slurmd)

  管理サーバーの指示を受けてユーザのジョブを実行

- データベースサーバー(slurmdbd)

  実行時間、ユーザ情報等の各種情報を保存

信頼性を上げるためにBackUpサーバーを置いて,Primaryサーバーに障害が発生した場合、BackUpサーバーが機能を引き継ぐようにすることが可能


ユーザ、管理者コマンド
scontrol
sbatch
sinfo
squeue
scancel
srun
slurmctld
(backup)
サーバー
slurmctld
(primary)
アカウント、ログ管理
slurmdbd
各種情報
slurmd
slurmd
slurmd
計算ノード

# PBSProfessonal（OpenPBS）

PBSは1991年6月にNASAで開発されました。今は、2003年にAltairがPBSの権利を購入して所有。NASAでの初期開発者もAltair社の社員になっています。

PBSには以下のVersionがあり用途によって使い分けることができます。

1. OpenPBS　———OpenSource版でAltairから提供／Linux版のみ
2. PBSProfessonal（PBS Pro）

　　Altair社が提供する有償のジョブ管理ソフトウェア

上記以外にTorqueがあり、OpenPBS同等のジョブ管理ソフトウェア、Adaptive　Computing社が管理をしている。

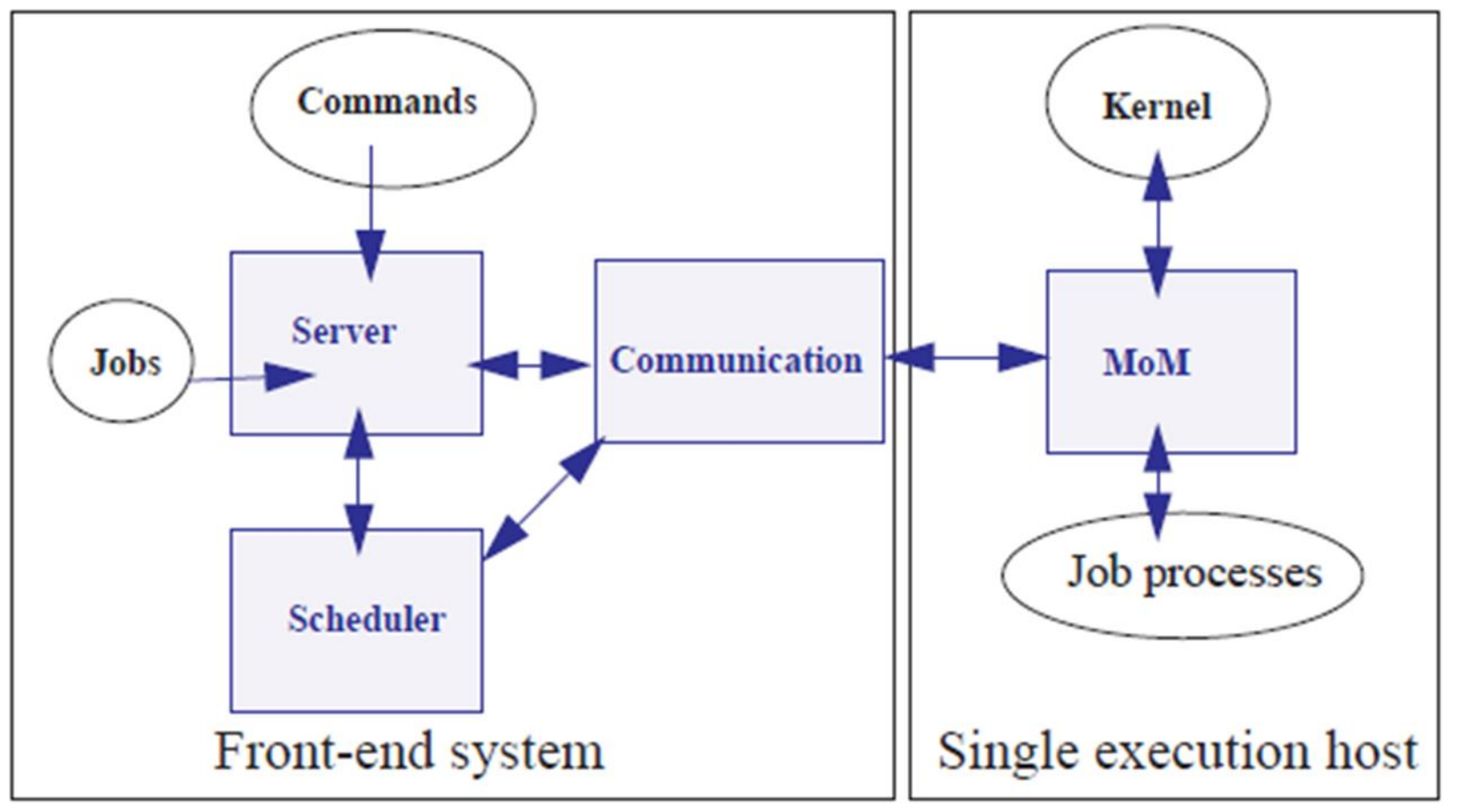
# PBSProfessonalの構成

V2020以降のサーバー構成はLinuxで構成、

計算ノードはLinuxかWindowsが可能

PBSProfessonalのクライアントは
LinuxかWindows

PBSProfessonalの
サーバーはLinux



# CPUとメモリ

# 性能向上の決定要因

# クラスタシステムの選択のポイント

CPU性能

（周波数×コア数×キャッシュ×メモリChannel×メモリタイプ）

×　クラスタ性能

（インターコネクト、ネットワーク×ファイルサーバー）

×　ライセンス費用

（並列数により価格上昇：ハード価格に比べて圧倒的に高価）

# CPUの高速化

## ➢ CPU

クラスタでは高速な計算を行うことが目的ですので、できる限り最新のCPUを採用することをお勧めします。　検討で悩むのが高速で少ないコア数か、高速でなくても大量のコアを搭載したCPUを選択するかです。

簡単に書くと以下の様になります。

- ✓CPUはなるべく高速のものを選択する。高速だとコア数が少なくなるので総合性能をSPECrate等で判断する。
- ✓CPUのアーキテクチャをそろえる。（x86/intel,AMD、arm/NVIDIA、独自）
- ✓CPU価格はコア数、周波数により価格が決まるので、価格とのバランスを考える。
- ✓商用ソフトウェアを使う場合は、並列数により価格が決まるので、商用ソフトウェアの価格がかなり重要になる。

# 最新CPUの特徴

96コアのCPUを32コアしか使わなければ1コア当たりの性能は向上

右Clockは早くないが、メモリチャネル、メモリ速度(MHz)、3次キャッシュが早く、大規模になっている。　これにより高速化に貢献している。

| Benchmark | Base Resu | 性能／コア数 | # Cores | # Chips | Processor | Processor MHz | Memory Channel | memory Type | 3次キャッシュ | Test Date | Published |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CFP2017rate | 2650.0 | 10.4 | 256 | 2 | Intel Xeon 6980P | 2000 | 12 | DDR5-6400 | 504 MB | Dec-24 | Jan-25 |
| CFP2017rate | 2410.0 | 6.3 | 384 | 2 | AMD EPYC 9965 | 2250 | 12 | DDR5-6000 | 384 MB | Dec-24 | Jan-25 |
| CFP2017rate | 1250.0 | 10.4 | 120 | 1 | Intel Xeon 6979P | 2100 | 12 | DDR5-6400 | 504 MB | Oct-24 | Oct-24 |
| CFP2017rate | 1100.0 | 11.5 | 96 | 2 | AMD EPYC 9454 | 2750 | 12 | DDR5-4800 | 256 MB | Jul-23 | Aug-23 |
| CFP2017rate | 666.0 | 13.9 | 48 | 2 | Intel Xeon Gold 6442Y | 2600 | 8 | DDR5-4800 | 60 MB | Jun-23 | Jul-23 |
| CFP2017rate | 531.0 | 8.3 | 64 | 2 | AMD EPYC 7543 | 2800 | 8 | DDR4-3200 | 256 MB | Apr-21 | May-21 |
| CFP2017rate | 291.0 | 6.1 | 48 | 2 | Intel Xeon Gold 6248R | 3000 | 6 | DDR4-2933 | 35.75 MB | Jun-20 | Aug-20 |
| CFP2017rate | 290.0 | 9.1 | 32 | 2 | AMD EPYC 7302 | 3000 | 8 | DDR4-3200 | 128MB | Nov-19 | Jan-20 |
| CFP2017rate | 228.0 | 7.1 | 32 | 2 | Intel Xeon Gold 6226R | 2900 | 6 | DDR4-2933 | 22 MB | Jun-20 | Aug-20 |
| CFP2006rate | 156.0 | 5.6 | 28 | 2 | Intel Xeon Gold 5120 | 2200 | 6 | DDR4-2400 | 19.25 MB | Sep-17 | Oct-17 |
| CFP2006rate | 139.0 | 8.7 | 16 | 2 | Intel Xeon Gold 6144 | 3500 | 6 | DDR4-2666 | 24.75 MB | Sep-17 | Oct-17 |
| CFP2006rate | 130.0 | 4.6 | 28 | 2 | Intel Xeon E5-2690 v4 | 2600 | 4 | DDR4-1600 DDR4-2400 | 35 MB | Apr-16 | May-16 |
| CFP2006rate | 113.0 | 4.7 | 24 | 2 | Intel Xeon E5-2690 v3 | 2600 | 4 | DDR4-1600 DDR4-2133 | 30 MB | Nov-15 | Dec-15 |
| CFP2006rate | 90.0 | 4.5 | 20 | 2 | Intel Xeon E5-2690 v2 | 3000 | 4 | DDR4-800 DDR4-1866 | 25 MB | Dec-13 | Dec-13 |
| CFP2006rate | 69.0 | 4.3 | 16 | 2 | Intel Xeon E5-2690 | 2900 | 4 | DDR4-800 DDR4-1600 | 20 MB | Feb-13 | Feb-13 |
| CFP2006rate | 52.0 | 4.3 | 12 | 2 | Intel Xeon E5-2640 | 2500 | 4 | DDR4-800 DDR4-1333 | 15 MB | May-12 | Jun-12 |
| CFP2017rate | 28.4 | 7.1 | 4 | 1 | Intel Xeon W-2104 | 3200 | 4 | DDR4-1600 DDR4-2400 | 8.25 MB | Oct-18 | Oct-18 |

# コンピュータの高速化

## CPUの微細化による高速化

コンピュータのCPUは微細化により性能向上を実現

- 素子と素子の距離が短くなり、それだけ信号の伝達時間が短くなる
- 素子が小さくなることにより集積度が上がり、より多くの処理が可能になる

性能向上はSPEC2006相当で2005年の17.7から2024年以は709.5で約40倍
1990年代の半ばからインテルやAMDのCPUが高速CPUとして採用される

| 年代 | 素子 | 大きさ | 比較 |
|---|---|---|---|
| 1906年 | 真空管 | 10cm | 10e-1m |
| 1950年 | 半導体 | 1cm | 10e-2m |
| 1960年 | IC | 1mm | 10e-3m |
| 1970年 | LSI | 10μm | 10e-5m |
| 2000年 | VLSI | 100nm | 10e-7m |
| 2014年 | VLSI | 14nm | 10e-8m |

2005年頃 ――― 2コア／１ＣＰＵ
2010年頃 ――― 4コア／１ＣＰＵ
2012年頃 ――― 8コア／１ＣＰＵ
2015年頃 ――― 24コア／１ＣＰＵ
2017年頃 ――― 64コア／１ＣＰＵ
2020年頃 ――― 96コア／１ＣＰＵ
2024年頃 ――― 192コア／１ＣＰＵ

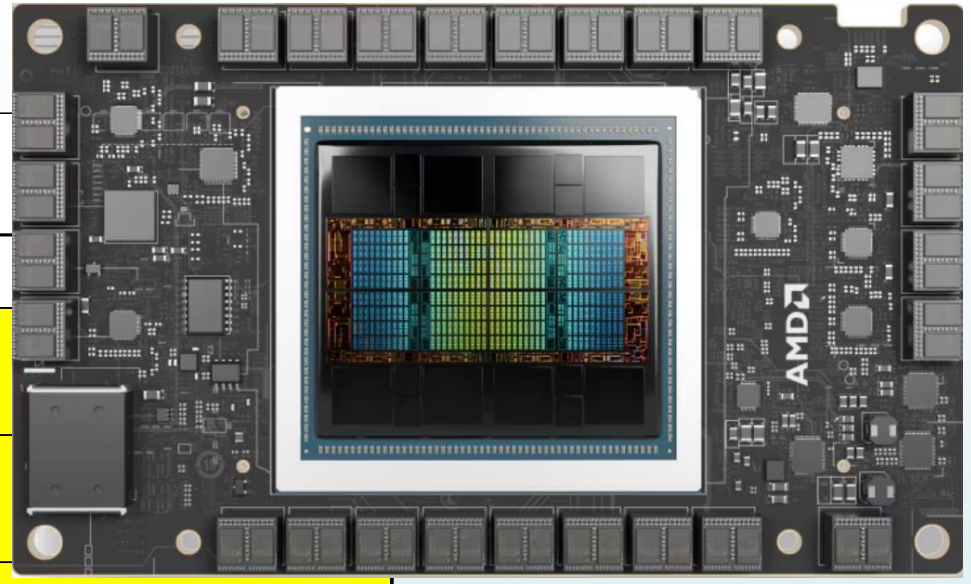
# GPUの変遷／新旧での相違

コア数は約8倍、性能は約30倍
数値計算指向からAI処理指向へ

| GPUアーキテクチャ | Blackwell<br>RTX PRO 6000 | Tesla K40 |
|---|---|---|
| 出荷 | 2025年 | 2008年頃 |
| CUDAコア | 24,064 | 2,880 |
| Tensorコア | 752 | |
| RTコア | 188 | |
| AI パフォーマンス | 4 PFLOPS | |
| 単精度性能(TFLOPS) | 125 | 4.29 |
| RTコア性能(TFLOPS) | 380 | |
| GPUメモリ | 96GB GDDR7 ECC | 12GB GDDR5 |
| メモリ インターフェイス | 512-bit | 384-bit |
| メモリ帯域幅 | 1792 GB/s | 288GB/s |
| システムインターフェイス | PCIe 5.0 x16 | PCI Express 3.0x16 |
| 最大消費電力 | 600W | 235W |

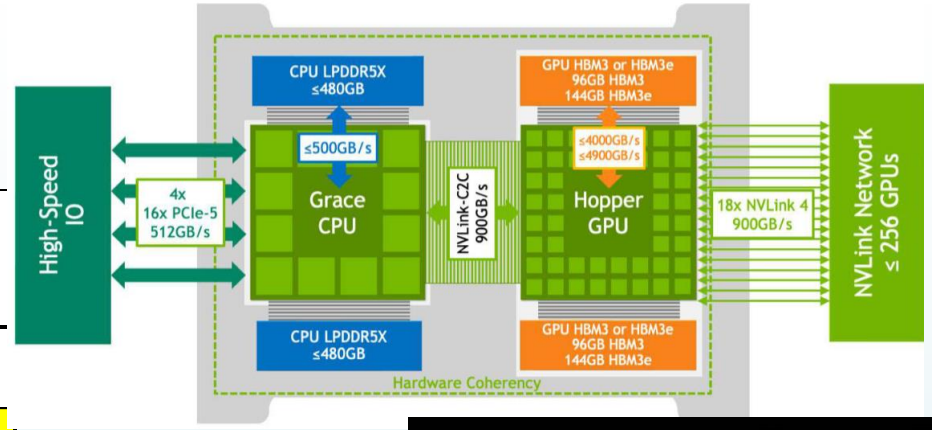
# GPUの変遷

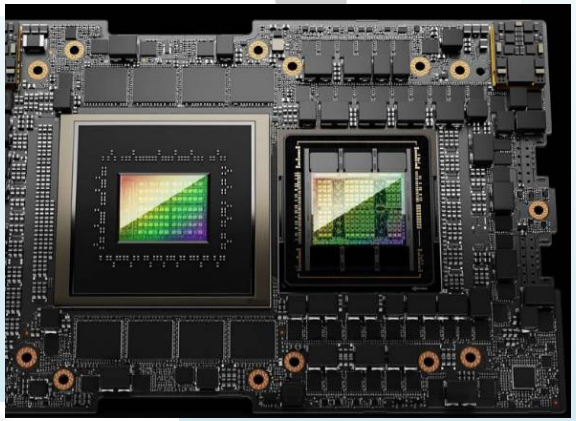
| GPUアーキテクチャ | Blackwell<br>RTX PRO 6000 | AMD<br>MI300 | |
|---|---|---|---|
| 出荷 | 2025年 | 2023年 | |
| CUDAコア<br>/ストリーミング プロセッサ | 24,064 | 14,592 | |
| Tensorコア<br>/マトリックス コア | 752 | 912 | |
| RTコア/Ray Tracing Cores | 188 | | |
| AI パフォーマンス | 4 PFLOPS | | |
| 単精度性能(TFLOPS) | 125 | 123 | 4.29 |
| RTコア性能(TFLOPS)<br>/マトリックス | 380 | 490 | |
| GPUメモリ | 96GB GDDR7 ECC | 128GB | 12GB GDDR5 |
| メモリ インターフェイス | 512-bit | | 384-bit |
| メモリ帯域幅 | 1792 GB/s | 5300 GB/s | 288GB/s |
| システムインターフェイス | PCIe 5.0 x16 | PCIe 5.0 x16 | PCI Express 3.0x16 |
| 最大消費電力 | 600W | 760W | 235W |

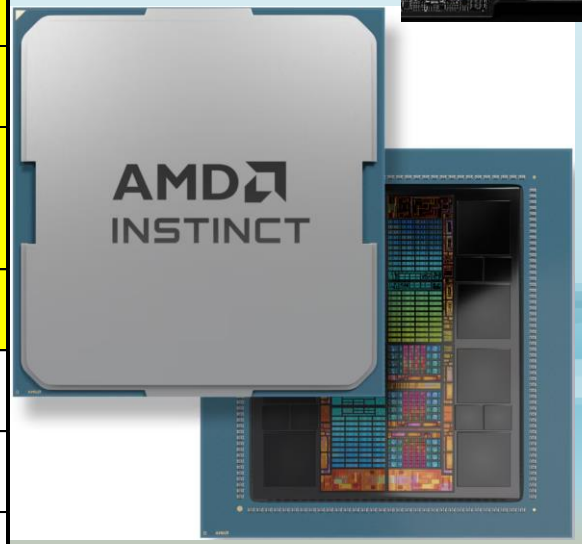


# CPU+GPUの変遷

| GPUアーキテクチャ | GH200<br>Grace Hopper | AMD<br>MI300A |
|---|---|---|
| 出荷 | 2024年 | 2023年 |
| CPU コア | 72 | 24 |
| CPU 周波数 | 3.1GHz | 2.1GHz(3.7GHz) |
| memory(Max) | 480GB | 128GB |
| 性能(FP64) TF(TeraFlops) | 34 | 61 |
| 性能(FP64 TensorCore)TF | 67 | 123 |
| 性能(FP32 TensorCore)TF | 989 | 981 |
| 性能(FP16 TensorCore)TF | 1979 | 1961 |
| メモリ帯域幅 | 384 GB/s | 5300 GB/s |
| システムインターフェイス | PCIe 5.0 x16 | PCIe 5.0 x16 |
| 最大消費電力 | 450Wk～1000W | 550Wk～760W |





# メモリの歴史と概要

## 記憶装置の階層構造

記憶装置は右の図に示すように階層構造になっています。これにより必要なデータを高速に利用できるようになります。

メモリのサイズは大規模化

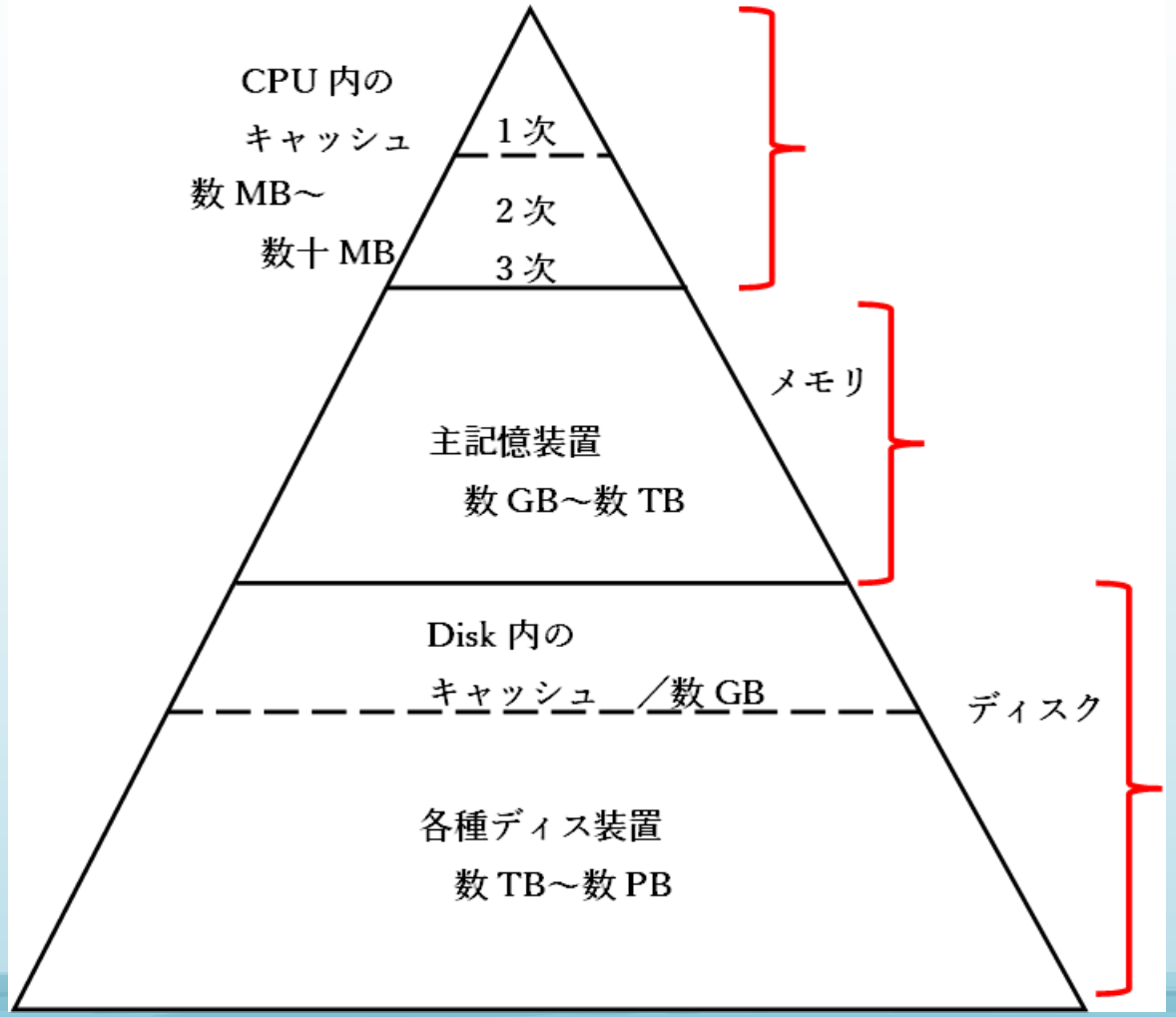
岩田の経験から

1990年代のメモリ容量は　100MB～10GB

2000年代のメモリ容量は　4GB～256GB

2010年頃のメモリ容量は　16GB～1TB

今は　　32GB　～　数TB

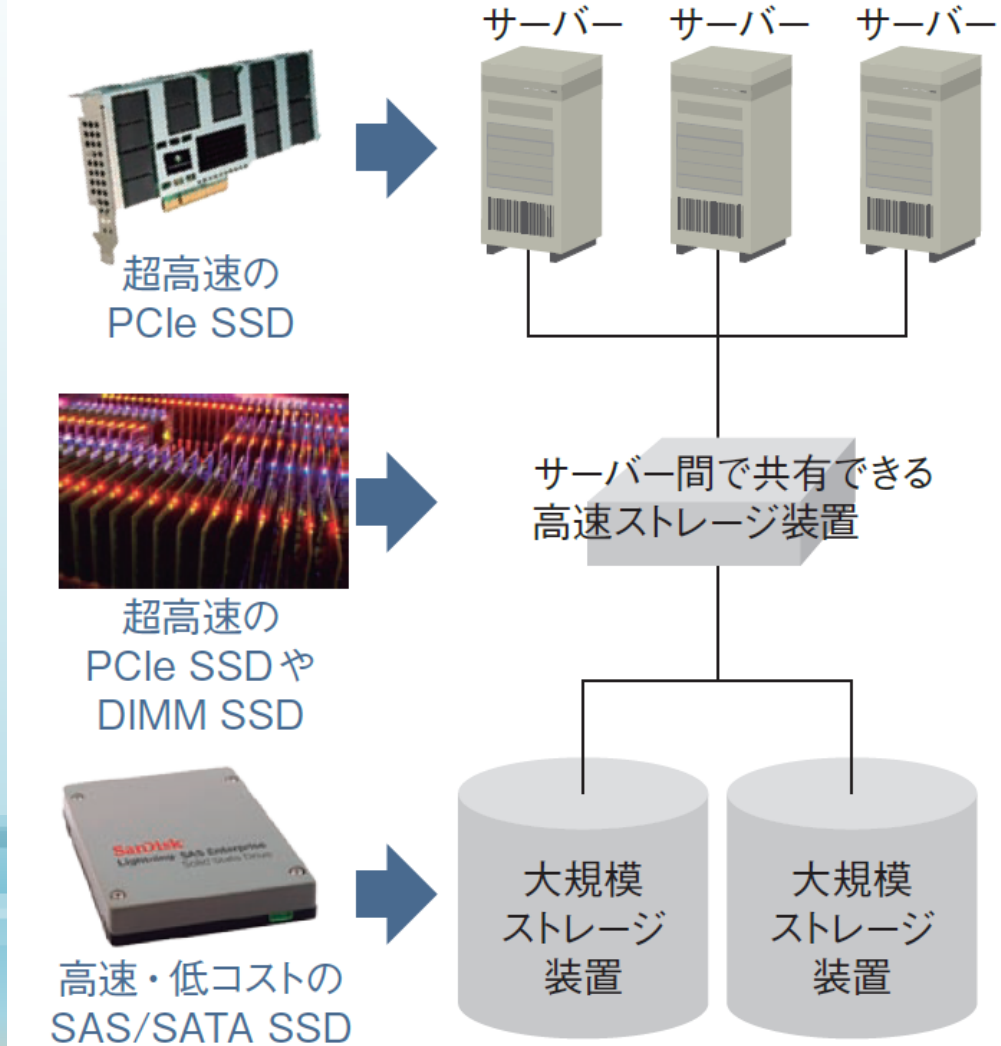



SRAMはキャッシュに使われ、DRAMは主記憶に使われる。SRAMはDRAMの10倍以上高速。

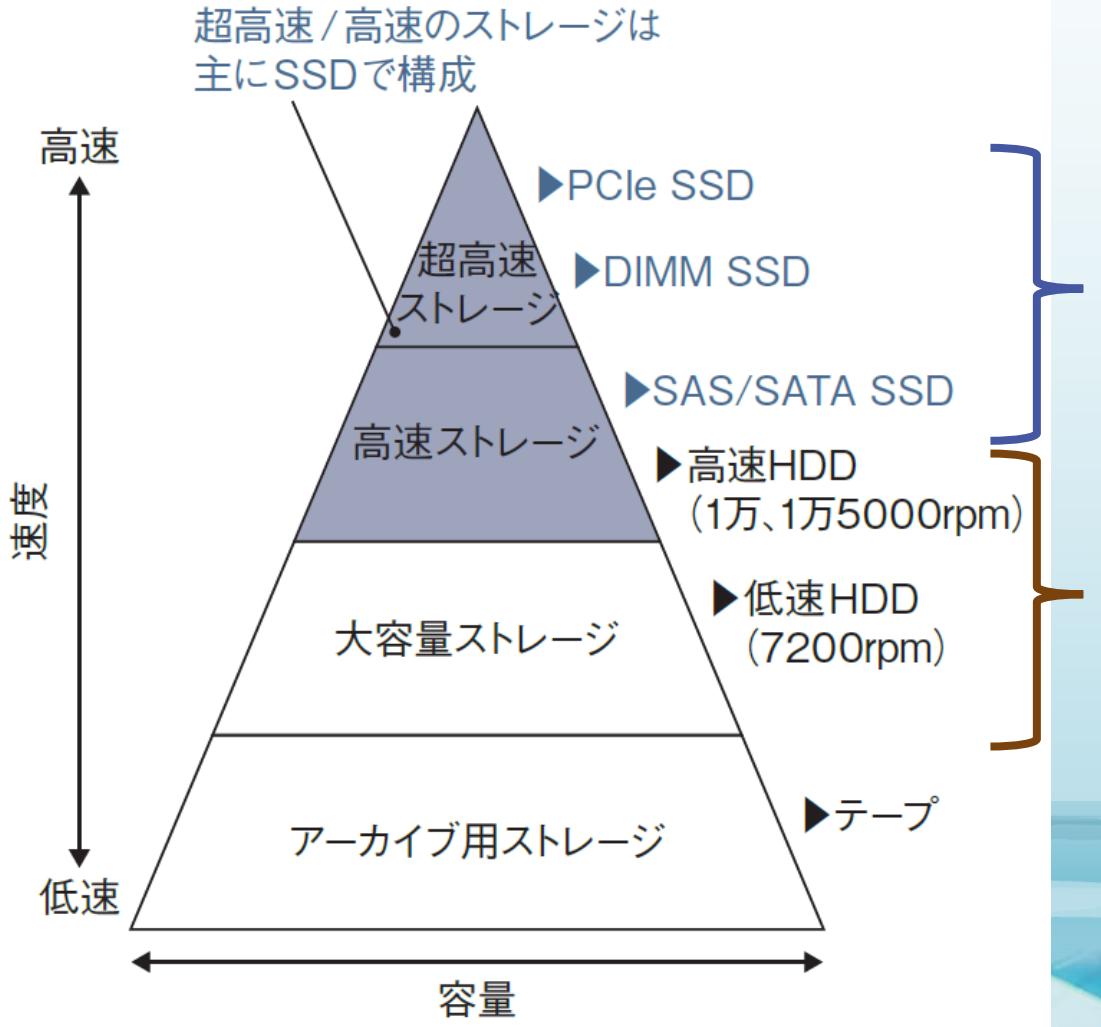
ディスクの書込みはDRAMの約10nsecに対して約10μsecと1000倍ほど遅い。

# 記憶装置（ディスク）の特性

(a)SSDは主に3種類




(b)SSDはストレージ階層の上位で利用

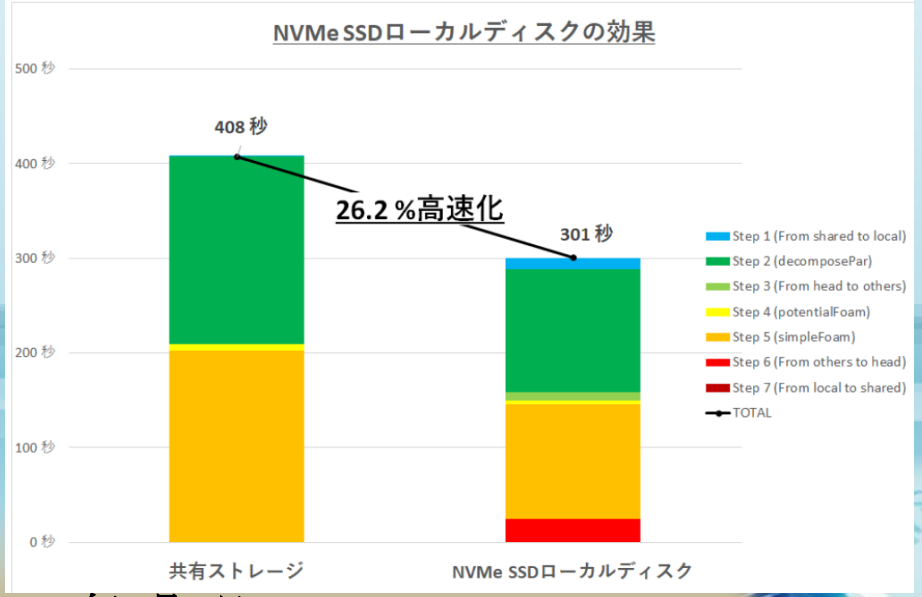


# 記憶装置（ディスク）の特性

| InterFace | 最大<br>読込み速度 | 最大<br>書込み速度 | 参考価格 |
|---|---|---|---|
| PCI Express5.0 | 9.5GB/s | 8.5GB/s | 4TB/10万円 |
| PCI Express4.0 | 7.0GB/s | 5.7GB/s | 8TB/17万円 |
| PCI Express3.0 | 3.48GB/s | 2.0GB/s | 1TB/7,000円 |
| Setrial ATA 6Gbps | 550MB/s | 510MB/s | 2TB/2万円 |
| SATA HDD | 約200MB/s | 約200MB/s | 8TB/3.5万円 |



共有ストレージを使用した場合とNVMe SSDローカルディスクを使用した場合の結果を以下に示します。
なお各値は、5回の計測値の平均値です。




## OracleでのOpenFOAMでのベンチマーク

ストレージによる性能の違い

# システム間を接続する高速な通信

# ➢ システム間接続

## ✓インターコネクト（ノード間の高速通信）

複数のノードを使って並列計算を行う場合は、高速で大量の通信ができることが必要です。しかし一番重要なのはノード間の通信の遅延時間が一番重要です。この遅延時間の短いものとして、2020年現在ではInfiniBandが最もよく使われています。

しかしEthernetに較べてPCIカードやスイッチが高価なので導入には十分検討する必要があります。

Infiniband以外には1990年代後半にはMyrinetが多く使われました。しかし、Infinibandの登場により、今は使われなくなっております。　この他にはintelが開発したOmni-Pathがありますが、今は別会社に移管されたようです。

上記以外には、各社が専用でインターコネクトを開発して、製品に組込んでいる製品もあります。例としてはクレイ社(今はHPE社に買収されている)の大規模クラスタでは専用のインターコネクトを使っております。　また富岳を作った富士通も　Tofuインターコネクトを開発して大規模なクラスターシステムを「6次元メッシュ／トーラス結合」でCPU間の通信を最短経路で遅延を最小にするような接続形態を開発しています。

## ✓ Myrinet

1994年に創立したMyricom社が作ったHPC向けのインターコネクト製品。2000年前後は10-Gigabitの性能を提供しクラスタ構築する上では重要な製品であったが、InfiniBandが出てきて性能、価格で優位性がなくなり、シェアを落とし2013年にCSP社に買収されて、今ではCSP社の中でも製品情報は見ることができなくなっています。　製品としては2000年前後の時代に高速な通信を提供できていたので一般的であった。しかし岩田がお客様に提供した50ノードのクラスタにMyrinetを使ったシステムでは障害が多く発生し改善に多大な時間がかかった。

## ✓ InfiniBand

複数のノードを使って並列計算を行う場合の主要なインターコネクト製品。

1998年後半にインテルを中心にスイッチ型ファブリックインターコネクトテクノロジをベースとした新しいI/Oアーキテクチャが現れ、2000年1月にInfiniBandの名前が生まれた。

それまではインテル、HP,IBM等の企業が個別に規格を策定していたが各規格をまとめてInfiniBandの規格ができております。

InfiniBandの特徴は

高速、低レイテンシ、低価格

が挙げられます。　低価格と言っても標準のEthernetに較べるとかなり高くなります。

性能を列挙すると以下の様になり、現在、市販されている製品は400GbpsのHDRが最高性能の製品です。

InfiniBand　性能

| | SDR | DDR | QDR | FDR | EDR | HDR | NDR | XDR |
|---|---|---|---|---|---|---|---|---|
| 速度(片方向) | 10 Gbps | 20 Gbps | 40 Gbps | 56 Gbps | 100 Gbps | 200 Gbps | 400 Gbps | 100 Gbps |
| 遅延 | 5μs | 2.5μs | 1.3μs | 0.7μs | 0.5μs | 0.5μs未満 | 90ns | |
| 規格の発行年 | 2000 | 2004 | 2004 | 2012 | 2012 | 2020 | 2021 | 2025予定 |

# クラスタの構成で重要なのはインターコネクトの遅延時間
# Ethernetに較べると、その性能の差が良くわかる

右の情報はSC16の
Comparison of High Performance Network Options:
EDR InfiniBand vs. 100Gb RDMA Capable Ethernet
より抜粋。

性能はメッセージのサイズにもよりますが、10倍前後の性能を発揮しております。

| 数値 | 単位名称 | |
|---|---|---|
| 1000000000 | ns | ナノ秒 |
| 1000000 | μs | マイクロ秒 |
| 1000 | ms | ミリ秒 |
| 1 | s | 秒 |

| Average Latency ／ 100Gbps | | |
|---|---|---|
| Message Size | Native IB（μs） | Native Ethernet（μs） |
| 8 | 0.892 | 12.662 |
| 16 | 0.938 | 12.692 |
| 32 | 0.95 | 12.718 |
| 64 | 0.958 | 12.712 |
| 128 | 1.344 | 12.78 |
| 256 | 1.402 | 12.884 |
| 512 | 1.518 | 13.104 |
| 1024 | 1.752 | 13.55 |

# データの転送能力において100Gbps仕様で50倍程度の性能を発揮

| Average Bandwidth ／ 100Gbps | | |
|---|---|---|
| Message Size | Native IB（MB/s） | Native Ethernet（MB/s） |
| 131,072 | 11,848 | 232 |
| 262,144 | 12,086 | 233 |
| 524,288 | 12,235 | 234 |
| 1,048,576 | 12,318 | 235 |
| 2,097,152 | 12,351 | 235 |
| 4,194,304 | 12,371 | 235 |

## ✓ InfiniBandを構成する部品

HCA（Host Channel Adapter）はシステムのPCIポートに装着するパーツ


InfiniBandケーブル。　HDR対応の銅ケーブルで　1本5万円前後、概要は以下の様にQSFP仕様である。

InfiniBand　Switch。　システム間を接続するもの、EDR、HDRだと36ポート以上になる