http://www.cradle.co.jp/

パネル討論

実用アプリの加速のために何が必要か

PCクラスタコンソーシアムに何ができるか

第十一回PCクラスタシンポジウム

2011年12月09日

秋葉原コンベンションホール

株式会社ソフトウェアクレイドル

技術部

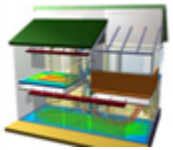
黒石浩之

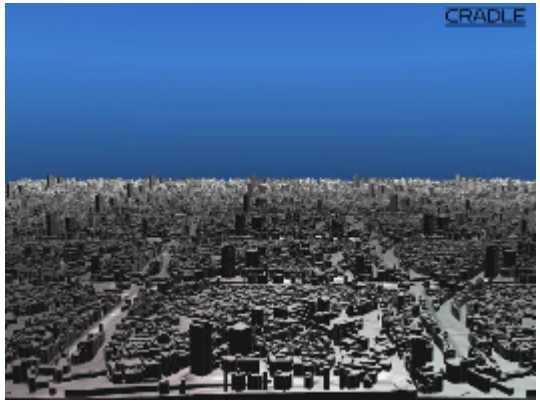
CRADLE

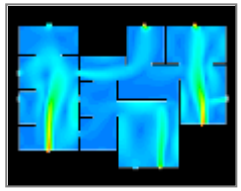
クレイドル

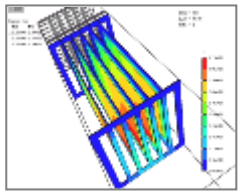
構造格子系

STREAM



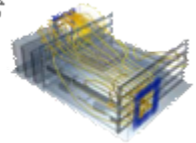


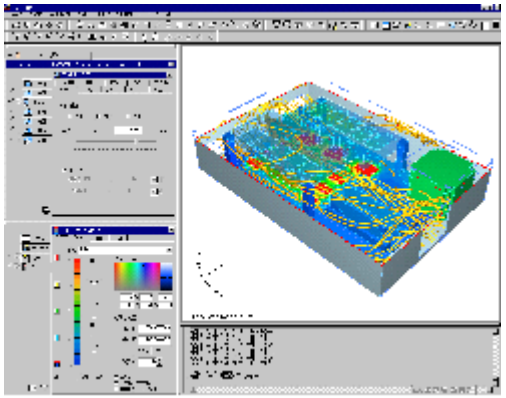


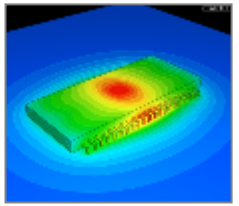


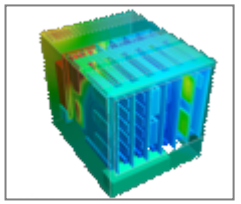
電子機器筐体の熱設計専用パッケージ

熱設計PAC



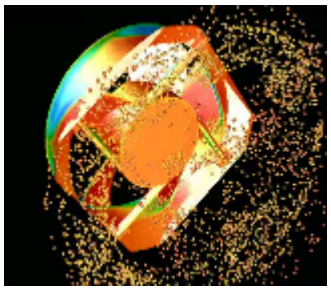


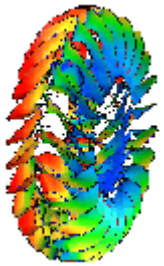


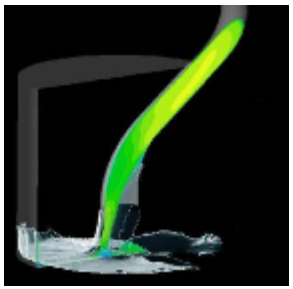


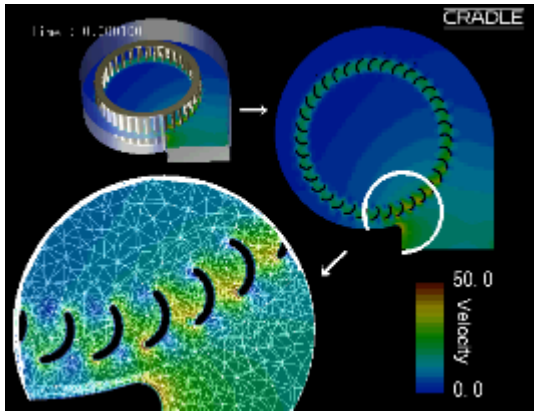
非構造格子系

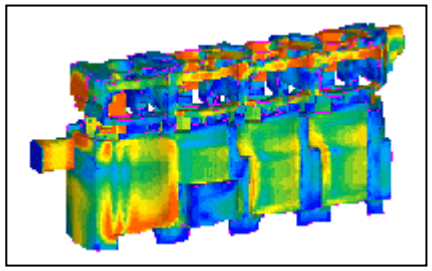
SCRYU/Tetra

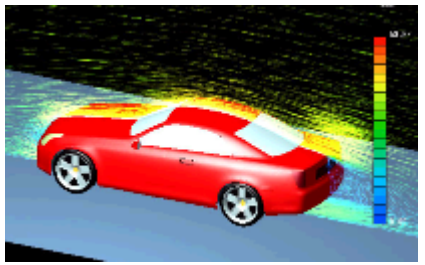




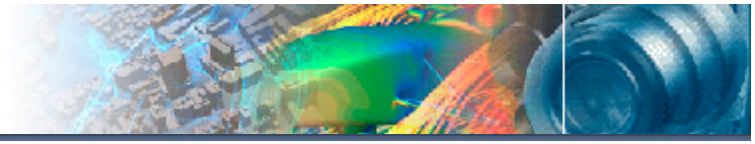








目次



- 並列性能を向上するための取り組みの状況
- 並列性能向上の上での問題点

現状

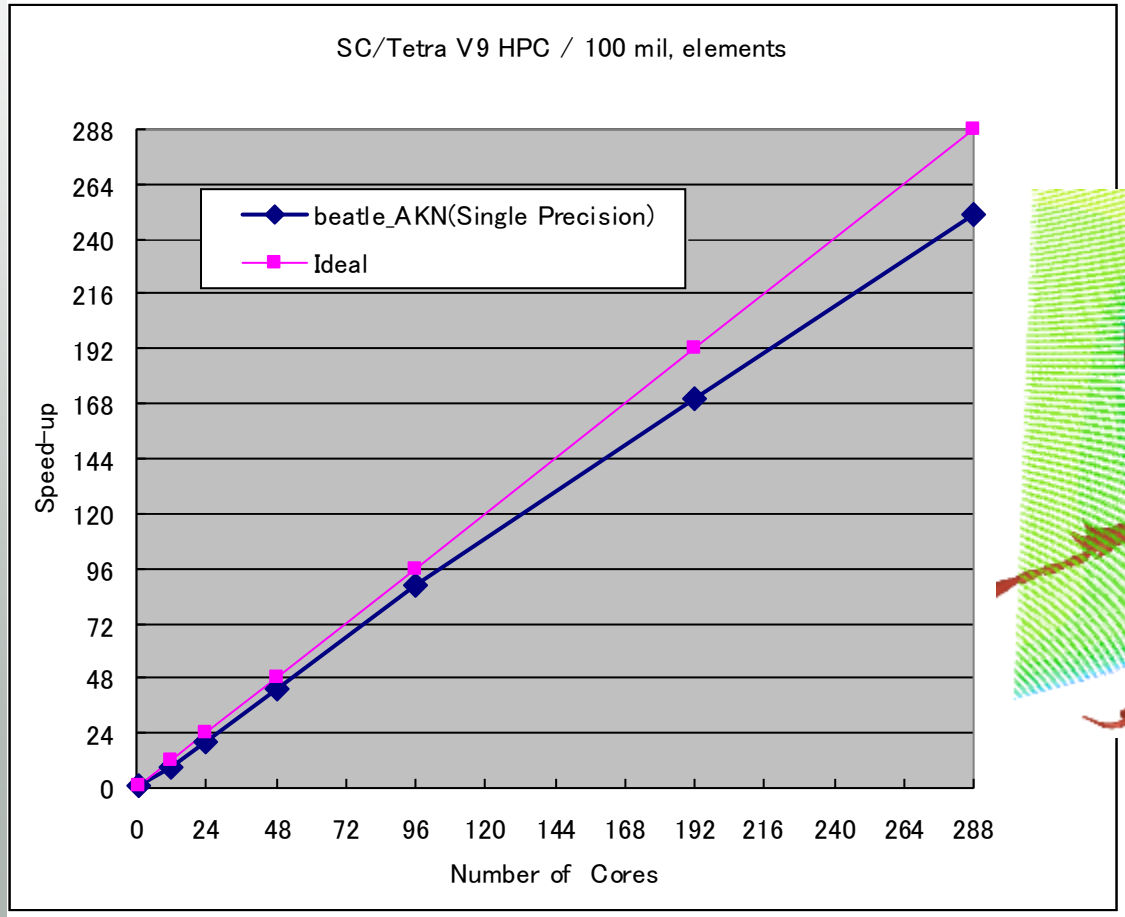
SCRYU/Tetra V9

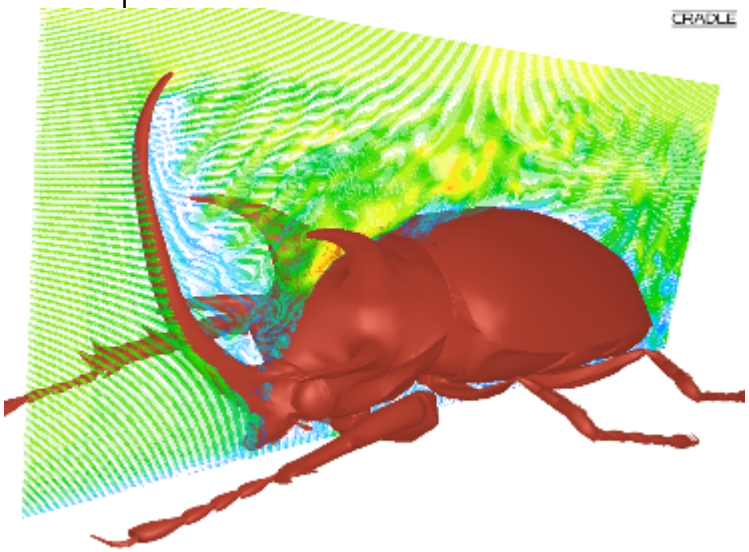
約1億要素

Intel®Xeon®X5680

(3.33GHz,12MBキャッシュ、6core) x2

Windows HPC Sever 2008 R2, Infiniband/QDR





現状

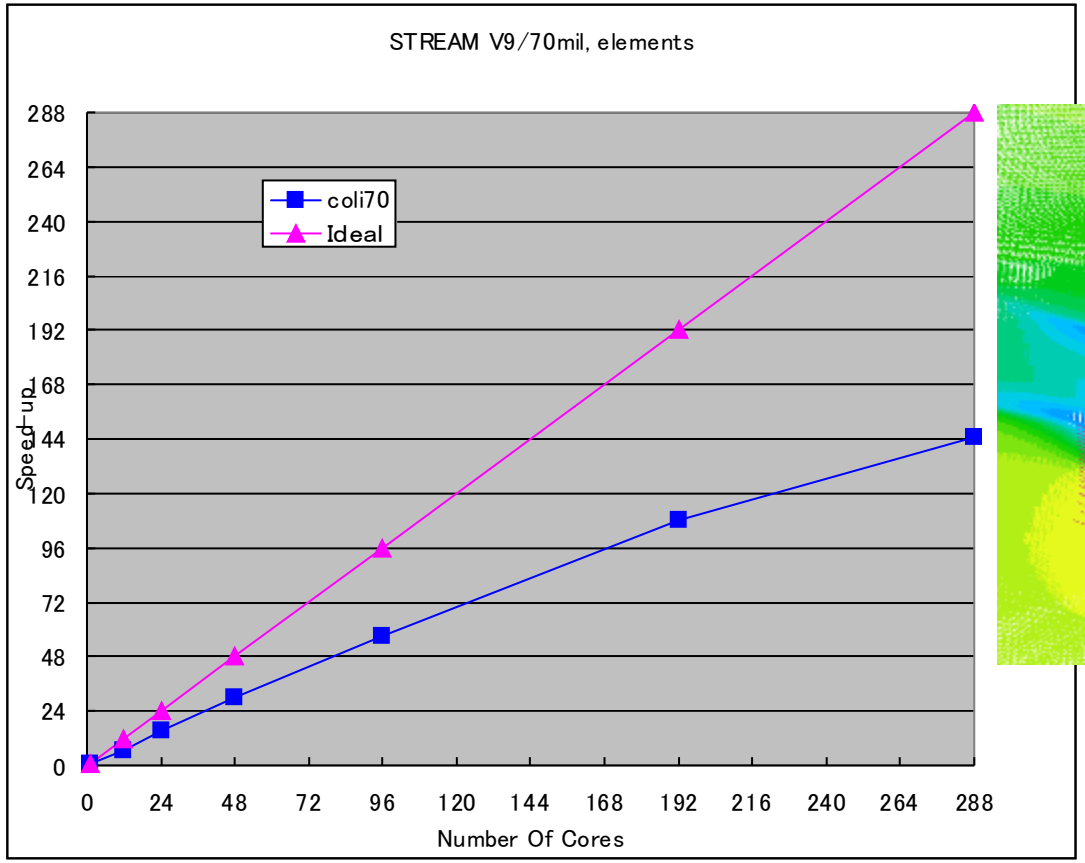
STREAM V9

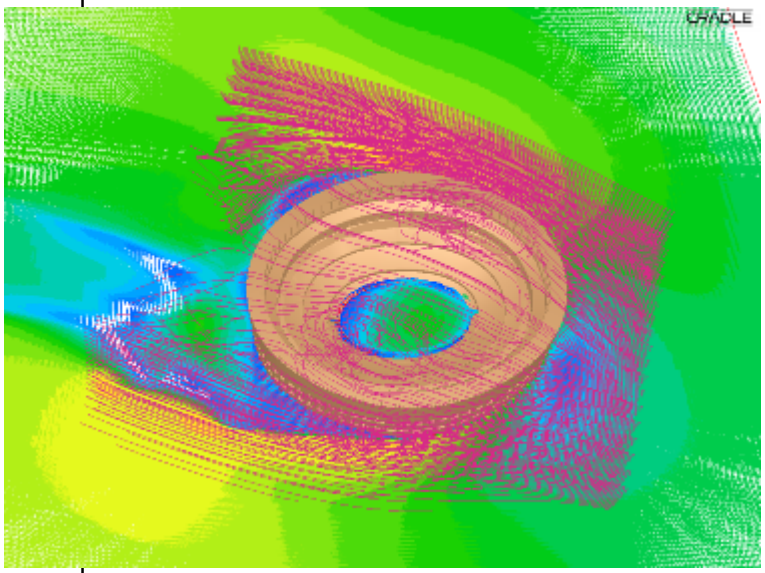
約7000万要素

Intel®Xeon®X5680

(3.33GHz,12MBキャッシュ、6core) x2

Windows HPC Sever 2008 R2, Infiniband/QDR

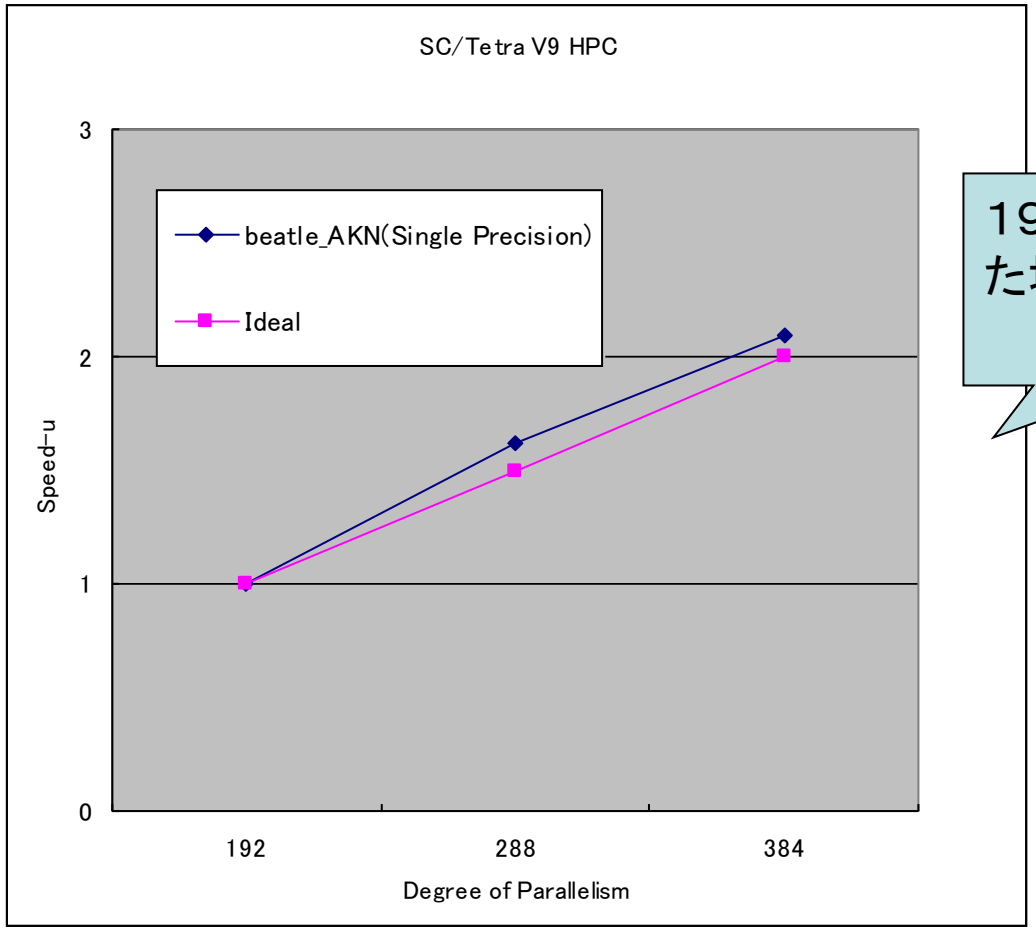




現状

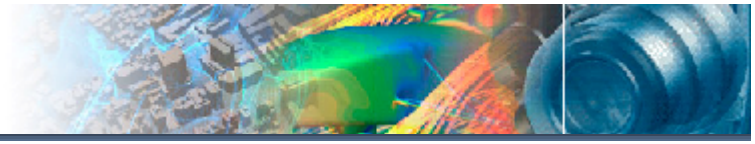
SCRYU/Tetra V9　　FOCUSスパコン

FOCUS　：　財団法人計算科学振興財団



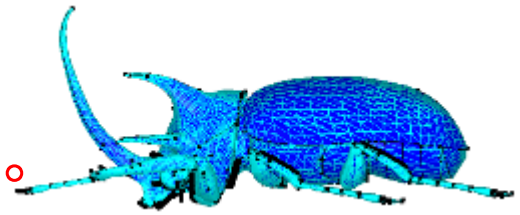
192並列を1とした場合の並列効率

目次



- 並列性能を向上するための取り組みの状況
- 並列性能向上の上での問題点

取り組み

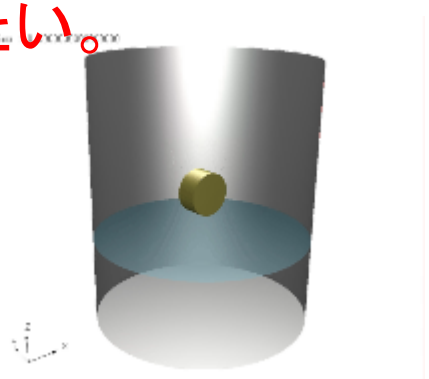


・解析ニーズ

・実験に代替可能な高精度な結果がほしい。
現象・原因の解明

・モデル形状を詳細（設計形状）に表現したい。

・高度な物理現象を解析したい。



・モデルの大規模化

・計算速度の向上：並列性能の向上

・ではなくて、如何に答えを速く出すか、すなわち
データの入力から結果の確認までを以下に速くするか。
その一部として、「並列性能の向上」がある。

取り組み

前処理(メッシュ生成)

SCTpre : 体積メッシュ生成(MPI並列)

表面メッシュ生成(スレッド並列)

STpre : フラクション計算(要素分割)(スレッド並列)

ソルバー

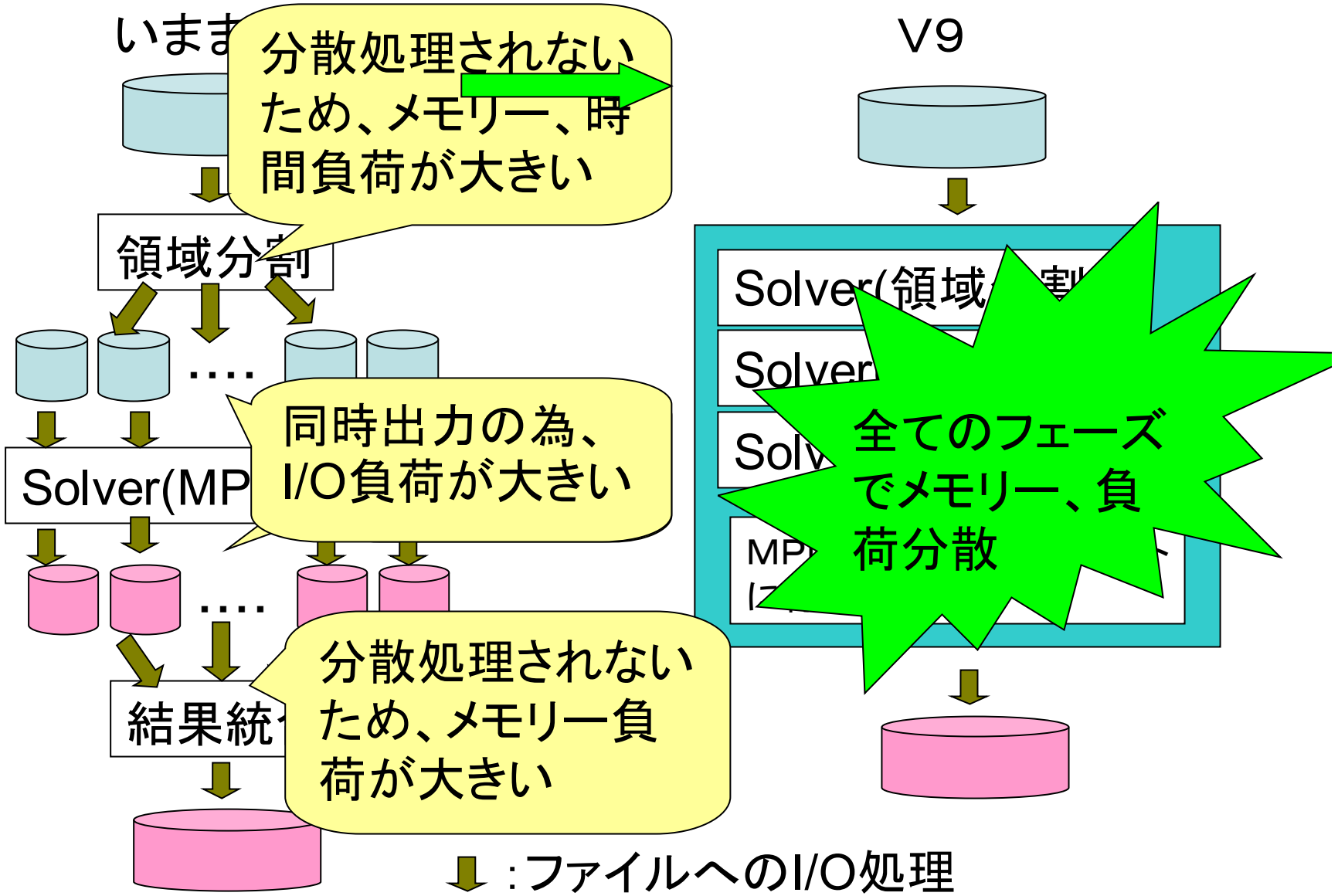
SCTsolver : MPI並列

STsolver : MPI並列

後処理(可視化)

SCT/STpost : MPI並列＋スレッド並列

取り組み



取り組み（ParMETIS)

領域分割のメモリー分散化のため、ParMETISプロジェクトの推進を図った。

ParMETIS/METIS：http://glaros.dtc.umn.edu/gkhome/

Minnesota大学　Professor George Karypisによるグラフ理論に基づく分割プログラム。

・2007年（ParMETIS),2008年(METIS)以来、開発が止まっていた。

・LINUX/Windows環境でのフル64bit対応をベースとして、有償対応をお願いし、2011年5月に正式リリースされた。

・上記WEBサイトでDL可能。但し、商用利用するにはMinnesota大学との契約及び使用料が必要です。

取り組み（出力）

全てのプロセスが同時に結果出力すると多大な負荷となる。
この負荷を軽減するために、3つの出力モードを用意。

大規模並列ファイルシステム向け

1．MPI並列I/O関数により、全てのプロセスが1つのファイルイメージへ出力。

中・小並列ファイルシステム及びベクター型RAIDシステム向け

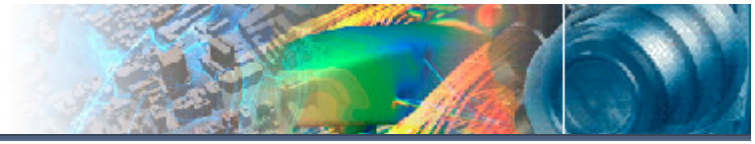
2．MPI通信にて収集し、MPI並列I/O関数により、1つのプロセスがファイルイへ出力。

ベクター型RAIDシステム及び標準環境向け

3．MPI通信にて収集し、標準関数により、1つのプロセスがファイルへ出力。

ハードウェアにあわせたモードを選択することで、効率良いI/O処理を行うことが可能になっている。

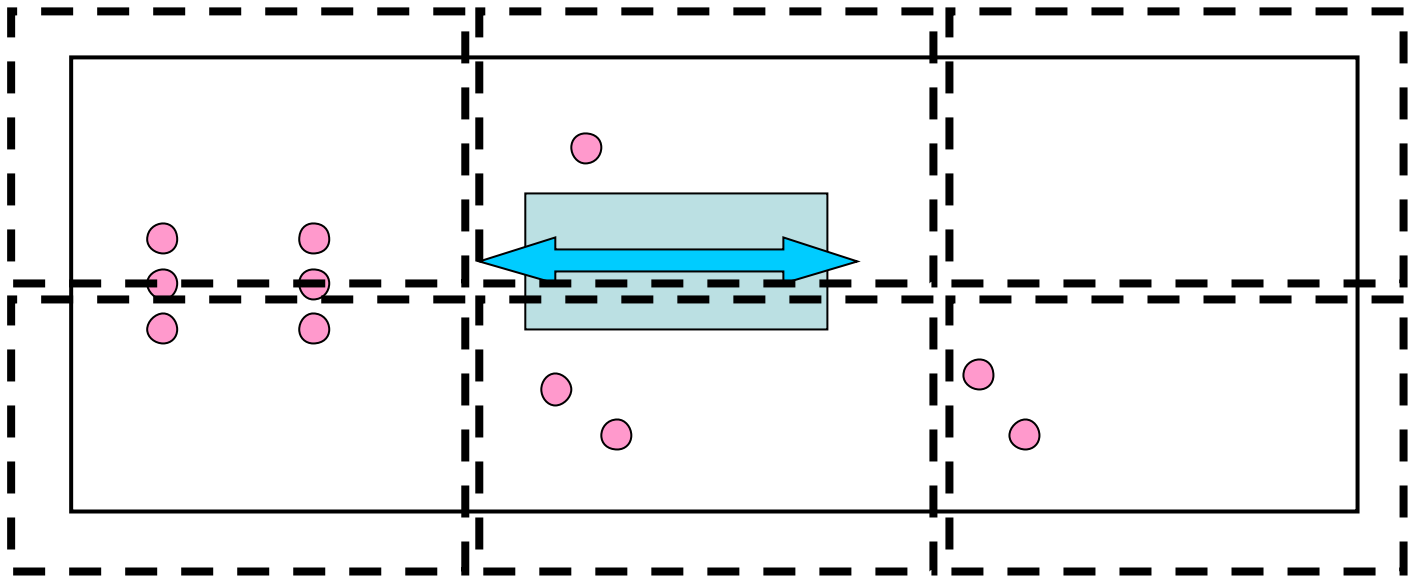
目次



- 並列性能を向上するための取り組みの状況
- **並列性能向上の上での問題点**

並列性能向上の上での問題点

- 現状（現行手法）
  - 機能による並列効率のばらつき



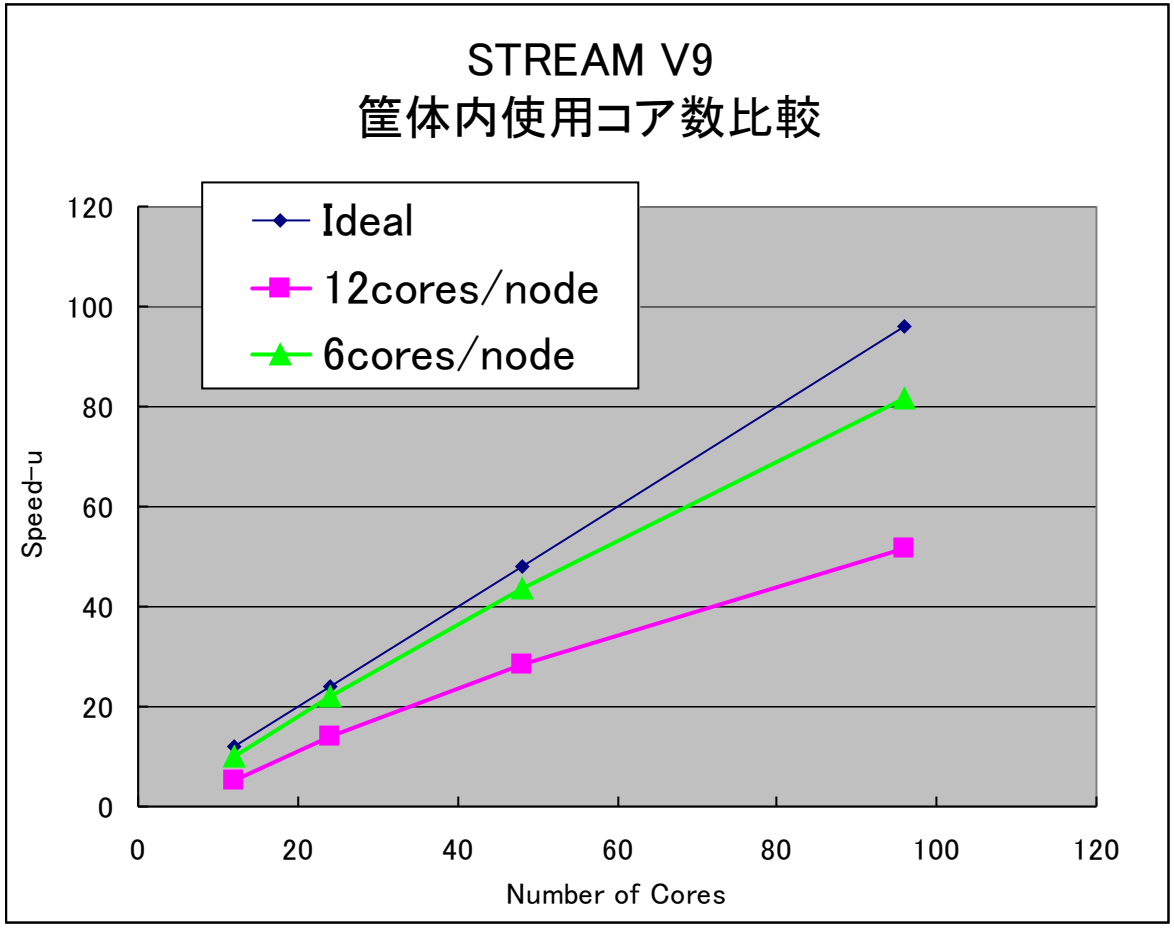
粒子追跡や移動境界など計算負荷バランスの不均一性

  - 集合通信のコストが高い
  - メモリーのレイテンシー

レイテンシー

Intel®Xeon®X5680(3.33GHz,12MBキャッシュ、6core) x2

にて、ノード当り6コア使用した場合との比較



並列性能向上の上での問題点

今後の方向？

MPI+MPI? MPI+OpenMP? MPI+GPGPU? …

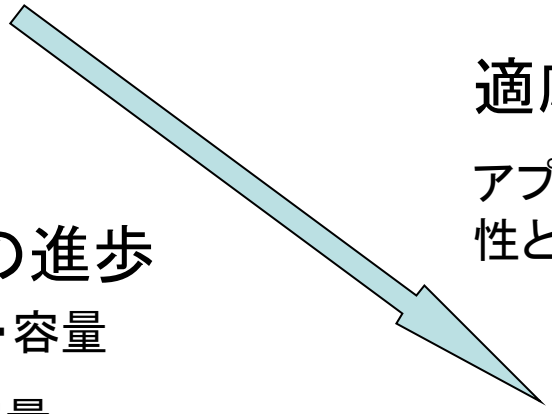
現在の資産

（スケーラビリティ）

と機能開発

開発時間

プログラマーの質



適応性

アプリケーション・機能の特性とのマッチング

ハードウェアの進歩

メモリー速度・容量

Disk速度・容量

ネットワーク速度

CPU/GPUの方向性

ターゲット

10億要素？1万コア？

ご清聴ありがとうございました。