

# 地球シミュレータと High Performance Fortran

地球シミュレータセンター  
(<http://www.es.jamstec.go.jp>)

村井 均  
([murai@jamstec.go.jp](mailto:murai@jamstec.go.jp))

# はじめに

---

- 地球シミュレータ(ES: Earth Simulator)は、現在世界最高速のスーパーコンピュータである。
  - ESでは、並列化の手段として、High Performance Fortran (HPF)を利用できる。
  - HPF推進協議会は、平成14年度より、地球シミュレータ共同プロジェクト「並列処理言語 HPF(High Performance Fortran)を用いた大規模並列実行の性能検証および新規機能の検討」を実施。
- ➡ 地球シミュレータとHPFの現状を報告する。

# 目次

---

## 1. 地球シミュレータ

- ハードウェア構成
- 施設・設備
- プログラミング環境
- 運用
- 共同プロジェクトと成果

## 2. High Performance Fortran

- HPFの概要
- HPF/ESの機能
- 地球シミュレータ共同プロジェクト
- PCクラスタ用HPFコンパイラ

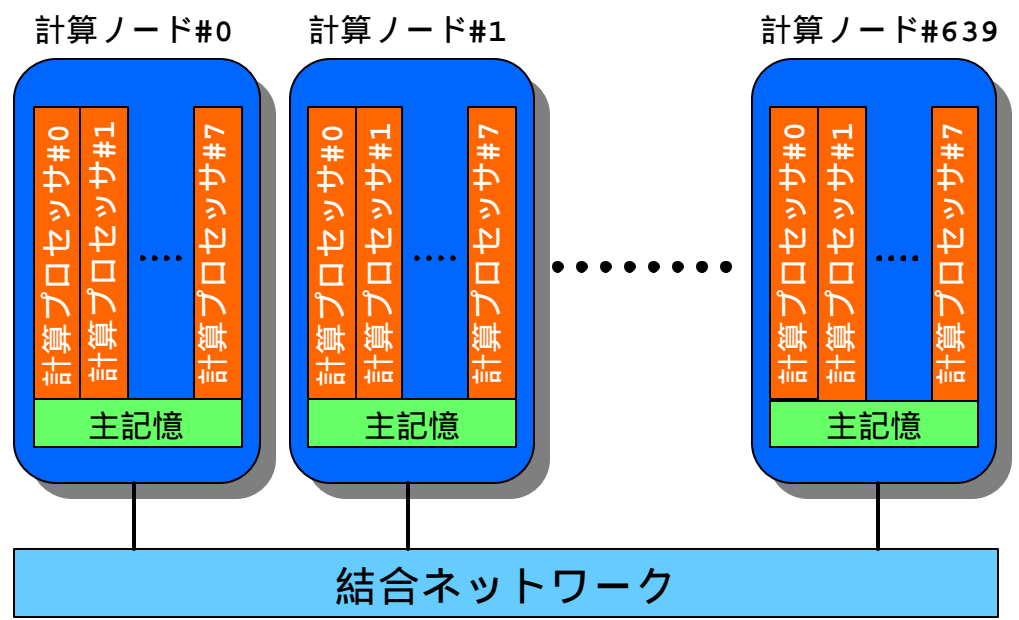
# 第一章: 地球シミュレータ

- 地球環境変動の解明・予測を目的とし、宇宙開発事業団(現 宇宙航空研究開発機構), 日本原子力研究所, 海洋科学技術センターの3法人が開発。
- 2002年3月より運用を開始。





# ハードウェア構成



- 640台の計算ノード(PN: Processing Node)を単段クロスバネットワークで結合した分散メモリ型並列計算機システム。
- 各PNは、メモリを共有する8台のベクトル型計算プロセッサ(AP: Arithmetic Processor)から成る。

PN内のAP数	8
総PN数	640
総AP数	5120
APのピーク性能	8Gflops

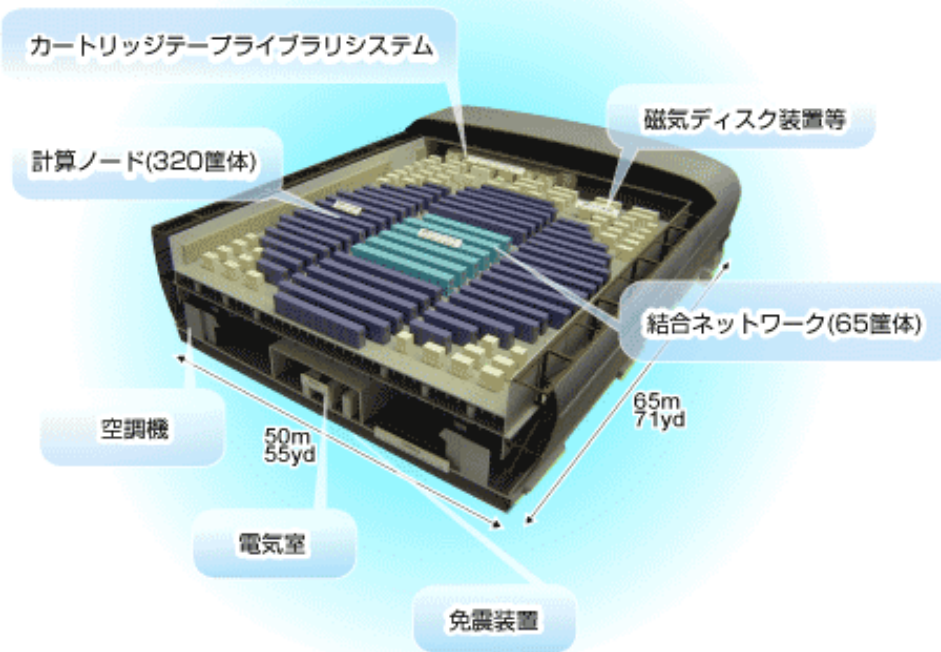
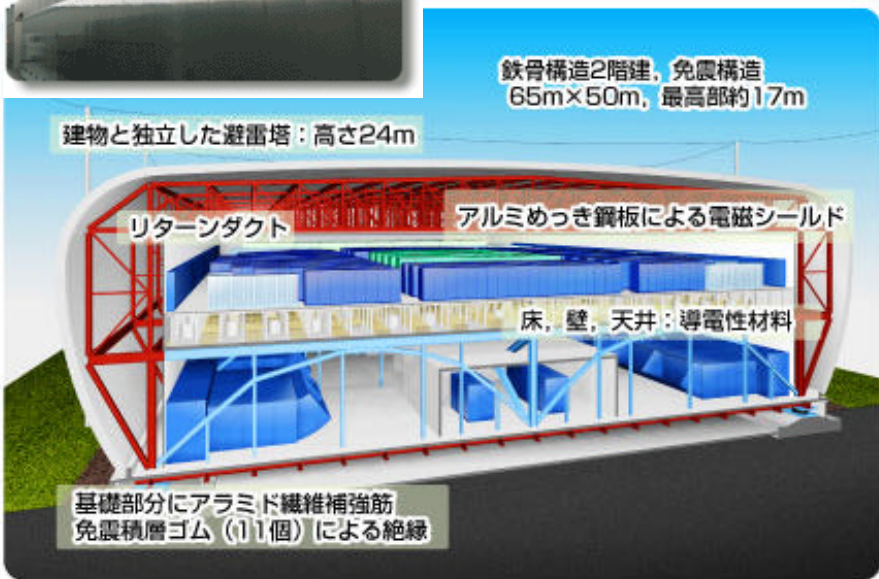
ピーク性能	40Tflops
PN内主記憶要領	16GB
主記憶容量	10TB
PN間通信性能	12.3GB/秒(双方向)

# 施設・設備



マシンルーム外の光源から、ライトチューブを通じて照明を行うライトガイド方式 (日本照明工学研究所「照明普及賞」受賞)

電磁波や地震の影響からシステムを保護するためにさまざまな設備を備える。

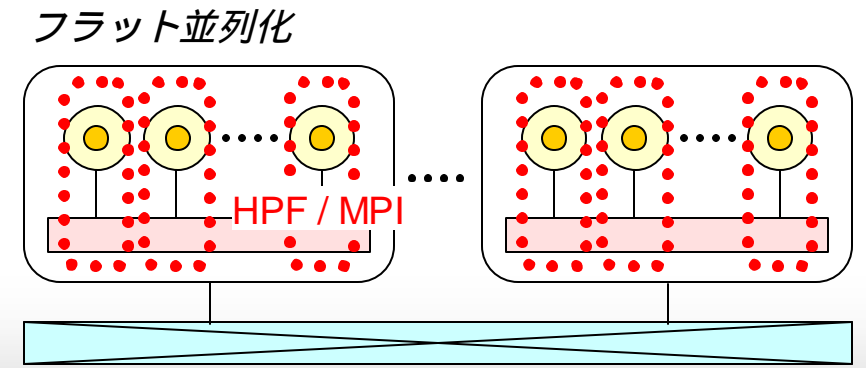
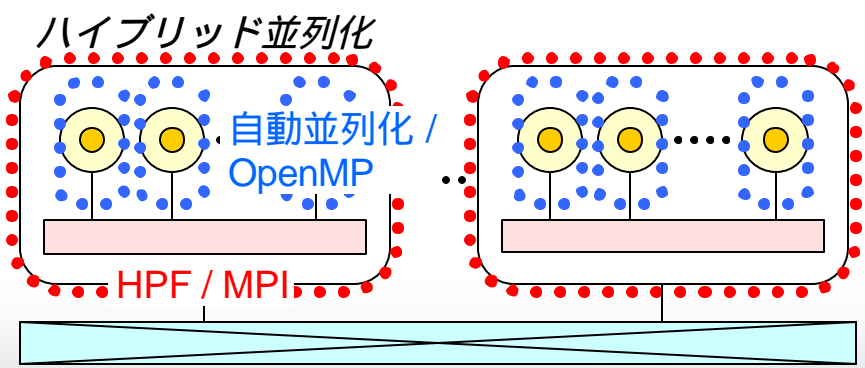


# プログラミング環境

HPF, MPI, OpenMPと、コンパイラの自動並列化/ベクトル化の機能を利用して、ESの階層並列性を活かしたプログラミングが可能。

- 二つの並列化方法:
- ハイブリッド並列化
  - フラット並列化

	ハイブリッド並列化	フラット並列化
PN間並列処理	HPF / MPI	HPF / MPI
PN内並列処理	自動並列化 / OpenMP	
ベクトル処理	自動ベクトル化	

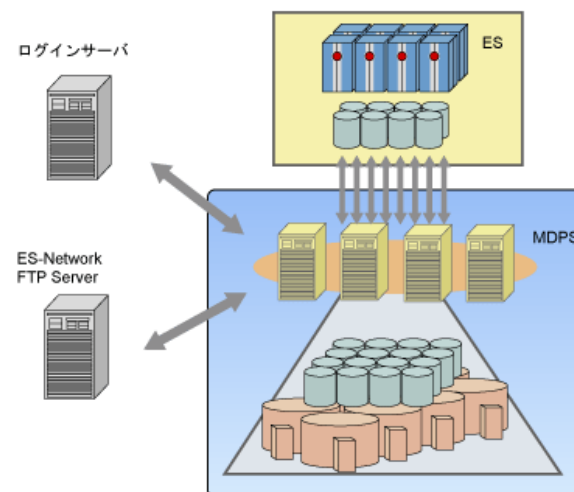


# 運用

- 640ノードを用途別に3種に分割
- 一つのノードは一つのジョブのみを実行
- キューイングシステムNQSIIIによるジョブ管理
- MDPS (Mass Data Processing System)により、高速かつ容易に大規模データを利用可能

## 計算ノードの分割運用

	ノード数	用途
インタラクティブ	2	コンパイル,デバッグ
Sクラスタ	14	シングルノード実行
Lクラスタ	624	マルチノード実行



# 共同プロジェクト

平成15年度の採択プロジェクト:

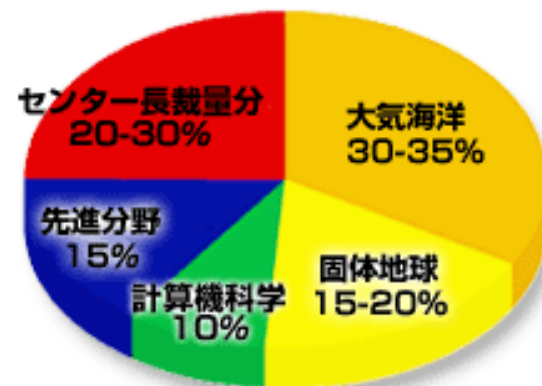
- 大気・海洋分野 12件
- 固体地球分野 9件
- 計算機科学分野 2件

「並列処理言語HPF(High Performance Fortran)を用いた大規模並列実行の性能検証および新規機能の検討」

- 先進・創出分野 11件



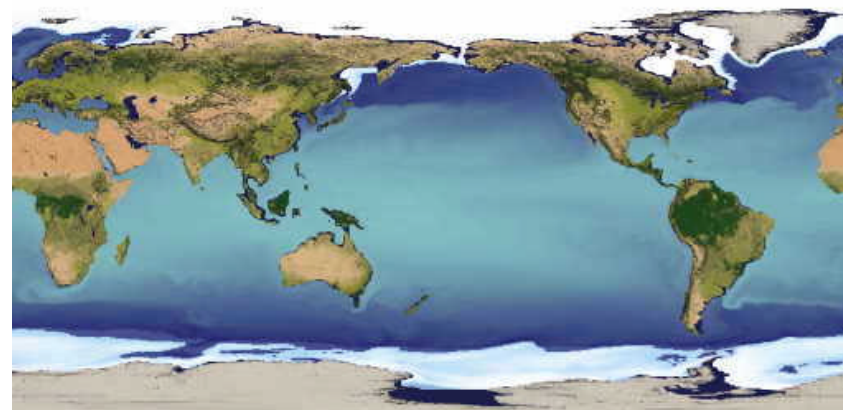
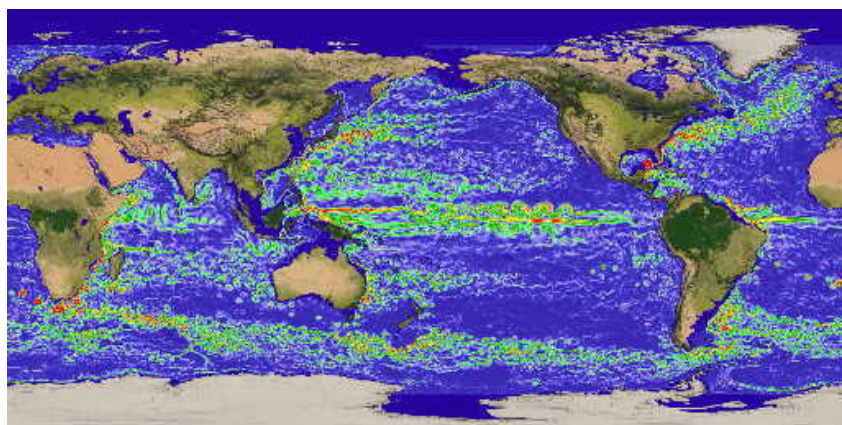
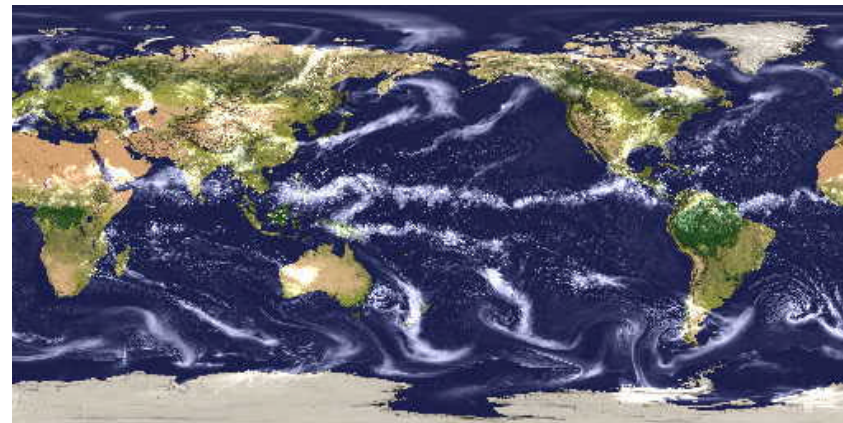
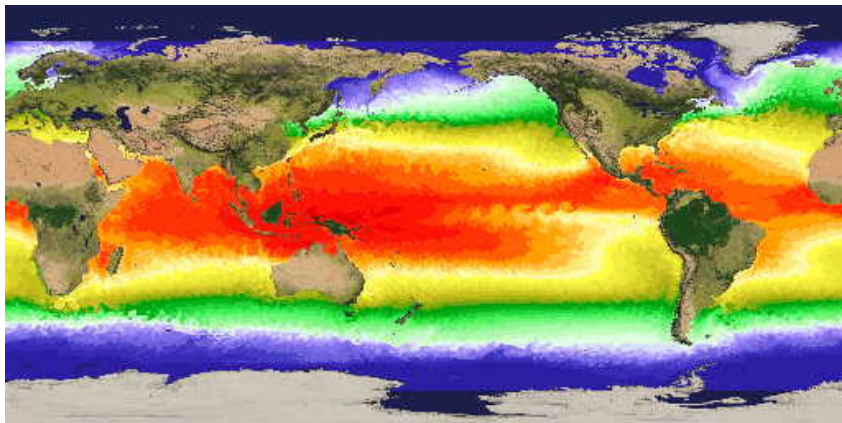
利用報告会(2004/1/10-11)



計算リソースの配分



# 成果の例(1) 大気・海洋



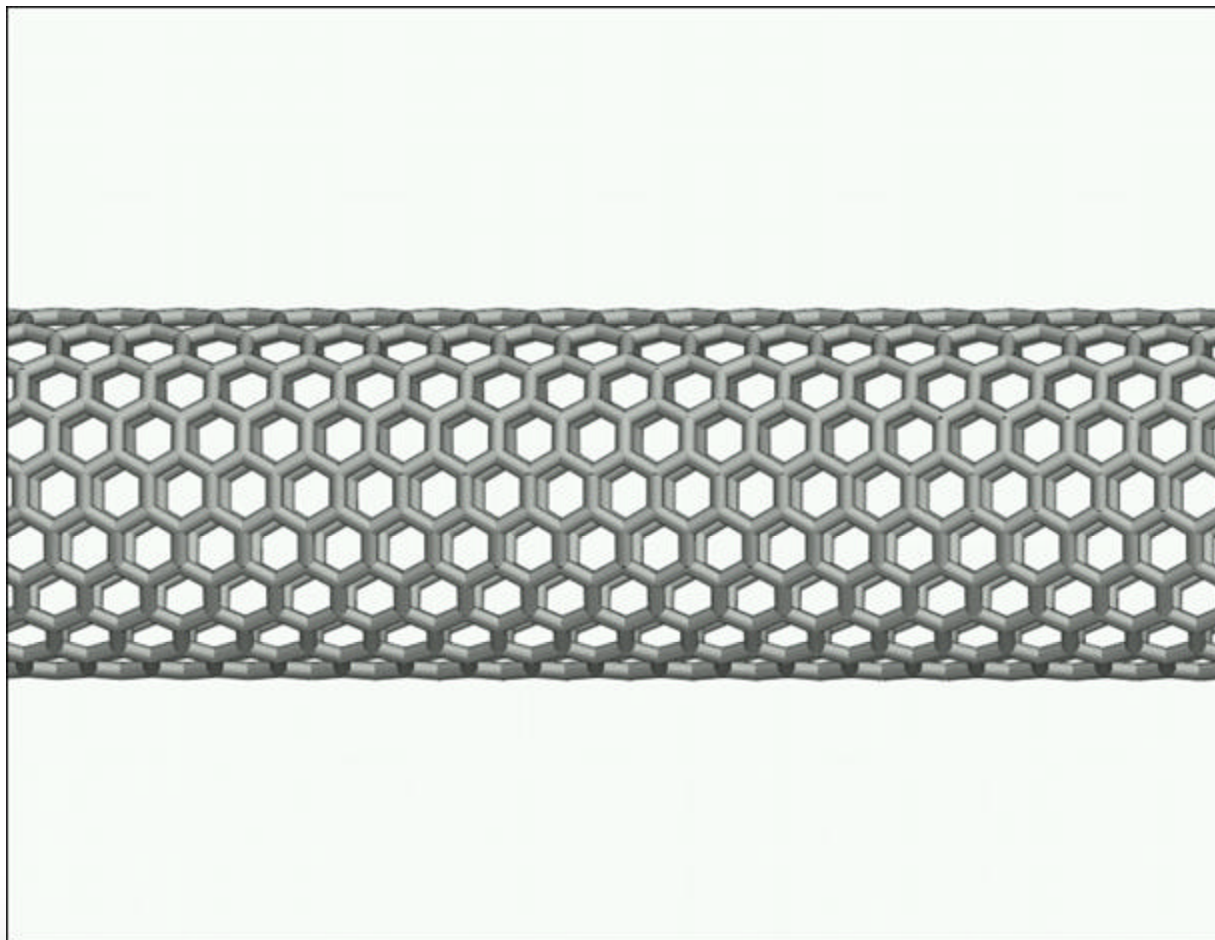
# 成果(2) 地震波の伝播

---



# 成果(3) カーボンナノチューブ

---

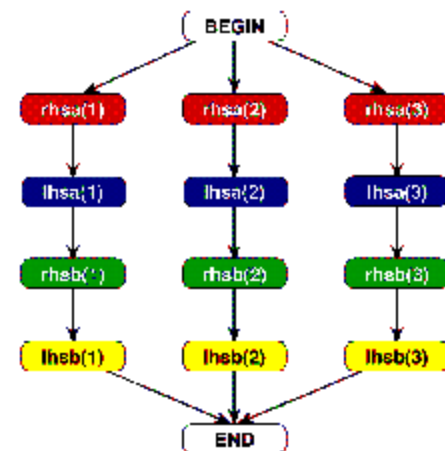




# 第二章: High Performance Fortran

High Performance Fortran (HPF)  
 = Fortranの拡張として定義される  
 データ並列言語

- 米国のHPF Forumが、1993年にHPF1.0を、1996年にHPF2.0を制定。
- 日本のHPF合同検討会(2001年にHPF推進協議会へ改組)が、HPF/JA拡張仕様を提案。



# HPF言語の特長

簡単な「指示文」の形式で(a)データの分割だけを指定すれば、コンパイラが(b)計算の分割と(c)通信の挿入を自動的に処理する。

➡ 小さな手間で、並列化を実現できる。

	MPI	HPF
(a) データの分割	手動	手動
(b) 計算の分割	手動	自動
(c) 通信の挿入	手動	自動

# 簡単な例

```

DIMENSION A(1000,1000)
!HPF$ DISTRIBUTE A(*,BLOCK)
!HPF$ INDEPENDENT
DO J = 1, 1000
  DO I = 1, 1000
    A(I,J) = 0.0
  END DO
END DO

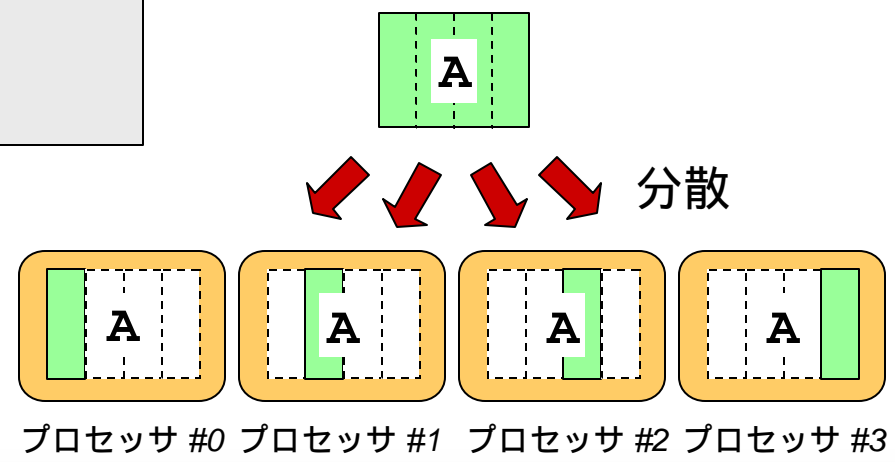
```

各プロセッサへ配列を分散する方法を指定。

直後のループが並列実行可能であることを宣言 オプショナル

各プロセッサは、自分が保持する配列部分に対する計算を実行する。

➡ オーナコンピュートルール



# HPF/ESの特徴

---

## 地球シミュレータ上のHPFコンパイラ

---

- 標準仕様に加え、さまざまな拡張仕様をサポート
  - HPF2.0基本仕様, 公認拡張仕様
  - HPF/JA1.0拡張仕様
- ESの持つ階層並列性を利用可能
- 不規則問題向け独自拡張
- 並列入出力

# HPF/ESの機能

## 公認拡張

- ポインタのマッピング
- 構造体成分のマッピング

- INDIRECT

## HPF2.0 (core)

- ON
- GEN\_BLOCK
- SHADOW
- 再マッピング

- タスク並列
- RANGE
- etc.

## HPF/ES

- HALO
- バッチデバッグ
- ベクトル化/マイクロタスク指示行
- 自動並列化
- 並列入出力
- MPI I/F

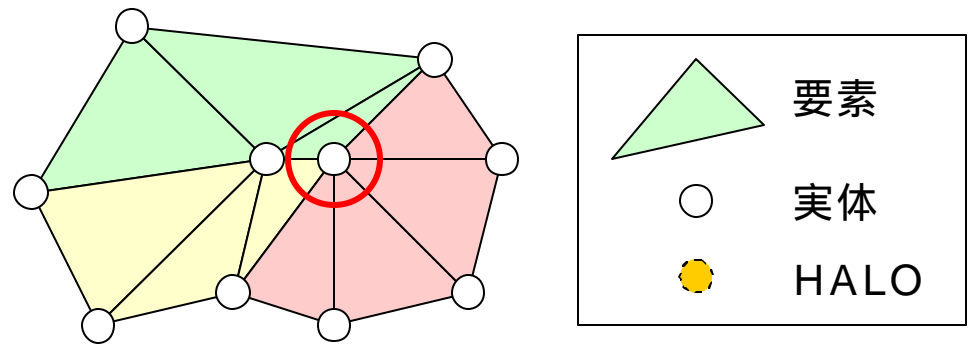
- LOCAL & REFLECT
- リダクション種別
- 通信スケジュールの再利用

- 非同期通信
- etc.

## HPF/JA

# HALO

HPFでは並列化が難しいとされる不規則問題(有限要素法など)を扱うための機能

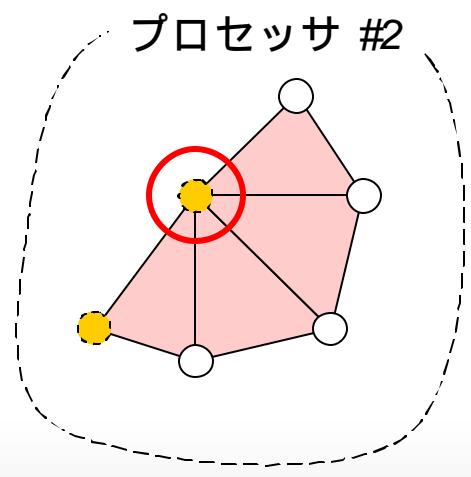
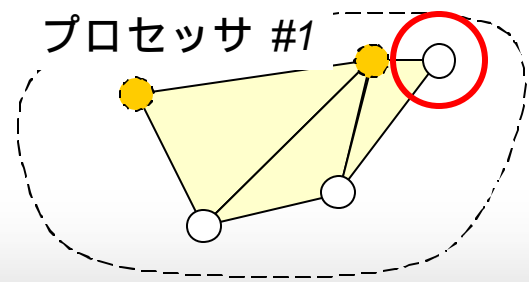
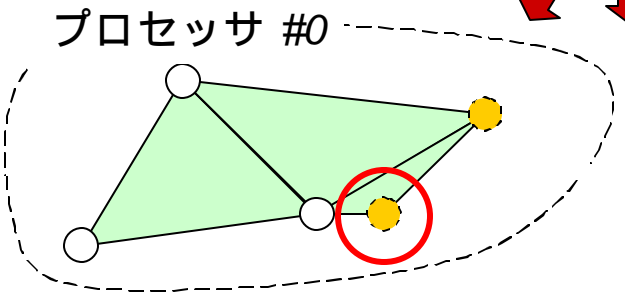


分散

分散境界上の節点は、  
• 一つのプロセッサに実体として、  
• 他のプロセッサにHALOとして、  
割り付ける。



「不規則な」アクセスや通信を効率よく処理できる。



# 並列入出力

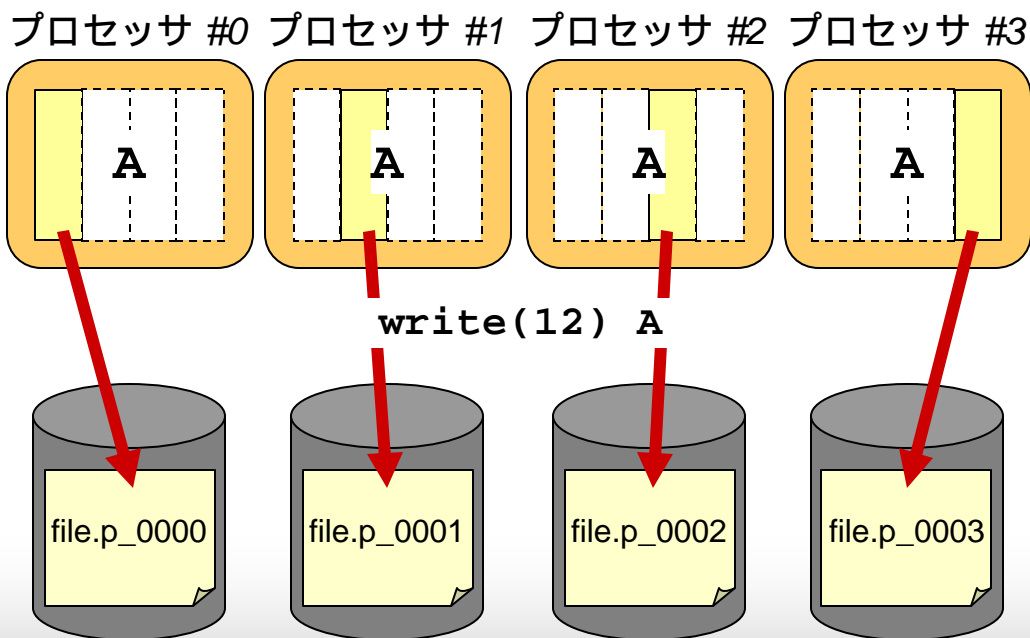
指定したファイルに対して、全く意識することなく並列に入出力が実行される。

(1) 12番の装置に対して、  
並列入出力を指定。

```
% setenv HPF_FFUNIT 12
```

並列ファイルを通常のFortranファイルへ変換したり、並列ファイルの再分散を行うツールが提供される。

(2) 並列入出力を実行



# HPFPROFプロファイラ

ループ単位の性能情報をGUIで表示(CUIでも利用可能)

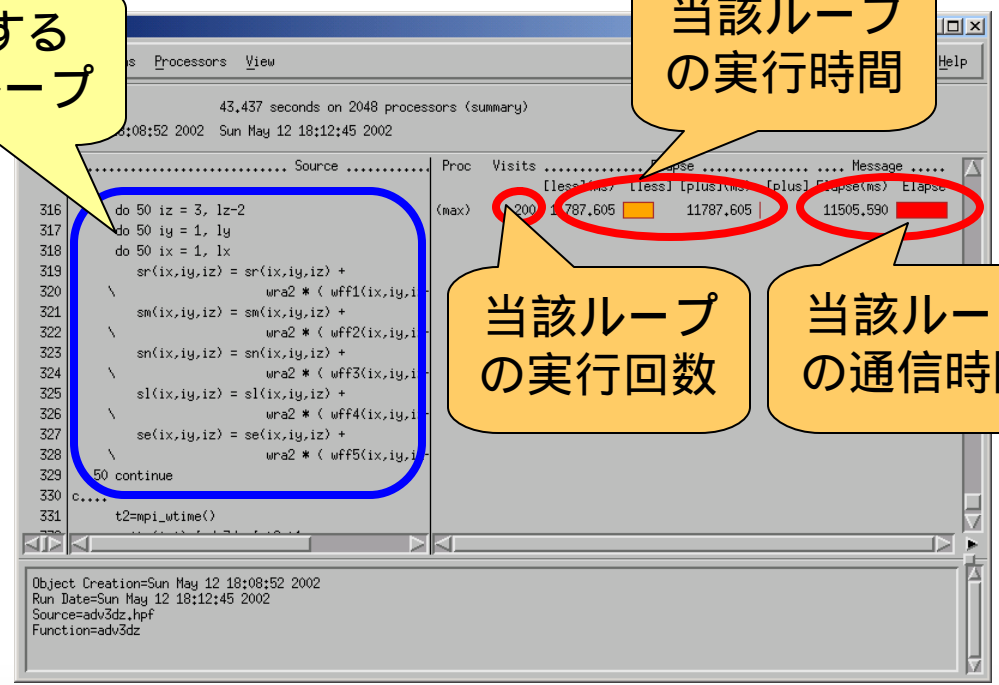
- 直感的に情報を把握できる。
- ループ単位の情報を得られる。

注目する  
多重ループ

当該ループ  
の実行時間

当該ループ  
の実行回数

当該ループ  
の通信時間





# 地球シミュレータ共同プロジェクト

- 平成14年度より、地球シミュレータ共同プロジェクト「**並列処理言語HPF (High Performance Fortran)**を用いた大規模並列実行の性能検証および新規機能の検討」を実施中。
- メンバー

「HPF推進協議会」とも連携してプロジェクトを遂行。

岡部 寿男	京都大学
村井 均	地球シミュレータセンター
秦 万美子	地球シミュレータセンター
坂上 仁志	姫路工業大学
窪田 昌史	広島市立大学

# 目的

---

- HPFで並列化された実アプリケーションの評価
  - HPFコンパイラの諸機能の有効性の検証
  - HPFによるプログラミング手法の確立
- ➡ ✓ 並列計算を容易に利用できる手段を提供
- ✓ 計算科学の振興

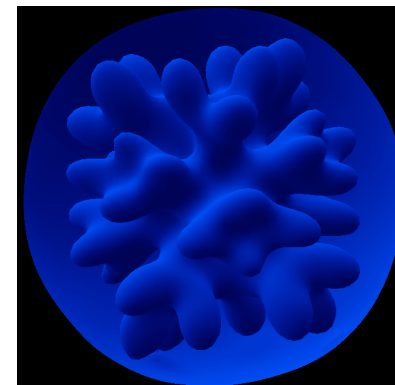
# 計画

---

- 1年目 (平成14年度) 終了
  - 実アプリケーションの並列化と評価
  - ベンチマーク評価
- 2年目 (平成15年度) 本年度
  - ベンチマーク評価(継続)
  - 実アプリケーションにおいて、様々なコーディング技法を評価
- 3年目 (平成16年度)
  - 拡張言語仕様の提案と、プリプロセッサによる実装と評価

# 昨年度までの成果

- IMPACT-3D (三次元プラズマシミュレーション)
  - 14.9 Tflops (512PN, ピーク比45%, フラット並列化)
  - SC2002 にて、Gordon Bell 賞「言語賞」を受賞<sup>[1]</sup>
- PFES (Princeton ocean model)
  - 10.5 Tflops (376PN, ピーク比43%, ハイブリッド並列化)



IMPACT-3Dの実行結果: レーザー核融合の爆縮過程における燃料・プッシャー接触面

[1] H. Sakagami, H. Murai, Y. Seo and M. Yokokawa. 14.9 TFLOPS Three-dimensional Fluid Simulation for Fusion Science with HPF on the Earth Simulator, *In proc. of SC2002*, Nov. 2002.

# 今年度の成果

---

- 実アプリケーションの評価
  - 全球大気シミュレーション
  - プラズマシミュレーション(IMPACT-3D)
- ベンチマーク評価
  - NAS Parallel Benchmark (NPB)
  - SPEC OpenMP
  - HPFBench benchmark suite
- 言語仕様の検討
- その他
  - マルチプラットフォーム上の評価
  - HPF/ES for PC Clusterを整備

# 実アプリケーションの評価

- 全球大気シミュレーション
  - オーストラリア気象局の全球モデル  
GAM (Global Atmospheric Model)
  - HPFでの記述と性能を、MPIと比較
  - 現在、作業中
- プラズマシミュレーション (IMPACT-3D)  
主に、並列入出力とSHIFT通信の評価

他のプロジェクトでは、非平衡超伝導ダイナミクスシミュレーションや宇宙環境シミュレーション、乱流シミュレーションなどで採用されている。

# NPBの評価

---

- NPB(BT, SP, LU, CG, MG )の、HPFによる最適な実装を求める。
- MPIによる実装(NPB2.4)と比較し、性能上の問題点と不足する機能・言語仕様の抽出を行う。

FT とEP はNPB3.0alphaで最適に近い(MPIと同等の)実装が提供されている。  
IS はFortranによる実装が提供されていないため保留。

# MGベンチマーク

---

## 「マルチグリッド法」

- ✓ HPF並列化のポイント
  - 再帰呼び出しによる階層並列化
  - 階層データの「自動」分散

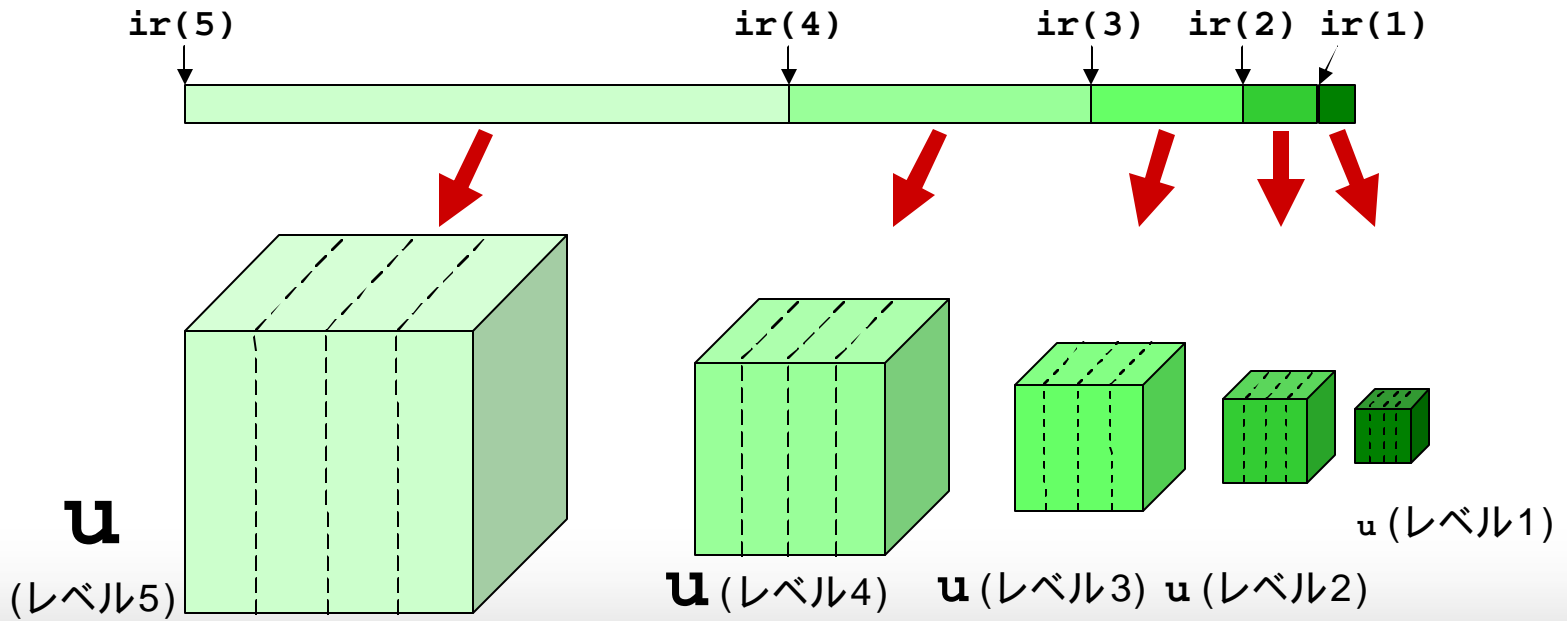


# NPB2.4における階層並列化

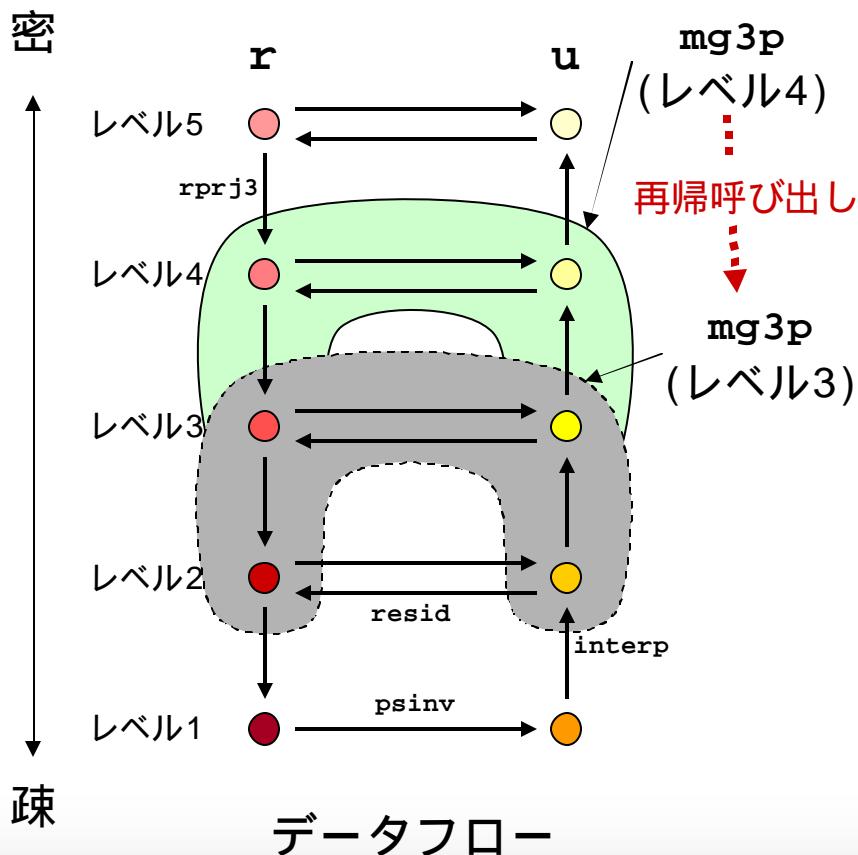
```
call psinv(u(ir(k)),...)
...
subroutine psinv(u,...)
real u(n1,n2,n3)
...
```

一次元配列の要素を渡し、三次元配列として受け取る。

➡ HPFでは分散できない。



# 再帰呼び出しによる階層並列化



```

recursive subroutine mg3P(u,r,k)
...
call rprj3(r,r2,k)

if (k > 2) then
    call mg3p(u2,r2,k-1)
else
    call psinv(r,u,1)
end if

call interp(u2,u,k)
call resid(u,r,k)
call psinv(r,u,k)

end
    
```

mg3Pは、kとk-1の2つのレベルだけを扱う。

# 階層データの「自動」分散

```

recursive subroutine mg3P(u,r,k)

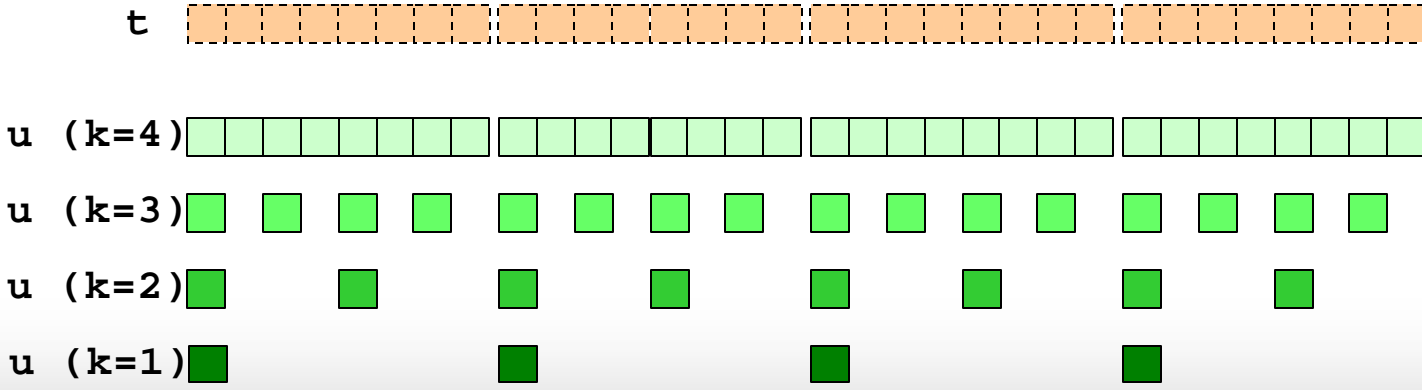
!HPF$ template t(N)
!HPF$ distribute (block) :: t

double precision u(:, :, :,)
!HPF$ align (*, *, i) with *t(i*(2**(LT-k))-m) :: u

double precision, allocatable :: u2(:, :, :,)
!HPF$ align (*, *, i) with t(i*(2**(LT-k+1))-m2) :: u2

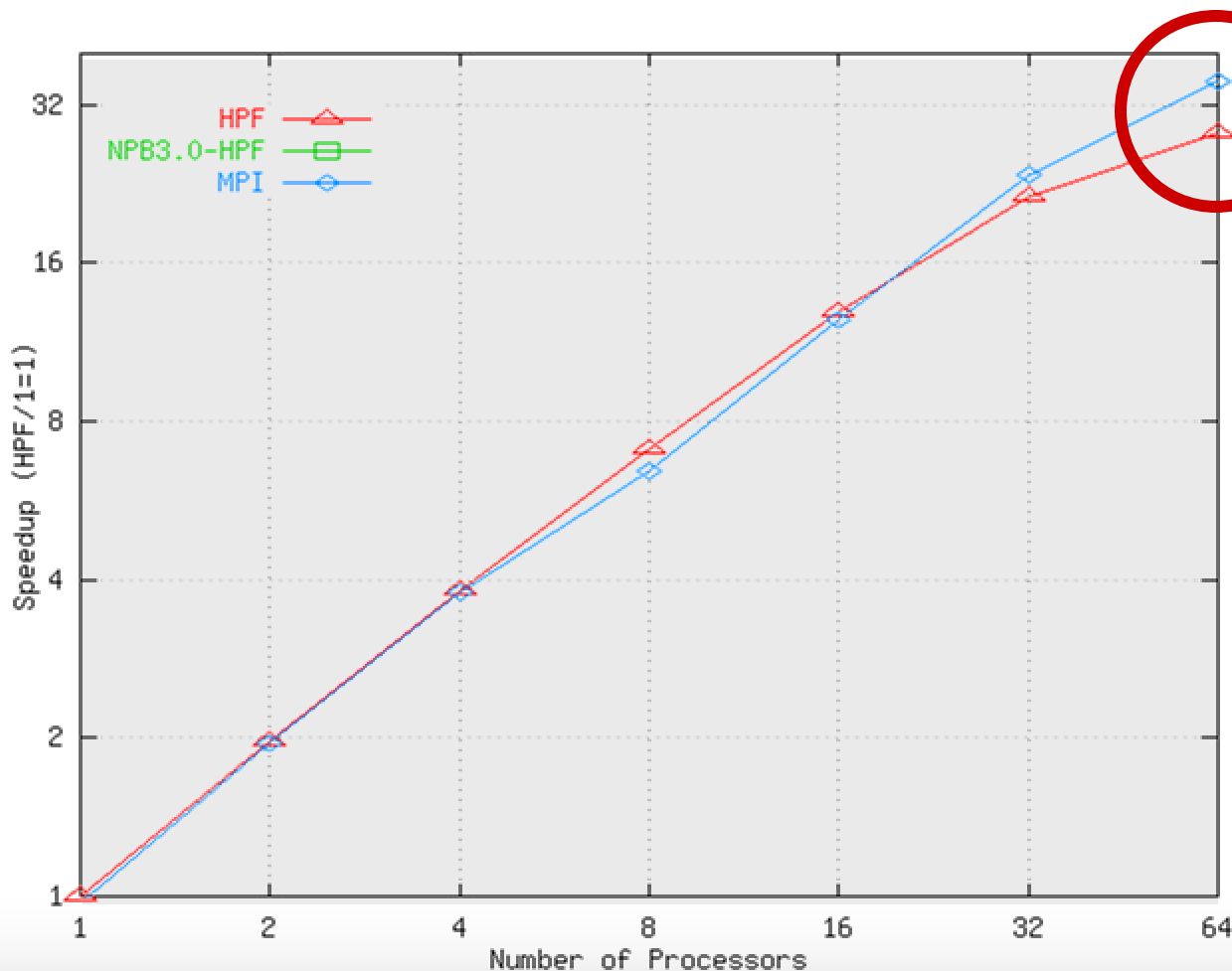
```

- mg3pは、レベルkの配列を引数として受け取り、レベルk-1の配列を割り付ける。
- レベルkの配列は、ストライド  $2^{LT-k}$  で整列する。



正しい整列関係

# 評価 (MG)

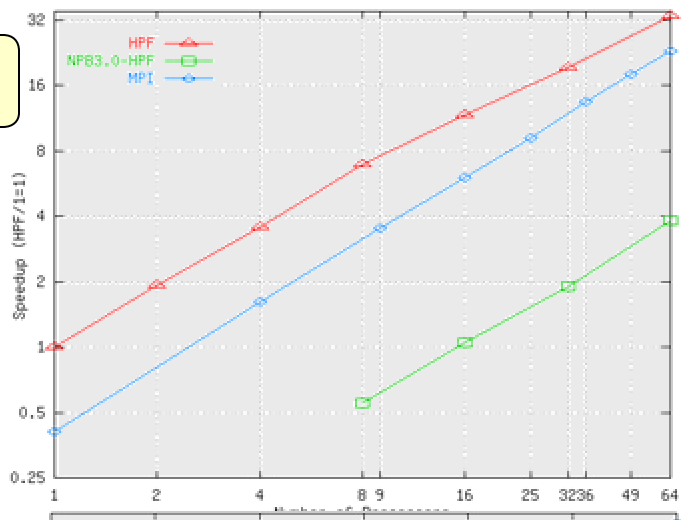


- MPI版と同等の性能。
- 高並列時に効率がやや低下するのは、グリッドが疎になったときの例外的な処理による。

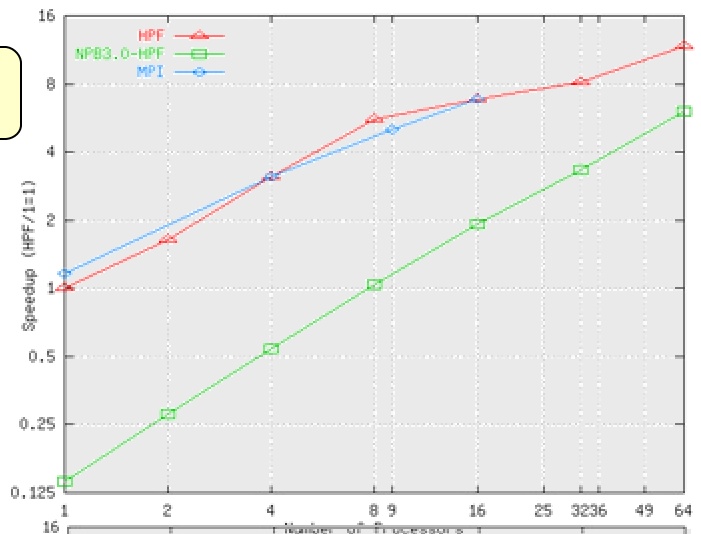
ESのL系バッチ  
ジョブ, フラット並  
列化による評価

# 評価 (BT, SP, LU, CG)

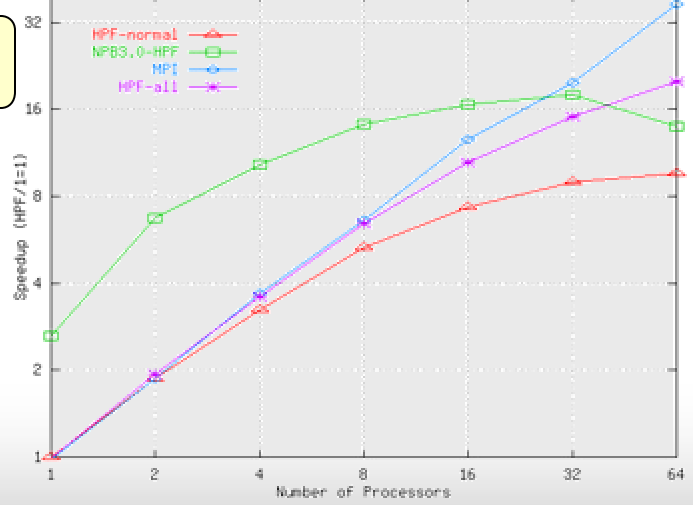
**BT**



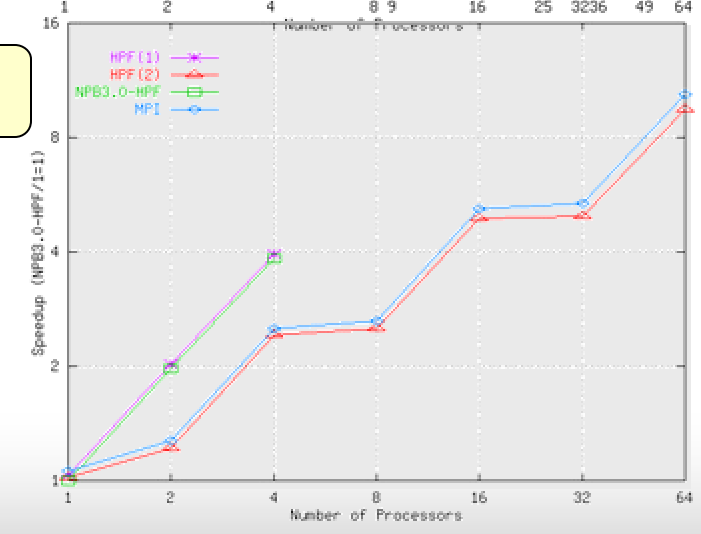
**SP**



**LU**



**CG**



# NPBの評価(まとめ)

---

- HPF/ESによって、NPBを実装した。
  - MPI版と同じ並列化
  - ベクトル化のためのチューニング
- ESの1～8ノードで評価を行った。
  - BT, SP, CG, MGではMPI版と同等以上、LUではほぼ同等の性能を達成した。
- HPFPCのホームページでコードを公開中。

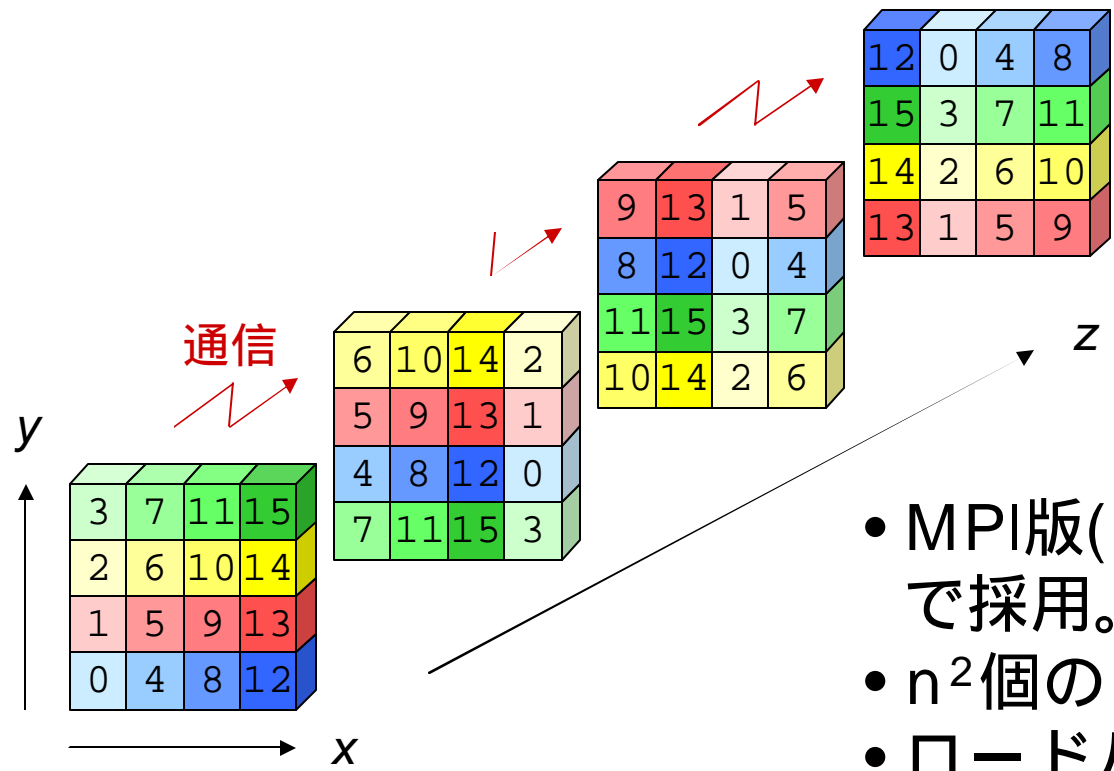
# 言語仕様

---

以下の機能または言語仕様が必要であることがわかった。

- シャドウ部の重複実行(EXT\_HOME)
- 部分的REFLECT
- GEN\_BLOCKと二次元分散の性能向上
- マルチパーティション
- パイプライン並列化
- ブロック分散の拡張
- 周期境界条件
- 標準計時関数

# マルチパーティション



捻りブロック分散  
(twisted-block distribution)とも呼ばれる。

- MPI版(NPB2.4)のBTとSPで採用。
- $n^2$ 個のプロセッサ。
- ロードバランスに優れる。
- 通信の効率が良い。



# PCクラスタ用HPFコンパイラ

---

ESCは、PCクラスタ上で動作するHPFコンパイラの提供を行っている。

## ◆HPF/ES for PC Cluster

- HPF/ES互換
- プラットフォーム限定(NEC製ハードウェアのみ)
- 希望者に配布可

## ◆fhpf

- HPF2.0仕様と公認拡張の一部をサポート
- プラットフォームの制限なし
- 開発中
- 詳細は、この後の岩下氏の発表にて。

# 評価(1)

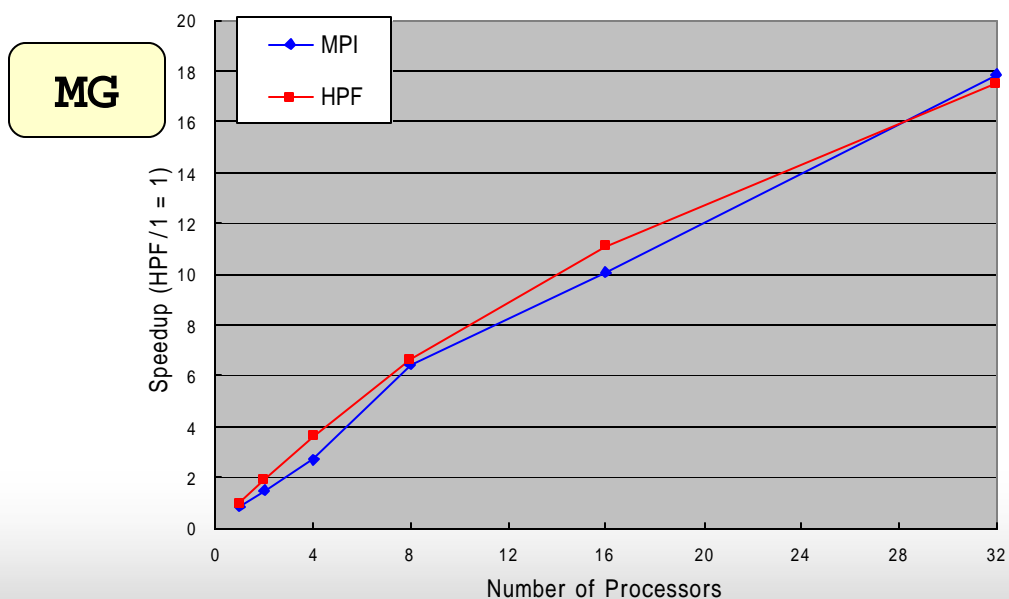
ESCのPCクラスタ「Express5800 Parallel PC-Cluster」上で評価を行った。

## 評価環境

- コントロールノード
  - CPU : Pentium III 1.2GHz × 2
  - メモリ : 1GB
  - 磁気ディスク : 72GB × 6 (RAID-5)
  - OS : RedHat Linux 7.2 (kernel 2.4.7)
  - ソフトウェア : sCore Ver.5.0.1
- ネットワーク
  - Myrinet
- ソフトウェア
  - HPF/ES for PC Cluster Rev.3.0(843)
  - Intel Fortran Compiler
  - MPICH
- 処理ノード × 64
  - CPU : Pentium III 1GHz
  - メモリ : 1GB
  - OS : RedHat Linux 7.2 (kernel 2.4.7)
  - ソフトウェア : sCore Ver.5.0.1

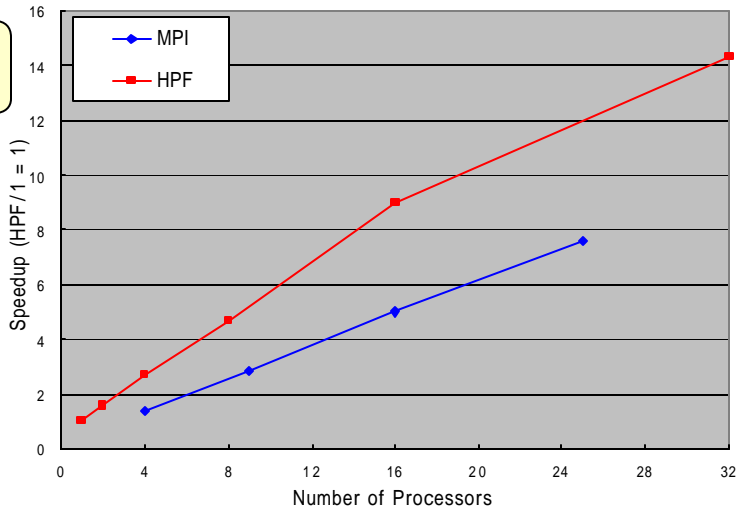
# 評価 (2)

- 基本的にES版と同じコードを使用。
- ただし、ベクトル実行を強く意識した(スカラプロセッサに極端に向かない)部分は、NPB3.0-serialのコードを利用した。

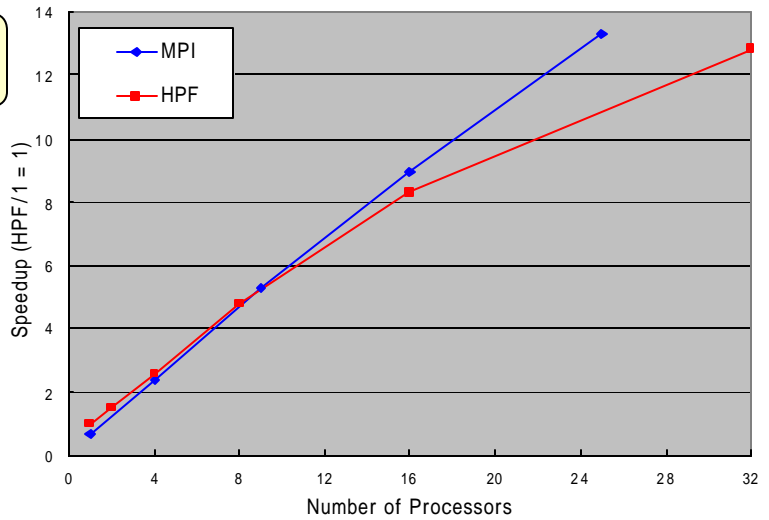


# 評価 (3)

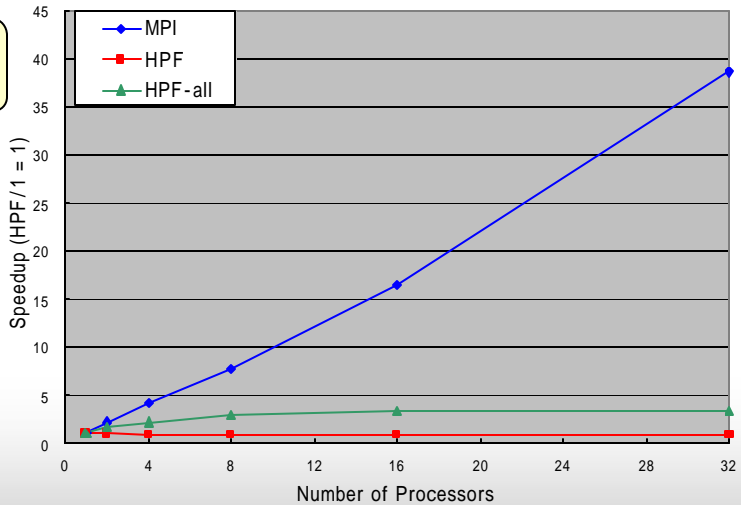
**BT**



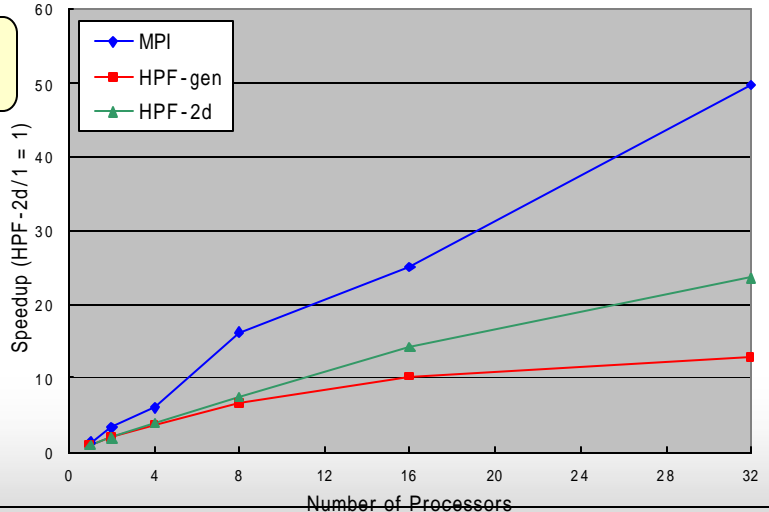
**SP**



**LU**



**CG**



# 考察 (1)

	結果	理由
MG	HPF = MPI	
BT	HPF > MPI	MPI版の「マルチパーティション」は、CPU数が少ない場合には効果が小さい？
SP	HPF < MPI	
LU	HPF < MPI	パイプライン処理を擬似的に実現するために頻発する隣接通信のオーバーヘッドが大きい。
CG	HPF < MPI	転置や総和の処理が重い？

## 考察(2)

- ESに比べるとネットワークやMPIライブラリの性能が劣る。
  - 通信が支配的である場合に性能差が顕著になる (LU)。
- 通信ランタイムが、ベクトル機であるES向けにチューニングされている。
  - 特に隣接通信以外でオーバヘッドが現れる？ (CG)

# まとめ

---

- 地球シミュレータのシステムと共同プロジェクトの成果について概説した。
- HPF/ESの機能を説明した。
- ES共同プロジェクト「並列処理言語HPF ~」の成果について述べた。

# 今後の課題

---

- ワークショップの開催や成果の発表, ユーザのサポートを通じてHPFの普及推進を図る。
- ES共同プロジェクト
  - 実アプリケーションの評価(継続)
  - PCクラスタ上での評価, 比較・検討
  - 拡張言語仕様および新規機能の提案
  - 提案した言語仕様と機能を実現するプリプロセッサ(?)を開発し、評価する。