HOKUSAI BigWaterfall User's Guide

Version 1.6

Apr 6, 2020

Information Systems Division,

RIKEN

Revision History

Ver.	Issued on	Chapter	Reason for revision
1.0	Oct. 11, 2017	-	First edition
1.1	Oct. 12, 2017	1.5.1	Fix result of listcpu command
		1.6	Fix description of priority control
		3.1	Fix description of /data, /gwdata
		3.1.3	Add application form for using storage area
		5.2.2.2	Correct the values of resource group of ACSL (from 72hrs to 24hrs of maximum elapsed time of batch and gaussian)
		5.2.2.4	Fix configuration of resource group of BWMPC (from 30hrs to 48hrs of maximum elapsed time of special)
		5.4.2	Add step job submission with multiple scripts
		5.6	Fix examples of interactive job on ACSG from without GPUs to with GPUs
		7	Add description of User Portal
1.2	Apr. 4,2017	5.2.1.1	Fix configuration of resource unit settings(ACSG, ACSL) for project
		5.2.2.2	Fix configuration of resource group for ACS with Large memory (ACSL)
		5.2.2.3	Fix configuration of resource group for ACS with GPU (ACSG)
1.3	Sep, 20, 2018	5.2.1.1	Fix configuration of resource unit settings(ACSL) for project
		5.2.2.2	Fix configuration of resource group for ACS with Large memory (ACSL)
1.4	Jan, 28, 2019	5.2.2.2	Fix configuration of resource group(ansys) for ACS with Large memory (ACSL)
		5.2.2.3	Fix configuration of resource group(ansys) for ACS with GPU (ACSG)
1.5	Apr, 12, 2019	5.2.2.3	Change of usage of group(adf)
		5.2.2.5	Added available area of ADF
1.6	Apr, 6, 2020	-	Change due to GreatWave operation termination

Table of Contents

INTRODUCTION

This User's Guide explains how to use the supercomputer system HOKUSAI BigWaterfall introduced by RIKEN. All users who use this system are strongly recommended to read this document, as this is helpful to gain better understanding of the system.

The content of this document is subject to change as required. The latest version of this document is available from the following User Portal:

https://hokusai.riken.jp/

In addition, on User Portal, you can know how to execute the softwares available on the HOKUSAI BigWaterfall system, the versions of those softwares, and you can registrate ssh public key.

The User Portal and mailing lists are used for public announcement of the system's operation status.If you have any questions about how to use the HOKUSAI BigWaterfall system or need for further assistance, you can send messages in an email to:

hpc@riken.jp

Unauthorized copy, reproduction, or republication of all or part of this document is prohibited.

1. HOKUSAI BigWaterfall System

1.1 System Overview

The HOKUSAI BigWaterfall system consists of the following key components:

- Massively Parallel Computers (BWMPC)
- Application Computing Server with Large memory(GWACSL)
- Front end servers that provide the users with the application interface for the system
- Two types of storages with different purposes, one of which is the Online Storage and the other of which is the Hierarchical Storage

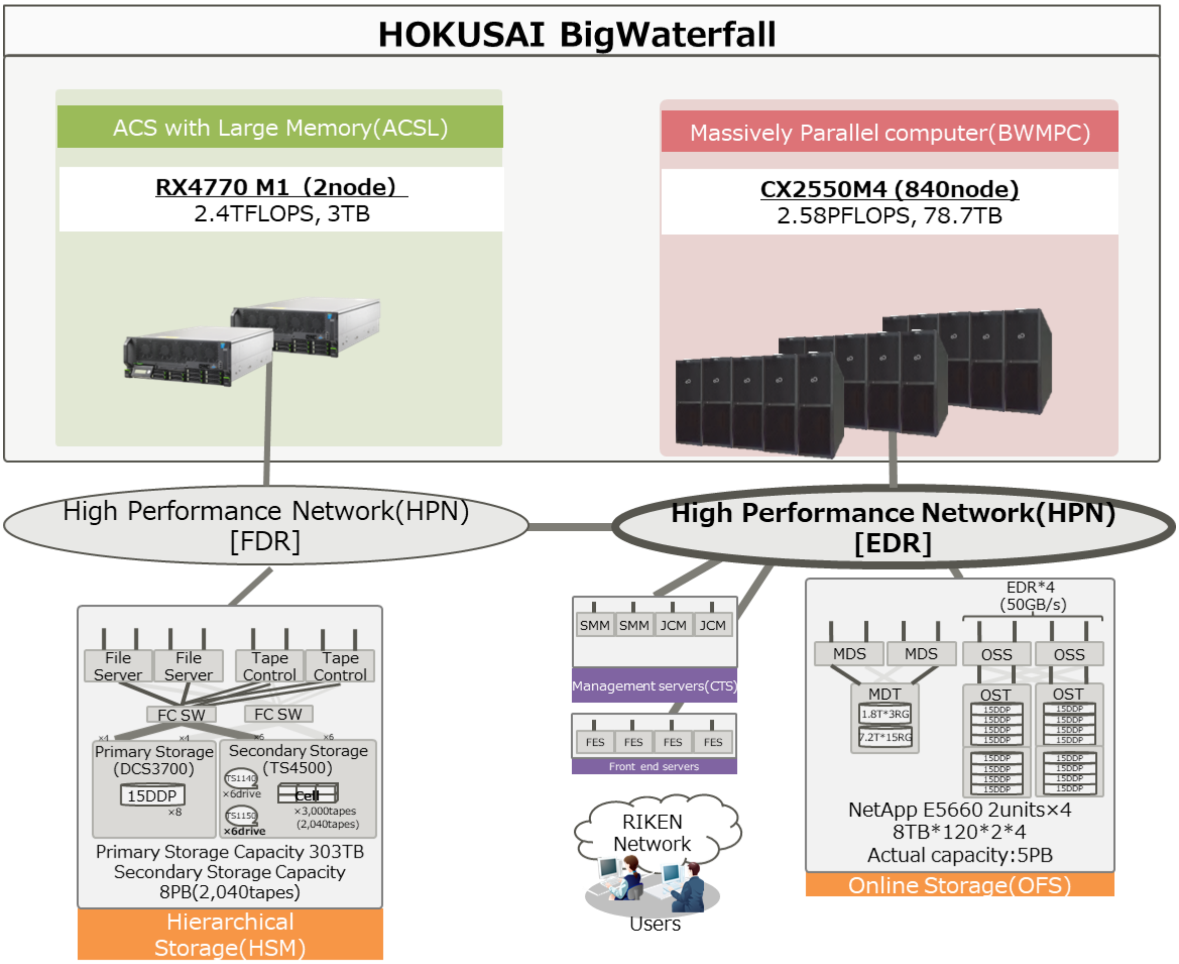


Figure 1-1 System diagram

The Massively Parallel Computer (BWMPC) comprises 840 nodes of CX2550 M4. Each node provides a theoretical peak performance of 3.07 TFLOPS and a memory capacity of 96 GB. The InfiniBand EDR of 12.6 GB/s is used to connect each node to enable high performance communication and file sharing.

The ACS with Large memory (GWACSL) comprises two nodes of PRIMERGY RX4770 M1. Each node provides a theoretical peak performance of 1.2 TFLOPS and a memory capacity of 1.5 TB. The InfiniBand FDR of 6.8 GB/s is used to connect each node to enable high performance communication and file sharing.

The storage environment consists of the Online Storage (OFS) and the Hierarchical Storage (HSM).

The Online Storage (OFS) is a high bandwidth online file system used for the users' home directories, the shared directories for projects and so on, and can be accessed from the Massively Parallel Computers, the Application Computing Servers with Large memory, and the front end servers. The total capacity is 5 PB.

The Hierarchical Storage (HSM) consists of the primary storage (cache disks) of 300 TB and the secondary storage (tape library devices) of 7.9 PB (uncompressed) and is the file system used to store large volumes of data files that should be retained for a long term. The users can read or write data to the tapes without manipulating the tape library devices.

You can access the HOKUSAI BigWaterfall system using ssh/scp for login/file transfer, or using HTTPS for the User Portal. On the front end servers, you can mainly do the following:

- create and edit programs
- compile and link programs
- manage batch jobs and launch interactive jobs
- tune and debug programs

1.2 **Hardware Overview**

1.2.1 Massively Parallel Computer (BWMPC)

- Computing performance
  CPU: Intel Xeon Gold 6148 (2.4GHz) 840 units (1,680 CPUs, 33,600 cores)
  Theoretical peak performance: 2.58 PFLOPS (2.4 GHz x 32 floating-point operations x 20 cores x 1,680 CPUs)
- Memory
  Memory capacity: 78.7 TB (96 GB x 840 units)
  Memory bandwidth: 255GB/s/CPU
  Memory bandwidth/FLOP: 0.08Byte/FLOP
- Local disk
  Disk capacity: 100.8TB (120GB x 30 units)
- Interconnect
  InfiniBand EDR
  Theoretical link throughput: 12.6 GB/s x 2 (bidirectional)

1.2.2 Application Computing Server with Large Memory (GWACSL)

- Computing performance
  CPU: Intel Xeon E7-4880v2 (2.50 GHz) 2units (8 CPUs, 120 cores)
  Theoretical peak performance: 2.4 TFLOPS (2.5 GHz x 8 floating-point operations x 15 cores x 8 CPUs)
- Memory
  Memory capacity: 3 TB (1.5TB x 2 units)
  Memory bandwidth: 42.7 GB/s/CPU
  Memory bandwidth/FLOP: 0.14 Byte/FLOP
- Local disk
  Disk capacity: 3.6 TB ((300 GB x 2 + 1.2 TB) x 2 units)
- Interconnect
  FDR InfiniBand
  Theoretical link throughput: 6.8 GB/s x 2 paths x 2 (bidirectional)

1.3 Software Overview

The softwares available on the HOKUSAI BigWaterfall system are listed as follows:

Table 1-1 Software overview

Category	Massively Parallel Computer (BWMPC)	Application Computing Server with Large Memory(ACS)	Front End Servers
OS	Red Hat Enterprise Linux 7 (x 56nodes) CentOS7(x 784nodes) (Linux kernel version 3.10)	Red Hat Enterprise Linux 7 (Linux kernel version 3.10)	Red Hat Enterprise Linux 7 (Linux kernel version 3.10)
Compiler	Intel Parallel Studio XE Cluster Edition Intel C/C++ and Fortran compiler Intel TBB Intel Distribution for Python		
Library	IntelParallel Studio XE Cluster Edition Intel MKL Intel MPI Library Intel IPP Intel DAAL		
Tool	Intel Parallel Studio XE Cluster Edition Intel VTune Amplifier XE Intel Advisor Intel Inspector Intel Trace Analyzer & Collector		
Application	Gaussian(Only supported Red Hat Enterprise Linux 7(x 56nodes), ADF, AMBER,Q-Chem,GAMESS, GROMACS, NAMD, ROOT	Gaussian, ADF, AMBER, ANSYS, GAMESS, GROMACS, NAMD, ROOT	GaussView, ANSYS(preppost) VMD, ROOT

The latest information about the applications, libraries and so on, available for the HOKUSAI BigWaterfall system, will be published in the following User Portal:

https://hokusai.riken.jp/

1.4 Service Time Period

The services of HOKUSAI BigWaterfall system are regularly available for 24 hours except for the time periods of the periodical maintenance, the emergency maintenance, and the equipment maintenance such as the air conditioner maintenance and the power-supply facility maintenance. The availability of HOKUSAI BigWaterfall system is announced via the User Portal or the mailing list.

1.5 Usage Category

We set up the following types of user categories. Please make an application appropriate for your research thesis.

- General Use
- Quick Use

For more detailed information, visit the following URL to see the "Supercomputer System Usage Policy"

http://accc.riken.jp/en/supercom/application/usage-policy/

1.5.1 Allocated Computational Resources (core time)

Allocated computational core time depends on usage category. You can check the project accounting summary such as project ID/project name, allocated computational core time, and used computational core time, expiration date of allocated computational core time by the *listcpu* command.

No new jobs can be submitted and executed when remained computational core time runs out.

```
[username@hokusai1 ~]$ listcpu
[G20000] Study of parallel programs
            Limit       Used       Used(%)  Expiry date
-----------------------------------------------------------
bwmpc     100,000.0   10,000.0     10.0%  2021-03-31
gwacsl     10,000.0   10,000.0    100.0%  2021-03-31

userA             -    9,000.0         -  -
userB             -    1,000.0         -  -
   :
```

Table 1-2 listcpu information

Item	Description
Limit (h)	Allocated computational core time (hour)
Used (h)	Used computational core time (hour)
Use (%)	Used computational core time / Allocated computational core time (%)
Expiry date	Expiration date of the allocated computational core time

1.5.2 Special Use

For large-scale jobs or exclusive use, priority use of the computing resources (48 hours elapsed time), will be given to researchers of approved projects. The special use is available within the allocated computational time for each project.

The special use is announced by the mailing list. Research proposals need to have sufficient necessity for the special use of the system.

1.6 Job exection order

In this system, the job execution order is decided by the priority of all jobs. The priority is evaluated by the following items.

Table 1-3 listcpu information

Evaluation order	Evaluatino item	Overview
1	Fairshare value of project	Value to determine the priority of project. Calculate for each project based on the recovery rate and job execution history.
2	Fairshare value of user within a project	Value to determine the priority of users in same project.
3	Job priority	Priority of the user own job.
4	Job submittion time	Execute by the submission order.

Because the evaluation result with smaller "Evaluation order" take priority, the job which belongs to the project with larger "Fairshare value of project" gets preference over the jobs which are submitted earlier. About "fairshare value" is described in the next section.

1.6.1 Fairshare function

In this system, job execution order is decided by the fairshare value of each project and each user within a project. Fairshare value is changed continuously by starting job or recovering with time. Jobs are preferentially executed in the order of fairshare value of project.

The following figure indicates the behavior of fairshare value.

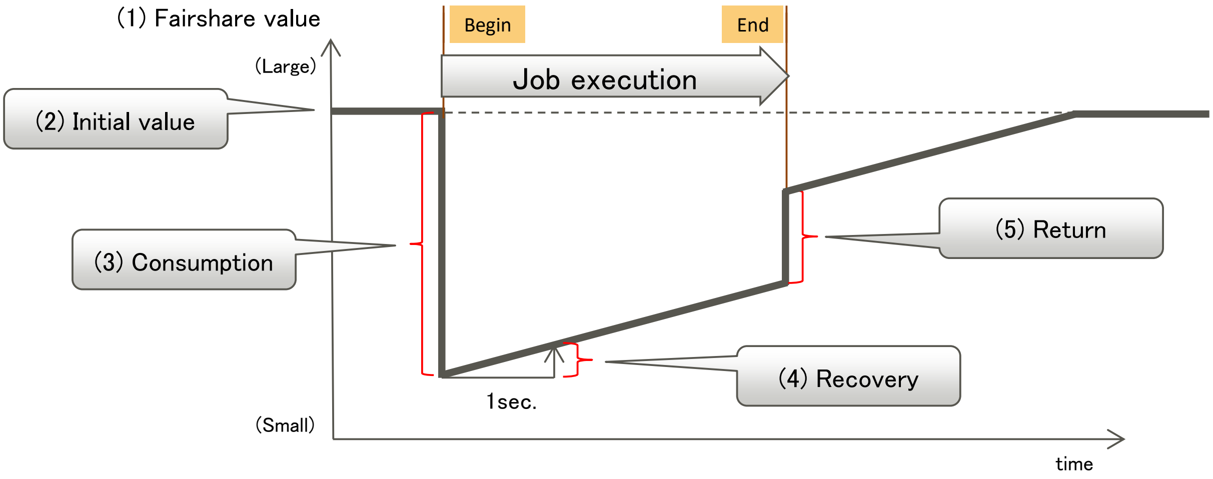


Figure 1-2 Behavior of fairshare value

Table 1-4 Term used in fairshare value

	Term	Meaning	Value in this system
(1)	Fairshare value	The priority of project	
(2)	Initial value	Initial and maximum value of fairshare value	1trillion
(3)	Consumption	Decrease from fairshare value at starting job	(Required number of cores) * (Elapse limit) [s]
(4)	Recovery	Add to fairshare value per second	Depending on the approved computation time of project
(5)	Return	Add value when the job is finished before reached to elapse limit	(Required number of cores) * (Elapse limit - Elapse time) [s]

⚠ The priority rank of project can be checked by *pjstat -p* command. Fairshare value is managed inside of system, so users cannot check and change them.

1.6.2 Backfill function

In this system, the backfill function is available to effectively use computing resource. Some idle computing resource will arise during the resource allocation process by previously described fairshare function. The job which can fill such idle resource will be run at an early time in spite of existence of other higher priority jobs. Backfilled jobs can be checked by pjstat command (its "START_DATE" is marked by "<").

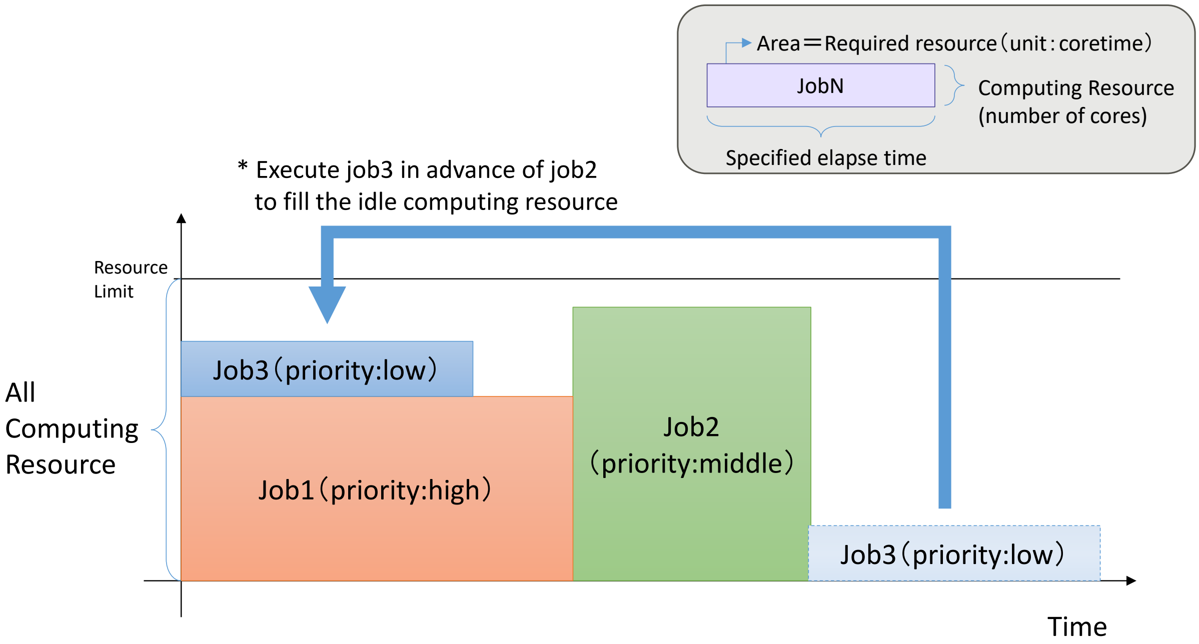

Area＝Required resource（unit：coretime）
JobN
Computing Resource (number of cores)
Specified elapse time
* Execute job3 in advance of job2 to fill the idle computing resource
Resource Limit
All Computing Resource
Job3（priority:low）
Job1（priority:high）
Job2（priority:middle）
Job3（priority:low）
Time

Figure 1-3 Behavior of backfill function

2. Login to HOKUSAI BigWaterfall System

2.1 Login Flow

The login flow for the HOKUSAI BigWaterfall system from account application to login as follows:

When the account is issued, the e-mail with the client certificate attachment is sent. After installing the client certificate on your PC, access the User Portal. You can login to the front end servers via SSH by registering your ssh public key on the User Portal.

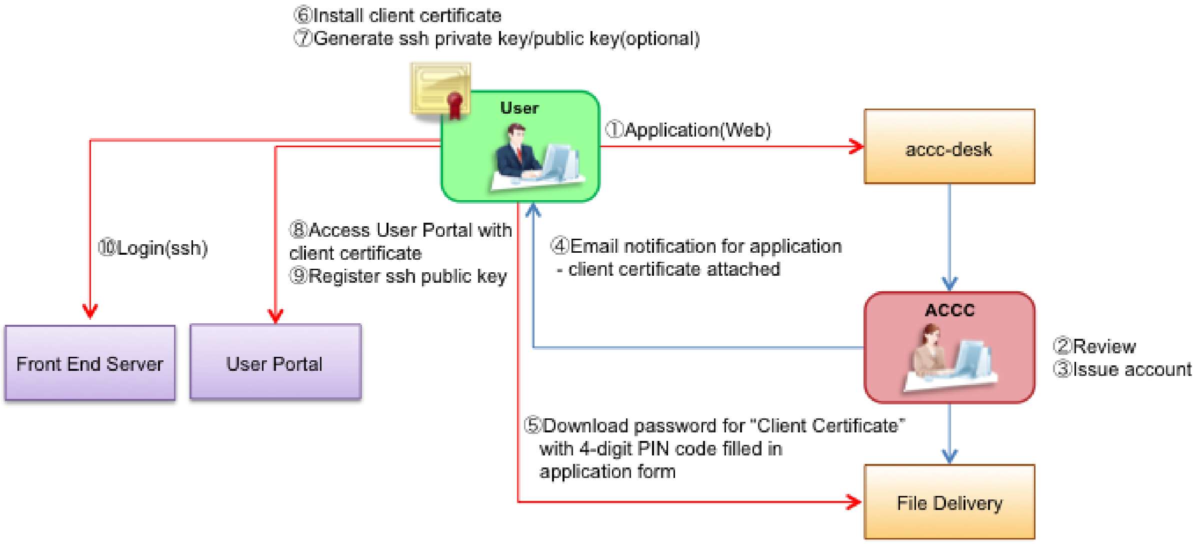

⑥Install client certificate
⑦Generate ssh private key/public key(optional)
User
①Application(Web)
accc-desk
⑩Login(ssh)
⑧Access User Portal with client certificate
⑨Register ssh public key
④Email notification for application
- client certificate attached
ACCC
②Review
③Issue account
Front End Server
User Portal
⑤Download password for "Client Certificate" with 4-digit PIN code filled in application form
File Delivery

Figure 2-1 Login flow

2.2 **Initial Settings**

When accessing the system for the first time, login to the User Portal and make sure to do the following initial settings:

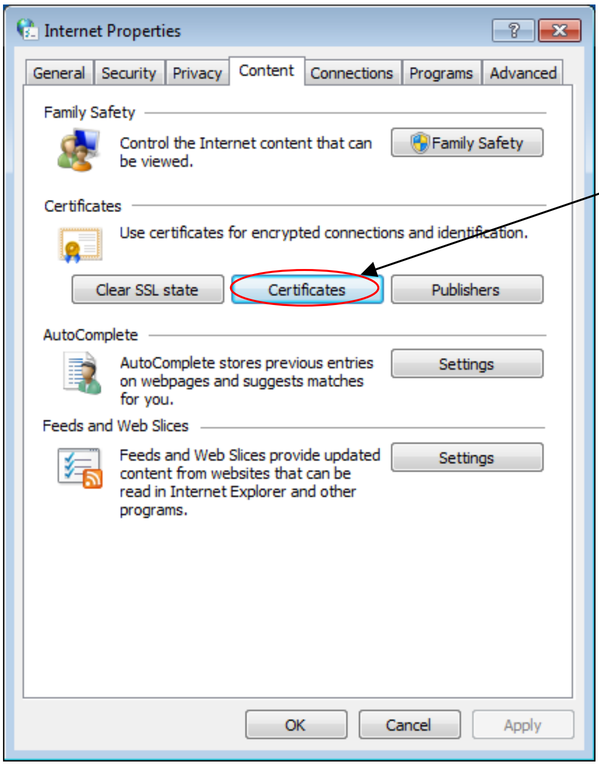
- ✓ 2.2.1 Install and Uninstall Client Certificate
- ✓ 2.2.2 Key Generation
- ✓ 2.2.3 Register ssh public key

2.2.1 Install and Uninstall Client Certificate

This section explains how to install and uninstall the client certificate on the Windows, MAC or Ubuntu. You might need to install it into the browser depending on your browser.

2.2.1.1 **Uninstall Client Certificate (Windows Environment)**

An old certificate is eliminated with a renewal of a client certificate.



Control the Internet content that can be viewed.

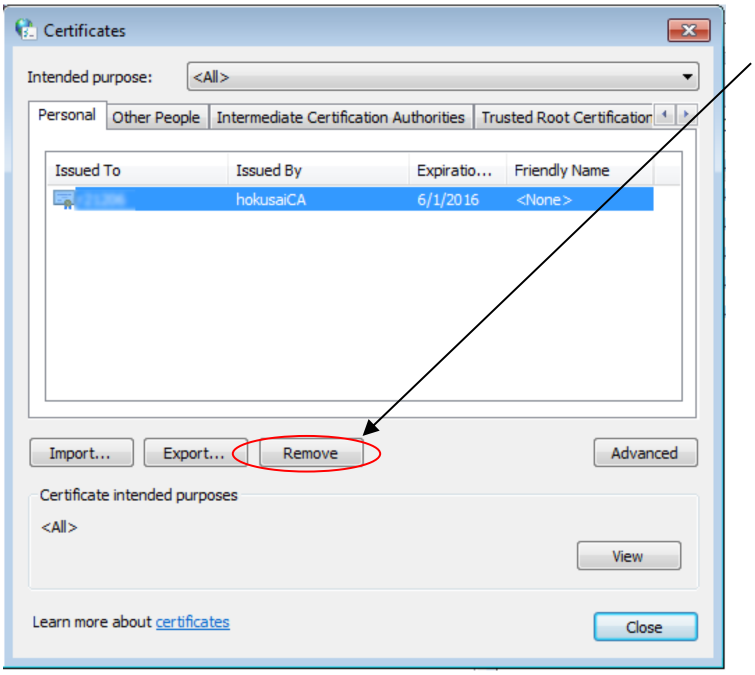
Use certificates for encrypted connections and identification.

AutoComplete stores previous entries on webpages and suggests matches for you.

Feeds and Web Slices provide updated content from websites that can be read in Internet Explorer and other programs.

1. Click [Start Menu]-> [Control Panel]-> [Internet Options].
2. Click “Certificate” button.

Figure 2-2 Screen of "Internet Properties"



Issued To	Issued By	Expiratio...	Friendly Name
[redacted]	hokusaiCA	6/1/2016	<None>

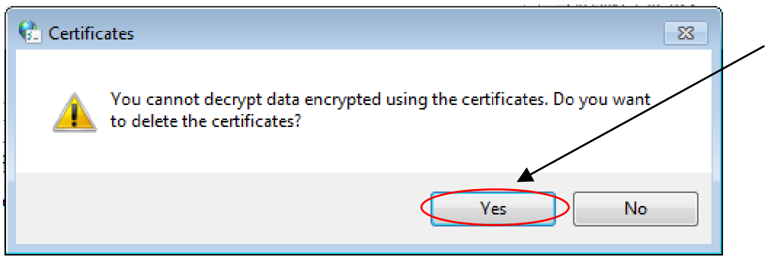
Certificate intended purposes

<All>

Learn more about certificates

1. Select the Certificate, Click “Remove” button.

Figure 2-3 First screen of "Certificates"



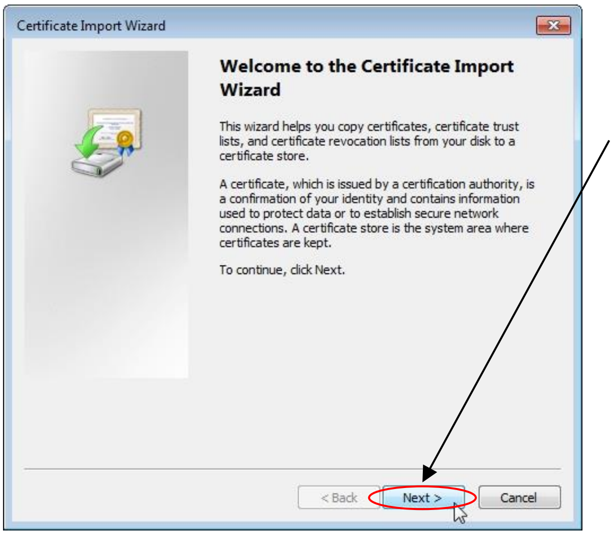
You cannot decrypt data encrypted using the certificates. Do you want to delete the certificates?

1. Click “Yes” button.

Figure 2-4 Second screen of "Internet Properties"

2.2.1.2 **Install Client Certificate (Windows Environment)**

Install the client certificate ACCC sent you by e-mail.



Certificate Import Wizard

Welcome to the Certificate Import Wizard

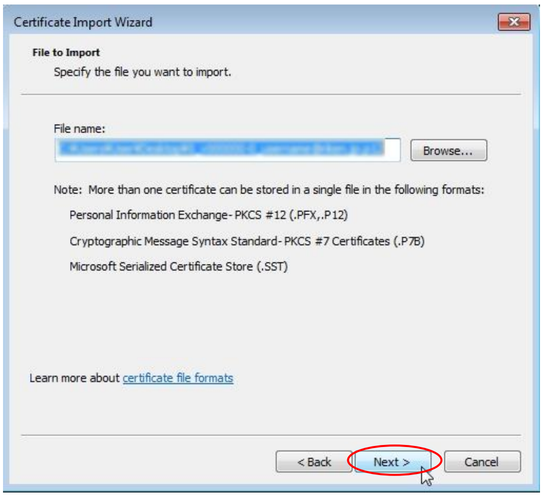
This wizard helps you copy certificates, certificate trust lists, and certificate revocation lists from your disk to a certificate store.

A certificate, which is issued by a certification authority, is a confirmation of your identity and contains information used to protect data or to establish secure network connections. A certificate store is the system area where certificates are kept.

To continue, click Next.

1. Double-click the client certificate provided by ACCC.
2. The Certificate Import Wizard starts. Click "Next" button.

Figure 2-5 First screen of "Certificate Import Wizard"



Certificate Import Wizard

File to Import

Specify the file you want to import.

File name:

[redacted]

Note: More than one certificate can be stored in a single file in the following formats:

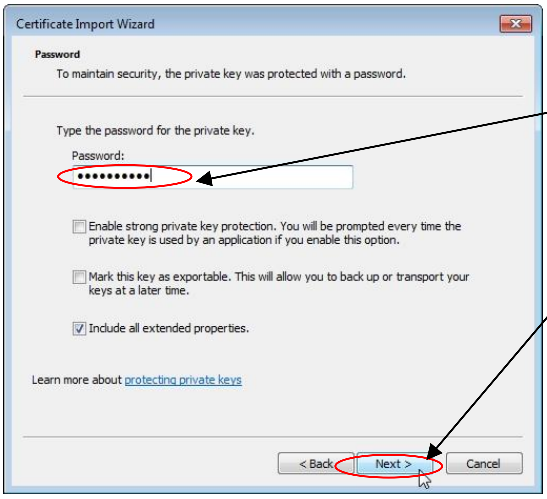
Personal Information Exchange- PKCS #12 (.PFX,.P12)

Cryptographic Message Syntax Standard- PKCS #7 Certificates (.P7B)

Microsoft Serialized Certificate Store (.SST)

Learn more about certificate file formats

Figure 2-6 Second screen of "Certificate Import Wizard"



Certificate Import Wizard

Password

To maintain security, the private key was protected with a password.

Type the password for the private key.

Password: ••••••••••

☐ Enable strong private key protection. You will be prompted every time the private key is used by an application if you enable this option.

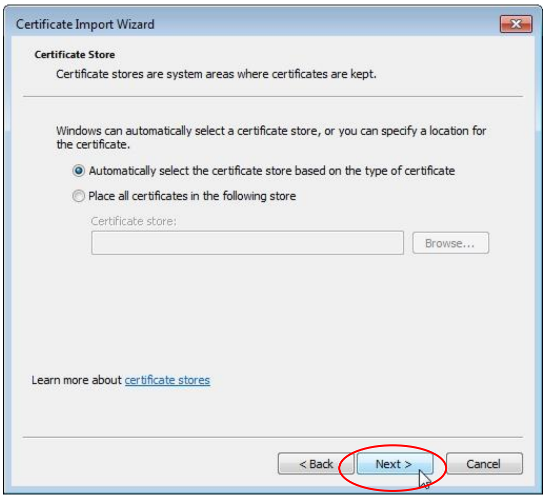
☐ Mark this key as exportable. This will allow you to back up or transport your keys at a later time.

☑ Include all extended properties.

Learn more about protecting private keys

1. Enter the password for "Client Certificate" issued by ACCC.
2. Click "Next" button.

Figure 2-7 Third screen of "Certificate Import Wizard"



Certificate Import Wizard

Certificate Store

Certificate stores are system areas where certificates are kept.

Windows can automatically select a certificate store, or you can specify a location for the certificate.

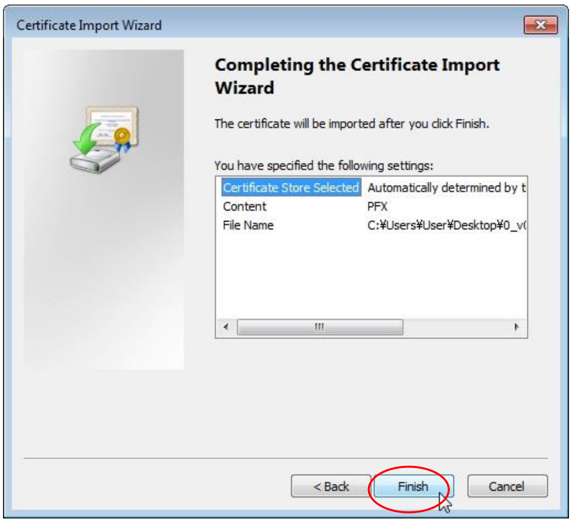
● Automatically select the certificate store based on the type of certificate

○ Place all certificates in the following store

Certificate store:

Learn more about certificate stores

Figure 2-8 Fourth screen of "Certificate Import Wizard"



Certificate Import Wizard

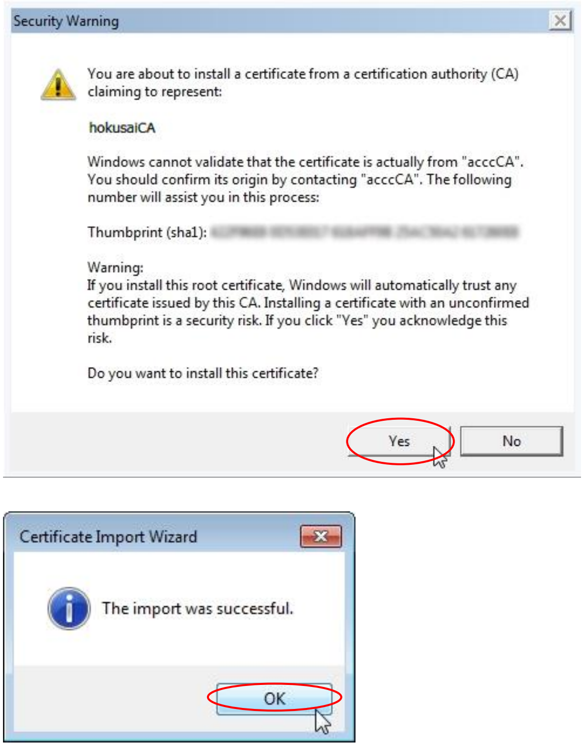
Completing the Certificate Import Wizard

The certificate will be imported after you click Finish.

You have specified the following settings:

Certificate Store Selected	Automatically determined by t
Content	PFX
File Name	C:¥Users¥User¥Desktop¥0_v(

Figure 2-9 Fifth screen of "Certificate Import Wizard"



Security Warning

You are about to install a certificate from a certification authority (CA) claiming to represent:

hokusaiCA

Windows cannot validate that the certificate is actually from "acccCA". You should confirm its origin by contacting "acccCA". The following number will assist you in this process:

Thumbprint (sha1): [redacted]

Warning:
If you install this root certificate, Windows will automatically trust any certificate issued by this CA. Installing a certificate with an unconfirmed thumbprint is a security risk. If you click "Yes" you acknowledge this risk.

Do you want to install this certificate?

Certificate Import Wizard

The import was successful.

Figure 2-10 Sixth screen of "Certificate Import Wizard"

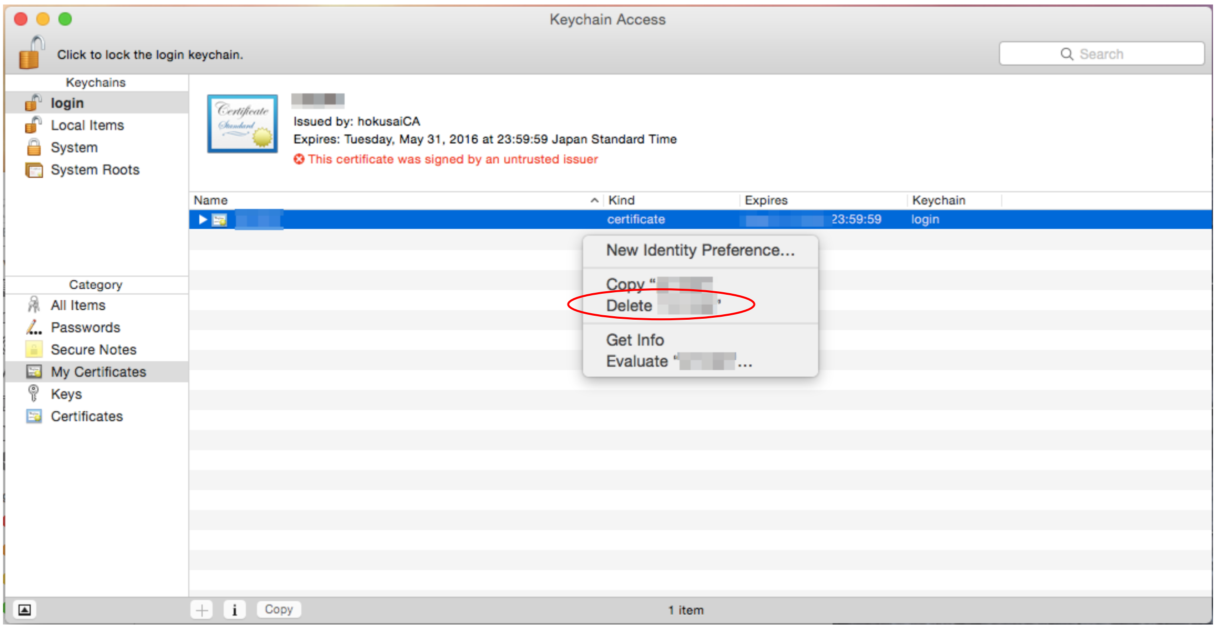
⚠ When you use the Firefox as the standard browser, refer to "2.2.1.6 Install Client Certificate (Ubuntu Environment)”

2.2.1.3 Uninstall Client Certificate (Mac Environment)

An old certificate is eliminated with a renewal of a client certificate.

Open "Keychain Access". (Finder > Application > Utility > Keychain Access)

Click "My Certificates" category to see available client certificates. Control-Click your client certificate for the HOKUSAI BigWaterfall system and select "New Identity Preference..." from the contextual menu.

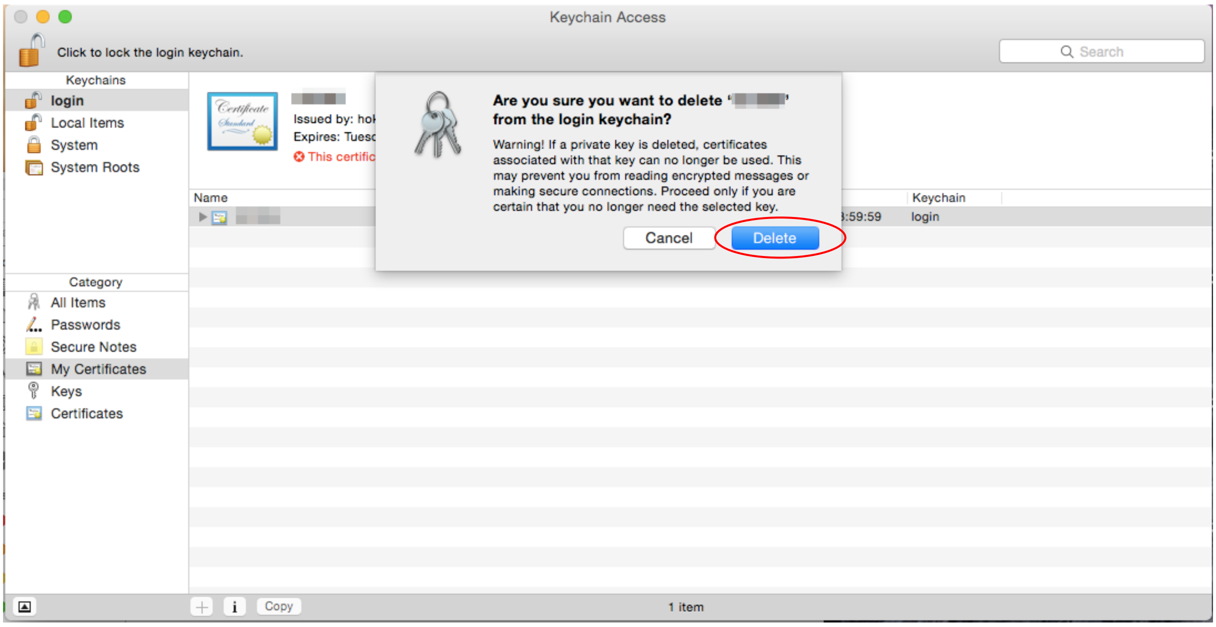


[redacted]
Issued by: hokusaiCA
Expires: Tuesday, May 31, 2016 at 23:59:59 Japan Standard Time
This certificate was signed by an untrusted issuer

Name	Kind	Expires	Keychain
[redacted]	certificate	[redacted] 23:59:59	login

Figure 2-11 Keychain Access

Click “Delete” button.



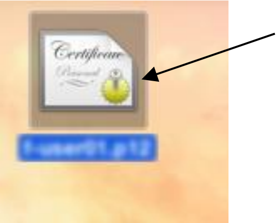
Are you sure you want to delete '[redacted]' from the login keychain?

Warning! If a private key is deleted, certificates associated with that key can no longer be used. This may prevent you from reading encrypted messages or making secure connections. Proceed only if you are certain that you no longer need the selected key.

Figure 2-12 Keychain Access Delete

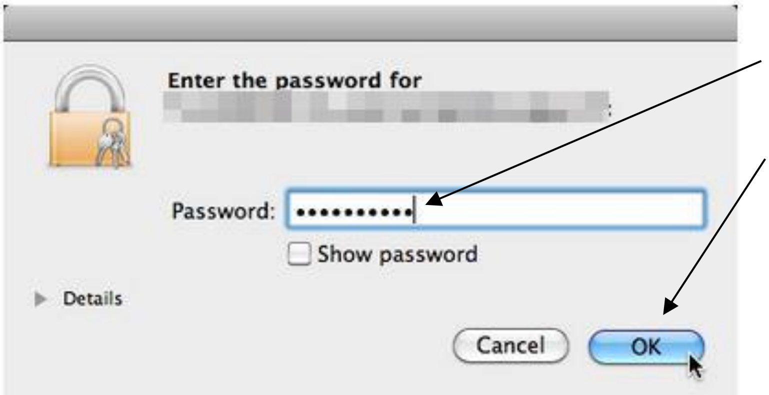
2.2.1.4 Install Client Certificate (Mac Environment)

Install the client certificate ACCC sent you by e-mail.



1. Double click the client certificate provided by ACCC.

Figure 2-13 Client certificate icon



Enter the password for [redacted]:

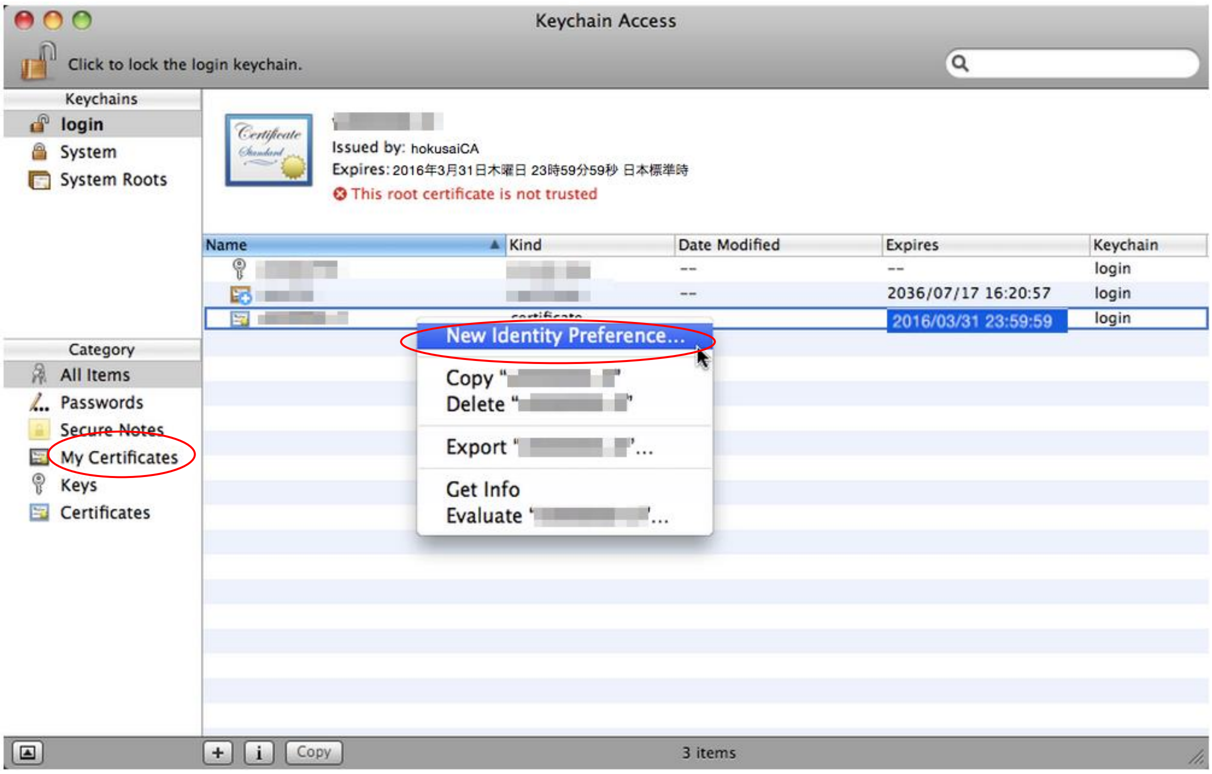
Password: ••••••••••

1. Enter the password for "Client Certificate" issued by ACCC.
2. Click "OK" button.

Figure 2-14 Install the client certificate

Open "Keychain Access". (Finder > Application > Utility > Keychain Access)

Click "My Certificates" category to see available client certificates. Control-Click your client certificate for the HOKUSAI BigWaterfall system and select "New Identity Preference..." from the contextual menu.

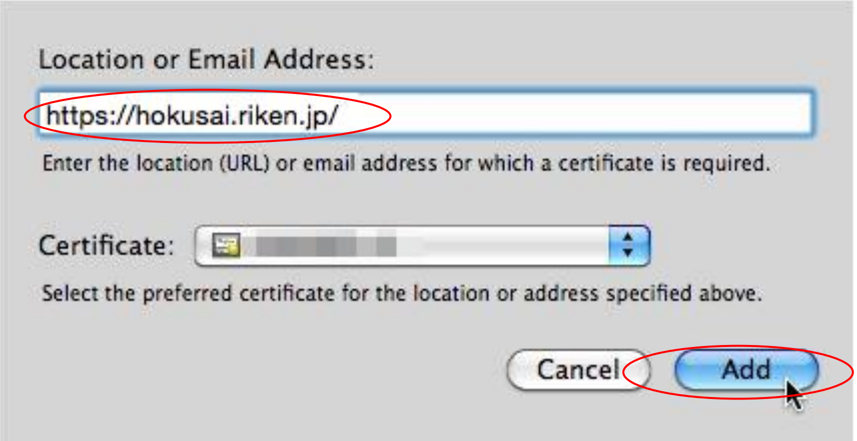


[redacted]
Issued by: hokusaiCA
Expires: 2016年3月31日木曜日 23時59分59秒 日本標準時
This root certificate is not trusted

Name	Kind	Date Modified	Expires	Keychain
[redacted]	[redacted]	--	--	login
[redacted]	[redacted]	--	2036/07/17 16:20:57	login
[redacted]	certificate		2016/03/31 23:59:59	login

Figure 2-15 Keychain Access

Enter https://hokusai.riken.jp/ in [Location or Email Address:] and click [Add] button.



Location or Email Address:

https://hokusai.riken.jp/

Enter the location (URL) or email address for which a certificate is required.

Certificate:

Select the preferred certificate for the location or address specified above.

Figure 2-16 New Identity Preference

When you use the Firefox as the standard browser, refer to "2.2.1.6 Install Client Certificate (Ubuntu Environment)”

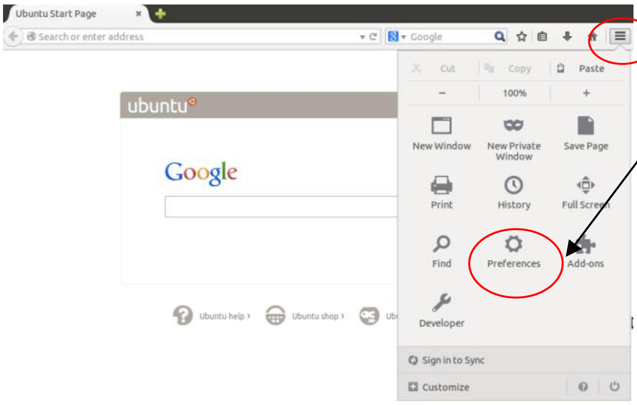
2.2.1.5 **Uninstall Client Certificate(Ubuntu Environment)**

An old certificate is eliminated with a renewal of a client certificate.

Import the client certificate ACCC sent you by e-mail.

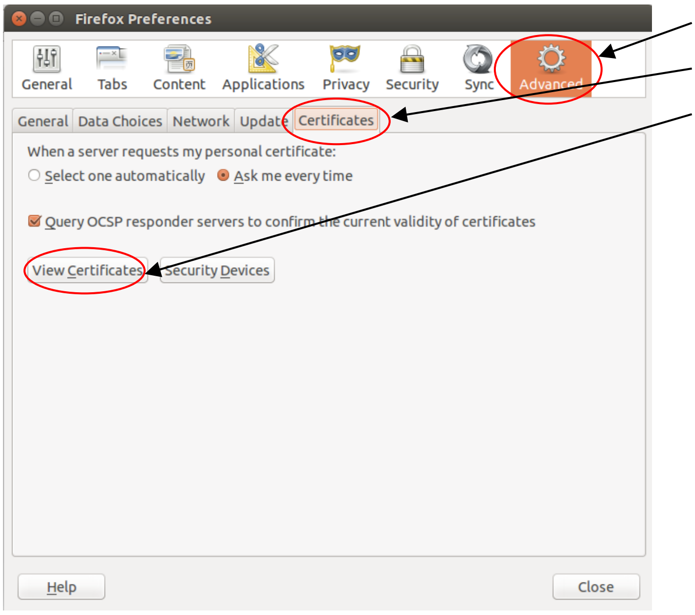
1. Launch Firefox.

Figure 2-17 Launch Firefox



1. Click [Open menu].
2. Click [Preferences].

Figure 2-18 Firefox menu

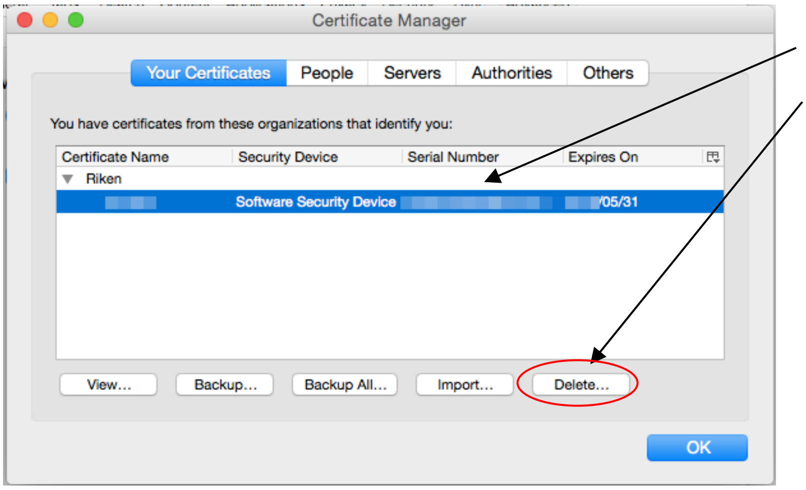


When a server requests my personal certificate:
○ Select one automatically ● Ask me every time

☑ Query OCSP responder servers to confirm the current validity of certificates

3. Click [Advanced] panel.
4. Click [Certificates] tab.
5. Click [View Certificates] button.

Figure 2-19 Firefox Preferences

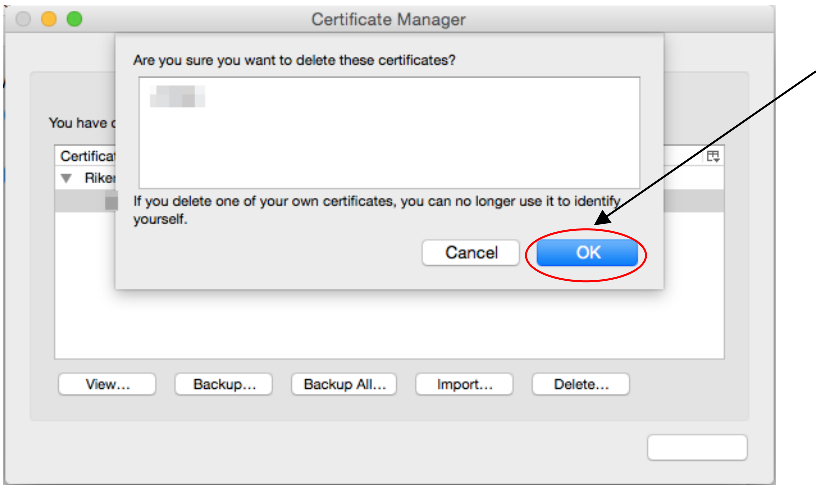


You have certificates from these organizations that identify you:

Certificate Name	Security Device	Serial Number	Expires On
▼ Riken			
[redacted]	Software Security Device	[redacted]	[redacted]/05/31

6. Click Client Certificate.
7. Click [Delete] button.

Figure 2-20 Firefox Preferences



Certificate Manager

Are you sure you want to delete these certificates?

If you delete one of your own certificates, you can no longer use it to identify yourself.

8. Click [OK] button.

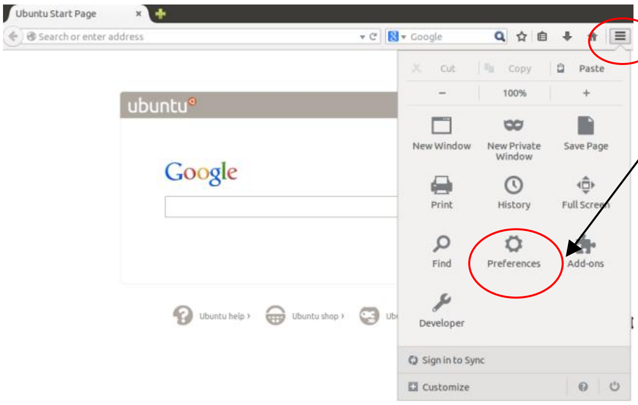
Figure 2-21 Firefox Certificate Manager Delete

2.2.1.6 Install Client Certificate (Ubuntu Environment)

Import the client certificate ACCC sent you by e-mail.

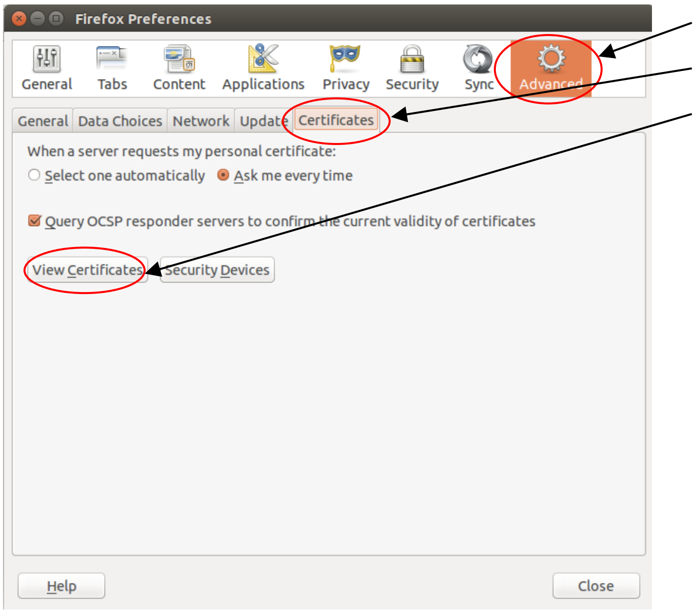
2. Launch Firefox.

Figure 2-22 Launch Firefox



3. Click [Open menu].
4. Click [Preferences].

Figure 2-23 Firefox menu



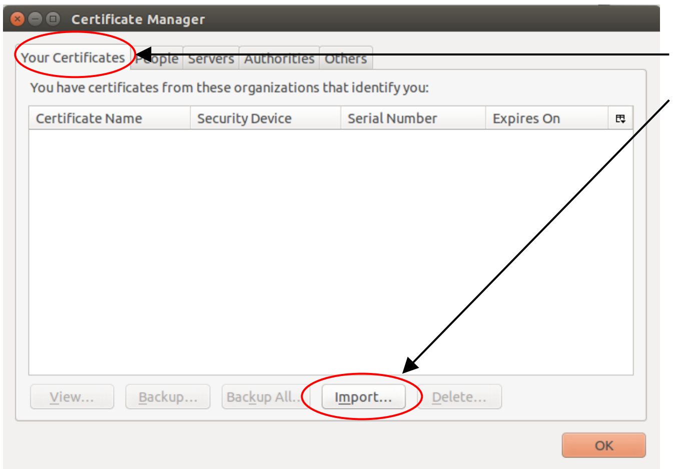
When a server requests my personal certificate:

○ Select one automatically ● Ask me every time

☑ Query OCSP responder servers to confirm the current validity of certificates

9. Click [Advanced] panel.
10. Click [Certificates] tab.
11. Click [View Certificates] button.

Figure 2-24 Firefox Preferences

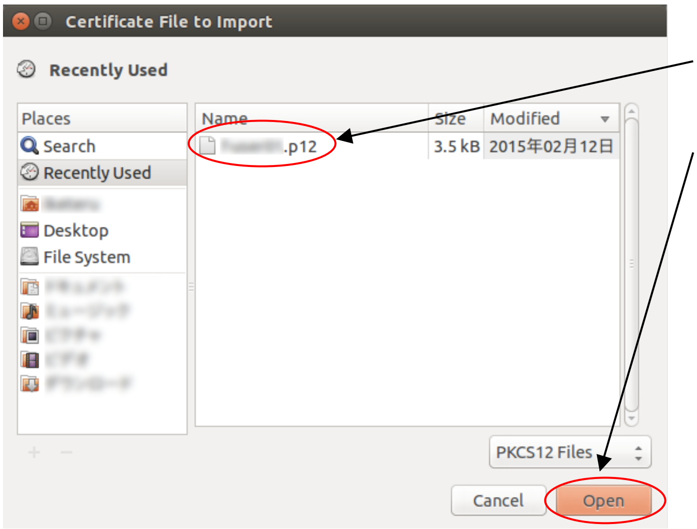


You have certificates from these organizations that identify you:

Certificate Name	Security Device	Serial Number	Expires On

6. Click [Your Certificates]
7. Click [Import] button.

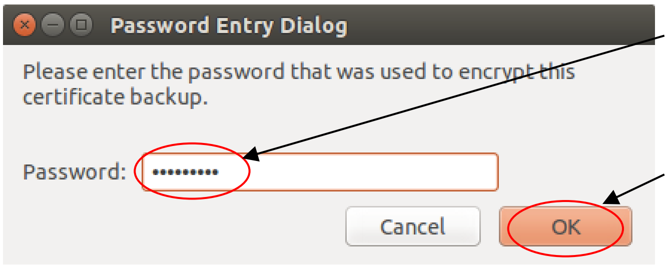
Figure 2-25 Firefox Certificate Manager



Name	Size	Modified
[redacted].p12	3.5 kB	2015年02月12日

8. Select "Client Certificate" sent by ACCC.
9. Click [Open] button.

Figure 2-26 Firefox Certificate Manager Import

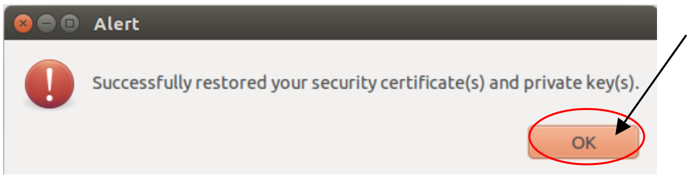


Please enter the password that was used to encrypt this certificate backup.

Password: •••••••••

10. Enter the password for "Client Certificate" issued by ACCC.
11. Click [OK] button.

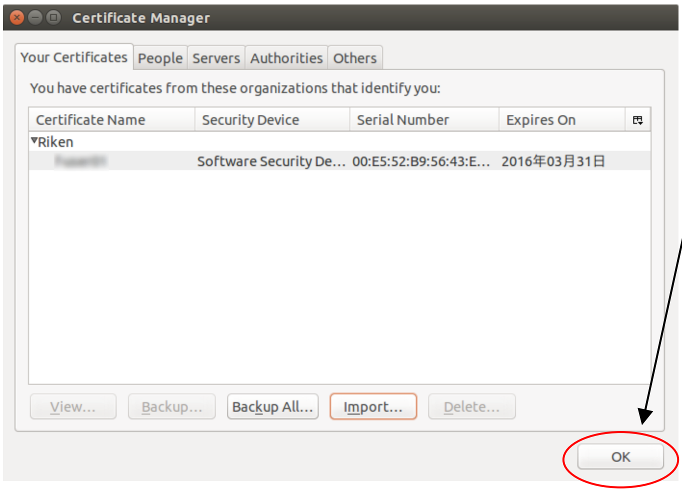
Figure 2-27 Firefox Password Entry Dialog



Successfully restored your security certificate(s) and private key(s).

12. Click [OK] button.

Figure 2-28 Firefox Alert



You have certificates from these organizations that identify you:

Certificate Name	Security Device	Serial Number	Expires On
▾Riken			
[redacted]	Software Security De...	00:E5:52:B9:56:43:E...	2016年03月31日

13. Click [OK] button.

Figure 2-29 Firefox Certificate Manager

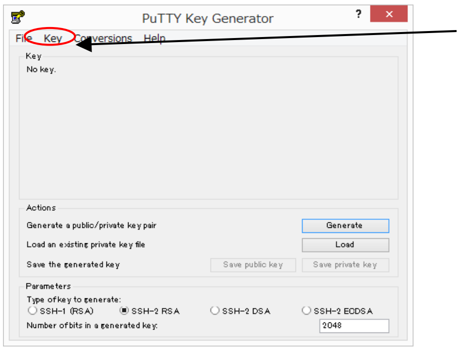
2.2.2 Key Generation

2.2.2.1 Key Generation (Windows Environment)

Generate the private key/public key pair with SSH-2 RSA method on your PC. To generate the keys, use PuTTYgen utility provided with PUTTY package. If the keys are already generated, skip this step.

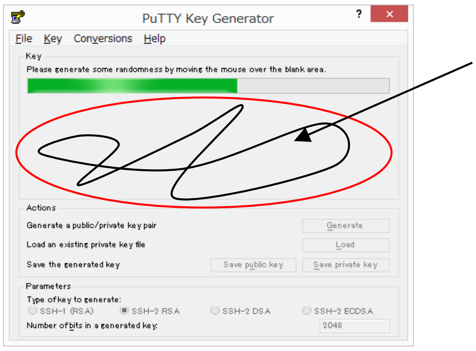
PuTTY can be downloaded from the following site:

PuTTY: http://www.chiark.greenend.org.uk/~sgtatham/putty/



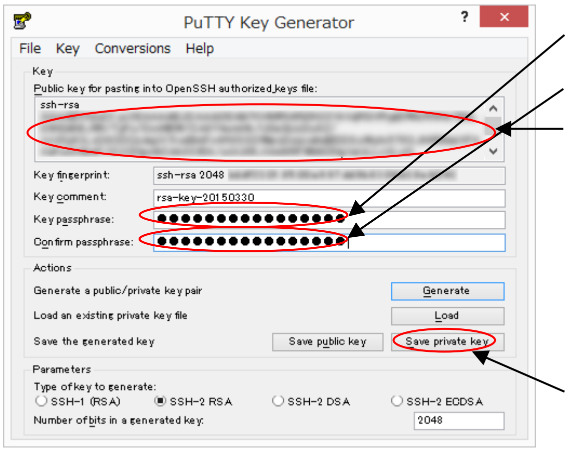
1. Click [Key] menu.
2. Click [Generate key pair].

Figure 2-30 First screen for private/public key generation with PuTTYgen


PuTTY Key Generator
File Key Conversions Help
Key
Please generate some randomness by moving the mouse over the blank area.
Actions
Generate a public/private key pair
Generate
Load an existing private key file
Load
Save the generated key
Save public key
Save private key
Parameters
Type of key to generate:
○ SSH-1 (RSA)
● SSH-2 RSA
○ SSH-2 DSA
○ SSH-2 ECDSA
Number of bits in a generated key:
2048

3. Move around the mouse to generate random numbers for key generation.

Figure 2-31 Second screen for private/public key generation with PuTTYgen



Public key for pasting into OpenSSH authorized_keys file:

ssh-rsa [redacted]

Key fingerprint: ssh-rsa 2048 [redacted]

Key comment: rsa-key-20150330

Key passphrase: ●●●●●●●●●●●●●●●●

Confirm passphrase: ●●●●●●●●●●●●●●●●

Number of bits in a generated key: 2048

4. Enter the passphrase
5. Re-enter the passphrase.
6. Copy the contents of public key and register them according to 2.2.3 Register ssh public key,
   or
   save to the file(name:id_rsa.pub) using the text editor.
7. Click [Save private key] button to save.

Figure 2-32 Third screen for private/public key generation with PuTTYgen

2.2.2.2 **Key Generation (UNIX/Mac Environment)**

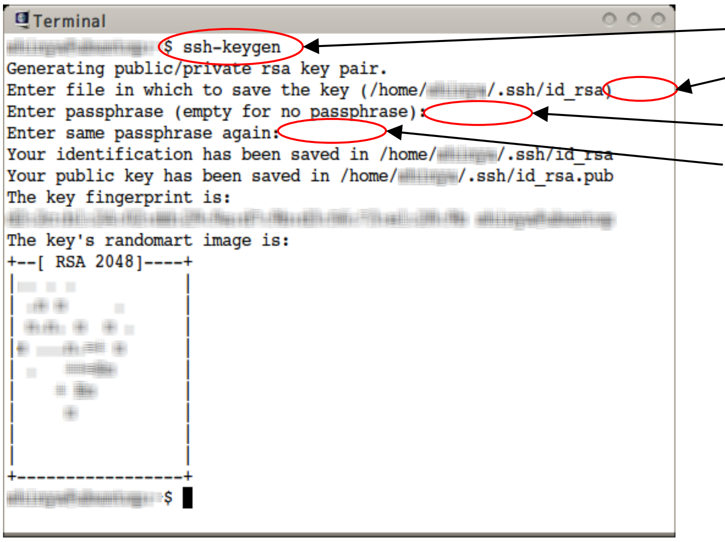
Generate the private key/public key pair with SSH-2 RSA method on your PC. To generate the keys, use the *ssh-keygen* command. If the keys are already generated, skip this step.

- UNIX/Linux

  Launch the terminal emulator, and run the *ssh-keygen* command.

- Mac

  Launch the Terminal, and run the *ssh-keygen* command.



```
[redacted]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/[redacted]/.ssh/id_rsa)
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/[redacted]/.ssh/id_rsa
Your public key has been saved in /home/[redacted]/.ssh/id_rsa.pub
The key fingerprint is:
[redacted]
The key's randomart image is:
+--[ RSA 2048]----+
|[redacted]       |
|                 |
|                 |
|                 |
|                 |
|                 |
|                 |
|                 |
|                 |
+-----------------+
[redacted]$ █
```

1. Run the *ssh-keygen* command.
2. Enter the Return key.
3. Enter the passphrase.
4. Re-enter the passphrase.

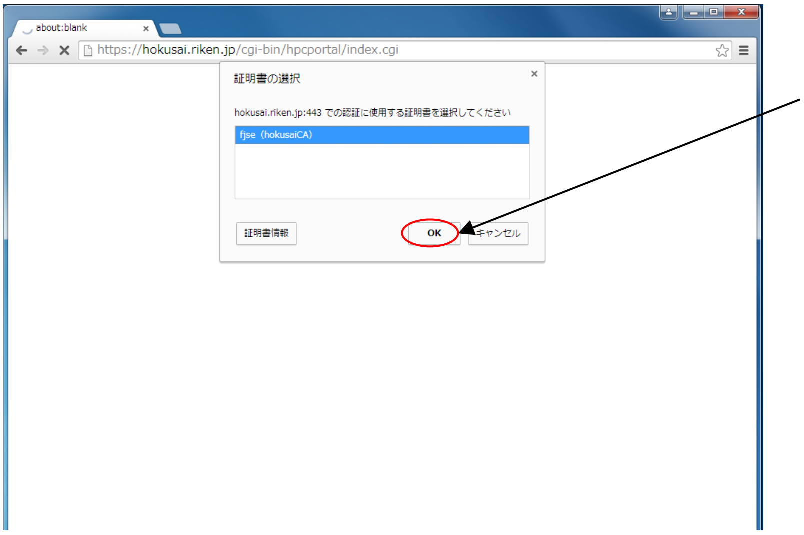
Figure 2-33 Generate key pair with the ssh-keygen command

2.2.3 Register ssh public key

(1) Access the following URL: (HTTP is not supported.)

https://hokusai.riken.jp/

(2) Login to the User Portal with the client certificate.



証明書の選択

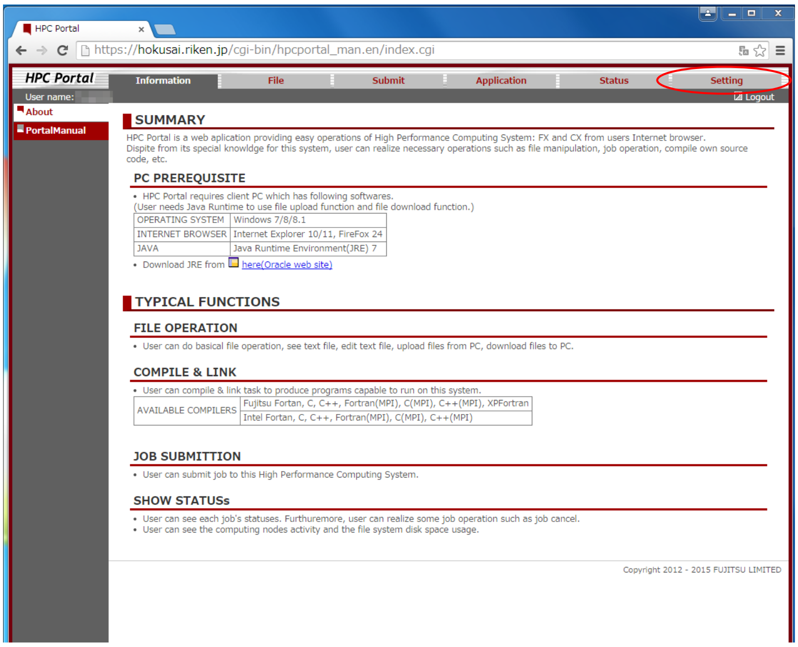
hokusai.riken.jp:443 での認証に使用する証明書を選択してください

fjse（hokusaiCA）

Select the client certificate for HOKUSAI BigWaterfall system, and click [OK] button.

Figure 2-34 Screen of certificate selection

(3) Click "Setting" menu.



User name: [redacted]

SUMMARY

HPC Portal is a web aplication providing easy operations of High Performance Computing System: FX and CX from users Internet browser.
Dispite from its special knowldge for this system, user can realize necessary operations such as file manipulation, job operation, compile own source code, etc.

PC PREREQUISITE

- HPC Portal requires client PC which has following softwares.

(User needs Java Runtime to use file upload function and file download function.)

OPERATING SYSTEM	Windows 7/8/8.1
INTERNET BROWSER	Internet Explorer 10/11, FireFox 24
JAVA	Java Runtime Environment(JRE) 7

- Download JRE from here(Oracle web site)

TYPICAL FUNCTIONS

FILE OPERATION

- User can do basical file operation, see text file, edit text file, upload files from PC, download files to PC.

COMPILE & LINK

- User can compile & link task to produce programs capable to run on this system.

AVAILABLE COMPILERS	Fujitsu Fortan, C, C++, Fortran(MPI), C(MPI), C++(MPI), XPFortran
	Intel Fortan, C, C++, Fortran(MPI), C(MPI), C++(MPI)

JOB SUBMITTION

- User can submit job to this High Performance Computing System.

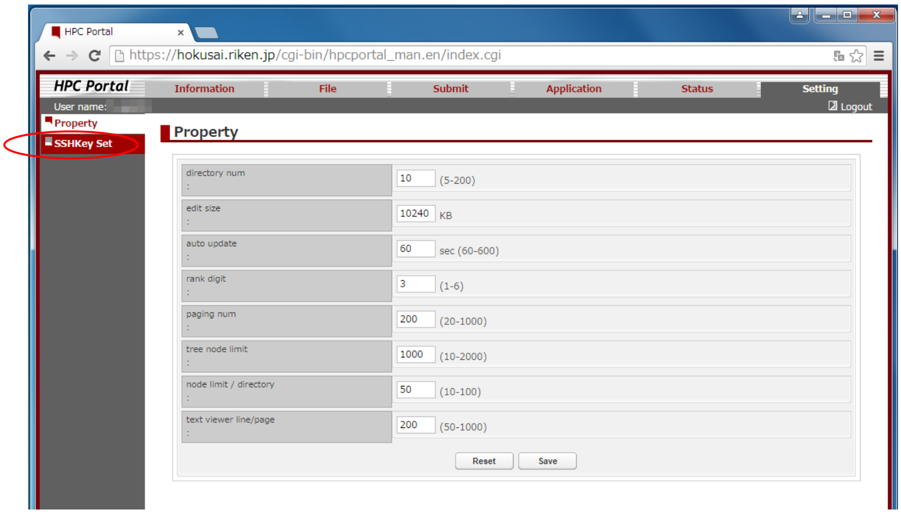
SHOW STATUSs

- User can see each job's statuses. Furthuremore, user can realize some job operation such as job cancel.
- User can see the computing nodes activity and the file system disk space usage.

Copyright 2012 - 2015 FUJITSU LIMITED

Figure 2-35 Screen of clicking "Setting" menu

(4) Click "SSHKey Set".



directory num : 10 (5-200)
edit size : 10240 KB
auto update : 60 sec (60-600)
rank digit : 3 (1-6)
paging num : 200 (20-1000)
tree node limit : 1000 (10-2000)
node limit / directory : 50 (10-100)
text viewer line/page : 200 (50-1000)

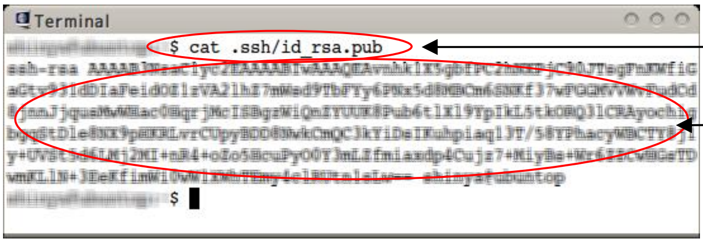
Figure 2-36 Screen of clicking "SSHKey Set"

(5) Register SSH public key

Refer to "2.2.2.1 Key Generation (Windows Environment)" for Windows and "2.2.2.2 Key Generation (UNIX/Mac Environment)" for UNIX/Mac about the key generation.

- Windows: Display the contents of the public key file (id_rsa.pub) with any text editor
- Mac: Launch Terminal, and display the contents of the public key with the *cat* command.
- UNIX/Linux: Launch virtual terminal, and display the contents of the public key with the *cat* command.

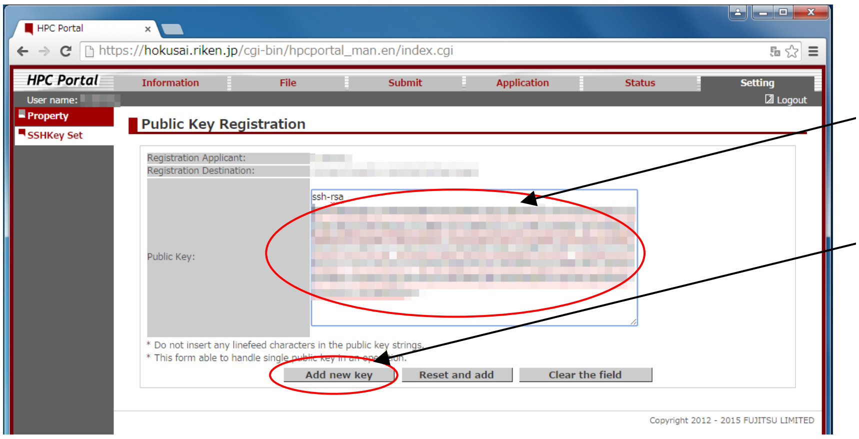
The default path of public key file is ~/.ssh/id_rsa.pub.



```
[redacted]$ cat .ssh/id_rsa.pub
[redacted]
[redacted]$ █
```

1. Display the public key.
2. Copy the contents.

Figure 2-37 Copy the contents of the public key



User name: [redacted]

Public Key Registration

Registration Applicant: [redacted]
Registration Destination: [redacted]

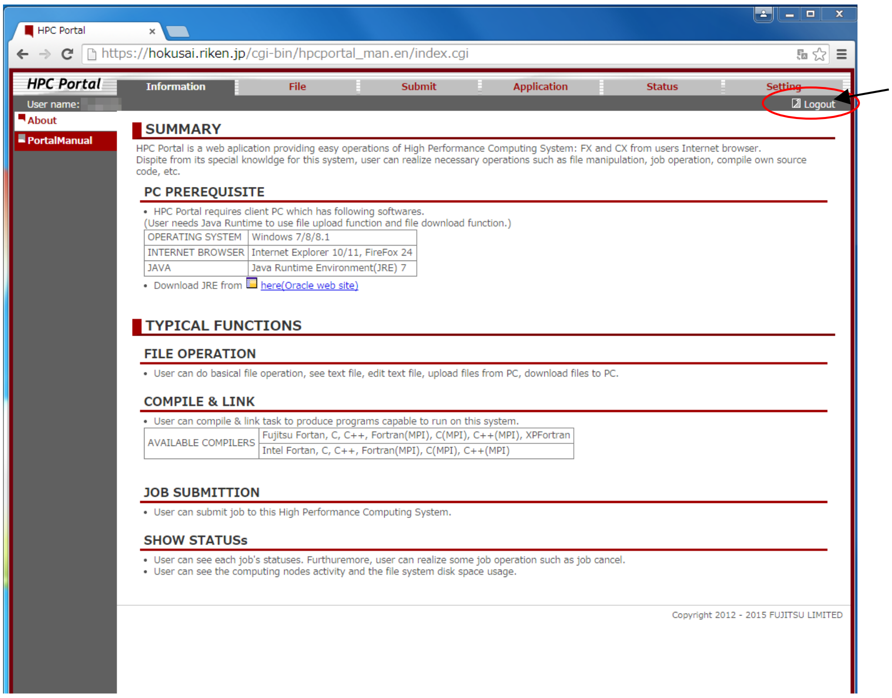
Public Key: ssh-rsa [redacted]

* Do not insert any linefeed characters in the public key strings.
* This form able to handle single public key in an operation.

Copyright 2012 - 2015 FUJITSU LIMITED

1. Paste the contents of public key.
2. Click [Add new key] button.

Figure 2-38 Register the public key



User name: [redacted]

SUMMARY

HPC Portal is a web aplication providing easy operations of High Performance Computing System: FX and CX from users Internet browser.
Dispite from its special knowldge for this system, user can realize necessary operations such as file manipulation, job operation, compile own source code, etc.

PC PREREQUISITE

- HPC Portal requires client PC which has following softwares.
(User needs Java Runtime to use file upload function and file download function.)

OPERATING SYSTEM	Windows 7/8/8.1
INTERNET BROWSER	Internet Explorer 10/11, FireFox 24
JAVA	Java Runtime Environment(JRE) 7

- Download JRE from here(Oracle web site)

TYPICAL FUNCTIONS

FILE OPERATION

- User can do basical file operation, see text file, edit text file, upload files from PC, download files to PC.

COMPILE & LINK

- User can compile & link task to produce programs capable to run on this system.

AVAILABLE COMPILERS	Fujitsu Fortan, C, C++, Fortran(MPI), C(MPI), C++(MPI), XPFortran
	Intel Fortan, C, C++, Fortran(MPI), C(MPI), C++(MPI)

JOB SUBMITTION

- User can submit job to this High Performance Computing System.

SHOW STATUSs

- User can see each job's statuses. Furthuremore, user can realize some job operation such as job cancel.
- User can see the computing nodes activity and the file system disk space usage.

Copyright 2012 - 2015 FUJITSU LIMITED

1. Click [Logout]

Figure 2-39 Logout from User Portal

2.3 Network Access

The servers within the HOKUSAI BigWaterfall system in which you can access via SSH/HTTPS are the front end servers. The front end server consists of 4 servers.

For SSH access, only public key authentication with SSH protocol version 2 is enabled.

The User Portal enables you to register the SSH public key, operate files, manage jobs and view manuals via HTTPS.

Destination hosts are as follows:

Table 2-1 Destination hosts

Host name (FQDN)	Service	Purpose to access
hokusai.riken.jp	SSH	• Virtual terminal • File transfer
	HTTPS	• Register the SSH public key • Operate files • Manage jobs • View manuals • Use Development Tools (FX100)

2.4 **SSH Login**

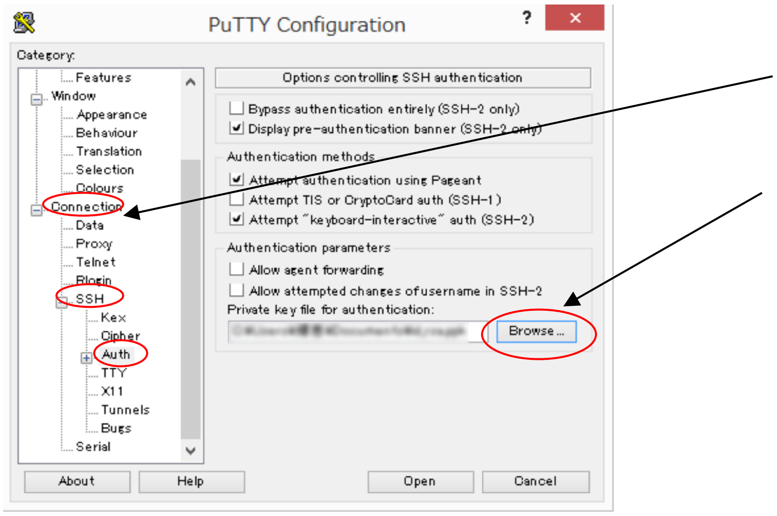
2.4.1 SSH Login (Windows Environment)

This section describes how to use PUTTY for virtual terminal while various terminal software tools are available on Windows. For users who use Cygwin, refer to 2.4.2 SSH Login (UNIX/Mac Environment).

PuTTY can be downloaded from the following site:

PuTTY: http://www.chiark.greenend.org.uk/~sgtatham/putty/

(1) Launch PuTTY



Options controlling SSH authentication

□ Bypass authentication entirely (SSH-2 only)
☑ Display pre-authentication banner (SSH-2 only)

Authentication methods

☑ Attempt authentication using Pageant
□ Attempt TIS or CryptoCard auth (SSH-1)
☑ Attempt "keyboard-interactive" auth (SSH-2)

Authentication parameters

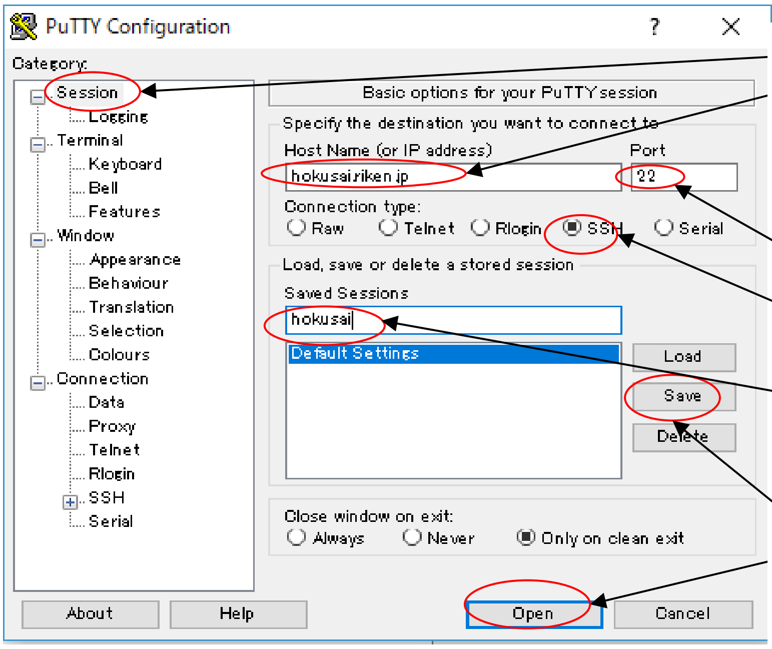
□ Allow agent forwarding
□ Allow attempted changes of username in SSH-2

Private key file for authentication: [redacted]

1. Click [Connection] - [SSH] - [Auth].
2. Click [Browse] button, and select the private key file.

Figure 2-40 Screen of selecting private key with PuTTY

(2) Open session to the HOKUSAI BigWaterfall system with PuTTY



Specify the destination you want to connect to

Host Name (or IP address): hokusai.riken.jp Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Load, save or delete a stored session

Saved Sessions: hokusai

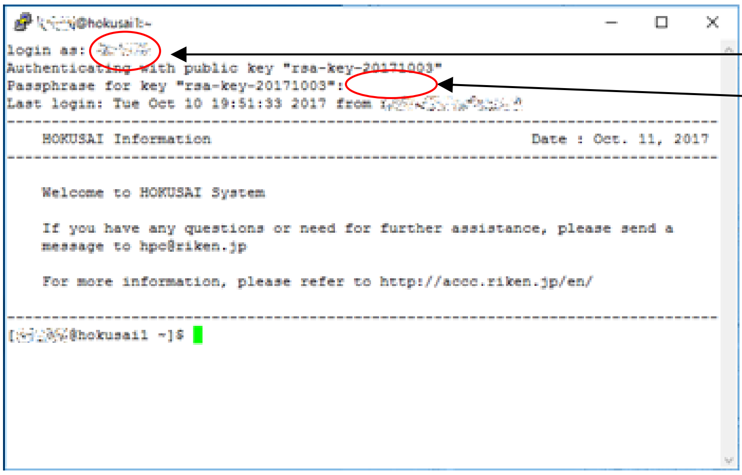
Default Settings

Close window on exit: ○ Always ○ Never ● Only on clean exit

1. Click [session]
2. Enter the following information
   [Host Name]
   hokusai.riken.jp
   [Port] 22
   [Connection type]
   SSH
3. [Saved Sessions]
   name(ex: greatwave)
4. Click [Save] button
5. Click [Open] button

Figure 2-41 PuTTY session screen

(3) Enter username and passphrase entered in "2.2.2 Key Generation".



```
login as: [redacted]
Authenticating with public key "rsa-key-20171003"
Passphrase for key "rsa-key-20171003":
Last login: Tue Oct 10 19:51:33 2017 from [redacted]
--------------------------------------------------------------------------------
    HOKUSAI Information                                       Date : Oct. 11, 2017
--------------------------------------------------------------------------------

    Welcome to HOKUSAI System

    If you have any questions or need for further assistance, please send a
    message to hpc@riken.jp

    For more information, please refer to http://accc.riken.jp/en/

--------------------------------------------------------------------------------
[[redacted]@hokusai1 ~]$
```

1. Enter username.
2. Enter passphrase.

Figure 2-42 PuTTY login screen

2.4.2 SSH Login (UNIX/Mac Environment)

To login to the HOKUSAI BigWaterfall system from your PC via SSH, use the *ssh* command.

```
$ ssh -l username hokusai.riken.jp
The authenticity of host '
Enter passphrase for key'/home/username/.ssh/id_rsa': ++++++++  ← Enter
passphrase
[username@hokusai1 ~]$
```

2.5 SSH Agent Forwarding

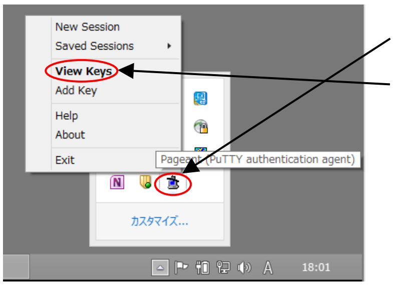
When you access external systems from the HOKUSAI BigWaterfall system, login to the front end servers enabling SSH Agent forwarding.

⚠ **To prevent the front end servers from being used as a step to access external system, it is prohibited to store the private key on the HOKUSAI BigWaterfall system.**

⚠ **The protocols permitted to access from HOKUSAI BigWaterfall system to external system are only SSH, HTTP and HTTPS.**

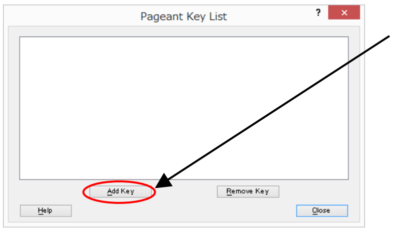
2.5.1 SSH Agent Forwarding (Windows Environment)

(1) Launch Pageant utility provided with PUTTY package

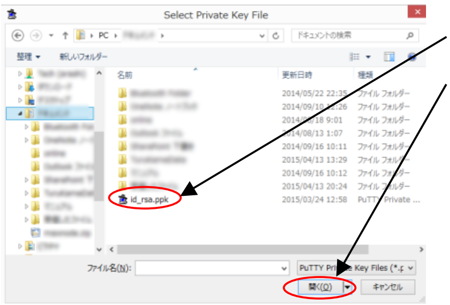


1. Right-click Pageant.
2. Click [View Keys].

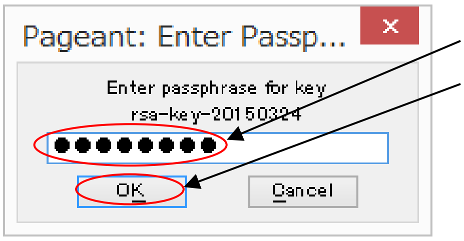
(2) Register the authentication key.



3. Click [Add Key] button.



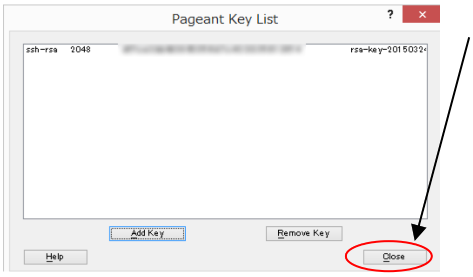
4. Select the private key.
5. Click [Open] button.


Pageant: Enter Passp...

Enter passphrase for key rsa-key-20150324

●●●●●●●●

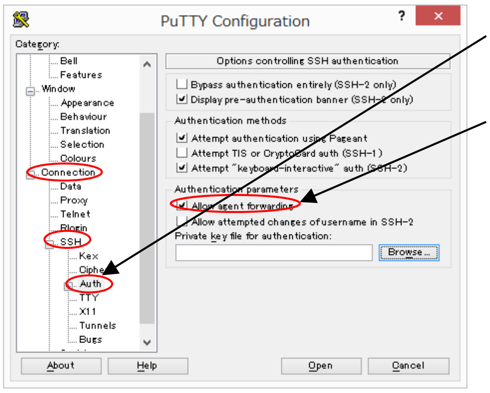
6. Enter the passphrase.
7. Click [OK] button.


Pageant Key List

ssh-rsa 2048 [redacted] rsa-key-2015032

8. Click "Close" button.

(3) Launch PuTTY, enable SSH Agent Forwarding and access to HOKUSAI BigWaterfall system.



Options controlling SSH authentication

☐ Bypass authentication entirely (SSH-2 only)
☑ Display pre-authentication banner (SSH-2 only)

Authentication methods

☑ Attempt authentication using Pageant
☐ Attempt TIS or CryptoCard auth (SSH-1)
☑ Attempt "keyboard-interactive" auth (SSH-2)

Authentication parameters

☑ Allow agent forwarding
☐ Allow attempted changes of username in SSH-2

Private key file for authentication:

9. Click [Connection] – [SSH] – [Auth]
10. Check [Allow agent forwarding]

2.5.2 SSH Agent Forwarding (Mac Environment/Ubuntu Environment)

The SSH Agent Forwarding is automatically launched on the Mac OS X(Yosemite) environment and the Ubuntu(14.10) environment. The SSH Agent Forwarding will be enabled if you specify the -A option when accessing to the HOKUSAI BigWaterfall system.

```
[username@Your-PC ~]$ ssh -A -l username hokusai.riken.jp
```

2.6 File Transfer

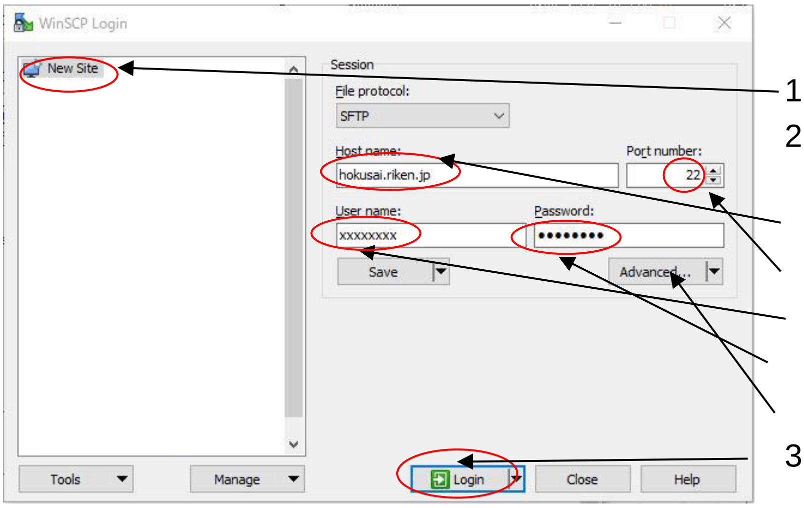
2.6.1 File Transfer (Windows Environment)

This section describes how to use WinSCP for file transfer between the your PC and HOKUSAI BigWaterfall system. WinSCP can be downloaded from the following site:

WinSCP: http://winscp.net/eng/index.php

WinSCP can be used to transfer files by drag-and-drop operation after logging into HOKUSAI BigWaterfall system.

(4) Launch WinSCP for login



WinSCP Login

Session

File protocol: SFTP

Host name: hokusai.riken.jp

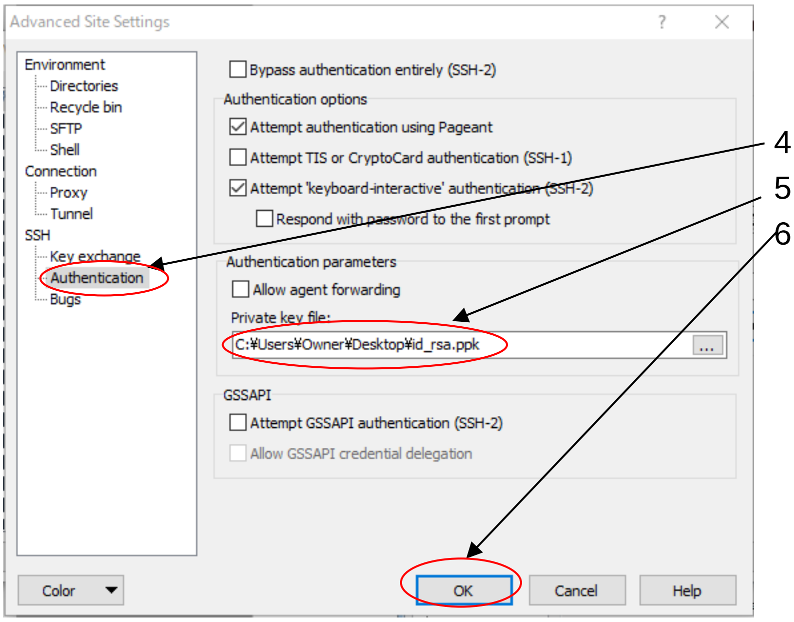
Port number: 22

User name: xxxxxxxx

Password: ••••••••

1. Click [Session]
2. Enter the following information:
   [Host name] greatwave.riken.jp
   [Port number] 22
   [User name] username
   [Password] passphrase
3. Click [Advanced] button

Figure 2-43 WinSCP Login



Advanced Site Settings

☐ Bypass authentication entirely (SSH-2)

Authentication options

☑ Attempt authentication using Pageant

☐ Attempt TIS or CryptoCard authentication (SSH-1)

☑ Attempt 'keyboard-interactive' authentication (SSH-2)

☐ Respond with password to the first prompt

Authentication parameters

☐ Allow agent forwarding

Private key file: C:¥Users¥Owner¥Desktop¥id_rsa.ppk

GSSAPI

☐ Attempt GSSAPI authentication (SSH-2)

☐ Allow GSSAPI credential delegation

4. Click [Authentication]
5. Select private key file
6. Click [OK] button

Figure 2-44 WinSCP Settings

(5) Files can be downloaded or uploaded by drag-and-drop operation.

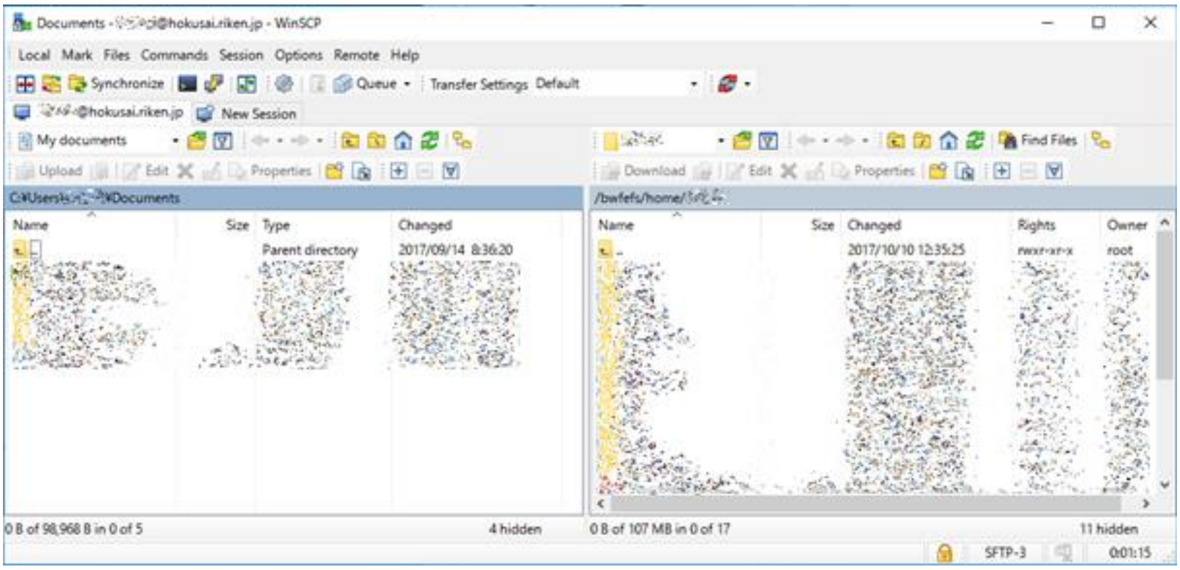


Figure 2-45 Screen after login with WinSCP

2.6.2 File Transfer (UNIX/Mac Environment)

To transfer files between the HOKUSAI BigWaterfall system and PC, use the *scp* (*sftp*) command.

```
$ scp local-file username@hokusai.riken.jp:remote-dir
Enter passphrase for key : ++++++++     ← Enter passphrase
local-file    100% |***********************|  file-size  transfer-time
```

2.7 Login Shell

The default login shell is set to bash. If you would like to change login shell, please contact hpc@riken.jp. The configuration files for login shell to use the HOKUSAI BigWaterfall system are located in each user's home directory. The original (Skelton) files are located on /etc/skel.

⚠ **When you customize environment variables such as PATH, you need to add the new path to the last. Otherwise, you cannot use the system correctly.**

3. File Systems Overview

3.1 Storage Area

The following storage areas are available in the HOKUSAI BigWaterfall system.

Table 3-1 Storage areas

Area	Size	Purpose
/home	5 PB	Home area
/data		Data area (application is required per project)
/tmp_work		Temporary area
/arc	7.9 PB	Archive area (application is required per project)

Each storage can be accessed from each server as follows.

Table 3-2 Accessibility of each server to each storage

Storage Type	Front end servers	BWMPC	GWACSL
Online Storage	○	○	○
Hierarchical Storage	○	×	×

○ : available × : unavailable

3.1.1 Online Storage

The Online Storage provides home area for users and data area for the projects. The quota of home area is 4 TB per user. To use data area, application is required.

Table 3-3 Quota of the Online Storage

Directory	User Directory	Block quota	Inode quota
/home	/home/username	4 TB	10 millions
/data	/data/projectID	4 TB - 52 TB	One million per 1 TB

3.1.2 Hierarchical Storage

The Hierarchical Storage can be accessed from the front end servers. To use this area, application is required.

Table 3-4 Quota of the Hierarchical Storage

User Directory	Quota	Inode quota
/arc	2 tape volumes/8TB - 13 tape volumes/52TB	40,000 per tape volume

The main purposes of use are as follows:

- Store a large data for a long period
- Backup

Since the Hierarchical Storage stores data in tapes, it is not suitable to store smaller files such as less than 100 MB. If you store many small files, create single file archive and then store.

3.1.3 Application of using storage area

To use /data or /arc storage area, fill the following format and send an e-mail to hpc@riken.jp.

Select subject from below:
New request: /data
Additional request: /data
New request: HSM
Additional request: HSM

Message:
(1) Project ID:
(2) Name of management representative(in case project representative delegate someone):
(3) Size of request: /data(   )TB, HSM (   )TB
(4) Current permitted size and usage rate (It is needed for additional request only):
(5) Reason of estimation of increase of data:

3.2 Disk usage

You can use the *listquota* command to display your disk usage and quota.

```
[username@hokusai1 ~]$ listquota
Disk quota for user username
          [Size]  Used(GB)  Limit(GB) Use(%) [Files]   Used(K)   Limit(K) Use(%)
--------------------------------------------------------------------------------
/home/username         289       4,000   6.7%               579     10,000   5.8%

Disk quota for project(s).
          [Size]  Used(GB)  Limit(GB) Use(%) [Files]   Used(K)   Limit(K) Use(%)
--------------------------------------------------------------------------------
Q99999
 +- /data/Q99999         -       4,000   0.0%                 -      1,000   0.0%
```

Table 3-5 listquota information

Item	Description
[Size]	Block usage (GB), quota (GB) and the ratio
[Files]	Inode usage (K), quota (K) and the ratio
/home/username /data/projectID /arc/projectID	Area /home : Online Storage (home area) /data: Online Storage (data area) *1 /arc: Hierarchical Storage (archive area) *1

*1 : The data area and archive area appear only when application is approved.

3.3 Temporary area

/tmp_work is available to store temporary files.

The data stored under /tmp_work is automatically removed when its modified date becomes older than one week. Use this storage area only for storing temporary files exclusively.

The users can use the *mktmp_work* command to pass data between each user. The *mktmp_work* command creates a temporary directory under /tmp_work, then the user can copy the file and change the permission to be passed on this directory, and notify the other party of this directory path.

```
[username@hokusai1 ~]$ mktmp_work
mktmp_work: INFO: /tmp_work/username.1G2XFQ3O/KXrWerIviNTzZnhO is created.
[username@hokusai1 ~]$ cp input.dat ¥
    /tmp_work/username.1G2XFQ3O/KXrWerIviNTzZnhO/
[username@hokusai1 ~]$ chmod o+r ¥
    /tmp_work/username.1G2XFQ3O/KXrWerIviNTzZnhO/input
```

Users other than the one who run the *mktmp_work* command are allowed only to view the files.

4. Compile and Link

4.1 Set environment settings

The *module* command enables you to set environment variables to use compilers, libraries, applications, tools and so on.

```
$ module <subcommand> <subcommand-args>
```

The sub-commands of the *module* command are the following:

Table 4-1 Sub-commands of the *module* command

Sub command	Description
avail	List all available settings
list	List loaded settings
load module...	Load setting(s) into the shell environment
unload module...	Remove setting(s) from the shell environment
purge	Unload all loaded settings
switch module1 module2	Switch loaded module1 with modules

Example) List all available settings.

```
[username@hokusai1 ~]$ module avail

------- /bwfefs/opt/modulefiles/hokusai/apps --------
ansys/19.2(default)      vmd/1.9.2(default)
gaussview/6.0.16(default)

----- /bwfefs/opt/modulefiles/hokusai/compilers -----
cuda/7.5
cuda/8.0(default)
gcc/4.8.4(default)
intel/17.2.174
intel/19.3.199
intel/19.5.281(default)
```

* The version listed with “(default)” is the recommended version on HOKUSAI BigWaterfall system.

Example) Load compiler's setting for the BWMPC.

```
[username@hokusai1 ~]$ module load intel
```

Example) List loaded settings.

```
[username@hokusai1 ~]$ module list
  Currently Loaded Modulefiles:
  1) /bwfefs/opt/modulefiles/x86_64/intelmpi/2019.5.281
  2) intel/19.5.281
```

You cannot load the settings which conflict with the loaded settings at once. If you try to load setting, the following error messages are displayed and it failed to set.

```
[username@hokusai1 ~]$ module load intel/19.3.199
  intel/19.3.199(60):ERROR:150: Module 'intel/19.3.199' conflicts with the
currently loaded module(s) 'intel/19.5.281'
intel/19.3.199(60):ERROR:102: Tcl command execution failed: conflict intel
```

To switch the compiler or switch the version of loaded library, use the "switch" sub-command.

Example) Switch the Intel compiler from version 19.5.281 to version 19.3.199.

```
[username@hokusai1 ~]$ module switch intel/19.5.281 intel/19.3.199
```

4.2 Compiler

On the front end servers of the HOKUSAI BigWaterfall system, the compilers to create load modules which run on the Massively Parallel Computer and the Application Computing Server with Large Memory are available.

Example) Compile and link for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

```
[username@hokusai1 ~]$ module load intel
[username@hokusai1 ~]$ module list
Currently Loaded Modulefiles:
  1) /bwfefs/opt/modulefiles/x86_64/intelmpi/2019.5.281
  2) intel/19.5.281
```

4.3 How to Compile and Link

The commands for compilation and linkage are as follows:

Table 4-1 Compile and link commands for the Massively Parallel Computer and the Application Computing Server with Large Memory.

Type	Programming language	Command	Automatic parallelization[*1]	OpenMP[*1]
Sequential (no MPI)	Fortran	ifort	-parallel	-qopenmp
	C	icc		
	C++	icpc		
MPI parallel	Fortran	mpiifort		
	C	mpiicc		
	C++	mpiicpc		

*1: Automatic parallelization and OpenMP options are not set by default.

4.3.1 Compile and Link for Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

4.3.1.1 Compile and link sequential programs

To compile and link sequential programs for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory. on the front end servers, use the *ifort/icc/icpc* command.

```
ifort/icc/icpc[option] file [...]
```

Example 1) Compile and link a sequential Fortran program for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

```
[username@hokusai1 ~]$ ifort seq.f
```

Example 2) Compile and link a sequential C program for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

```
[username@hokusai1 ~]$ icc seq.c
```

4.3.1.2 Compile and link thread parallelization programs

To compile and link multi-threaded programs for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory. on the front end servers, use the *ifort/icc/icpc* command.

```
ifort/icc/icpc thread-option [option] file [...]
```

Example 1) Compile and link a Fortran program with automatic parallelization for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

```
[username@hokusai1 ~]$ ifort -parallel para.f
```

Example 2) Compile and link a C program with automatic parallelization for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory..

```
[username@hokusai1 ~]$ icc -parallel para.c
```

Example 3) Compile and link an OpenMP Fortran program for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

```
[username@hokusai1 ~]$ ifort -qopenmp omp.f
```

Example 4) Compile and link an OpenMP C program for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

```
[username@hokusai1 ~]$ icc -qopenmp omp.c
```

Example 5) Compile and link an OpenMP Fortran program with automatic parallelization for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

```
[username@hokusai1 ~]$ ifort -parallel –qopenmp omp_para.f
```

Example 6) Compile and link an OpenMP C program with automatic parallelization for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

```
[username@hokusai1 ~]$ icc -parallel –qopenmp omp_para.c
```

4.3.1.3 Compile and link MPI programs

To compile and link MPI programs for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory on the front end servers, use the *mpiifort/mpiicc/mpiicpc* command.

```
mpiifort/mpiicc/mpiicpc [option] file [...]
```

Example 1) Compile and link a MPI Fortran program for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

```
[username@hokusai1 ~]$ mpiifort mpi.f
```

Example 2) Compile and link a MPI C program for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

```
[username@hokusai1 ~]$ mpiicc mpi.c
```

Example 3) Compile and link a Hybrid (MPI + OpenMP) Fortran program for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

```
[username@hokusai1 ~]$ mpiifort –qopenmp mpi_omp.f
```

Example 4) Compile and link a Hybrid (MPI + OpenMP) C program for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory.

```
[username@hokusai1 ~]$ mpiicc -qopenmp mpi_omp.c
```

4.3.1.4 Optimize for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory

⚠ Optimization may affect computation results. If you apply optimization, verify execution of the execution result.

Figure 4-2 General optimization option

Compile Options	Description
-O0	Disables all optimizations.
-O1	Enables optimizations for speed and disables some optimizations that increase code size and affect speed. To limit code size, this option.
-O2	Enables optimizations for speed. This is the generally recommended optimization level. Vectorization is enabled at -O2 and higher levels.
-O3	Performs -O2 optimizations and enables more aggressive loop transformations such as Fusion, Block-Unroll-and-Jam, and collapsing IF statements.
-fast	Maximizes speed across the entire program. It sets the following options: -ipo, -O3, -no-prec-div, -static, -fp-model fast=2, -xHost * The -static option is not available when linking MPI programs.
-qopt-report[=*n*]	Tells the compiler to generate an optimization report. You can specify values 0 through 5 as *n*. The default is level 2.
-qopt-report-phase[=*list*]	Specifies one or more optimizer phases for which optimization reports are generated. For more detail, refer to man manual.
-qopt-report-help	Displays the optimizer phases available for report generation and a short description of what is reported at each level. No compilation is performed.
-qopt-report-routine=string	Tells the compiler to generate an optimization report for each of the routines whose names contain the specified substring. When optimization reporting is enabled, the default is -qopt-report-phase=all.

Table 4-3 Parallel peformance options

Compile Options	Description
-qopenmp	Enables the parallelizer to generate multi-threaded code based on OpenMP directives.
-parallel	Tells the auto-parallelizer to generate multi-threaded code for loops that can be safely executed in parallel.
-par-threshold[*n*]	Sets a threshold for the auto-parallelization of loops. (from n=0 to n=100. Default: n=100)。 0 – Loops get auto-parallelized always, regardless of computation work volume. 100 – Loops get auto-parallelized when performance gains are predicted based on the compiler analysis data. Loops get auto-parallelized only if profitable parallel execution is almost certain. To use this option, you must also specify option -parallel.
-guide[=*n*]	Lets you set a level of guidance for auto-vectorization, auto parallelism, and data transformation. When this option is specified, the compiler does not produce any objects or executables. You must also specify the -parallel option to receive auto parallelism guidance. The values available are 1 through 4. Value 1 indicates a standard level of guidance. Value 4 indicates the most advanced level of guidance. If n is omitted, the default is 4.
-qopt-matmul	Enables or disables a compiler-generated Matrix Multiply (matmul). The -qopt-matmul options tell the compiler to identify matrix multiplication loop nests (if any) and replace them with a matmul library call for improved performance. The resulting executable may get additional performance gain. This option is enabled by default if options -O3 and -parallel are specified. To disable this optimization, specify -qno-opt-matmul. This option has no effect unless option O2 or higher is set.
-coarray=shared	Enables the coarray feature of the Fortran 2008 standard.

Table 4-4 Processor-specific optimization options

Compile Options	Description
-x*target*	Tells the compiler which processor features it may target, including which instruction sets and optimizations it may generate.　When you build only for ACS with GPU, specify option -xCORE-AVX2 for the Haswell microarchitecture.
-xhost	Tells the compiler to generate instructions for the highest instruction set available on the compilation host processor.

Table 4-5 Interprocedural Optimization (IPO) options and Profile-guided Optimization (PGO) options

Compile Options	Description
-ip	Determines whether additional interprocedural optimizations for single-file compilation are enabled.
-ipo[=*n*]	Enables interprocedural optimization between files. If n is 0, the compiler decides whether to create one or more object files based on an estimate of the size of the application. It generates one object file for small applications, and two or more object files for large applications. If you do not specify n, the default is 0.
-ipo-jobs[*n*]	Specifies the number of commands (jobs) to be executed simultaneously during the link phase of Interprocedural Optimization (IPO). The default is -ipo-jobs1.
-finline-functions -finline-level=2	Enables function inlining for single file compilation. Interprocedural optimizations occur. if you specify -O0, the default is OFF.
-finline-factor=*n*	Specifies the percentage multiplier that should be applied to all inlining options that define upper limits. The default value is 100 (a factor of 1).
-prof-gen	Produces an instrumented object file that can be used in profile-guided optimization.
-prof-use	Enables the use of profiling information during optimization.
-profile-functions	Inserts instrumentation calls at a function's entry and exit points.
-profile-loops	Inserts instrumentation calls at a function's entry and exit points, and before and after instrumentable loops.

Figure 4-6 Floating-point operation optimization options

Compile Options	Description
-fp-model *name*	Controls the semantics of floating-point calculations.
-ftz[-]	Flushes denormal results to zero when the application is in the gradual underflow mode. It may improve performance if the denormal values are not critical to your application's behavior.
-fimf-precision:*name*	Lets you specify a level of accuracy (precision) that the compiler should use when determining which math library functions to use. The *name* is high, medium or low. This option can be used to improve run-time performance if reduced accuracy is sufficient for the application, or it can be used to increase the accuracy of math library functions selected by the compiler. In general, using a lower precision can improve run-time performance and using a higher precision may reduce run-time performance.
-fimf-arch-consistency=*true*	Ensures that the math library functions produce consistent results across different microarchitectural implementations of the same architecture. The -fimf-arch-consistency option may decrease run-time performance. Deafult is "false”.
-prec-div	Improves precision of floating-point divides. The result is more accurate, with some loss of performance.
-prec-sqrt	Improves precision of square root implementations. The result is fully precise square root implementations, with some loss of performance.

Figure 4-7 Detailed tuning options

Compile Options	Description
-unroll[*n*]	Tells the compiler the maximum number of times to unroll loops. To disable loop enrolling, specify 0. The default is -unroll, and the compiler uses default heuristics when unrolling loops.
-qopt-prefetch[=*n*]	Enables or disables prefetch insertion optimization. The n (0:Disable-4) is the level of software prefetching optimization desired. The option -qopt-prefetch=3 is enabled by default if option -O2 or higher is set.
-qopt-block-factor=*n*	Lets you specify a loop blocking factor.
-qopt-streaming-stores *mode*	This option enables generation of streaming stores for optimization. The *mode* is as follows: "always": Enables generation of streaming stores for optimization. The compiler optimizes unde the assumption that the application is memory bound. "never": Disables generation of streaming stores for optimization. "auto": Lets the compiler decide which instructions to use.
-fno-alias	Determines whether aliasing should be assumed in the program. Default is -fno-alias.
-fno-fnalias	Specifies that aliasing should be assumed within functions. Default is -ffnalias.
-fexceptions	Enables exception handling table generation. This option enables C++ exception handling table generation, preventing Fortran routines in mixed-language applications from interfering with exception handling between C++ routines. The -fno-exceptions option disables C++ exception handling table generation, resulting in smaller code. When this option is used, any use of C++ exception handling constructs (such as try blocks and throw statements) when a Fortran routine is in the call chain will produce an error.
-vec-threshod *n*	Sets a threshold for the vectorization of loops based on the probability of profitable execution of the vectorized loop in parallel. (from n=0 to n=100. Default: n =100) 0 – loops get vectorized always, regardless of computation work volume. 100 – loops get vectorized when performance gain are predicted based on the compiler analysis data.
-vec-report[=*n*]	Controls the diagnostic information reported by the vectorizer. The n is a value denoting which diagnostic messages to report. Default is 0.

4.4 BLAS/LAPACK/ScaLAPACK for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory

When you use BLAS/LAPACK/ScaLAPACK libraries for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory, the following options are available:

Table 4-8 BLAS/LAPACK/ScaLAPACK options

Library	Parallelism	Option	Remark
BLAS	Sequential	-mkl=sequential	
	Thread parallel	-mkl=parallel	
LAPACK	Sequential	-mkl=sequential	
	Thread parallel	-mkl=parallel	
ScaLAPACK	MPI parallel	-mkl=cluster	

Example 1) Use the sequential version BLAS/LAPACK.

```
$ ifort -mkl=sequential blas.f
```

Example 2) Use the thread parallel version BLAS/LAPACK.

```
$ ifort -mkl=parallel blas.f
```

Example 3) Use ScaLAPACK (linking the sequential version of BLAS/LAPACK).

```
$ mpiifort -mkl=cluster scalapack.f
```

About the combinations than the above, refer to the following URL:

https://software.intel.com/en-us/articles/intel-mkl-link-line-advisor

5. Batch Job and Interactive Job

5.1 Job Management System

The job management system manages all batch jobs and interactive jobs over the HOKUSAI BigWaterfall system. Users request job requirements such as resource unit, resource group, number of nodes, number of cores, and elapsed time to the job management system for the job to be executed.

There are two types of jobs users can submit to the HOKUSAI BigWaterfall system.

Table 5-1 Job types

Job type	Usage
Batch job	Execute jobs in batch mode. When a node failure occurs, your job is re-executed if the --restart option is given.
Interactive job	Execute jobs in interactive mode by entering data on user terminals. Mainly used for debugging. Jobs are not re-executed when an error such as a node failure occurs.

Batch jobs can be classified into three types depending on how they are submitted.

Table 5-2 Batch job types

Job classification	Purpose	How to submit
Normal job	Execute jobs based on a job script.	Refer to "5.4.1 Normal "
Step job	Handle multiple jobs as a group having the execution order or dependency relationship.	Refer to "5.4.2 Step Job"
Bulk job	Consist of multiple instances of the same normal job submitted at the same time for execution.	Refer to "5.4.3 Bulk Job"

Users can use the following commands to run jobs.

Table 5-3 Job commands

Function	Command
Submit a job	pjsub
See a job status	pjstat
Delete a job	pjdel
Display a job script	pjcat

5.2 Job Execution Resource

When submitting a job, specify the “resource unit” that means the hardware where a job runs and the “resource group” that means the software.

5.2.1 Resource Unit

The following three types of resource units that specify the hardware a job runs are prepared:

Table 5-4 Resource units

Resource Unit	Where the job is executed
bwmpc	Massively Parallel Computer (BWMPC)
gwacsl	ACS with Large memory (ACSL)

Although specification of the resource unit is mandatory, the following file can be used to contain the settings by which the resource unit to be used is fixed. The settings in this file are ignored when the resource unit is explicitly specified in the job script or on the command line.

Table 5-5 Fixing the resource unit to be used

Setting file name	Setting value
/home/*username*/.cltkrc	DEF_RSCUNIT="resource unit name"

Example)

```
[username@hokusai1 ~] cat $HOME/.cltkrc
DEF_RSCUNIT=bwmpc
```

5.2.1.1 Resource Unit settings for Project

Each project can use the following resources at a time.

Table 5-6 Concurrent resource usage limit for Project

Resource Unit	Number of running cores	Number of running nodes	Number of submitted jobs	Number of submitted bulk subjobs
bwmpc	General: 5,120 Quick: 1,280	General: 128 Quick: 32	500	5,000
gwacsl	120	2	100	100

5.2.2 Resource Group

The resource groups that specify the software to be executed are prepared on each resource unit. If you run an ISV application, specify an appropriate resource group when submitting a job. With some resource group, the user of general subject is allowed to use more resources than the user of simple subject.

When no resource group is specified, the following resource group is automatically selected by default.

Table 5-7 Default resource group

Job type	Default resource group
Batch job	batch
Interactive job	interact

The following section describes the list of the available resource group.

5.2.2.1 Resource Group for Massively Parallel Computer (BWMPC)

Table 5-8 Resource group for Massively Parallel Computer (BWMPC)

Resource Group	Specific use	Job Type	Maximum elapsed time*3	Maximum number of cores	Maximum number of nodes
batch	Gerenal job	Batch	72hrs	640	16
			24hrs	General:5,120 Quick:1,280	General:128 Quick: 32
gaussian	Gaussian	Batch	72hrs	40	1
qchem	Q-Chem	Batch	72h	640	16
interact*2	Interactive use	Interactive	2hrs	80	2
special*1	Large scale parallel	Batch	48hrs	33,600	840

*1 Application is required. (See the section 5.2.2.3)

*2 A user can submit and run only 1 interactive job at the same time.

*3 Default elapsed time for batch jobs is 12hrs.

5.2.2.2 **Resource Group for ACS with Large memory (ACSL)**

Table 5-9 Resource Group for ACS with Large memory (ACSL)

Resource Group	**Specific use**	**Job Type**	**Maximum elapsed time** [*2]	**Maximum number of cores**	**Maximum number of nodes**
batch	Gerenal job	Batch	24hrs	120	2
gaussian	Gaussian	Batch	24hrs	60	1
interact[*2]	Interactive use	Interactive	2hrs	120	2
special[*1]	Large scale parallel	Batch	48hrs	120	2

*1 Application is required. (See the section 5.2.2.3)

*2 A user can submit and run only 1 interactive job at the same time.

*3 Default elapsed time for batch jobs is 12hrs.

5.2.2.3 **Resource Group for Which Submitting an Application Required**

To use some resource groups, it is required to submit an application.

Table 5-10 Resource groups for which submitting an application is required

Resource Group	**Description**
special	Large scale parallel jobs that are not allowed to run during the regular operation are allowed to be executed during the specific period specified by ACCC.
ansys	ANSYS (multiphysics) can be executed for only one job simultaneously in the HOKUSAI BigWaterfall system.
adf	The ADF license only allows access from within the Wako site.

The user who wants to use above resource groups should prepare the following information and contact hpc@riken.jp

- User name, Project ID, Period
- Reason

5.3 **Job Submission Options**

When submitting jobs, specify the following three options as necessary.

- Basic Options
- Resource Options

5.3.1 Basic Options

The basic options you can specify to your job are the following.

Table 5-11 Basic options for submitting a job

Option	Description
-g *projectID*	Specify a project ID that consumes core time to execute a job
-j	Direct output of standard error of the job to standard output
--mail-list	Set an email address
-m	Set email notification
b	Send email notification on starting a job
e	Send email notification on finishing a job
r	Send email notification on re-executing a job
-N *name*	Specify a job name
-o *filename*	Write standard out put to a specified file
--restart	Specify a job not to be re-executed when a failure occurs (default: --norestart)
--interact	Submit a job as an interactive job
--step	Submit a job as a step job
jid=*jobid*	Set a job ID to associate with
sn=*subjobid*	Set an own sub-job number
sd=*form*	Set a dependency statement
--bulk --sparam start-end	Submit a job as a bulk job
-X	Inherit environment variables for used for job submission to the job execution environment
-s	Output job statistic information when finishing a job

5.3.2 Resource Options

You can specify the resources to be allocated to a job by using the -L option.

Table 5-12 Resource options (common)

Option		Description
-L		Specify upper limits of resources needed to execute jobs
	rscunit=*name*	Specify resource unit (required option)
	rscgrp=*name*	Specify resource group
	elapse=*elapselimit*	Specify elapsed time ([[hour:]minute:]second)
	vnode=*num*	Specify the number of nodes
	vnode-core=*num*	Specify the number of cores per node. - Maximum number of BWMPC: 40 - Maximum number of ACSL: 60
	core-mem=*size*	Specify the amount of memory per core - Maximum amount of BWMPC: 88,000Mi - Maximum amount of ACSL: 960Gi
	proc-core=*size*	Specify a maximum core file size limit for a process (default: 0, maximum: unlimited)
	proc-data=*size*	Specify a maximum data segment size limit for a process (default: unlimited)
	proc-stack=*size*	Specify a maximum stack segment size limit for a process (default: unlimited)

When you set the amount of memory, the units can be set as following string:

Table 5-13 Available unit for the amount of memory

Unit	Description
Ki	kibibyte (2^10)
Mi	mebibyte (2^20)
Gi	gibibyte (2^30)

The default amount of memory per core is as follows:

Table 5-14 Default amount of memory per core

System	Default amount of memory per core
Massively Parallel Computer (BWMPC)	2,200Mi
ACS with Large memory (ACSL)	24Gi

When you require the memory more than default amount of memory per core, more cores could be allocated based on required memory. Be aware that the core time is calculated based on the number of allocated cores and elapsed time.

Example) Request 8800Mi as amount of memory per core for the Massively Parallel Computer(BWMPC)

```
[username@hokusai1 ~]$ pjsub --interact -L rscunit=bwmpc -L "vnode-core=
1,core-mem=8800Mi" -g G99999
pjsub: WARNING: submitted job uses more cpu-core than specified due to
the size of memory. (1 -> 4)
[INFO] PJM 0000 pjsub Job 29774 submitted.
[INFO] PJM 0081 .connected.
[INFO] PJM 0082 pjsub Interactive job 29774 started.
[username@ bwmpc0837 ~]$ numactl --show
policy: default
preferred node: current
physcpubind: 0 1 2 3        ← 4cores are allocated
cpubind: 0
nodebind: 0
membind: 0 1
```

When you don't specify the number of processes/threads with the MPI options/OMP_NUM_THREADS environment variable, the program may run with the unintended number of processed/threads and the performance may degrade.

5.4 **Submit Batch Jobs**

To execute a batch job, the user creates a "job script" in addition to a program and submits the job script to the job management system as a batch job. The description of a command line includes options such as a resource unit, a resource group, elapsed time and the number of nodes as well as commands to be executed. The user uses the *pjsub* command to submit a job script. The submitted jobs are automatically executed by the job management system based on the status of free computing resources and the priority among projects.

5.4.1 Normal Job

To submit a normal job, use the *pjsub* command with the job script which is executed as a batch job.

```
pjsub [option] [job-script]
```

- If a job script is not specified, a script is read from standard input.
- Job submission options can be set by defining directives in a job script or in standard input.
- If a job is successfully submitted, an identification number (job ID) is assigned to the job.

Example) Submit a normal job.

```
[username@hokusai1 ~]$ pjsub  run.sh
[INFO]PJM 0000 pjsub Job 12345 submitted.
```

5.4.2 Step Job

A step job is a job model that aggregates multiple batch jobs and defines a job chain having an execution order and dependency of the batch jobs. A step job consists of multiple sub-jobs, which are not executed concurrently. The figure below outlines the process sequence of a step job.

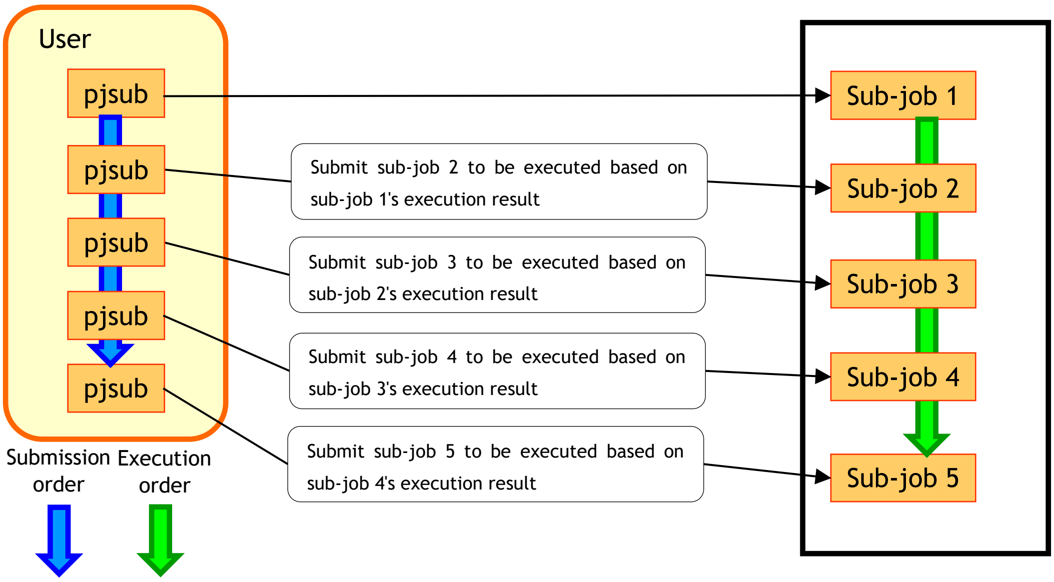

User
pjsub
pjsub
pjsub
pjsub
pjsub
Submission order
Execution order
Submit sub-job 2 to be executed based on sub-job 1's execution result
Submit sub-job 3 to be executed based on sub-job 2's execution result
Submit sub-job 4 to be executed based on sub-job 3's execution result
Submit sub-job 5 to be executed based on sub-job 4's execution result
Sub-job 1
Sub-job 2
Sub-job 3
Sub-job 4
Sub-job 5

Figure 5-1 General flow of a step job

The format of submitting a step job is as follows:

```
pjsub --step [--sparam "sn=stepno[,dependency]"] jobscript[,jobscript...]
```

Example 1) Submit a step job containing three sub-jobs

```
[username@hokusai1 ~]$ pjsub --step stepjob1.sh
[INFO]PJM 0000 pjsub Job 12345_0 submitted.
[username@hokusai1 ~]$ pjsub --step --sparam jid=12345 stepjob2.sh
[INFO]PJM 0000 pjsub Job 12345_1 submitted.
[username@hokusai1 ~]$ pjsub --step --sparam jid=12345 stepjob3.sh
[INFO]PJM 0000 pjsub Job 12345_2 submitted.
```

Example 2-1) Submit a step job containing three sub-jobs at a time (When a failure occurred, the affected job is failed and the following jobs will be continued.)

```
[username@hokusai1 ~]$ pjsub --step step1.sh step2.sh step3.sh
[INFO]PJM 0000 pjsub Job 12345_0 submitted.
[INFO]PJM 0000 pjsub Job 12345_1 submitted.
[INFO]PJM 0000 pjsub Job 12345_2 submitted.
```

Example 2-2) Submit a step job containing three sub-jobs at a time (When a failure occurred, the affected job is run again.)

```
[username@hokusai1 ~]$ pjsub --step --restart step1.sh step2.sh ¥
step3.sh
[INFO]PJM 0000 pjsub Job 12345_0 submitted.
[INFO]PJM 0000 pjsub Job 12345_1 submitted.
[INFO]PJM 0000 pjsub Job 12345_2 submitted.
```

Example 2-3) Submit a step job containing three sub-jobs at a time (When a failure occurred, the affected job and the following jobs are canceled.)

```
[username@hokusai1 ~]$ pjsub --step --sparam "sd=pc!=0:all" ¥
step1.sh step2.sh step3.sh
[INFO]PJM 0000 pjsub Job 12345_0 submitted.
[INFO]PJM 0000 pjsub Job 12345_1 submitted.
[INFO]PJM 0000 pjsub Job 12345_2 submitted.
```

Example 3) Submit a step job containing three sub-jobs with step number and dependency statement options

```
[username@hokusai1 ~]$ pjsub --step --sparam "sn=1" ¥
       stepjob1.sh
[INFO]PJM 0000 pjsub Job 12345_1 submitted.
[username@hokusai1 ~]$ pjsub --step --sparam ¥
       "jid=12345, sn=2, sd=ec!=0:after:1" stepjob2.sh
[INFO]PJM 0000 pjsub Job 12345_2 submitted.
[username@hokusai1 ~]$ pjsub --step --sparam ¥
       "jid=12345, sn=3, sd=ec==0:one:1" stepjob3.sh
[INFO]PJM 0000 pjsub Job 12345_3 submitted.
```

Table 5-15 Step job dependency statements

Condition	Description
NONE	Indicate no dependency
ec == value[,value,value..] ec != value[,value,value..] ec > value ec >= value ec < value ec <= value	Value can be any number For "==" and "!=", multiple values separated with a comma can be specified. Example: ec==1,3,5 → True if the termination status is 1, 3 or 5 ec!=1,3,5 → True if the termination status is not 1, 3 or 5

Table 5-16 Cancellation types available for step job dependency statements

Cancellation type	Description
one	Cancel only the current job
after	Cancel the current job and recursively cancel jobs dependent on the current job
all	Cancel the current job and all subsequent jobs

5.4.3 Bulk Job

A bulk job consists of multiple instances of the same normal job submitted at the same time for execution. For example, suppose the user wants to change the job parameters and check the execution results for each change. The user would need to submit one normal job for each change. However, by using a bulk job, the user can submit multiple patterns at one time for one job.

The format of submitting a bulk job is as follows:

```
pjsub --bulk --sparam start-end jobscript
```

A job script for a bulk job is designed such that input/output of the job can be changed for each sub job. For this reason, the bulk job uses the bulk number that is set for the sub job. The bulk number is set in the PJM_BULKNUM environment variable in the sub job.

5.4.4 Job Output

A batch job's standard output file and standard error output file are written under the job submission directory or to files specified at job submission.

Standard output generated during the job execution is written to a standard output file and error messages generated during the job execution are written to a standard error output file. If no standard output and standard error output files are specified at job submission, the following files are generated for output.

Jobname.oXXXXX --- Standard output file
Jobname.eXXXXX --- Standard error output file
(XXXXX is a job ID assigned at job submission)

5.4.5 Job Script

To submit a batch job, create a job script using the *vi* command or the *emacs* command.

(1) At the top of a job script, put "#!" followed by a path of shell.
[Sample]

```
#!/bin/sh
```

⚠ If your login shell is not bash and you execute the module commands in the job script written in sh, you need to specify "#!/bin/sh -l".

(2) From the second line onward, specify submission options using directives starting with "#PJM".
[Sample]

```
#PJM -L vnode=1                      Specify a number of nodes
#PJM -L elapse=1:00:00      Specify elapsed time limit
#PJM -j                     Merge the standard error
```

(3) After job submission options, set runtime environment variables and specify program execution.
[Sample]

```
export OMP_NUM_THREADS=20   Set environment variable
./a.out                     Run a program
```

5.4.6 NUMA Architecture

The each node of Massively Parallel Computer (BWMPC) and the Application Computing Server with Large Memory uses the NUMA (Non-Uniform Memory Access) architecture. It is expected that the assigning the processes and threads in consideration of the memory access decreases the execution time. For example, we recommend to specify the following the number of threads when you execute a multi-threaded program.

Table 5-17 Recommended number of threads

System	Recommended number of threads
Massively Parallel Computer (BWMPC)	20 or less
ACS with Large memory (ACSL)	15 or less

5.4.7 Execute MPI Program

5.4.7.1 Mpirun options

Table 5-18 mpirun options

Option	Description
-np *n*	Specifies the number of parallel processes for the MPI program.
-ppn *n*	Places consecutive *n* processes on each host
-rr	Involves "round robin" startup scheme. Equivalent to -ppn 1.
-s *spec*	Redirects stdin to all or 1,2 or 2-4,6 MPI processes (0 by default).
-prepend-rank	Prepends rank to output.

5.5 Example script for batch job

5.5.1 Job Script for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory

5.5.1.1 Sequential Job Script on Single Node for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory

The following is a sample script for executing the job below.

• Resource Unit	: bwmpc
• Resource Group	: batch
• Number of nodes	: 1 node
• Number of processes (threads)	: 1 process (1 thread)
• Elapsed time	: 60 minutes
• ProjectID	: Q99999
• Merging standard error output with standard output	: Yes

```
[username@hokusai1 ~]$ vi bwmpc-seq.sh
 #!/bin/sh
 #------ pjsub option --------#
 #PJM -L rscunit=bwmpc
 #PJM -L rscgrp=batch
 #PJM -L vnode=1
 #PJM -L vnode-core=1
 #PJM -L elapse=60:00
 #PJM -g Q99999
 #PJM -j
 #------- Program execution -------#
 ./a.out
```

5.5.1.2 Multi-threaded Job Script on Single Node for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory

The following is a sample script for executing the job below.

• Resource Unit	: bwmpc
• Resource Group	: batch
• Number of nodes	: 1 node
• Number of processes (threads)	: 1 process (10 threads)
• Elapsed time	: 60 minutes
• ProjectID	: Q99999
• Merging standard error output with standard output	: Yes

```
[username@hokusai1 ~]$ vi bwmpc-para.sh
 #!/bin/sh
 #------ pjsub option --------#
 #PJM -L rscunit=bwmpc
 #PJM -L rscgrp=batch
 #PJM -L vnode=1
 #PJM -L vnode-core=10
 #PJM -L elapse=60m
 #PJM -g Q99999
 #PJM -j
 #------- Program execution -------#
 export OMP_NUM_THREADS=10
 ./a.out
```

When you run a thread parallelized program on the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory, OMP_NUM_THREADS environment variable must be specified. According to specifying amount of memory, a number of allocated cores changes. When you don't specify the number of threads, the program may run with the unintended number of threads and the performance may degrade.

OMP_NUM_THREADS ：Number of threads (-L vnode-core option)

5.5.1.3 MPI Parallel Job Script on Single Node for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory

The following is a sample script for executing the job below.

• Resource Unit	: bwmpc
• Resource Group	: batch
• Number of nodes	: 1 node
• Number of processes (threads)	: 20 processes (1 thread)
• Elapsed time	: 1 hour
• ProjectID	: Q99999
• Merging standard error output with standard output	: Yes

```
[username@hokusai1 ~]$ vi bwmpc-single-mpi.sh
 #!/bin/sh
 #------ pjsub option --------#
 #PJM -L rscunit=bwmpc
 #PJM -L rscgrp=batch
 #PJM -L vnode=1
 #PJM -L vnode-core=20
 #PJM -L elapse=1h
 #PJM -g Q99999
 #PJM -j
 #------- Program execution -------#
 mpirun -np 20 ./a.out
```

When you run a MPI program on the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory, the -np option of the *mpirun* command must be specified. According to specifying amount of memory, a number of allocated cores changes. When you don't specify the number of processes, the program may run with the unintended number of processes and the performance may degrade.

-np　　　：Number of total processes (-L vnode-core)

5.5.1.4 **Hybrid (Multi-thread + MPI) Parallel Job Script on Single Node for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory**

The following is a sample script for executing the job below.

• Resource Unit	: bwmpc
• Resource Group	: batch
• Number of nodes	: 1 node
• Number of processes (threads)	: 2 processes (10 threads)
• Number of cores	: 20 cores (2 x 10)
• Elapsed time	: 3,600 seconds
• ProjectID	: Q99999
• Merging standard error output with standard output	: Yes

```
[username@hokusai1 ~]$ vi bwmpc-single-hybrid.sh
 #!/bin/sh
 #------ pjsub option --------#
 #PJM -L rscunit=bwmpc
 #PJM -L rscgrp=batch
 #PJM -L vnode=1
 #PJM -L vnode-core=20
 #PJM -L elapse=3600
 #PJM -g Q99999
 #PJM -j
 #------- Program execution -------#
 export OMP_NUM_THREADS=10
 mpirun -np 2 ./a.out
```

When you run a Hybrid program on the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory, the -np option of the *mpirun* command and OMP_NUM_THREADS environment variable must be specified. According to specifying amount of memory, a number of allocated cores changes. When you don't specify the number of processes/threads, the program may run with the unintended number of processes/threads and the performance may degrade.

OMP_NUM_THREADS ：Number of threads (-L vnode-core / number of total processes)
-np ：Number of total processes

5.5.1.5 **MPI Parallel Job Script on Multinode for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory**

The following is a sample script for executing the job below.

• Resource Unit	: bwmpc
• Resource Group	: batch
• Number of nodes	: 2 node
• Number of processes (threads)	: 80 processes (1 threads)
• Number of processes per node	: 40 processes
• Elapsed time	: 90 minutes
• ProjectID	: Q99999
• Merging standard error output with standard output	: Yes

```
[username@hokusai1 ~]$ vi bwmpc-multi-mpi.sh
 #!/bin/sh
 #------ pjsub option --------#
 #PJM -L rscunit=bwmpc
 #PJM -L rscgrp=batch
 #PJM -L vnode=2
 #PJM -L vnode-core=40
 #PJM -L elapse=90m
 #PJM -g Q99999
 #PJM -j
 #------- Program execution -------#
 mpirun -np 80 -ppn 40 ./a.out
```

When you run a MPI program on the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory, the -np option and the --ppn option of the *mpirun* command must be specified. According to specifying amount of memory, a number of allocated cores changes. When you don't specify the number of processes, the program may run with the unintended number of processes and the performance may degrade.

-np	：Number of total processes
-ppn	：Number of processes per node

5.5.1.6 Hybrid (Multi-thread + MPI) Parallel Job Script on Single Node for the Massively Parallel Computer (GWMPC) / Application Computing Server with Large Memory

The following is a sample script for executing the job below.

• Resource Unit	: bwmpc
• Resource Group	: batch
• Number of nodes	: 2 node
• Number of processes (threads)	: 4 processes (20 threads)
• Number of processes per node	: 2 processes
• Elapsed time	: 1 hour 30 minutes
• ProjectID	: Q99999
• Merging standard error output with standard output	: Yes

```
[username@hokusai1 ~]$ vi bwmpc-multi-hybrid.sh
 #!/bin/sh
 #------ pjsub option --------#
 #PJM -L rscunit=bwmpc
 #PJM -L rscgrp=batch
 #PJM -L vnode=2
 #PJM -L vnode-core=40
 #PJM -L elapse=1:30:00
 #PJM -g Q99999
 #PJM -j
 #------- Program execution -------#
 export OMP_NUM_THREADS=20
 mpirun -np 4 -ppn 2 ./a.out
```

When you run a Hybrid program on the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory, the -np option and the --ppn option of the *mpirun* command, and OMP_NUM_THREADS environment variable must be specified. According to specifying amount of memory, a number of allocated cores changes. When you don't specify the number of processes/threads, the program may run with the unintended number of processes/threads and the performance may degrade.

OMP_NUM_THREADS ：Number of threads (-L vnode-core / number of processes per node)
-np ：Number of total processes
-ppn ：Number of processes per node

5.6 Execute Interactive Jobs

To execute an interactive job, specify the "--interact" option on the *pjsub* command line. The job management system allocates interactive jobs to execute in interactive mode.

When submitting an interactive job, job submission options are specified as arguments on the command line.

```
pjsub --interact [--sparam wait-time=sec] [option...]
```

By specifying the wait time, the interactive job will wait for the specified time and resource assignment if the computing resource is insufficient. (The interactive job does not wait without specifying wait-time.)

⚠ When no command is executed for 10 minutes in the interactive job, the interactive job ends

Example 1) Execute an interactive job for the BWMPC.

```
[username@hokusai1 ~]$ pjsub --interact -L rscunit=bwmpc -g Q99999
[INFO] PJM 0000 pjsub Job 12345 submitted.
[INFO] PJM 0081 .connected.
[INFO] PJM 0082 pjsub Interactive job 12345 started.
[username@bwmpc0001 ~]$ ifort hello_world.f95
[username@bwmpc0001 ~]$ ./a.out
Hello world
[username@bwmpc0001 ~]$ exit
exit
[INFO] PJM 0083 pjsub Interactive job 12345 completed.
```

5.7 Job Status

Use the *pjstat* command to check the status of submitted jobs and resource information.

```
pjstat [option] [JOBID[JOBID…]]
```

Table 5-19 pjstat option

Option	Description
None	Display information of queuing jobs and running jobs.
-A	Display information of jobs of all users in the same project.
-g projectID	Display information of jobs which belong specified project.
-E	Display step job and bulk job information.
-v	Display additional job information that is not included in the standard format.
-s	In addition to information displayed with the -v option, detailed information such as resources usage status and resource limitations is displayed.
--rsc	Display resource group information.
--un	Display node status
--uc	Display core status
-p	Display priority order of projects
-x	Display max resources (cores, nodes, elapse)

5.7.1 Job status

The *pjstat* command displays status of jobs that are currently running or are in the queue.

<u>Because a projected time on the START_DATE field the indication, a projected time fluctuates based on system congestion and priority amoung projects.</u>

```
[username@hokusai1 ~]$ pjstat


   ACCEPT QUEUED  STGIN  READY RUNING RUNOUT STGOUT   HOLD  ERROR   TOTAL
        0      1      0      0      1      0      0      0      0       2
s       0      1      0      0      1      0      0      0      0       2
JOB_ID  JOB_NAME MD ST  USER     START_DATE      ELAPSE_LIM NODE_REQUIRE  VNODE  CORE V_MEM
1234    job.sh   NM RUN username 01/01 01:00:00  0012:00:00 -             2      12   1024 MiB
1235    job.sh   NM QUE username (01/02 00:00)   0012:00:00 -             2      12   1024 MiB
```

Table 5-20 Job status

Field	Description
JOB_ID	Job ID For sub-jobs, Subjob ID
JOB_NAME	Job name
MD	Job model (NM: Normal job, ST: Step job, BU: Bulk job)
ST	Job state (See Table 5-21 Job state)
USER	User name who executed the job
START_DATE	Projected start time or time started "(MM/DD hh:mm)" After the execution is started "MM/DD hh:mm:ss" As for jobs to which backfill is applied, "<" is added after the time. "(MM/DD hh:mm)<" or "MM/DD hh:mm:ss<"
ELAPSE_LIM	Elapsed time limit "hhhh:mm:ss"
NODE_REQUIRE	"-" is output.
VNODE	Number of nodes
CORE	Number of cores per node
V_MEM	Amount of memory per node

Table 5-21 Job state

Status	Description
ACC	Accepted job submission
QUE	Waiting for job execution
RNA	Acquiring resources required job execution
RUN	Executing job
RNO	Waiting for completion of job termination processing
EXT	Exited job end execution
CCL	Exited job execution by interruption
ERR	In fixed state due to an error
RJT	Rejected job submission

5.7.2 Detailed Job Status (-v option)

The -v option displays detailed job information.

```
[username@hokusai1 ~]$ pjstat -v


  ACCEPT QUEUED  STGIN  READY RUNING RUNOUT STGOUT   HOLD  ERROR   TOTAL
       0      0      0      0      1      0      0      0      0       1
s      0      0      0      0      1      0      0      0      0       1


JOB_ID     JOB_NAME   MD ST  USER     GROUP    START_DATE      ELAPSE_TIM ELAPSE_LIM NODE_REQUIRE
VNODE  CORE V_MEM        V_POL E_POL RANK      LST EC  PC  SN PRI ACCEPT         RSC_UNIT REASON
10056876   STDIN      NM RUN username projectID   03/19 10:25:20  0000:00:05 0002:00:00 -
1      4    35200 MiB    A_UPK SHARE -         RNP 0   0   0  127 03/19 10:25:19 bwmpc    -
```

Table 5-22 Job detailed information (Additional field in -v option)

Field	Description
GROUP	ProjectID
ELAPSE_TIM	Elapsed time limit
V_POL	Arrangement policy of virtual node
E_POL	Execution mode policy
RANK	The allocation rule of the rank
LST	Last processing state of the job
EC	Job script exit code
PC	PJM code
SN	Signal number
PRI	Job priority (0: low <-> 255: high)
ACCEPT	Job submission date
RSC_UNIT	Resource unit
REASON	Error message

5.7.3 Ended Job Status (-H option)

The -H option displays ended job information in addition to submitted jobs.

```
[username@hokusai1 ~]$ pjstat -H

  ACCEPT QUEUED  STGIN  READY RUNING RUNOUT STGOUT   HOLD  ERROR   TOTAL
       0      0      0      0      0      0      0      0      0       0
s      0      0      0      0      0      0      0      0      0       0

  REJECT   EXIT CANCEL   TOTAL
       0      0      0       0
s      0    180      0     180

JOB_ID     JOB_NAME   MD ST  USER     START_DATE      ELAPSE_LIM NODE_REQUIRE    VNODE  CORE V_MEM
2135       run.sh     NM EXT username 02/08 09:58:02  0012:00:00 -               1      60   16384
MiB
```

Table 5-23 Ended job status

Field	Description
JOB_ID	Job ID For sub-jobs, Subjob ID
JOB_NAME	Job name
MD	Job model (NM: Normal job, ST: Step job, BU: Bulk job)
ST	Job state (See Table 5-21 Job state)
USER	User name who executed the job
START_DATE	Start time
ELAPSE_LIM	Elapsed time limit "hhhh:mm:ss"
NODE_REQUIRE	"-" is output.
VNODE	Number of nodes
CORE	Number of cores per node
V_MEM	Amount of memory per node

5.7.4 Resource Unit and Resource Group Status (--rsc option)

The --rsc option displays resource groups available for the user.

```
[username@hokusai1 ~]$ pjstat  --rsc
RSCUNIT                  RSCUNIT_SIZE  RSCGRP                          RSCGRP_SIZE
--------------------------------------------------------------------------------
bwmpc  [ENABLE,START]    840           batch         [ENABLE,START]    780
                                       gaussian      [ENABLE,START]     56
                                       qchem         [ENABLE,START]     56
                                       interact      [ENABLE,START]      4
--------------------------------------------------------------------------------
gwacsl [ENABLE,START]    2             batch         [ENABLE,START]      2
                                       gaussian      [ENABLE,START]      2
                                       interact      [ENABLE,START]      2
--------------------------------------------------------------------------------
* [ENABLE/DISABLE]: New jobs can be submitted or not.
  [START/STOP]    : QUE jobs can be started or not.
```

Table 5-24 Resource unit and resource group information

Field	Description
RSCUNIT	Resource unit name and its status. Displayed statuses are the following. ENABLE : Jobs can be submitted DISABLE : Jobs cannot be submitted START : Jobs can be executed STOP : Jobs cannot be executed
RSCUNIT_SIZE	Size of resource unit. The number of nodes N which makes up the resource unit is displayed.
RSCGRP	Resource group name and its status
RSCGRP_SIZE	Size of resource group The number of nodes N that makes up the resource unit is displayed.

5.7.5 Status of Node and Core Usage (-un option and uc option)

The -un option displays the status of node usage HOKUSAI BigWaterfall system.

```
[username@hokusai1 ~]$ pjstat -un
The status of node usage                                       Ratio Used/Total
--------------------------------------------------------------------------------
bwmpc    *************************************************-    99.6%( 837/ 840)
gwacsl   **************************************************   100.0%(   2/   2)
```

Table 5-25 Status of node usage

Field	Description
Ratio	Used ratio
Used	Used number of nodes
Total	Total number of nodes

The -uc option displays the status of core usage HOKUSAI BigWaterfall system.

```
[username@hokusai1 ~]$ pjstat -uc
The status of core usage                                     Ratio  Used/Total
--------------------------------------------------------------------------------
bwmpc    *************************************************-  99.5%(33444/33600)
gwacsl   ************************************************** 100.0%(  120/  120)
```

Table 5-26 Status of core usage

Field	Description
Ratio	Used ratio
Used	Used number of cores
Total	Total number of cores

⚠ **The jobs are scheduled by priority order of projects. When the jobs wait whose priority order is higher than your priority order, your jobs are not executed if the unused nodes or cores exist.**

5.7.6 Priority Order of Projects (-p option)

The -p option displays the priority order of projects per resource unit.

```
[username@hokusai1 ~]$ pjstat -p
Project priority in fair-share function
[Q99999]
 +- bwmpc :   2nd
 +- gwacsl:   3rd
```

5.7.7 Resource limit of job submission (-x option)

The -x option displays the upper limit of number of cores, nodes and elapse time of each resource group.

The following example indicates that you can submit up to 72 hours job with less than or equal 640 cores (16 nodes) and up to 24 hours job with less than or equal 1,280 cores (32 nodes) to the batch resource group of gwmpc.

```
[username@hokusai1 ~]$ pjstat -x
Limits on resources

PROJECT  RSCUNIT  RSCGRP         CORE (NODE)  ELAPSE
=========================================================
Q99999   bwmpc    batch             640 (  16)    72:00:00
                                   1280 (  32)    24:00:00
                  gaussian           40 (   1)    72:00:00
                  qchem             640 (  16)    72:00:00
                  interact           80 (   2)     2:00:00
         ------------------------------------------------
```

5.8 **Cancel jobs**

Use the *pjdel* command to cancel submitted jobs.

```
pjdel JOBID [JOBID…]
```

Specify job IDs to cancel to the argument on the *pjdel* command line.

```
[username@hokusai1 ~]$ pjdel 12345
[INFO] PJM 0100 pjdel Job 12345 canceled.
```

5.9 **Display a job script**

Use the *pjcat* command to display the job script.

```
pjcat -s JOBID
```

Specify a job ID to display to the argument on the *pjcat* command line.

```
[username@hokusai1 ~]$ pjcat -s 12345
#!/bin/sh

#PJM -L rscunit=bwmpc
#PJM -L rscgrp=batch
#PJM -L vnode=(core=1)

./a.out
```

5.10 **Environment Variable**

5.10.1 Environment Variables for Thread Parallelization and MPI Execution

This section explains main environment variables specified to execute thread parallelized programs or MPI programs.

Table 5-27 Runtime environment variables

Environment Variable	Description
OMP_NUM_THREADS	When executing a multi-threaded program by OpenMP or auto parallelization, set the number of threads to the OMP_NUM_THREADS environment variable. If this variable is not specified, the number of cores available for the job is set.
KMP_AFFINITY	Control to bind threads to physical processors. Default of HOKUSAI BigWaterfall is "compact"
I_MPI_PIN_DOMAIN	Control to bind MPI processes to physical processors.

5.10.2 Environment Variables for Jobs

The job management system configures the following environment variables for jobs.

Table 5-28 Available environment variables in jobs

Envirnment variable	Description
PJM_ENVIRONMENT	"BATCH" for a batch job; ”INTERACT” for an interactive job
PJM_JOBID	Job ID
PJM_JOBNAME	Job name
PJM_O_WORKDIR	The current directory at the *pjsub* command execution
PJM_COMMENT	The string set to the --comment option on the *pjsub* command line
PJM_MAILSENDTO	The mail destination user set to the --mail-list option on the *pjsub* command line
PJM_STEPNUM	Step number (set only for step jobs)
PJM_BULKNUM	Bulk number(set only for bulk jobs)
PJM_SUBJOBID	Sub-job ID (set only for step jobs)

In addition, some resource options specified at job submitting are set as the following environment variables.

Table 5-29 Available environment variables in jobs

Envirnment variable	Description
PJM_RSCUNIT	Name of resource unit (-L rscunit)
PJM_RSCGRP	Name of resource group (-L rscgrp)
PJM_NODE	Number of nodes value of "-L vnode"
PJM_NODE_CORE	Cores per node value of "-L vnode-core"
PJM_TOTAL_CORE	Total number of cores (PJM_NODE * PJM_NODE_CORE)
PJM_NODE_MEM	Amount of memory per node value of "-L vnode-mem"
PJM_NODE_MEM_BYTE	PJM_NODE_MEM in the byte unit
PJM_CORE_MEM	Amount of memory per core value of "-L core-mem"
PJM_CORE_MEM_BYTE	PJM_CORE_MEM in the byte unit
PJM_ELAPSE	Elapse limit (value of "-L elapse")
PJM_ELAPSE_SEC	PJM_ELAPSE in the second unit

You should use the -X option on the *pjsub* command line to pass environment variables set before job submission on a login node to a job. Otherwise, no environment variables are passed.

6. Development Tools

6.1 Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory

The following tools for the Massively Parallel Computer (BWMPC) / Application Computing Server with Large Memory are available.

- Intel VTune Amplifier XE (Performance profiler)
- Intel Inspector XE (Memory and Thread Debugger)
- Intel Advisor XE (Thread design and prototype)
- Intel Trace Analyzer & Collector (MPI Communications Profiling and Analysis)

7. User Portal

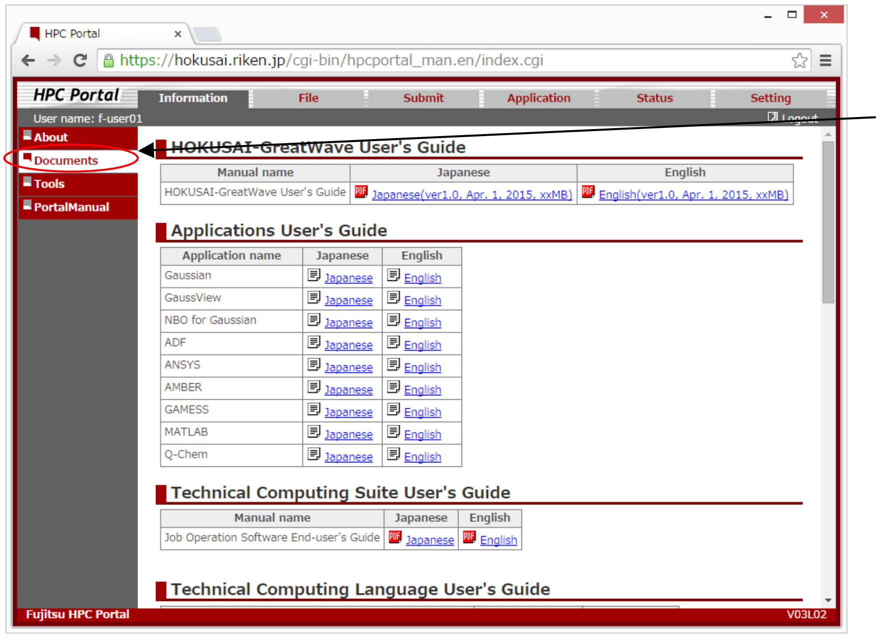
To use the User Portal, launch the browser and access to the following URL.

https://hokusai.riken.jp

On User Portal, you can know how to execute the softwares available on the HOKUSAI BigWaterfall system, the versions of those softwares, and you can registrate ssh public key.

7.1 Manuals

All reference manuals can be downloaded from the User Portal.



HOKUSAI-GreatWave User's Guide

Manual name	Japanese	English
HOKUSAI-GreatWave User's Guide	Japanese(ver1.0, Apr. 1, 2015, xxMB)	English(ver1.0, Apr. 1, 2015, xxMB)

Applications User's Guide

Application name	Japanese	English
Gaussian	Japanese	English
GaussView	Japanese	English
NBO for Gaussian	Japanese	English
ADF	Japanese	English
ANSYS	Japanese	English
AMBER	Japanese	English
GAMESS	Japanese	English
MATLAB	Japanese	English
Q-Chem	Japanese	English

Technical Computing Suite User's Guide

Manual name	Japanese	English
Job Operation Software End-user's Guide	Japanese	English

Technical Computing Language User's Guide

Click [Documents]

Figure 7-1 Screen of Documents