

# Omni XcalableMP Compilerの概要

下坂 健則  
理化学研究所 計算科学研究機構

2011/11/01

# 目次

- 開発概要
- Omni XcalableMP Compilerの構造
- Omni XcalableMP Compilerの特徴
- インストール方法
- 講習会活動
- 課題
- まとめ

# 開発概要

- 筑波大CCSと理研AICSで開発中。
- オープンソースプロジェクト。
- XMP/C, XMP/Fortranをパッケージ化して提供。
- 2013/3にVer. 0.6.1をリリース済み。XcalableMP規格仕様v1.1のおおよその機能を使用可能。サポート機能の詳細は、以下のサイトで公開中。  
<http://www.hpcs.cs.tsukuba.ac.jp/omni-compiler/xcalablemp/download.html>
- SC13にて, ver. 0.7.0を, リリース予定。
- サポートプラットフォーム  
Linux x86系クラスタ, 京, FX10, Cray XE5,  
NEC SX\*, Earth Simulator\*, SR16000\*, BlueGene/Q\*  
\* インストール確認中

## Ver. 0.7.0の主なサポート項目

- OpenMP指示文のサポート
- task指示文(XMP/F)のサポート
- gmove指示文による通信の機能拡張
- reflect指示文による通信の性能改善
- Coarray機能の京向け性能改善
- post/wait指示文の京向け性能改善

# 提供サイト



HOME

Download

Document

Publication

Mailing List

Links

## Omni XcalableMP Compiler

### Stable Version

<https://sourceforge.net/projects/xcalablemp/files/>

### Nightly Build Version

[omnixmp-R1412.tar.bz2](#) (2013/10/28)

### License

The Omni XcalableMP compiler is a free software. You can redistribute it and/or modify it under the terms of the **GNU Lesser General Public License version 2.1** published by the Free Software Foundation.

### Implementation Status

- Global-View

Directive	Kind, Clause	XMP/C		XMP/Fortran	
		Stable	Nightly	Stable	Nightly
nodes	-				

# HPC Challenge Class2 & 展示履歴

<b>Publication</b>
<b>Mailing List</b>
<b>Links</b>

International Conference on PGAS Programming Models, Edinburgh, Scotland, UK, October 2013.

- ▶ Masahiro Nakao, Jinpil Lee, Taisuke Boku and Mitsuhsisa Sato. "Productivity and Performance of Global-View Programming with XcalableMP PGAS Language", CCGrid 2012 – The 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, Ottawa, Canada, May, 2012. [[Slide](#)]
- ▶ Jinpil Lee, Minh Tuan Tran, Tetsuya Odajima, Taisuke Boku and Mitsuhsisa Sato. "An Extension of XcalableMP PGAS Language for Multi-node GPU Clusters", Ninth International Workshop on Algorithms, Models and Tools for Parallel Computing on Heterogeneous Platforms (HeteroPar 2011), 2011.
- ▶ Masahiro Nakao, Jinpil Lee, Taisuke Boku, Mitsuhsisa Sato. "XcalableMP Implementation and Performance of NAS Parallel Benchmarks", Fourth Conference on Partitioned Global Address Space Programming Model (PGAS10), NewYork, USA, Oct., 2010. [[Slide](#)]
- ▶ Jinpil Lee and Mitsuhsisa Sato. "Implementation and Performance Evaluation of XcalableMP: A Parallel Programming Language for Distributed Memory Systems", The 39th international Conference on Parallel Processing Workshops (ICPPW10), pp.413-420, 2010.

## Presentations and others

- ▶ [SC12 HPCC Class2 Submission \[Paper\]](#) [[Slide](#)]
- ▶ SC12 Research Exhibition [[Poster](#)]
- ▶ XcalableMP Ver. 1.0 Highly Productive Parallel Programming Language, [WPSE2012 \[Presentaion\]](#)
- ▶ SC11 Research Exhibition [Poster[1](#), [2](#), [3](#)] [[leaflet](#)] [[Presentation](#)]
- ▶ SC10 Research Exhibition [Poster[1](#), [2](#)] [pamphlet[1](#), [2](#), [3](#), [4](#)] [[Presentation](#)]
- ▶ [SC10 HPCC Class2 Submission \[Paper\]](#) [[Slide](#)]
- ▶ Preliminary Performance Report on programming in XcalableMP, [WPSE2010 \[Presentaion\]](#)
- ▶ SC09 Research Exhibition [[leaflet](#)]
- ▶ [SC09 HPCC Class2 Submission \[Paper\]](#) [[Slide](#)]

過去3回挑戦。  
SC09, SC10では  
Honorable  
Mentionを獲得

# サポート

[HOME](#)[Download](#)[Document](#)[Publication](#)[Mailing List](#)[Links](#)

## Omni XcalableMP Compiler

### Omni XcalableMP Compiler Q & A Mailing List

To help users, we have established an Omni XcalableMP Compiler Users mailing list. We provides two mailing lists, **xmp-users** mailing list and **xmp-users-ja** mailing list. English is used in the xmp-users mailing list, and Japanese is used in the xmp-users-ja mailing list.

#### xmp-users in English

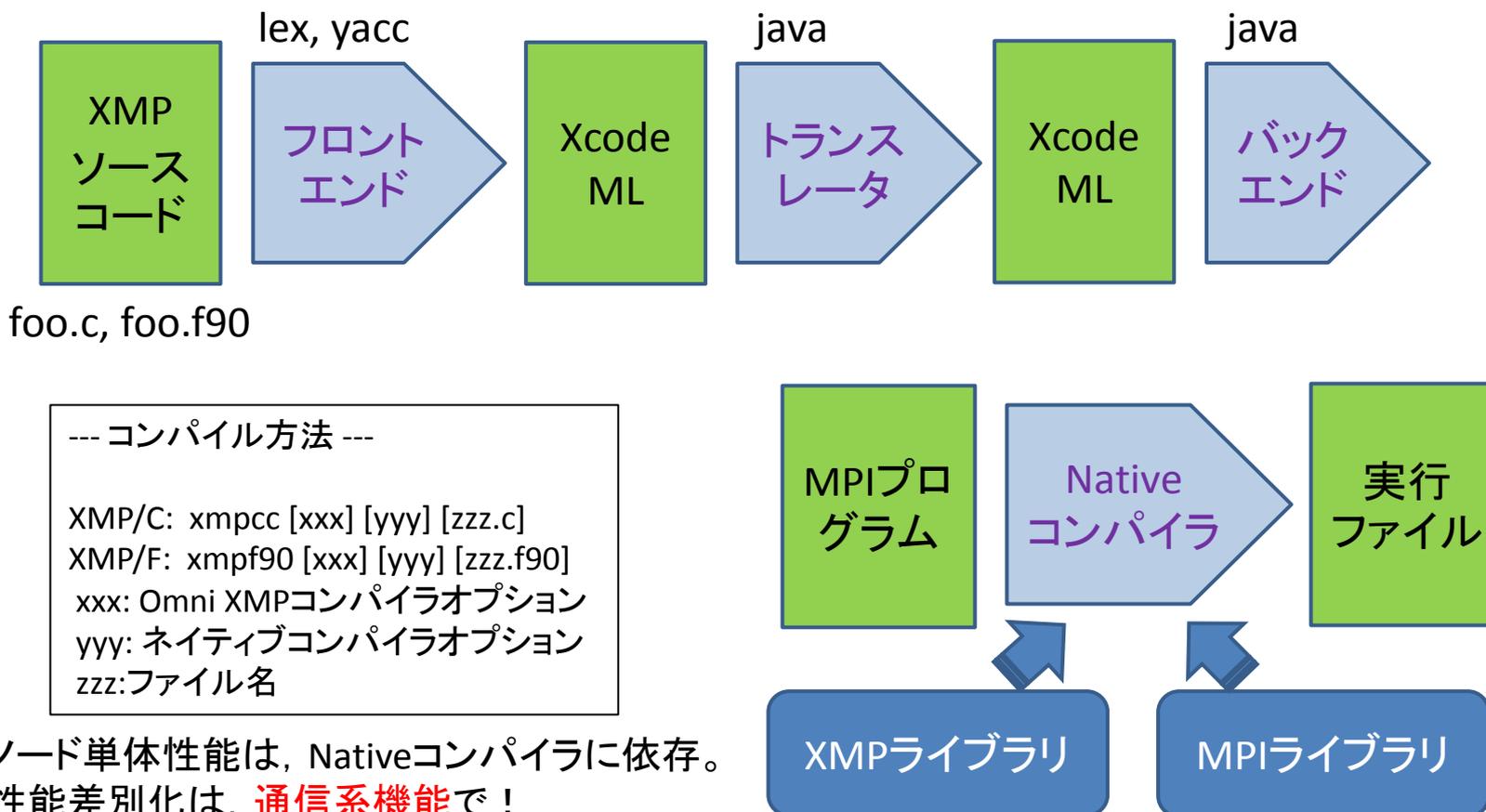
1. To join, please visit <https://mail.hpcs.cs.tsukuba.ac.jp/mailman/listinfo/xmp-users>.
2. If you have any questions, please send email to "xmp-users [at] hpcs.cs.tsukuba.ac.jp" or search mail archive (Mail archive page is [HERE](#)).

#### xmp-users-ja in Japanese

1. To join, please visit <https://mail.hpcs.cs.tsukuba.ac.jp/mailman/listinfo/xmp-users-ja>.
2. If you have any questions, please send email to "xmp-users-ja [at] hpcs.cs.tsukuba.ac.jp" or search mail archive (Mail archive page is [HERE](#)).

# Omni XcalableMP Compilerの構造

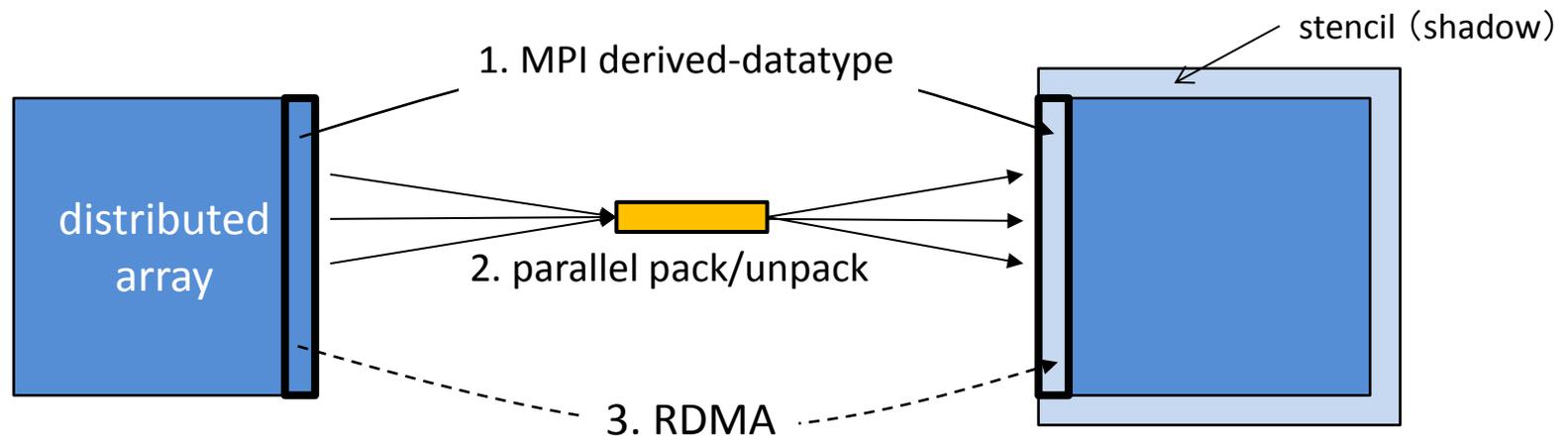


# Omni XcalableMP Compilerの特徴

- Omni XcalableMPライブラリでの最適化
- MPI -> XMP, XMP->MPIの双方の呼び出しが可能
- アクセラレータに対応(XMP-dev)
- プロファイリングインターフェース  
(Scalasca, tlog連携)
- 外部ライブラリインタフェース  
(MUMPS, ScaLAPACKの一部)

# ライブラリでの最適化

- reflect指示文に対する最適化
  - プラットフォームにより、以下の3通りから静的に選択



- Coarray記法に関する最適化
  - XMP/Cにて、GASNetもしくはRDMAで実装

# MPI -> XMP呼び出し例

## MPIプログラム

```

.....
if (p->c_size){
    xmp_pzfft1d0_( a, b,
        c + i * (nx * 2 + FFTE_NP),
        c + i * (ny * 2 + FFTE_NP),
        w, &nx, &ny, timings );
}
else{
    xmp_pzfft1d0_( a, b,
        b + i * (nx * 2 + FFTE_NP),
        a + i * (ny * 2 + FFTE_NP),
        w, &nx, &ny, timings );
}
.....

```

## XMPプログラム

```

SUBROUTINE XMP_PZFFT1D0(A,B,CX,CY,W,NX,NY,TIMINGS)
IMPLICIT REAL*8 (A-H,O-Z)

```

!

```

COMPLEX*16 A(NX,NY)
COMPLEX*16 B(NY,NX)
COMPLEX*16 CX(*),CY(*)
COMPLEX*16 W(NY,NX)
DIMENSION TIMINGS(*)

```

A, B, Wは  
グローバル配列

```

!$XMP nodes p(*)
!$XMP template tx(NX)
!$XMP template ty(NY)
!$XMP distribute tx(block) onto p
!$XMP distribute ty(block) onto p
!$XMP align A(*,i) with ty(i)
!$XMP align B(*,i) with tx(i)
!$XMP align W(*,i) with tx(i)

```

この場合、  
MPI\_COMM\_WORLD  
相当のプロセス集合

....

# XMP -> MPI呼び出し例

## XMPプログラム

```

SUBROUTINE XMP_PZFFT1D0(A,B,CX,CY,W,NX,NY,TIMINGS)
  IMPLICIT REAL*8 (A-H,O-Z)
  !
  ....
  !$XMP nodes p(*)
  !$XMP template tx(NX)
  !$XMP template ty(NY)
  !$XMP distribute tx(block) onto p
  !$XMP distribute ty(block) onto p
  !$XMP align A(*,i) with ty(i)
  !$XMP align B(*,i) with tx(i)
  !$XMP align W(*,i) with tx(i)
  ....
  CALL ZFFT1D(B,NY,0,CY(ITHEADS*(NY*2+8)+1))
  ....

```

## MPIプログラム

```

SUBROUTINE ZFFT1D(A,N,IOPT,B)
  IMPLICIT REAL*8 (A-H,O-Z)
  INCLUDE 'param.h'
  COMPLEX*16 A(*),B(*)
  COMPLEX*16 C((NDA2+NP)*NBLK),D(NDA2)
  COMPLEX*16 WX(NDA2),WY(NDA2)
  DIMENSION IP(3),LNX(3),LNY(3)
  SAVE WX,WY
  C
  CALL FACTOR(N,IP)
  ....

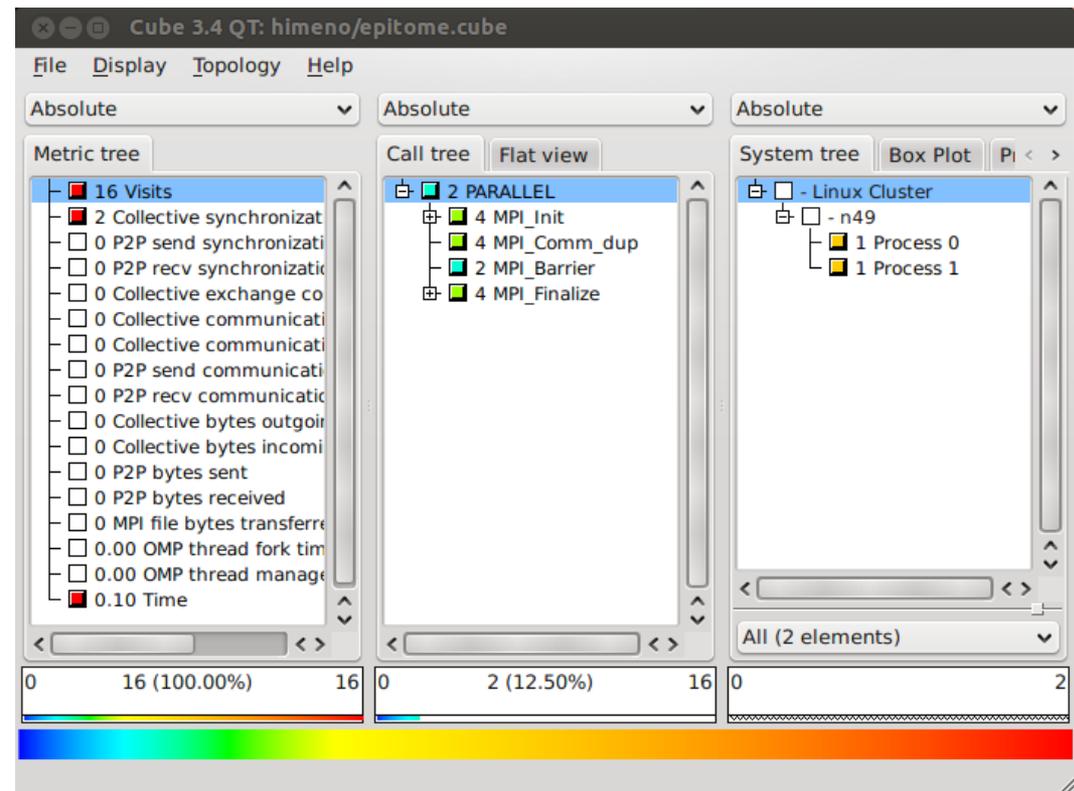
```

分散配列のアドレスはXMPコンパイラ  
がローカルアドレスに変換。

# プロファイリングインタフェース(Scalasca)

1. Scalasca ver. 1.4.xをインストール。
2. PATH, SCALASCA\_HOME環境変数を設定。
3. `xmpcc -with-scalasca a.c`
4. `scalasca -analyze mpiexec -n 2 ./a.out epik_xxx/xxx.cube`ファイルが生成。
5. X11(仮想環境上のLinuxなど),  
が使える環境で, 以下のコマンドを実行  
`scalasca -examine xxx.cube`
6. 右図のような画面が表示される。

※上記手順の動作プラットフォーム例  
VMWare Player + Ubuntu環境  
+ Linux x86\_64マシン



# プロファイリングインタフェース(tlog)

## XMPプログラム(foo.c)

```
.....
#pragma xmp reflect (p)

#pragma xmp loop (i,j) on t(i,j)
for (i = 1 ; i<imax-1 ; i++)
  for (j = 1 ; j<jmax-1 ; j++)
    for (k = 1 ; k<kmax-1 ; k++){
      s0 = a[0][i][j][k] * p[i+1][j][k] +...
      ss = ( s0 * a[3][i][j][k] - p[i][j][k] ) * bnd[i][j][k];
      gosa+= ss*ss;
      wrk2[i][j][k] = p[i][j][k] + omega * ss;
    }
}
```

```
#pragma xmp reduction(+:gosa) profile
```

```
#pragma xmp loop (i,j) on t(i,j)
for (i = 1 ; i < imax-1; ++i)
  for (j = 1 ; j < jmax-1; ++j)
    for (k = 1 ; k < kmax-1; ++k)
      p[i][j][k] = wrk2[i][j][k];
.....
```

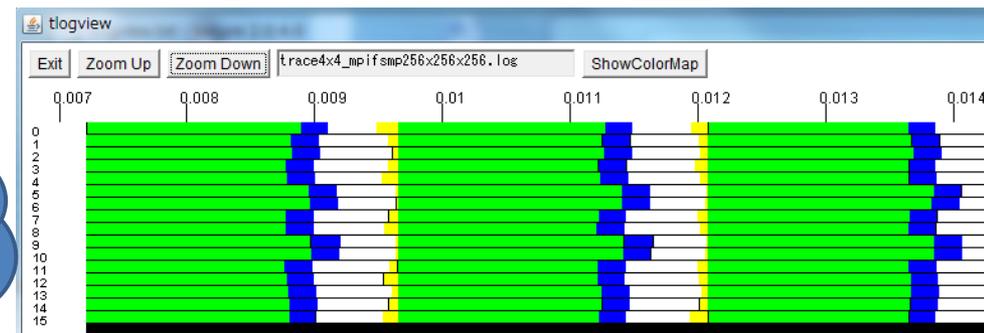
指示文単位  
に経過時間  
のログを表示

指示文を限定  
したい場合、  
profileを追記。

```
$xmpcc -with-tlog foo.f90 (-profile)
(現在、xmpccのみ使用可)
```

実行後、trace.logファイルが出力される。  
X11が使える環境(仮想環境上のLinuxなど)  
で、以下のコマンドを実行する。

```
$tlogview trace.log
```



# 外部ライブラリインタフェース例(MUMPS)

## XMPプログラム

```
#pragma xmp nodes p(*)
#pragma xmp template t(0:nnz-1)
#pragma xmp distribute t(cyclic) onto p
#pragma xmp align a[i] with t(i)
#pragma xmp align irn[i] with t(i)
#pragma xmp align jcn[i] with t(i)
```

```
int main(int argc, char *argv[]){
```

```
.....
```

```
id.icntl[17]=3;
```

```
id.job=6;
```

```
ixmp_dmumps_c(&id, xmp_desc_of(irn),
              xmp_desc_of(jcn), xmp_desc_of(a));
```

```
.....
```

分散配列のデータ  
マッピング情報を  
渡すため、ネイティブ  
インタフェースに、  
ディスクリプタを追加。

データマッピング情報は、XMPが提供  
する問い合わせ関数で取得する。

## XMPインタフェース実装

```
#include "stdlib.h"
#include "dmumps_c.h"
#include "xmp.h"
```

```
void ixmp_dmumps_c(DMUMPS_STRUC_C *id,
                  xmp_desc_t dirn, xmp_desc_t djcn,
                  xmp_desc_t da){
```

```
.....
```

```
xmp_array_laddr(dirn,&irn_laddr);
```

```
xmp_array_laddr(djcn,&jcn_laddr);
```

```
xmp_array_laddr(da,&a_laddr);
```

```
id->irn_loc = *(void **)irn_laddr;
```

```
id->jcn_loc = *(void **)jcn_laddr;
```

```
id->a_loc = *(void **)a_laddr;
```

```
dmumps_c(id);
```

```
.....
```

```
}
```

# インストール方法(1)

事前にインストールしておくべきアプリケーション(前提PP)

- Lex, Yacc
- C Compiler (supports C99)
- Java Compiler (latest)
- Apache Ant (1.8.1 or later)
- MPI Implementation (supports MPI-2)
- libxml2
- GASNet (If you want to use coarray functions)

※Debian系Linuxの場合、あらかじめ以下を実行しておくとうい。

```
#aptitude install flex ant gfortran openmpi-dev openmpi-bin  
libxml2-dev byacc openjdk-7-jdk make
```

# インストール方法(2)

Linux x86\_64システム上での実行手順  
(Coarray機能を使わない場合)

1. 必要に応じ、前提PPのパスを設定

2. configure, makeの実行

```
e.g.) $ ./configure --prefix=/home/you/XMP  
$ make  
$ make install
```

# インストール方法(3)

※Coarray機能(GASNet)を使う場合のインストール例(Linux x86\_64 gnuコンパイラ+ Infiniband)

## 1. GASNetのダウンロード

<http://gasnet.cs.berkeley.edu/GASNet-1.20.2.tar.gz>

## 2. GASNetのインストール

```
$ tar xvzf GASNet-1.20.2.tar.gz
```

```
$ cd GASNet-1.20.2
```

```
$ ./configure --enable-ibv --prefix=(GASNetのインストール場所)
```

出力結果が、

OpenIB/OpenFabrics IB Verbs network conduit (ibv) ON となっていることを確認。

```
$ make
```

```
$ make install
```

## 3. XcalableMPのインストール

```
$ ./configure --prefix=(XMPのインストール場所) ¥
```

```
  --with-gasnetDir=(GASNetのインストール場所)¥
```

```
  --with-gasnet-conduit=ibv
```

```
$ make
```

```
$ make install
```

# 講習会活動

- 2回のXcalableMP利用講習会を実施済み。  
日時:7/18, 9/13 10:00～16:00  
内容: 1) XcalableMP概要  
2) reflect指示文を使ったステンシル計算の演習  
3) Coarray記法を使った行列積計算の演習

講習会資料は、以下で公開済み。

<http://www.hpcs.cs.tsukuba.ac.jp/omni-compiler/xcalablemp/document.html>

- 今後の予定  
第3回XcalableMP利用講習会  
日時:12/4(水) 10:00～16:00  
場所: FOCUS 2F実習室

# 課題

- リリース済み規格仕様の早期キャッチアップ。
- 外部ライブラリ, ツール連携の強化。
- 64bitインデックス対応。

# まとめ

- 開発概要の説明
  - SC13で, ver. 0.7.0をリリース予定
- Omni XcalableMP Compilerの特徴
  - XMP <-> MPI 呼び出しが可能
  - プロファイリングツールScalasca, tlog連携機能を使用可能
  - 外部ライブラリインターフェースが使用可能
- Omni XcalableMP Compilerの構造と使い方
  - コンパイルの流れ, 方法, インストール手順を示した。
- 講習会活動
  - 次回は, 12/4開催予定。