

# XcalableACC PGAS Language for Accelerated Clusters and XMPT Tool Interface

Hitoshi Murai RIKEN AICS



#### **Introduction (1)**

Directives for Accelerato

- Accelerated clusters have become very popular HPC platforms.
- MPI+CUDA style programming lowers productivity.
- Two directive-based languages exist:
  - **Calable MP** (XMP) as an alternative to MPI
  - OpenACC as an alternative to CUDA



### **Introduction (2)**

#### • Challenges in accelerated clusters

- Hierarchical parallelism
  - among nodes -> XMP
  - (within a node -> OpenMP)
  - among ACCs -> ?
  - within ACCs -> OpenACC



- Direct comm. among ACCs (physical or logical)
  - Tightly Coupled Accelerators (TCA)
  - NVLink
  - NVIDIA GPUDirect



#### Goals

- Proposing a new programming language for accelerated clusters by <u>combining</u> <u>XcalableMP and OpenACC</u>
  - hierarchical parallelism
  - direct communication among accelerators
- Developing its compiler

# Realizing high performance and productivity on accelerated clusters



### **Outline of This Talk**

- What are XcalableMP and OpenACC?
- Design of XcalableACC
- Implementation of the Omni XcalableACC compiler
- Evaluation with a Lattice QCD code





www.xcalablemp.org

#### Directive-based PGAS extension for Fortran and C

- Proposed by XMP Spec. WG of PC Cluster Consortium.
- Ver. 1.3 spec. to be published soon.
- Now Ver. 2.0 on the table.
- Adopted by Oakforest-PACS and post-K in Japan. / Data Mapping

# Supports two parallelization models:

- Global-view (with HPF-like data/work mapping directives)
- Local-view (with coarray)

#### Allows mixture with MPI and/or OpenMP. Work Mapping





#### What's OpenACC?

- Directive-based extension to program accelerators for C/C++/Fortran
  - Developed by Cray, CAPS, PGI (NVIDIA)
- Based on the <u>offload</u> <u>model</u>
  - A host (CPU) offloads data/work to devices (accelerators, ACCs)
- Portability across OSs, CPUs, and ACCs.





#### **Basic Concepts of XACC**

• XACC = XMP + OpenACC + XACC Extensions

XMP directives	distributed-memory parallelism among nodes
OpenACC directives	accelerator(s) within a node
XACC Extensions	<ul><li>hierarchical parallelism</li><li>direct comm. among ACCs</li></ul>

• With XACC, XMP features (including coarray) could be applied to ACCs for productivity.



### **Execution Model of XACC**





#### **XACC Extensions**

- pragma xmp (for direct comm. among ACCs)
  - acc clause
- pragma acc (for hierarchical parallelism)
  - device directive
  - on\_device clause
  - layout clause
  - shadow clause
  - barrier\_device directive











## **Omni XcalableACC**

omni-compiler.org





#### **Evaluation**

#### • Evaluation environment

- HA-PACS@CCS, U. Tsukuba

Host	Intel Xeon-E5 2680v2 2.8GHz x 2 Sockets, DDR3 1866MHz 128GB
GPU	NVIDIA Tesla K20X (GDDR5 6GB) x 4 GPUs
Network	InfiniBand Mellanox Connect-X3 Dual-port QDR 8GB/s
Compiler	Intel 16.0.2, CUDA 7.5.18, Omni XACC compiler 1.1, MVAPICH2 2.1



### Lattice QCD Miniapp

- Developed by Dr. Hideo Matsufuru, KEK
- 850 lines in the original serial code written in C
- Parallelized in the directive-based global-view model



- Results:
  - Performance: 85-97% of MPI+CUDA and 96-101% of MPI+OpenACC
  - Productivity: Delta-SLOC is 20 % of MPI+CUDA and 70% of MPI+OpenACC



	MPI+CUDA	MPI+OpenACC	XACC
Delta-SLOC	832	223	160
Add	322	160	154
Delete	73	0	0
Modify	437	63	6





- A new programming language XcalableACC for accelerated clusters is proposed.
- XACC = XMP + OpenACC + XACC extensions
- We are developing the Omni XACC compiler.
- The evaluations showed high performance and productivity of XACC.



#### XMPT Tool I/F

- A tool API of XMP (including XACC)
- Objective:
  - providing a more generic tool API of XMP.
- Basic ideas inspired by OMPT (OpenMP Tools API)
  - event- and callback-based
- Planned targets:
  - MUST correctness checking tool (SPPEXA)
  - Score-P / Scalasca (JSC)
  - Extrae (BSC)
  - etc.



# **Basic Design of XMPT**

Callbacks are registered through xmpt set callback.



(\*xmpt bcast end)(...); -

myx bcast end(...);

void



### **Current Status of XMPT**

- Definition of the XMPT events is almost completed.
  - not yet for coarray accesses and XACC.
- A prototype implementation is available for evaluation.
  - on a branch of Omni XMP's repository on GitHub.
  - supports a part of the events.
- Todos:
  - evaluate the overhead.
  - complete implementation.
  - integration testing with MUST



#### **Correctness Checking of XMP Programs Using XMPT**

• Errors in the XMP directives

n = xmp\_node\_num() !\$xmp bcast (a(n<del>))</del> An error in collectiveness of the bcast directve

- Data races of coarrays
  - MUST could detect data races of coarrays using additional XMPT events on coarray accesses and image control statements.

Accesses of a coarray on multiple images in unordered segments could causes data race.





#### Summary

- XMPT is a generic tool API of XMP.
  - based on events and callbacks
- MUST could detect errors in XMP programs using XMPT.
  - errors in the XMP directives
  - data races of coarrays
- We are now implementing XMPT and will release it soon.



#### Syntax of XACC

- Basically, diagonal combination of XMP and OpenACC
  - XMP outer and OpenACC inner (first distribute among nodes, and then onto accelerators)
- Two kinds of directives (sentinels)
  - **#pragma xmp:** XMP directives and XACC extensions for direct comm.
  - **#pragma acc:** OpenACC directives and XACC extensions for hierarchical parallelism.



## *Omni XcalableMP*

omni-compiler.org

- A reference compiler being developed by RIKEN AICS and U. Tsukuba
- Open source
- Translator (source-to-source)
   + Runtime (based on MPI)
- A stable ver. 1.1.1 is available on GitHub.





#### **Issues**

- How to make a tool library that contains a strong/weak symbol?
  - libxmp.{a|so} contains xmp\_init and weak xmpt\_initialize.
  - libmyx.{a|so} contains strong xmpt\_initialize.