



# 大規模AIクラウド計算システム「ABC I」 利用チュートリアル動画（ジョブ実行編）

国立研究開発法人 産業技術総合研究所

情報・人間工学領域

デジタルアーキテクチャ研究センター

# 目次

- ABCIシステムを使う、その前に
- ABCI2.0システムの概要
- ABCIシステム利用の流れ
- ABCIシステムへログイン
- データのアップ/ダウンロード
  - ターミナルからscpコマンドを使う場合 –
  - WinSCPを使う場合 –
- ジョブ実行のサービス
- ジョブ実行で使う計算リソース
- ジョブの実行
- ジョブの実行の流れ – まとめ –
- 参考リンク

# ABCIシステムを使う、その前に

ABCIシステムを使うには、ご利用の端末においてLinuxのコマンドが使える必要があります



The screenshot shows the ABCI User Group website. The main content is a video titled "ビデオ：ABCI利用チュートリアル（事前準備）" (Video: ABCI Usage Tutorial (Preparation)). The video description includes a link to a YouTube video: 「初めてのABCI利用法 How to use ABCI」 (https://youtu.be/WwWPo9Bxm-g). Below the video, there is a table of contents (目次) listing topics such as "クライアントPCの環境構築" (Client PC environment setup), "ABCIシステム利用手順" (ABCI system usage procedure), "利用者ポータルへアクセス" (Access to the user portal), "公開鍵・秘密鍵生成" (Public/private key generation), "公開鍵の登録" (Public key registration), and "ABCIシステムへログイン" (Login to the ABCI system). There is also a "スライドデータ" (Slide data) section with a link to "大規模AIクラウド計算システム「ABCI」利用チュートリアル動画（事前準備編）" (Large-scale AI cloud computing system "ABCI" usage tutorial video (Preparation)).

下記のABCI User Groupのホームページに  
ABCIに接続するまでの手順を紹介をしています。  
ABCIに接続したことない人はご参考ください。

ABCI User Group 「みんなのABCI」  
ビデオ：ABCI利用チュートリアル（事前準備）

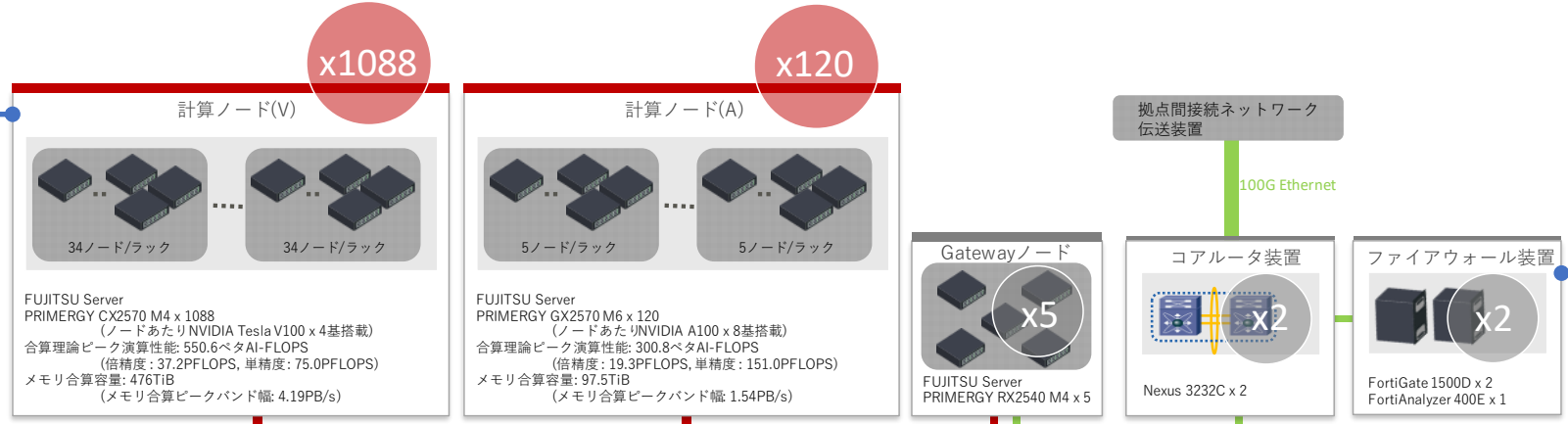
URL : [https://abciug.abci.ai/入門編\\_1/ビデオabci利用チュートリアル事前準備\\_i81](https://abciug.abci.ai/入門編_1/ビデオabci利用チュートリアル事前準備_i81)

参考：[ビデオ：ABCI利用チュートリアル\(事前準備\) ABCI User Group](#)

# ABCI2.0システムの概要

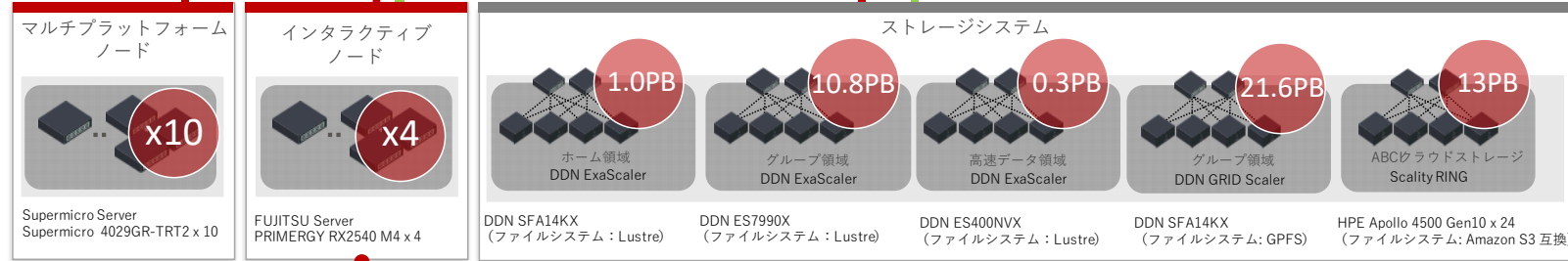
ABCI2.0システムは、計算リソース・大容量ストレージを高速ネットワークで結合したシステム

① 計算リソース  
計算ノード(V) 1088台  
計算ノード(A) 120台



④ ファイアウォールなど

③ InfiniBand  
ネットワーク

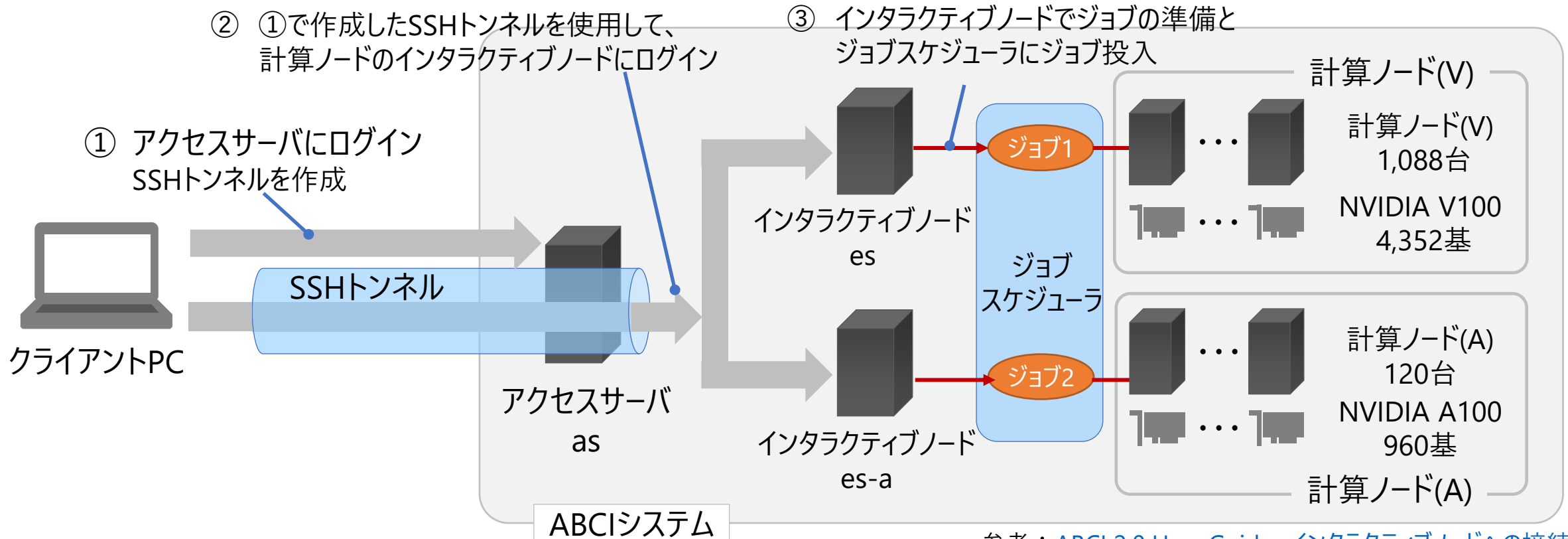


② ストレージシステム  
合計47PB  
共有ファイルシステム  
ABCIクラウドストレージ

インタラクティブノード

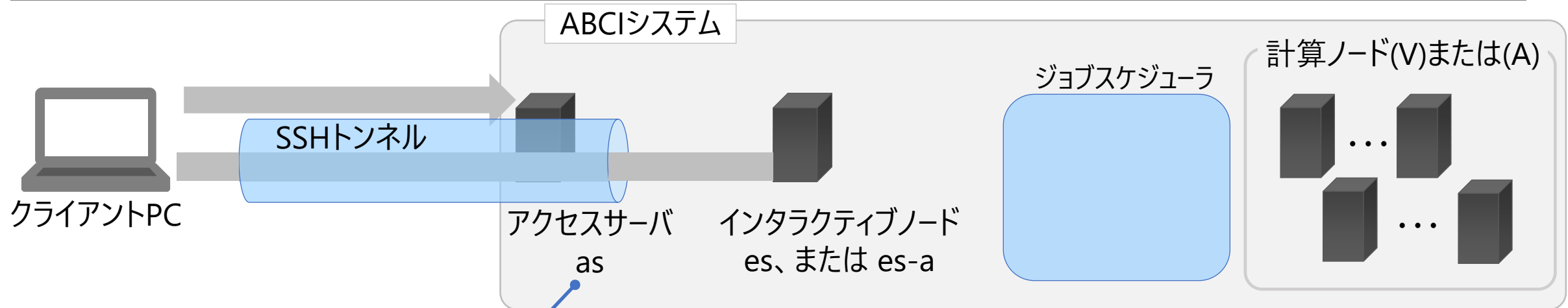
# ABCIシステムへログインからジョブ実行の流れ

ABCIシステムへアクセス、ssh接続を確立し、インタラクティブノードでジョブの準備  
ジョブの実行はジョブスケジューラから計算ノードへ投入



# ABCIシステムにおける各ノードの役割 (1/3)

まず最初はアクセスサーバ(as)にログインして、SSHトンネルを作成する

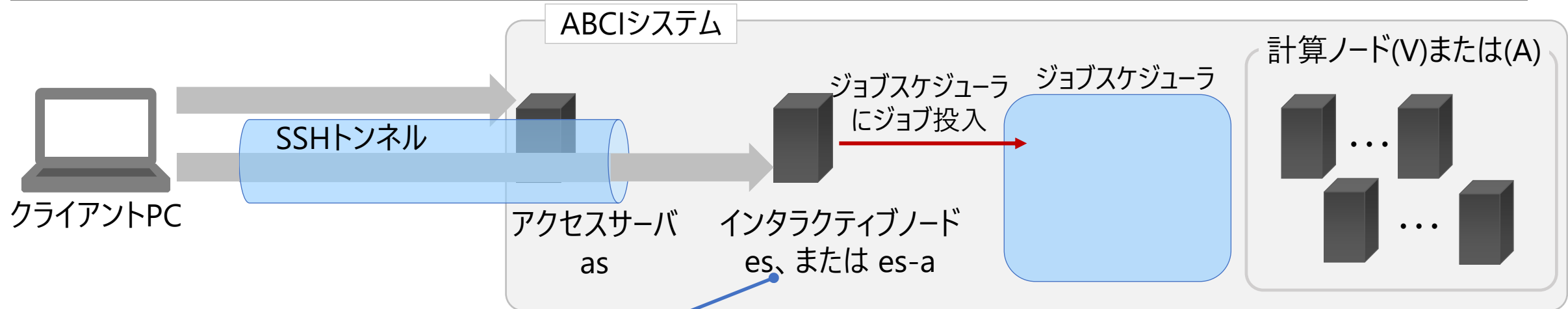


## アクセスサーバ

- インタラクティブノードにアクセスするため、最初にログインするサーバ。操作不可の踏み台サーバ。
- インタラクティブノードにアクセスするSSHトンネルを作る。
- コマンド操作をおこなうと、インタラクティブノードなどへのSSHセッションが切れることに注意。

# ABCIシステムにおける各ノードの役割 (2/3)

続いて、インタラクティブノードへログインし、ジョブの実行の準備をする



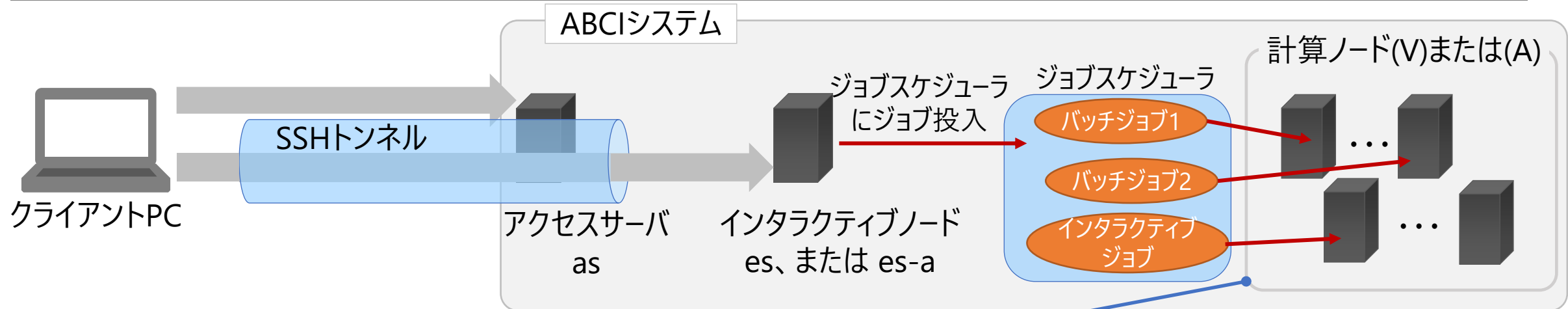
## インタラクティブノード

利用者がログインし、ジョブ実行などの作業を行うABCIのサーバ

- コンパイル、アプリのインストール、ジョブ実行スクリプトの用意
- ジョブスケジューラにジョブ実行依頼
- ABCI ヘデータアップロード、ダウンロード

# ABCIシステムにおける各ノードの役割 (3/3)

ジョブスケジューラにジョブを登録して、確保された計算ノードでジョブを実行する



## 計算ノード (V)または(A)

利用者のジョブ（プログラム）を実際に実行するサーバ群

- ジョブスケジューラから受けたジョブを実行
- ジョブの実行方法

### ① Spotサービス（バッチジョブ実行）

ジョブスケジューラにジョブの実行を依頼し、確保された計算ノードでジョブを実行

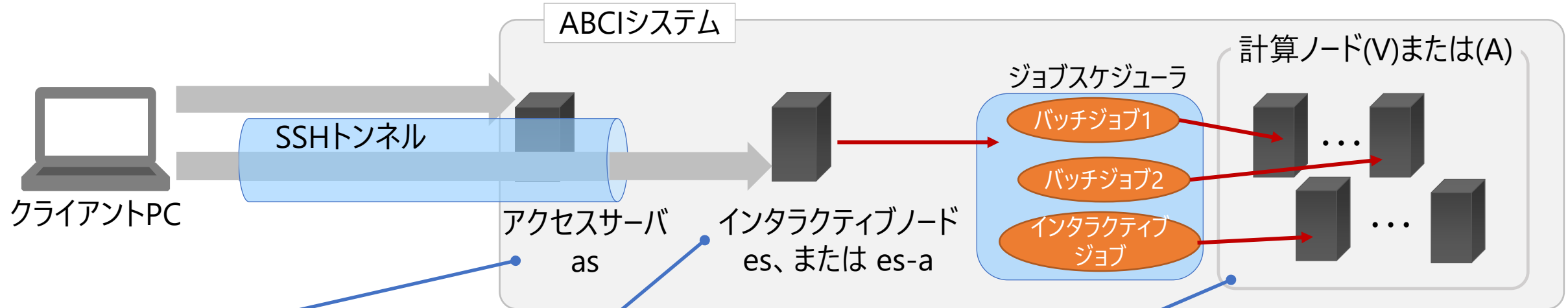
### ② On-demandサービス（インタラクティブジョブ実行）

ジョブスケジューラが確保した計算ノードに利用者が直接ログインしプログラムを実行



# ABCIシステムにおける各ノードの役割 (まとめ)

ABCIシステムにおけるジョブの実行には、複数のノードを経由して計算ノードで実行する



## アクセスサーバ

- インタラクティブノードにアクセスするため、最初にログインするサーバ。
- コマンド操作をおこなうと、SSHセッションが切れるので注意

## インタラクティブノード

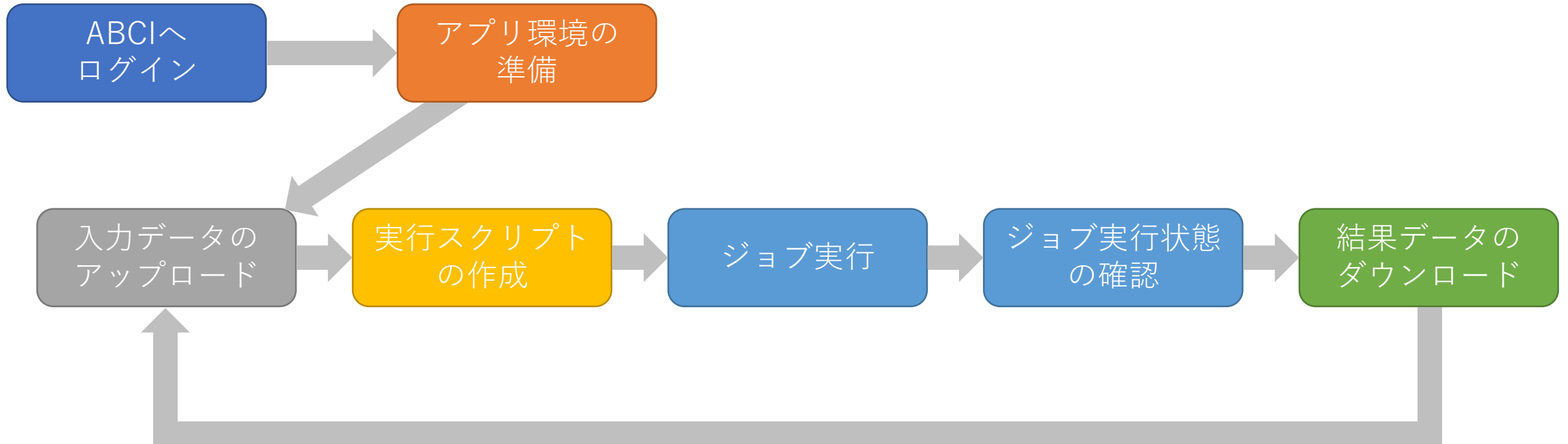
- 利用者がログインし、ジョブ実行などの作業を行うABCIのサーバ
- コンパイル、アプリのインストール、ジョブ実行スクリプトの用意
  - ジョブスケジューラにジョブ実行依頼
  - データアップロード、ダウンロード

## 計算ノード

- 利用者のジョブ（スクリプト）を実際に実行するサーバ群
- ジョブスケジューラから受けたジョブを実行
  - ジョブの実行方法
    - ① Spotサービス（バッチジョブ実行）
    - ② On-demandサービス（インタラクティブジョブ実行）

# ABCIシステム利用の流れ

ジョブの実行結果を確認しながら、再度、データのアップロードとジョブ実行を繰り返す

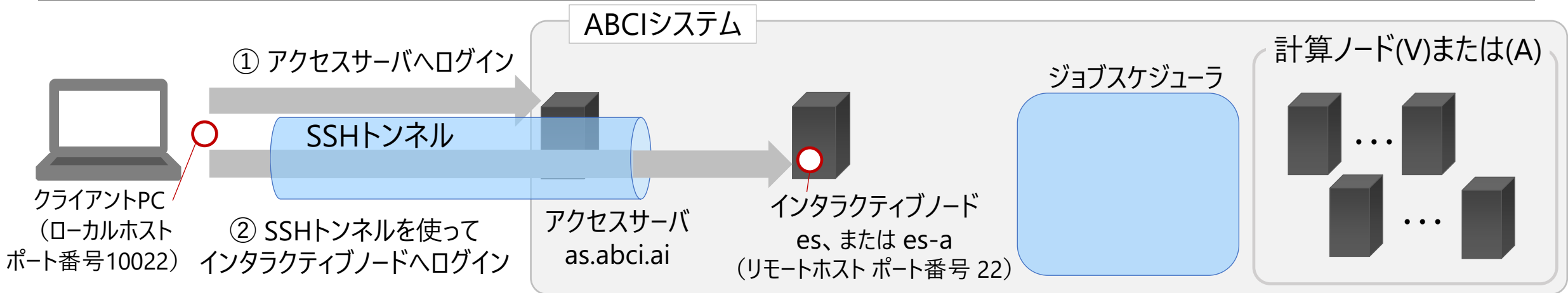


結果を見て、再度繰り返し

# ABCIシステムへログイン

## - ターミナルからログインする場合 -

- ① ターミナルから、アクセスサーバ(as)にSSHでログインし、SSHトンネルを設定します
- ② 別のターミナルを開き、インタラクティブノード(es)へログインします



- ① アクセスサーバへログイン コマンド書式例

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

-L (クライアントPCで受けるポート番号) : (接続先インタラクティブノード名) : (インタラクティブノードのポート番号) -l (ABCIアカウント名) (アクセスサーバ名)

- ② インタラクティブノードへログイン コマンド書式例

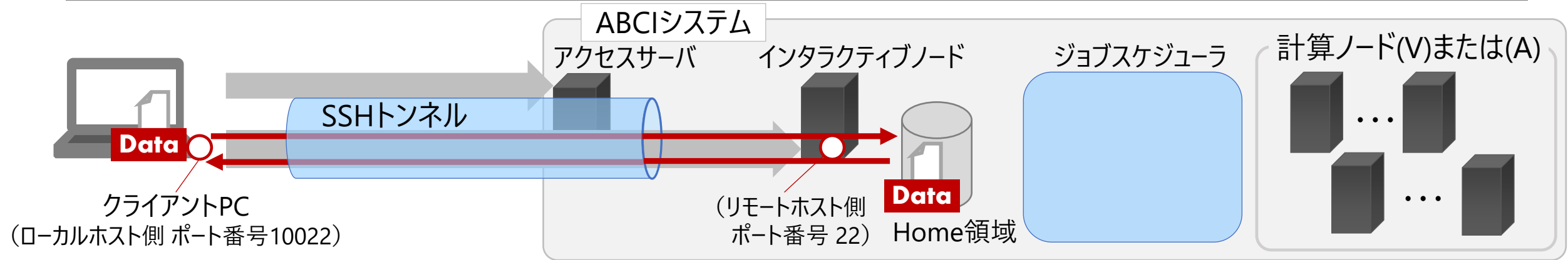
```
ssh -p 10022 -l aaa12345xx localhost
```

-p (クライアントPCで受けるポート番号) -l (ログインするABCIアカウント名) (ホスト名)

# データのアップ/ダウンロード

## – ターミナルからscpコマンドを使う場合 (1/3) –

SSHトンネルを利用して、データのアップロード・ダウンロードが可能



### ① クライアントPCからデータをアップロードする場合

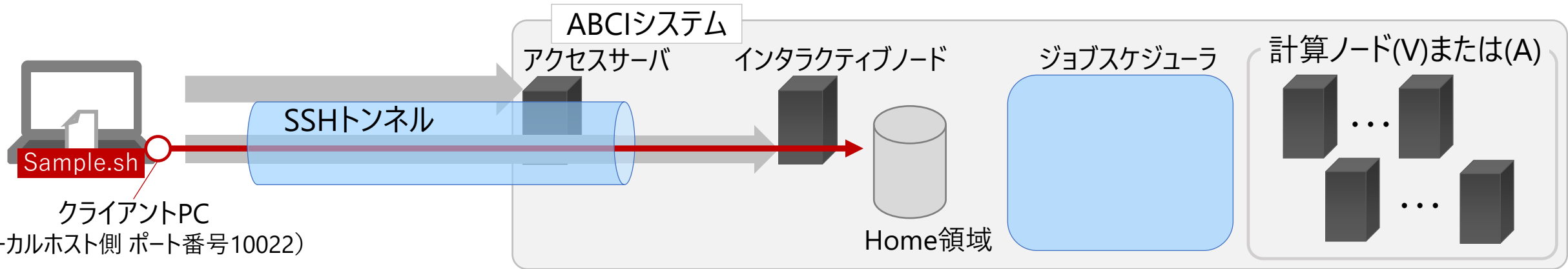
コマンド書式 `$scp -P 10022 ローカルファイル名 aaa12345xx@localhost:./###/リモートの転送先ファイル名`  
 -P (クライアントPCのローカルホストポート番号) (転送するファイル) (ABCIアカウント名)@localhost: (転送先で受けるディレクトリのパスとファイル名)

### ② データをクライアントPCへダウンロードする場合

コマンド書式 `$scp -P 10022 aaa12345xx@localhost:リモートのファイル名 ./###/ローカル転送先ファイル名`  
 -P (クライアントPCのローカルホストポート番号) (ABCIアカウント名)@localhost: (転送するファイル) (クライアントPC側の保存パスとファイル名)

# データのアップロード

## ー ターミナルからscpコマンドを使う場合 (2/3) ー

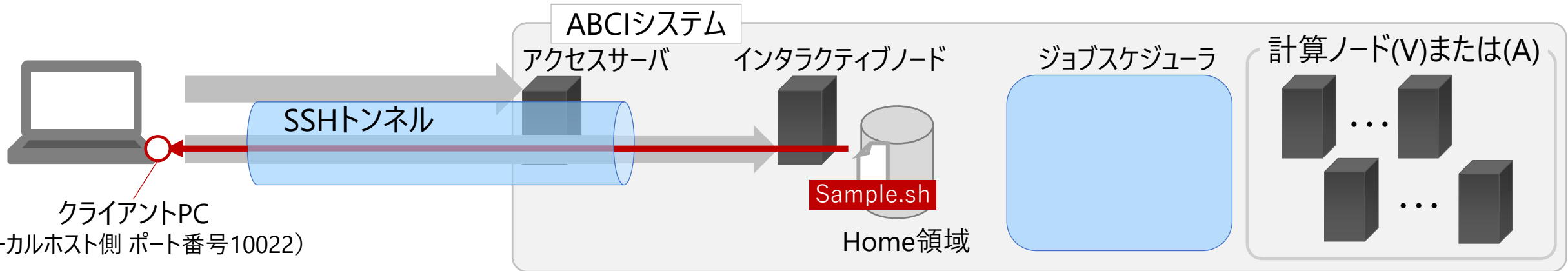


### コマンド例

インタラクティブノード側	<pre>[abc username@es1 ~]\$ls</pre>	ホームディレクトリのファイルを表示するが、データは何もない状態
クライアントPC側	<pre>yourpc\$ scp -P 10022 sample.sh aaa12345xx@localhost:./sample.sh</pre> Enter passphrase for key '/home/username/.ssh/id_rsa': sample.sh 100% ##(ファイルサイズ) ##KB/s(転送速度) ##:##(転送時間)	クライアントPCのカレントディレクトリにある「sample.sh」をABCISystem上の自身のホームディレクトリに転送 設定した鍵のパスフレーズを入力すると、指定したファイルが転送される
インタラクティブノード側	<pre>[abc username@es1 ~]\$ls</pre> sample.sh	scpコマンドでファイル転送後、ホームディレクトリのファイルを表示すると、指定したファイルが格納されている

# データのダウンロード

## ー ターミナルからscpコマンドを使う場合 (3/3) ー



### コマンド例

クライアント  
PC側

```
yourpc$ scp -P 10022 aaa12345xx@localhost:./sample.sh ./sample.sh
```

```
Enter passphrase for key '/home/username/.ssh/id_rsa':
```

```
sample.sh      100%  ##(ファイルサイズ)  #.#KB/s(転送速度)  ##:##(転送時間)
```

```
yourpc$ls
```

```
sample.sh
```

クライアントPCのカレントディレクトリにある「sample.sh」を ABCI上の自身のホームディレクトリに転送

設定した鍵のパスフレーズを入力すると、指定したファイルが転送される

scpコマンドでファイル転送後、作業ディレクトリのファイルを表示すると、指定したファイルが格納されている

# データのアップ/ダウンロード

## － WinSCPを使う場合 (1/7) －

### ① WinSCPをダウンロード・インストールする



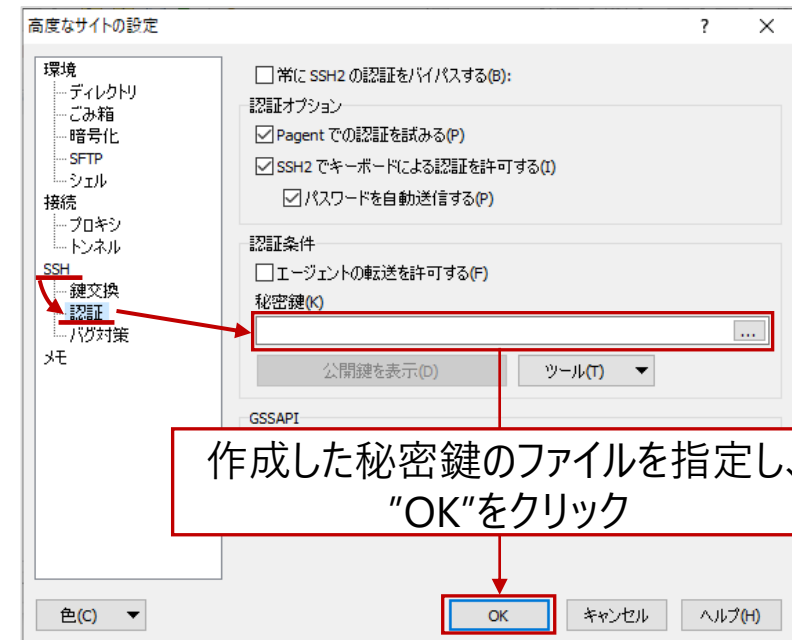
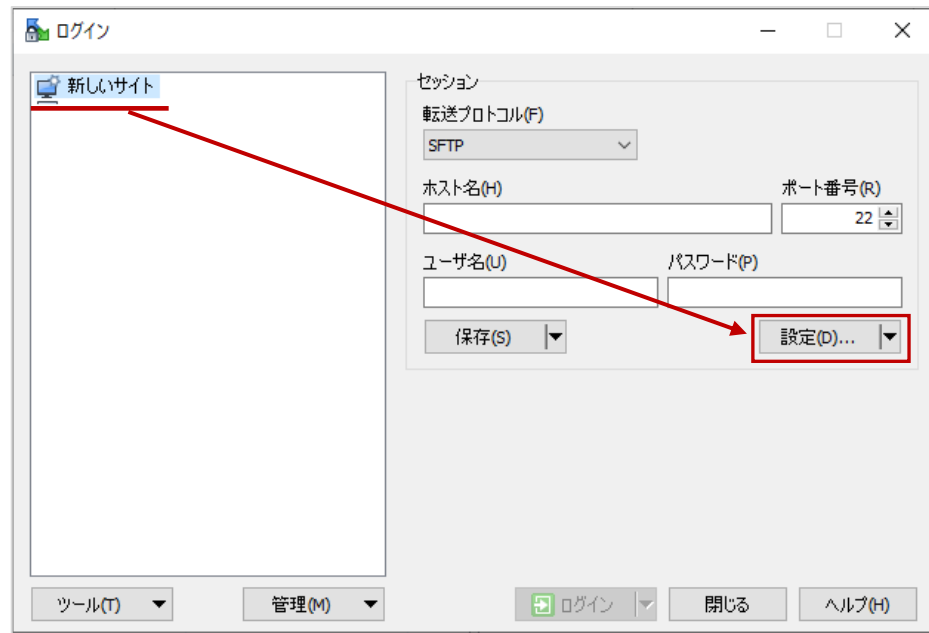
【Windowsをお使いの方向け】

- WinSCPダウンロードURL  
<https://winscp.net/eng/download.php>
- インストールはガイドに従ってインストールする
- 本説明ではWinSCPの「コマンダー」と呼ばれる、ローカル側とリモート側の2つのディレクトリが表示されるモードを利用する
- PuTTYでABCIのログイン設定を設定済みの場合、そのログイン設定をインポートしてくれる

# データのアップ/ダウンロード

## － WinSCPを使う場合 (2/7) －

### ② 秘密鍵情報を登録する



- WinSCPを起動すると[ログイン]ウィンドウが開くので、  
[新しいサイト] → [設定(D)...]をクリック
- もし[ログイン]ウィンドウが開かない場合  
WinSCPの画面のメニューバーから  
[セッション(S)] → [新しいセッション(N)] を選択する

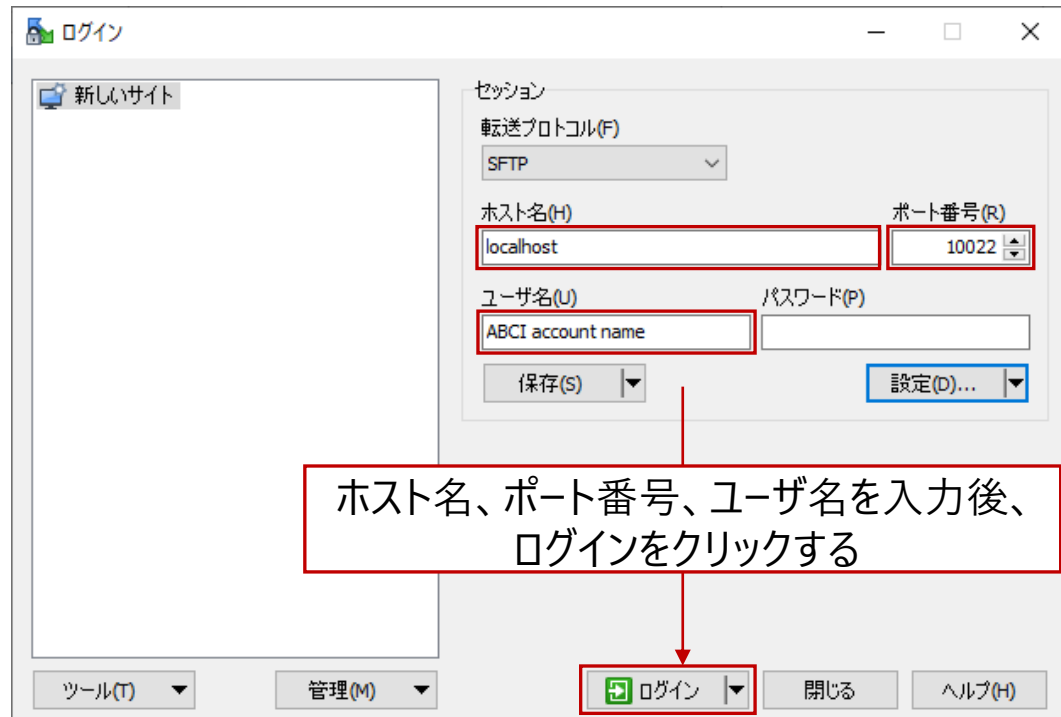
- [高度なサイトの設定]画面の左ペインメニューより  
[SSH] → [認証]  
を選択する
- 画面右側の[秘密鍵(K)]欄において、作成した秘密鍵を指定して、  
画面下部の[OK]をクリックして、設定情報を保存する



# データのアップ/ダウンロード

## － WinSCPを使う場合 (3/7) －

### ③ インタラクティブノードへの接続情報を登録して、ログインする

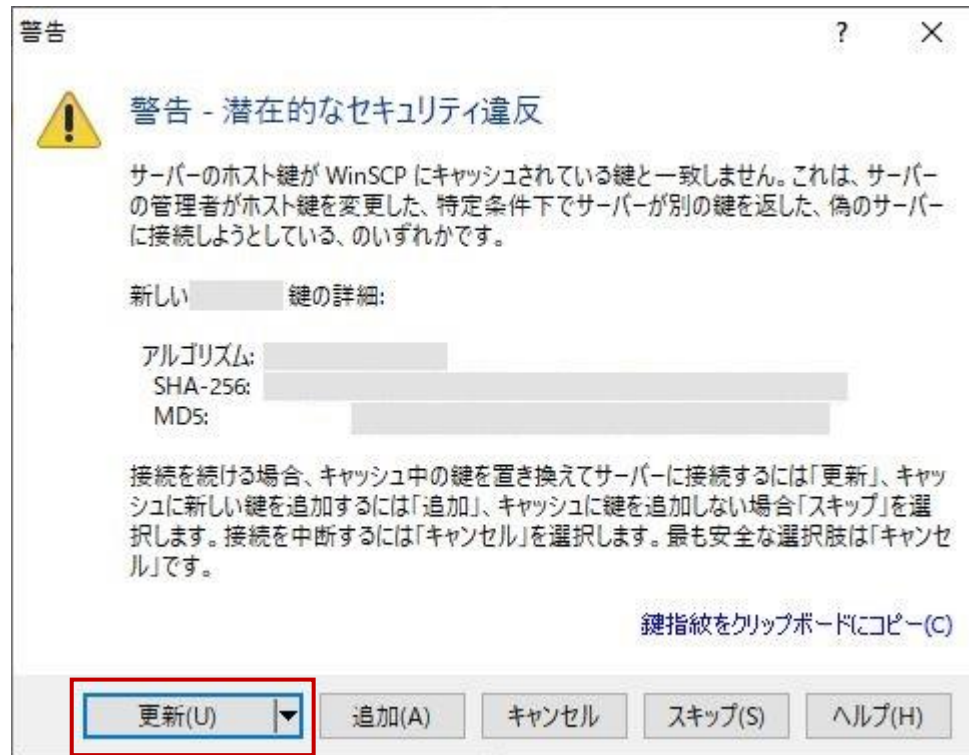


- 左のCategoryメニューより
  - ✓ ホスト名(H) : localhost
  - ✓ ポート番号(R) : 10022  
(SSHトンネルで設定したポート番号を指定)
  - ✓ ユーザ名(U) : ABCIアカウント名  
を入力する
- [ログイン]をクリックして、インタラクティブノードへ接続する

# データのアップ/ダウンロード

## － WinSCPを使う場合 (4/7) －

### もし「警告」が表示されたら

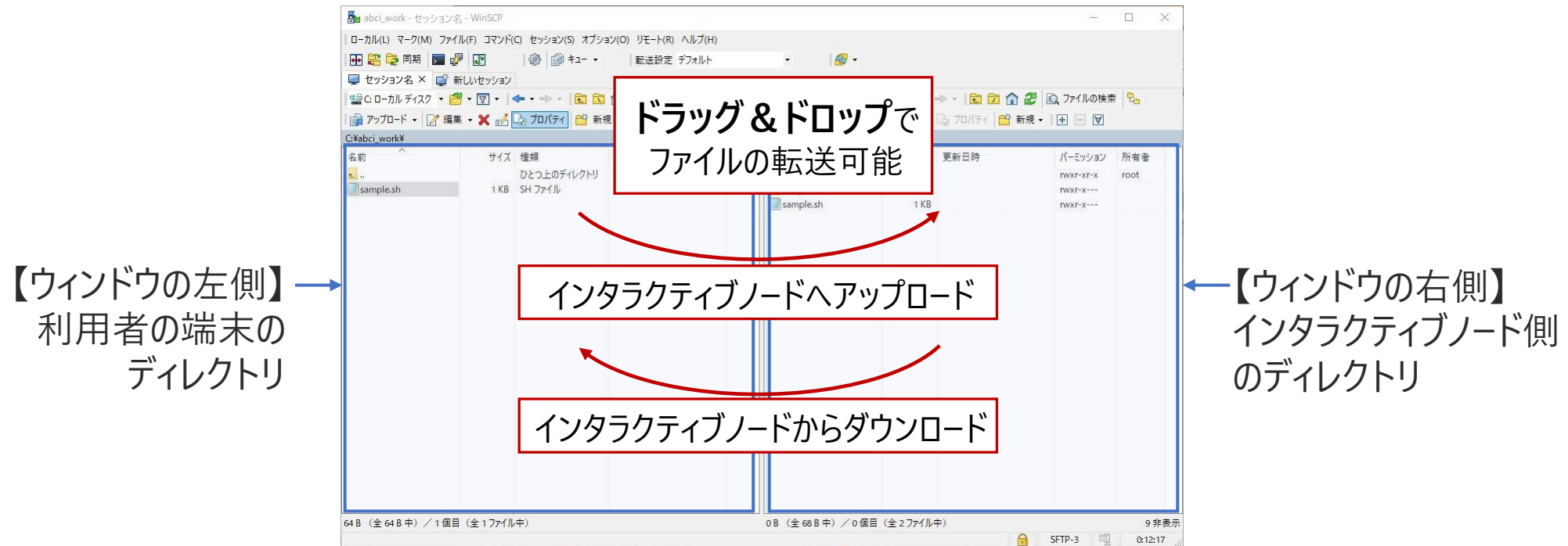


- 初めてWinSCPでABCIに接続する場合に表示されます。
- 「更新」ボタンを選択します。  
(お使いの環境によっては「Warning」画面が表示され、選択するボタンは「Update」を選択します。)

# データのアップ/ダウンロード

## － WinSCPを使う場合 (5/7) －

### ④ ファイルをドラッグ & ドロップするだけで、ファイルのアップロード・ダウンロードが可能



- インタラクティブノードにログインできると、ローカル側・接続先のリモート側、両方のディレクトリが表示される。  
左側：利用者端末のディレクトリ、右側：インタラクティブノードのディレクトリ
- ファイルをドラッグ & ドロップするだけで、ファイルのアップロード・ダウンロードが可能

# データのアップロード

## － WinSCPを使う場合 (6/7) －

### ⑤ ファイルをドラッグ & ドロップするだけで、ファイルのアップロードが可能

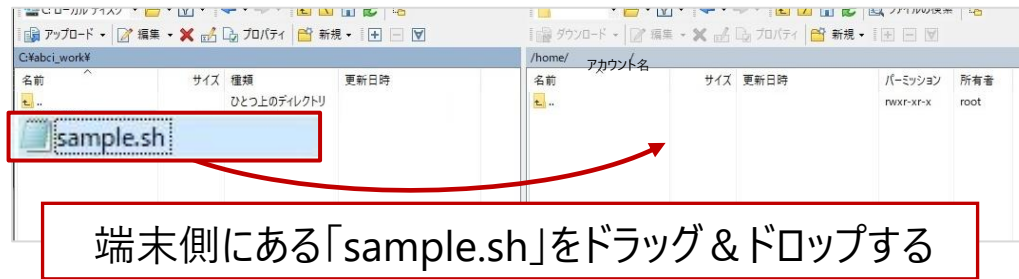
- WinSCPでファイルをアップロードする前

```
- https://abci.ai/en/how_to_use/user_support.html (In English)
- https://abci.ai/ja/how_to_use/user_support.html (In Japanese)

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

ファイル・データは何もない状態

- ログインしたインタラクティブノードには、ファイルやデータは何もない状態



- 左側：利用者端末のディレクトリを表示するエリアにあるファイルを、右側：インタラクティブノードのディレクトリを表示するエリアに、ドラッグ & ドロップして、ファイルのアップロードを実行

- WinSCPでファイルをアップロードした後

```
- https://abci.ai/ja/how_to_use/user_support.html (In Japanese)

[##### @es2 ~]$ ls
[##### @es2 ~]$ ls
sample.sh
[##### @es2 ~]$
```

転送したファイルがある状態

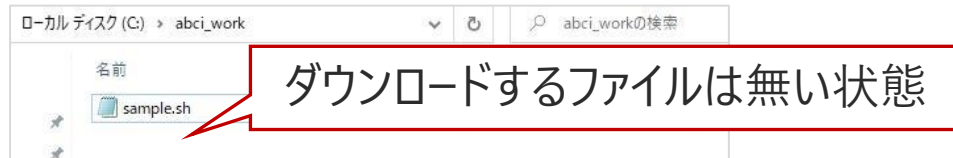
- ファイル・データは何もなかったところに WinSCPで転送したファイル「sample.sh」がある

# データのダウンロード

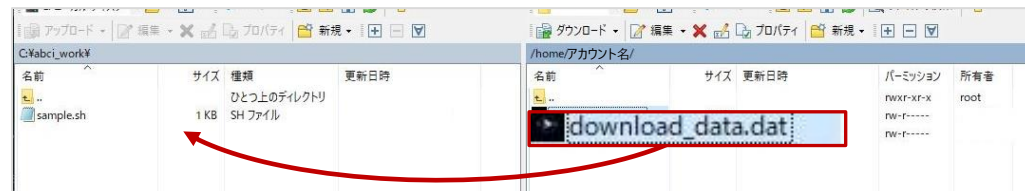
## － WinSCPを使う場合 (7/7) －

### ⑥ ファイルをドラッグ & ドロップするだけで、ファイルのダウンロードが可能

#### ・ データをダウンロードする前



- ・ ダウンロードしたいファイルはローカル側のディレクトリにはない状態



- ・ 右側：インタラクティブノードのディレクトリにあるファイルを、左側：利用者端末のディレクトリを表示するエリアにドラッグ & ドロップして、ファイルのダウンロードを実行

#### ・ WinSCPでファイルをダウンロードした後



- ・ WinSCPで転送したファイル「download\_data.dat」がローカル側の端末のディレクトリ内にある

# ジョブ実行のサービス

ジョブの実行には、3種類のサービスから処理方法に応じた適切なサービスを選択する

## ジョブ実行のサービスの種類

- ① Spotサービス : ジョブ実行スクリプト作成し、ジョブスケジューラにバッチ処理依頼する
- ② On-demandサービス : ジョブスケジューラに計算ノードの確保を依頼し、計算ノードでプログラムを実行する
- ③ Reservedサービス : 事前に計算ノードの予約をジョブスケジューラに依頼するサービス

本説明では、① Spotサービスと、② On-demandサービスについてご紹介  
ジョブの実行には、

「**利用する計算リソースの種類とその数量**」、「**自分が所属するABCIグループ名**」を引数に使用

Spotサービスの  
コマンド例

```
qsub -l rt_G.small=1 -g gxx12345 ./sample.sh
```

リソースの種類

グループ名

実行スクリプト

参考: [ABCI 2.0 User Guide – ジョブサービス](#)

# ジョブ実行で使う計算リソース

計算ノード(V)、計算ノード(A)に加え、メモリンテンシブノードの3種に  
多数のリソースメニューを用意

- 計算ノード(V) : GPUにNVIDIA V100を搭載した計算ノード、計算ノード(V)には5種類のメニューがある
- 計算ノード(A) : GPUにNVIDIA A100を搭載した計算ノード、計算ノード(A)には2種類のメニューがある
- メモリンテンシブノード : Intel Optaneを搭載した大容量のメモリを持つノード、2種類のメニューがある  
(注意 : 2023年10月にサービス提供終了予定)

それぞれのリソースに特徴があり、またメニューごとにCPU/メモリなどのスペックが異なり、同時に使えるノード数の制約などがある

# ジョブ実行で使う計算リソース

計算ノード(V)

資源タイプ	資源タイプ名	説明	割り当て物理CPUコア数	割り当てGPU数	メモリ (GiB)	ローカルストレージ (GB)	資源タイプ課金係数
Full	rt_F	ノード占有	40	4	360	1440	1.00
G.large	rt_G.large	ノード共有 GPU利用	20	4	240	720	0.90
G.small	rt_G.small	ノード共有 GPU利用	5	1	60	180	0.30
C.large	rt_C.large	ノード共有 CPUのみ利用	20	0	120	720	0.60
C.small	rt_C.small	ノード共有 CPUのみ利用	5	0	30	180	0.20

計算ノード(A)

資源タイプ	資源タイプ名	説明	割り当て物理CPUコア数	割り当てGPU数	メモリ (GiB)	ローカルストレージ (GB)	資源タイプ課金係数
Full	rt_AF	ノード占有	72	8	480	3440	3.00
AG.small	rt_AG.small	ノード共有 GPU利用	9	1	60	390	0.50

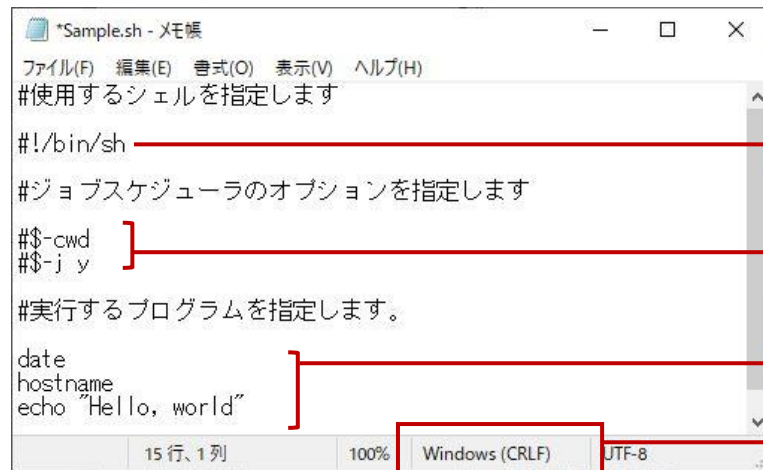
メモリ  
インテンシ  
ブノード

資源タイプ	資源タイプ名	説明	割り当て物理CPUコア数	割り当てGPU数	メモリ (GiB)	ローカルストレージ (GB)	資源タイプ課金係数
M.Large	rt_M.large	ノード共有 CPUのみ利用	8	-	800	480	0.40
M.Small	rt_M.small	ノード共有 CPUのみ利用	4	-	400	240	0.20



# 実行スクリプトの作成

実行するプログラムのコマンドを記述したスクリプトを端末で作成して、  
ABCIへアップロードする



```
*Sample.sh - メモ帳
ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)
#使用するシェルを指定します
#!/bin/sh
#ジョブスケジューラのオプションを指定します
#$-cwd
#$-j y
#実行するプログラムを指定します。
date
hostname
echo "Hello, world"
15行、1列 100% Windows (CRLF) UTF-8
```

使用するシェルを指定

ジョブ実行のコマンドのオプションを「#\$」のあとに指定

実行するプログラムを記述

注意：Windowsの改行コードは「CRLF」、  
Linuxでは扱えず、変換が必要



ファイルを保存するときは、文字コードは「UTF-8」で保存

作成・保存したスクリプトファイルをABCIへアップロードする

# ジョブの実行

## — Spotサービス：スクリプトの作成 —

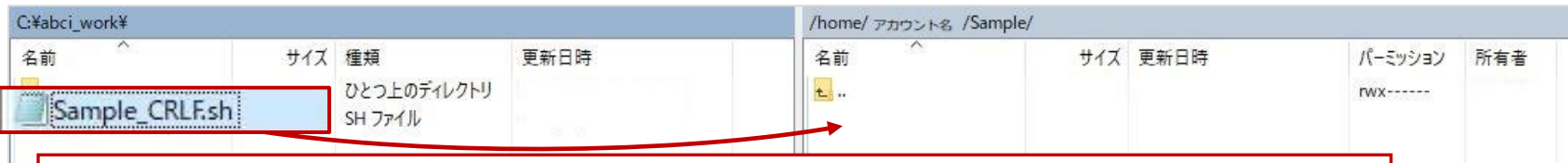
作成したスクリプトをABCIへアップロードして、改行コードを変更する

コマンド例  
インタラクティブ  
タイプノード側

```
[abci username@es1]$mkdir Sample
[abci username@es1]$cd Sample
[abci username@es1 Sample]
```

ジョブ実行用に  
「Sample」ディレクトリ  
を作成する

クライアント  
PC側



端末側にある「Sample\_CRLF.sh」を作成した「Sample」ディレクトリへアップロードする

WinSCPまたはscpコマンド  
で、作成した「Sample」  
ディレクトリにスクリプトを  
アップロードする

```
yourpc$ scp -P 10022 Sample_CRLF.sh aaa12345xx@localhost:./Smple/Sample_CRLF.sh
```

インタラクティブ  
タイプノード側

```
[abci username@es1 Sample] $nkf -Lu ./Sample_CRLF.sh > sample.sh
```

nkfコマンドで  
改行コードを変更する

# ジョブの実行

## — Spotサービス (文字コード・改行コード変更) —

文字コード・改行コードを変換と変換の確認もできる

- 改行コードがCR+LFまたはCRの場合：  
→ 改行コードをLinux環境(LF)に変更し、sample.shで保存する

インタラクティブ  
タイプノード  
側

```
[abc@username@es1 ~] $nkf -Lu ./Sample_CRLF.sh > sample.sh
```

- 文字コードがShift-JIS、改行コードがCR+LFになっている場合：  
→ 文字コードと改行コードをLinux環境(UTF-8, LF)に一括変換する

インタラクティブ  
タイプノード  
側

```
[abc@username@es1 ~] $nkf -wd Sample_SJIS-CRLE.sh > sample.sh
```

- 文字コードと改行コードを確認したい場合

インタラクティブ  
タイプノード  
側

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

# ジョブの実行

## — Spotサービス —

### バッチジョブの実行

`$qsub -l リソースタイプ = 数量 -g グループ名 実行スクリプトのパス`

コマンド例  
インタラクティブ  
ノード側

```
[abci username@es1 Sample] $cat sample.sh
```

```
#使用するシェルを指定します
```

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

```
#ジョブスケジューラのオプションを指定します
```

```
#$-cwd
```

```
#$-j y
```

```
#実行するプログラムを指定します。
```

```
date
```

```
hostname
```

```
echo "Hello, world"
```

```
[abci username@es1 Sample] $chmod u+x ./sample.sh
```

```
[abci username@es1 Sample] $ls -l
```

```
-rwxr--r-- 1 username usergroup 231 Jan 1 12:00 sample.sh
```

```
[abci username@es1 Sample] $qsub -l rt_G.small=1 -g gaa12345 ./sample.sh
```

```
Your job 12345678 ("sample.sh") has been submitted
```

実行スクリプトの確認

実行スクリプトに実行権を付ける  
実行権"x"を確認

リソースタイプを指定 所属グループ名を指定

# 実行したジョブの結果を確認

## — Spotサービス —

ジョブ終了後に出力されたファイルに実行結果がしるされている

コマンド例  
インタラクティブ  
ノード側

```
[abci username@es1 Sample] $ls -l
-rwxr--r- 1 username usergroup 231 Jan1 12:00 sample.sh
-rw-r--r- 1 username usergroup 59  Jan 1 12:00 sample.sh.o12345678 → ジョブの標準エラー出力ファイル
[abci username@es1 Sample] $
[abci username@es1 Sample] $cat sample.sh.o12345678
Sun Jan 1 12:00:00 JST 2023
g0009.abci.local
Hello, world
[abci username@es1 Sample] $
```

→ 計算ノードで実行した結果

# ジョブの実行

## — On-demandサービス —

### インタラクティブジョブの実行

`$qrrsh -l リソースタイプ = 数量 -g グループ名`

コマンド例

インタラクティブ  
ノード側

```
[abci username@es1] $qrrsh -l rt_G.small=1 -g gaa12345
[abci username@g0009.abci.ai] $
[abci username@g0009.abci.ai] $uname -n
g0009.abci.local
[abci username@g0009.abci.ai] $cd Sample
[abci username@g0009.abci.ai Sample] $ls -l
-rwxr--r-- 1 username usergroup 231 Jan 1 12:00 sample.sh
[abci username@g0009.abci.ai Sample] $./sample.sh
Sun Jan 1 12:00:00 JST 2023
g0009.abci.local
Hello, world
[abci username@g0009.abci.ai Sample] $
```

リソースタイプと所属グループを指定

計算ノードにログインしている状態

計算ノードで実行した結果

# ジョブの実行状態の確認

## コマンド書式

`$qstat` **-(オプションコード)**

オプションコード

`-r` ジョブのリソース情報を表示します  
`-j` ジョブに関する追加情報を表示します

コマンド例  
バッチジョブの場合

```
[abci username@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
```

コマンド例  
インタラクティブジョブ  
の場合

```
[abci username@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
```

# ジョブの実行の流れ

## — まとめ —

ABCIでジョブの実行は実はシンプル。是非試してみましょう。

① スクリプトファイルの作成

② スクリプトファイルのアップロード

```
$scp -P 10022 ローカルのファイル名 aaa12345xx@localhost:./###/リモートの転送先ファイル名
```

③ 文字コード・改行コードの変更

```
[abci username@es1] $nkf -Lu ./Sample_CFLR.sh > sample.sh
```

実行権の付与

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

④ ジョブの実行  
オプションでリソースタイプと  
グループ名を指定

```
[abci username@es1] $qsub -l rt_G.small=1 -g gaa12345 ./sample.sh
```

```
[abci username@es1] $qssh -l rt_G.small=1 -g gaa12345
```



# 参考リンク まとめ

- ABCI公式サイト (<https://abci.ai/ja/>)
- ABCI 2.0 User Guide (<https://docs.abci.ai/ja/>)  
ABCI 2.0の技術的詳細や利用方法について説明。
- WinSCP (<https://winscp.net>)

