



Grid Datafarmにおけるスケジュー リング・複製手法の性能評価

竹房あつ子 (お茶の水女子大学)

建部修見 (産業技術総合研究所)

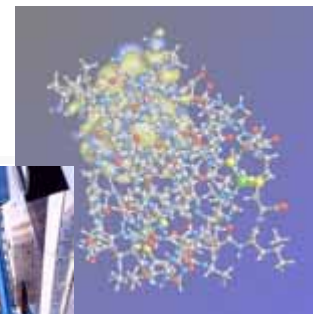
松岡聡 (東京工業大学/国立情報学研究所)

森田洋平 (高エネルギー加速器研究機構)

ペタスケール データコンピューティング

- 大規模データ計算科学,
データマイニング
 - 高エネルギー物理学, 粒子物理学
 - CERN Large Hadron Collider(LHC)
実験 (2007年)
 - 天文台, 地球惑星, 生命情報工学
- 大規模ビジネスデータベース
 - e-Japan, 電子政府, 電子商取引
 - データウェアハウス
 - 検索エンジン

→データグリッド

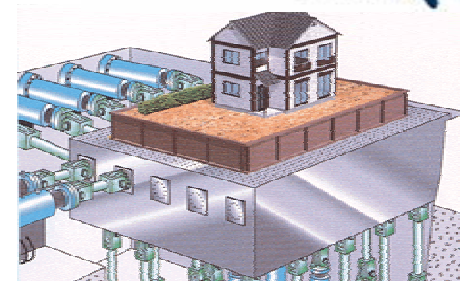


すばる望遠鏡

Large Hadron Collider at CERN



CERN LHC実験



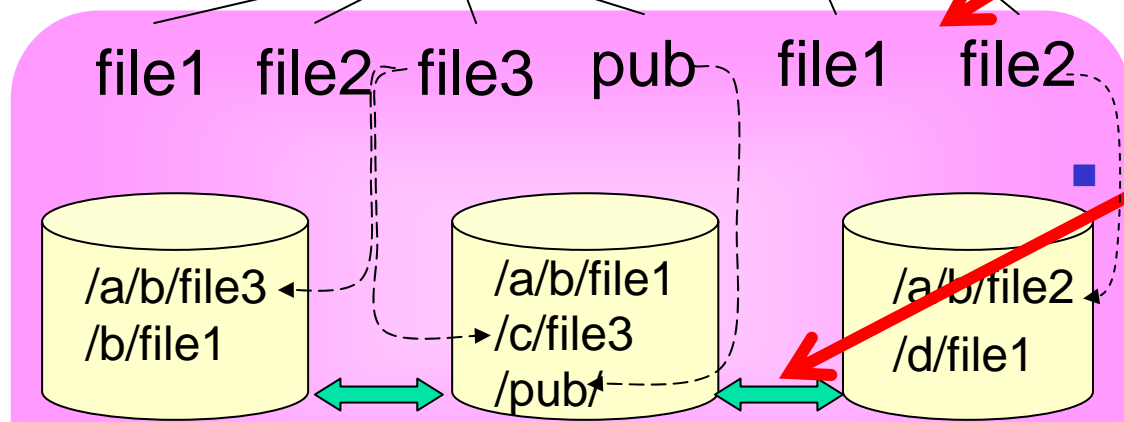
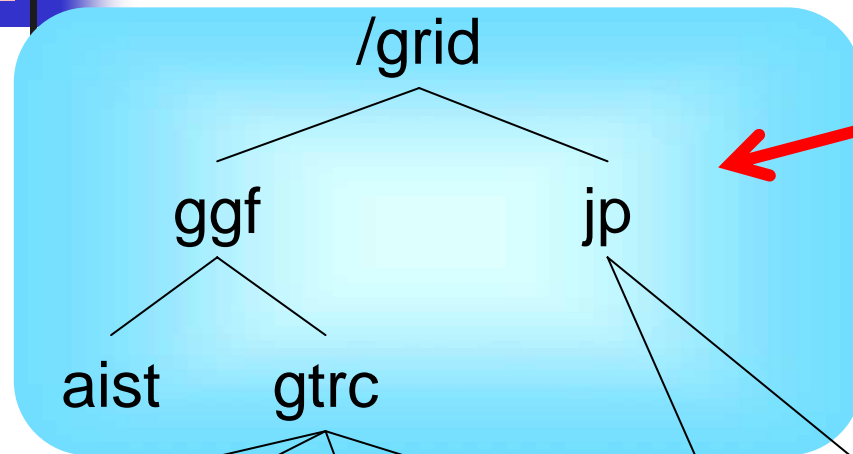
3次元地震シミュレータ



データグリッドにおける要求項目

- 分散した装置, 計算機, 人, 可視化装置を**高速接続, 安全に共有, 高速データ処理**する技術
 - スケーラブルな並列I/Oバンド幅
 - > 100GB/s, > 1TB/s (システム内, システム間)
 - スケーラブルな計算パワー
 - > 1TFLOPS, > 10TFLOPS
 - 安全な認証, 制御されたデータ / プログラム共有, アクセス制限
- システムモニタと管理
- 耐故障性 / 動的再配置 / データ復元, 再計算

Grid Datafarm: グリッド仮想ファイルシステム



<gsiftp://aist.go.jp/> <http://u-tokyo.ac.jp/> <ftp://soumu.go.jp/>

■ 仮想ファイルシステム

- 仮想的な階層構造をもつパス名でファイルアクセス
- 階層的なアクセス制御

■ データ複製管理

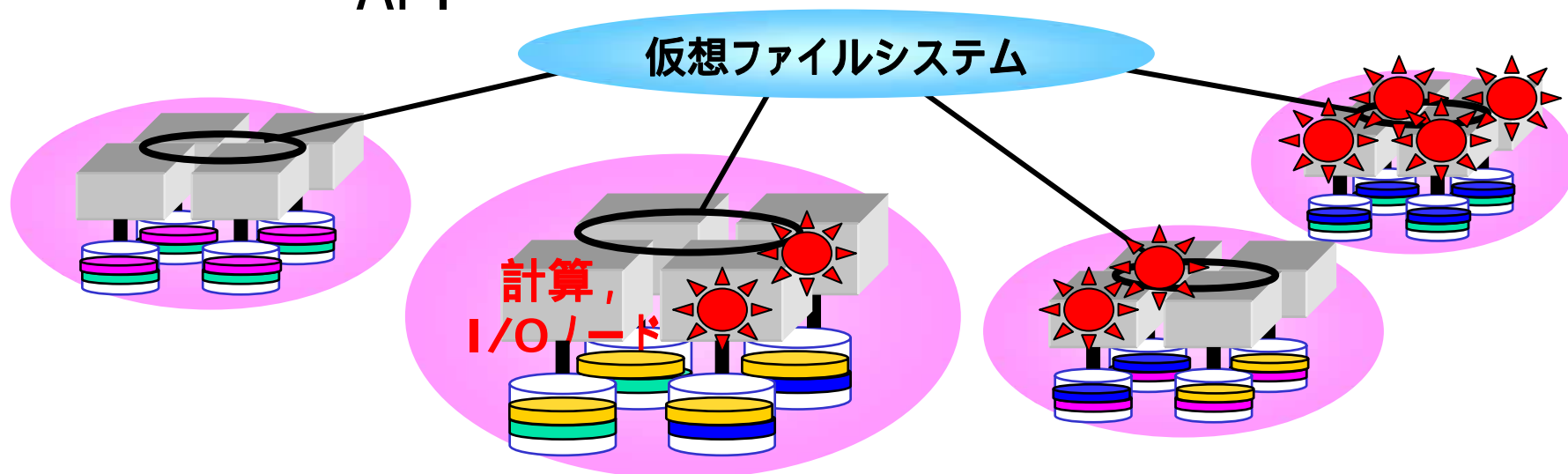
- 耐故障性, 負荷分散
- e.g. OGSA Data Replication Service, Giggle

■ 高速ファイル転送

- ファイル転送, 遠隔ファイルアクセス
- e.g. GridFTP

Grid Datafarm: データ並列サポート

- ローカルI/Oを利用したスケラビリティ
 - データ分散に応じたファイルアフィニティスケジューリング
 - ローカルファイルビュー – グリッド並列I/O API





Grid Datafarmにおけるスケジューリング・複製手法の性能評価

- Grid Datafarmアーキテクチャを想定した評価
- 2007年のLHC実験のパラメータを利用
- Bricksグリッドシミュレータのデータグリッド拡張とBricksによる評価
- データグリッドモデルの比較
 - MONARC型複数サイト分散クラスタ
vs. 単一サイト巨大クラスタ
- スケジューリングとデータ複製手法の比較
 - Owner Computes + バックグラウンドデータ複製
vs. MCT + オンデマンドデータ複製

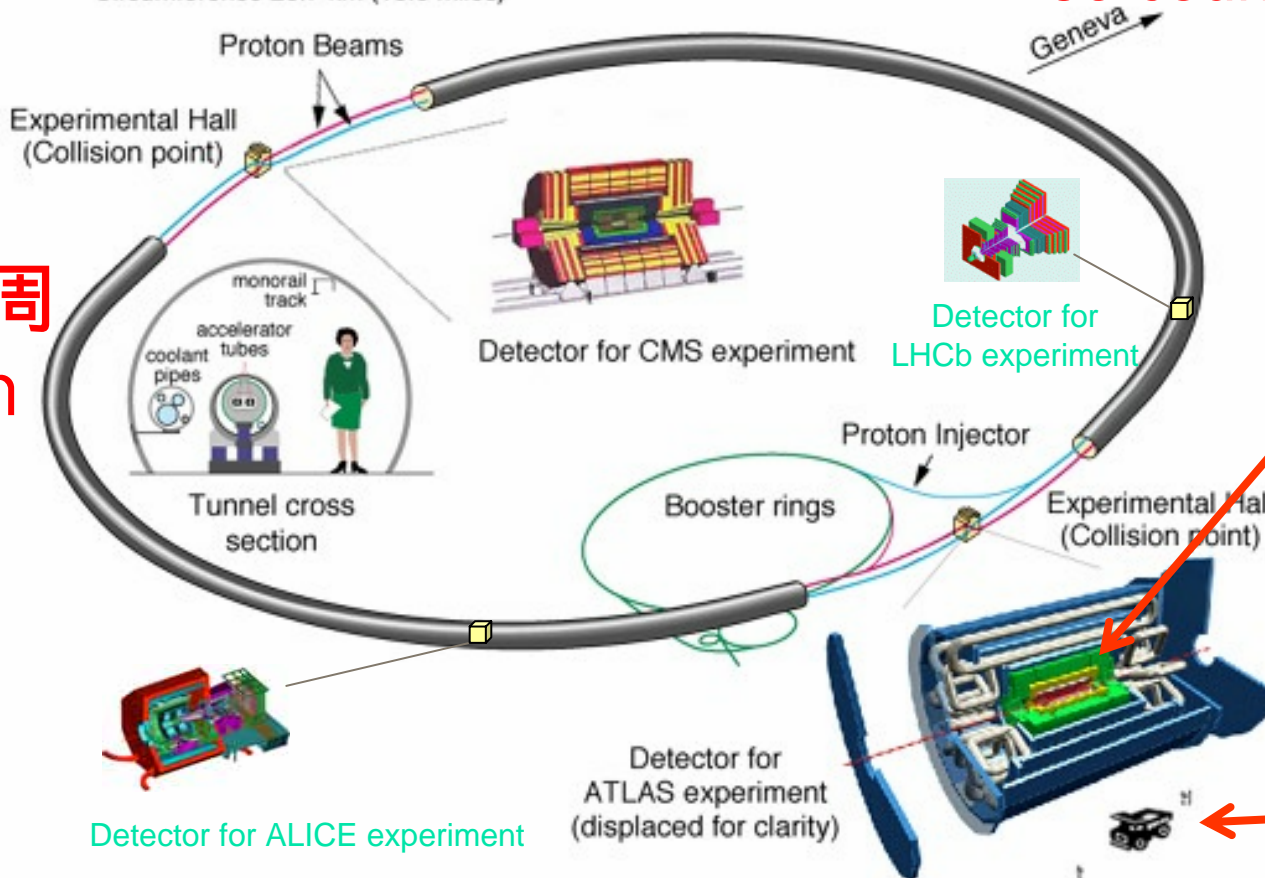
データグリッドアプリケーション: CERN LHC実験

Large Hadron Collider at CERN

Circumference 26.7 km (16.6 miles)

~2000 physicists from
35 countries

LHC円周
26.7km

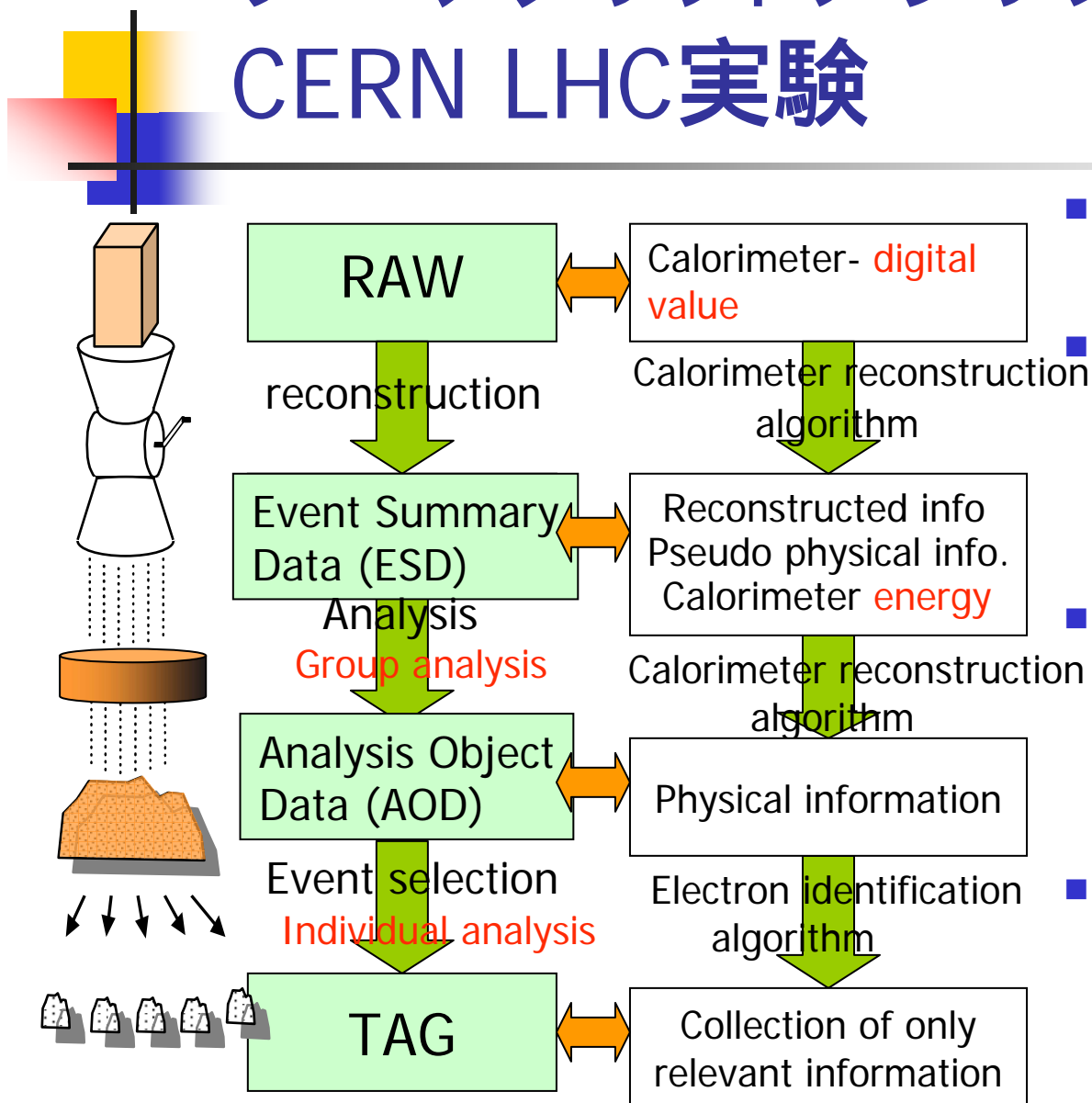


ATLAS
検出器
40mx20m
7000トン

粒子衝突の
観測データを
得て、解析

トラック

データグリッドアプリケーション: CERN LHC実験



- 観測データの段階的な解析
- RAW→ESD (**Large**)
頻度: 2-4/年
入力: 1PB, 出力: 100TB
計算: 1000 GSI95*sec
- ESD→AOD (**Medium**)
頻度: 1/月
入力: 100TB, 出力: 10TB
計算: 25 GSI95*sec
- AOD→TAG (**Small**)
once/4 hours
入力: 10TB, 出力: 0.1TB
計算: 5 GSI95*sec

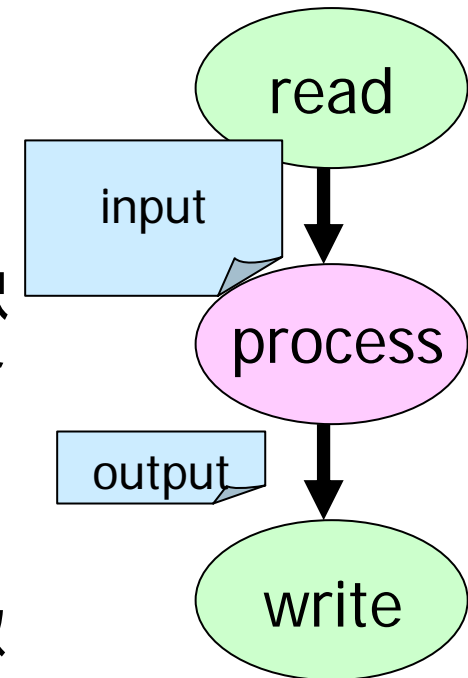
シミュレーションモデル: ジョブ処理

- LHC実験ジョブの処理手順:

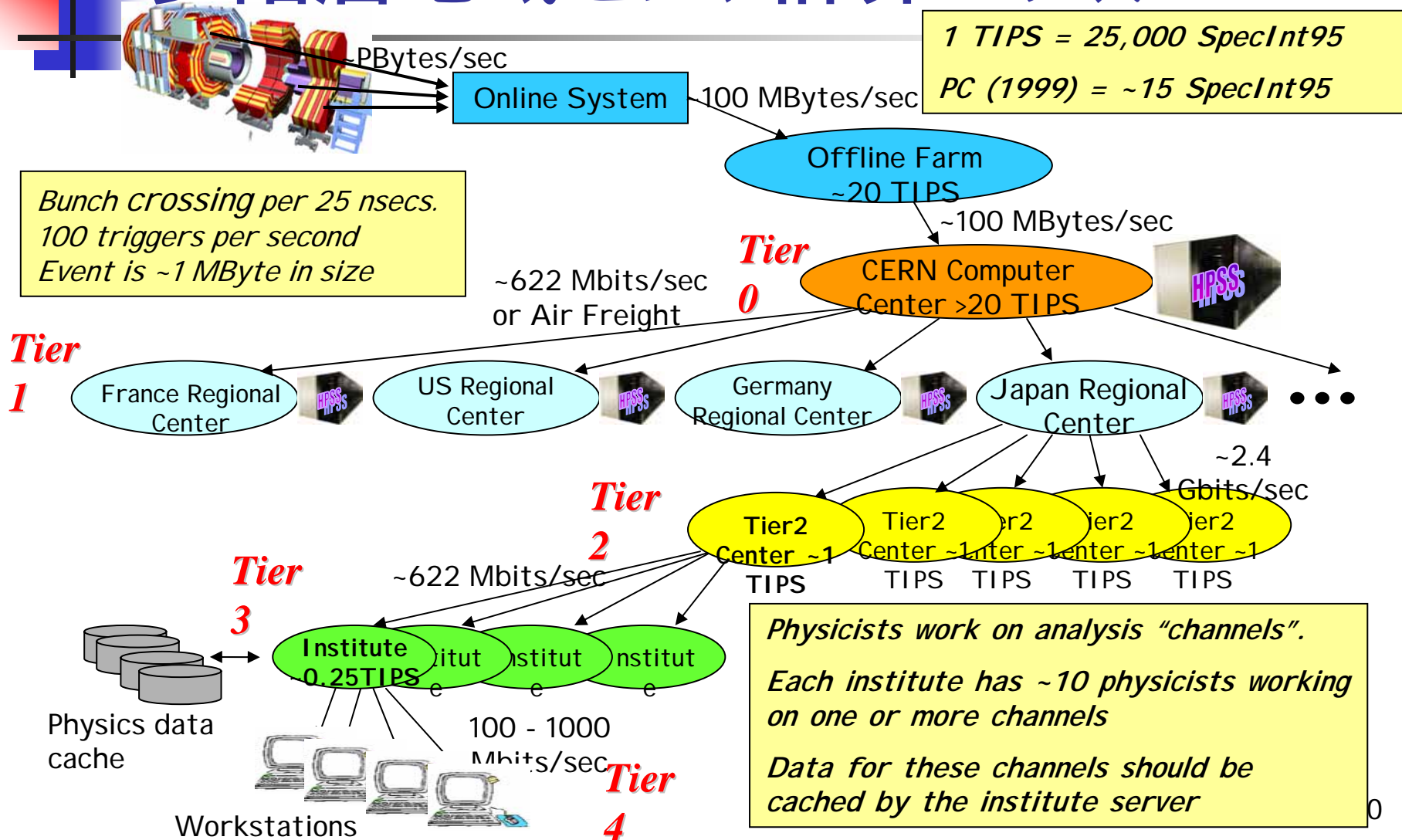
- ユーザ(物理学者)はクライアント計算機から解析ジョブを投入
- データグリッドスケジューラは適切なサーバ群を選択
- ローカルにジョブが必要とするデータがなければ, 各サーバはデータ断片をダウンロード
- サーバは割り当てられたタスクを処理
- サーバは出力データを指定されたディスクに送信
- (クライアントは計算で得られた統計情報のみ受け取る→非常に小さいので無視できる)

- ジョブ処理に要する時間:

$$T_{response} = T_{read} + T_{process} + T_{write}$$



LHCのためのMONARC型 多階層地域センタ計算モデル



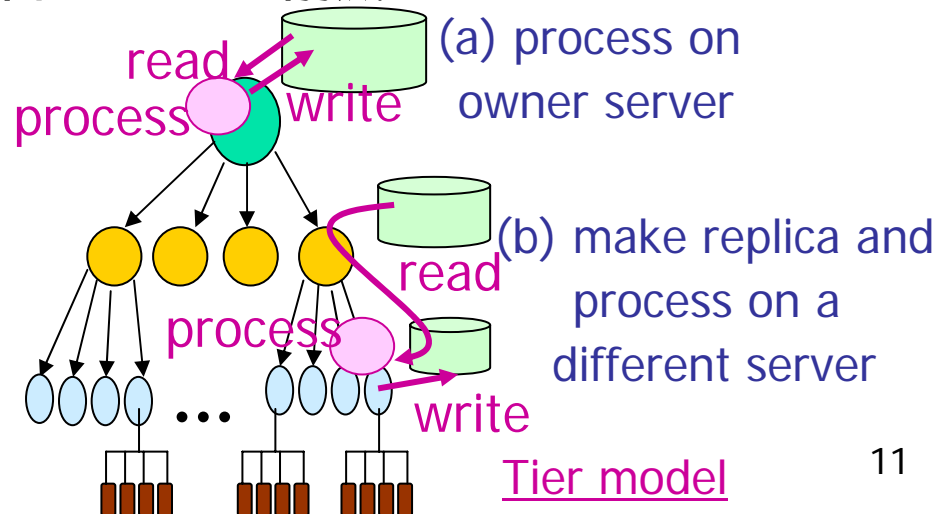
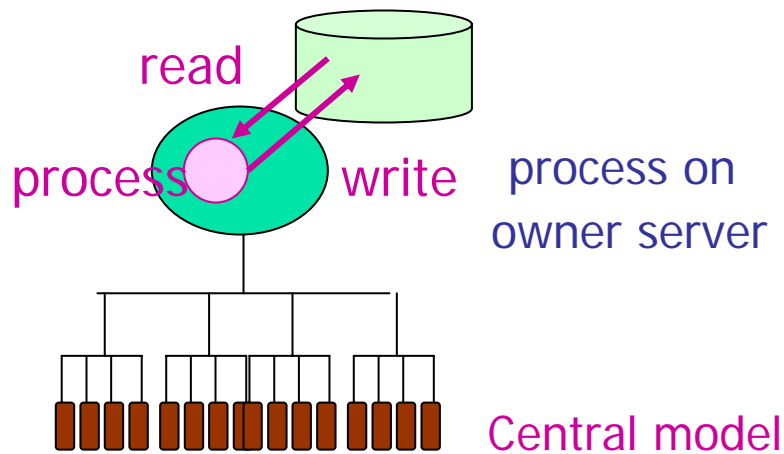
データグリッドモデル

- Centralモデル:

- 単一巨大サイトで全てのジョブを処理
- 性能面, 管理面で効率よいが設備・管理コスト大

- Tierモデル (MONARC型):

- 階層的な地域センタでジョブを分散処理
 - スケジューリングとデータ複製手法を利用
- 1拠点での電力, 予算, 管理コストの削減





Tierモデルのための スケジューリングとデータ複製手法

- Centralモデルではファイルアフィニティスケジューリング(ディスクowner-computerルール)を適用
- Tierモデルのためのスケジューリングとデータ複製手法
 - オンラインスケジューリング手法
(→オンデマンド複製)
 - データ複製手法 (→バックグラウンド複製)



オンラインスケジューリング手法

スケジューラはDataSourceHost(I/Oホスト),
ComputeHost(計算ホスト),
DataDestinationHost(I/Oホスト)を決定

- Greedy(MCT: Minimum Completion Time)
ジョブの応答時間が最短となるホストを選択
- OwnerComputes
ジョブ処理に要するデータを格納しているホストからMCT
となるホストを選択
- LoadBound-Read/-Write
指定した性能を超えるホストから, MCTとなるホストを選択
(I/Oホストと計算ホストが異なる場合は, -Readでは読み
込み時に, -Writeでは書き出し時に適宜複製を作成)



バックグラウンド複製手法

- 複製マネージャは定期的にシステム上の計算ホストの状況を調べ、適宜複製を作成
- **LoadBound-Replication:**
 - データを格納しているホストに対し $Perfestimated$ を算出
$$Perfestimated = Perf / (LoadAvg + 1)$$
 - $Perfspecified > Perfestimated$ の場合、 $Perfestimated$ が最小のホストから最大のホストへアクセス率 AR の高いデータの複製を生成
$$AR = Naccesses / (Tcurrent - Tstored)$$
- **Aggressive-Replication:**
ジョブが終了すると、 $Perfestimated$ が最大のホストへデータの複製を生成



複製削除アルゴリズム

複製の作成でデータグリッドシステム上のディスク領域が不足した($x\%$ のホストの空きディスク領域が $y\%$ になった)場合, 複製を削除する

1. システム上に複製を持つデータのリストを作成
 2. 1のリストを最後にアクセスされた時刻(LRU)が古い順にソート
 3. 2のリストの最初から M 個のデータに対し, アクセス率 A_{Relim} を算出 (N_{copies} は複製の総数)
$$A_{Relim} = N_{accesses} / (T_{current} - T_{stored}) / N_{copies}$$
 4. 以下の条件を満たすまで最初の N 個のデータから, A_{Relim} が最小となるデータを削除 ($Compactness$ は削除頻度を決定するパラメータ)
$$TotalDiskSize \times Compactness > AvailableDiskSize$$
 5. 4の条件が満たされなければ, 3に戻る
- (シミュレーションでは $x=80$, $y=90$, $N_{accesses}=10$, $Compactness=0.9$)



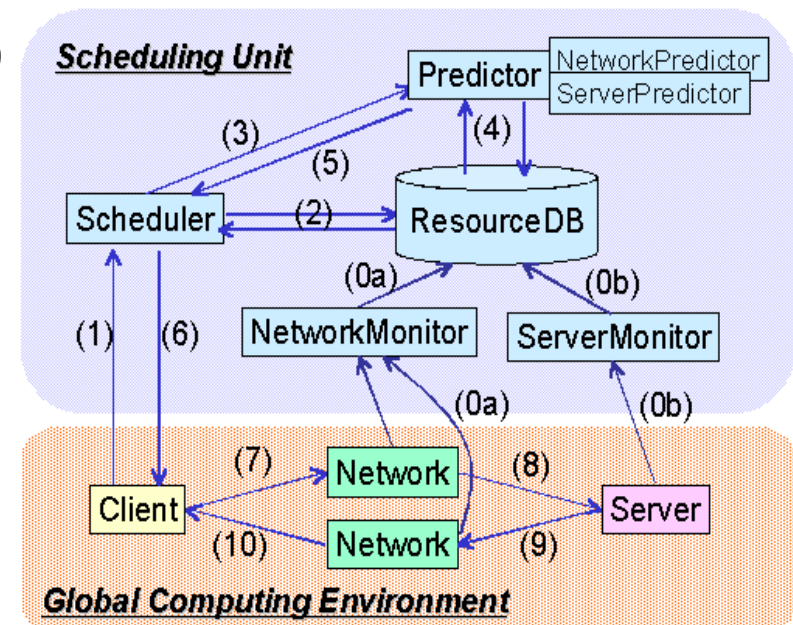
シミュレーションによる評価

- Bricksを用いたシミュレーションによる評価
- LHC実験を想定した性能評価
 - Grid Datafarmアーキテクチャ
 - Centralモデル vs. Tierモデル
 - Tierモデルでは12通りのスケジューリング・複製手法
 - データアクセスの局所性(ランダム/**時間的局所性**)
- 評価指標
 - **平均応答時間**

Bricksグリッドシミュレータ

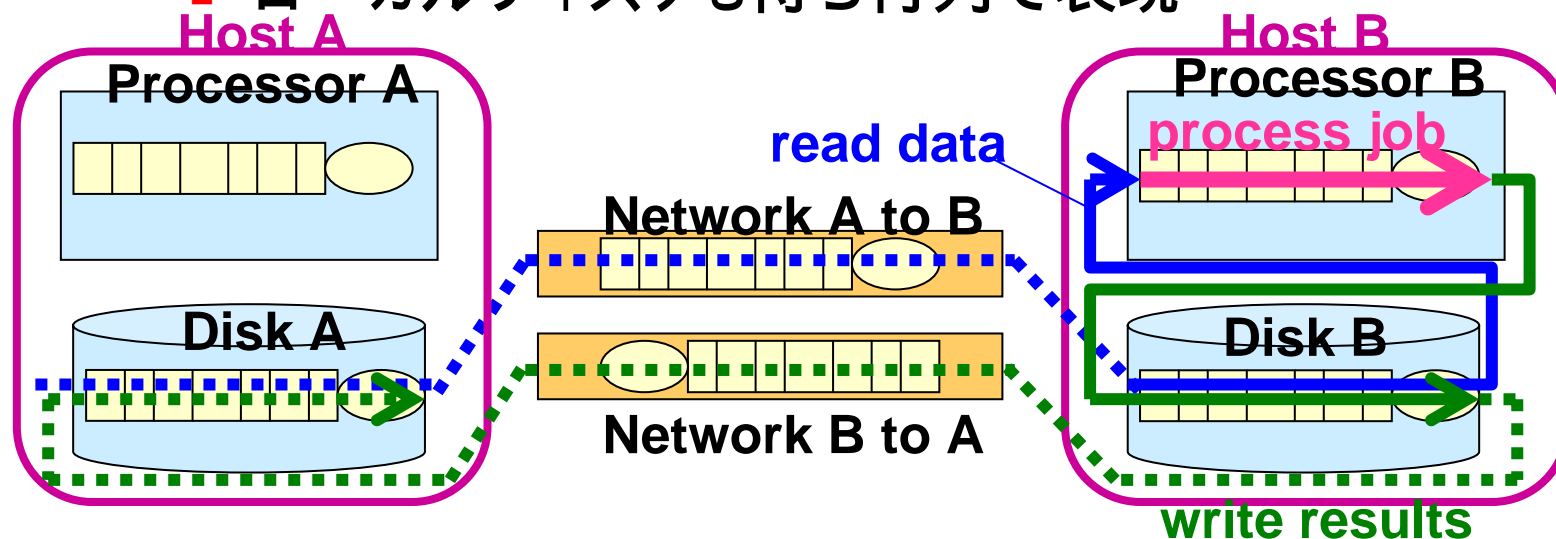
- Javaベースの離散イベントシミュレータ
- ネットワーク, サーバを待ち行列で表現
- 一般的なグリッドスケジューリングモジュールの提供
- 柔軟なシミュレーション環境設定
 - グリッドトポロジ
(e.g. 階層ネットワークトポロジ)
 - サーバ, ネットワークモデル
 - クライアントモデル
- 動的なグリッド環境で様々なスケジューリング手法の性能解析が可能

<http://grid-team.is.titech.ac.jp/bricks/>



Bricksのデータグリッド拡張

- スケジューリングユニットに複製マネージャを追加
- 複製カタログの提供
- ディスク管理機構
- ローカルディスクI/Oオーバーヘッドの表現
 - ローカルディスクも待ち行列で表現



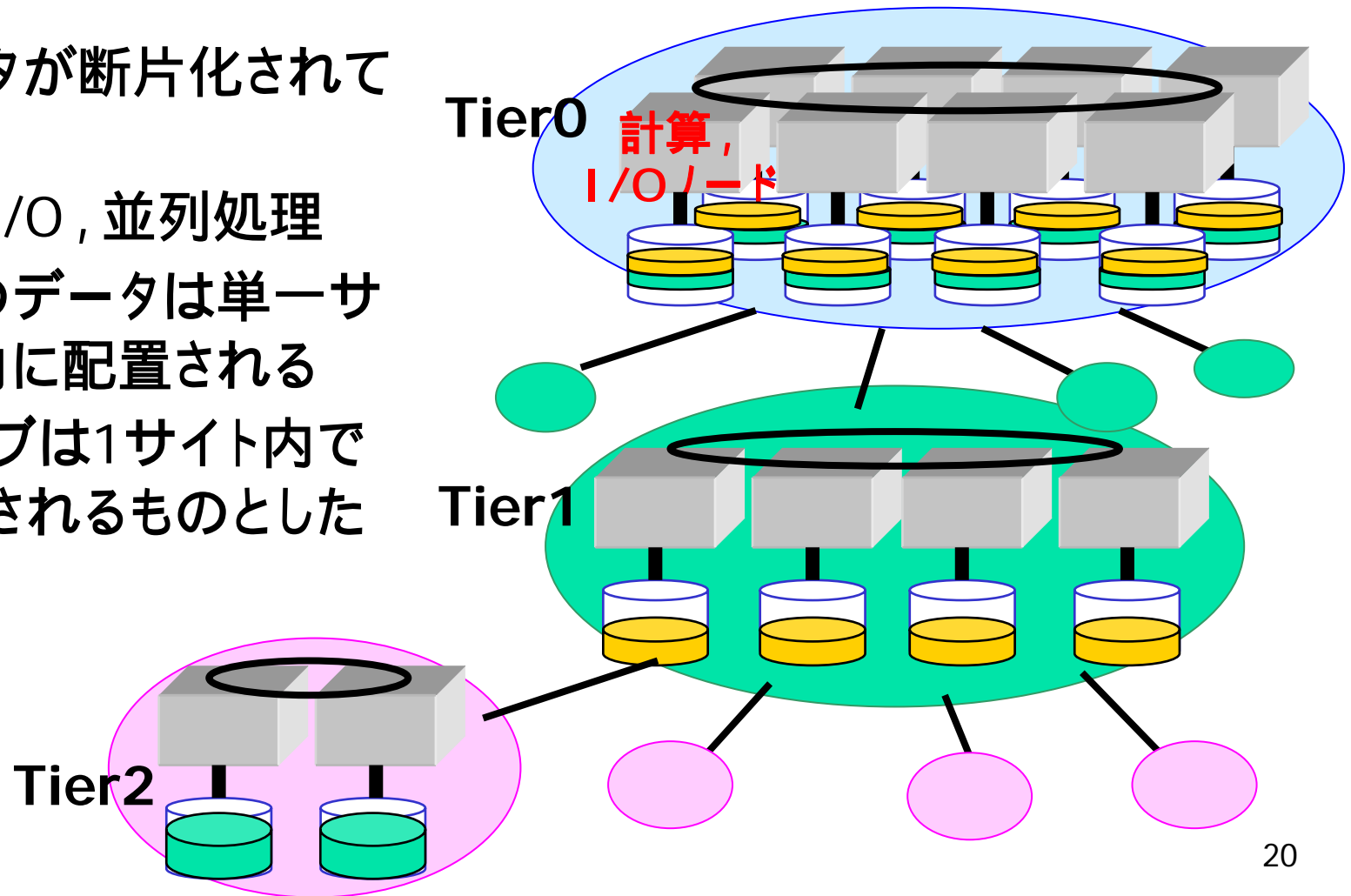


シミュレーションによる評価

- Bricksを用いたシミュレーションによる評価
- LHC実験を想定した性能評価
 - Grid Datafarmアーキテクチャ
 - Centralモデル vs. Tierモデル
 - Tierモデルでは12通りのスケジューリング・複製手法
 - データアクセスの局所性(ランダム/時間的局所性)
- 評価指標
 - 平均応答時間

Grid Datafarmシステムの性能 評価設定

- データが断片化されて管理
- 並列I/O, 並列処理
- 1つのデータは単一サイト内に配置される
- 1ジョブは1サイト内で実行されるものとした



シミュレーション設定：環境

モデル	ディスク容量 [PB]	サイトの総性能 [MSpecInt95]	サイト内 ノード数	総I/Oバンド幅
Central	2	0.5-1.8	10000	1[TB/sec]
Tier	T0 (x1): 2	0.6/0.5/ 0.4	10000	1[TB/sec]
	T1 (x4): 1	0.3/0.25/ 0.2	5000	500[GB/sec]
	T2 (x16): 0.1	0.03/0.025/ 0.02	500	50[GB/sec]

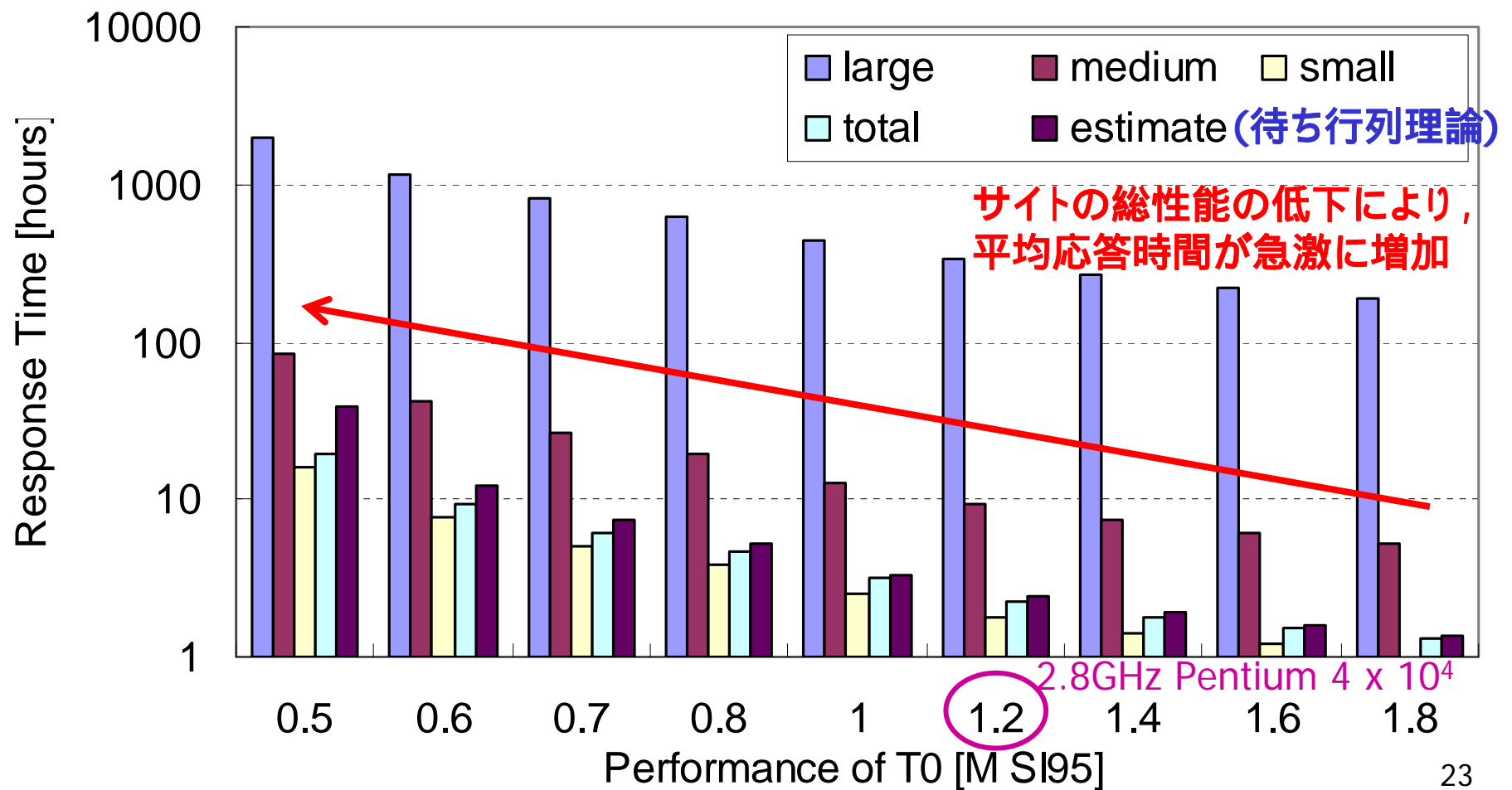
- Tierの性能比はGriPhyNシミュレーション[Grid2001]と同じ
- 待ち行列理論でのCentralの平均応答時間の見積もり
38.575-1.337 [hours] , 0.453318[MSI95]で飽和
- WAN/ローカルI/Oバンド幅は10[Gbps]と100[MB/sec]
- Grid Datafarmアーキテクチャでは,
総I/Oバンド幅=ローカルI/Oバンド幅×サイト内ノード数

シミュレーション設定: ジョブ

Job	# events / Job	計算サイズ [GSI95*sec]	頻度 (Avg.)	入力サイズ[TB]	出力サイズ [TB]
Large: RAW→ESD	1G	1000	1/4 [月]	1000	100
Medium: ESD→AOD		25	1/1 [月]	100	10
Small: AOD→TAG		5	1/4[時間]	10	0.1

- MONARCレポートで挙げられたLHC実験での実パラメータ
- 全シミュレーションでTier0サイトにデータ (1PBx1, 100TBx2, 10TBx4) を格納
- 各アルゴリズムに対し, 1年間のシミュレーション×10を実施
- 東工大Presto IIIクラスタ (Dual Athlon MP 1900+, 768MB memory, 256 nodes) 上でシミュレーションを実行

Centralモデル平均応答時間



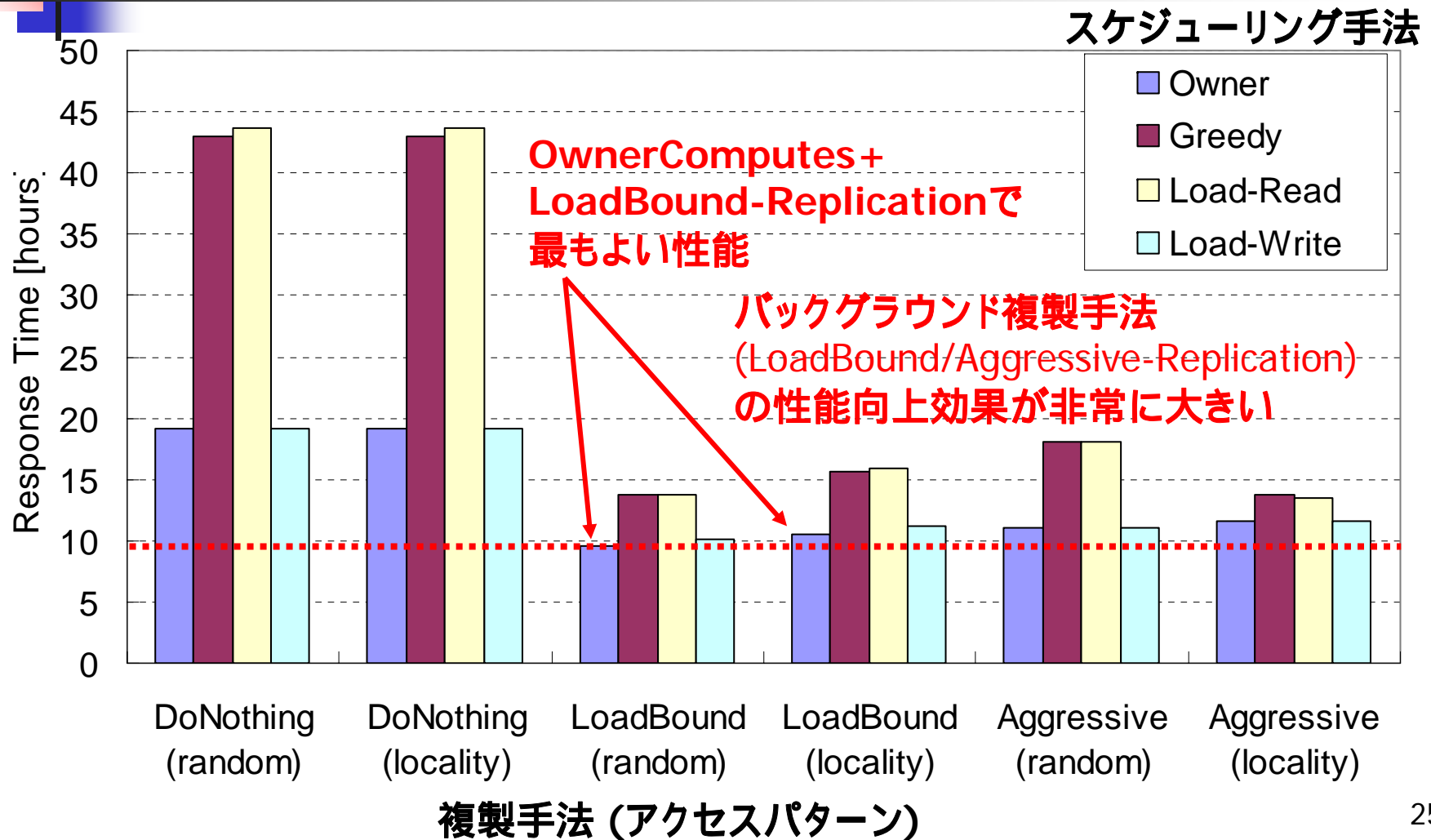


Tierモデルでのスケジューリングと複製手法の組み合わせ

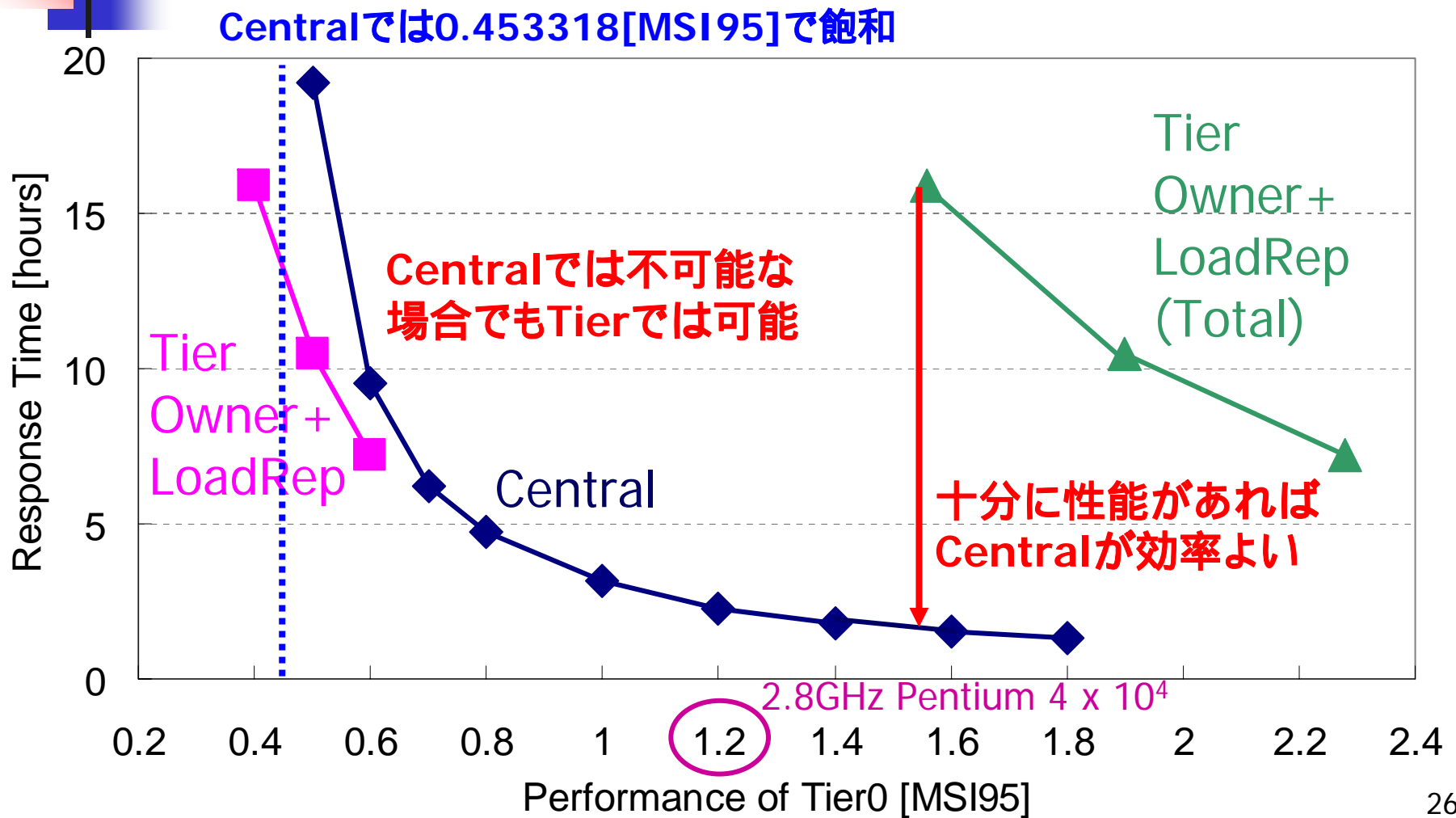
- スケジューリング手法(オンデマンド複製)
 - Greedy
 - OwnerComputes
 - LoadBound-Read
 - LoadBound-Write
- 複製手法(バックグラウンド複製)
 - LoadBound-Replication
 - Aggressive-Replication
 - DoNothing (複製をまったく作らない)
- スケジューリング手法×複製手法=計12通り

Tierモデル平均応答時間

$$(T_0, T_1, T_2) = (0.5, 0.25, 0.025)$$



CentralモデルとTierモデルの 平均応答時間の比較





関連研究: GriPhyNのシミュレーションによる評価[HPDC-11, '02]

- GriPhyN [Univ. of Chicago et al]
 - CERN LHC実験解析がターゲット
 - Globusベースのペタスケール仮想データグリッド構築
- シミュレーションによるスケジューリングと複製手法の評価
 - 外部スケジューラ, ローカルスケジューラ, データセットスケジューラによるシステムモデルを提案
 - 外部スケジューリングとデータセットスケジューリング手法の評価
 - JobDataPresent (OwnerComputes) + 複製手法でよい性能
 - LHC実験パラメータでない
(ジョブ粒度小, 短期間, データの増加なし)

→Grid Datafarmアーキテクチャを対象
LHC実験パラメータを利用した評価



まとめと今後の課題

- Bricksグリッドシミュレータをデータグリッドに拡張し、データグリッドモデルの性能を評価
- Grid Datafarmアーキテクチャ+LHCを想定し、CentralモデルとMONARC型Tierモデルの比較
 - バックグラウンド複製が有効
 - 十分な計算性能を確保できればCentralが効率よい
 - 確保できない場合、Tierで適切なスケジューリング・複製手法を用いることでLHCジョブの処理が可能
- 効率的な複製削除方法の考案
- より適切なスケジューリング・複製手法を提案し、大規模環境で様々な評価していく