2022/4/22

第175回お試しアカウント付き並列プログラミング講習会

「スーパーコンピュータ超入門」

より進んだ利用に向けて

2022/4/18 v1.0

東京大学 THE UNIVERSITY OF TOKYO
東京大学情報基盤センター INFORMATION TECHNOLOGY CENTER, THE UNIVERSITY OF TOKYO

どのように使うか

- 一般利用
  - 研究代表者（大学、公的研究機関の方）
  - グループでの利用も可

➡ 研究実施に必要なトークン量を見積もる必要

システム	利用負担金 (年額，税込)	利用可能 ノード数	割当資源（トークン）	ディスク量 /work
Oakbridge-CX OBCX	**100,000円** /年 (1セット)	最大256 ノード	**8,640 ノード時間** （1ノード×360日相当）	4TB

＃　電気代相当の経費を負担

＃　一般利用の資源利用状況は show_token, show_quota コマンドで確認可

【企業の方は、企業利用制度により利用（詳細HP参照）】

利用予定プログラムで計算時間見積もり

OBCX の実機上でも、自分の研究室のマシンでも

- 担当講師の手元にあるプログラム

  自分で書いた分子動力学コード，MPI + OpenMP に対応

やりたいこと：

16,384,000 粒子　100 ナノ秒 = 10^8 ステップ　が必要

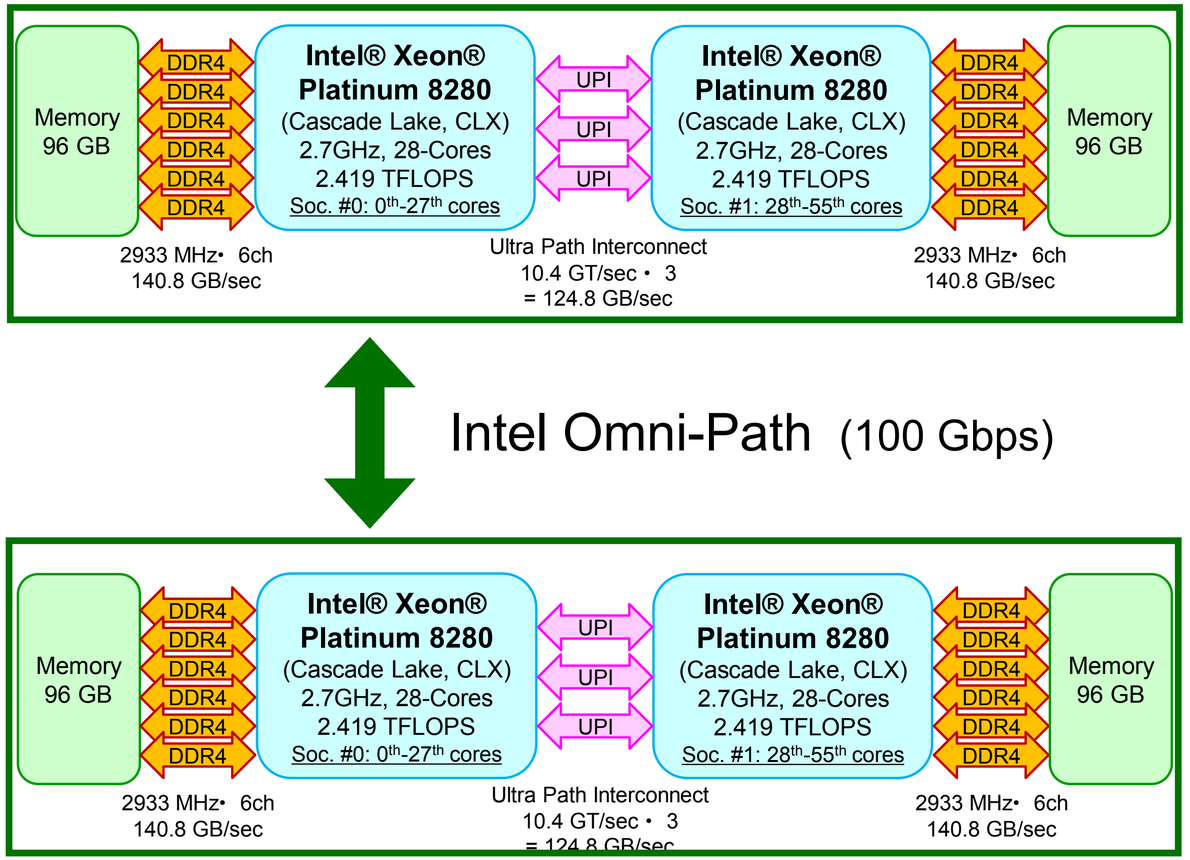
★OBCX 1ノードで実行，10^4 ステップから確認

MPI + OpenMP
ハイブリッド並列

OBCX 搭載のCPU
= Intel Xeon Platinum 8280 (CascadeLake)

1 CPU = 28 コア
OpenMP 7スレッド
x MPI 4プロセス

合計 2 ノード
OpenMP 7スレッド
x MPI 16プロセス



OpenMP

＝ 指示文挿入による並列化

プログラムの大幅な書換えは
しなくて良いが、ノード内のみ

MPI

＝ 通信の明示的指定

ノードまたぎ並列には必須

基本は 1 コアあたり
1スレッド, 1プロセス

実行結果

1ノード，56 コア　(8 MPI x 7 OpenMP)

32,000 粒子　　11.84 sec

```
Array Use Information:
 MMD:
  max of Nproc: 341003  (curr. lim=341936)
  max of celenum[cn]: 45  (curr. lim=62)
  max of nnlmmax[n]: 127  (curr. lim=149)
  average of cell list duration: 0.018844

Function Load
  514.293sec
     347.768sec  [67.6205%]  force()
  0.00851301sec  [0.00165529%]  dist_rand()
     6.53144sec  [1.26998%]  LFone()
     3.96262sec  [0.770499%]  LFtwo()
      87.233sec  [16.9617%]  comm_particles()
     15.6167sec  [3.03654%]     [array_sendrecv]
     53.7682sec  [10.4548%]     [comm_ispq]
     17.8437sec  [3.46957%]     [communicateparticles]
     67.4933sec  [13.1235%]  structclist()
     50.0715sec  [9.73599%]     [makennlist]
    0.358424sec  [0.0696925%]     [recvindexsearch]
```

- 単純に必要計算資源を見積もると…

  16,384,000 粒子　10^8 ステップ

  ➡　11.84 sec × (16,384,000 / 32,000) × (10^8 / 10^4)

  = 6.602 x 10^7 sec = 16389 hours

1 ノード換算で　トークン16400 ノード時間が必要 (約2セット)

まずは、1ノードで比較

10^4 ステップ

全て 1ノード，56 コア　(8 MPI x 7 OpenMP)

10^8 ステップ

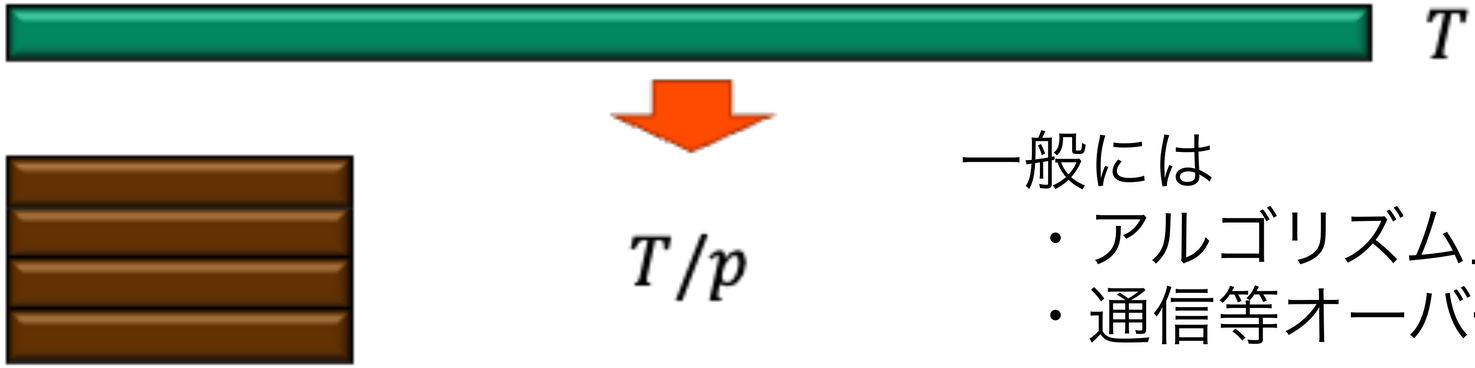
➡ 10000 時間 = 416日

もう少し詳しい並列性能評価の話

多くのアプリケーションは並列度を増やすと性能低下

目的：　逐次実行のプログラム ＝ 実行時間 T_s

p 台を用いて実行時間 T_pを T_s/p にしたい



一般には

・アルゴリズム上の困難

・通信等オーバーヘッド

並列化効率 $E_p = 100 \times (T_p/T_s) / p$

並列化後の実行時間短縮　$S_p = T_p/T_s$

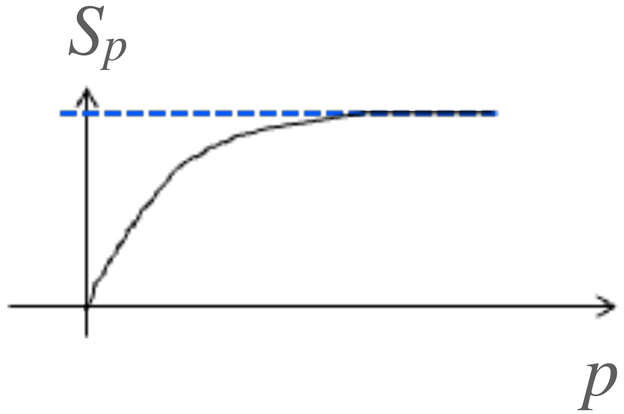
もう少し詳しい並列性能評価の話

アムダールの法則

- 逐次実行時間をT_sとし，このうち並列化できる割合をαとする
- 台数効果は以下のように計算できる：

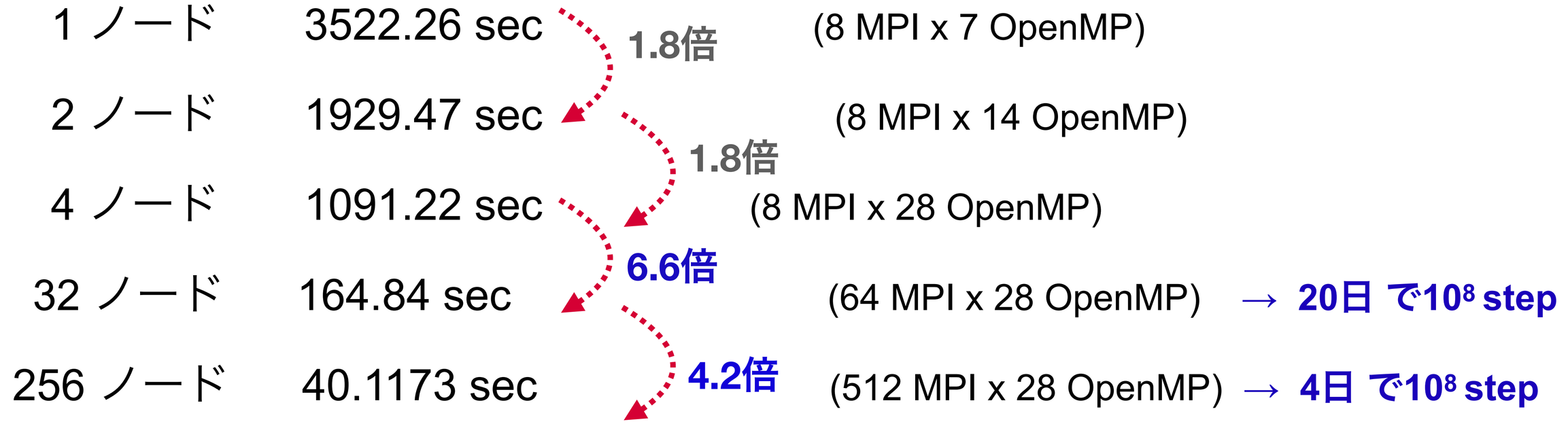
$$S_p = \frac{T_s}{\frac{\alpha T_s}{p} + (1-\alpha)T_s} = \frac{1}{\frac{\alpha}{p} + 1 - \alpha} = \frac{1}{1 + \alpha\left(\frac{1}{p} - 1\right)}.$$

- 無限大の数のプロセッサを使っても，台数効果は$1/(1-\alpha)$が上限（アムダールの法則）
  - 全体の90%が並列化できたとしても，台数効果は最大で$\frac{1}{1-0.9} = 10$倍
  - →高性能を達成するためには，少しでも並列化効率を上げる実装をすることが重要



並列化効率評価 = 多ノード実行性能を推定

16,384,000 粒子，10^4 ステップに対して評価　(8 MPI x 7 OpenMP)



注）お試しアカウントでは 8 ノードまでしか実行できません

台数効果を大雑把に逆算してみると。。。

- 10^4 ステップに要する逐次実行時間　（1コアとしたときの概算）

  T_s = 3522.26 sec x 56 コア ~ 200000 sec.

- 256 ノード = 14436 コア 使用した時の実行時間　40.1 sec ~ 5000 倍 = S_p

- 台数効果の式から逆算

$$S_p = \frac{T_s}{\frac{\alpha T_s}{p} + (1-\alpha)T_s} = \frac{1}{\frac{\alpha}{p} + 1 - \alpha} = \frac{1}{1 + \alpha\left(\frac{1}{p} - 1\right)}.$$

α ~ 0.99987

全体の計算のうちの99.987 % が並列化されていた

並列計算を開始する前に

- 99.99 % の並列化率というのは高い数字ですが、それでも100ノードを超えると、かなり計算速度は減少。

- この事情は、例えばオープンソースの並列プログラム （OpenFOAM, LAMMPS, Gromacs, etc… ）を使用する場合も同じです。

- 並列化率だけでなく、並列度に応じたメモリレイアウトの変化、データの読み込みや書き出しなどに影響される場合もあります。

GUI開発環境

一定以上の規模のコード開発には対話型開発環境 (IDE) が便利

ー　GUI 環境でコード開発から検証まで

Eclipse，Xcode (mac)，Android Studio (androidアプリ)，....

Python IDE = JupyterLab （後述）

Microsoft Visual Studio Code　フリー

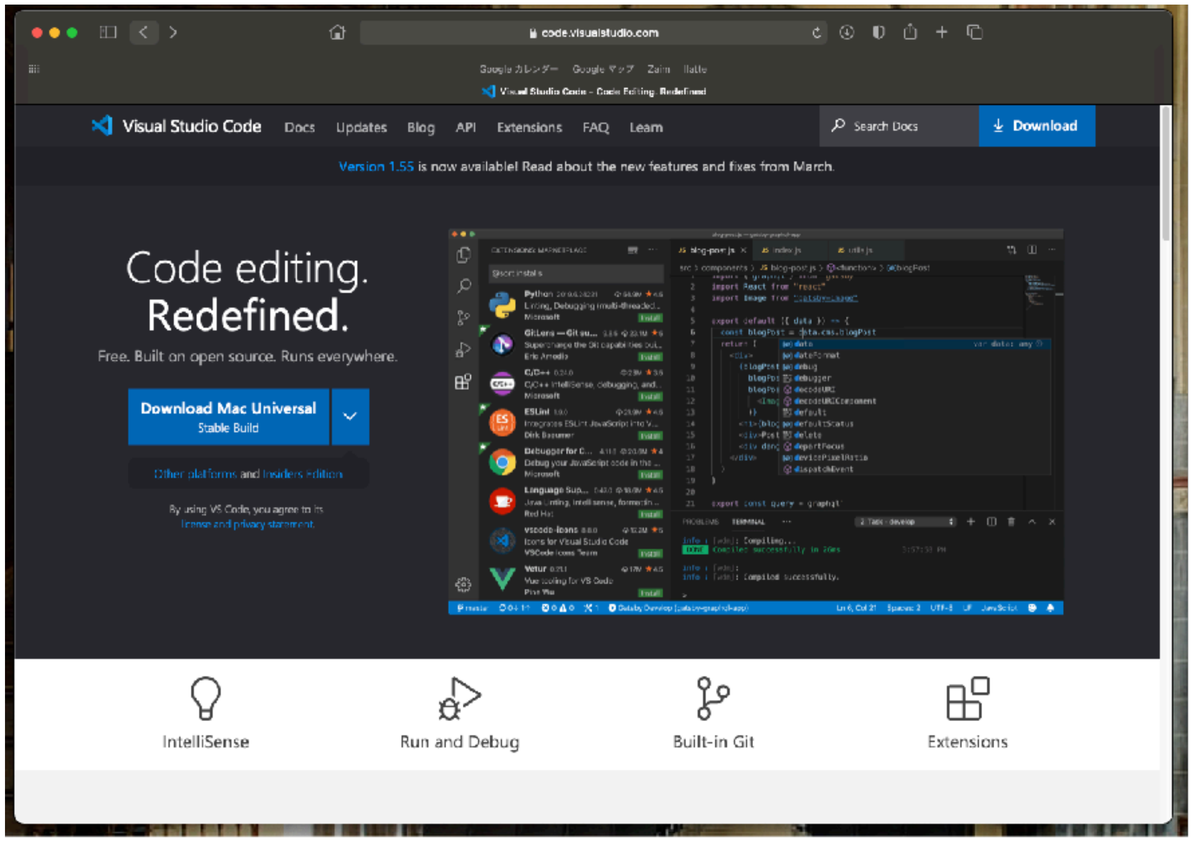
Remote-SSH 機能が2019年より追加

スパコン上での利用も便利に

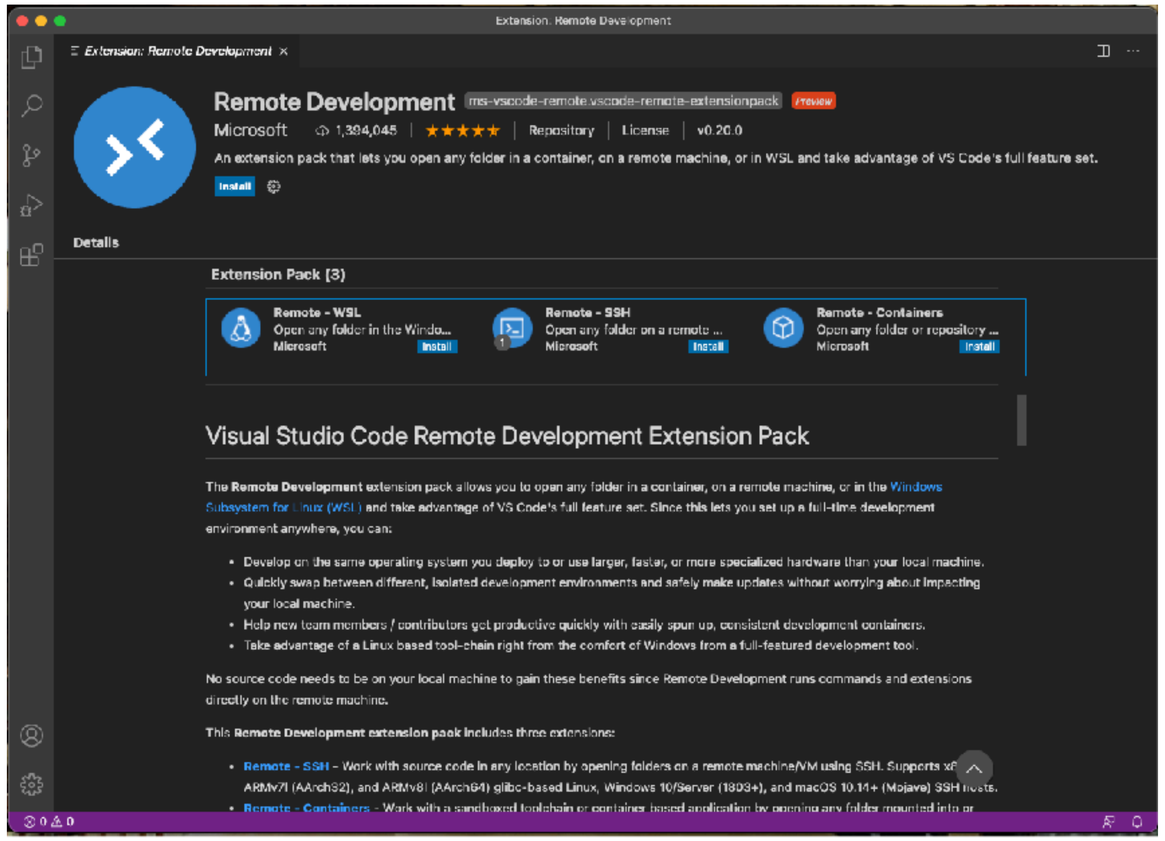
https://marketplace.visualstudio.com/items?itemName=ms-vscode-remote.vscode-remote-extensionpack

Visual Studio Code　— ダウンロード

ウェブブラウザからダウンロード
https://code.visualstudio.com



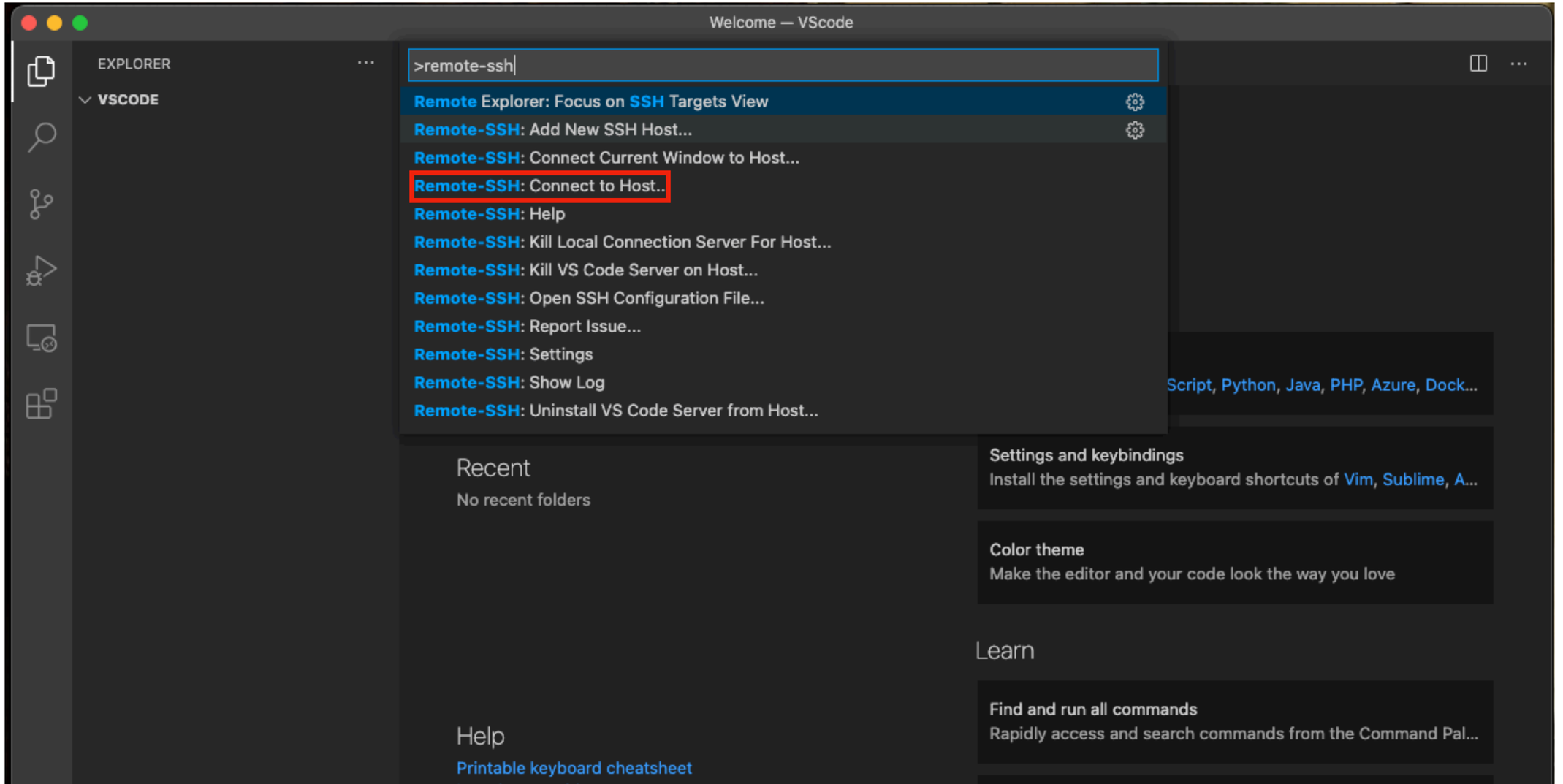
拡張機能
→　Remote Development　→　Install



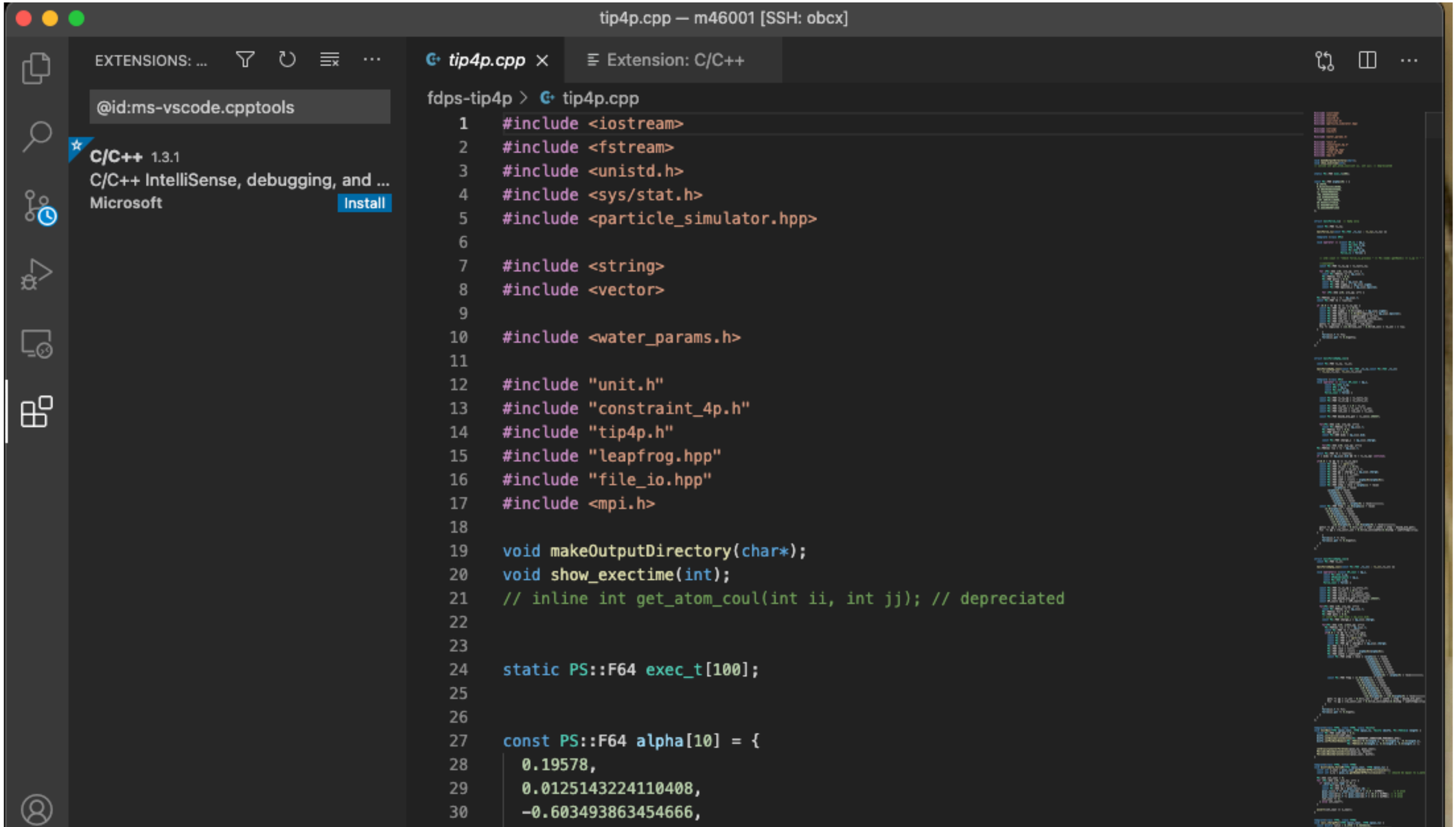
view > command palette ＞ “remote-ssh ” と入力 → 選択

接続先　tUVXYZ@obcx.cc.u-tokyo.ac.jp　を入力

■ .ssh/config を使っている方は、インポートできて便利です。



コードブロックの全体も見えて、作業がしやすくなっています。

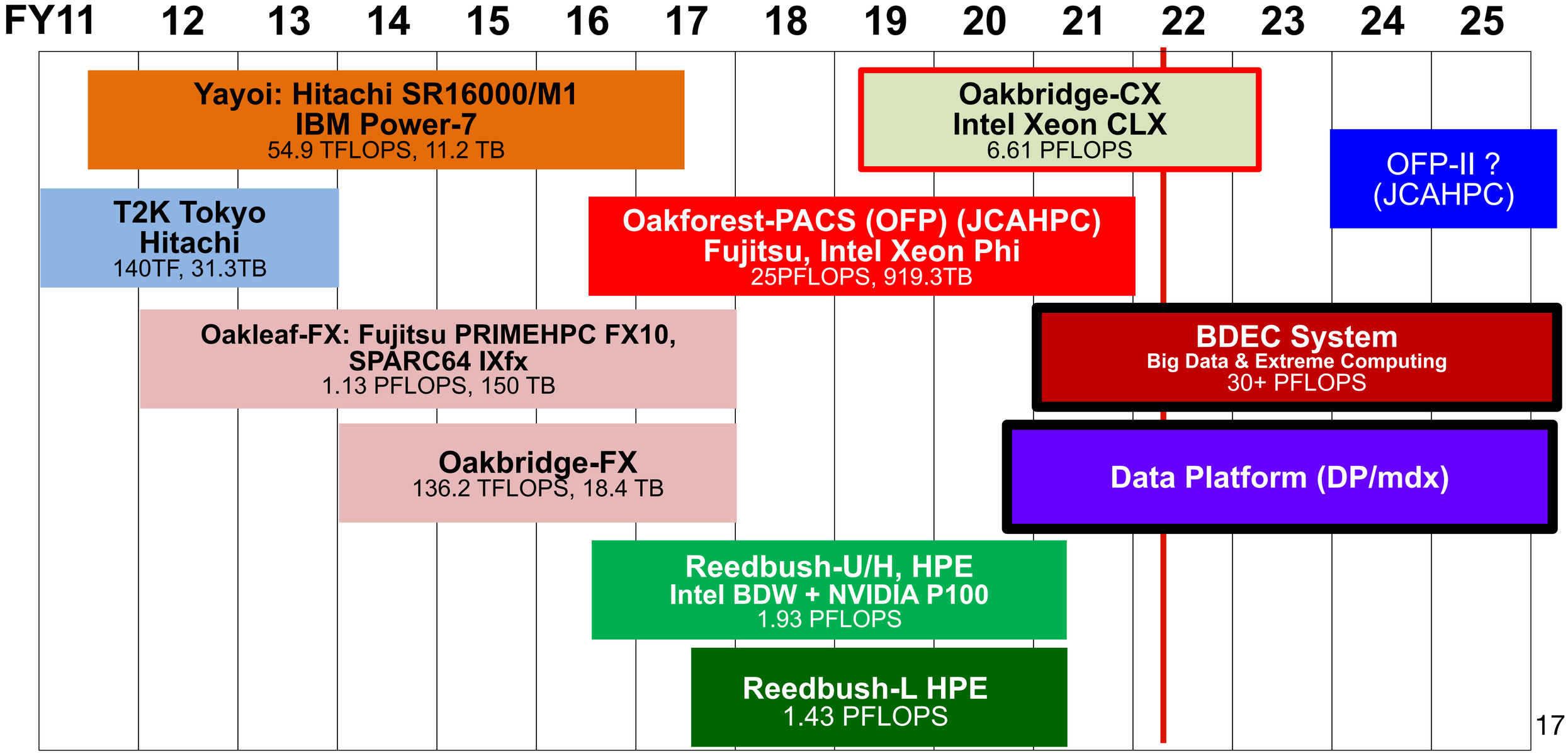


東大情報基盤センターの
スーパーコンピュータ利用案内

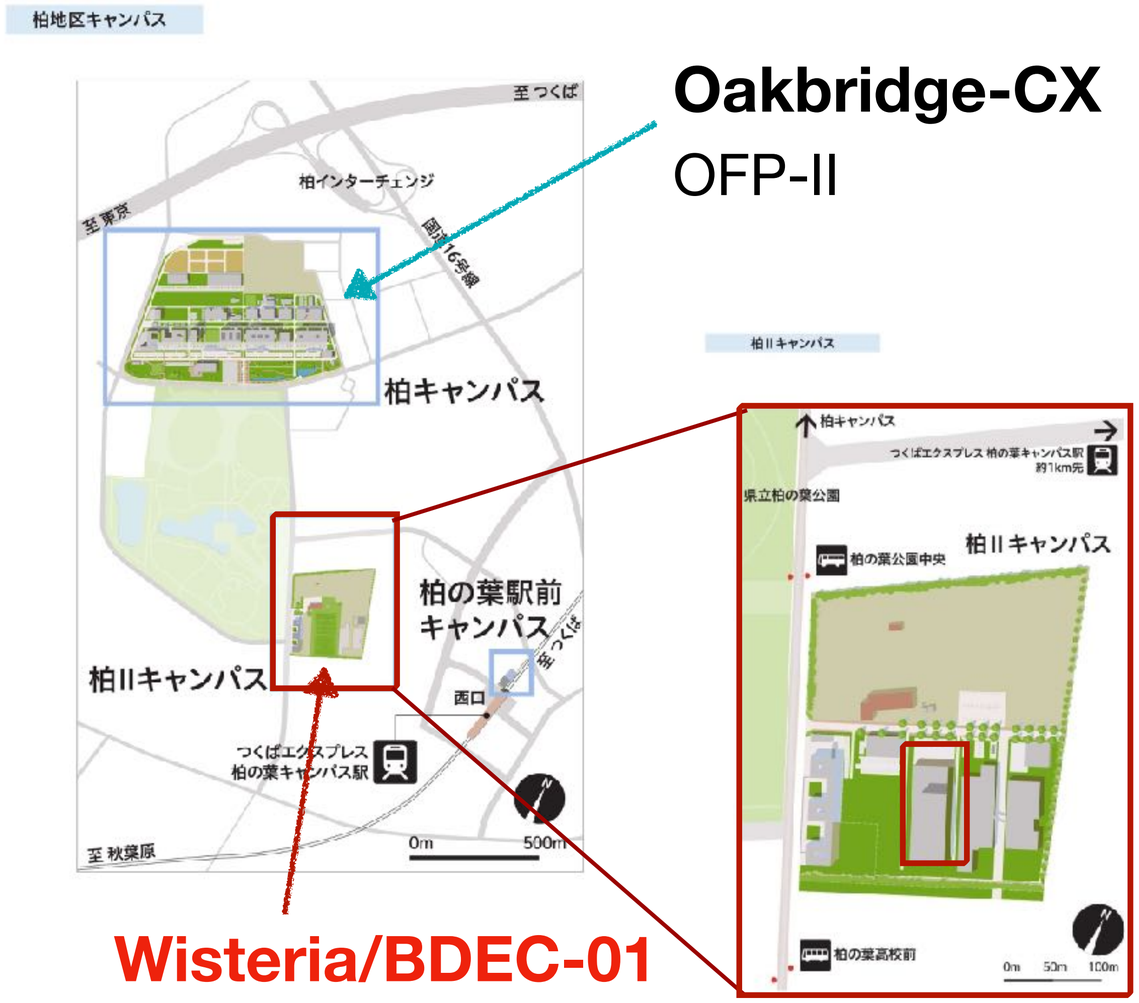
Supercomputers @ ITC/U.Tokyo

Information Technology Center, The University of Tokyo

東京大学情報基盤センター
INFORMATION TECHNOLOGY CENTER, THE UNIVERSITY OF TOKYO



東京大学のスーパーコンピュータは 柏キャンパス＋柏II キャンパスへ



Oakbridge-CX

OFP-II

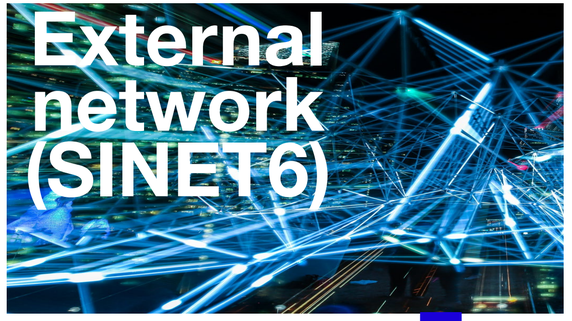
Wisteria/BDEC-01

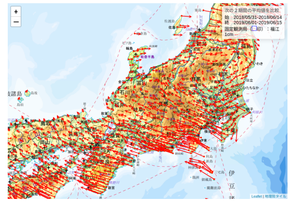
mdx + 産総研 ABCI

Wisteria/BDEC-01

2021年5月~

Wisteria = 紫藤





800 Gbps

Fast File System
1 PB, 1.0 TB/s

Simulation Nodes:
Odyssey
Fujitsu/PRIMEHPC FX1000
Arm A64FX processor 7680 nodes
25.9PF, 7.8 PB/s

2.0 TB/s

Data/Learning Nodes:
Aquarius
360 NVIDIA A100/SXM GPUs
(on 45 nodes of Intel IceLake CPUs)
7.20 PF, 578.2 TB/s

Shared File System
25.8 PB, 500 GB/s

CPU機
「富岳」と同型

GPU機
8台 / ノード

全体構成

	Odyssey	Aquarius
理論演算性能	25.9 PFLOPS	7.2 PFLOPS
ノード数	7,680 (10x8x8x2x3x2)	45
総メモリ量	240 TiB	36.5 TiB
総メモリバンド幅	**7.8 PB/s**	578.2 TB/s
ネットワークトポロジ	6D Mesh Torus	Full-bisection BW Fat Tree
インタコネクト	**Tofu-D**	**InfiniBand HDR**
バイセクションバンド幅	13 TB/s	4.5 TB/s
ノード間結合・ストレージ ネットワーク バンド幅	InfiniBand HDR, 2.0 TB/s	

ノード構成

項 目		Odyssey	Aquarius
		Fujitsu PRIMEHPC FX1000	Fujitsu PRIMERGY GX257 (Succesor model)
CPU	Processor name	Fujitsu A64FX	Next Intel Xeon IceLake processor
	#CPU (cores)	1 (48+2 or 4)	2 (36 + 36)
	Frequency	2.2 GHz	2.4GHz
	Peak performance	3.3 TFLOPS	5.53 TFlops
Memory		32 GB, 1TB/s	512 GB / 409.6 GB/s
GPU	GPU name	-	NVIDIA A100 Tensorcore
	Peak performance		19.5 TFlops (FP64 Tensor Core)
	Memory		40 GB, 1.55 TB/s
	#GPU / node		**8**, NVswitch connected
インターコネクト		Tofu-D	InfiniBand HDR (200 Gbps) x4
External connection		-	25 Gbps/node

一般利用料金

- １セット = 60,000円 /年　（1 年間 Odyssey 1 node 使用する量に相当）

	最大利用可能ノード数	トークン量	ディスク量
Wisteria-Odyssey	2,304 nodes = 110,592 cores (予定)	8640 (=24x360) / (node hours)	2TB / set
Wisteria-Aquarius	8 nodes = 64 GPUs (予定)	2880 (=24x120) / (GPU hours)	2TB / set

他、ノード固定 or GPU占有利用、外部接続ノード利用など
詳細の料金情報はホームページ 参照

https://www.cc.u-tokyo.ac.jp

東大情報基盤センターの利用申し込み枠（○：代表者，△：参加者）

制度名	種別	大学等	企業	学生	個人	審査	無料	報告書	A	B	C	D	備考	募集
通常利用	一般	○	△	○					✓	✓	✓			随時
	トライアル	○	△	○				✓			✓	✓	年度内	随時
お試し利用		○	○	○	✓		✓				✓	✓	1ヶ月限定	随時
JHPCN		○	○	△		書類	✓	✓		✓				年1回（1月）
HPCI	一般・若手	○	△	△		書類	✓	✓		✓				年1回（10-11月）
	産業		○			書類	✓	✓		✓				
若手女性	一般	○	○	○	✓	書類	✓	✓		✓	✓			年2回（8・2月）
	インターン			○	✓	書類	✓	✓			✓			年1回（夏季）
AI for HPC		○	○	△		書類	✓	✓		✓	✓			年1回（2月）
HPCチャレンジ		○	○	○		書類	✓	✓						年数回
講習会		△	△	△	✓		✓						1ヶ月有効UID	年20回程度
教育利用		○	○	○		書類	✓	✓					企業研修等可	随時
企業利用	一般	△	○	△		＋面接		✓		✓				年2回（8・2月）
	トライアル	△	○	△		＋面接	一部	✓			✓	✓	3ヶ月無料 年度内	随時，年4回審査

（A:トークン移行，B:ノード固定，C:Odyssey⇔Aquarius移行可能，D:1システム1回限り応募可能）

情報基盤センターの資源を無料で使いたい場合

（大学・公的研究機関に所属の方）

- 若手・女性利用課題

  40歳以下または女性、年２回の募集（次回は 8月末〆切）

- 学際大規模情報基盤共同利用・共同研究拠点課題

  基盤センターとの共同研究,　2022年度は募集終了

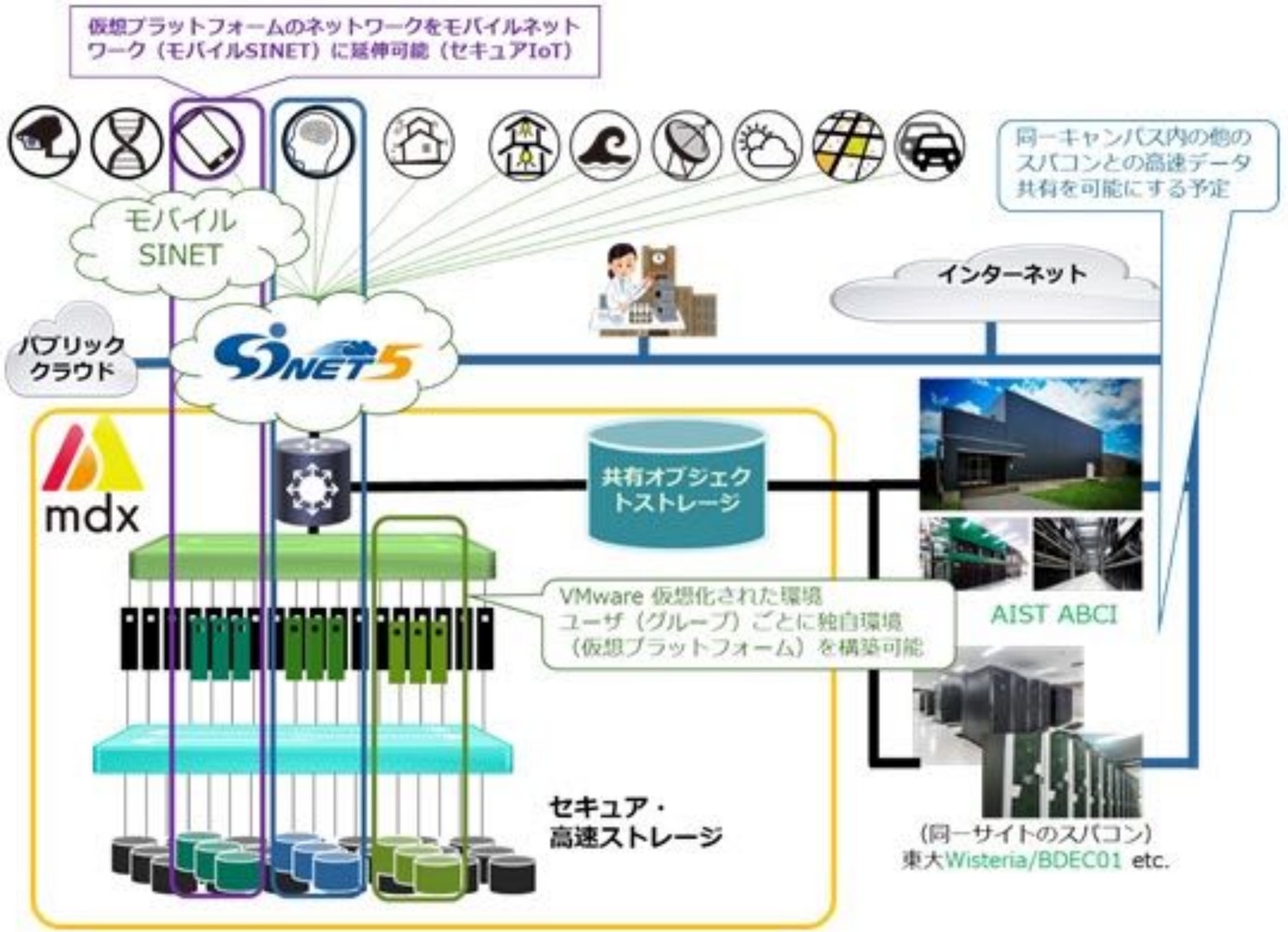
- HPCI 課題（RIST）

  2022年度分の一般課題について、各基盤センター分は募集終了

mdx https://mdx.jp

いわゆるスパコンとは違う
新たなクラウド型
プラットフォーム

- 仮想マシンおよびVPNで互いに隔離された占有環境
- コンテナ・機械学習利用
- SINET6（2022導入）の活用
  400Gbps 全国ネットワーク
- ７大学 + ２研究所の共同Pj



mdx https://mdx.jp

性能・構成

- **368** CPU nodes (Intel Ice Lake)
- **40** GPU nodes (NVIDIA A100 x 8)

Storages

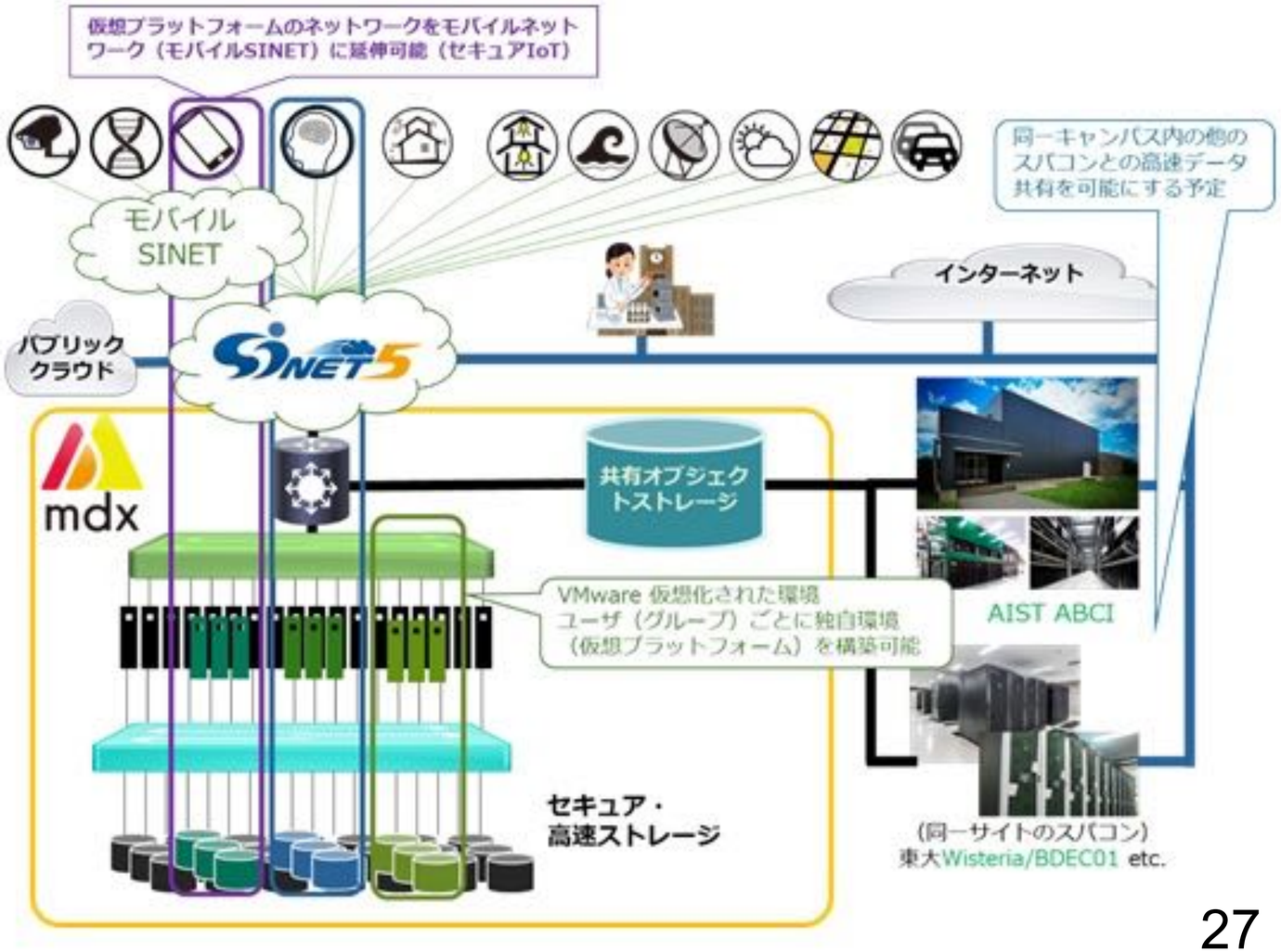
high-speed NVMe SSD
(1.0PB @ 250GB/s)
large HDD (16.3PB @ 157.5GB/s)
S3 compatible storage
(10.3PB @ 63GB/s)

Networks

Ethernet 25 Gbps
RDMA 100 Gbps (to SINET5)

Sortwares

VM & Container, IaaS



お疲れ様でした