

XcalableMPの現状と課題

佐藤 三久
理研 AICS

XcalableMP

- 分散メモリ環境を対象とした指示文ベースの
並列言語
- 次世代並列プログラミング言語検討委員会
→ 当部会において仕様を検討、提案。
- 2つの並列プログラミングモデルをサポート
 - グローバルビューモデルによる定型的な並列化
 - ローカルビューモデルによる自由度の高い並列化

XcalableMP as evolutionary approach

- We focus on migration from existing codes.
 - Directive-based approach to enable parallelization by adding directives/pragma.
 - Also, should be from MPI code. Coarray may replce MPI.
- Learn from the past
 - Global View for data-parallel apps. Japanese community had experience of HPF for Global-view model.
- Specification designed by community
 - Spec WG is organized under the PC Cluster Consortium, Japan
- Design based on PGAS model and Coarray (From CAF)
 - PGAS is an emerging programming model for exascale!
- Used as a research vehicle for programming lang/model research.
 - XMP 2.0 for multitasking.
 - Extension to accelerator (XACC)

関連したプロジェクト

- E-scienceプロジェクト検討 (2008)
- PC Cluster コンソーシアム XcalableMP 規格部会での議論
- 日仏 FP3Cプロジェクト
- 理研計算科学研究機構で、京コンピュータ向けに開発
- ポストペタCREST (研究代表者: 朴@筑波大、分担 村井)
 - XcalableACC (XMP+OpenACC) for accelerated Cluster
- SPPEXA (研究代表者: 朴@筑波大、分担 村井)
 - ドイツ アーヘン大学、フランス MDLSとのXMPのプログラム論理検証技術
- FS2020 プロジェクト
 - ポスト京向けにXMP2.0 を検討

XcalableMPの現況

- PCクラスタコンソーシアムで規格を議論
 - 2016/11に現行のVersion 1.2.3仕様を公開。
 - Version 1の系統は「収束」した。
 - エラッタを除き、検討を終了。
 - 次期仕様「XMP2.0」の検討を開始。
 - 2016/11公開予定(?)

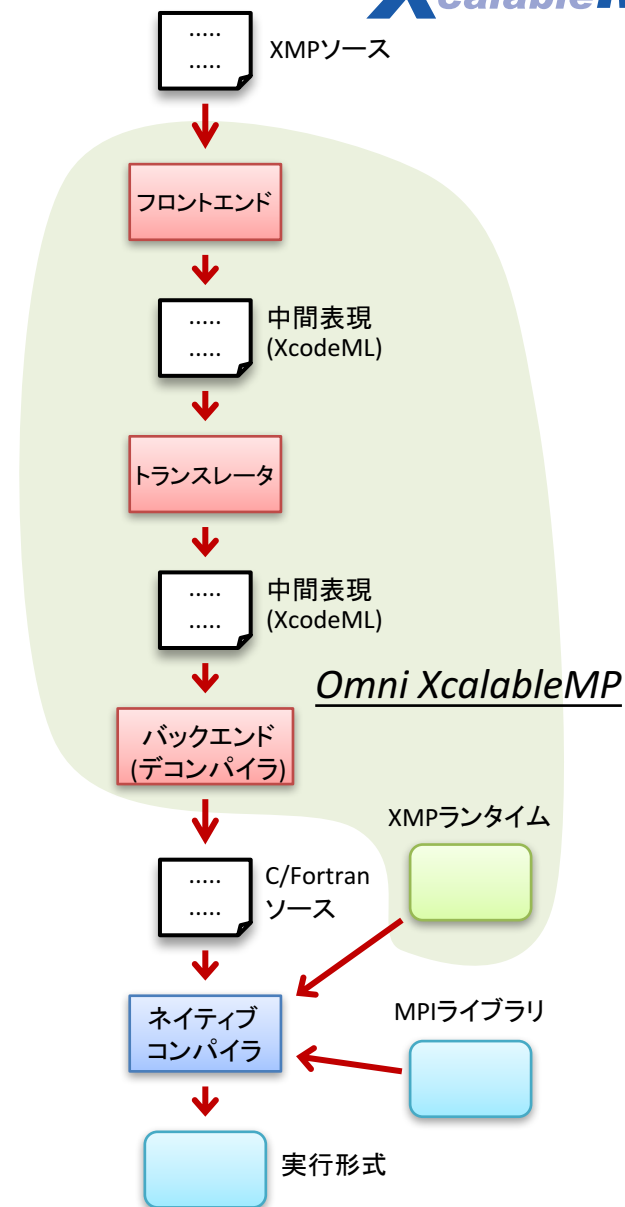
www.xcalablemp.org

- 理研・筑波大で、レファレンス実装
 - Omni XMP コンパイラ

omni-compiler.org

Omni XcalableMP

- 理研AICSと筑波大で開発中のXMP処理系
 - XMP/C
 - XMP/Fortran
- オープンソース
- トランスレータ + ランタイム (MPIベース)
- OpenACC、XcalableACC対応



XcalableMPプロジェクトの これからの方向性 (1)

- 言語、コンパイラ研究とともに、使ってもらうことが大切
- これまでの方針：並列プログラムを容易にする
 - Global view: 逐次プログラムの並列化を容易にする。
 - Local view: PGAS(one-sided comm)を用いた並列化。MPIよりも理解が用意。(MPIと同等なプログラミングが可能)
 - Fortran90(95)のサポート
- **すでにMPIの実装がある場合には、何がメリットか？**

XcalableMPプロジェクトの これからの方向性 (2)

- **これからの方針 (私案)**
 - **性能を得るための並列プログラミング**
 - 目標: MPIと同等**以上**。
 - 通信ライブラリとのco-design
 - 基本的には、one-sided commは、light-weightなはず。
 - 現在、MPI3を含め、いろいろな通信ライブラリで性能評価を行っている
 - マルチコアで性能を得るためのプログラミングモデル
 - KNLでは、多数のコアが通信しないと性能がでない、場合がある。

“MPI+X” for exascale?

- **X is OpenMP!**
- **“MPI+Open” is now a standard programming for high-end systems.**
 - I'd like to celebrate that OpenMP became “standard” in HPC programming
- **Questions:**
 - “MPI+OpenMP” is still a main programming model for exa-scale?

Question

- What happens when executing code using all cores in manycore processors like this ?

```
MPI_recv ...  
#pragma omp parallel for  
for ( ... ; ... ; ... ) {  
    ... computations ...  
}  
MPI_send ...
```

Data comes into “main” in shared memory”

Cost for “fork” become large

data must be taken from Main memory

Cost for “barrier” become large

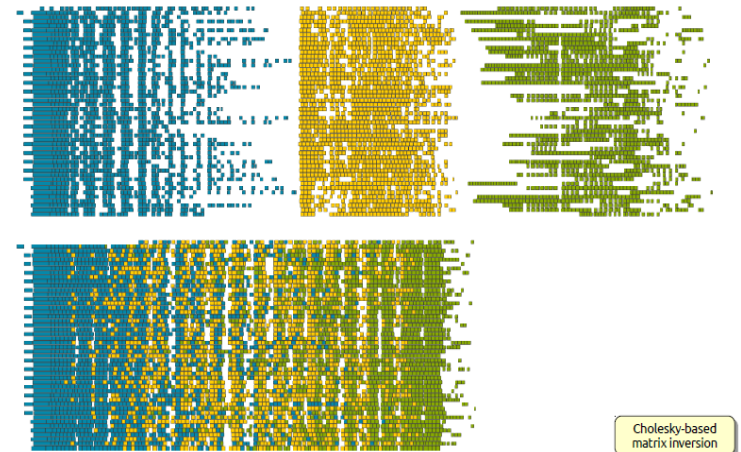
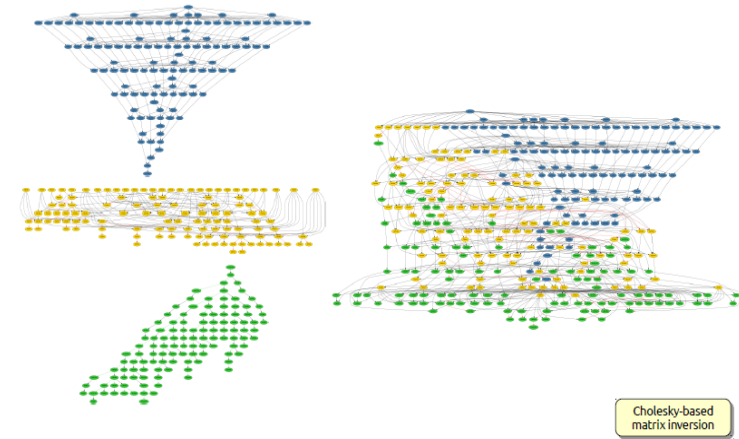
MPI must collect data from each core to send

- What are solutions?

- MPI+OpenMP runs on divided small “NUMA domains” rather than all cores?

XcalableMP 2.0の設計・実装に向けて

- ターゲットはメニコア
- XMP2.0=multitasking+PGAS
- Multi-taking機能については、Argobotを用いる予定。
 - Tasklet 指示文
 - Thread-to-thread synchronization /communications rather than barrier
- 利点
 - マルチスレッド環境下での通信と計算のオーパラップ
 - 各コアからの通信を行うことにより、メッセージのスループットにより、バンド幅を稼ぐ
 - コストの高い大域的な同期(バリア)を行わない。



From PLASMA/QUARK slides by ICL, U. Tennessee

Invited Talk

- Sangmin Seo (Argonne National Laboratory)
 - **User-Level Threads and OpenMP**
 - Runtime for XMP 2.0 multithreading
- Valentin Clément (C2SM, ETH Zurich / MeteoSwiss)
 - **CLAW: One code to rule them all**
 - User of Fortran Front-end