

TSUBAME2.0の ペタスケールデータ処理基盤

東京工業大学

佐藤 仁

情報爆発時代

- ▶ 人類の扱う情報量の爆発的増大
 - ▶ 2000年以降は指数的な増加
 - ▶ 例. CERNのLHC実験では **1年に15PB**のデータが生成
 - ▶ 他の科学技術分野(バイオインフォマティクス)なども同様
 - ▶ 解析器などのデバイスの性能向上も影響
- ▪ **常に大量の(ペタスケールの)データが生成される**
 - **生成された大量のデータに対する解析が必要**

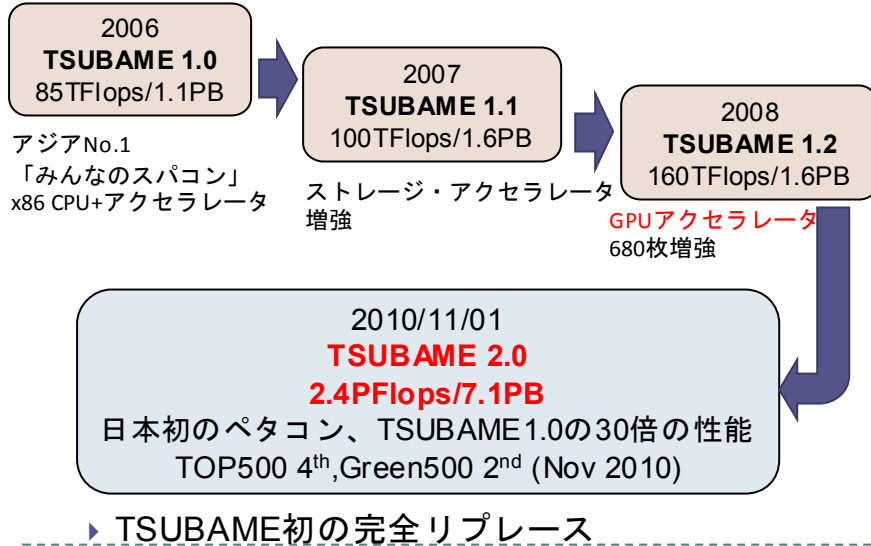


ペタスケールのデータを処理するための課題

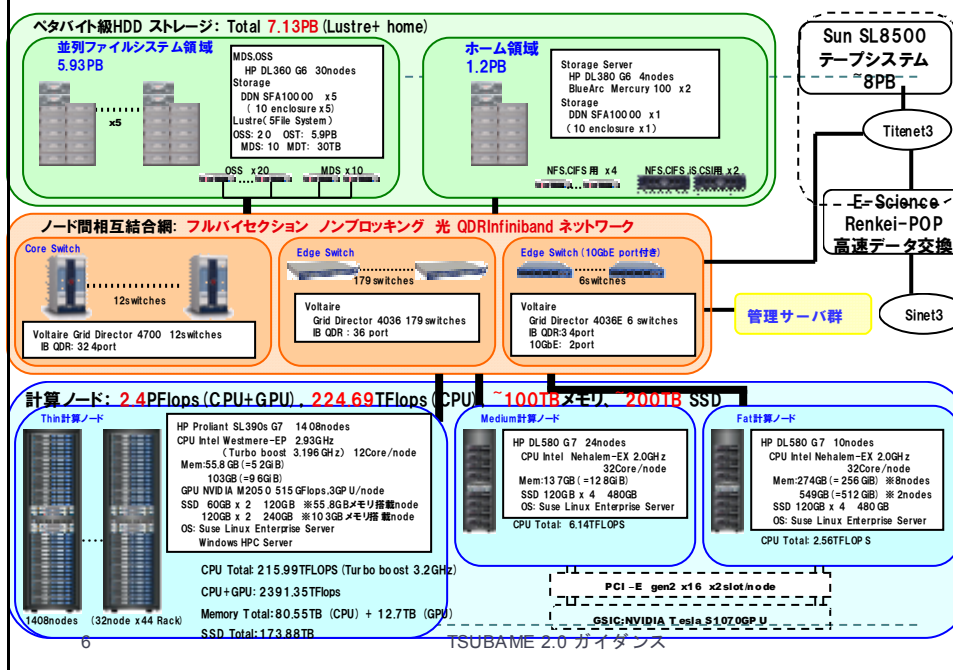
- ▶ スパコンのストレージの設計
 - ▶ 大規模データ処理基盤の整備
 - ▶ 大規模データ共有方法
-

スパコンのストレージ

東工大TSUBAME



TSUBAME2.0 システム概念図





TSUBAME2.0 Nov. 2010 w/ NEC/HP

- ▶ 2.4 PF Next gen multi-core x86 + next gen GPGPU
 - ▶ 1432 nodes, Intel Westmere/Nehalem EX
 - ▶ 4224 NVIDIA Tesla (Fermi) M2050 GPUs
 - ▶ ~100,000 total CPU and GPU "cores", High Bandwidth
- ▶ 0.7 Petabyte/s aggregate mem BW,
 - ▶ Effective 0.3-0.5 Bytes/Flop, restrained memory capacity (95TB)
- ▶ Optical Dual-Rail IB-QDR BW, full bisection BW (Fat Tree)
 - ▶ 200Tbits/s, Likely fastest in the world, still scalable
- ▶ Flash/node, ~200TB (1PB in future), 660GB/s I/O BW
 - ▶ >7 PB IB attached HDDs, 15PB Total HFS incl. LTO tape
- ▶ Low power & efficient cooling, comparable to TSUBAME 1.0 (~1MW); PUE = 1.28 (60% better c. f. TSUBAME1)
- ▶ Virtualization and Dynamic Provisioning of Windows HPC
- + Linux, job migration, etc. ---

Storage problems in Cloud-based SCs

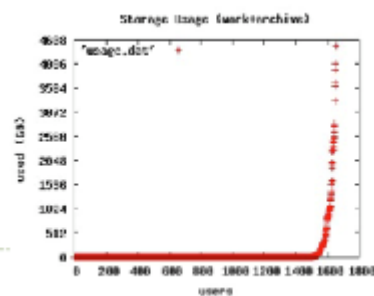
- ▶ Support for Various I/O workloads
 - ▶ Storage Usage
 - ▶ Usability
-

I/O workloads

- ▶ Various HPC apps. run on TSUBAME2.0
 - ▶ Concurrent Parallel R/W I/O
 - ▶ MPI (MPI-IO), MPI with CUDA, OpenMP, etc.
 - ▶ Fine-grain R/W I/O
 - ▶ Checkpoint, Temporal files
 - ▶ Gaussian, etc.
 - ▶ Read mostly I/O
 - ▶ Data-intensive apps, Parallel workflow, Parameter survey
 - ▶ Array job, Hadoop, etc.
 - ▶ Shared Storage
 - ▶ I/O concentration
-

Storage Usage

- ▶ Data life cycle management
 - ▶ Few users occupy most of storage volumes on TSUBAME1.0
 - ▶ Only 0.02 % of users use more than 1TB of storage volumes
- ▶ Storage resource characteristics
 - ▶ HDD : $\sim 150\text{MB/s}$, 0.16 \$/GB, 10 W/disk
 - ▶ SSD : 100 $\sim 1000\text{ MB/s}$, 4.0 \$/GB, 0.2 W/disk
 - ▶ Tape : $\sim 100\text{MB/s}$, 1.0 \$/GB, low power consumption



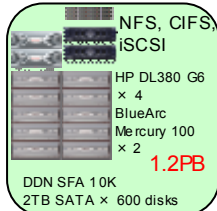
Usability

- ▶ Seamless data access to SCs
 - ▶ Federated storage between private PCs, clusters in lab and SCs
 - ▶ Storage service for campus like cloud storage services
- ▶ How to deal large data sets
 - ▶ Transfer big data between SCs
 - ▶ e.g.) Web data mining on TSUBAME1
 - ▶ NICT (Osaka) $\rightarrow$ TokyoTech (Tokyo) : Stage-in **2TB** of initial data
 - ▶ TokyoTech $\rightarrow$ NICT : Stage-out **60TB** of results
 - Transfer data via the Internet : 8days
 - Fedex?

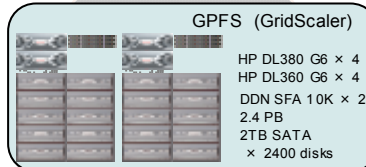
TSUBAME2.0 Storage Overview

TSUBAME2.0 Storage 11PB (7PB HDD, 4PB Tape)

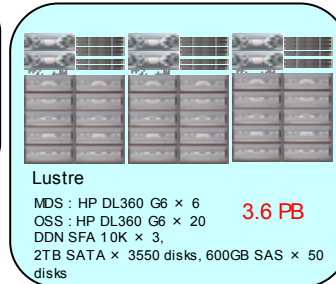
Home



Phome



Work



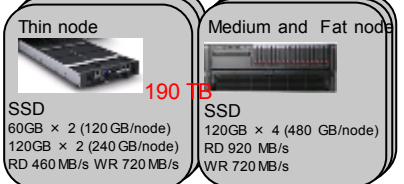
Grid Storage



HSM
with GPFS, Tivoli Storage Manager
2.4 PB HDD + ~4PB Tape

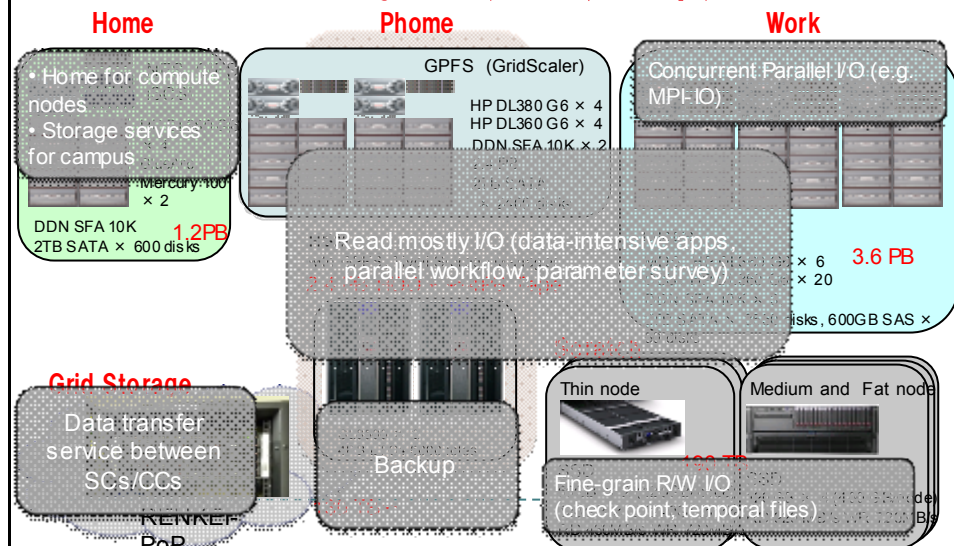


Scratch



TSUBAME2.0 Storage Overview

TSUBAME2.0 Storage 11PB (7PB HDD, 4PB Tape)



大規模データ処理基盤の整備

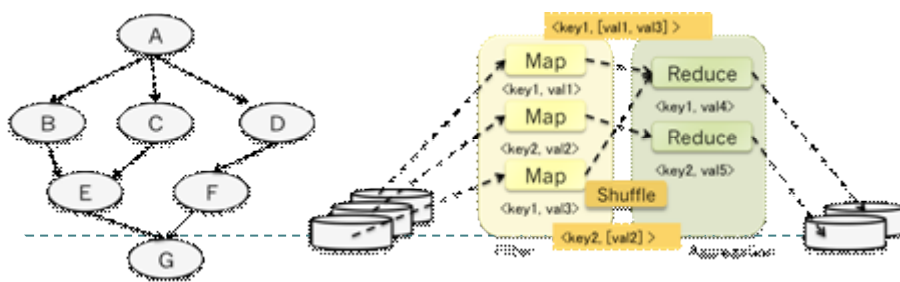
大規模データ処理環境

▶ ワークフロー

- ▶ ジョブのワークフローをDAGとして表現
- ▶ ex.) DAGMan, Pegasus, Dryad, GXP Make, etc.

▶ MapReduce

- ▶ ex.) Google MapReduce, Hadoop, etc.



GXP / GXP make

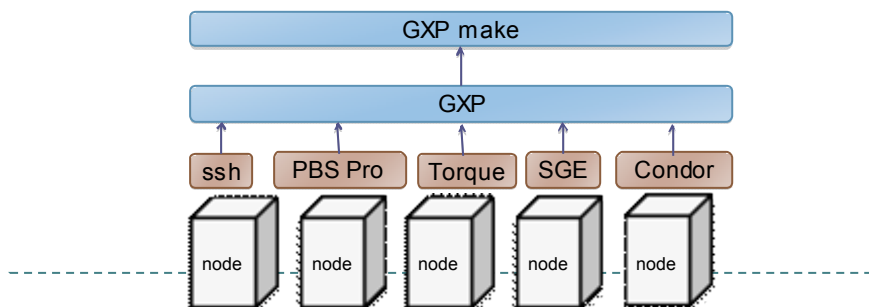
- ▶ <http://www.logos.ic.i.u-tokyo.ac.jp/gxp>

▶ GXP :

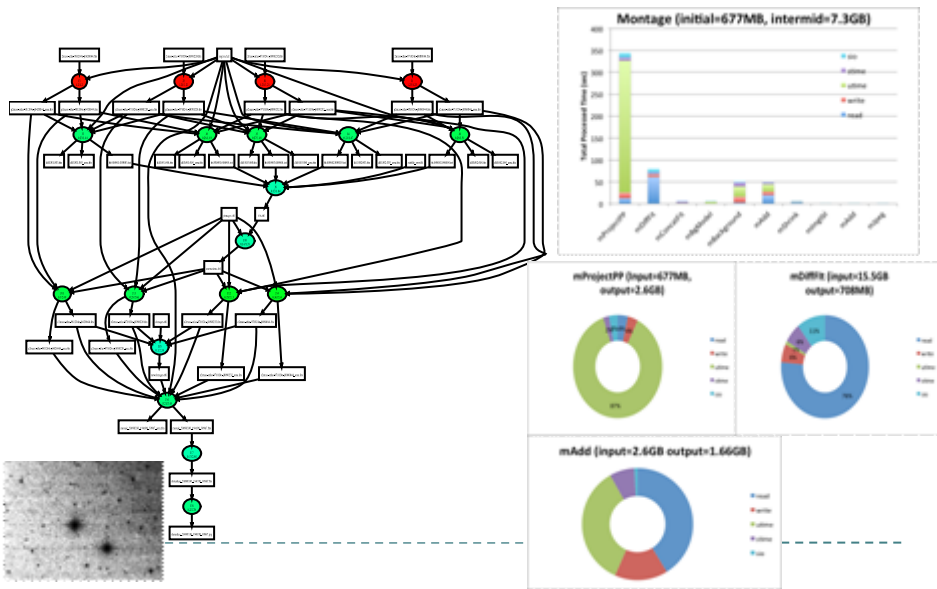
- ▶ 並列シェル (プロセス起動)

▶ GXP make :

- ▶ (並列)ワークフロー,makeでワークフローを記述

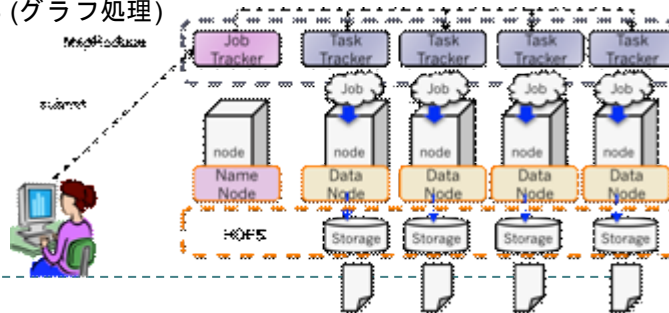


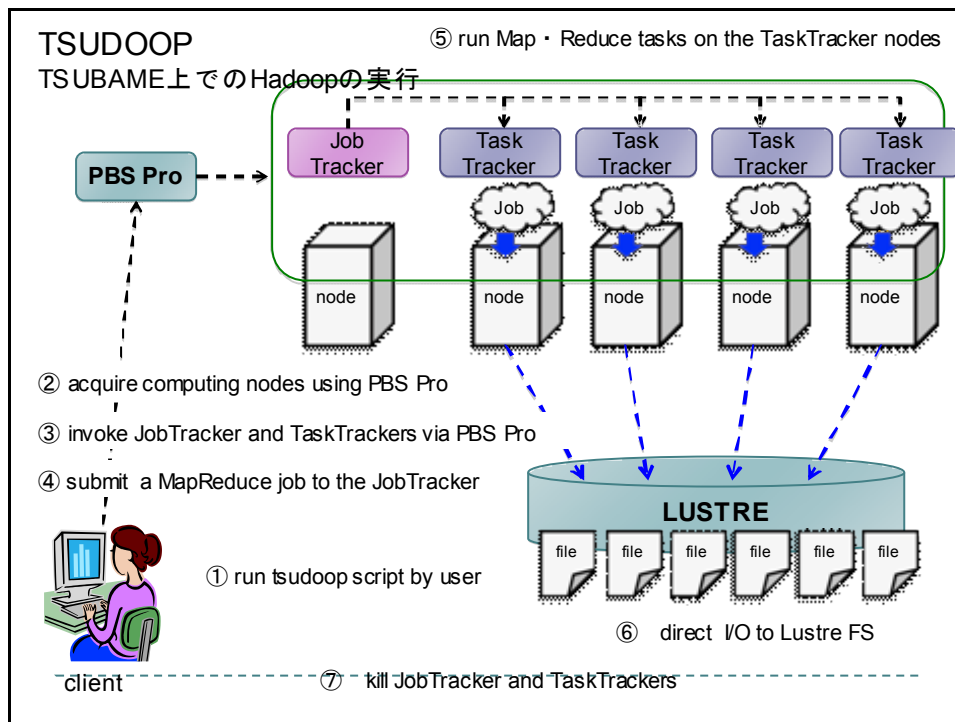
TSUBAME2.0上での GXP makeによるMontage (天文アプリ)の実行



MapReduce (Hadoop)

- ▶ Hadoop
 - ▶ ApacheプロジェクトによるOSSのMapReduce処理系
 - ▶ MapReduce + HDFS
 - ▶ エコシステム
 - ▶ Mahout (機械学習)
 - ▶ HAMA (BSP処理)
 - ▶ PEGASUS (グラフ処理)





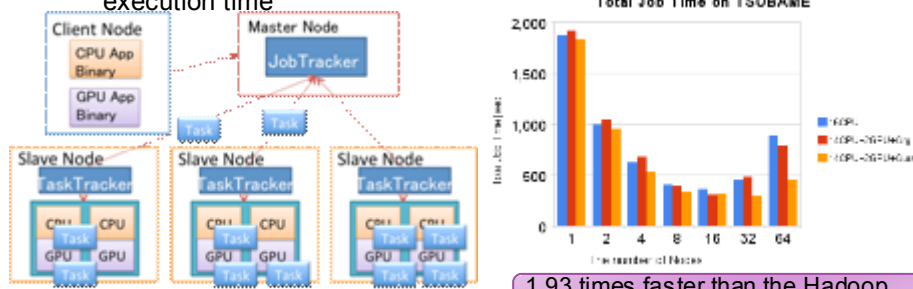
MapReduce for GPU-based Heterogeneous Clusters

Problems :

- Map Task Scheduling for efficient execution
 - Depends on **running task characteristics** and **underlying environments**

Hybrid Map Task Scheduling

- Automatically detects map task characteristics by monitoring
- Scheduling map tasks to minimize overall MapReduce job execution time



1.93 times faster than the Hadoop original scheduling

大規模データ共有

Grid Storage

RENKEI: REsources liNKage for E-science

PoP : Point of Presences

▶ RENKEI Cloud

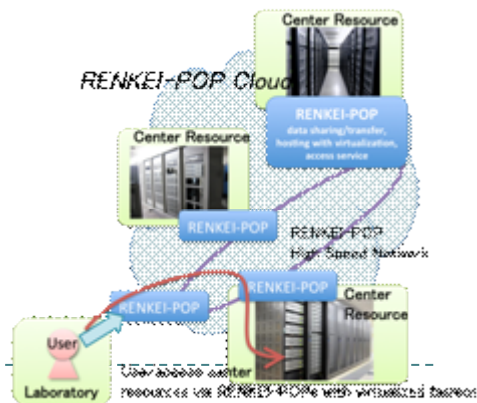
▶ High-performance data transfer service for e-Science

▶ Seamless data access between SCs/CCs

- w/ Grid security, Authentication

▶ Multi-protocol data access

- scp, gridftp, gsiscp
- 164TB of Gfarm FS



RENKEI-PoP

- ▶ 大容量ストレージを持ったサーバプライアンス
 - ▶ 個々の拠点にPoPを1つ設置
 - ▶ 拠点リソースにアクセスするためのゲートウェイ
 - ▶ 拠点間ポリシーと拠点内ポリシーの調停
 - 認証, 情報サービスなど
 - ▶ 拠点間の高速データ転送, 共有環境を提供
 - ストレージを転送時のキャッシュ的に使う
 - RENKEI-PoP間Disk-to-Disk理論転送性能を1GBpsで設計

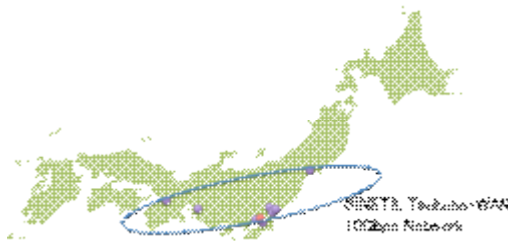


ファイルシステムIO性能 単位 (MB/s)

	Write	Re-Write	Read	Re-Read
16GB	750	1030	800	810

CPU	Core i7 975 Extreme (3.33 GHz)
Memory	12GB (DDR3 PC3-10600 , 2GB*6)
NIC	10GBASE-SR or LR
System Disk	500GB HDD
Storage	30TB (RAID 5, 2TB HDD x 16)
OS	CentOS

RENKEI-POP配備状況



RENKEI-POP設置拠点（設置順）

東京工業大学 (titech, titech2)

大阪大学 (osaka)

国立情報学研究所 - 千葉 (nii)

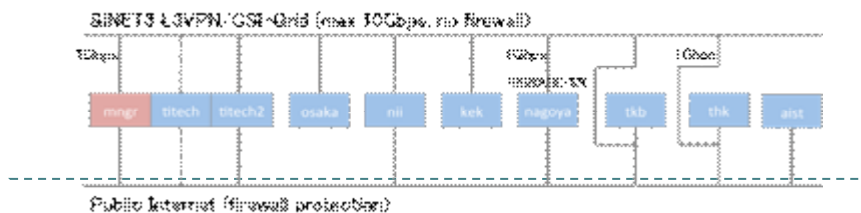
高エネルギー加速器研究機構 (kek)

名古屋大学 (nagoya)

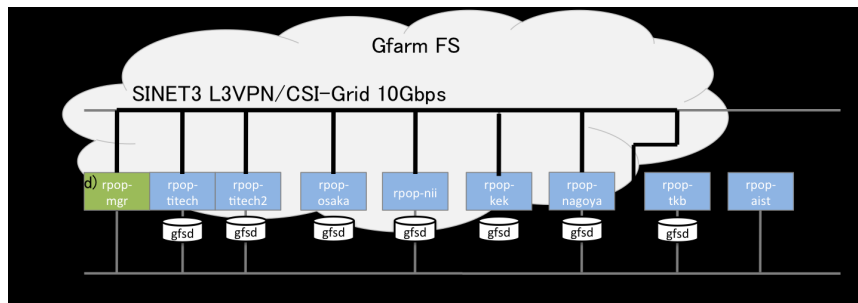
筑波大学 (tkb)

産業技術総合研究所 (aist)

東北大学 (thk)



Storage Federation in RENKEI Cloud

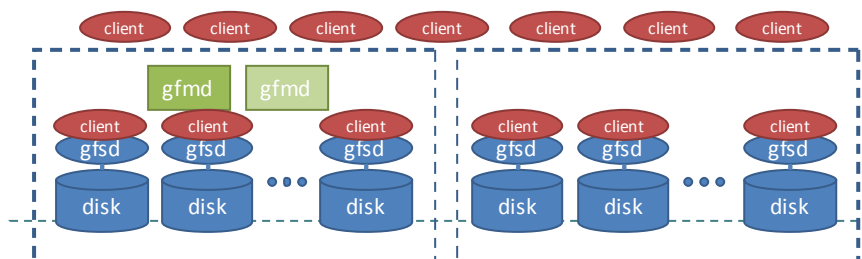


- ▶ A single system image by using **Gfarm FS**
- ▶ File movement & replication between RENKEI PoPs via SINET3
- ▶ GSI-based user authentication
 - ▶ Account integration with SCs (TokyoTech etc.)
 - ▶ RENKEI-PoP account

Gfarmファイルシステム



- ▶ オープンソース広域分散ファイルシステム
<http://sf.net/projects/gfarm/>
- ▶ **広域**で性能が**スケールアウト**するファイルシステム
 - ▶ ファイルサーバ, クライアント追加によるスケールアウト
 - ▶ ローカル (近いサーバへの) アクセス優先, (自動)ファイル複製
- ▶ **Better NFS**としても利用可



ファイル自動複製機能

- ▶ Gfarm-2.4.1以降でサポート
 - ▶ （祖先の）ディレクトリの拡張属性で複製数を指定
 - ▶ Close時に自動的にファイル複製を作成
- % **echo** -n **3** | **gfxattr** -s / **gfarm.ncopy**
-

更新型複製間一貫性制御

- ▶ Gfarm-2.4.0以降でサポート
 - ▶ ファイル更新時，古いファイル複製を自動的に更新
 - ▶ ファイルを更新しても複製数は変わらない
-

クォータによる利用制限

- ▶ Gfarm-2.3.1以降でサポート
 - ▶ See doc/quota.en
 - ▶ 管理者（gfarmadm）が設定可能
 - ▶ ユーザ，グループごと
 - ▶ 利用容量，ファイル数の制限
 - ▶ ファイルによる制限と複製も考慮した物理制限
 - ▶ ハードリミットと猶予期間のあるソフトリミット
 - ▶ ファイルオープン時にチェック
 - ▶ 注意：越えたら作成できないが，既にオープンしているファイルは容量制限を超えることが可能
-

XML拡張属性

- ▶ Gfarm-2.3.0以降でサポート
 - ▶ 通常の拡張属性に加え，XMLをvalueとする
% **gfxattr** -x -s -f value.xml filename xmlattr
 - ▶ Xpathによるdirectory treeのXML拡張属性の検索
% **gffindxmlattr** [-d depth] XPath path
-

障害対応

- ▶ メタデータサーバのリブート, 故障, フェールオーバー
 - ▶ 書込オープン中のファイル以外は, アプリケーションはメタデータサーバの復旧を待って正常続行
 - ▶ ファイルシステムノードのリブート, 故障
 - ▶ そのファイルシステムノードのファイルをオープン中のプロセスを除き, ファイル複製, 利用可能なファイルシステムノードがある限り正常続行
 - ▶ アプリケーションの障害
 - ▶ オープン中のファイルはその時点でクローズ
-

局所的ディスクのNo space対応

- ▶ Minimum_free_disk_space
 - ▶ デフォルトでは空きが128MB以下の場合, 書込対象とならない
 - ▶ Readonly
 - ▶ 空きが少なくなってきたらreadonly (=強制的に利用率100%) にすることによりリスクを減らせる
 - ▶ 見かけ上利用率100%にしているだけなので削除は可能
-

Samba VFS for Gfarm

- ▶ Gfarm2fsを利用しなくてもSambaからGfarmを利用するためのモジュール
 - ▶ 2011/12/30に実験的公開
-

Gfarm GridFTP DSI

- ▶ Globus GridFTPでGfarmを利用するためのストレージI/F
 - ▶ GridFTPはGSI認証，データ接続認証，EBLOCKモード導入による並列転送などFTPの拡張[GFD.20]
 - ▶ <http://sf.net/projects/gfarm/>
 - ▶ JLDG (Japan Lattice Data Grid)で利用
 - ▶ GSI認証であるため，サーバにアカウントを作成する必要なし
 - ▶ Anonymousおよびclear textでの認証も可能
-

Debian packaging

▶ Squeezeのパッケージへの取り込み

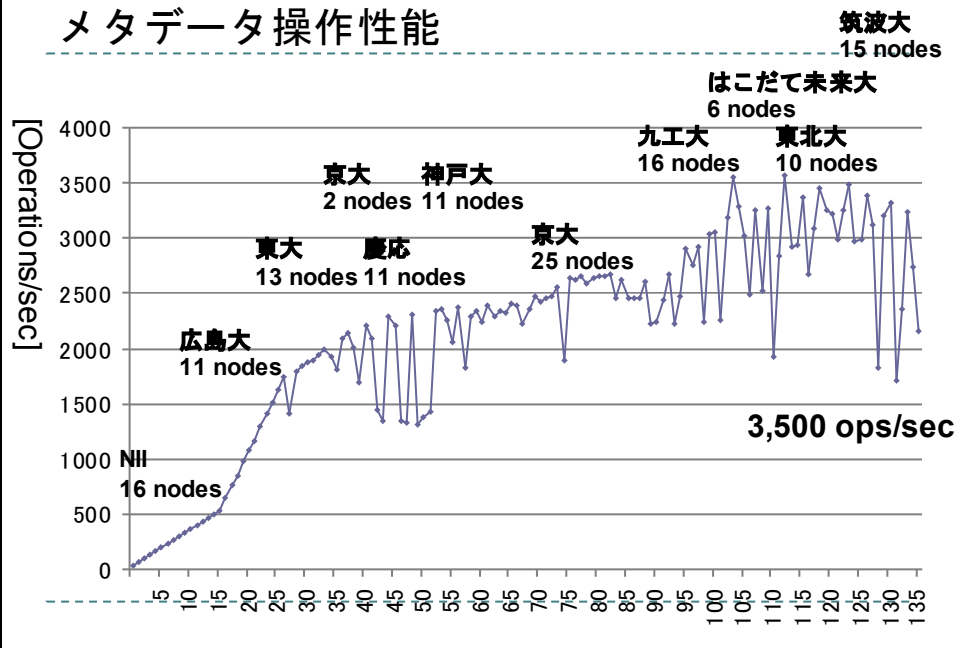


Gfarmの運用事例

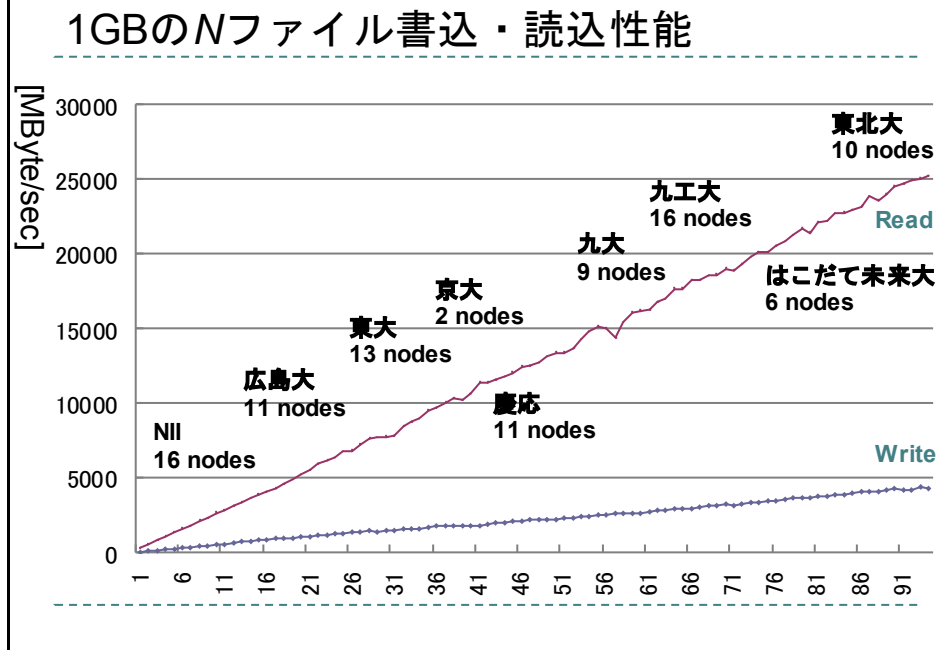
- ▶ InTrigger（情報爆発科研プラットフォーム）
 - ▶ はこだて未来大，東北大，筑波大，東京大，NII，東工大，早稲田大，慶応大，京都大，神戸大，広島大，九州大，九工大
- ▶ Gfarmファイルシステム
 - ▶ メタデータサーバ（MDS）：筑波大ノード
 - ▶ 14拠点，239ノード，146 TBytes
 - ▶ RTT ~50 msec
- ▶ 一年以上安定稼働中！！

```
% gfd -a
1K-blocks    Used    Avail Capacity    Files
119986913784 73851629568 46135284216 62%    802306
```

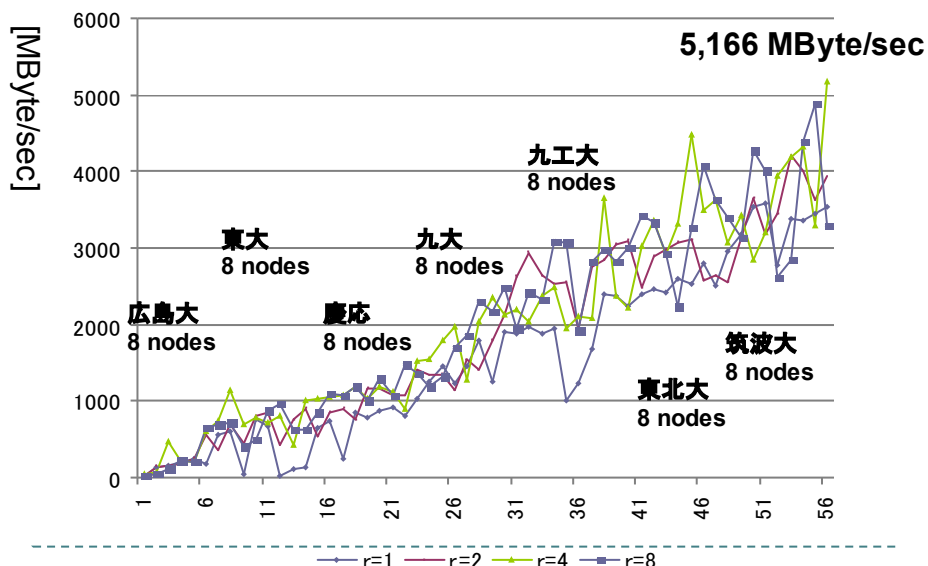
メタデータ操作性能



1GBのNファイル書込・読込性能



1GBの共有ファイル読込性能



RENKEI-PoP上でのGfarmデータ転送性能

- ▶ MontageデータセットをGfarmで転送した時のスループットを計測
 - ▶ 各RENKEI-PoPからtitechへgfreepで複製作成
- ▶ 単一ファイルにアーカイブ（非圧縮）して転送した結果を紹介

gfarm構成

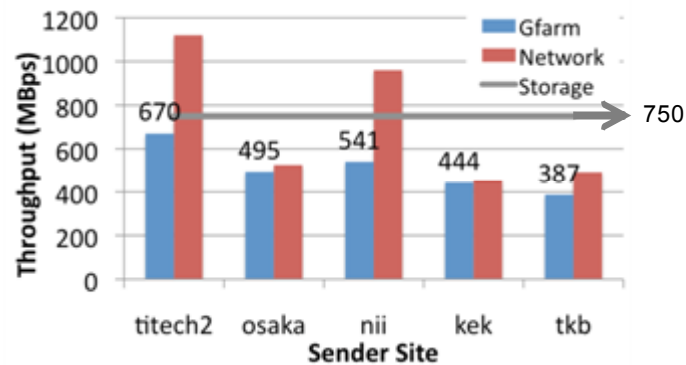
バージョン	2.3.2
gfarm mds	管理用サーバ@東工大
gfarm fs	CSI-Grid接続RENKEI-PoP
認証	ホスト間: 共有秘密鍵 ホスト-ユーザ: gs認証
通信パラメータ	gfarm デフォルト
暗号化通信	なし

データ特徴

総容量	14GB
総数	6756個
最小サイズ	292B
最大サイズ	27MB
平均サイズ	2MB
平均分散	400KB



Gfarmデータ転送性能 – 結果



- ▶ titech2, NIIではStorage Write性能に律速

RENKEI-PoP 将来計画

- ▶ 新学術領域研究「ゲノム支援」のデータ転送基盤として
 - ▶ <http://www.genome-sci.jp/>
 - ▶ 新型DNAシーケンサにより大量の小容量(数MB～数百MB)データをバースト的に生成し、データベース化. 遺伝研, 東大(柏), 宮崎大, 東工大で共有
- ▶ 乱流データ解析プロジェクト
 - ▶ 名大ストレージにある乱流データを筑波大に転送して解析

RENKEI-PoPでの研究開発, ノウハウをHPCIストレージへ応用

まとめ

- ▶ スパコンのストレージの設計
 - ▶ TSUBAME2.0のストレージ構成の紹介
- ▶ 大規模データ処理基盤の整備
 - ▶ TSUBAME2.0上でのデータ処理環境の紹介
- ▶ 大規模データ共有方法
 - ▶ RENKEI-PoP → HPCIのストレージへ
