AI Bridging Cloud Infrastructure “ABCI” Tutorial (Submit Jobs)

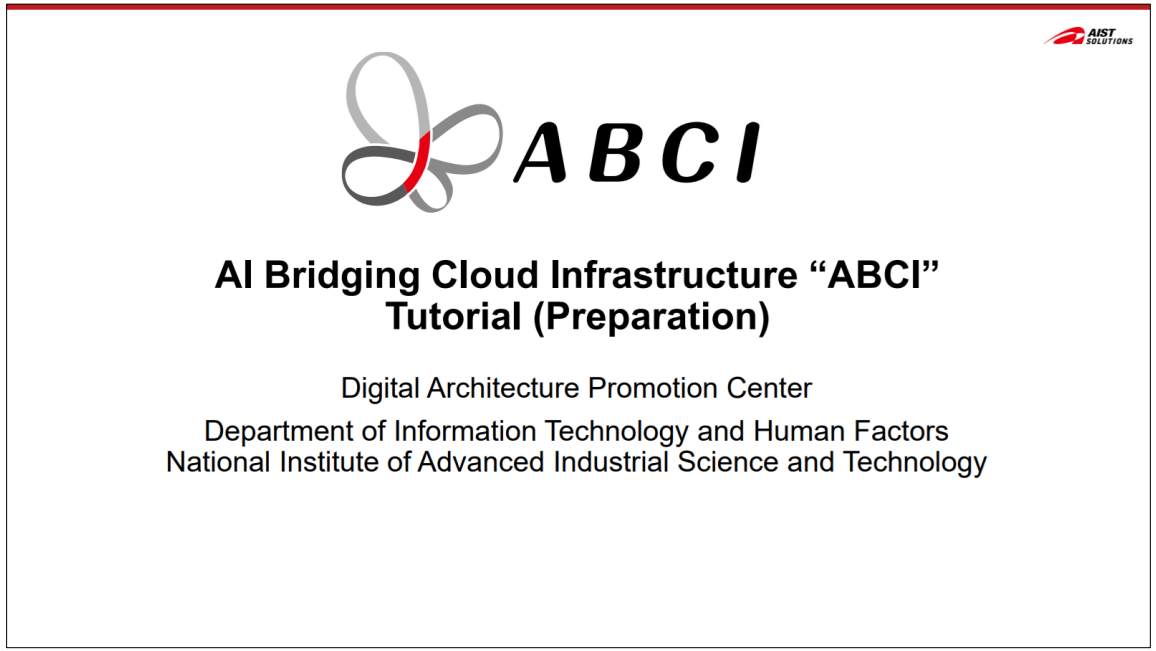
Digital Architecture Promotion Center

Department of Information Technology and Human Factors
National Institute of Advanced Industrial Science and Technology

CONTENTS

Before Using the ABCI System

**To use the ABCI system,
you must be able to use Linux commands on your terminal.**



The “Preparation” section, which shows a step-by-step guide to connect to ABCI, is available on the ABCI Portal.

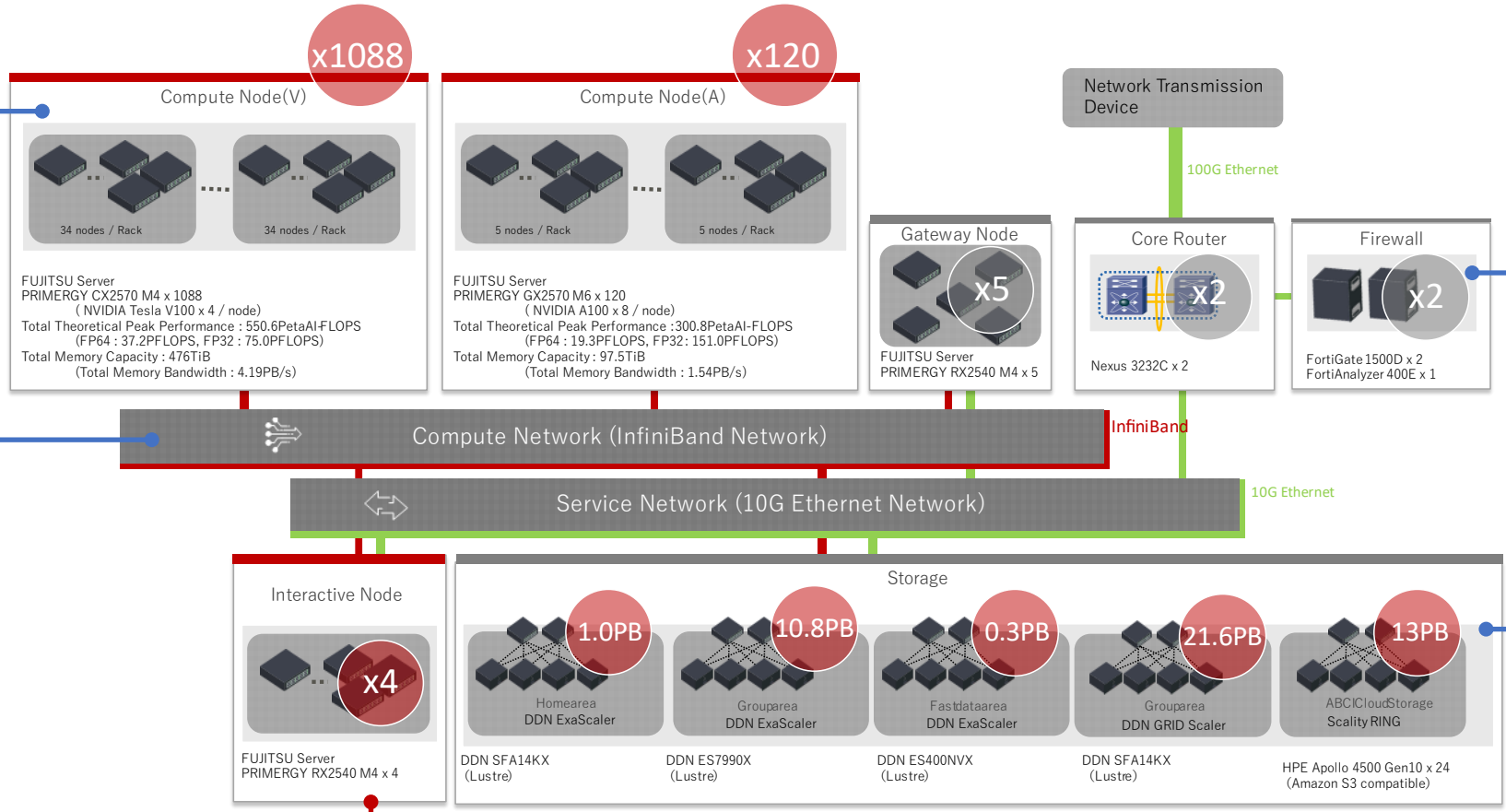
If you have not connected to ABCI yet, please refer to this document.

Reference: ABCI Tutorial (Preparation)

ABCI 2.0 System Overview

ABCI 2.0 system combines computing resources and large-capacity storage with high-speed networking

① **Computing Resource**
Compute Node (V) ×1088
Compute Node (A) ×120



③ **InfiniBand**
High-Speed Network

④ **Firewall, etc.**

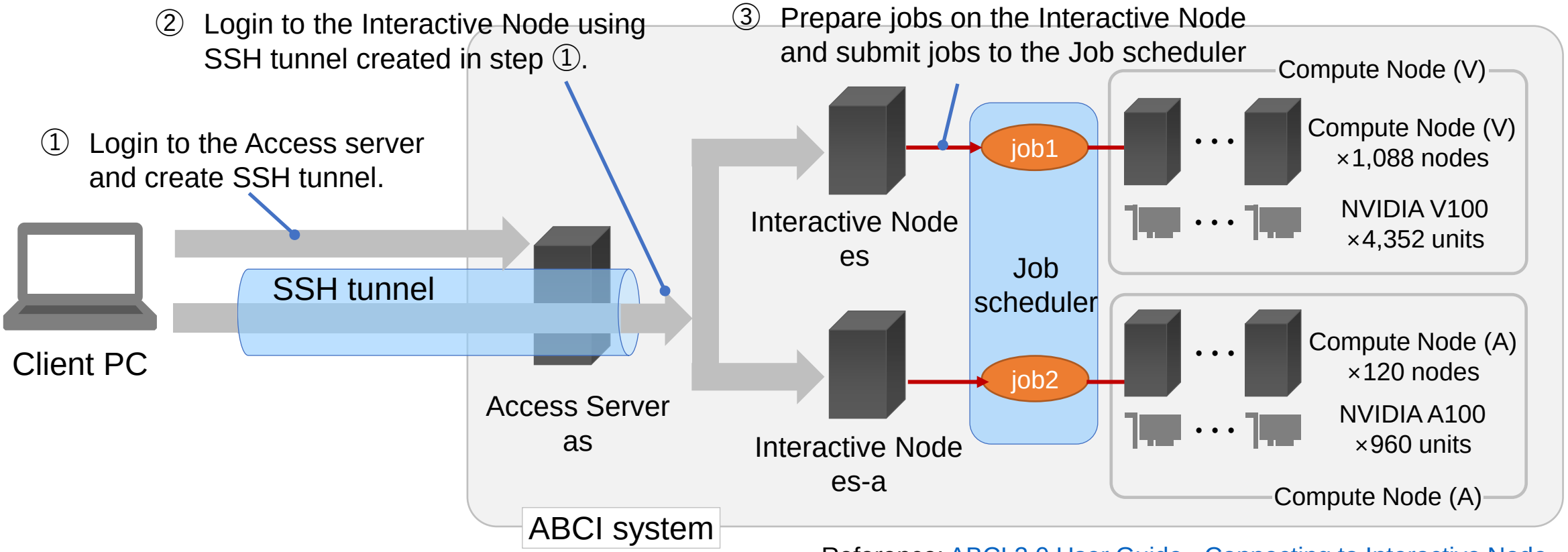
② **Storage System**
Total 47PB
Shared File System
ABCI Cloud Storage

Interactive Node

Reference: [ABCI 2.0 User Guide – ABCI System Overview]

ABCI System Usage Flow

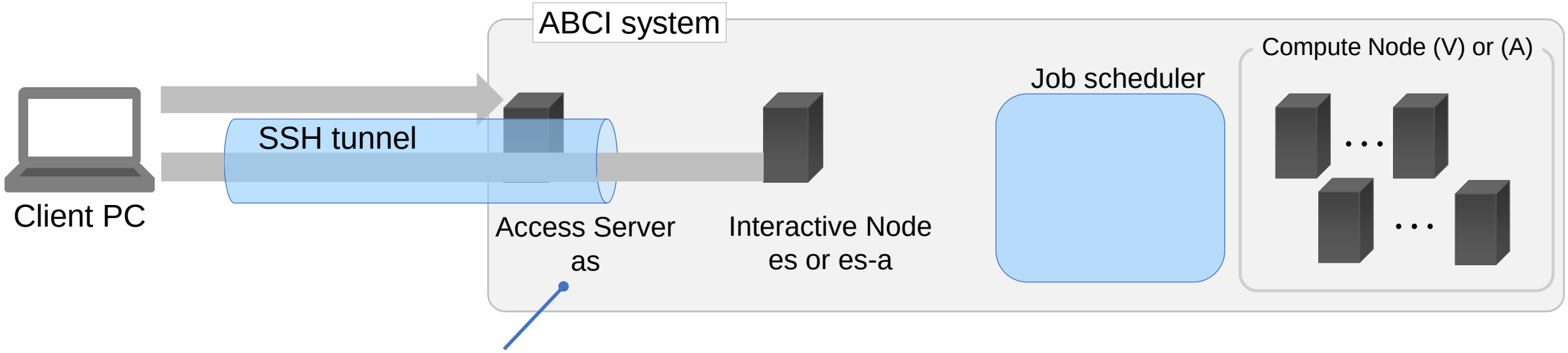
Access to ABCI system, establishment of ssh connection, job preparation on the Interactive Nodes, job execution submitted to the Compute Nodes from the job scheduler



Reference: [ABCI 2.0 User Guide - Connecting to Interactive Node]

Role of each node in ABCI system (1/3)

First, log in to the Access Server (as) and create an SSH tunnel

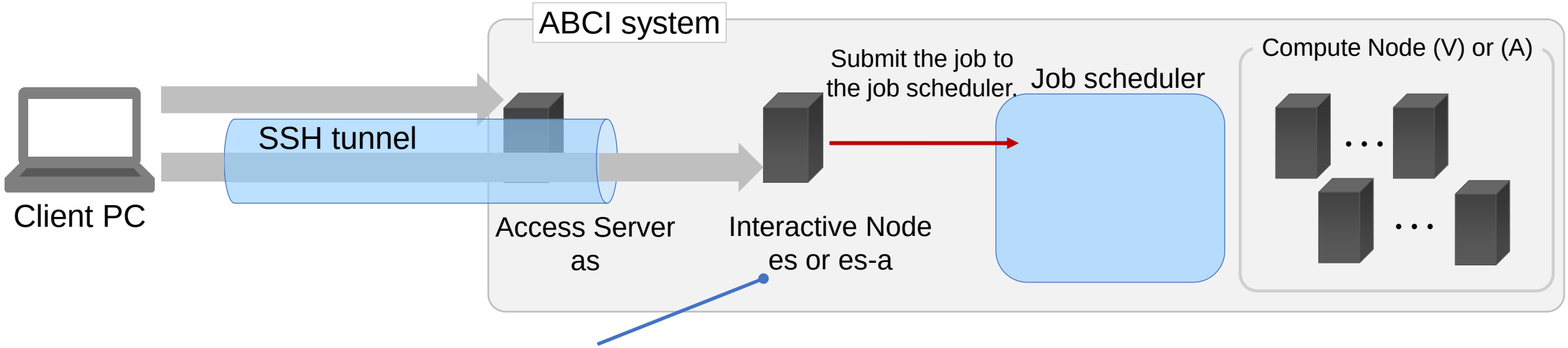


Access Server

- The server to log in to access an Interactive Node.
  Bastian server.
- Create an SSH tunnel to access an Interactive Node.
- Note that any operations at the terminal window will cause disconnection of the SSH session.

Role of each node in ABCI system (2/3)

Next, log in to the Interactive Node and prepare to run the job.



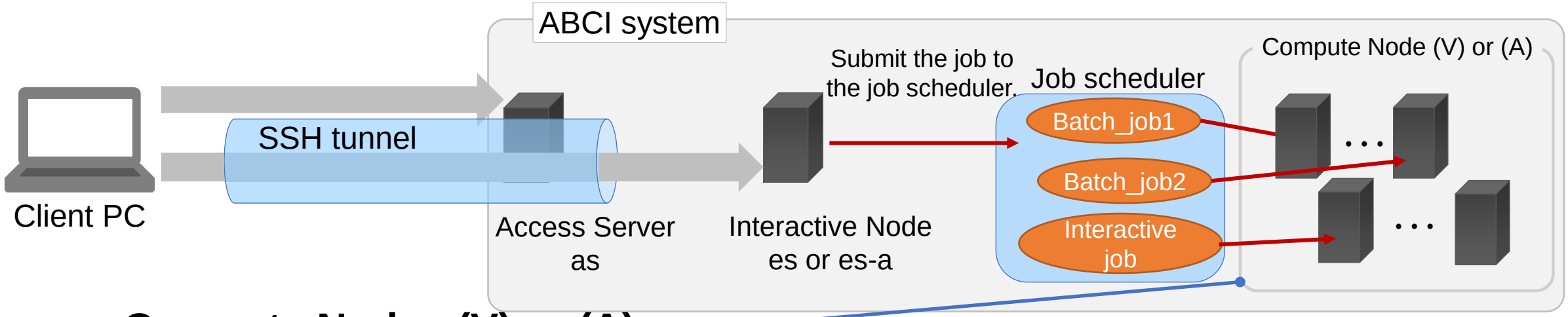
Interactive Nodes

Servers where users log in and perform tasks such as job execution on the ABCI

- Compile, deploy application program, and prepare a job execution script
- Request job execution to the job scheduler
- Upload and download data to ABCI

Role of each node in ABCI system (3/3)

Then, register the job in the job scheduler and execute the job on the allocated Compute Node.



Compute Nodes (V) or (A)

Servers that execute users' jobs (programs)

- Execute jobs received from the job scheduler
- Job execution method:

① Spot service (batch job execution)

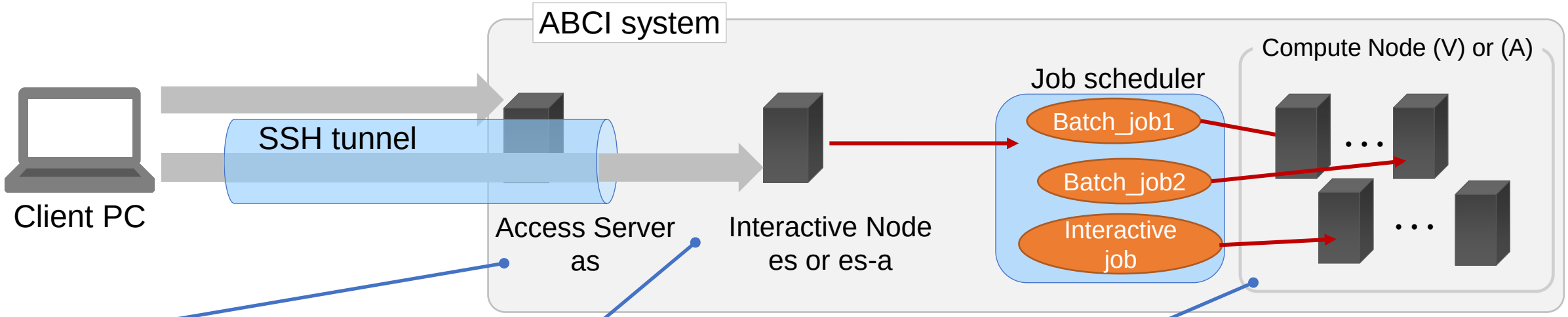
Requests the job scheduler to execute a job, executes the job on the allocated Compute Nodes.

② On-demand service (interactive job execution)

A user logs-in directly to a Compute Node allocated by the job scheduler and executes the program.

Role of each node in ABCI system (summary)

To execute a job in an ABCI system, the job is handed over to a Compute Node via multiple nodes.



Access Server

- Bastian server to log in to access an Interactive Node.
- Note that any operation at the terminal window will cause disconnection of the SSH session.

Interactive Nodes

ABCI server where users login and perform tasks such as job execution.

- Compile, deploy application program, and prepare a job execution script.
- Request job execution to job scheduler.
- Upload and download data.

Compute Nodes

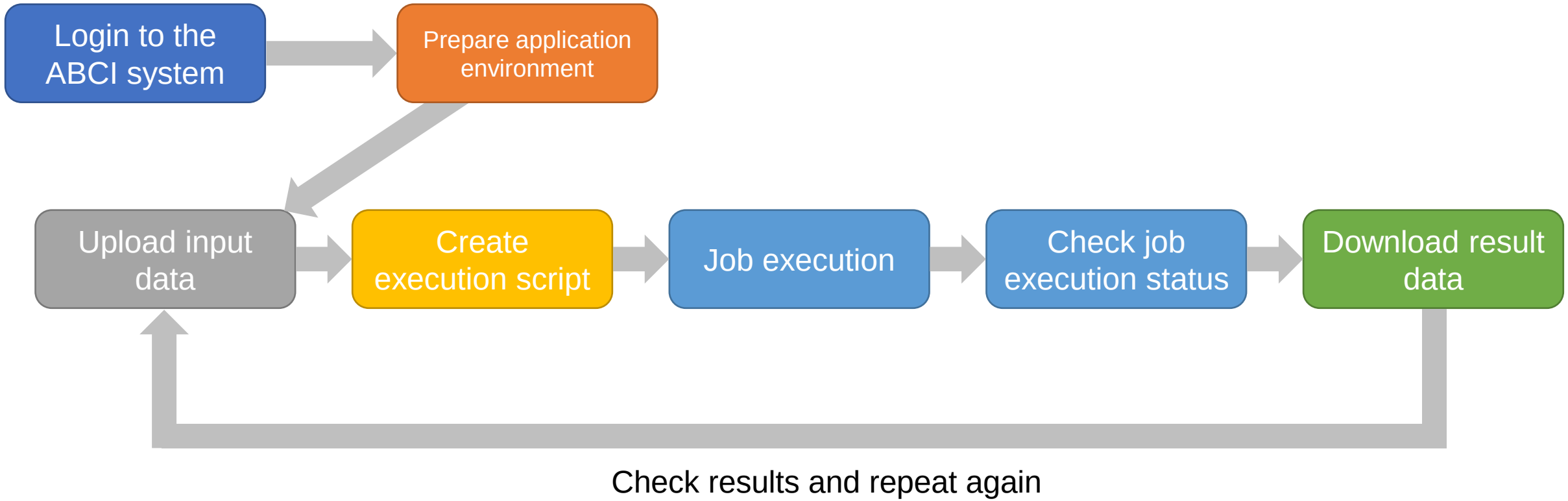
A cluster of servers that execute users' jobs (scripts)

- Execute jobs received from the job scheduler.
- Job execution method:
  ① Spot service (batch job execution)
  ② On-demand service (interactive job execution)

Reference: ABCI 2.0 User Guide - Computing Resources

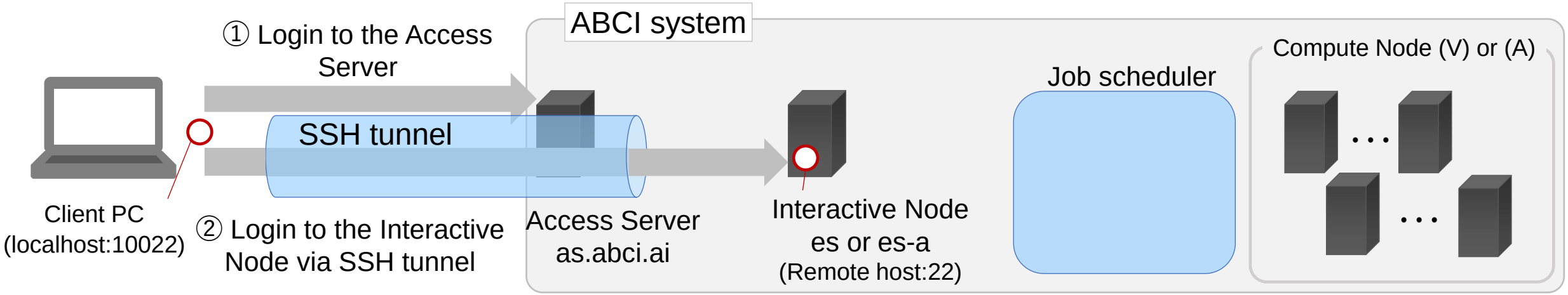
ABCI system Usage Flow

Repeat data upload and job execution while checking job execution results.



Login to the ABCI system
– Using Terminal –

① From a terminal, login to the Access Server (as) via SSH and create an SSH tunnel.
② Open another terminal and login to the Interactive Node (es)



① Login to the Access Server – Command syntax:

```
ssh -L 10022:es:22 -l aaa12345xx as.abci.ai
```

--L {port number to receive on Client PC}:{interactive node name to connect to}:{port number of interactive node} -l {ABCI account name} {Access Server name}

② Login to the Interactive Node – Command syntax

```
ssh -p 10022 -l aaa12345xx localhost
```

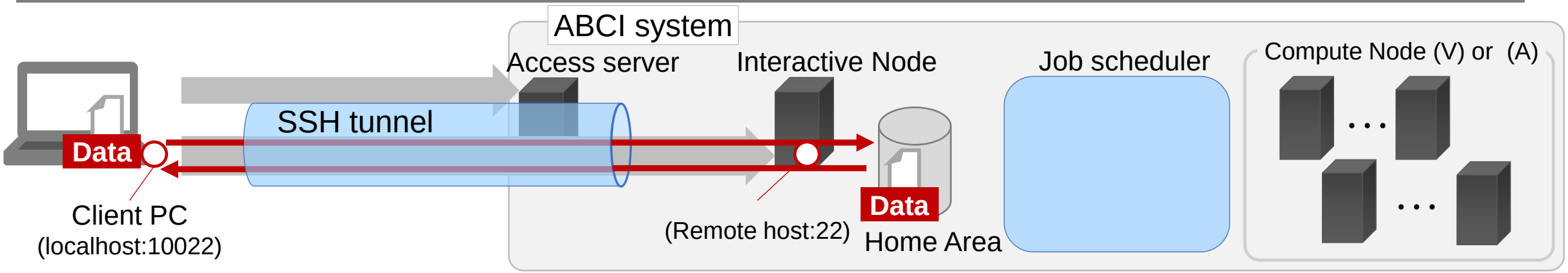
--p {port number to receive on client PC} -l {ABCI account name to login} {host name}

Reference: ABCI 2.0 User Guide – Login using an SSH Client

Upload / Download Data

− Using scp command from terminal (1/4) −

Data can be uploaded/downloaded via SSH tunnel



① Uploading data from a client PC

Command syntax

```
$ scp -P 10022 {local filename} aaa12345xx@localhost:./###/{path name of destination}
```

-P {port number of client PC} {source file} {ABCI account}@localhost:{path name of destination directory}

② Downloading data to a client PC

Command syntax

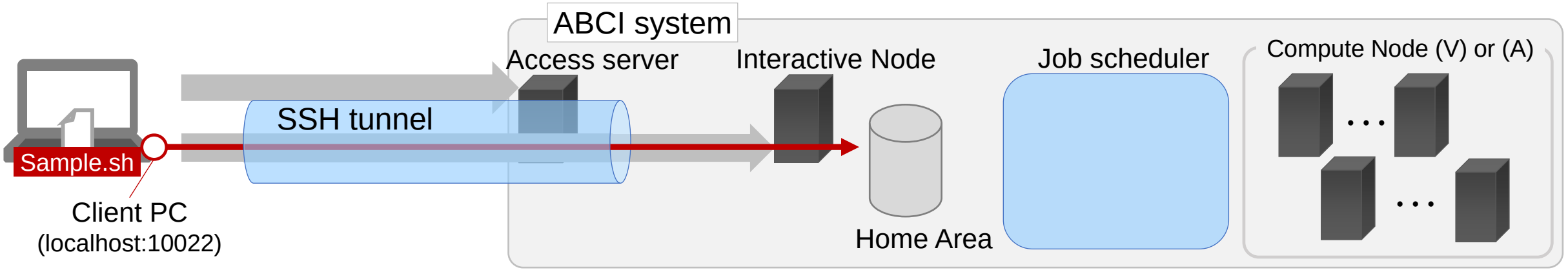
```
$ scp -P 10022 aaa12345xx@localhost:{Remote filename} ./
```

-P {port number of client PC} {ABCI account name}@localhost:{remote files} {path name of destination directory}

Reference: ABCI 2.0 User Guide − File Transfer to Interactive Node

Upload Data

– Using scp command from terminal (2/4) –



Command syntax Interactive Node side

```
[aaa12345xx@es1 ~]$ ls
```

Show file list in your Home directory, but empty.

Client PC side

```
yourpc$ scp -P 10022 Sample.sh.txt aaa12345xx@localhost:./
Enter passphrase for key '/home/username/.ssh/id_rsa':
Sample.sh.txt        100%  ##(file size)  #.#KB/s(transfer-rate)  ##:##(transfer time)
```

Transfer "sample.sh" in the current directory of the client PC to your own Home directory on ABCI.

After entering the passphrase of the key you set, the specified file is transferred.

Interactive Node side

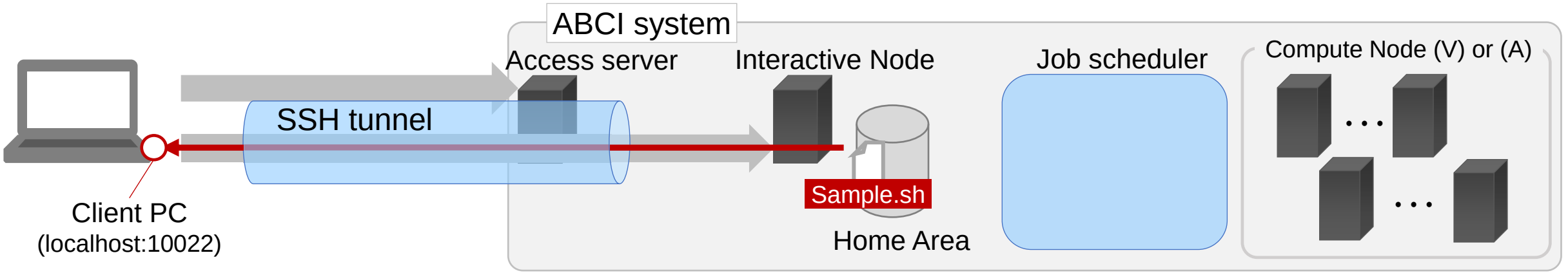
```
[aaa12345xx@es1 ~]$ ls
Sample.sh.txt
```

After file transfer using scp command, show the file list in the Home directory, and the sent file has been stored.

Reference: ABCI 2.0 User Guide – File Transfer to Interactive Node

Download Data

− Using scp command from terminal (3/4) −



Command syntax

Client PC side

```
yourpc$ scp -P 10022 aaa12345xx@localhost:./sample.sh ./sample.sh
Enter passphrase for key '/home/username/.ssh/id_rsa':
sample.sh       100%  ##(file size)  #.#KB/s(transfer-rate)  ##:##(transfer time)
yourpc$ ls
sample.sh
```

Transfer "sample.sh" in the Home directory of ABCI to the current directory on the client PC.

After entering the passphrase of the key you set, the specified file is transferred.

After file transfer using scp command, show the file list in the current directory, and the received file has been stored.

Reference: ABCI 2.0 User Guide − File Transfer to Interactive Node

Encoding Conversion

– Using scp command from terminal (4/4) –

Newline codes are different for each OS

- Linux and other Unix systems: LF
- Windows: CRLF
- MacOS(9 or earlier): CR
- MacOS(X or later): LF

For data containing Japanese characters, the Kanji code differs for each OS, application, and its version.

- Linux: UTF-8
- Major Unix systems: EUC
- MS-DOS, Windows 3.1 or earlier, Windows 9X, Me or earlier: Shift-JIS
- Windows NT, 2000, XP or later: Basically UTF-8, but each application has different support.
- MacOS(9 or earlier): Shift-JIS
- MacOS(X or later): Basically UTF-8, but each application has different support.

Command syntax

On Interactive Node

```
[aaa12345xx@es1 ~]$ nkf --guess *
Sample.sh.txt: UTF-8 (CRLF)
[aaa12345xx@es1 ~]$ nkf -Lu Sample.sh.txt >Sample.sh
[aaa12345xx@es1 ~]$ nkf --guess *
Sample.sh.txt: UTF-8 (CRLF)
Sample.sh: UTF-8 (LF)
```

Check the encoding parameters of the file.

If the newline code is CRLF (Windows) or CR (old Mac), convert the newline code to LF (for Linux) and save the file with Sample.sh

On Interactive Node

```
[aaa12345xx@es1 ~]$ nkf --guess Sample_SJIS.sh.txt
Sample_SJIS.sh.txt: Shift_JIS (CRLF)
[aaa12345xx@es1 ~]$ nkf -wd SJIS.sh.txt >SJIS.sh
[aaa12345xx@es1 ~]$ nkf --guess SJIS*
SJIS.sh: UTF-8 (LF)
SJIS.sh.txt: Shift_JIS (CRLF)
```

If the character code is Shift-JIS (old Windows), it can be converted to UTF-8. At the same time, the newline code is converted to LF.

Upload/Download Data

－ Using WinSCP (1/9) －

① Download and install WinSCP

【for Windows users only】

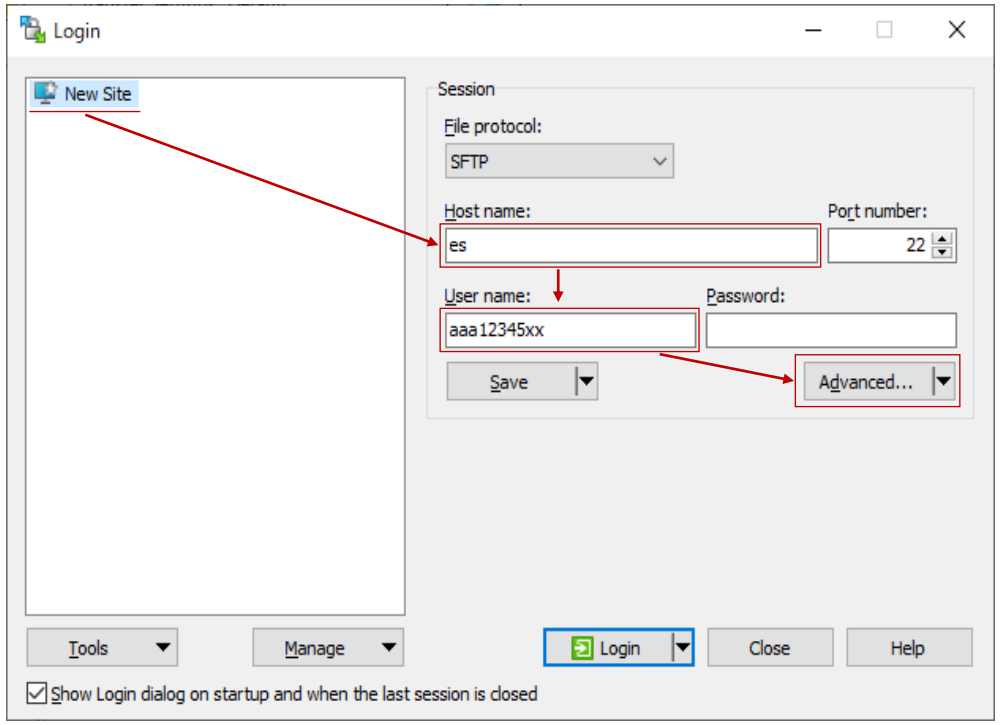
- WinSCP download URL
  https://winscp.net/eng/download.php
- Install according to the guide.

- This chapter uses WinSCP's "Explorer" mode, in which only the remote side's directory is displayed and transferred in combination with Windows Explorer.
- If you have already configured your ABCI login settings in PuTTY, it will import them.

Upload/Download Data

– Using WinSCP (2/9) –

② Enter the information for connecting to an Interactive Node

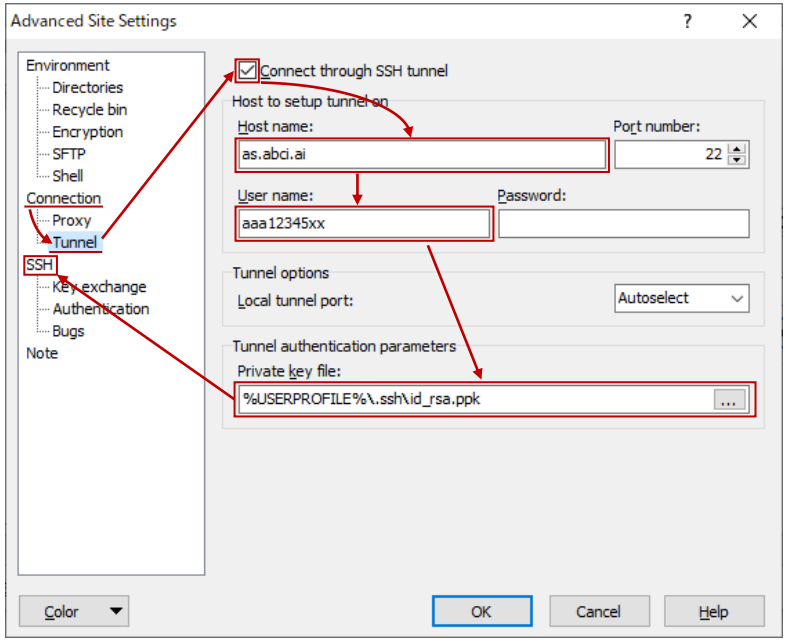


- When you start WinSCP, the [Login] dialog box will appear.
  If the [Login] dialog box does not appear, select [Tabs] → [Sites] → [Site Manager...] from the menu bar of the WinSCP window.
- From [New Site], fill in the following fields:
  1. Host name: es or es-a
  2. User name: ABCI account name
- Click the [Advanced...] button.

Upload/Download Data

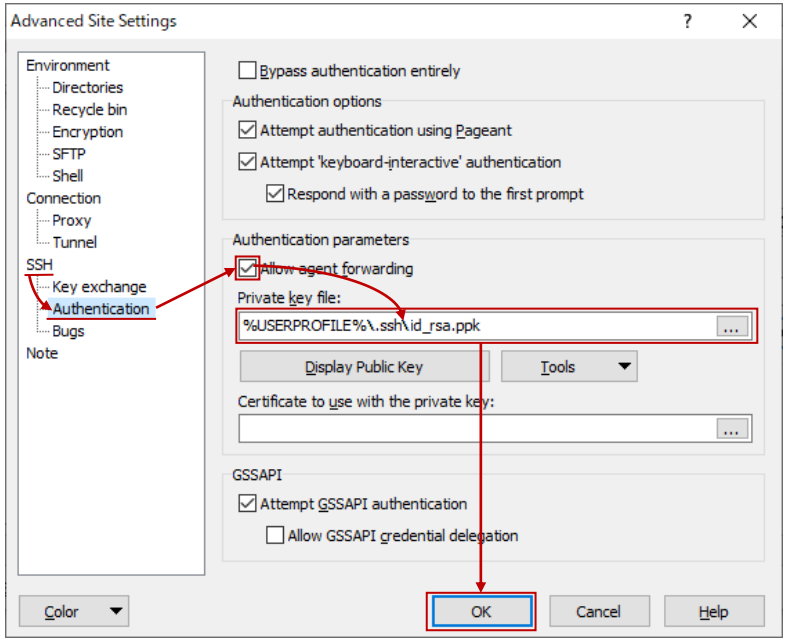
– Using WinSCP (3/9) –

③ Register your private key information



From the left pane of the [Advanced Site Settings] dialog box, select [Connection] → [Tunnel].

- Check [Connect through SSH tunnel] , fill in the following fields:
  1. Host name: as.abci.ai
  2. User name: ABCI account name
- Click [...] at the right of the [Private key file:] field and choose the private key file or type the path directly.



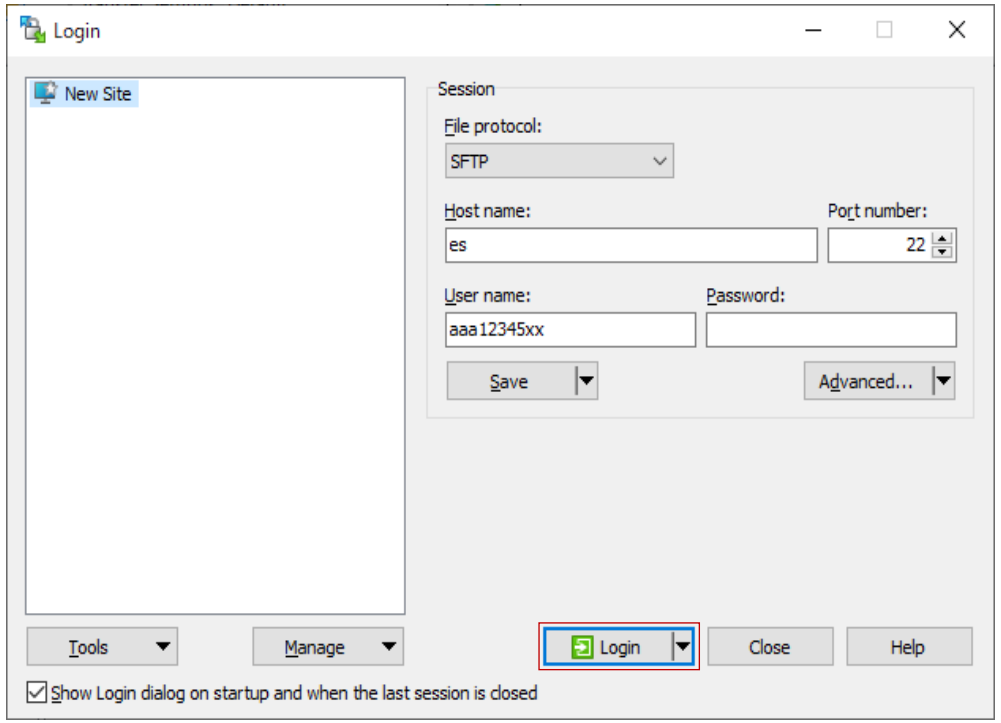
Select [SSH] → [Authentication] from the left pane.

- In the [Authentication parameters] section, check [Allow agent forwarding] and choose the private key as in the previous section (left side of this page).
- Leave the other items in their default state and click the [OK] button to close the [Advanced Site Settings] dialog box.

Upload/Download Data

– Using WinSCP (4/9) –

④ Let’s login to the host

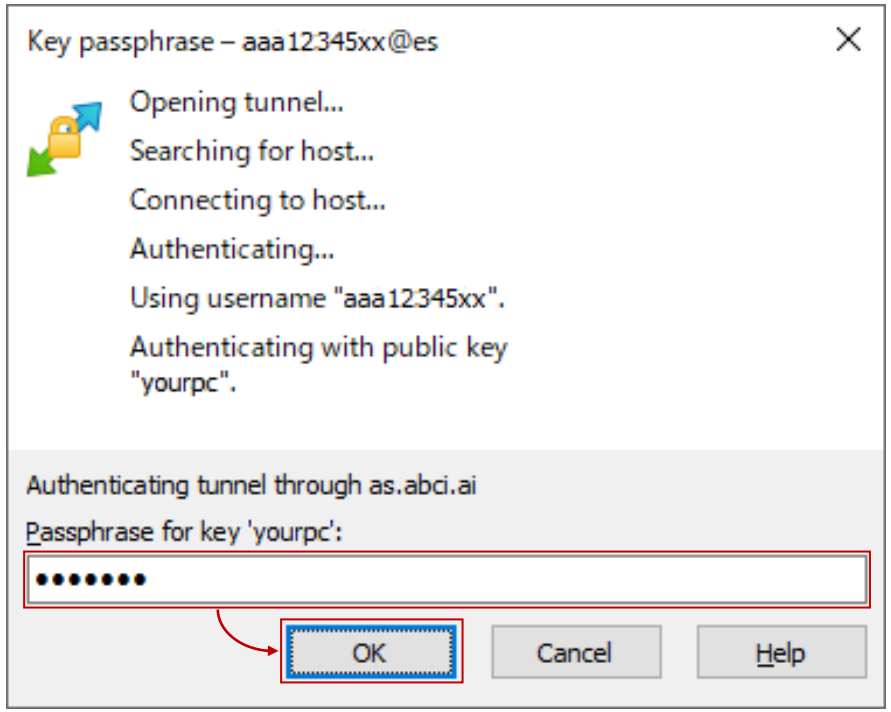


- Click [Login] button to connect to the host.

Upload/Download Data

– Using WinSCP (5/9) –

Enter the Passphrase for your Private Key

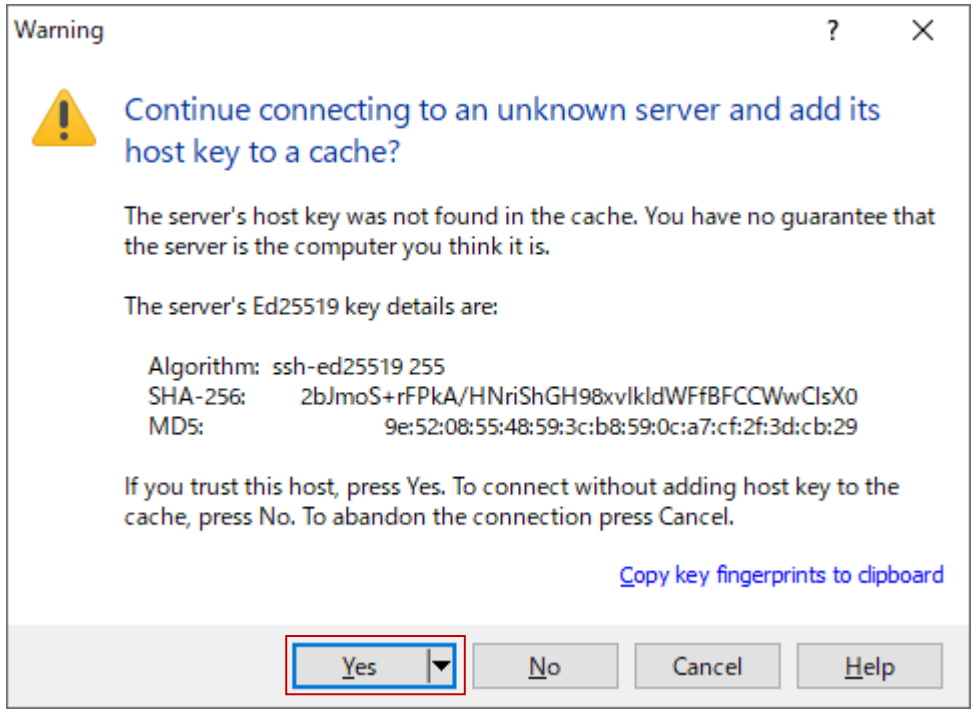


- After starting the connection, you will be asked several times for your passphrase, enter it and click [OK].

Upload/Download Data

– Using WinSCP (6/9) –

When "Warning" is appeared



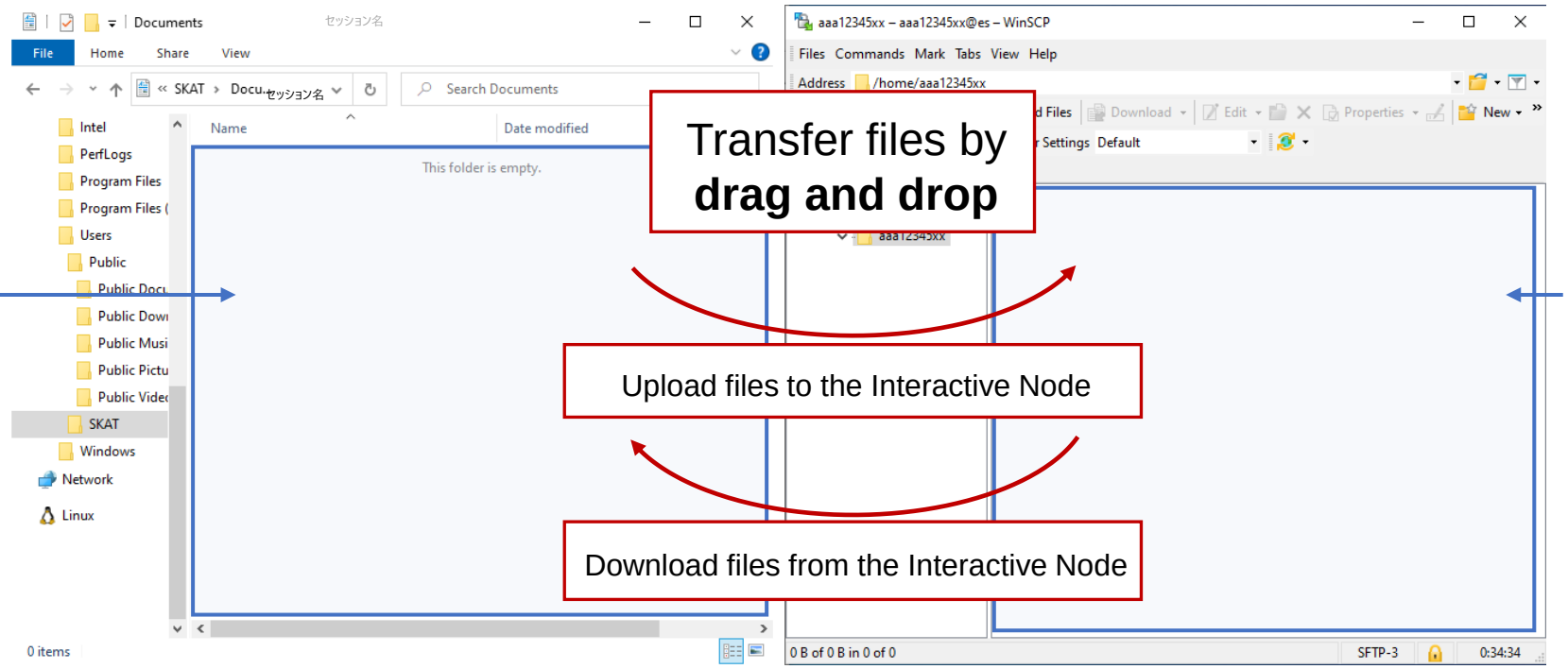
- When connecting to ABCI with WinSCP for the first time, this message will appear several times.
- Click the [Yes] button. (Depending on your environment, click [Update].)

Upload/Download Data

– Using WinSCP (7/9) –

⑤ Upload and download files by simply dragging and dropping files

A window of [File Explorer] shows the folder on your PC



Transfer files by **drag and drop**

Upload files to the Interactive Node

Download files from the Interactive Node

A window of [WinSCP] shows the directory on the Interactive Node

- When you log in to the Interactive Node, the directory on the remote node to which you are connected is displayed.
- Open [File Explorer] in another window and place them side by side.
  Left side: Folders on the PC Right side: Directories on the Interactive node
- Upload and download files by simply dragging and dropping them.

Upload Data

− Using WinSCP (8/9) −

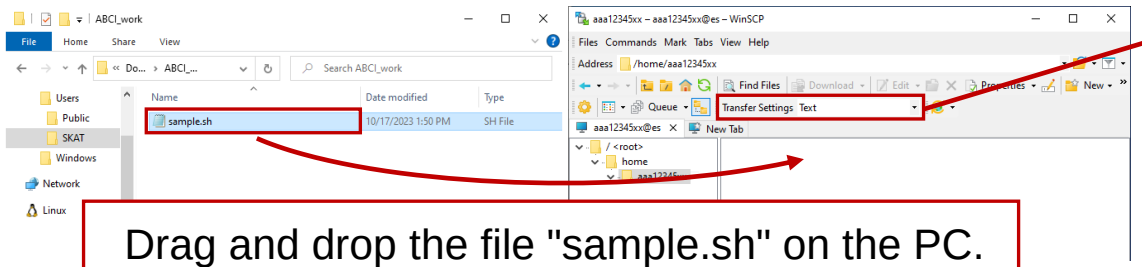
⑥ Upload files by simply dragging and dropping files

- Before uploading files

```
[aaa12345xx@es2 ~]$ ls
[aaa12345xx@es2 ~]$
```

There are no files

- The logged-in Interactive Node has no files.



Drag and drop the file "sample.sh" on the PC.

- Set [Transfer Settings] to "Text".
- Upload files by dragging and dropping files on the PC in the [File Explorer] into the frame displaying the directory on the Interactive Node in [WinSCP].

- After uploading files

```
[aaa12345xx@es2 ~]$ ls
[aaa12345xx@es2 ~]$ ls
sample.sh
[aaa12345xx@es2 ~]$ nkf --guess *
sample.sh: UTF-8 (LF)
[aaa12345xx@es2 ~]$
```

Transferred file exists

Newline code is already converted to LF.

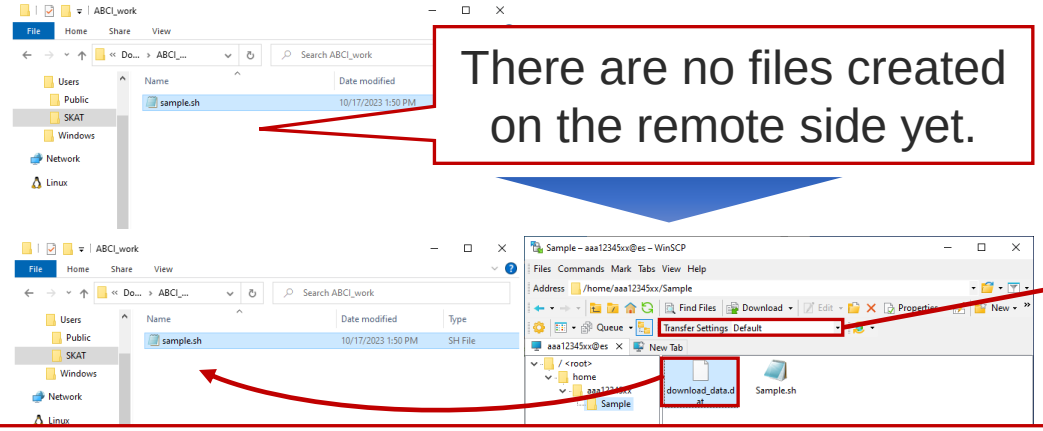
- Where there has been no files, there is a file "sample.sh" transferred by [WinSCP].
- Since [Transfer Settings] is set to "Text", the newline code is automatically converted to "LF".

Download Data

– Using WinSCP (9/9) –

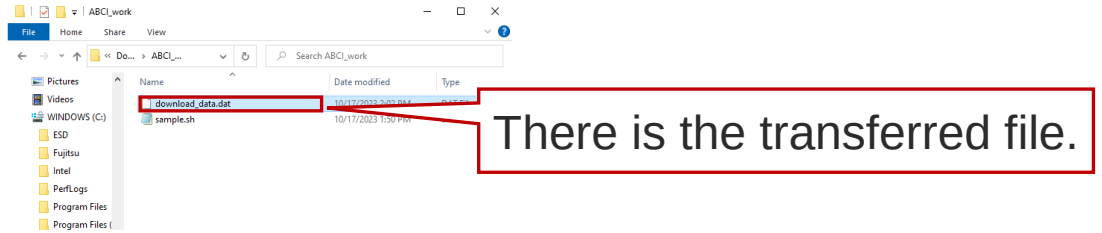
⑦ Download files by simply dragging and dropping files

- Before downloading files



- The file you want to download is not in the local folder.
- Set [Transfer Settings] to "Default".
- **Drag and drop** the file on the directory of the Interactive Node (right figure) into the folder of the PC displayed in File Explorer (left figure) to download the file.

- After downloading files



- The file "download_data.dat" transferred by WinSCP is in the folder of the local terminal.

Job Execution Services

To execute a job, choose the appropriate service for the processing method from the three types of services

Types of Services for Job Execution

① Spot Service: Create a job script and request batch processing from the job scheduler.

② On-demand Service: Request the job scheduler to reserve Compute Nodes and execute programs on them.

③ Reserved Service: Requests the job scheduler to reserve Compute Nodes in advance.

In this section, we introduce ① Spot service and ② On-demand service.
To execute a job, **"The type and quantity of computing resources to be used"** and **"The ABCI Group name which you belong"** are used as arguments.

Command syntax for Spot Service

```
qsub  -l rt_G.small=1 -g gaa54321  ./sample.sh
```

Resource type — Group Name — Job Script

Reference: ABCI 2.0 User Guide – Job Service

Computing Resources for Job Execution

Three types of computing resources:
Compute Nodes (V), Compute Nodes (A), and Memory-intensive Nodes
Many resource menus are available

- Compute Node (V) : Compute Node with NVIDIA V100 GPU, five types of menus are available
- Compute Node (A) : Compute Node with NVIDIA A100 GPU, two types of menus are available

Each resource has its own characteristics, and each menu has different CPU/memory and other specifications. There are restrictions on the number of nodes that can be used at the same time.

Reference: ABCI 2.0 User Guide – Job Execution Resource

Computing Resources for Job Execution

Compute Node (V)

Resource type	Resource type name	Description	Assigned physical CPU core	Number of assigned GPU	Memory (GiB)	Local storage (GB)	Resource type charge coefficient
Full	rt_F	Node-exclusive	40	4	360	1440	1.00
G.large	rt_G.large	Node-sharing with GPU	20	4	240	720	0.90
G.small	rt_G.small	Node-sharing with GPU	5	1	60	180	0.30
C.large	rt_C.large	Node-sharing CPU only	20	0	120	720	0.60
C.small	rt_C.small	Node-sharing CPU only	5	0	30	180	0.20

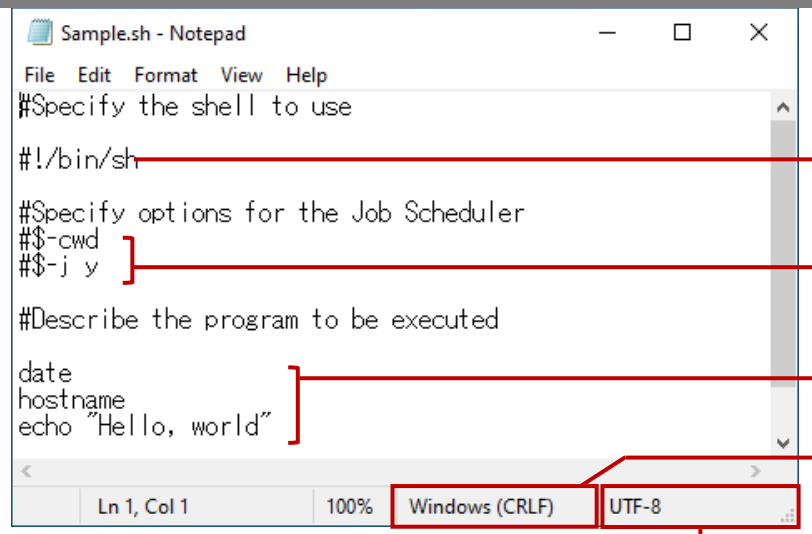
Compute Node (A)

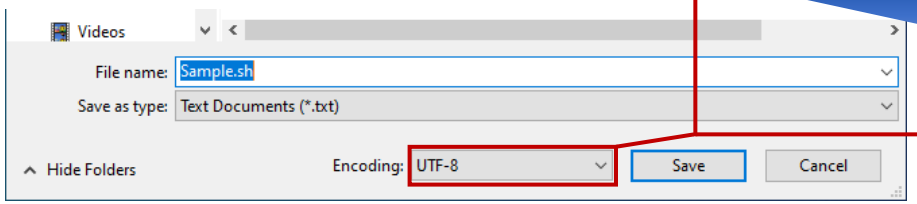
Resource type	Resource type name	Description	Assigned physical CPU core	Number of assigned GPU	Memory (GiB)	Local storage (GB)	Resource type charge coefficient
Full	rt_AF	Node-exclusive	72	8	480	3440	3.00
AG.small	rt_AG.small	Node-sharing with GPU	9	1	60	390	0.50

Reference: ABCI 2.0 User Guide – Available Resource Types

Creating a Job Script

Create a script on your terminal that describes the commands of the program to be executed, Upload it to ABCI





Specify the shell to use

Specify the options for the job execution command after "#$"

Describe the program to be executed

Note: Windows newline code is "CRLF", and it needs to be converted to "LF" to be handled in Linux.

When saving the file, use "UTF-8" (without BOM) for character encoding.

Upload the created and saved script file to ABCI

Reference: ABCI 2.0 User Guide – Job Execution Options

Upload Job Script

- Command line (1/2) -

Upload the created script to ABCI and change the newline code

Command Syntax

On Interactive Node

```
[aaa12345xx@es1 ~]$ mkdir Sample
[aaa12345xx@es1 ~]$ cd Sample
[aaa12345xx@es1 Sample]$
```

Create a directory (another name of Folder) "Sample" for job execution.

On Client PC

```
yourpc$ scp -P 10022 Sample.sh.txt aaa12345xx@localhost:~/Sample/
```

Upload the script to the created directory “Sample” by scp command.

On Interactive Node

```
[aaa12345xx@es1 Sample]$ nkf -Lu ./Sample.sh.txt > sample.sh
```

Require to change the newline code to LF by nkf command.

Upload Job Script

- Command line (2/2) -

Substitute and check character code and newline code

- If the file has CR+LF or CR for newline code:
  → Substitute newline code to LF for the Linux environment, and save as “sample.sh”

On Interactive Node

```
[aaa12345xx@es1 ~]$ nkf -Lu ./Sample.sh.txt > sample.sh
```

- If character code is Shift-JIS and newline code is CR+LF:
  → Substitute character code and newline code for the Linux environment (UTF-8, LF).

On Interactive Node

```
[aaa12345xx@es1 ~]$ nkf -wd ./SAMPLESH.TXT > sample.sh
```

- Check character code and newline code

On Interactive Node

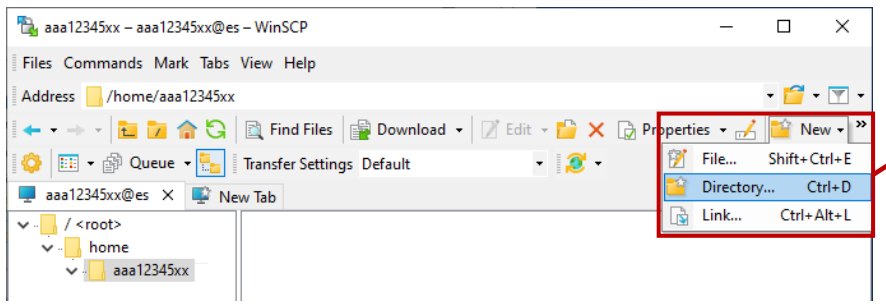
```
[aaa12345xx@es1 ~]$ nkf --guess sample.sh
sample.sh: UTF-8 (LF)
```

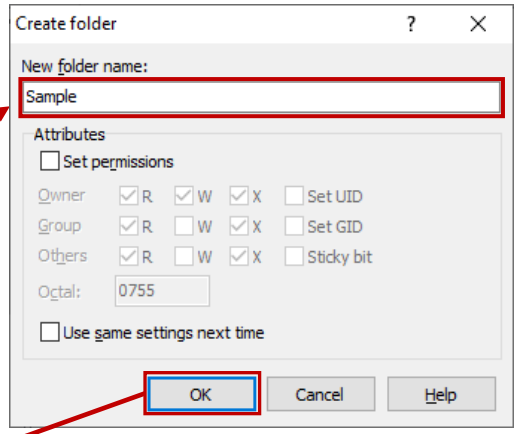
Upload Job Script

- WinSCP (1/2) -

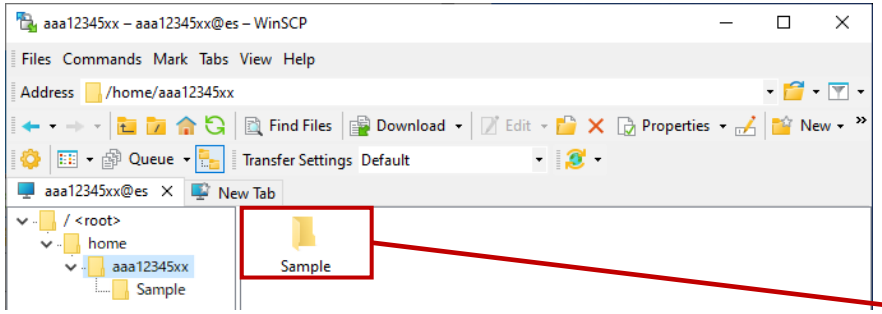
Created a new Directory on the Interactive Node

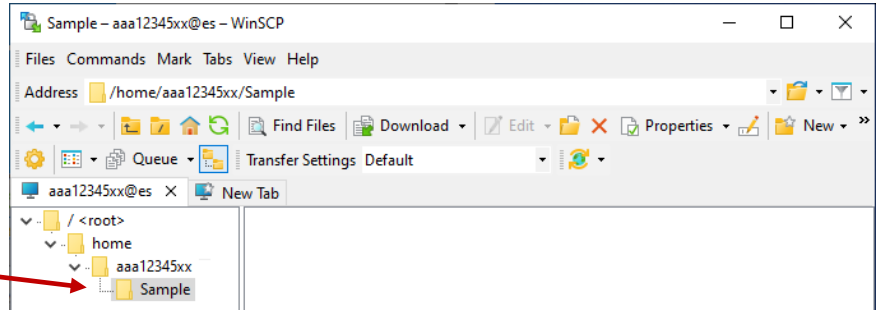
Create a new Directory (another name of Folder) “Sample”.





Move the current working directory





Upload Job Script

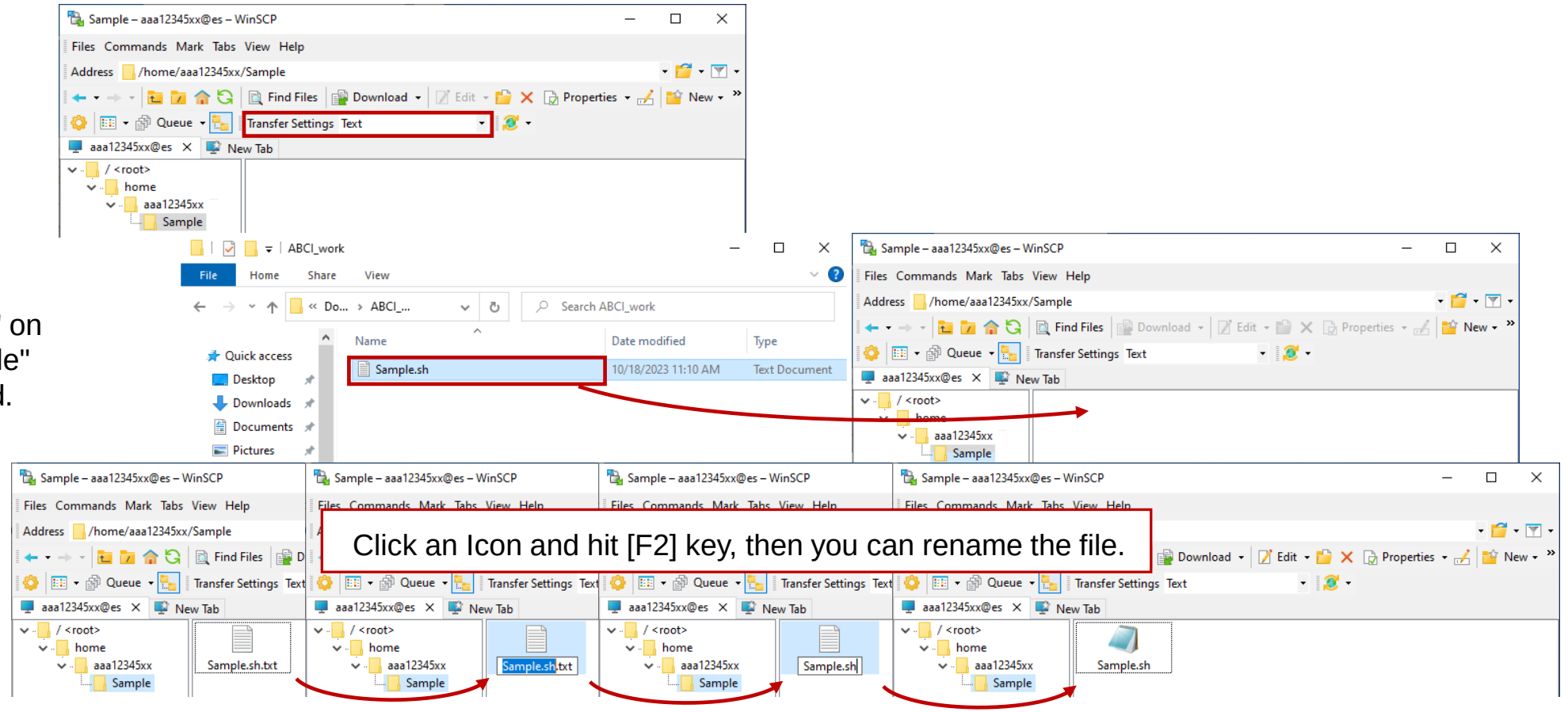
- WinSCP (2/2) -

Upload the created script to ABCI and change the file extension

Select “Text” for [Transfer Settings]

Upload "Sample.sh.txt" on your PC to the "Sample" directory you created.

Rename the file extension (.txt) to change (delete) it



Job Execution
— Spot Service —

Submit a batch job
$ qsub –l {Resource type}={quantity} –g {Group Name} {path to the Script file}

Command syntax

On Interactive Node

```
[aaa12345xx@es1 Sample]$ cat sample.sh
#Specify the shell to use
#!/bin/sh
#Specify job schedule option
#$-cwd
#$-j y
#Specify a program to run
date
hostname
echo "Hello, world"
[aaa12345xx@es1 Sample]$ chmod u+x ./sample.sh
[aaa12345xx@es1 Sample]$ ls -l
-rwxr--r-- 1 username usergroup 231 Jan 1 12:00 sample.sh
[aaa12345xx@es1 Sample]$ qsub -l rt_G.small=1 -g gaa54321 ./sample.sh
Your job 12345678 ("sample.sh") has been submiitted
```

Review the job script

Grant execution rights to the job script

Confirm that the job script has "x" execution rights

Specify Resource type

Specify your Group Name

Reference: ABCI 2.0 User Guide – Job Execution Options

Show Result of Executed Job

— Spot Service —

The result of the execution is recorded in the file output after the job is finished.

Command syntax

On Interactive Node

```
[aaa12345xx@es1 Sample]$ ls -l
-rwxr--r- 1 username usergroup 231 Jan 1 12:00 sample.sh
-rw-r--r- 1 username usergroup 59  Jan 1 12:00 sample.sh.o12345678
[aaa12345xx@es1 Sample]$
[aaa12345xx@es1 Sample]$ cat sample.sh.o12345678
Sun Jan 1 12:00:00 JST 2023
g0009.abci.local
Hello, world
[aaa12345xx@es1 Sample]$
```

Output file of stderr of the job

Result executed on the Compute Node

Job Execution
— On-demand Service —

Execute an interactive job.
$ qrsh –l {Resource type}={quantity} –g {Group Name}

Command syntax

On Interactive Node

```
[aaa12345xx@es1 ~]$ qrsh -l rt_G.small=1 -g gaa54321
[aaa12345xx@g0009 ~]$
[aaa12345xx@g0009 ~]$ uname -n
g0009.abci.local
[aaa12345xx@g0009 ~]$ cd Sample
[aaa12345xx@g0009 Sample]$ ls -l
-rwxr--r-- 1 username usergroup 231 Jan 1 12:00 sample.sh
[aaa12345xx@g0009 Sample]$ ./sample.sh
Sun Jan 1 12:00:00 JST 2023
g0009.abci.local
Hello, world
[aaa12345xx@g0009 Sample]$
```

Specify the Resource type and your Group Name

Where are you logging in

Result executed on the Compute Node

Reference: ABCI 2.0 User Guide – Interactive Jobs

Show Status of Job Execution

Command Syntax

$ qstat –{option}

Where Option

-r Show the Job Resource information
-j Show Additional information of the Job

Example:
in Batch Jobs

```
[aaa12345xx@es1 ~]$ qstat
job-ID     prior      name  user         state  submit/start at            queue          jclass  slots ja-task-ID
-----------------------------------------------------------------------------------------------------------------
114535     0.25586    sample.sh   aaa12345xx  r     01/01/2023 12:00:00  gpu@g0016              10
```

Example:
in Interactive Jobs

```
[aaa12345xx@g0009 ~]$ qstat
job-ID     prior      name  user         state  submit/start at            queue          jclass  slots ja-task-ID
-----------------------------------------------------------------------------------------------------------------
114535     0.25586    QRLOGIN     aaa12345xx  r     01/01/2023 12:00:00  gpu@g0016              10
```

Reference: [ABCI 2.0 User Guide – Show the status of batch jobs]

Flow of Job Execution

— Summary —

Executing a job with ABCI is actually simple. Let's try it!

① Create a script file

② Upload the script file

```
$ scp -P 10022 {localfile} aaa12345xx@localhost:./
```

③ Change character code and newline code

```
[aaa12345xx@es1]$ nkf -Lu ./Sample.sh.txt > sample.sh
```

Grant execution rights

```
[aaa12345xx@es1]$ chmod u+x ./sample.sh
```

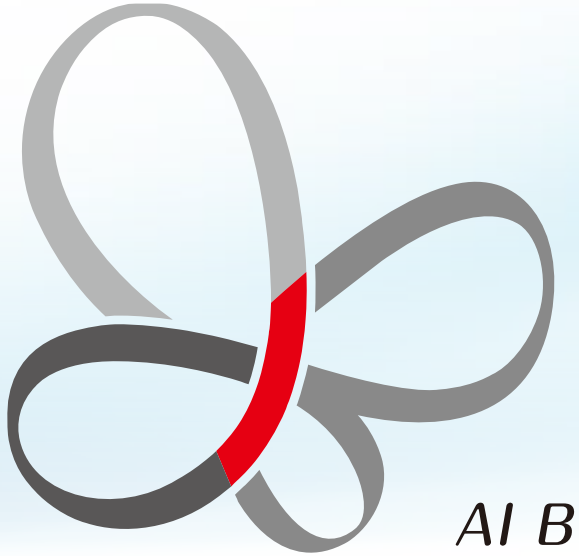
④ Execute the job
Optionally specify the resource type and group name

```
[aaa12345xx@es1]$ qsub -l rt_G.small=1 -g gaa54321 ./sample.sh
```

```
[aaa12345xx@es1]$ qrsh -l rt_G.small=1 -g gaa54321
```

Reference Links

- ABCI Official Website (https://abci.ai/)
- ABCI 2.0 User Guide (https://docs.abci.ai/en/)
  Describes the technical details of ABCI 2.0 and how to use it.
- WinSCP (https://winscp.net)



ABCI

AI Bridging Cloud Infrastructure

https://abci.ai/