

CUDA Fortran/C OpenACCによる高速化!

ポータリングサービス

自社解析プログラムのGPU化の効果・予測値等を知りたい

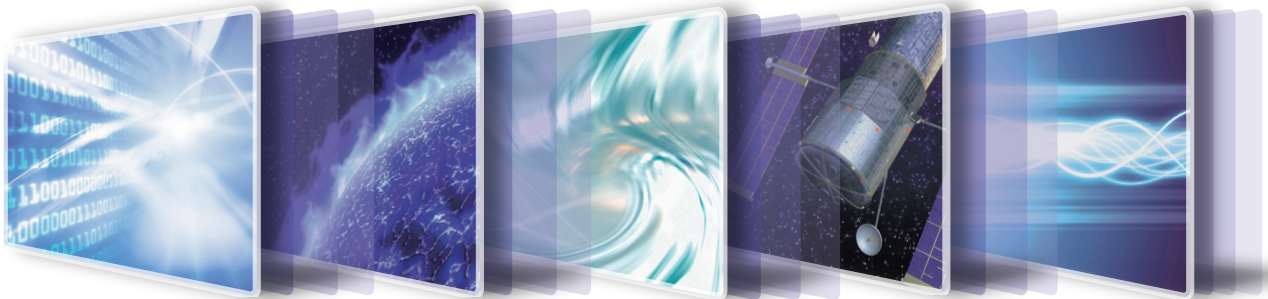
自社解析プログラムの並列化アドバイスをもらいたい

自社解析プログラムの実行時間に不満がある

▶▶▶ アルゴリズム
再考箇所を知りたい

▶▶▶ もっと速く
計算出来ないか?

このような方々に
オススメ!



▶ ソースコード解析によるスピーディー診断!

- ・独自のツール(コールツリーグラフ、依存解析グラフ)、コンパイラの自動並列化オプション、プロファイリングツールを使って、ソースコード解析から並列化の可能性を診断
- ・部分実装や経験値をもとに効果の予測値を提示

▶ CUDA Fortran/C OpenACCによるGPU化!

- ・ホットスポットを含むプログラムの実行開始から終了までプログラム全体の高速化を実現
- ・FortranコードはFortranのまま高速化

▶ グラフィックから大規模科学計算まで対応

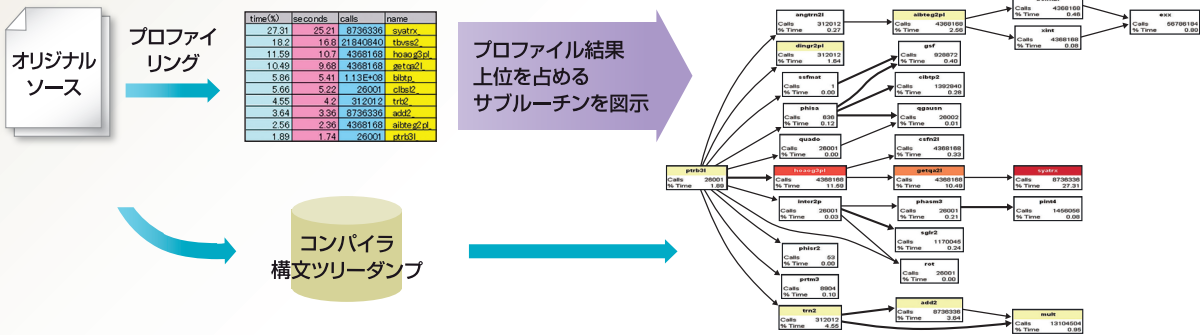
- ・ビデオコーデック、トランスコード
- ・流体解析(津波など)、構造解析
- ・ビジュアルゼーション、各種画像処理
- ・分子動力学計算
- ・気象(大気/海洋/温暖化)、宇宙科学



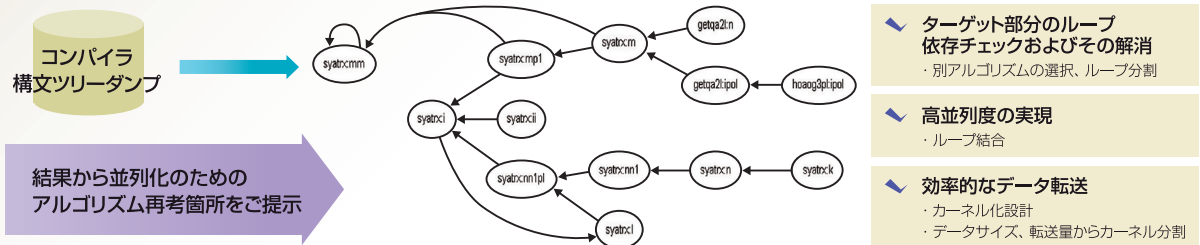
ポータリングサービス

ソースコード解析でGPU化の流れ

1. 初期解析(プロファイリング) 処理時間の大半を占めるホットスポットを見つける!!



2. 並列化検討 ホットスポット部分のループ依存、タスク依存を変数参照によってチェック



3. コード生成

GPUのための並列化コード作成
CUDA Fortran/C OpenACC

- ループスケジューリング調整
- ループ内コードの効率化

チューニング

4. 性能評価

プロファイラ等でチェックしながら、さらなる最適化を実施

- 各種ツールを使った診断
- メモリバンド幅算出による診断

※記載されている全ての会社名および商品名は、各社の商標または登録商標です。 ※製品の仕様は予告なく変更することがありますので、あらかじめご了承ください。