



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

オンライン講習会の手引き OpenMPによるマルチコア・メニィコア 並列プログラミング入門

東京大学情報基盤センター
2021年4月

質問は中島まで:[nakajima\(at\)cc.u-tokyo.ac.jp](mailto:nakajima(at)cc.u-tokyo.ac.jp)
(相談窓口にはメールしないでください)

概 要

- 講習会ホームページ
 - <https://www.cc.u-tokyo.ac.jp/events/lectures/151/>
- 本講習会では、理解を深めるため、[Oakbridge-CXスーパーコンピュータシステム](#)を利用した実習を実施する予定である。
- 本講習会はZoomによるオンライン講習会として実施される予定である。
- 本資料は、オンライン講習会受講のための準備について記載したものである。

お願い等

- 東京大学情報基盤センタースーパーコンピュータシステム利用
規程を熟読し、遵守してください。
- ハンズオンのためのPC, Zoom及びスパコンへ接続するためのネットワーク環境は各受講者でご準備ください。
- PCは Windows/Microsoft Update, Apple Security Updateなどで最新のセキュリティアップデートを行ってください。
- 必ずウィルス対策ソフトウェアをインストールし、ウィルス検索を実行して問題がないことを事前に確認してから受講してください。
 - セキュリティ対策未実施の場合はオンライン講習会受講を認めません。
- OSは、Windows、Macどちらでも構いませんが、SSHを用いてスーパーコンピュータへ接続ができることが必須です（後述）。
- 受講生にセンターのスーパーコンピュータを1月間利用できる無料アカウント（お試しアカウント）を発行します。

準備について

- FortranまたはC言語によるプログラミング経験のあることが望ましい(必須ではありません)
- UNIXの経験があることが望ましい
 - 最低限emacsまたはvi等によりファイルの編集をできる必要があります
 - emacs, vi等未経験の場合は, 各自予め自習しておいてください
- 以下に示す「PC上でのソフトウェアの準備」, 「スパコンへのログイン」, 「ログインしたら・・・」について受講日までに完了していることを前提として, 講習会を実施いたしますので, よろしくお願いたします。
 - もし支障がある場合には, オンラインでの相談の機会を設けますので, 希望者は申し出てください
 - 連絡方法については追ってご連絡いたします
- 「東大情報基盤センターのスパコン」も予め目を通してください

- PC上のソフトウェア類の準備
- 東大情報基盤センターのスパコン
- スパコンへのログイン
- ログインしたら・・・

インストールすべきソフトウェア

	C	Fortran
Windows	Cygwin Paraview FVM-in-C	Cygwin Paraview FVM-in-F
Mac	Paraview FVM-in-C	Paraview FVM-in-F
Linux		

- Cygwin <https://www.cygwin.com/>
- Paraview <http://www.paraview.org> 可視化用です
- 有限体積法によるプログラム
 - FVM-in-F (Fortran) <http://nkl.cc.u-tokyo.ac.jp/files/multicore-f.tar>
 - FVM-in-C (C) <http://nkl.cc.u-tokyo.ac.jp/files/multicore-c.tar>
 - **Makefile**におけるデフォルトコンパイル名は「**cc (FVM-in-C, C)**」, 「**gfortran (FVM-in-F, Fortran)**」となっております。各自のPC上のコンパイラに応じてMakefileを変更願います。

Cygwin: Windows上のUNIXライクな環境

<https://www.cygwin.com/>

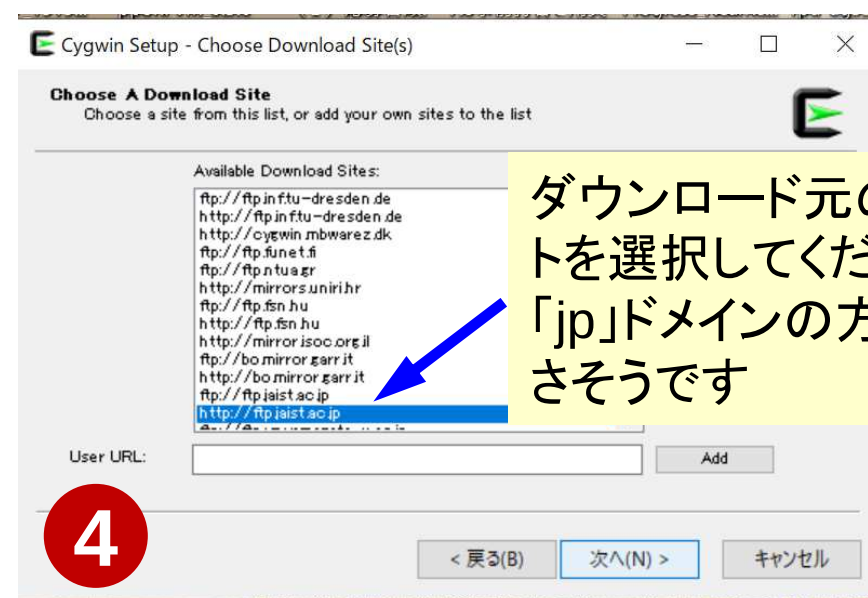
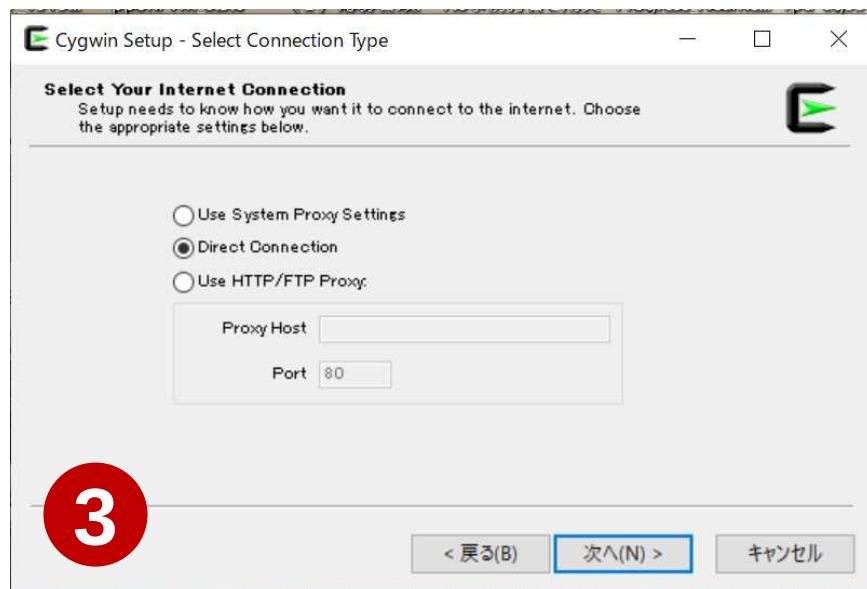
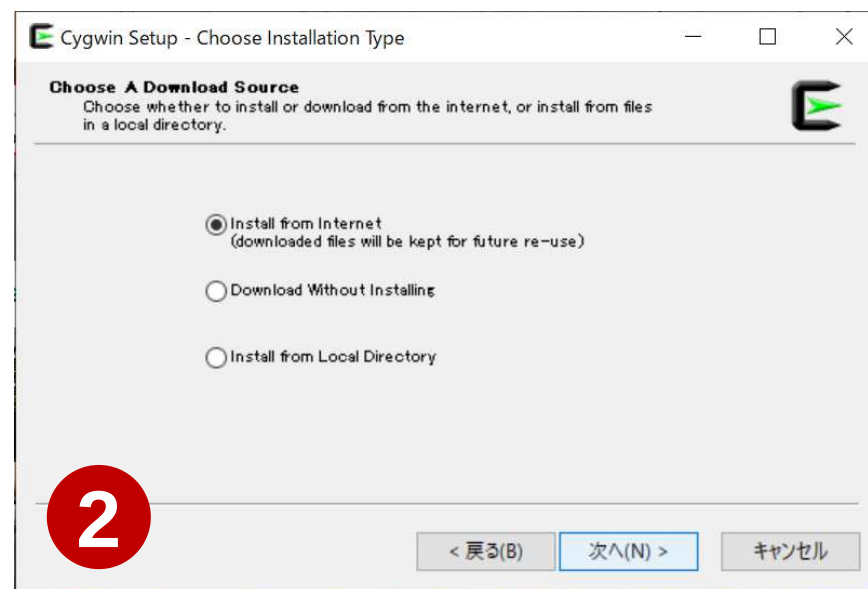
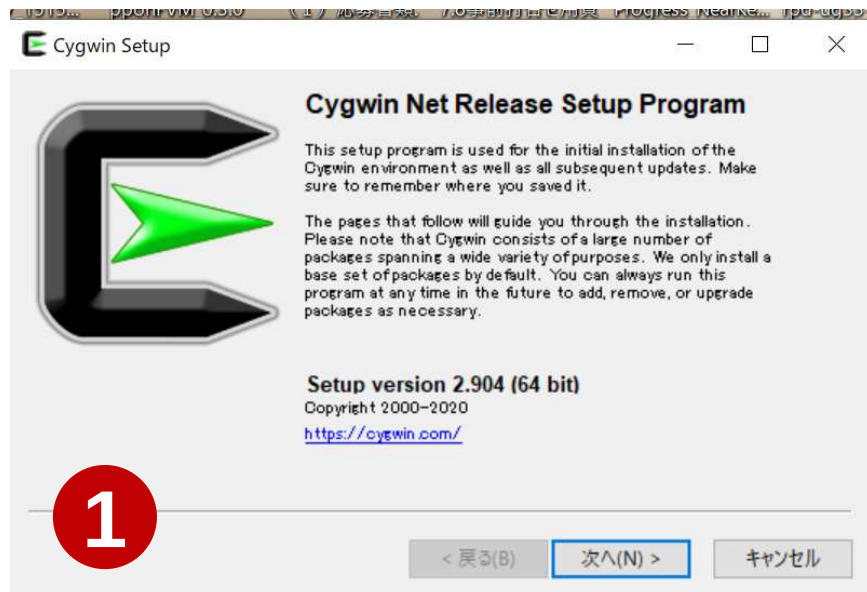
- 利用者ガイド
 - <https://cygwin.com/cygwin-ug-net.html>
- まずやるべきこと
 - インストーラ(setup-x86_64/32.exe)を予め入手
 - インターネットに接続し, インストーラをダブルクリック
 - 以下指示に従ってインストールを進めてください

以下しばらくはCygwinの話



指示に従ってください

<https://www.cygwin.com/>



ダウンロード元のサイトを選択してください。「jp」ドメインの方がよさそうです

まずはデフォルト機能のインストールから

<https://www.cygwin.com/>

- 基本的な機能はデフォルトのインストールでOKですが、本講習会で必要なものが抜けている可能性があります。
 - 従ってマニュアルでインストールする必要があります
 - インストーラをダブルクリックすれば後で追加も可能です
- 本講習会では下記が必須です（デフォルトのインストールでは抜けている可能性あり）
 - gcc-core (for C/C++ users)
 - gcc-fortran (for Fortran users)
 - openssh (for all users)
 - openssl (for all users)
 - make (for all users)
 - emacs, vim etc.
- インストールされているかどうかは確認が可能です

“gcc-core”の有無に関するチェック

Type “gcc-core”

Cygwin Setup - Select Packages

Select Packages
Select packages to install

View: Full

Search: Clear

☐ Keep ☒ Best ☐ Sync ☐ Test

Package	Current	New	Src?	Categories	Size	Description
cygwin32-gcc-core		Skip	☐	Devel	16,464k	GCC for Cygwin 32bit toolchain (C, OpenMP)
djgpp-gcc-core		Skip	☐	Devel	7,926k	GCC for DJGPP toolchain (C)
gcc-core	9.3.0-1	Keep	☐	Devel	20,500k	GNU Compiler Collection (C, OpenMP)
mingw64-i686-gcc-core		Skip	☐	Devel	16,851k	GCC for Win32 (i686-w64-mingw32) toolchain (C, OpenMP)
mingw64-x86_64-gcc-core		Skip	☐	Devel	17,464k	GCC for Win64 toolchain (C, OpenMP)

☒ Hide obsolete packages

< 戻る(B) 次へ(N) > キャンセル

“gcc-core”の有無に関するチェック

Type “gcc-core”

Cygwin Setup - Select Packages

Select Packages
Select packages to install

View: Full

Search: Clear

Buttons: ☐ Keep ☒ Best ☐ Sync ☐ Test

Package	Current	New	Src?	Categories	Size	Description
cygwin32-gcc-core		Skip	☐	Devel	16,464k	GCC for Cygwin 32bit toolchain (C, OpenMP)
cygwin64-gcc-core		Skip	☐	Devel	7,926k	GCC for DJGPP toolchain (C)
gcc-core	9.3.0-1	Keep	☐	Devel	20,500k	GNU Compiler Collection (C, OpenMP)
mingw32-gcc-core		Skip	☐	Devel	16,851k	GCC for Win32 (i686-w64-mingw32) toolchain (C, OpenMP)
mingw64-x86_64-gcc-core		Skip	☐	Devel	17,464k	GCC for Win64 toolchain (C, OpenMP)

Buttons: < 戻る(B) 次へ(N) > キャンセル

☒ Hide obsolete packages

20:19
2020/04/14

“Keep”と出てきたら
“gcc-core”はインストール済み

“g++”のインストール例 (1/4)

Type “g++”

Cygwin Setup - Select Packages

Select Packages
Select packages to install

View Full Search g++ Clear

Keep Best Sync Test

Package	Current	New	Src?	Categories	Size	Description
cygwin32-gcc-g++		Skip	☐	Devel	10,456k	GCC for Cygwin 32bit toolchain (C++)
diagon-gcc-g++		Skip	☐	Devel	8,279k	GCC for DJGPP toolchain (C++)
gcc-g++		Skip	☐	Devel	16,257k	GNU Compiler Collection (C++)
mingw64-i686-gcc-g++		Skip	☐	Devel	14,358k	GCC for Win32 (i686-w64-mingw32) toolchain (C++)
mingw64-x86_64-gcc-g++		Skip	☐	Devel	14,781k	GCC for Win64 toolchain (C++)

“Skip” が現れたら「未インストール」を意味する

以下「g++」を例にマニュアルインストールの実施方法を紹介する

< 戻る(B) 次へ(N) > キャンセル

Hide obsolete packages

20:18 2020/04/14

“g++”のインストール例 (2/4)

Cygwin Setup - Select Packages

Select Packages
Select packages to install

View Full Search g++ Clear

Keep Best Sync Test

Package	Current	New	Src?	Categories	Size	Description
cygwin32-gcc-g++		Skip	☐	Devel	10,456k	GCC for Cygwin 32bit toolchain (C++)
diagon-gcc-g++		Skip	☐	Devel	8,279k	GCC for DJGPP toolchain (C++)
gcc-g++		Skip	☐	Devel	16,257k	GNU Compiler Collection (C++)
mingw64-i686-gcc-g++		Skip	☐	Devel	14,358k	GCC for Win32 (i686-w64-mingw32) toolchain (C++)
mingw64-x86_64-gcc-g++		Skip	☐	Devel	14,781k	GCC for Win64 toolchain (C++)

プルダウンメニュー

< Hide obsolete packages

< 戻る(B) 次へ(N) > キャンセル

20:18 2020/04/14

“g++”のインストール例 (3/4)

Cygwin Setup - Select Packages

Select Packages
Select packages to install

View Full Search g++ Clear

Keep Best Sync Test

Package	Current	New	Src?	Categories	Size	Description
cygwin32-gcc-g++		Skip	☐	Devel	10,456k	GCC for Cygwin 32bit toolchain (C++)
diipp-gcc-g++		Skip	☐	Devel	8,279k	GCC for DJGPP toolchain (C++)
gcc-g++		Skip	☐	Devel	16,257k	GNU Compiler Collection (C++)
mingw64-i686-gcc-g++		Skip	☐		14,358k	GCC for Win32 (i686-w64-mingw32) toolchain (C++)
mingw64-x86_64-gcc-g++		Skip	☐		14,781k	GCC for Win64 toolchain (C++)

Uninstall
Skip
7.4.0-1
8.3.0-1 (Test)
9.2.0-1 (Test)
9.2.0-2
9.2.0-3
9.3.0-1

適切なバージョンを選択
(通常は最新版)

Hide obsolete packages

< 戻る(B) 次へ(N) > キャンセル

20:18
2020/04/14

“g++”のインストール例 (4/4)

Cygwin Setup - Select Packages

Select Packages
Select packages to install

View Full Search g++ Clear

☐ Keep ☒ Best ☐ Sync ☐ Test

Package	Current	New	Src?	Categories	Size	Description
cygwin32-gcc-g++		Skip	☐	Devel	10,456k	GCC for Cygwin 32bit toolchain (C++)
diipp-gcc-g++		Skip	☐	Devel	8,279k	GCC for DJGPP toolchain (C++)
gcc-g++		9.3.0-1	☐	Devel	16,257k	GNU Compiler Collection (C++)
mingw64-i686-gcc-g++		Skip	☐	Devel	14,358k	GCC for Win32 (i686-w64-mingw32) toolchain (C++)
mingw64-x86_64-gcc-g++		Skip	☐	Devel	14,781k	GCC for Win64 toolchain (C++)

「Skip」のかわりにバージョン番号が出てきたら“g++”のインストール準備完了
(インストールは完了していない)

ここをクリック

< 戻る(B) 次へ(N) > キャンセル

Hide obsolete packages

“gcc”：インストールの確認

```
$ gcc -v
```

組み込み spec を使用しています。

COLLECT_GCC=gcc

COLLECT_LTO_WRAPPER=/usr/lib/gcc/x86_64-pc-cygwin/9.3.0/lto-wrapper.exe

ターゲット: x86_64-pc-cygwin

configure 設定: /cygdrive/i/szsz/tmp/gcc/gcc-9.3.0-1.x86_64/src/gcc-9.3.0/configure --
srcdir=/cygdrive/i/szsz/tmp/gcc/gcc-9.3.0-1.x86_64/src/gcc-9.3.0 --prefix=/usr --exec-prefix=/usr --
localstatedir=/var --sysconfdir=/etc --docdir=/usr/share/doc/gcc --htmldir=/usr/share/doc/gcc/html -C
--build=x86_64-pc-cygwin --host=x86_64-pc-cygwin --target=x86_64-pc-cygwin --without-libiconv-prefix -
without-libintl-prefix --libexecdir=/usr/lib --enable-shared --enable-shared-libgcc --enable-static -
enable-version-specific-runtime-libs --enable-bootstrap --enable-__cxa_atexit --with-dwarf2 --with-
tune=generic --enable-languages=c,c++,fortran,lto,objc,obj-c++ --enable-graphite --enable-
threads=posix --enable-libatomic --enable-libgomp --enable-libquadmath --enable-libquadmath-support --
disable-libssp --enable-libada --disable-symvers --with-gnu-ld --with-gnu-as --with-cloog-
include=/usr/include/cloog-isl --without-libiconv-prefix --without-libintl-prefix --with-system-zlib -
enable-linker-build-id --with-default-libstdcxx-abi=gcc4-compatible --enable-libstdcxx-filesystem-ts

スレッドモデル: posix

gcc バージョン 9.3.0 (GCC)

“gfortran” : インストールの確認

```
$ gfortran -v
```

組み込み spec を使用しています。

COLLECT_GCC=gfortran

ターゲット: x86_64-pc-cygwin

configure 設定: /cygdrive/i/szsz/tmp/gcc/gcc-9.3.0-1.x86_64/src/gcc-9.3.0/configure --srcdir=/cygdrive/i/szsz/tmp/gcc/gcc-9.3.0-1.x86_64/src/gcc-9.3.0 --prefix=/usr --exec-prefix=/usr --localstatedir=/var --sysconfdir=/etc --docdir=/usr/share/doc/gcc --htmldir=/usr/share/doc/gcc/html -C --build=x86_64-pc-cygwin --host=x86_64-pc-cygwin --target=x86_64-pc-cygwin --without-libiconv-prefix --without-libintl-prefix --libexecdir=/usr/lib --enable-shared --enable-shared-libgcc --enable-static --enable-version-specific-runtime-libs --enable-bootstrap --enable-__cxa_atexit --with-dwarf2 --with-tune=generic --enable-languages=c,c++,fortran,lto,objc,obj-c++ --enable-graphite --enable-threads=posix --enable-libatomic --enable-libgomp --enable-libquadmath --enable-libquadmath-support --disable-libssp --enable-libada --disable-symvers --with-gnu-ld --with-gnu-as --with-cloog-include=/usr/include/cloog-isl --without-libiconv-prefix --without-libintl-prefix --with-system-zlib --enable-linker-build-id --with-default-libstdcxx-abi=gcc4-compatible --enable-libstdcxx-fs-backend=fts

スレッドモデル: posix

gcc バージョン 9.3.0 (GCC)

“ssh-keygen (OpenSSH)” : インストールの確認

```
$ ssh-keygen --h
```

```
ssh-keygen: unknown option -- -
```

```
usage: ssh-keygen [-q] [-b bits] [-C comment] [-f output_keyfile] [-m format]
                  [-t dsa | ecdsa | ecdsa-sk | ed25519 | ed25519-sk | rsa]
```

```
                  [-N new_passphrase] [-O option] [-w provider]
ssh-keygen -p [-f keyfile] [-m format] [-N new_passphrase]
            [-P old_passphrase]
```

```
ssh-keygen -i [-f input_keyfile] [-m key_format]
```

```
ssh-keygen -e [-f input_keyfile] [-m key_format]
```

```
ssh-keygen -y [-f input_keyfile]
```

(...)

```
ssh-keygen -L [-f input_keyfile]
```

```
ssh-keygen -A [-f prefix_path]
```

```
ssh-keygen -k -f krl_file [-u] [-s ca_public] [-z version_number]
file ...
```

```
ssh-keygen -Q -f krl_file file ...
```

```
ssh-keygen -Y find-principals -s signature_file -f allowed_signers_file
```

```
ssh-keygen -Y check-novalidate -n namespace -s signature_file
```

```
ssh-keygen -Y sign -f key_file -n namespace file ...
```

```
ssh-keygen -Y verify -f allowed_signers_file -I signer_identity
-n namespace -s signature_file [-r revocation_file]
```

“ssh (OpenSSH)” : インストールの確認

```
$ ssh
```

```
usage: ssh [-46AaCfGgKkMNnqsTtVvXxYy] [-B bind_interface]
          [-b bind_address] [-c cipher_spec] [-D [bind_address:]port]
          [-E log_file] [-e escape_char] [-F configfile] [-I pkcs11]
          [-i identity_file] [-J [user@]host[:port]] [-L address]
          [-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port]
          [-Q query_option] [-R address] [-S ctl_path] [-W host:port]
          [-w local_tun[:remote_tun]] destination [command]
```

“make, emacs, vi etc” : インストールの確認

```
$ make -version
```

```
GNU Make 4.3
```

```
このプログラムは x86_64-pc-cygwin 用にビルドされました
```

```
Copyright (C) 1988-2020 Free Software Foundation, Inc.
```

```
ライセンス GPLv3+: GNU GPL バージョン 3 以降 <http://gnu.org/licenses/gpl.html>
```

```
これはフリーソフトウェアです: 自由に変更および配布できます.
```

```
法律の許す限り、 無保証 です.
```

```
$ emacs -version
```

```
GNU Emacs 26.3
```

```
Copyright (C) 2019 Free Software Foundation, Inc.
```

```
GNU Emacs comes with ABSOLUTELY NO WARRANTY.
```

```
You may redistribute copies of GNU Emacs
```

```
under the terms of the GNU General Public License.
```

```
For more information about these matters, see the file named COPYING.
```

```
$ vi -version
```

```
VIM - Vi IMproved 8.2 (2019 Dec 12, compiled Mar 30 2020 21:54:08)
```

```
Garbage after option argument: "-version"
```

```
More info with: "vim -h"
```

その他

- Cygwinのインストールには60-90分程度を要します
- 週末中にインストールしておくことをお勧めいたします。
- Windows 10以降であればCygwinの代わりにWSL (Windows Subsystem for Linux)を使うことも考えられます。
 - Windows, Linuxに詳しい方はお試しいても結構です, がC, Fortran, OpenSSH等の機能のインストールをご確認ください
- [List of Unix/Linux Commands \(Wikipedia\)](#)
- [Online Manual for Emacs \(Screen Editor for Linux/Unix\)](#)

- PC上のソフトウェア類の準備
- 東大情報基盤センターのスパコン
- スパコンへのログイン
- ログインしたら・・・

3システム：利用者2,600+, 学外55+%

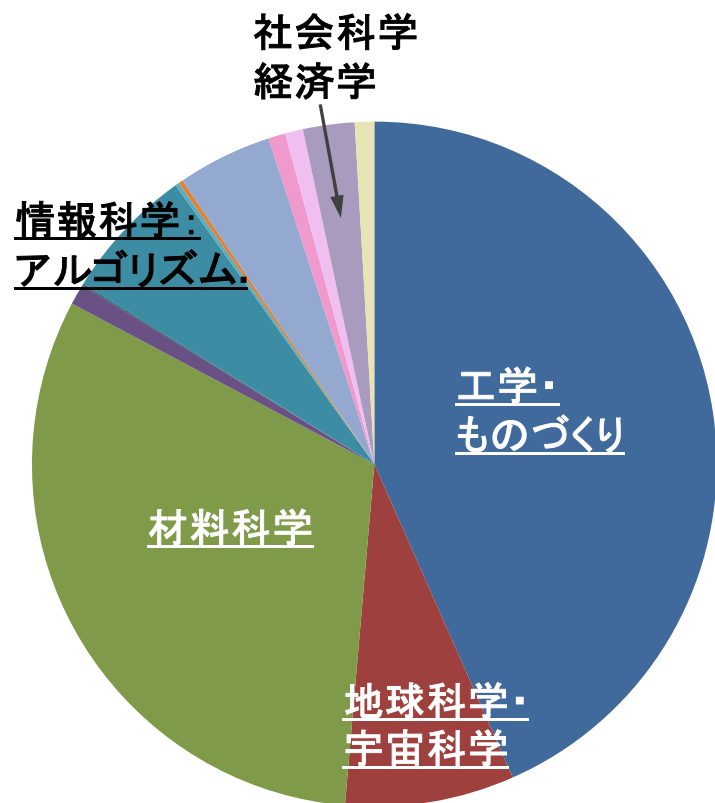
- **Reedbush (HPE, Intel BDW + NVIDIA P100 (Pascal))**
 - データ解析・シミュレーション融合スーパーコンピュータ
 - 3.36 PF, 2016年7月～ 2021年11月末(予定)
 - Reedbush-U (CPU only) (2020年6月末退役)
 - Reedbush-H (2GPU's/n), Reedbush-L (4GPU's)
 - 東大ITC初GPUクラスタ (2017年3月より), DDN IME (Burst Buffer)
- **Oakforest-PACS (OFP) (富士通, Intel Xeon Phi (KNL))**
 - JCAHPC (筑波大CCS & 東大ITC)
 - 25 PF, TOP500で22位 (日本4位) (2020年11月)
 - Omni-Path アーキテクチャ, DDN IME (Burst Buffer)
- **Oakbridge-CX (富士通, Intel Xeon Platinum 8280)**
 - 大規模超並列スーパーコンピュータシステム
 - 6.61 PF, 2019年7月 ～ 2023年6月, TOP500で69位 (2020年11月)
 - 全1,368ノードの内128ノードにSSDを搭載



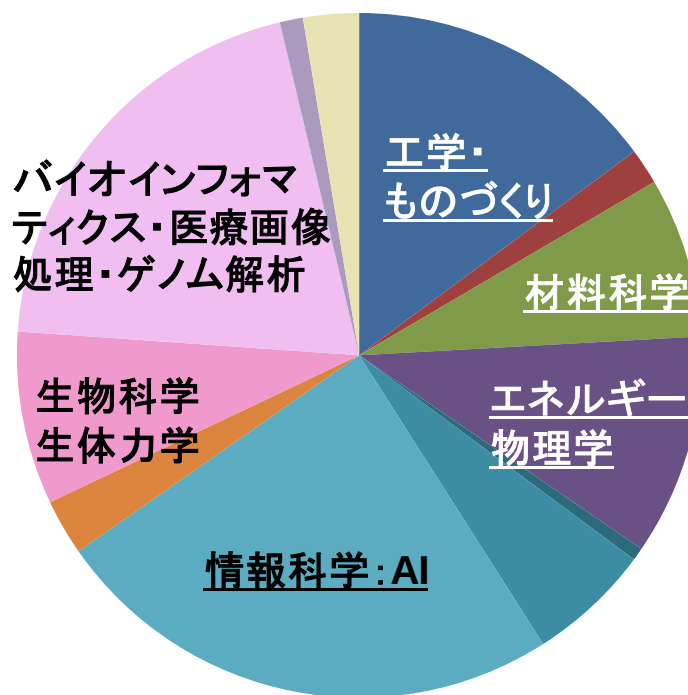
GFLOPS当たり利用負担(円):電気代, GFLOPS/W (Green 500)

System	JPY/GFLOPS Small is Good	GFLOPS/W Large is Good
Oakleaf-FX/Oakbridge-FX (Fujitsu) (Fujitsu SPARC64 IXfx)	125	0.8663
Reedbush-U (HPE) (Intel BDW)	61.9	2.310
Reedbush-H (HPE) (Intel BDW+NVIDIA P100x2/node)	15.9	8.575
Reedbush-L (HPE) (Intel BDW+NVIDIA P100x4/node)	13.4	10.167
Oakforest-PACS (Fujitsu) (Intel Xeon Phi/Knights Landing)	16.5	4.986
Oakbridge-CX (Fujitsu) (Intel Cascade Lake (CLX))	20.7	5.076

研究分野別利用CPU時間割合(2019年度)



マルチコアクラスタ
Intel BDW Only
(Reedbush-U)

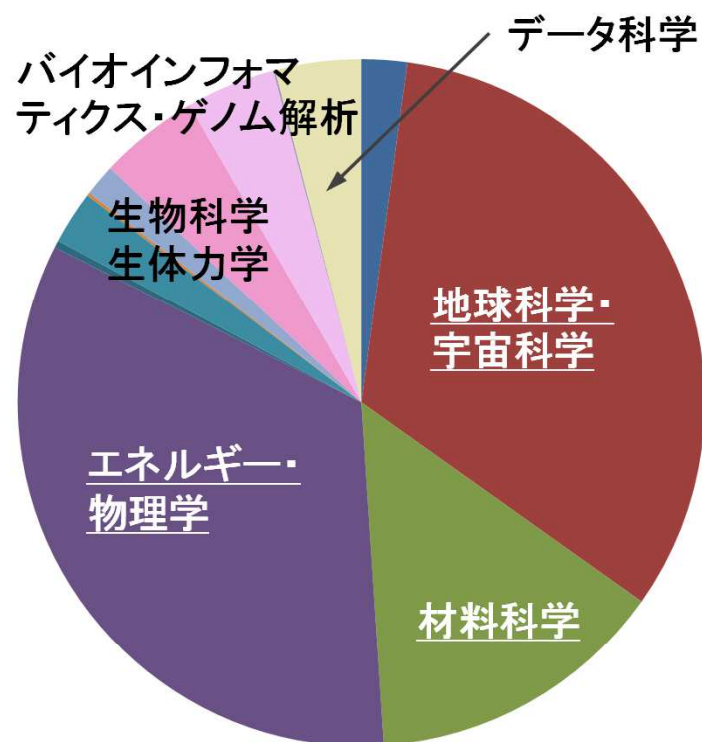


GPUクラスタ
Intel BDW + NVIDIA P100
(Reedbush-H)

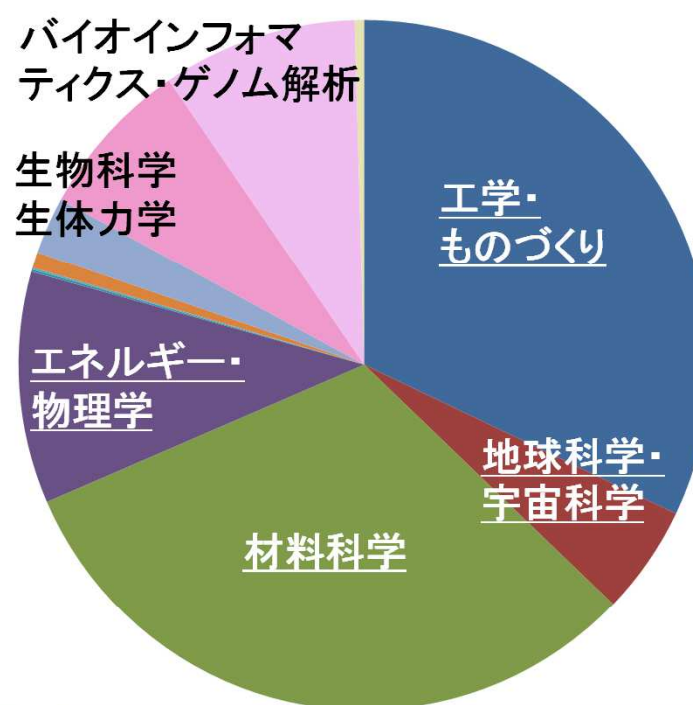
- 工学・ものづくり
- 地球科学・宇宙科学
- 材料科学
- エネルギー・物理学
- 情報科学: システム
- 情報科学: アルゴリズム
- 情報科学: AI
- 教育
- 産業利用
- 生物科学・生体力学
- バイオインフォマティクス
- 社会科学・経済学
- データ科学・データ同化

研究分野別利用CPU時間割合(2019年度)

OBCXは2019年10月～2020年9月(12ヶ月)



メニコアクラスタ
Intel Xeon Phi
(Oakforest-PACS)

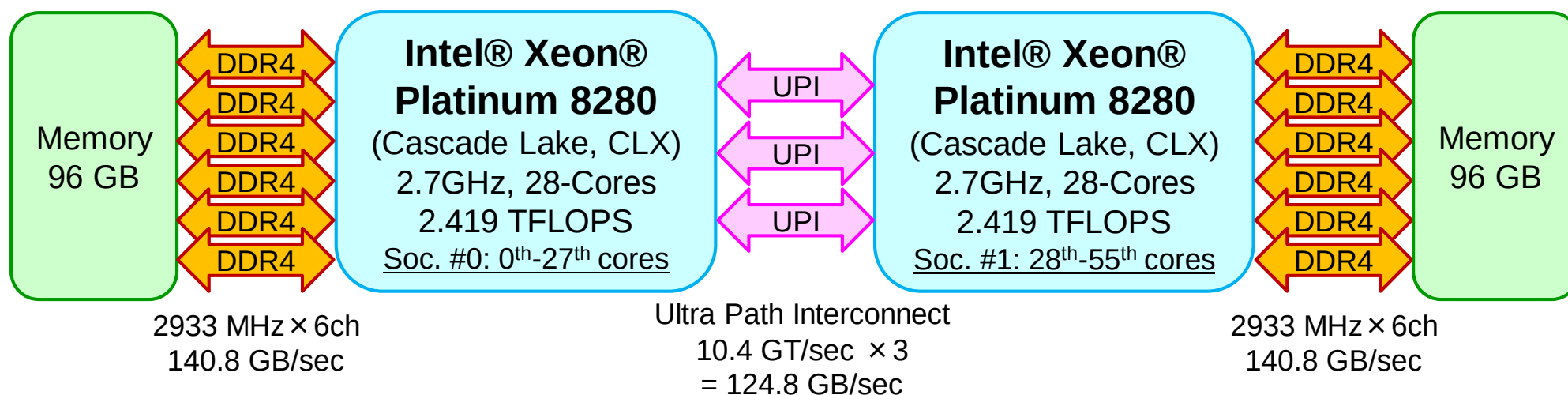


マルチコアクラスタ
Intel CLX
(Oakbridge-CX)

- 工学・ものづくり
- 地球科学・宇宙科学
- 材料科学
- エネルギー・物理学
- 情報科学: システム
- 情報科学: アルゴリズム
- 情報科学: AI
- 教育
- 産業利用
- 生物科学・生体力学
- バイオインフォマティクス
- 社会科学・経済学
- データ科学・データ同化

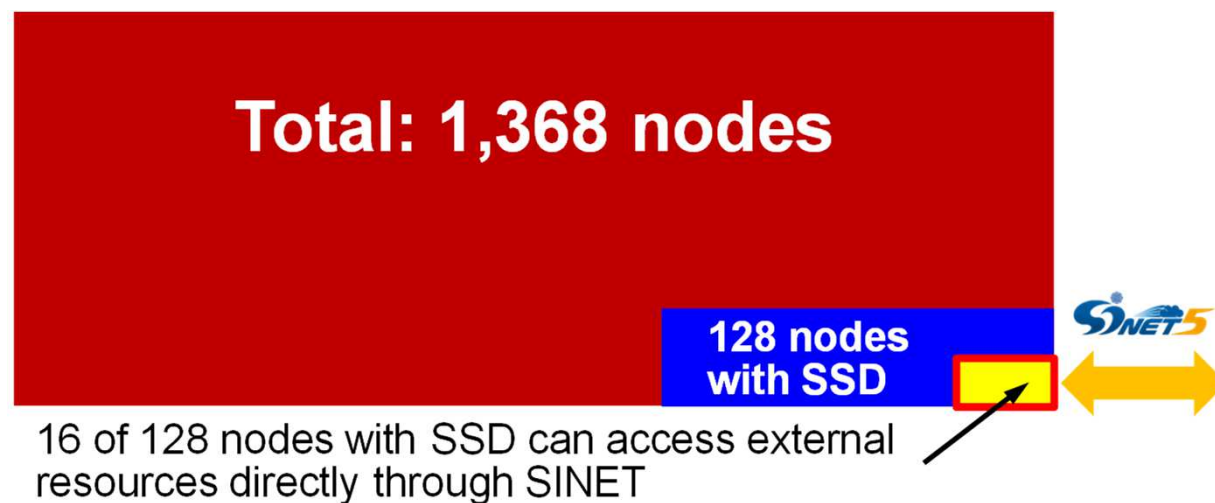


Intel Xeon Platinum 8280 (Cascade Lake, CLX)



Oakbridge-CX (OBCX)

- Intel Xeon Platinum 8280 (Cascade Lake, CLX), Fujitsu
 - 1,368 nodes, 6.61 PF peak, 385.1 TB/sec, 4.2+ PF for HPL
#60 in 55th Top500
 - 高速キャッシュ: 128ノードのSSD搭載: Intel SSD, BeeGFS  **BeeGFS**®
 - 1.6 TB/node, 3.20/1.32 GB/s/node for R/W, 200+TBの高速ファイルシステム
 - ステージング, チェックポイント, データ科学アプリケーション
 - 128ノードのうち16ノードはSINET経由で外部リソース(サーバー, ストレージ, センサーネットワーク)に直接接続
 - Network: Intel Omni-Path, 100 Gbps, Full Bi-Section
 - Storage: DDN EXAScaler (Lustre) 12.4 PB, 193.9 GB/sec
 - Power Consumption: 950.5 kVA
- 2019年7月1日運用開始



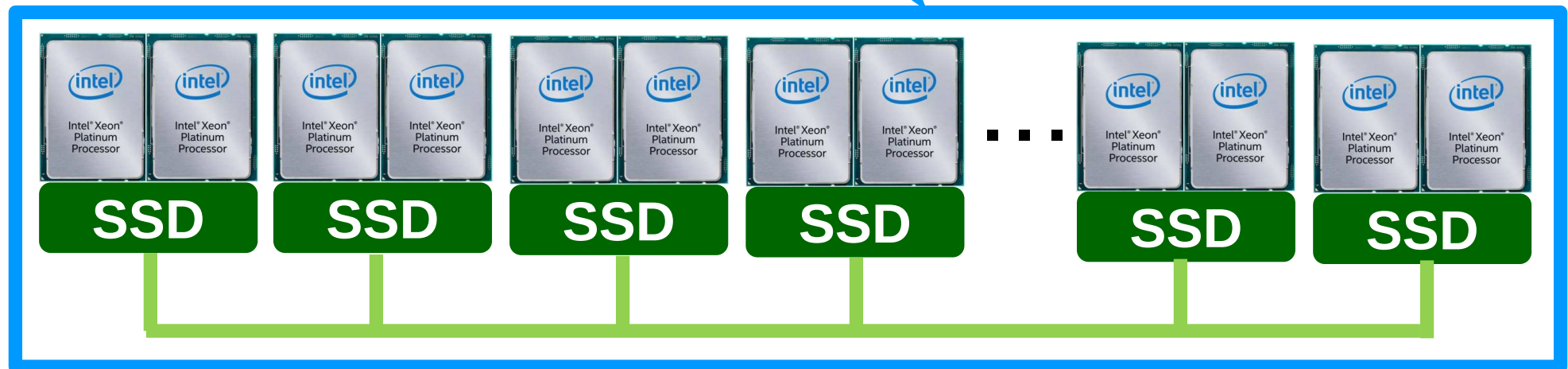
128xSSD's: Single Fast File System

Total: 1,368 nodes

128 nodes
with SSD

16 of 128 nodes with SSD can access external
resources directly through SINET

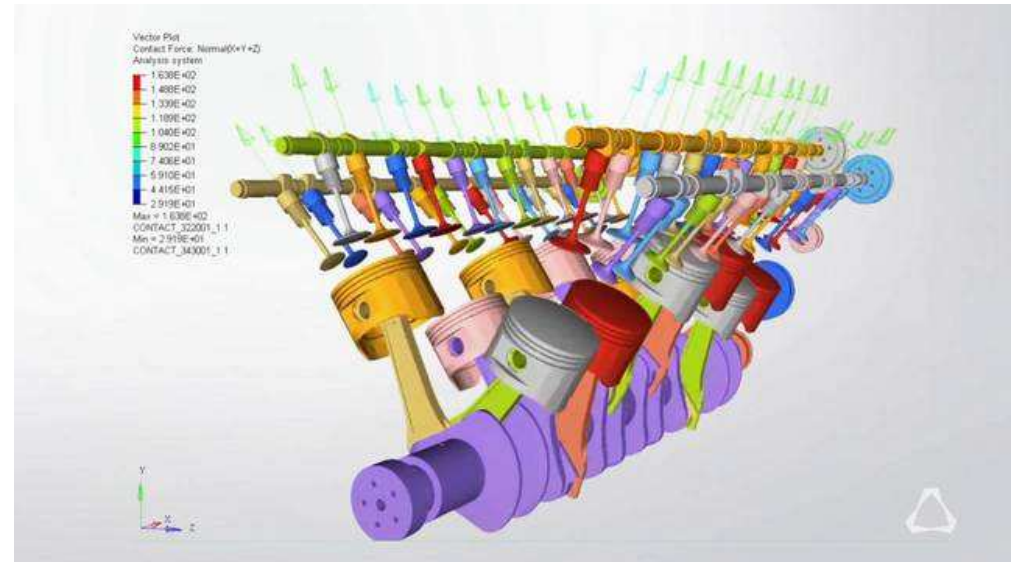
SINET5



Single Fast File System with 200+ TB

Altair Hyperworks

- 汎用CAEコード群
- 現状では大学教職員・学生のみ無料で利用可能
 - Reerbush
 - OBCX
- 公的研究機関，企業利用者はライセンスを取得すれば利用可能



- PC上のソフトウェア類の準備
- 東大情報基盤センターのスパコン
- スパコンへのログイン
- ログインしたら・・・

スパコンへのログイン

- SSH公開鍵認証 (SSH Public Key Authentication, SSH=Secure Shell)に基づく
 - パスワード認証よりも安全, と言われている
- 手順
 - **Windows: Cygwinを立ち上げる, Mac・Unix: Terminal起動**
 - ①PC上で鍵(秘密鍵, 公開鍵)を生成する
 - 秘密鍵, 公開鍵
 - **パスフレーズ (Passphrase)**: 鍵認証のためのパスワード
 - **「空のパスフレーズ (empty passphrase)」は禁止!**
 - ②スパコンポータルサイトにログインする
 - センターから供給された**利用者ID (t00XYZ)**と「初期パスワード」を使用
 - ポータルサイトにログイン後, **パスワード (Password)**変更を求められる, 字数, 使用文字等に色々規則があるので注意すること
 - ③スパコンポータルサイトに「公開鍵」を登録する
 - ④PCからsshによってスパコンにログインする

①PC上で鍵(秘密鍵, 公開鍵)を生成(1/3)

```
$ ssh-keygen -t rsa
```

Generating public/private rsa key pair.

Enter file in which to save the key (/home/user/.ssh/id_rsa):

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Your identification has been saved in /home/user/.ssh/id_rsa.

Your public key has been saved in /home/user/.ssh/id_rsa.pub.

The key fingerprint is:

SHA256:vt880+PTcscHk0yabvxGjeRsMWLAWds+ENsDcReNwKo nakajima@KNs-NEW-VAIO

The key's randomart image is:

```
+---[RSA 2048]---+
|             |
|  . 0=00. 0+  |
|  + 0. . . .  |
|  .+0+.       |
|  +0B.        |
| So *0*       |
| .E   B. o    |
|  . . = .  o  |
|  . =0B o  +  |
|  .+0+*0 . .  |
|             |
+---[SHA256]-----+
```

操作手順

- `ssh-keygen -t rsa <Return>`
- `<Return>`
- `<Return>`
- `<Return>`
- 「空のPassphrase」は厳禁

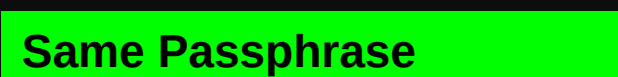
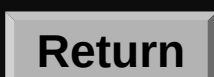
①PC上で鍵(秘密鍵, 公開鍵)を生成(1/3)

```
$ ssh-keygen -t rsa
```

Generating public/private rsa key pair.

Enter file in which to save the key (/home/user/.ssh/id_rsa): 

Enter passphrase (empty for no passphrase):  

Enter same passphrase again:  

Your identification has been saved in /home/user/.ssh/id_rsa.

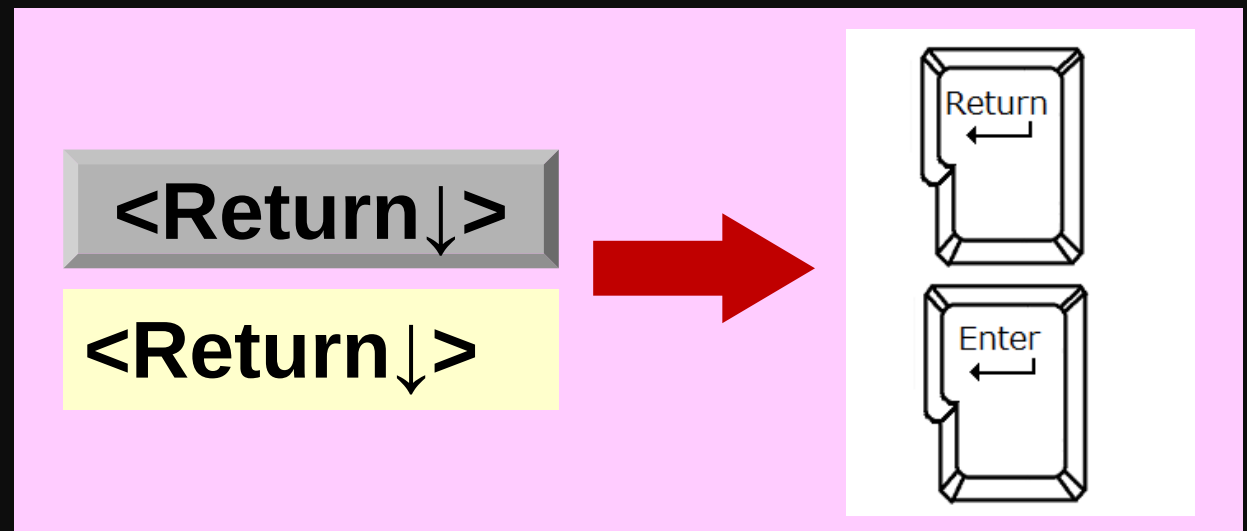
Your public key has been saved in /home/user/.ssh/id_rsa.pub.

The key fingerprint is:

SHA256:vt880+PTcscHk0yabvxGjeRsMWLAWds+ENsDcReNwKo nakajima@KNs-NEW-VAIO

The key's randomart image is:

```
+---[RSA 2048]---+
|             |
|  . 0=00. 0+  |
|  + 0. . . .  |
|  .+0+.      |
|  +oB.       |
| So *o*      |
| .E  B. o    |
|  . = .  o   |
|  . =oB o  +  |
|  .+o+*0 . .  |
+---[SHA256]-----+
```



①PC上で鍵(秘密鍵, 公開鍵)を生成(1/3)

```
$ ssh-keygen -t rsa
```

Generating public/private rsa key pair.

Enter file in which to save the key (/home/user/.ssh/id_rsa):

Return

Enter passphrase (empty for no passphrase): Your Favorite Passphrase

Return

Enter same passphrase again: Same Passphrase

Return

Your identification has been saved in /home/user/.ssh/id_rsa.

Your public key has been saved in /home/user/.ssh/id_rsa.pub.

The key fingerprint is:

SHA256:vt880+PTcscHk0yabvxGjeRsMWLAWds+ENsDcReNwKo nakajima@KNs-NEW-VAIO

The key's randomart image is:

```
+---[RSA 2048]---+
|                |
|  . 0=00. 0+    |
| + 0...  .      |
|  +0+         |
|  +0B         |
| So *0*        |
| .E  B. 0      |
| .. = . 0      |
|  . =0B 0  +   |
|  +0+*0 ..     |
|                |
+---[SHA256]-----+
```

操作手順

- `ssh-keygen -t rsa <Return>`
- `<Return>`
- `お好きなPassphrase <Return>`
- `同じPassphrase <Return>`
- 「空のPassphrase」は厳禁

①PC上で鍵(秘密鍵, 公開鍵)を確認(2/3)

```
$ cd .ssh
```

```
$ ls
```

```
id_rsa
```

⇒秘密鍵 (Private Key)

```
id_rsa.pub
```

⇒公開鍵 (Public Key)

```
$ cat id_rsa.pub
```

```
ssh-rsa
```

```
AAAAB3NzaC1yc2EAAAADAQABAAQDA6Inm0YYaCrWjQDukjiNEfdW8veUwJyZtEI3oDu0A28  
eey6p0wbtI7JB09xnI1707HG4yYv0M81+/nIAHy5tAfJly0dsPzjTgdTBLdgi3cSf5pWEY6U96  
yaEr0Ei8Wge1HkXrhcwUjGDVTzvT0Refe6zLdRziL/KNmmesSQfR5lsZ/ihsjMgFxGaKsHHq/I  
ErCtHIIf9V/Ds2yj6vkAaWH6asBn+ZsRiRFvwHPhkYAnp/j3LY6b8Qfqg0p4WZRenh/HgySWT  
YIGi8x67VzMaUIm9qIK0QFMCaK2rivX1fmbwyWJ/vrWDqiek6YXoxLDu+GPeQ4CPvxJcZnqF9g  
f3 nakajima@KNs-NEW-VAIO
```

①公開鍵をコピー(3/3)

```
$ cd .ssh
```

```
$ ls
```

```
id_rsa  
id_rsa.pub
```

```
$ cat id_rsa.pub
```

操作手順

- `cat id_rsa.pub <Return>`
- “ssh-rsa”にカーソルを合わせ
- 最後の行の”f3”までを選択して「Copy」によって記憶
- 最後の「nakajima@KNs-NEW-VAIO」まで含んでも良いが、ここに漢字が含まれていると登録に失敗する

```
ssh-rsa
```

```
AAAAB3NzaC1yc2EAAAADAQABAAQDA6Inm0YYaCrWjQDukjiNEfdW8veUwJyZtEI3oDu0A28  
eey6p0wbtI7JB09xnI1707HG4yYv0M81+/nIAHy5tAfJly0dsPzjTgdTBLdgi3cSf5pWEY6U96  
yaEr0Ei8Wge1HkXrhcwUjGDVTzvT0Refe6zLdRziL/KNmmesSQfR5lsZ/ihsjMgFxGaKsHHq/I  
ErCtHIIIf9V/Ds2yj6vkAaWH6asBn+ZsRiRFvwHPhkYAnp/j3LY6b8Qfqg0p4WZRenh/HgySWT  
YIGi8x67VzMaUlm9qIKOQFMCaK2rivX1fmbwyWJ/vrWDqiek6YXoxLDu+GPeQ4CPvxJcZnqF9g  
f3 nakajima@KNs-NEW-VAIO
```

②スパコンポータルサイトにログイン(1/3)

情報基盤センターから送付されたファイル

Oakbridge-CX_workshop_t00100.pdf - Adobe Acrobat Reader DC

ファイル(F) 編集(E) 表示(V) ウィンドウ(W) ヘルプ(H)

ホーム ツール 2019年源泉徴収.pdf ... Oakbridge-CX_work... × ログイン

共有

(教育利用)

Oakbridge-CX 利用登録のお知らせ

Notification of Your Account of Supercomputer System (Oakbridge-CX)

様

下記のとおり登録しましたのでお知らせします。

2020年4月8日
東京大学情報基盤センター長
Information Technology Center, The University of Tokyo

公印
省略

プロジェクト名称	お試しアカウント付き並列プログラミング講習会		プロジェクトコード Project code	gt00
利用期間	2020年4月 ~ 2020年6月			

利用者番号 User ID	t00XYZ	初期パスワード※ Password	01234567
------------------	--------	----------------------	----------

情報基盤センターから送付された利用者ID(t00XYZ)

情報基盤センターから送付された初期パスワード(8桁)

14:58
2020/04/12

各システム情報・利用支援ポータルサイト

- Reedbush 日本語
 - <https://www.cc.u-tokyo.ac.jp/supercomputer/reedbush/service/>
 - <https://reedbush-www.cc.u-tokyo.ac.jp/>
- Reedbush English
 - <https://www.cc.u-tokyo.ac.jp/en/supercomputer/reedbush/service/>
 - <https://reedbush-www.cc.u-tokyo.ac.jp/>
- Oakforest-PACS (OFP) 日本語
 - <https://www.cc.u-tokyo.ac.jp/supercomputer/ofp/service/>
 - <https://ofp-www.jcahpc.jp/cgi-bin/hpcportal.ja/index.cgi>
- Oakforest-PACS (OFP) English
 - <https://www.cc.u-tokyo.ac.jp/en/supercomputer/ofp/service/>
 - <https://ofp-www.jcahpc.jp/cgi-bin/hpcportal.en/index.cgi>
- Oakbridge-CX (OBCX) English
 - <https://www.cc.u-tokyo.ac.jp/supercomputer/obcx/service/>
 - <https://obcx-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal.ja/index.cgi>
- Oakbridge-CX (OBCX) English
 - <https://www.cc.u-tokyo.ac.jp/en/supercomputer/obcx/service/>
 - <https://obcx-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal.en/index.cgi>

②スパコンポータルサイトにログイン(2/3)

<https://obcx-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal.ja/index.cgi>

The screenshot shows a web browser window displaying the 'Oakbridge-CX 利用支援ポータル' (Oakbridge-CX User Support Portal) login page. The browser's address bar shows the URL <https://obcx-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal.ja/index.cgi>. The page title is 'Oakbridge-CX 利用支援ポータル'. Below the title, there is a language selector '[English / Japanese]' and a 'ログイン' (Login) link. The main content area is titled 'ログイン' (Login) and contains a form with the following fields and buttons:

- A red box highlights the 'ユーザ名:' (Username) field, which is currently empty.
- An orange box highlights the 'パスワード:' (Password) field, which is currently empty, and the 'リセット' (Reset) button below it.
- A 'ログイン' (Login) button is located to the right of the username field.

Below the login form, there are two text boxes providing instructions:

- A pink box contains the text: '情報基盤センターから送付された利用者ID (t00XYZ)' (User ID sent from the Information Infrastructure Center (t00XYZ)).
- A yellow box contains the text: '情報基盤センターから送付された初期パスワード' (Initial password sent from the Information Infrastructure Center).

The bottom of the browser window shows the Windows taskbar with various application icons and the system clock displaying '20:47' on '2020/04/15'. The footer of the page reads 'Copyright 2019 FUJITSU LIMITED'.

②初期パスワードの変更(3/3)

The screenshot shows a web browser window with the URL https://obcx-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal_u.ja/index.cgi. The page title is "Oakbridge-CX 利用支援ポータル". On the left is a sidebar menu with items like "お知らせ", "SSH公開鍵登録", "メール転送設定", "パスワード変更", "トークン表示", "ディスク使用量表示", "プリポスト予約", "ドキュメント閲覧", and "OSS". The main content area is titled "パスワード変更" and contains a form with three input fields: "現在のパスワード" (highlighted in yellow), "新しいパスワード" (highlighted with a blue border), and "新しいパスワード(再入力)" (also highlighted with a blue border). A "変更" button is at the bottom of the form. A yellow callout box points to the first field with the text "情報基盤センターから送付された初期パスワード". A light blue callout box points to the second and third fields with the text "変更後のパスワードを入力(2回)". Below the form, a yellow box titled "パスワード規約" lists the password requirements.

Oakbridge-CX 利用支援ポータル

ログアウト

パスワード変更

本機能で変更可能なパスワードは、Oakbridge-CXシステムの利用支援ポータル用パスワードです。

現在のパスワード

新しいパスワード

新しいパスワード(再入力)

変更

情報基盤センターから送付された初期パスワード

変更後のパスワードを入力(2回)

パスワード規約

- 8文字以上, 現在と3文字以上異なる
- 2世代前までと異なる
- 英字(小文字, 大文字), 数字, 特殊文字各1字以上
- Linux辞書に登録されている語は不可
- 全角文字不可

Copyright 2019 FUJITSU LIMITED

18:38
2020/04/14

③ 公開鍵登録(id_rsa.pub)

Oakbridge-CX 利用支援ポータル

SSH公開鍵登録

公開鍵を削除しました。

登録方式

☒ 直接入力

☐ ファイルアップロード

ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDa6InmOYYaCrWjQDukjiNEfdW8veUwJyZtEI3oDu0
A28eey6p0wbtI7JB09xnI17O7HG4yYvOM81+/nIAHy5tAfJly0dsPzjTgdTBLdgi3cSf5pWEY6U9
6yaEr0Ei8Wge1HkXrhcwUjGDVTzvt0Refe6zLdRziL/KNmmsSQfR5lsZ/ihsjMgFxGaKsHHq
/IErCtHIIIf9V/Ds2yj6vkAaWH6asBn+ZsRiRFvwHPhkYAnp/j3LY6b8Qfqg0p4WZRenh
/HgySWTYIGi8x67VzMaUlm9qlK0QFMCaK2rivX1fmbwyWJ
/vrWDqiek6YXoxLDu+GPeQ4CPvxJcZnqF9gf3

登録

1. 「SSH公開鍵登録」を選択
2. 先ほどCopyした「id_rsa.pub」を貼り付ける
3. 「登録」をクリック

Copyright 2019 FUJITSU LIMITED

22:34
2020/04/15

③ うまく行かない場合は(id_rsa.pub)を 直接アップロード(3/4)

ファイルのアップロード

← → ↕ ↑ PC > ローカル ディスク (C:) > cygwin64 > home > nakajima > .ssh .sshの検索

整理 ▾ 新しいフォルダー

名前	更新日時	種類	サイズ
id_dsa	2012/12/12 17:00	ファイル	1 KB
id_rsa	2013/11/29 17:36	ファイル	2 KB
known_hosts	2020/05/08 13:30	ファイル	8 KB
known_host_keys	2017/07/26 17:37	1~ ファイル	1 KB
known_hosts.xz	2018/11/29 17:06	ファイル	6 KB

¥cygwin64¥home¥XXXX¥.ssh¥id_rsa.pub

④PCからログイン(1/2)

```
$ ssh t00XYZ@obcx.cc.u-tokyo.ac.jp
```

Return

The authenticity of host 'obcx.cc.u-tokyo.ac.jp' can't be established.
ECDSA key fingerprint is SHA256:/XXXXX ...

Are you sure you want to continue connecting (yes/no/[fingerprint])? yes

Return

Warning: Permanently added 'obcx.cc.u-tokyo.ac.jp' to the list of known hosts.

Enter passphrase for key '/home/nakajima/.ssh/id_rsa': Your Passphrase

Return

1. `ssh t00XYZ@obcx.cc.u-tokyo.ac.jp` <Return>
2. `yes` <Return>
3. 鍵生成時に打ち込んだPassphrase <Return>

④PCからログイン(1/2)(2回目以降)

\$ ssh t00XYZ@obcx.cc.u-tokyo.ac.jp

Enter passphrase for key '/home/nakajima/.ssh/id_rsa':

1. ssh t00XYZ@obcx.cc.u-tokyo.ac.jp <Return>
2. <Return>

④PCからログイン(2/2)

Last login: Sun Apr 12 15:05:47 2020 from obcx01.cc.u-Tokyo.ac.jp

Oakbridge-CX Information

Date: Apr. 03, 2020

Welcome to Oakbridge-CX system

* Operation Schedule

04/24 (Fri)	09:00 - 04/24 (Fri)	20:00	System Maintenance
04/24 (Fri)	20:00 -		Normal Operation

For more information about this service, see

<https://www.cc.u-Tokyo.ac.jp/supercomputer/schedule.php>

* How to use

Users Guide can be found at the User Portal (<https://obcx-www.cc.u-Tokyo.ac.jp/>).

If you have any questions, please refer to the following URL and contact us:

<https://www.cc.u-tokyo.ac.jp/supports/contact/>

* Updated OBCX Users Guide

10/01 (Tue): v1.0

ログインに成功したら、今後のメンテナンスのスケジュールなどが表示される

このURL(質問)にはアクセスしないこと

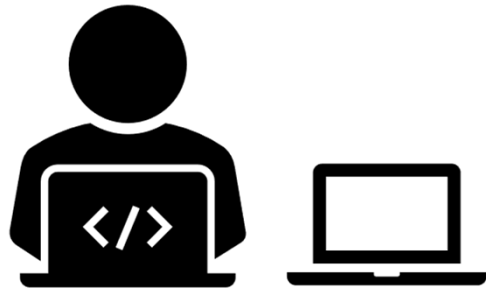
Warning: /home/t00100/.forward does not exist.

Set your email address on the User Portal [<https://obcx-www.cc.u-tokyo.ac.jp/>]

[t00XYZ@obcx01 ~]\$

SSH公開鍵認証の手順(1/4)

①PC上での秘密鍵・公開鍵作成



```
$> ssh-keygen -t rsa
```

id_rsa

秘密鍵/Private Key

+ Passphrase

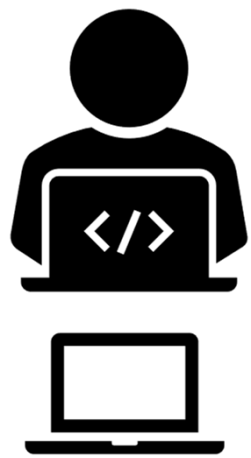
id_rsa.pub

公開鍵/Public Key



SSH公開鍵認証の手順(2/4)

②スパコンポータルサイトへのログイン



tXYZZZ
+ Password

Portal Site
OBCX



id_rsa
秘密鍵/Private Key

+ Passphrase

id_rsa.pub
公開鍵/Public Key



tABCCC
+ Password

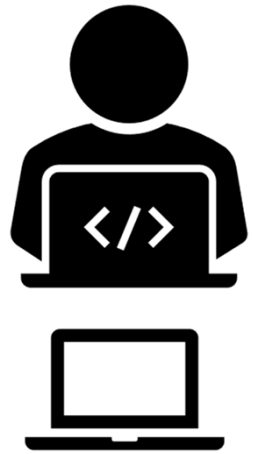
Portal Site
OFP



SSH公開鍵認証の手順(3/4)

③公開鍵(id_rsa.pub)の登録

同じ公開鍵を複数のスパコンに登録可能



id_rsa
秘密鍵/Private Key
+ Passphrase

id_rsa.pub
公開鍵/Public Key

Portal Site
OBCX



id_rsa.pub
公開鍵/Public Key

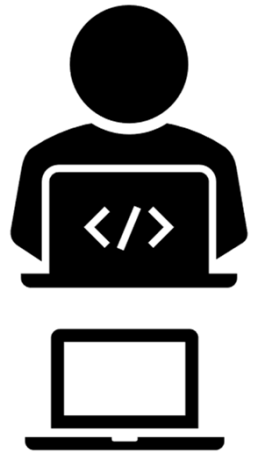
Portal Site
OFP



SSH公開鍵認証の手順(3/4)

③公開鍵(id_rsa.pub)の登録

同じ公開鍵を複数のスパコンに登録可能



Portal Site
OBCX



Portal Site
OFP



id_rsa
秘密鍵/Private Key

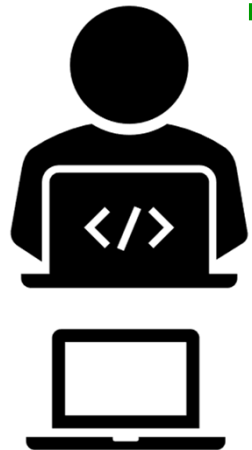
+ Passphrase



SSH公開鍵認証の手順(4/4)

④PCからスパコンへのログイン

秘密鍵(id_rsa) + Passphrase



```
$> ssh tXYZZZ@obcx.cc.u-tokyo.ac.jp
```

id_rsa

秘密鍵/Private Key

+ Passphrase



```
$> ssh tABCCC@ofp.jcahpc.jp
```

id_rsa

秘密鍵/Private Key

+ Passphrase



SSH Public Key Authentication

SSH公開鍵認証

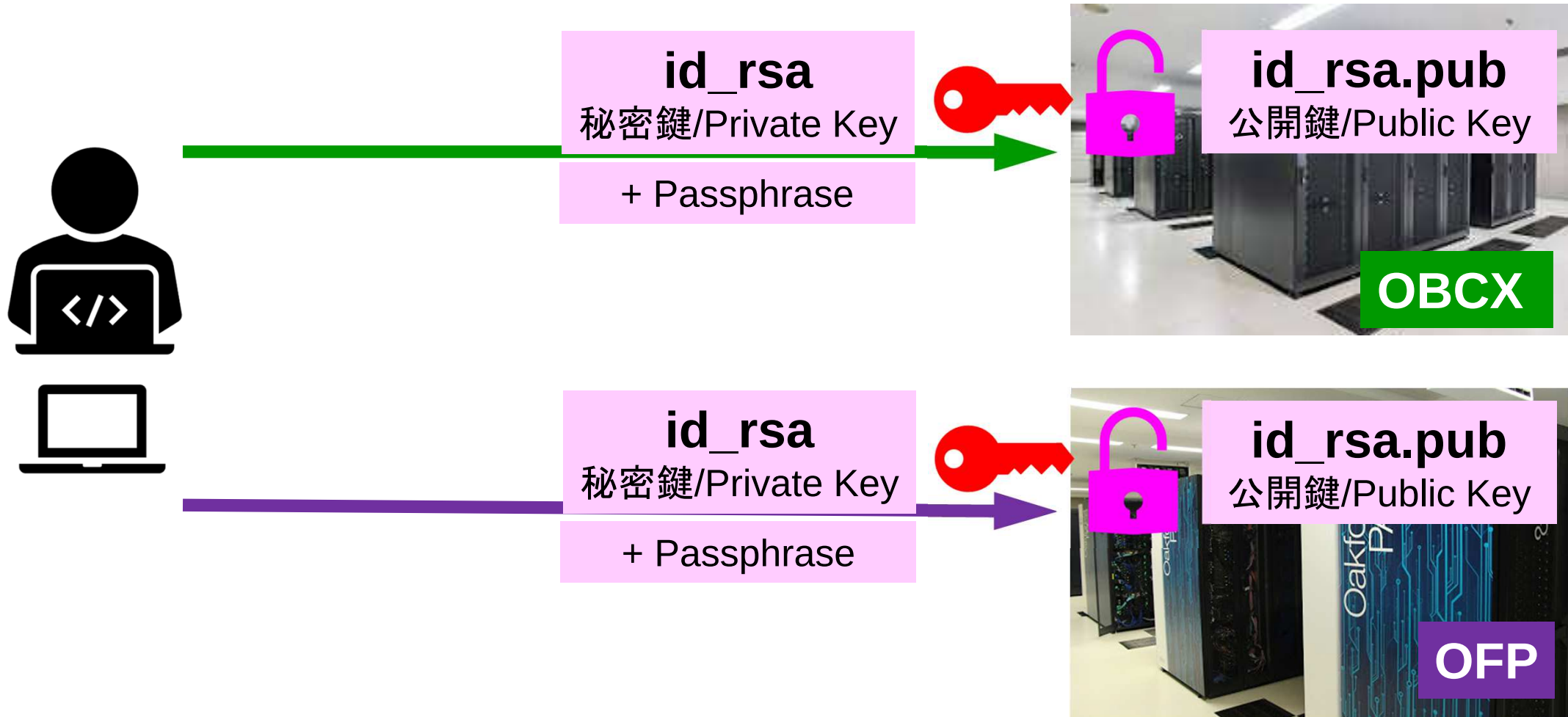
SSH= Secure Shell

- **id_rsa**
 - Private Key(秘密鍵) : PC上
 - 文字通り「秘密」にしておくこと
 - 他の人に送ってはいけない
 - 基本的には作成した場所からコピーしたり移動することもしないこと
- **id_rsa.pub**
 - Public Key(公開鍵) : スパコン上
 - コピー可能, 他の人にe-mailで送ることも可能
- **もし複数のPCからスパコンにログインする場合は, 各PCごとに「公開鍵・秘密鍵」のペアをssh-keygenによって作成**
 - **各スパコンに複数の公開鍵を登録することは可能**
 - **スパコン上の公開鍵のうちの 하나가PC上の「秘密鍵 + Passphrase」とマッチすると確認されるとログインできる**

SSH公開鍵認証の手順(4/4)

④PCからスパコンへのログイン

秘密鍵(id_rsa) + Passphrase



複数のPCからスパコンへログインする場合には各PCで「公開鍵・秘密鍵」のペア作成

```
$> ssh-keygen -t rsa
```



id_rsa
秘密鍵/Private Key

+ Passphrase

id_rsa.pub
公開鍵/Public Key



id_rsa
秘密鍵/Private Key

+ Passphrase

id_rsa.pub
公開鍵/Public Key

Portal Site
OBCX



Portal Site
OFP



スパコンには複数の公開鍵を登録できる

Oakbridge-CX 利用支援ポータル

https://obcx-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal_u.ja/index.cgi

Oakbridge-CX 利用支援ポータル

ログアウト

お知らせ

SSH公開鍵登録

メール転送設定

パスワード変更

トークン表示

ディスク使用量表示

プリポスト予約

ドキュメント閲覧

OSS

SSH公開鍵登録

公開鍵を登録しました。

登録されている公開鍵	ssh-rsa AAAAB3NzaC.....JcZnqF9gf3	表示	削除
	ssh-rsa AAAAB3NzaC.....pWGVie6w==	表示	削除

登録方式

☒ 直接入力

☐ ファイルアップロード

Copyright 2019 FUJITSU LIMITED

ページ内検索

すべて強調表示(A) 大文字/小文字を区別(C) 発音区別符号を区別(I) 単語単位(W)

22:36 2020/04/15

スパコンには複数の公開鍵を登録できる

```
$ cd .ssh  
$ ls authorized_keys
```

```
authorized_keys
```

.ssh/authorized_keysには登録された公開鍵が格納されている。このファイルの後ろに新たな公開鍵(new_public.key)を付け加えることができる。

```
$ cat authorized_keys
```

```
ssh-rsa  
HGAB3NzaC1yc2EAAAABIwAAAQEA1r0Hr8M1JIJB02n9S0GQm0xzGCwh3PpcJo7Z8oDr6HCAXhbK  
zHA0ibRMJFCwDJCRGNJIYiHEYHWzouuXGNa9teso7aXYkq2Pxb076C60ZCPoLqf/jQRqnUSnjHJ4  
UgmDdIQWaAks+q/2Ex0wjBB6GZmaHGijTxim0FGiM1DI780HkHC8pFzjvP2kT9yRvykv0VvIG10V  
Yi+5CawYfuR0iRBjfUS47RS0ICzjNP20pY057DUCf0v+/8B1+l1wiIbjKQHjuNp5XucIFfFdGaxf  
JchD/sB5sRxtYfz80xzwGmN8pVecpUjd//xAqdYYHmLAKUE2oH8MnBIRybpWGVie6w64
```

```
ssh-rsa  
AAAAB3NzaC1yc2EAAAADAQABAAQDA6Inm0YYaCrWjQDukjiNEfdW8veUwJyZtEI3oDu0A28ee  
y6p0wbtI7JB09xnI1707HG4yYvOM81+/nIAHy5tAfJly0dsPzjTgdTBLdgi3cSf5pWEY6U96yaEr  
0Ei8Wge1HkXrhcwUjGDVTzvT0Refe6zLdRziL/KNmmesSQfR5lsZ/ihsjMgFxGaKsHHq/IErCtHI  
IIf9V/Ds2yj6vkAaWH6asBn+ZsRiRFvwHPhkYAnp/j3LY6b8Qfqg0p4WZRenh/HgySWTYIGi8x67  
VzMaUIm9qIK0QFMCaK2rivX1fmbwyWJ/vrWDqiek6YXoxLDu+GPeQ4CPvxJcZnqF9gf3
```

```
$ cp authorized_keys tmp  
$ cat tmp new_public.key > authorized_keys
```


各スパコンに複数の鍵を登録する

```
$> ssh-keygen -t rsa
```



id_rsa
秘密鍵/Private Key

+ Passphrase

id_rsa.pub
公開鍵/Public Key



id_rsa
秘密鍵/Private Key

+ Passphrase

id_rsa.pub
公開鍵/Public Key

Portal Site
OBCX



id_rsa.pub
公開鍵/Public Key



id_rsa.pub
公開鍵/Public Key

Portal Site
OFP



id_rsa.pub
公開鍵/Public Key



id_rsa.pub
公開鍵/Public Key

- PC上のソフトウェア類の準備
- 東大情報基盤センターのスパコン
- スパコンへのログイン
- ログインしたら・・・

PCからログイン

```
$ ssh t00XYZ@obcx.cc.u-tokyo.ac.jp  
Enter passphrase for key '/home/nakajima/.ssh/id_rsa: Your Passphrase Return
```

1. `ssh t00XYZ@obcx.cc.u-tokyo.ac.jp` <Return>
2. **鍵生成時に打ち込んだPassphrase** <Return>

ログインしたら

```
$ pwd
```

```
/home/t00XYZ
```

```
$ cd /work/gt00/t00XYZ
```

```
$ pwd
```

```
/work/gt00/t00XYZ
```

```
$ cd
```

```
$ pwd
```

```
/home/t00XYZ
```

1. ログインしたら「/home/t00XYZ」に入る
2. /homeは容量が少ないので「/work/gt00/t00XYZ」に移動すること
3. 「cd」でホームに戻れます

Copy: PC to OBCX

```
$ scp ./a.dat t00XYZ@obcx.cc.u-tokyo.ac.jp:
```

PC上のCurrent Directoryにある「a.dat」をOBCXのホームディレクトリ（/home/t00XYZ）の下にコピーする

```
$ scp ./a.dat t00XYZ@obcx.cc.u-tokyo.ac.jp:/work/gt00/t00XYZ/test/
```

PC上のCurrent Directoryにある「a.dat」をOBCX上のディレクトリ「/work/gt00/t00XYZ/test」の下にコピーする

```
$ scp -r ./testL t00xyz@obcx.cc.u-tokyo.ac.jp:
```

PC上のCurrent Directory下にあるディレクトリ「testL」及びその中身をOBCXのホームディレクトリ（/home/t00XYZ）の下にコピーする

```
$ scp -r ./testL t00xyz@obcx.cc.u-tokyo.ac.jp:/work/gt00/t00XYZ/test
```

PC上のCurrent Directory下にあるディレクトリ「testL」及びその中身をOBCX上のディレクトリ「/work/gt00/t00XYZ/test」の下にコピーする

Copy: OBCX to PC

```
$ scp t00XYZ@obcx.cc.u-tokyo.ac.jp:~/a.dat ./
```

OBCXのホームディレクトリ（/home/t00XYZ）下にある「a.dat」を、PC上のCurrent Directory下にコピーする

```
$ scp t00XYZ@obcx.cc.u-tokyo.ac.jp:/work/gt00/t00XYZ/test/a.dat ./
```

OBCX上のディレクトリ「/work/gt00/t00XYZ/test/」下にある「a.dat」をPC上のCurrent Directoryにコピーする

```
$ scp -r t00XYZ@obcx.cc.u-tokyo.ac.jp:~/L1 ./
```

OBCXのホームディレクトリ（/home/t00XYZ）下にあるディレクトリ「/home/t00XYZ/L1」とその中身を、PC上のCurrent Directoryにあるディレクトリ「L1」にコピーする

```
$ scp -r t00XYZ@obcx.cc.u-tokyo.ac.jp:/work/gt00/t00XYZ/test/L1 ./
```

OBCX上のディレクトリ「/work/gt00/t00XYZ/test/L1」とその中身をPC上のCurrent Directory下にあるディレクトリ「L1」にコピーする

ポータルサイトでのマニュアル等閲覧(1/2)

Oakbridge-CX 利用支援ポータル

https://obcx-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal_u.ja/index.cgi

Oakbridge-CX 利用支援ポータル

ログアウト

お知らせ

SSH公開鍵登録

メール転送設定

パスワード変更

トークン表示

ディスク使用量表示

プリポスト予約

ドキュメント閲覧

OSS

ドキュメント閲覧の利用について

Oakbridge-CX マニュアルの Web 閲覧サービスを利用するにあたっては、以下の禁止事項を遵守していただきます。

- 核兵器又は生物化学兵器及びこれらを運搬するためのミサイル等の大量破壊兵器の開発、設計、製造、保管及び使用等の目的に利用しない。
- スーパーコンピュータの利用が認められた利用者本人のみが利用し、他者には利用させない。
- 本マニュアルの情報（印刷、コピーしたものを含む）を、利用者以外に開示または提供しない。
- 当センターが上記条項の違反、その他不正使用を検知した場合、当センターは利用者の Web 閲覧サービスの利用を直ちに停止することができる。また、利用者はこれに対して一切異議を唱えない。

上記禁止事項を

遵守する

ページ内検索

すべて強調表示(A) 大文字/小文字を区別(C) 発音区別符号を区別(I) 単語単位(W)

Copyright 2019 FUJITSU LIMITED

22:37 2020/04/15

ポータルサイトでのマニュアル等閲覧(2/2)

Oakbridge-CX 利用支援ポータル

https://obcx-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal_uja/index.cgi

Oakbridge-CX 利用支援ポータル

ログアウト

- お知らせ
- SSH公開鍵登録
- メール転送設定
- パスワード変更
- トークン表示
- ディスク使用量表示
- プリポスト予約
- ドキュメント閲覧**
- OSS

Oakbridge-CX 利用手引書

ドキュメント名	言語	最新更新日
Oakbridge-CX システム利用手引書	日本語	2019/10/01
Oakbridge-CX グループコース プロジェクト管理者用利用手引書	日本語	2019/07/03

製品マニュアル

インテルParallel Studio XE 2019

ドキュメント名	言語	最新更新日
スタートアップガイド	日本語 英語	2019/07/01
Fortranコンパイラ19.0 スタートアップガイド	日本語 英語	2019/07/01
C++コンパイラ19.0 スタートアップガイド	日本語 英語	2019/07/01

インテルMPIライブラリ 2019

ドキュメント名	言語	最新更新日
スタートアップガイド	英語	2019/07/01

インテルMKL 2019

ドキュメント名	言語	最新更新日
スタートアップガイド	日本語 英語	2019/07/01

Copyright 2019 FUJITSU LIMITED

ページ内検索

すべて強調表示(A) 大文字/小文字を区別(C) 発音区別符号を区別(I) 単語単位(W)

質問等は中島まで
相談窓口には連絡しないでください！