シームレス高生産・高性能プログラミング環境

東京大学情報基盤センター
石川裕

T2K Open Supercomputer Alliance

目次

全体背景

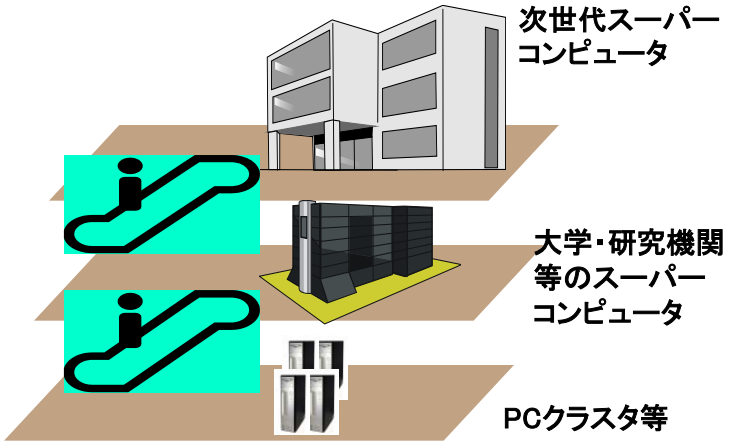
- 科学技術・学術研究における萌芽的研究は、研究室レベルで進められているが、利用できるコンピュータの能力程度の問題規模を解くに留まることが多い。

  理由：

  – 研究室レベルで使用されているコンピュータ上で開発されたアプリケーションを計算センターに設置されているスパコン上に持っていっても動かないことがある。

  – また動いたとしても大規模問題を解くことができないことが多い

- 大規模問題を解くプログラムは、並列処理に関する深い知識に加えて、スパコンの性能を引き出す豊富なノウハウを持つ、限られたプログラマにしか作れない



並列アプリケーション生産性拡大のための道具
- **並列言語処理系**
- **並列スクリプト言語**
- **ライブラリ＆チューニングツール**

情報系研究者 •ベンダ 応用系研究者 工学・科学

産学連携コンソーシアムによる
教育・普及、規格化、ユーザ支援
PCクラスタコンソーシアム

計算物理、計算化学、
生物情報、検索、マルチメディア

高性能並列プログラミング言語 (1/2)

- 現状と課題
  - 並列プログラムの大半はMPI通信ライブラリによるプログラミング

    生産性が悪く、並列化のためのコストが高い
  - 並列プログラミング教育のための簡便で標準的な言語がない
  - 研究室のPCクラスタから、センター、ペタコンまでに到るスケーラブルかつポータブルな並列プログラミング言語が求められている
- 目標
  - 大規模並列システム上でのプログラミングを助け、生産性を向上させる並列プログラミング言語の設計・開発
  - 既存プログラミング言語(Fortran, C)をコンパイラ指示文(コメント行に記述)により拡張
  - ユーザ普及および標準化のためのコミュニティ形成

BEFORE (現状)

```
int array[YMAX][XMAX];

main(int argc, char**argv){
 int i,j,res,temp_res, dx,llimit,ulimit,size,rank;

 MPI_Init(argc, argv);
 MPI_Comm_rank(MPI_COMM_WORLD, &rank);
 MPI_Comm_size(MPI_COMM_WORLD, &size);
 dx = YMAX/size;
 llimit = rank * dx;
 if(rank != (size - 1)) ulimit = llimit + dx;
 else ulimit = YMAX;

 temp_res = 0;
 for(i = llimit; i < ulimit; i++)
  for(j = 0; j < 10; j++){
   array[i][j] = func(i, j);
   temp_res += array[i][j];
  }

 MPI_Allreduce(&temp_res, &res, 1, MPI_INT, MPI_SUM, MPI_COMM_WORLD);
 MPI_Finalize();
}
```

MPIしか、使えるものがない
MPIの並列プログラムはむずかしい… いっぱい書き換えないといけないし、時間がかかる、デバックもむずかしいし、…

AFTER (3年後)

```
int array[YMAX][XMAX];

#pragma ompd distvar(var = array;dim =

main(){
 int i, j, res;
 res = 0;

#pragma ompd for affinity(array) reduction(+:res)
 for(i = 0; i < 10; i++)
  for(j = 0; j < 10; j++){
   array[i][j] = func(i, j);
   res += array[i][j];
  }
}
```

data distribu

add to the serial coc
incremental parallelize

work sharing and data synchronization

いまのプログラムに指示文を加えるだけだから、簡単！性能チューニングも可能、…
どこでも使えるから安心、並列プログラミングも習得にもお勧め！

【注】これは、現在検討中の例で最終的な仕様ではありません。

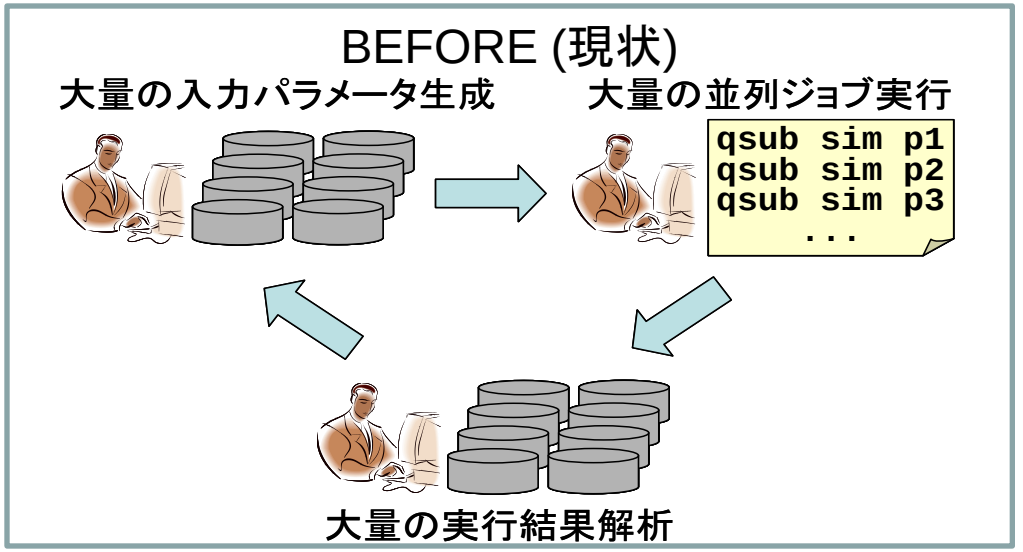
高性能並列プログラミング言語 (2/2)

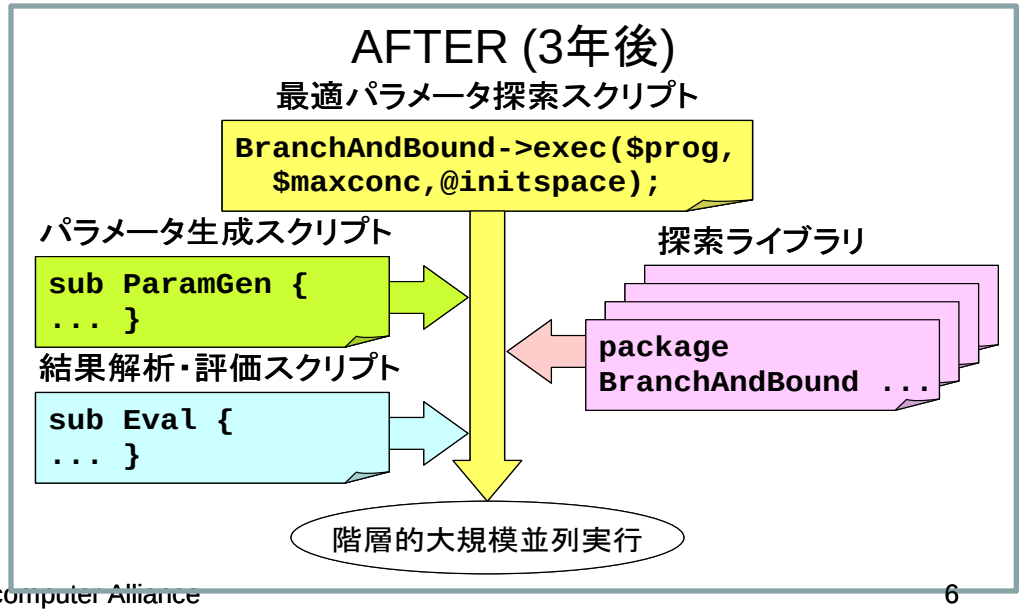
- 研究項目
  - 言語仕様の設計・決定
    - 実施責任者佐藤既存研究成果であるOmni OpenMP処理系およびOpenMD言語、HPCベンダが規格化したHPF言語の経験を反映
  - HPCベンダのコンパイラに接続するための中間コードの仕様を検討
    - **国内HPCベンダNEC(ベクトル並列計算機メーカ)、富士通＆日立(スカラ並列計算機メーカ)と協調**
  - ベース言語として採用するFortran言語とC言語コンパイラの開発
  - 国際展開・標準化のためのコミュニティ作り
- 期待される成果
  - 並列プログラムの生産性向上と、これに大規模並列システムが活用を促進、計算科学の発展に寄与
  - 並列プログラミングの人材育成に寄与
    - 普及・標準化を行うことにより、MPIよりも高度で簡便な並列プログラミングの手段として教育に用いることができる

高生産スクリプト言語 (1/2)

- 現状と課題
  - 車体の形状や創薬など多くの応用分野では、多数の設計パラメータから最適解を検索する作業が生じる。このために並列アプリケーションプログラムを大量に実行し、その結果を解析する。完全自動化されず、手作業が生じる
  - 最適パラメータ探索記述言語の欠如
    1. パラメータセットの生成
    2. 各パラメータを入力とするアプリケーションプログラムの並列実行
    3. 結果の集約・解析

    の繰り返しを記述する簡便な処理系が存在しない
- 目標
  - 最適パラメータ探索など粗粒度の大規模な階層的並列処理を、簡便かつ柔軟に記述可能で処理効率に優れたスクリプト言語とその処理系の開発



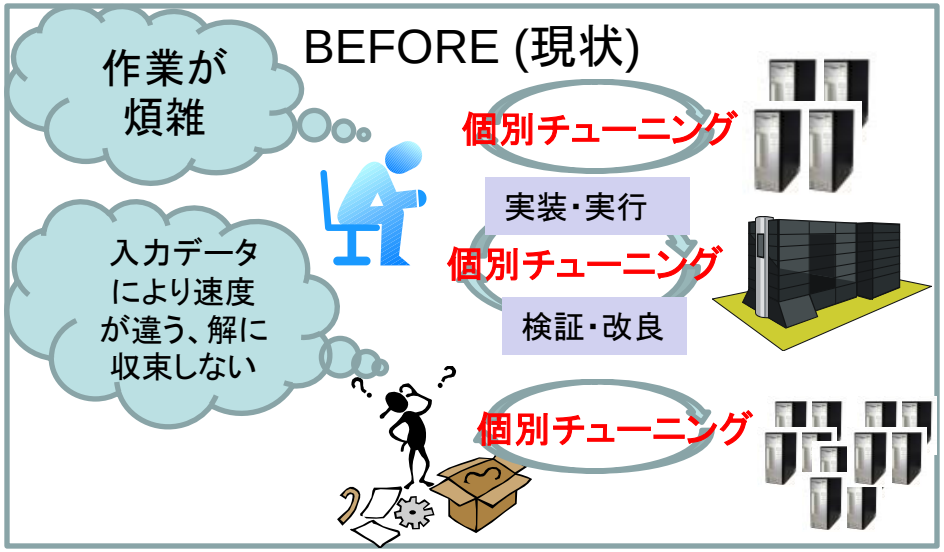


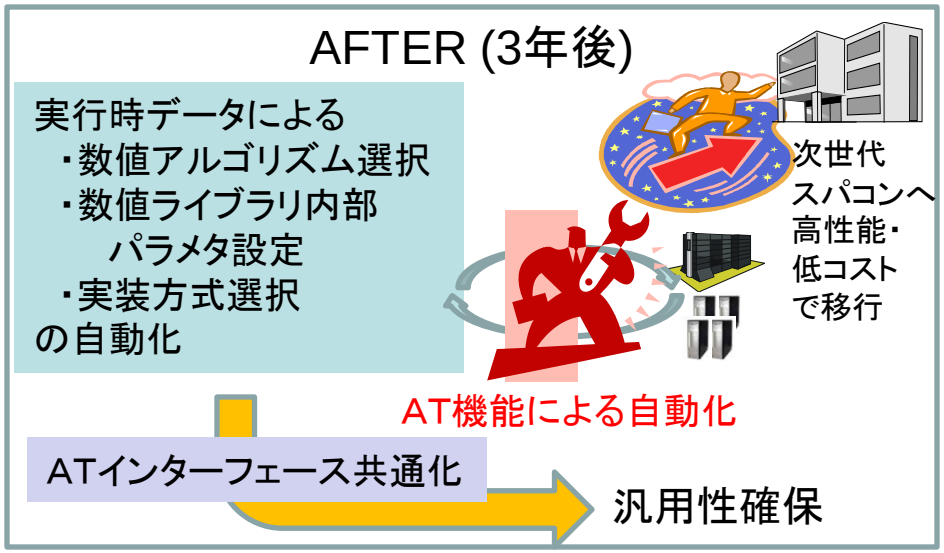
高生産スクリプト言語 (2/2)

- 研究項目
  - 既存スクリプト言語の選定、仕様設計
    - 実施責任者（中島）の既存研究成果であるMegascriptおよび**富士通**が有する並列計算向けスクリプト言語PJOを基礎
    - バッチジョブに投入するアプリケーションプログラムの実行制御およびインタフェースの抽象化
    - スクリプト単体のデバッグ機能
  - 高度な探索アルゴリズムのライブラリ化
- 期待される成果
  - 複雑な探索パラメータセットの生成や実行結果の解析に基づく探索アルゴリズムが容易にプログラム可能
  - 探索アルゴリズムのライブラリ化により、最適パラメータ探索などの生産性が向上

高性能高可搬性ライブラリ[自動チューニング付き数値計算　ライブラリ] (1/2)

- 現状と課題
  - 数値計算プログラムの性能チューニングコストの増大
    - PCクラスタから計算センターに設置されているスパコンなど、様々なアーキテクチャ毎にチューニングしなければならない
  - 実行時に与えられるデータセットの内容により、最適な数値計算アルゴリズムや実装方式が異なる
- 目標
  - 自動チューニング（AT）機構を含む数値計算ライブラリを開発し、数値計算コード部分のチューニングに関わる開発時間をなくす





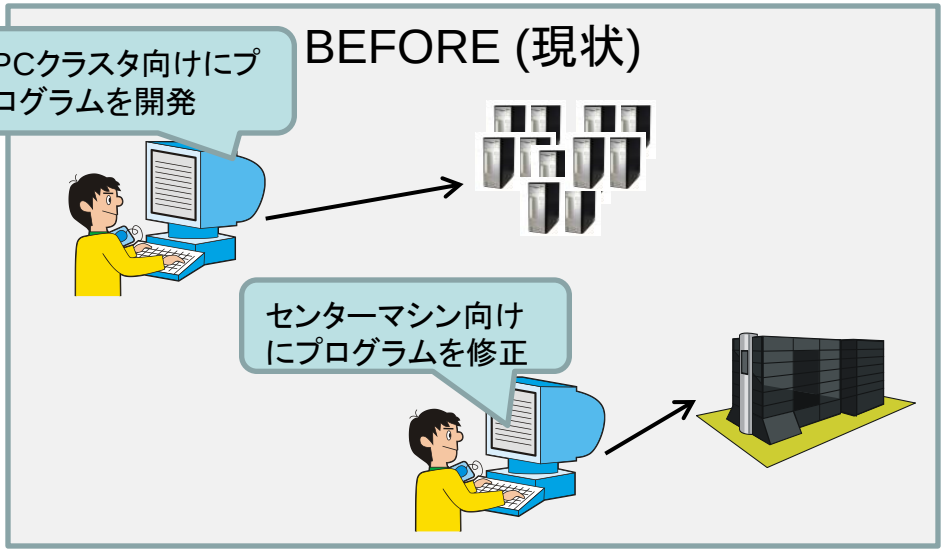
高性能高可搬性ライブラリ[自動チューニング付き数値計算　ライブラリ] (2/2)

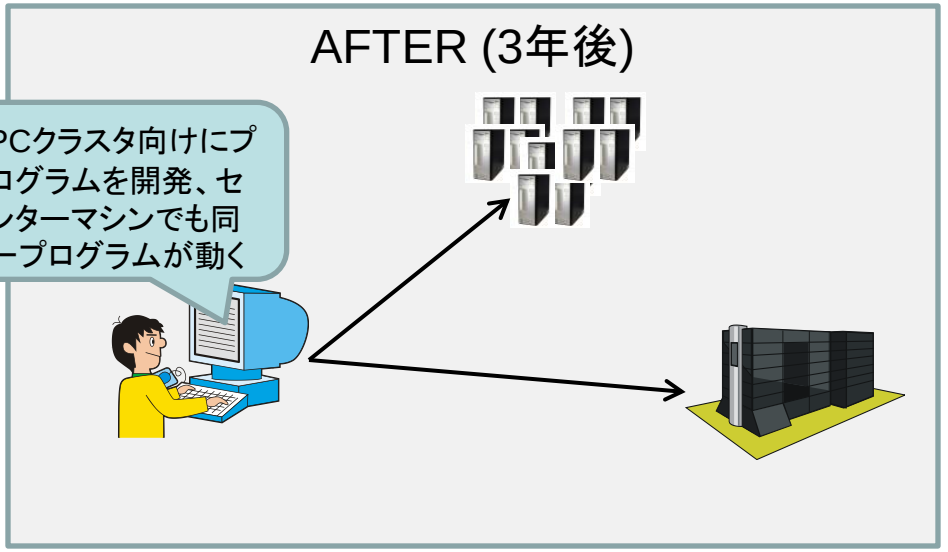
- 研究項目
  - 実施責任者の片桐の既存研究成果である自動チューニング(AT)機能付き数値計算ライブラリABCLibを大規模計算シミュレーションプログラムに適用・評価
  - AT機能のAPI化と軽量化、マルチコア向けAT方式
    - 日立中央研究所と共同研究
  - 対象プログラムは、固有値解析、反復解法の数値計算ライブラリ
- 期待される成果
  - AT機能をAPI化することにより、他の数値解法ライブラリにAT機能を追加しやすくなる
  - 開発する数値計算ライブラリを用いることで、PCクラスタからセンターマシンに至る広範な計算機環境においても、コード修正なしで高い性能を達成
  - 実行時に与えられるデータの変化に追従する自動チューニング機能により、従来の数値計算ライブラリでは達成できなかった高速化が実現

高性能高可搬性ライブラリ [Single Runtime Environment] (1/2)

- 現状と課題
  - PCクラスタとスパコン環境の違いにより、環境毎にプログラムの変更が必要
    - PCクラスタ：ローカルディスクが使える

      スパコン：ローカルディスクの扱いはセンターポリシに依存
    - MPI通信ライブラリのIO機能可搬性問題
  - 1万CPU規模向けファイルシステム環境の欠如
- 目標
  - PCクラスタでも基盤センタースパコン(1万規模CPU)でも単一実行時環境を提供するSingle Runtime Environment Image環境の提供



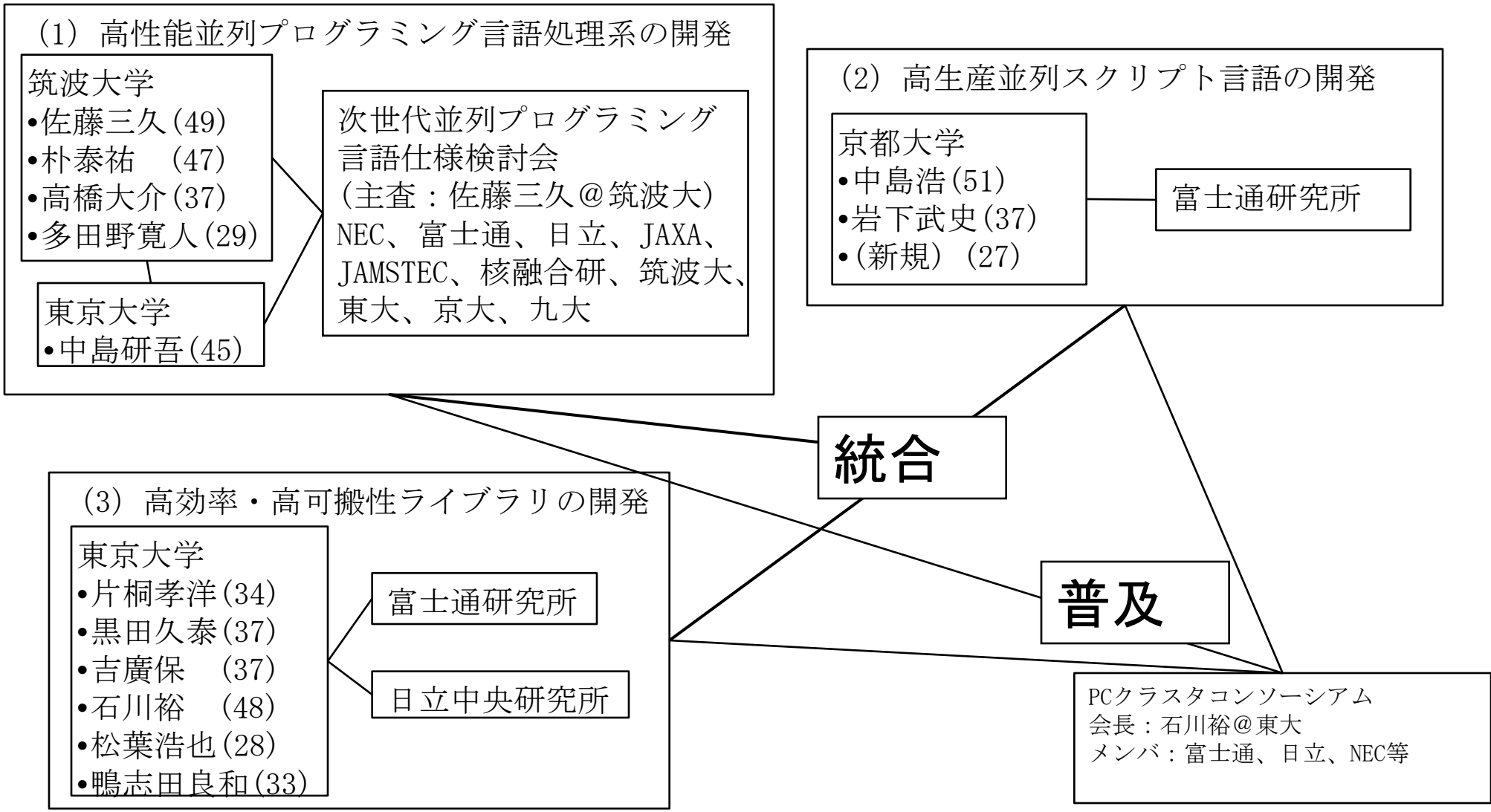


高性能高可搬性ライブラリ [Single Runtime Environment] (2/2)

- 研究項目
  - 1万プロセスジョブの各プロセスがファイルを生成できるユーザレベルスケーラブルファイルシステム（ステージング含む）の開発
  - 可搬性を持ったMPI-IOの開発
  - 管理者権限を必要としない、PCクラスタからセンターマシンで実行できる単一実行時環境の開発

  これら研究項目は実施責任者石川の既存研究成果であるSCoreクラスタシステムソフトウェアを基礎とし、富士通研究所、日立中央研究所と共同研究
- 期待される成果
  - PCクラスタでもセンターマシンでも同一の実行時環境を提供するライブラリの実現
  - PCクラスタ上で開発されたアプリケーションの実行イメージが、PCクラスタだけでなく、PCと同一CPUアーキテクチャを持つセンターマシンでも実行できるシングルバイナリを生成できる環境が実現される

体制


(1) 高性能並列プログラミング言語処理系の開発
筑波大学
•佐藤三久(49)
•朴泰祐　(47)
•高橋大介(37)
•多田野寛人(29)
東京大学
•中島研吾(45)
次世代並列プログラミング言語仕様検討会
(主査：佐藤三久@筑波大)
NEC、富士通、日立、JAXA、JAMSTEC、核融合研、筑波大、東大、京大、九大
(2) 高生産並列スクリプト言語の開発
京都大学
•中島浩(51)
•岩下武史(37)
•(新規) (27)
富士通研究所
統合
(3) 高効率・高可搬性ライブラリの開発
東京大学
•片桐孝洋(34)
•黒田久泰(37)
•吉廣保　(37)
•石川裕　(48)
•松葉浩也(28)
•鴨志田良和(33)
富士通研究所
日立中央研究所
普及
PCクラスタコンソーシアム
会長：石川裕@東大
メンバ：富士通、日立、NEC等

独創性、優位性

- 高性能並列プログラミング言語
  - 既存言語(Fortran, C)をコンパイラ指示文により拡張することにより並列化機能を提供、習得のコストを低減、性能チューニング作業を支援することにより、生産性と実用性を向上
  - 国内3メーカ（NEC、富士通、日立）とユーザ（JAXA、JAMSTEC、核融合研）による言語仕様検討委員会を組織し、OpenMP, OpenMPD, HPFの経験と意向を取り入れた仕様検討。実施責任者は、OpenMP処理系であるOmni OpenMPを開発、OpenMPD仕様を設計
- 高生産スクリプト言語
  - 既存スクリプト言語では不可能な、バッチジョブシステムで実行される大量並列計算処理によるパラメータ最適化作業の自動化を記述する言語の実現
  - 実施責任者が開発したMegaScript、富士通研究所で開発されたPJOを基礎とすることによるスクリプト言語仕様の連続性確保
- 高生産高可搬性ライブラリ
  - 自動チューニング（AT）付き数値計算ライブラリ実行時における、入力データ特徴を考慮したアルゴリズムや実装方式の選択。既存ファイルシステムでは不可能な、1万ファイルを超えるファイル生成を可能とする可搬性に優れたファイルシステムの提供
  - 実施責任者が開発したABCLibおよび国内メーカにおける自動チューニング研究者（日立中央研究所）との共同研究による実用化支援。実施責任者が開発したSCoreクラスタシステムソフトウェアを基礎とした早期ユーザ獲得

成果の利活用計画

- **成果適用対象**
  - 研究室レベルで使用されているPCクラスタから大学情報基盤センターで運用されているスパコンで使われるソフトウェア
- **実用化に向けた計画、普及方策、知的財産権の活用方法**
  - 提案機関の東大、筑波大、京大の**各情報基盤センターのユーザに利用**
  - PCクラスタユーザに普及
    - 開発したソフトウェアはオープンソースとして公開
    - **PCクラスタコンソーシアムと連携**
  - 国内外の計算センターに普及
  - 7大学情報基盤センターで進めているスパコン民間利用の取り組み（共用イノベーション）と連携し、産業界での利用促進も進める
- **アウトリーチ活動**
  - **PCクラスタコンソーシアムと連携**してアカデミックユーザおよび一般企業に向けた発信を行っていく。また、WEB上での情報発信や、大学の一般公開時に本研究開発活動の紹介を行っていく

社会的効果

- 本事業で開発されたシステムソフトウェアは、PCクラスタからスパコンまでを橋渡しするものである
- PCクラスタ上で開発されたアプリケーションが大学に設置されたスパコン等にスムーズに移行できることにより、PCクラスタユーザが大規模並列処理を行うようになり、次世代スパコンユーザ予備軍が増大する
- 産業界での利用促進により、イノベーション創出に貢献する
- 並列プログラミング言語に関しては、高性能化する組み込み向けのシステムへの技術転用の可能性もある

経済波及効果

- スパコンユーザの増大は、並列処理市場の拡大につながり、ハード・ソフトの両面で産業界の活性につながる
- シミュレーションの速度が向上すること、および、その種類・応用範囲が広がることで、ものづくり・開発費用が削減される
- 多くの高品質なアプリケーションが短期間で開発され、結果として、アプリケーション開発コストの低減につながる
- 一つのアプリケーションを、4つの異なる種類のスパコンで動かすための工数を考えると、コストは1/4になる。
- 本事業で開発されたシステムソフトウェアを大学、民間で利用されることにより、科学・工学系の研究者や技術者のスパコン利用が加速する
- 2006年の国内スパコン市場は430億円であり、2005年に比べて20.3%減少しているという報告があるが、利用者の増大により、市場拡大が望める
- 仕様の国際標準化やオープンソース化を進めていくことにより、今までスパコンを利用してこなかったユーザ層（例えば生産管理システム）が増大し並列処理市場が活性する。これにより、国内産業が活性化する

用語集 (1/2)

- 並列コンピュータ
  - たくさんのCPUが搭載されたコンピュータ。ひとつのアプリケーションプログラムは、これらCPUを使って高速に実行される
- 共有メモリ型並列コンピュータ
  - 複数のCPUが主記憶（メモリ）を共有し、共有データを直接参照できる並列コンピュータ
- 分散メモリ型並列コンピュータ
  - CPU毎に主記憶を持ち、CPU間で共有データを参照するためにはMPI通信ライブラリのようなソフトウェアを必世とする並列コンピュータ
- スカラ型並列計算機、ベクトル型並列計算機
  - 一般に普及しているCPUを使用した並列計算機をスカラ型並列計算機と称し、ベクトル計算を高速に計算するための特殊なCPUが搭載された並列計算機をベクトル並列型並列計算機と称する
- PCクラスタ
  - 多数のパソコンをネットワークで繋いだ並列コンピュータ。比較的安価に高性能システムを構築する手段として、大学の研究室や企業で普及
- MPI通信ライブラリ
  - 分散メモリ型並列コンピュータで必要となるデータ交換を指示するためのライブラリ
- 並列プログラミング言語
  - 並列コンピュータ上で効率よくプログラムを実行できるようにするためのプログラミング言語。MPI通信ライブラリを使わなくてもデータ交換を自動的に行う
- Omni OpenMP、OpenMPD
  - 筑波大学の佐藤三久が開発を主導したOpenMP 処理系。OpenMPDは、OpenMPを拡張した言語。本提案の並列プログラミング言語処理系の基礎となる。
- OpenMP
  - 共有メモリ型並列コンピュータ向けに開発されたプログラミング言語。OpenMPでは、FortranやCプログラムにコメント行として並列化のための指示文を記述するだけで良い

用語集 (2/2)

- スクリプト言語
  - 本来、アプリケーションプログラムの実行、実行結果の編集操作を記述する目的の言語を指していた。Perl, Python, Ruby等が代表的スクリプト言語だがバッチジョブの記述ができない
- MegaScript、PJO
  - 京都大学の中島浩により開発された並列コンピュータ向けスクリプト言語。多数の並列プログラムをストリームと呼ぶ通信路で結合する機能が特徴
  - 富士通研究所により開発された並列コンピュータ向けスクリプト言語。投入した多数のバッチジョブの個々の実行結果に依存する処理をパターンマッチングにより簡潔に記述できるのが特徴
- バッチジョブ
  - アプリケーションプログラムを必要な資源が確保できしだい実行するバッチスケジューリングにおける実行単位。階層型並列処理では多数のバッチジョブが並列に実行される
- パラメータ探索
  - 入力パラメータを様々に変化させて多数のアプリケーションを実行し、その結果から最適なものを選択することで、最適なパラメータ値などを見出す探索処理
- ABCLib
  - 東京大学片桐孝洋により開発された自動チューニング付きライブラリ
- PCクラスタコンソーシアム
  - 経済産業省（当時通産省）の国家プロジェクトにおいて開発されたSCore(エスコア）クラスタシステムソフトウエアならびにOmni OpenMP処理系を中核としたクラスタシステムソフトウエアの開発の継続、維持、普及を通して、PCクラスタシステム市場育成に貢献することを目的として、2001年10月に発足。現在、会員企業23社（富士通、NEC、日立）、研究機関・大学等3団体。会長は東京大学の石川裕
- SCore (エスコア)
  - 東京大学の石川裕が開発を主導したクラスタシステムソフトウェア。SCoreは、筑波大学のPACS-CSや理化学研究所のSuper Combined Clusterなど、計算センター運用されているスーパーコンピュータで使用されている。本提案のSingle Runtime Environmentの基礎となる

高生産スクリプト言語の記述イメージ（PJO ベース）

```
package MySearch;
use base Search;  use BranchAndBound;                    # ライブラリ利用宣言
$prog=new("mpiexec –n 1024 ./sim", ...);                 # 実行プログラムの宣言的記述
$maxconc=16;                                             # 上位階層並列度の定義
@initspace=(...);                                        # 初期探索空間の設定
BranchAndBound->exec($prog,$maxconc,@initspace);         # 探索実行
```

```
sub ParamGen { return【探索対象ノードに対する入力引数･ファイルを生成】; }
```

```
sub Eval { return【実行結果から評価値を算出】; }
```

```
package BranchAndBound;                                  # システム定義探索ライブラリ
sub exec { my ($prog,$conc,@space)=@_;
  ($conc,$space)=execute($prog,$conc,$space);            # 最初のジョブ群投入
  when {{ job=$job }} {{                                 # 実行結果による探索継続
    my $eval=$prog->Eval($job->results());
    push(@space,Branch($job->{Node}) if (!Bounded($eval));
    ($conc,@space)=execute($proc,++$conc,$space);
  }}
}
sub execute { my ($prog,$conc,@space)=@_;                # ジョブ実行関数
  while ($conc-->0 && @space) {
    my $node=shift(@space);
    my $job=$prog->newjob($node,$prog->ParamGen($node));
    do_job {{ job=$job }} {{ system($job->{COMMAND});
    }}
  }
  return($conc,@space);
}
```

高性能並列プログラミング言語

プログラム例

```
int array[YMAX][XMAX];

#pragma ompd distvar(var = array;dim = 2)

main(){
  int i, j, res;
  res = 0;

#pragma ompd for affinity(array) reduction(+:res)
  for(i = 0; i < 10; i++)
    for(j = 0; j < 10; j++){
      array[i][j] = func(i, j);
      res += array[i][j];
    }
}
```

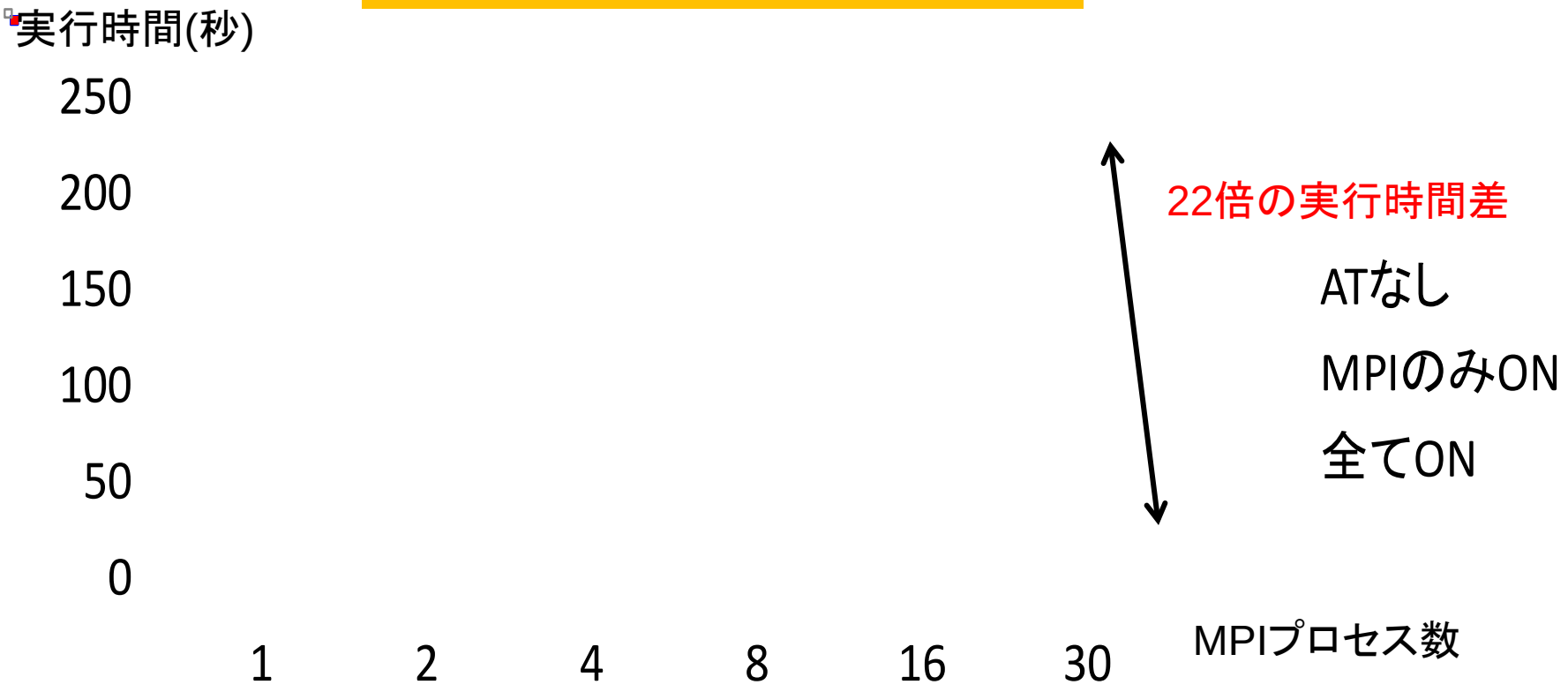
本提案で開発される処理系は、#pragma ompdで始まる指示行を解釈して、最適な並列コードを生成する。既存処理系は、これら指示行をコメント行として扱われる。したがって、指示行を挿入したことにより既存処理系でコンパイルできなくなることはない。

自動チューニング付き数値計算ライブラリ

疎行列連立一次方程式反復解法ライブラリ
・行列サイズ：512,000 (3次元楕円型偏微分方程式の境界値問題の7点差分式) 非零要素：1行あたり最大7個
・GMRES法 ・PCクラスタ（WindowsCCS、デュアルコア Xeon 3040（1.86GHz、FSB 1066MHz、共有L2キャッシュ 2MB）

実装したAT機能（実行時に選択）
1．前処理選択 2．リスタート周期設定 3．疎行列積用アンローリング方式選択 4．通信方式選択（MPI）

自動チューニング機能による効果の例



自動チューニング付き数値計算ライブラリ

疎行列反復解法におけるAT実現の一例