# *JCAHPCの新スーパーコンピュータ Oakforest-PACS*

## *～ 1年の運用を通してみた利用と成果～*

朴　泰祐

JCAHPC・運用支援部門長／筑波大学計算科学研究センター

資料協力:
JCAHPC
井戸村泰宏氏@原研
石川裕氏@AICS
中尾昌広氏@AICS

# アウトライン

- Oakforest-PACSの導入
- システム仕様
- 運用・システムソフトウェア
- アプリケーション
- 運用を通じて

# Oakforest-PACSの導入

# 9 大学スパコンセンター導入計画 (Feb. 2017)

JCAHPC

| Fiscal Year | 2014 | 2015 | 2016 | 2017 | 2018 | 2019 | 2020 | 2021 | 2022 | 2023 | 2024 | 2025 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hokkaido | HITACHI SR16000/M1 (172TF, 22TB)<br>Cloud System BS2000 (44TF, 14TB)<br>Data Science Cloud / Storage HA8000 / WOS7000 (10TF, 1.96PB) | | | | 3.2 PF (UCC + CFL/M) 0.96MW<br>0.3 PF (Cloud) 0.36MW | | | | | 30 PF (UCC + CFL-M) 2MW | | |
| Tohoku | NEC SX-9他 (60TF) | | SX-ACE(707TF,160TB, 655TB/s)<br>LX406e(31TF), Storage(4PB), 3D Vis, 2MW | | | | ~30PF, ~30PB/s Mem BW (CFL-D/CFL-M) ~3MW | | | | | |
| Tsukuba | HA-PACS (1166 TF)<br>COMA (PACS-IX) (1001 TF) | | | | PACS-X 10PF (TPF) 2MW | | | | | | | |
| Tokyo | Fujitsu FX10 (1PFlops, 150TiB, 408 TB/s),<br>Hitachi SR16K/M1 (54.9 TF, 10.9 TiB, 28.7 TB/s) | | | Oakforest-PACS 25 PF 4.2 MW | | Reedbush 1.8～1.9 PF 0.7 MW | 50+ PF (FAC) 3.5MW | | 100+ PF 4.5MW (UCC + TPF) | | 200+ PF (FAC) 6.5MW | |
| Tokyo Tech. | TSUBAME 2.5 (5.7 PF, 110+ TB, 1160 TB/s), 1.4MW | | | TSUBAME 2.5 (3～4 PF, extended) | TSUBAME 3.0 12 PF, UCC/TPF 2.0MW | | | | TSUBAME 4.0 (100+ PF, >10PB/s, ~2.0MW) | | | |
| Nagoya | FX10(90TF)<br>CX400(470TF)<br>SGI UV2000 (24TF, 20TiB) | Fujitsu FX100 (2.9PF, 81 TiB)<br>Fujitsu CX400 (774TF, 71TiB) | 2MW in total | | 50+ PF (FAC/UCC + CFL-M) up to 4MW | | | | 100+ PF (FAC/UCC+CFL-M) up to 4MW | | | |
| Kyoto | Cray: XE6 + GB8K + XC30 (983TF)<br>Cray XC30 (584TF) | | | | 5 PF(FAC/TPF) 1.5 MW | | | | 50-100+ PF (FAC/TPF + UCC) 1.8-2.4 MW | | | |
| Osaka | NEC SX-ACE (423TF) NEC Express5800 (22.4TF) | | | | 0.7-1PF (UCC) | | | | 3.2PB/s, 5-10Pflop/s, 1.0-1.5MW (CFL-M) | | 25.6 PB/s, 50-100Pflop/s,1.5-2.0MW | |
| Kyushu | HA8000 (712TF, 242 TB)<br>SR16000 (8.2TF, 6TB)<br>FX10 (272.4TF, 36 TB)<br>CX400 (966.2 TF, 183TB) | 2.0MW | | | 6.9PF+3.0PF ??MW<br>FX10 (90.8TFLOPS) | | | | 100-150 PF (FAC/TPF + UCC/TPF) 3MW | | | |

K Computer (10PF, 13MW) ???? Post-K Computer (??. ??)

Power consumption indicates maximum of power supply (includes cooling facility)

# Oakforest-PACS (OFP)

- 2016年12月1日稼働開始
- 8,208 Intel Xeon/Phi (KNL), ピーク性能25PFLOPS
  - 富士通が構築
- **TOP 500初出　6位（国内1位），HPCG 3位（国内2位）**
- **最先端共同HPC 基盤施設(JCAHPC: Joint Center for Advanced High Performance Computing)**
  - 東京大学情報基盤センター
  - 筑波大学計算科学研究センター
    - 東京大学柏キャンパスの東京大学情報基盤センター内に、両機関の教職員が中心となって設計するスーパーコンピュータシステムを設置し，最先端の大規模高性能計算基盤を構築・運営するための組織
  - http://jcahpc.jp

# TOP500 list on Nov. 2017 (#50)

| # | Machine | Architecture | Country | Rmax (TFLOPS) | Rpeak (TFLOPS) | MFLOPS/W |
|---|---|---|---|---|---|---|
| 1 | TaihuLight, NSCW | MPP (Sunway, SW26010) | China | 93,014.6 | 125,435.9 | 6051.3 |
| 2 | Tianhe-2 (MilkyWay-2), NSCG | Cluster (NUDT, CPU + KNC) | China | 33,862.7 | 54,902.4 | 1901.5 |
| 3 | Piz Daint, CSCS | MPP (Cray, CPU + GPU) | Switzerland | 19,590.0 | 25,326.3 | 10398.0 |
| 4 | Gyoukou, JAMSTEC | MPP (Exascaler, PEZY-SC2) | Japan | 19,125.8 | 28,192.0 | 14167.3 |
| 5 | Titan, ORNL | MPP (Cray, CPU + GPU) | United States | 17,590.0 | 27,112.5 | 2142.8 |
| 6 | Sequoia, LLNL | MPP (IBM, BlueGene/Q) | United States | 17,173.2 | 20,132.7 | 2176.6 |
| 7 | Trinity, NNSA/ LABNL/SNL | MPP (Cray, KNL, Xeon) | United States | 14,137.3 | 43,902.6 | 3667.8 |
| 8 | Cori, NERSC-LBNL | MPP (Cray, KNL) | United States | 14,014 | 7 | 3556.7 |
| 9 | Oakforest-PACS, JCAHPC | Cluster (Fujitsu, KNL) | Japan | 13,554.6 | 25,004.9 | 4985.1 |
| 10 | K Computer, RIKEN AICS | MPP (Fujitsu) | Japan | 10,510.0 | 11,280.4 | 830.2 |

KNLシステムとしては最高の55%

赤字：新規またはJun.2017からのアップグレード

# Oakforest-PACS (OFP)


- ピーク性能25 PFLOPS
- 8208 KNL CPUs
- OmniPathによるFBB Fat-Tree
- HPL 13.55 PFLOPS
  2016/11:
  国内第1位
  世界第6位
  ⇒ 2位、9位 (2017/11)
- HPCG 0.385PFLOSP
  (2.8% of HPL)
  2016/11: 世界第3位
  ⇒ 6位
- Green500
  2016/11: 6位
  ⇒ 22位
- IO500
  2017/11: 1位

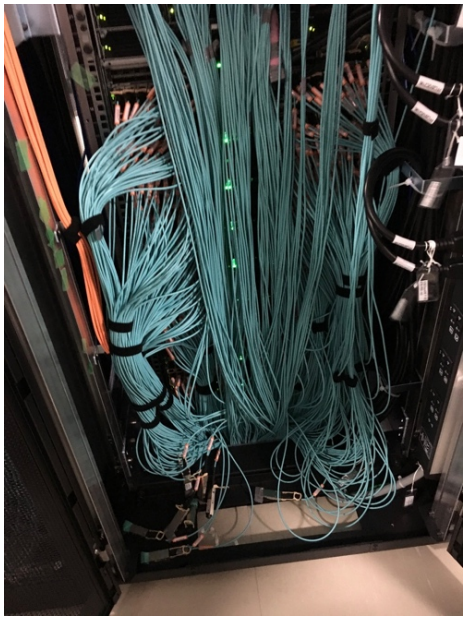
# 計算ノードとシャーシ

Computation node (Fujitsu PRIMERGY CX1640 M1)
with single chip Intel Xeon Phi (Knights Landing, 3+TFLOPS)
and Intel Omni-Path Architecture card (100Gbps)

Chassis with 8 nodes, 2U size
(Fujitsu PRIMERGY CX600 M1)

JCAHPC

# 水冷パイプ、リアパネル冷却、OPA、ファイルサーバ



# システム仕様

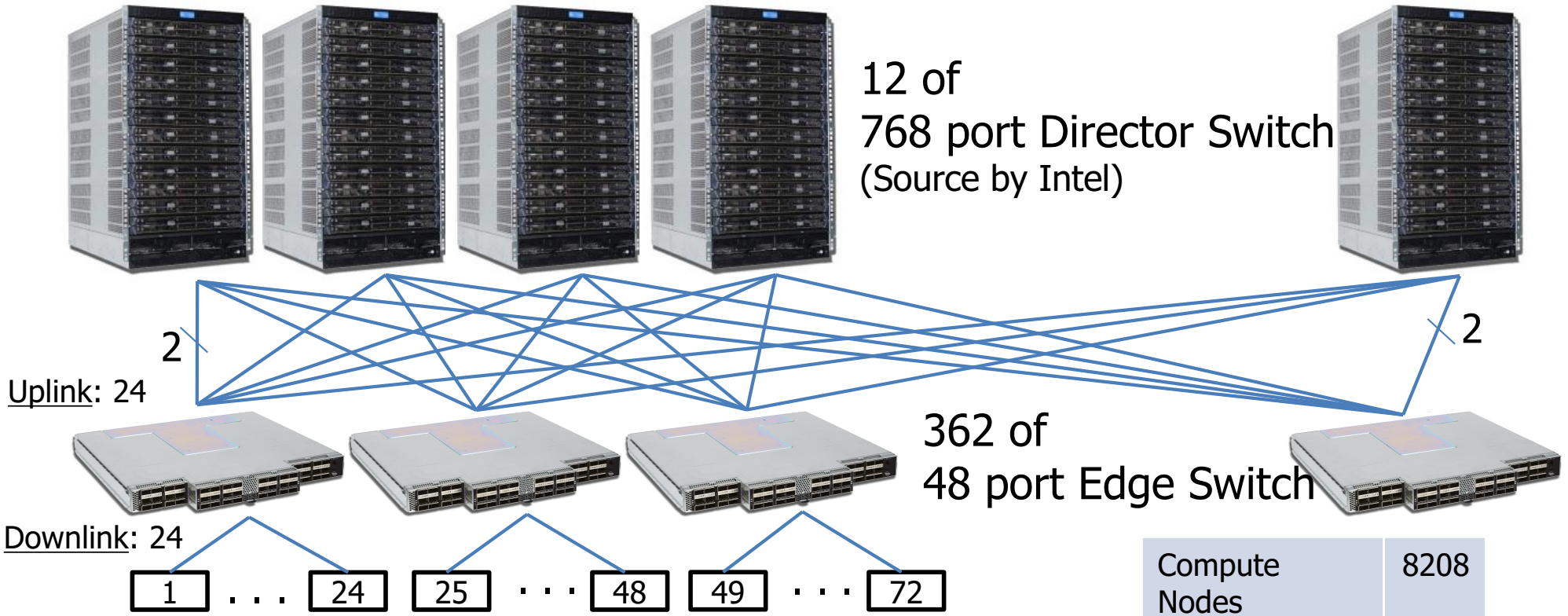
# Oakforest-PACSのシステム仕様

| | | | |
|---|---|---|---|
| Total peak performance | | | 25 PFLOPS |
| Total number of compute nodes | | | 8,208 |
| Compute node | Product | | Fujitsu Next-generation PRIMERGY server for HPC (PRIMERGY CX1640 M1) |
| | Processor | | Intel® Xeon Phi™ (Code name: Knights Landing), 68 cores |
| | Memory | MCDRAM | 16 GB, > 400 GB/sec (effective rate) |
| | | DDR4 | 96 GB, 115.2 GB/sec (DDR4-2400 x 6ch, peak rate) |
| Inter-connect | Product | | Intel® Omni-Path Architecture |
| | Link speed | | 100 Gbps |
| | Topology | | Fat-tree with (completely) full-bisection bandwidth |
| Login node | Product | | Fujitsu PRIMERGY RX2530 M2 server |
| | # of servers | | 20 |
| | Processor | | Intel Xeon E5-2690v4 (2.6 GHz 14 core x 2 socket) |
| | Memory | | 256 GB, 153 GB/sec (DDR4-2400 x 4ch x 2 socket) |

# Oakforest-PACSのI/O仕様

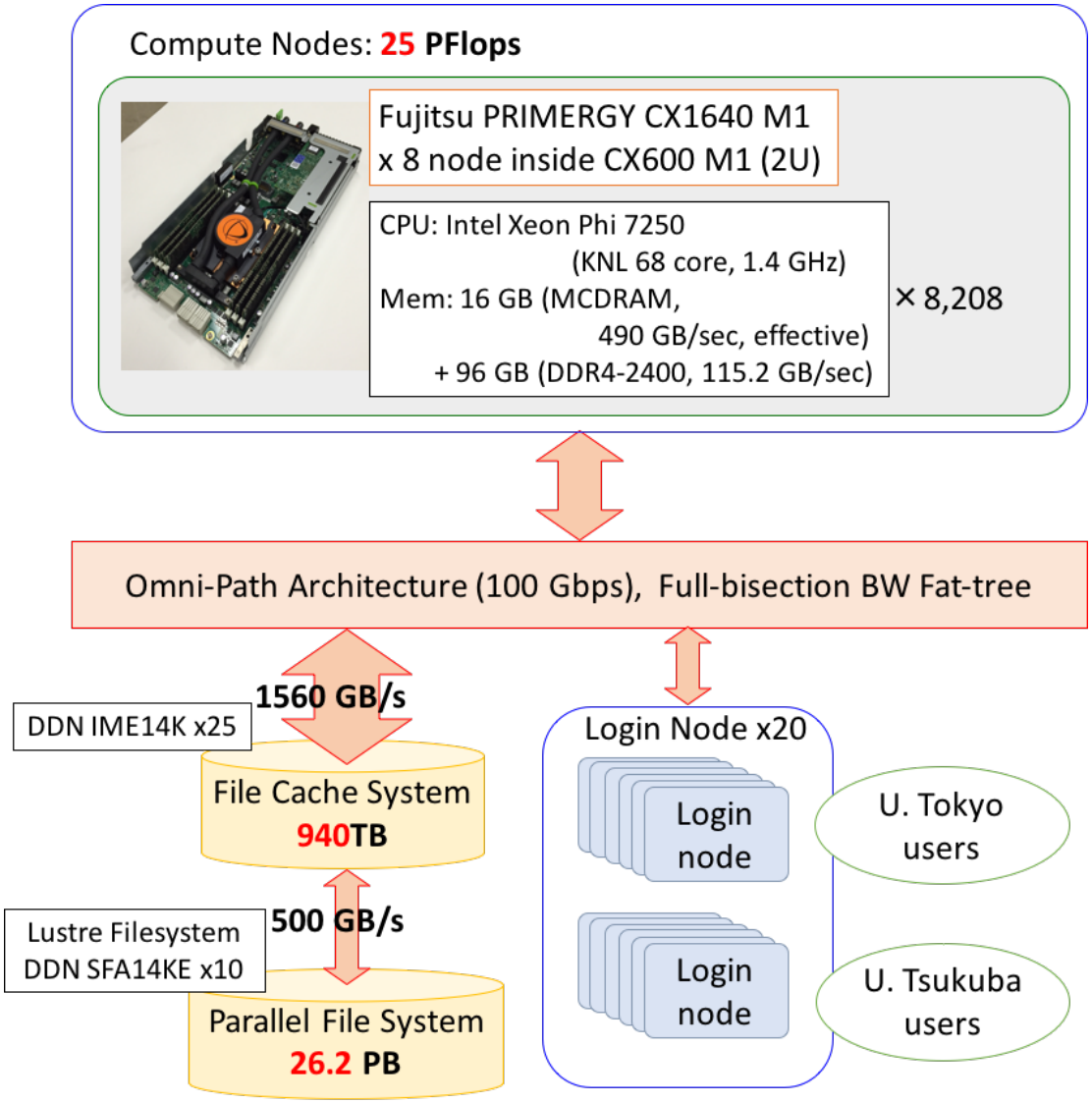
| | | | |
|---|---|---|---|
| Parallel File System | Type | | Lustre File System |
| | Total Capacity | | 26.2 PB |
| | Meta data | Product | DataDirect Networks MDS server + SFA7700X |
| | | # of MDS | 4 servers x 3 set |
| | | MDT | 7.7 TB (SAS SSD) x 3 set |
| | Object storage | Product | DataDirect Networks ES14K |
| | | # of OSS (Nodes) | 10 (20) |
| | | Aggregate BW | 500 GB/sec |
| Fast File Cache System | Type | | Burst Buffer, Infinite Memory Engine (by DDN) |
| | Total capacity | | 940 TB (NVMe SSD, including parity data by erasure coding) |
| | Product | | DataDirect Networks IME14K |
| | # of servers (Nodes) | | 25 (50) |
| | Aggregate BW | | 1,560 GB/sec |

# Intel® Omni-Path Architectureによる フルバイセクションバンド幅の相互結合網




| | |
|---|---|
| Compute Nodes | 8208 |
| Login Nodes | 20 |
| Parallel FS | 64 |
| IME | 200 |
| Mgmt, etc. | 8 |
| **Total** | **8600** |

# 利用者から見たシステム

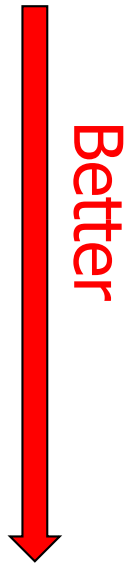


# Oakforest-PACSの特徴

- 最先端のHPC向けメニーコアCPUの利用
  - 極めて高い並列性（ノード内，ノード間）を持つHPCアプリケーションが主要ターゲット（3TFLOPS）
  - 高度なチューニングにより性能を大幅に向上可能
  - 初期プログラム移植の容易さ（OpenMP+MPI）
- 最先端の高性能相互結合網をfull-bisectionバンド幅で装備
  - 100Gbpsのリンク速度
  - ジョブスケジューラによるノード配置の自由度が高い
  - ノード位置にかかわらず全てのファイルを高速にアクセス可能
  - 超並列アプリケーションを容易に実行可能
- 高性能並列ファイルシステムとバーストバッファ
  - Lustreによる高並列・高性能アクセス（500GB/s）
  - ファイルキャッシュ（バーストバッファ）による1TB/s越えの超高速アクセス

# JPY (=Watt)/GFLOPS 比

| System | JPY/GFLOPS |
|---|---|
| Oakleaf/Oakbridge-FX (Fujitsu)<br>(Fujitsu PRIMEHPC FX10) | 125 |
| Reedbush-U (SGI)<br>(Intel BDW) | 62.0 |
| Reedbush-H (SGI)<br>(Intel BDW+NVIDIA P100) | 17.1 |
| Oakforest-PACS (Fujitsu)<br>(Intel Xeon Phi/Knights Landing) | 16.5 |



# 運用・システムソフトウェア

# 運用状況

- 2016/12~2017/3 試験運用（無償）
  - システム安定稼働チェック
  - 機能・性能の確認
  - （特別）大規模HPCチャレンジ：GBP
- 2017/4~ 公開運用
  - HPCI, 各大学の個別運用プログラム
  - 実運用だがユーザによってはここでチューニング開始
  - 試験運用で実績を積んだグループはジャンプスタート（素粒子、光物性等）
  - 月末のメンテナンス前に大規模HPCチャレンジとして約24時間の全系占有利用
- 稼働状況
  - 実稼働率（メンテナンス、停電等を除く）：ほぼ100%
  - 利用率：45~55%程度（後述）

# 運用：KNLのメモリモード

- メモリモード
  - Cache:Flat = 50:50 (4096+4096 nodes)
  - 様子を見て比率を変える予定だが現状ではバランスしている
- 動的メモリモード変更
  - オンデマンドでCache:Flatの比率を緩やかに動的変更することを計画中
  - ノードのリブートが予想より時間がかかるため様子見
- 大規模ジョブ（最大2048ノード）が通常利用可能
  - メモリモードの関係もあり、リソース確保が難しい
  - キューイングの自由度に制限（fair shareを実行していない）
  - 利用率の抑制原因なのでは？⇒ さらに解析・改善が必要

# Oakforest-PACS のソフトウェア

- OS: Red Hat Enterprise Linux (ログインノード)、
  CentOS および McKernel (計算ノード、切替可能)
  - McKernel: 理研AICSで開発中のメニーコア向けOS
    - Linuxに比べ軽量、ユーザプログラムに与える影響なし
    - ポスト京コンピュータにも搭載される予定。
- コンパイラ：GCC, Intel Compiler, XcalableMP
  - XcalableMP: 理研AICSと筑波大で共同開発中の並列プログラミング言語
    - CやFortranで記述されたコードに指示文を加えることで、性能の高い並列アプリケーションを簡易に開発することができる。
- ライブラリ・アプリケーション： オープンソースソフトウェア
  - ppOpen-HPC, OpenFOAM, ABINIT-MP, PHASE system, FrontFlow/blue ,
    LAPACK, ScaLAPACK, PETSc, METIS, SuperLU etc.

# McKernel on OFP

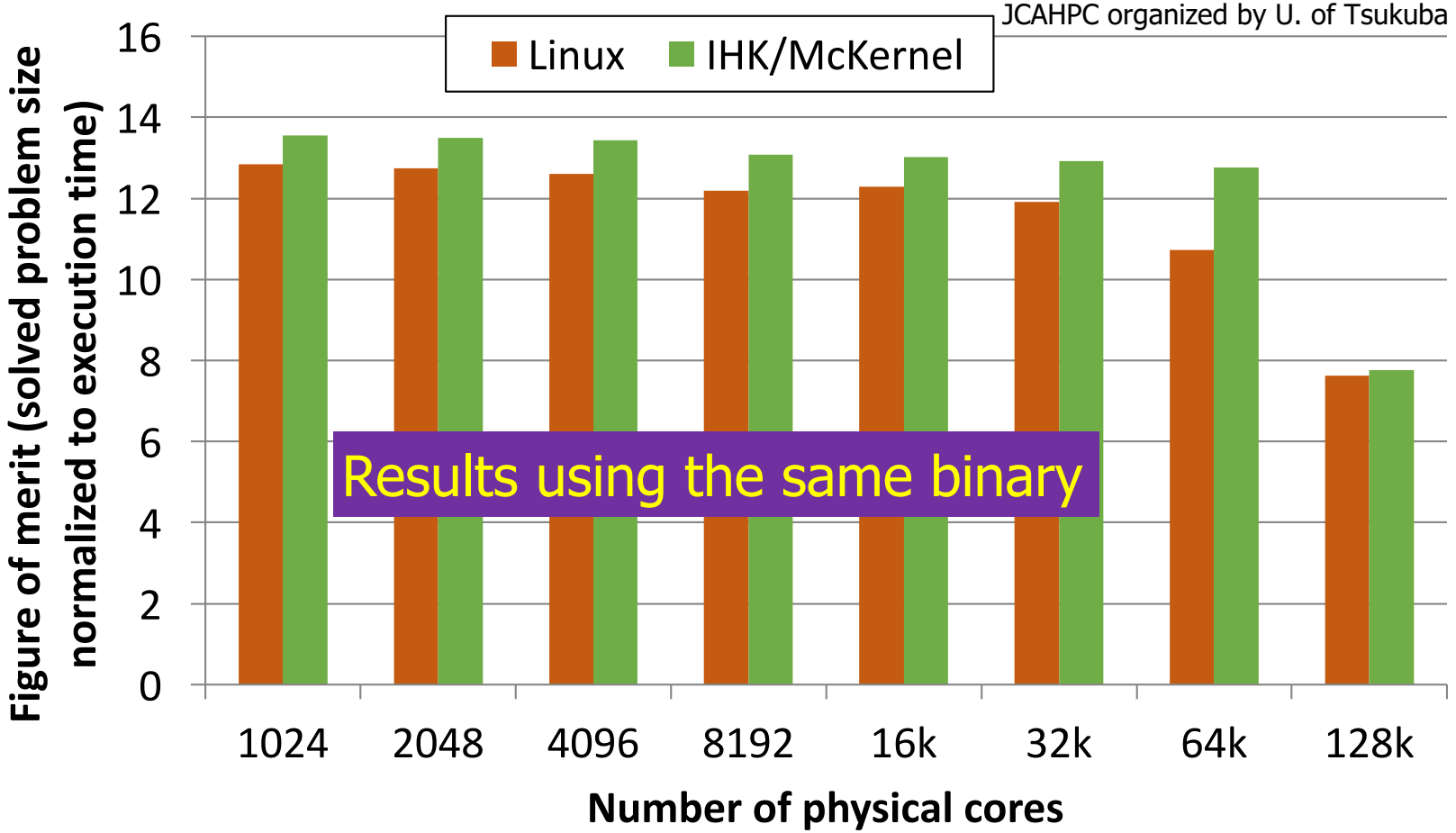
- McKernel on OFPの開発状況
  - 基本部分は動作しており運用に向けて準備中。
  - Linux only modeからMcKernel modeに移行する際に時間がかかる場合がありバッチジョブシステム側でジョブをKillしてしまうが、その対策方法がわかったのでその対策を施したときの影響を調査中。
    ⇒ MCDRAM上でクリーンな連続領域を確保するのに時間を要する
    ⇒ MCDRAMを Hot-Pluggable Memoryとすることで解決する見込み
  - 別途、mcexecが不具合を起こす場合があり、調査中。
  - 上記が済んで大規模でのアプリ実行動作を確認した後、一般に運用に向けての最終検討を行う予定。
- McKernel on OFPの意義
  - OFP上でのアプリケーション性能向上
  - ポスト「京」に向けてのMcKernel開発（理研AICS with 東大）
  - ユーザにMcKernelに慣れてもらう⇒ポスト「京」

# McKernel評価：GeoFEM (University of Tokyo)

(slide courtesy by Y. Ishikawa)

- ICCG with Additive Schwartz Domain Decomposition - weak scaling
- Up to 18% improvement

Acknowledgement: Kengo Nakajima, University of Tokyo, for providing GeoFEM. This result is on Oakforest-PACS supercomputer, 25 PF in peak, at JCAHPC organized by U. of Tsukuba and U. of Tokyo

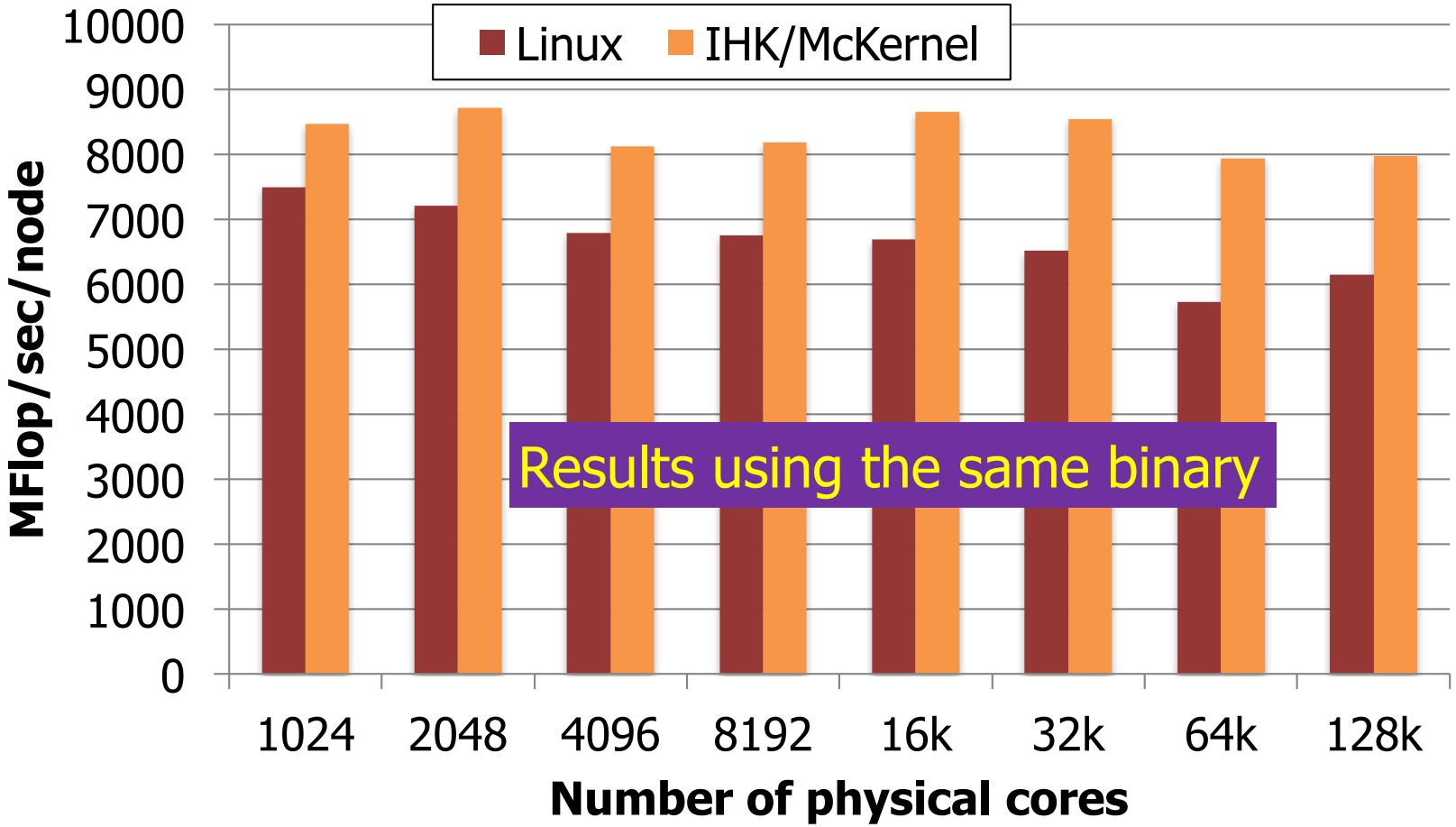


# McKernel評価：CCS-QCD (University of Tsukuba)

(slide courtesy by Y. Ishikawa)

- Lattice quantum chromodynamics code - weak scaling
- Up to 38% improvement

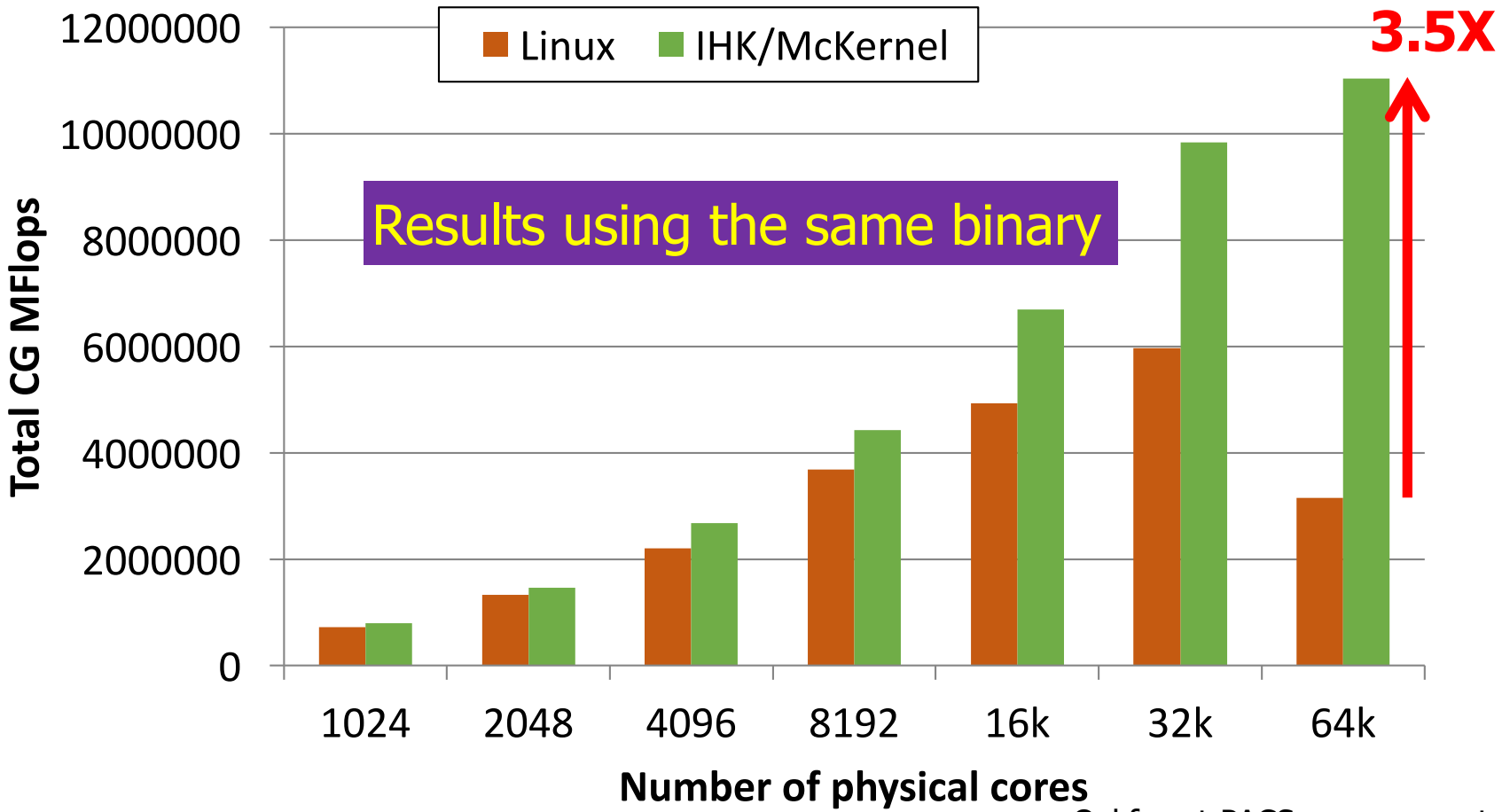
Acknowledgement: Ken'ichi Ishikawa, Hiroshima University, providing CCS-QCD. This result is on Oakforest-PACS supercomputer, 25 PF in peak, at JCAHPC organized by U. of Tsukuba and U. of Tokyo



(slide courtesy by Y. Ishikawa)

# McKernel評価：miniFE (CORAL benchmark suite)

- Conjugate gradient - strong scaling
- Up to 3.5X improvement (Linux falls over.. )




Oakforest-PACS supercomputer, 25 PF in peak, at JCAHPC organized by U. of Tsukuba and U. of Tokyo

# XcalableMP (XMP)

- XMP on OFP
  - 理研AICSと筑波大の共同研究で実装・評価中
  - OFPのノード間MPI通信とノード内MPI+OpenMPでの性能評価
- QCDミニアプリでの性能評価（by 中尾昌広氏）
  - ポスト「京」重点領域研究アプリ (Fibre) の一つ
  - 4次元格子上のステンシル計算、倍精度複素数演算
  - 4次元隣接通信とAllreduce通信
  - ここでは2次元プロセス配列に4次元格子をマッピング
  - OFPの最大256ノードを利用（64スレッド／ノード）

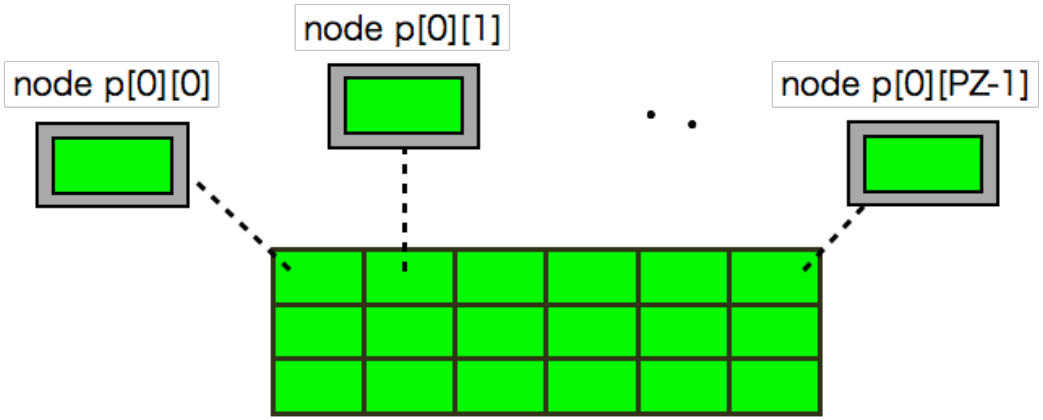
# 分散配列の定義と袖交換

(slide courtesy by M. Nakao)

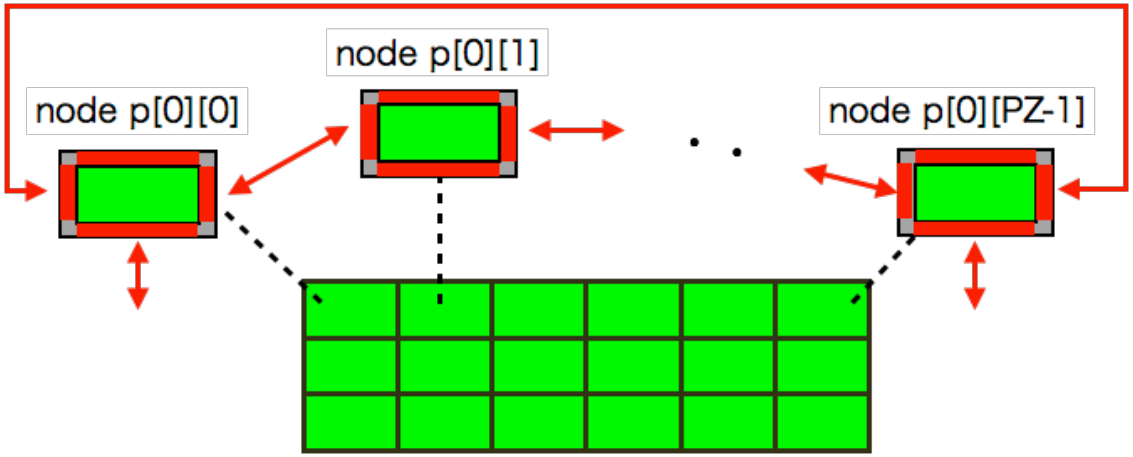
```
Quark_t v[NT][NZ][NY][NX];
#pragma xmp template t[NT][NZ]
#pragma xmp nodes p[PT][PZ]
#pragma xmp distribute t[block][block] onto p
#pragma xmp align v[i][j][*][*] with t[i][j]
#pragma xmp shadow v[1][1][0][0]
```

4次元の配列データを2次元ブロック分散する

幅1の袖を分散配列に追加する




```
#pragma xmp reflect(v) width(/periodic/1,/periodic/1,0,0) orthogonal
WD(..., v); // ステンシル計算
```



# forループ文の分割

(slide courtesy by M. Nakao)

```
#pragma xmp reflect(v) width(/periodic/1,/periodic/1,0,0) orthogonal
WD(..., v); // ステンシル計算
```

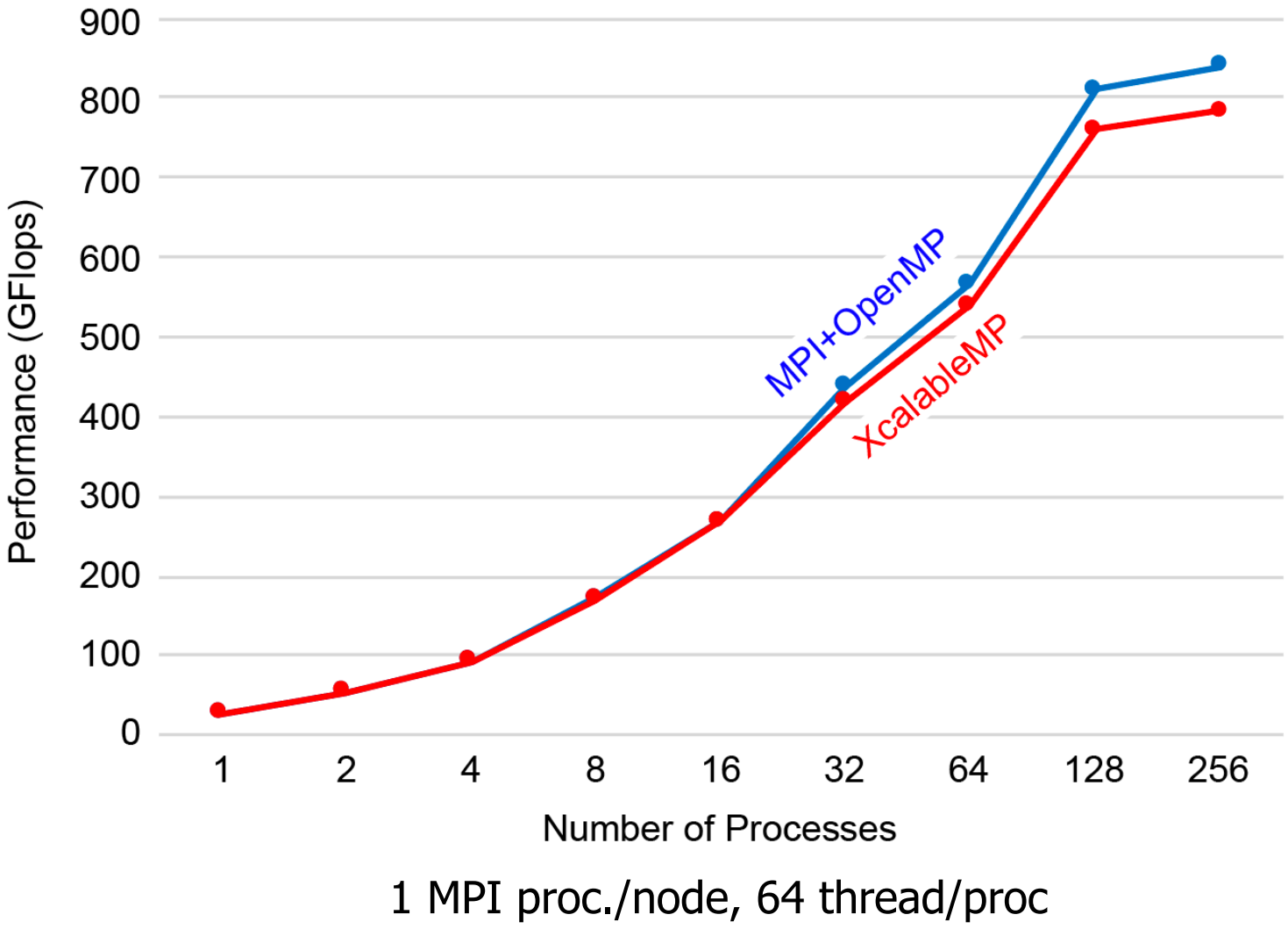
```
void WD(…, Quark_t v)
{
#pragma xmp loop (it,iz) on t[it][iz]
#pragma omp parallel for collapse(4)
  for(it = 0; it < NT; it++){
    for(iz = 0; iz < NZ; iz++){
      for(iy = 0; iy < NY; iy++){
        for(ix = 0; ix < NX; ix++){
          …
```

XMP指示文で各プロセスに分割したfor文に対して，
さらにOpenMP指示文でスレッド分割を行う．

# 性能評価（計算ノード内は64スレッド）

(slide courtesy by M. Nakao)

問題サイズは32x32x32x32

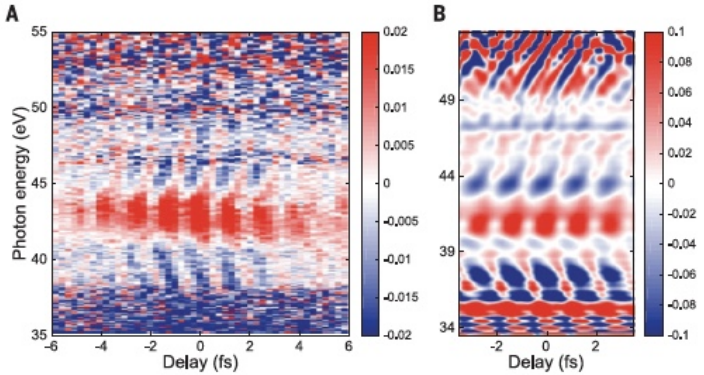


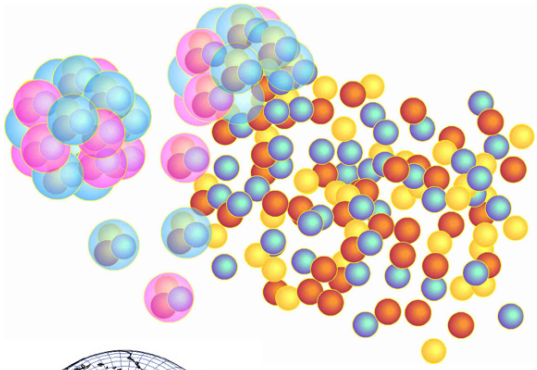

1 MPI proc./node, 64 thread/proc

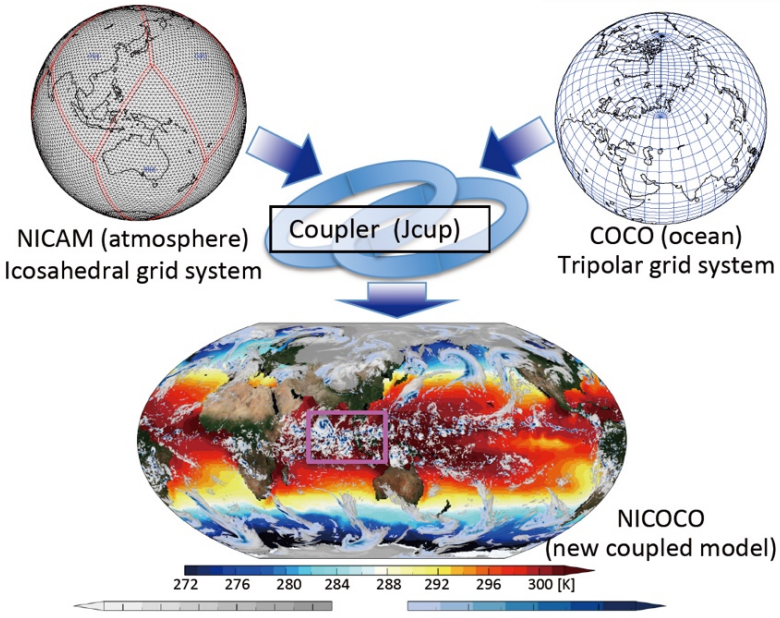
# Oakforest-PACSのアプリケーション

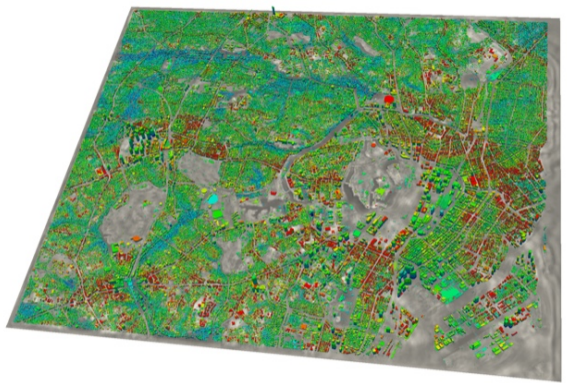
# Oakforest-PACS：代表的アプリケーション

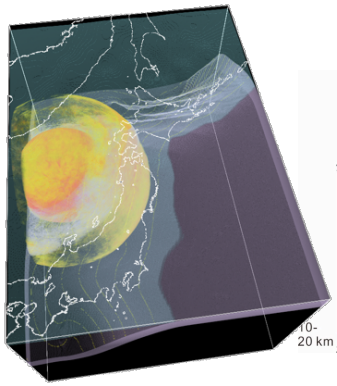
- **SALMON/ARTED**
  - 電子動力学
- **Lattice QCD**
  - 格子量子色力学
- **NICAM-COCO**
  - 全地球大気・海洋連成
- **GHYDRA**
  - 三次元地盤震動（FEM）
- **Seism3D/OpenSWPC**
  - 三次元広域波動伝搬（FDM）

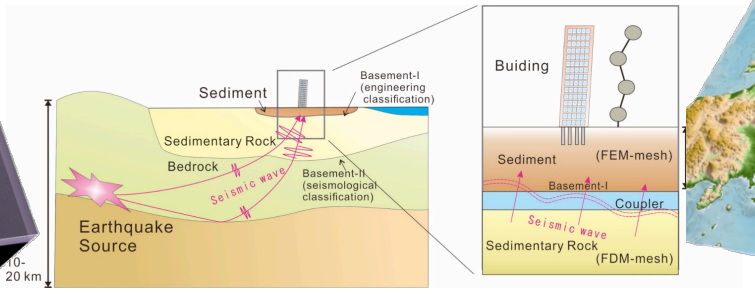


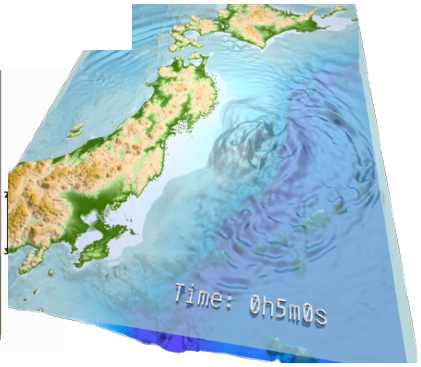







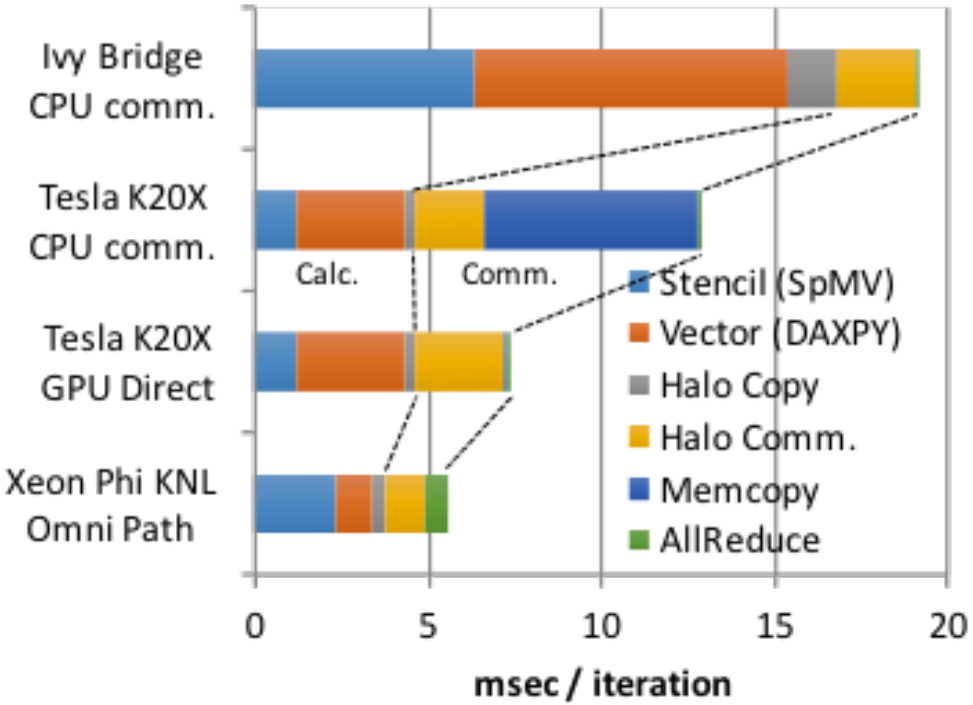
# アプリケーション紹介

- GT5D (PI: 井戸村泰宏@原研)
  - 核融合トカマクシミュレーション
  - 5次元自由度の超並列・超大規模シミュレーション
  - 「京」コンピュータ等での実績
  - 「第1回OFP利活用報告会」における井戸村氏の発表から抜粋
- ARTED（PI: 矢花一浩@筑波大）
  - 光物性等のための電子動力学法シミュレーション
  - 超並列波数空間・小規模実空間格子シミュレーション
  - 「京」コンピュータ、GPUクラスタ等での実績

# Many core optimization of GT5D kernel

(slide courtesy by Y. Idomura)

- Many core optimization of finite difference kernel [Asahi, SC15, IEEE-TPDS2017]
  - CPU: reuse stencil data on large shared cache loaded by other threads
  - MIC: optimized 2D thread mapping for distributed small L2 caches
  - GPU: physics based loop design to reuse registers and avoid warp divergence
- Multi-GPU/MIC optimization of Krylov (GCR) solver [Matsumoto, AESJ2016]
  - Direct communication dramatically improves parallel performance
  - Cost of the remaining communication is relatively large




Parallel performance with 16MPI processes
(GCR solver for 4D grids 128x128x32x128)

HA-PACS@Tsukuba Univ.
IvyBridge(224GFlops) x 2 / node
K20X(~1.3TFlops) x 4 / node
InfinibandQDR (8GB/s)

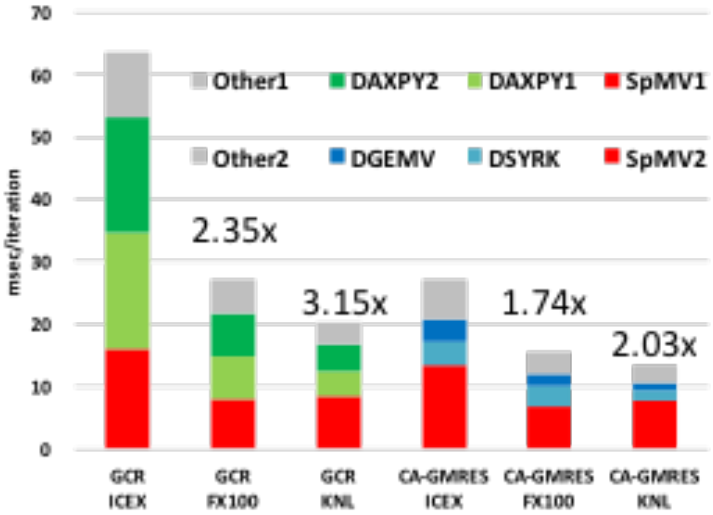
Oakforest-PACS@JCA
KNL(~3TFlops) x 1 /node
Intel Omni Path (12.5GB/s)

(slide courtesy by Y. Idomura)

# Performance of GCR/CA-GMRES on ICEX/FX100/KNL

[Idomura et al., ScalA@SC17]

Krylov solvers@2 nodes (GT5D: 160x160x32x96, $m_i/m_e$=1836)




| | | ICEX | | FX100 | | | KNL | | |
|---|---|---|---|---|---|---|---|---|---|
| Kernel | Flop /Byte | Gflops /node | Roofline ratio | Gflops /node | Roofline ratio | ICEX ratio | Gflops /node | Roofline ratio | ICEX ratio |
| SpMV1 | 1.03 | 81.1 | 0.76 | 162.7 | 0.68 | 2.01 | 153.2 | 0.35 | 1.89 |
| DAXPY1 | 0.08 | 8.4 | 0.88 | 22.9 | 0.93 | 2.72 | 38.1 | 0.95 | 4.53 |
| DAXPY2 | 0.21 | 21.4 | 0.90 | 57.2 | 0.97 | 2.68 | 95.4 | 0.97 | 4.47 |
| SpMV2 | 1.29 | 90.0 | 0.69 | 178.0 | 0.63 | 1.98 | 157.7 | 0.30 | 1.75 |
| DSYRK | 3.00 | 264.2 | 1.03 | 303.1 | 0.63 | 1.15 | 547.7 | 0.55 | 2.07 |
| DGEMV | 0.23 | 23.8 | 0.89 | 43.6 | 0.66 | 1.83 | 76.7 | 0.69 | 3.22 |

- Performance comparison between ICEX and FX100/KNL
  - GCR(SpMV,DAXPY) x2.35/x3.15
  - CA-GMRES(SpMV,DSYRK,DGEMV) x1.74/x2.03

  →Memory intensive kernels get larger performance gain

- Performance comparison between GCR and CA-GMRES
  - CA-GMRES is x2.34/x1.74/x1.51 faster than GCR on ICEX/FX100/KNL

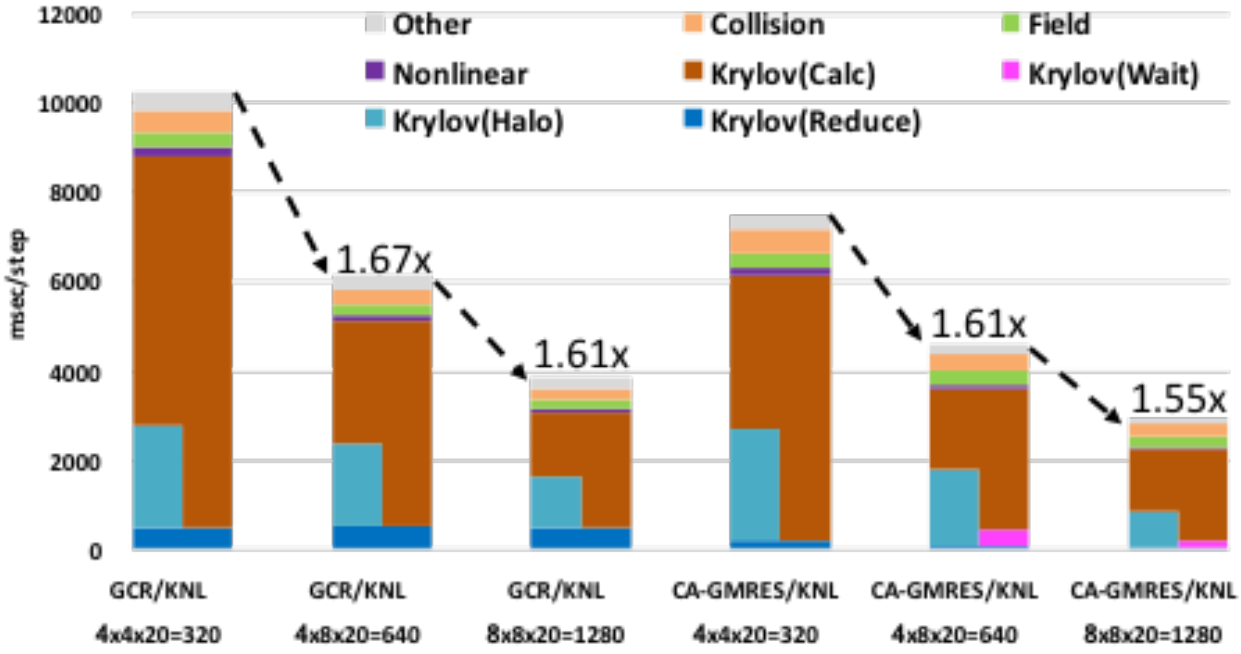
  →Compute intensive CA-Krylov methods are suitable for low-B/F machines

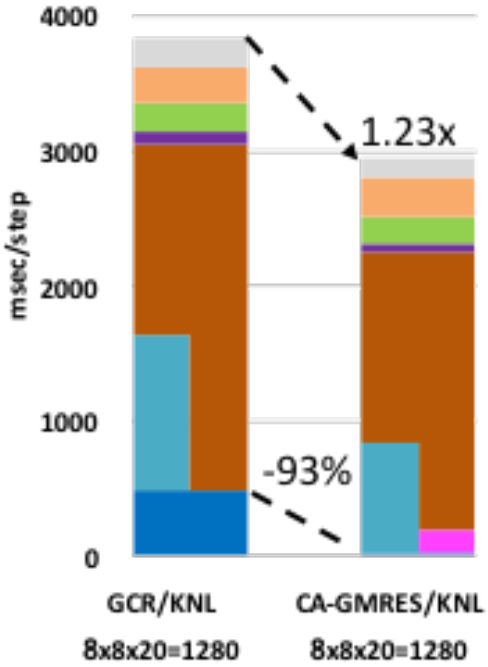
# Strong scaling of GT5D on Oakforest-PACS

(slide courtesy by Y. Idomura)

[Idomura et al., ScalA@SC17]

Strong scaling from 320-1280 nodes (GT5D: 320x320x32x96x20, $m_i/m_e$=1836)

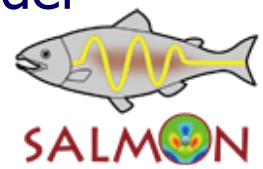


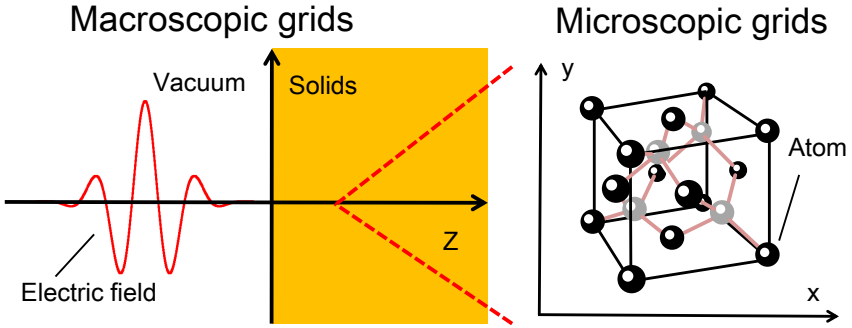

- Good strong scaling up to 1280 nodes (87k cores)
- Communication overlap hides almost all halo communication cost
- Cost of All_Reduce is reduced from 12.5%(GCR) to 1.0%(CA-GMRES)

→Promising feature for Exa-scale computing

# Xeon Phi tuning on ARTED (with Y. Hirokawa under collaboration with Prof. K. Yabana, CCS) → SALMON now




- ARTED – Ab-initio Real-Time Electron Dynamics simulator
- Multi-scale simulator based on RTRSDFT (Real-Time Real-Space Density Functional Theory) developed in CCS, U. Tsukuba to be used for Electron Dynamics Simulation
  - RSDFT : basic status of electron (no movement of electron)
  - RTRSDFT : electron state under external force
- In RTRSDFT, RSDFT is used for ground state
  - RSDFT : large scale simulation with 1000～10000 atoms (ex. K-Computer)
  - RTRSDFT : calculate a number of unit-cells with 10 ~ 100 atoms




RSDFT : Real-Space Density Functional Theory
RTRSDFT : Real-Time RSDFT

# Stencil code (original)

```
integer, intent(in) :: IDX(-4:4,NL),IDY(-4:4,NL),IDZ(-4:4,NL)

! NL = NLx*NLy*NLz
do i=0,NL-1
  ! x-computation
  v(1)=Cx(1)*(E(IDX(1,i))+E(IDX(-1,i))) ...
  w(1)=Dx(1)*(E(IDX(1,i))-E(IDX(-1,i))) ...

  ! y-computation
  v(2)=Cy(1)*(E(IDY(1,i))+E(IDY(-1,i))) ...
  w(2)=Dy(1)*(E(IDY(1,i))-E(IDY(-1,i))) ...

  ! z-computation
  v(3)=Cz(1)*(E(IDZ(1,i))+E(IDZ(-1,i))) ...
  w(3)=Dz(1)*(E(IDZ(1,i))-E(IDZ(-1,i))) ...

  ! update
  F(i) = B(i)*E(i) + A*E(i) - 0.5d0*(v(1)+v(2)+v(3)) - zI*(w(1)+w(2)+w(3))
end do
```

Original code just compiled on KNC with "-O3" option → less than 5% of peak!

# Stencil code (original)

```
integer, intent(in) :: IDX(-4:4,NL),IDY(-4:4,NL),IDZ(-4:4,NL)

 ! NL = NLx*NLy*NLz
 do i=0,NL-1
   ! x-computation
   v(1)=Cx(1)*(E(IDX(1,i))+E(IDX(-1,i))) ...
   w(1)=Dx(1)*(E(IDX(1,i))-E(IDX(-1,i))) ...

   ! y-computation
   v(2)=Cy(1)*(E(IDY(1,i))+E(IDY(-1,i))) ...
   w(2)=Dy(1)*(E(IDY(1,i))-E(IDY(-1,i))) ...

   ! z-computation
   v(3)=Cz(1)*(E(IDZ(1,i))+E(IDZ(-1,i))) ...
   w(3)=Dz(1)*(E(IDZ(1,i))-E(IDZ(-1,i))) ...

   ! update
   F(i) = B(i)*E(i) + A*E(i) - 0.5d0*(v(1)+v(2)+v(3)) - zI*(w(1)+w(2)+w(3))
 end do
```

indirect index array:
keeping nearest neighbor index

write-only in the loop

vector length=4, for DP-complex vector calculation-> 512-bit AVX fittable

# For automatic vectorization

```
real(8),   intent(in)  :: B(0:NLz-1,0:NLy-1,0:NLx-1)
complex(8),intent(in)  :: E(0:NLz-1,0:NLy-1,0:NLx-1)
complex(8),intent(out) :: F(0:NLz-1,0:NLy-1,0:NLx-1)
```

convet to 3D array

```
#define IDX(dt) iz,iy,iand(ix+(dt)+NLx,NLx-1)
#define IDY(dt) iz,iand(iy+(dt)+NLy,NLy-1),ix
#define IDZ(dt) iand(iz+(dt)+NLz,NLz-1),iy,ix
```

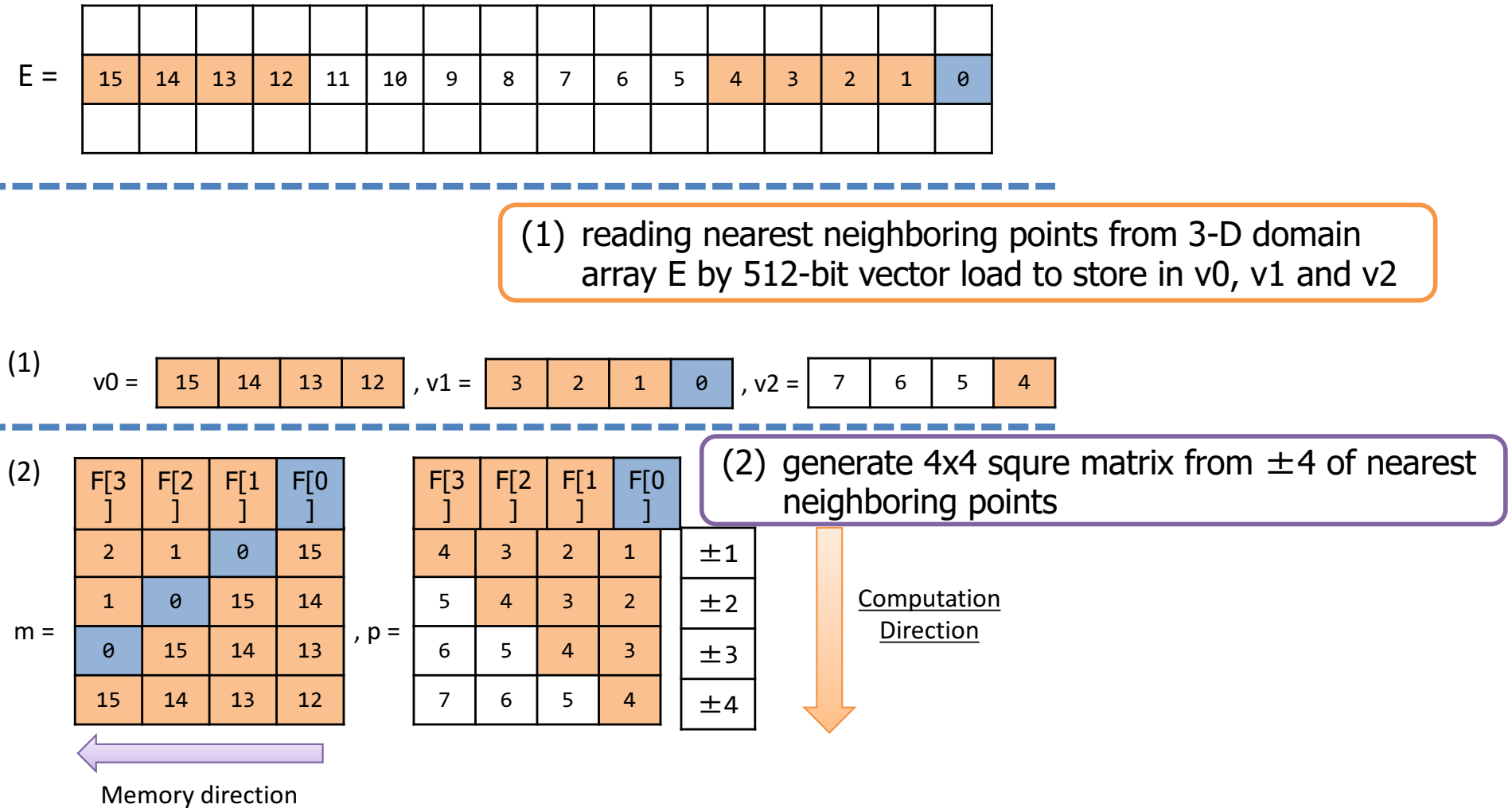
direct index calculation

```
do ix=0,NLx-1
do iy=0,NLy-1
!dir$ vector nontemporal(F)
do iz=0,NLz-1
  v=0; w=0
  ! z-computation
  v=v+Cz(1)*(E(IDZ(1))+E(IDZ(-1))) ...
  w=w+Dz(1)*(E(IDZ(1))-E(IDZ(-1))) ...
  ! y-computation
  ! x-computation
  F(iz,iy,ix) = B(iz,iy,ix)*E(iz,iy,ix) &
  &           + A          *E(iz,iy,ix) &
  &           - 0.5d0*v - zI*w
end do
end do
end do
```

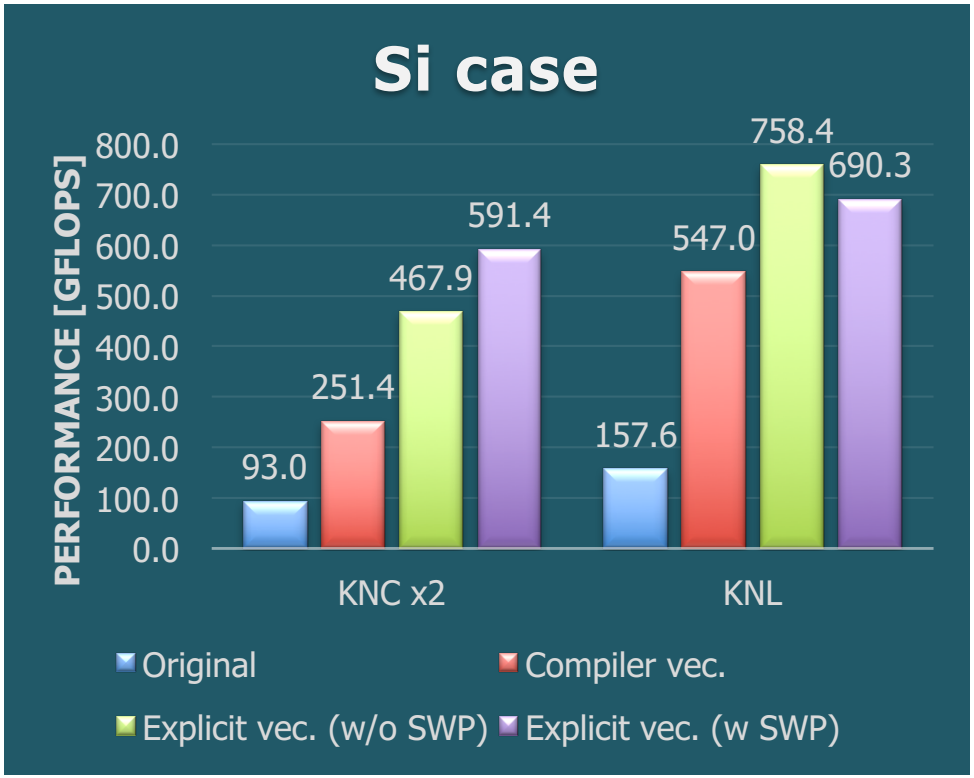
non-temporal write without cache

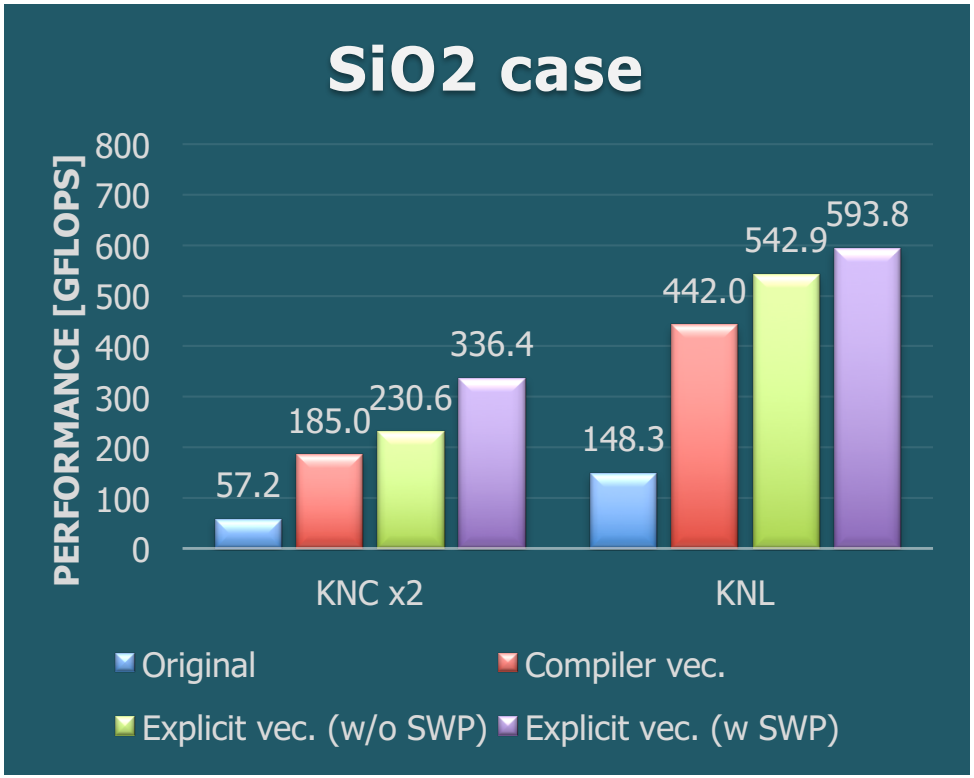
reordering according to memory access sequence

# Hand vectorization – unit-stride memory access optimization (how to utilize AVX-512 SIMD load and operation)



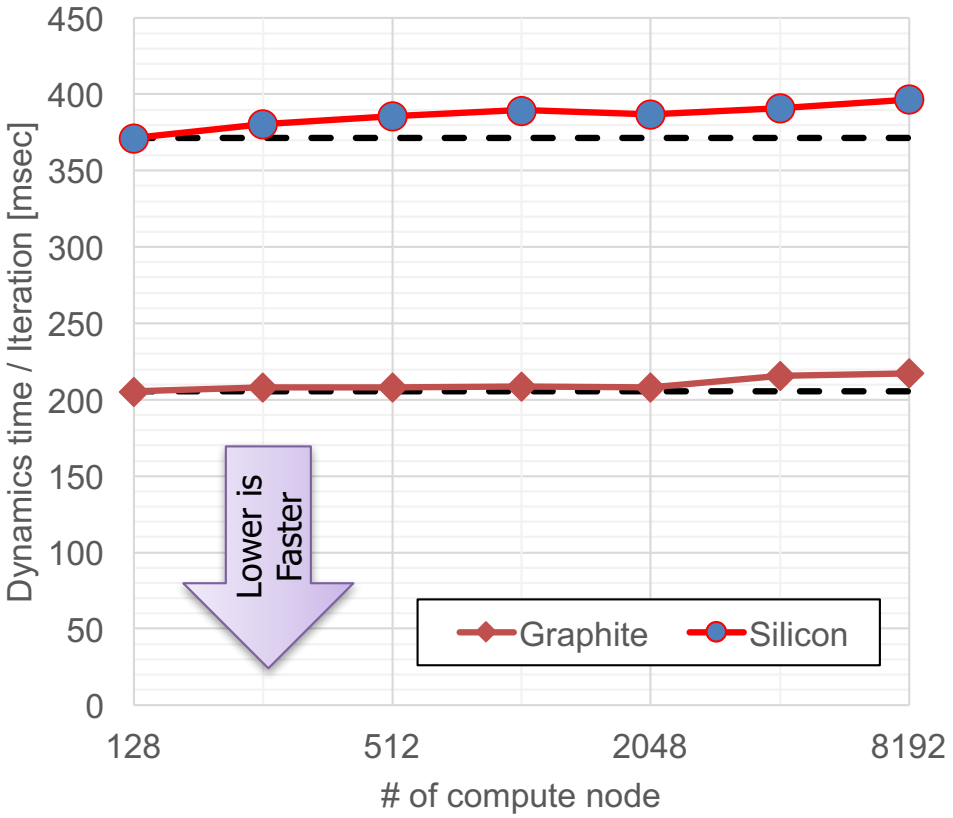
# Stencil computation (3D) performance

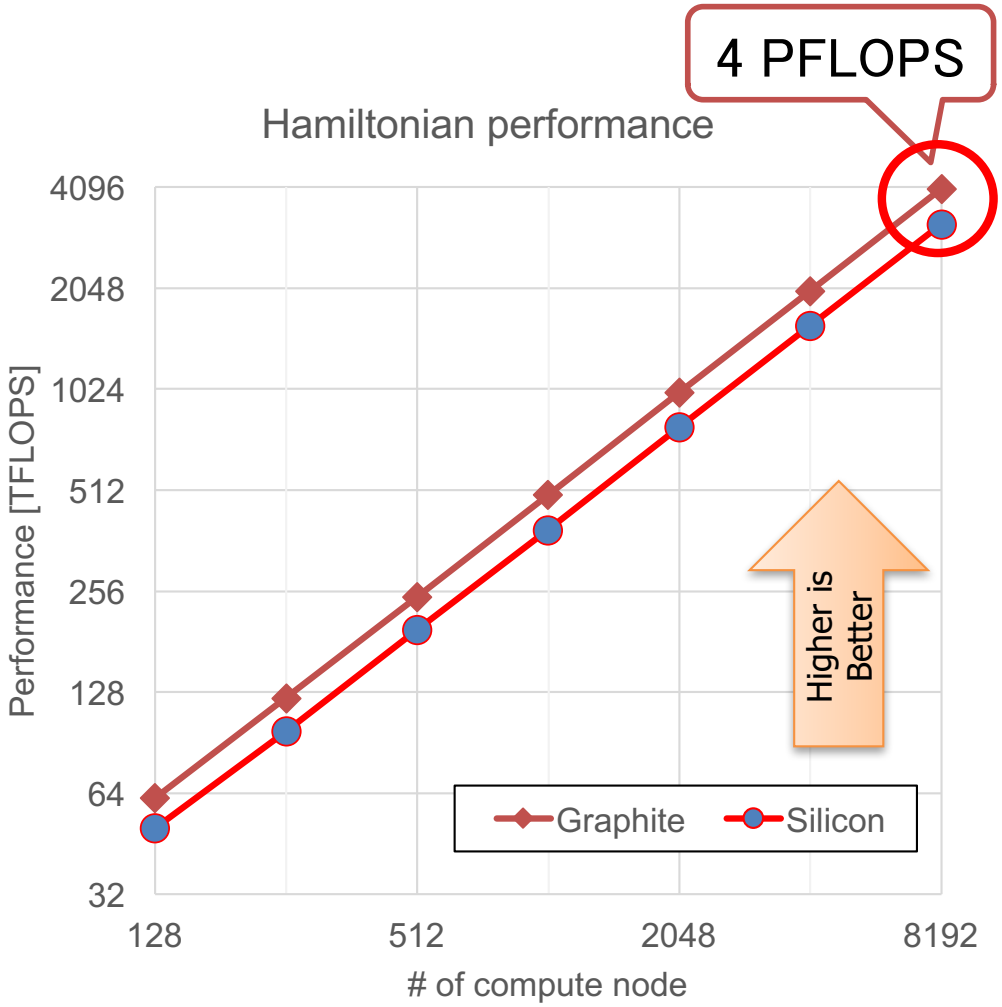



>2x faster than KNC (at maximum) -> up to 25% of theoretical peak of KNL

# Weak scaling on OFP full system



# 運用を通じて

# 運用を通じて

- １年間の運用（実運用としては8ヶ月）を通じて
  - システムは基本的に安定動作している。各部品の故障率については想定範囲内。
  - ユーザによっては大規模・大量ジョブで運用しているが、ジョブサイズは多岐に渡っている
  - KNLチューニングの度合いはまちまち？
  - KNLは温度環境の性能（turbo boost freq.) に対するsensitivityが高い？
- 運用上の懸案
  - Burst Buffer（ファイルキャッシュ）は性能は出ているがいくつかの制約があり試験運用を継続中⇒間もなく本運用に入る予定
  - Cache/Flatメモリモードの効率的制御はノード単位での高速・安定リブートが可能になれば実施したい
  - ジョブ利用率はまだ不足、スケジューリングパラメータ等のチューニングを要する

# OFPの位置付け

- 一般的に利用可能な国内リソースとしては最大性能
  - 大型ジョブをはじめとする各種・各サイズアプリケーションの資源としての利用
  - 大規模HPCチャレンジにより「京」を超えるリソースを提供
  - 今後、「京」が運用を停止し、ポスト「京」が出るまでのHPCI等における重要な資源となる
- メニーコア・アプリケーション開発
  - x86系科学技術計算向けメニーコア・アプリの開発プラットフォーム（AVX512等の命令に依存するコードはSkylake等にも引き継がれる）
  - McKernel、XcalableMP等の環境を提供
    ⇒ ポスト「京」への布石

# まとめ

- OFPは実運用に供されているスーパーコンピュータとしては依然国内最高性能で、HPCIをはじめとする東大・筑波大の様々な利用プログラムに活用されている
- 1年間の運用を通じ、システムは基本的に安定稼働しているが、Burst Bufferやquick rebootが難しい等の運用上の改善が望まれる
- スケジューリングパラメータをチューニングし、利用率を上げていきたい
- McKernel, XMPといったシステムソフトウェア開発はOFPの性能改善、プログラミング環境改善だけでなくポスト「京」の開発にもつながる
- 大口・大規模ユーザは性能チューニングをよく行っており、研究を強力に推進している
  ⇒ セミナーやワークショップを通じてノウハウを共有していきたい
- 一般運用・大規模HPCチャレンジを通じて計算科学・計算工学研究・教育に一層貢献していきたい