

1.1.6 仮想サーバーインスタンス新環境移行（RHEL+Oracle DB）データ移行手順

つながろう。驚きを。幸せを。



■ 前提条件および注意事項

- 移行元に展開したミドルウェア、アプリケーション等は移行先に新規にインストールする必要があります。ミドルウェア、アプリケーション等の移行手順は、提供元のベンダーにお問合せください。
※Oracleについては、本手順にて移行先環境構築やDB移行を行います。
- 本手順では、データ移行にデータボリュームを利用します。
- Oracle Databaseデータベースの移行には「Oracle Data Pump」を使用します。
- 移行作業にかかる時間は、データの容量やネットワーク環境によって異なります。

1.1.6 仮想サーバーインスタンス新環境移行（RHEL+Oracle DB）データ移行手順

- 移行作業内容（Linux Server）
 1. 移行先サーバーの作成
 2. バックアップデータ保存用ボリュームの作成
 3. バックアップ作業
 4. バックアップデータ保存用ボリュームの移行
 5. リストア作業
 6. バックアップデータ保存用ボリューム・移行元サーバーの削除

1. 移行先サーバーの作成

以下リンクを参照して、移行先サーバーを作成します。

[2. Oracle Database 19c \(Linux版\)のご利用方法 - Oracle チュートリアル](#)

※チュートリアルを参考に、移行先サーバー作成に必要な手順を実施します。

SDPFの標準的な構成の場合は、「2.1. はじめに」～「2.6. データベース作成後の確認」までを実施します。

ネットワーク構成については、既存のお客様環境に合わせて設定してください。

2. バックアップデータ保存用ボリュームの作成

以下リンクを参照して、バックアップデータ保存用のボリュームを作成します。
データボリュームの新規作成

作成するデータボリュームは、以下の通りです

- ・ボリュームソース : ソースの指定なし(空のボリューム)
- ・サイズ : 移行元インスタンス作成時のデータボリュームの容量
- ・ボリューム種別 : Type-A

以下リンクの「4.1.2インスタンスへのデータボリュームのアタッチ」を参照して、バックアップデータ保存用のボリュームを移行元サーバーのインスタンスへ接続します。

[4.1.データボリュームの新規作成方法 - サーバーインスタンス チュートリアル](#)

※ボリュームのアタッチ/デタッチは仮想サーバーインスタンスを“停止”した状態で行ってください。

<https://sdpf.ntt.com/faq/virtual-server-35/>

2. バックアップデータ保存用ボリュームの作成

SDPFポータルで、[仮想サーバー]>[サーバーインスタンス]>[インスタンス]メニューを開きます。



2. バックアップデータ保存用ボリュームの作成

移行元サーバーを選択し、詳細画面を開き、[コンソール]タブを選択します。

The screenshot shows the NTT docomo Business Smart Data Platform interface. The top navigation bar includes the 'Smart Data Platform' logo, the 'docomo Business' logo, and a 'メニュー' (Menu) dropdown. The main content area is titled 'インスタンスの詳細:' (Instance Details). On the left, there is a sidebar menu with options: 'テナント情報' (Tenant Information), '仮想サーバー' (Virtual Server), '物理サーバー' (Physical Server), and 'ハイパーバイザー' (Hypervisor). Below the title, there are three tabs: '概要' (Overview), 'セキュリティグループ' (Security Group), and 'コンソール' (Console). The 'コンソール' tab is highlighted with a red rectangular border.

2. バックアップデータ保存用ボリュームの作成

移行元サーバーにログインし、下記コマンドを入力し、バックアップデータ保存用ボリュームが接続されていることを確認します。

```
# lsblk
```

```
# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
vda                  252:0    0   40G  0 disk
├─vda1                252:1    0   500M  0 part /boot
├─vda2                252:2    0  39.5G  0 part
│   └─rhel-var        253:0    0   10G  0 lvm  /var
│   └─rhel-root       253:1    0  21.5G  0 lvm  /
│   └─rhel-swap       253:2    0    8G  0 lvm  [SWAP]
vdb                  252:16    0    1T  0 disk
└─vdb1                252:17    0 1024G  0 part /data1
```

2. バックアップデータ保存用ボリュームの作成

下記コマンドを入力し、現在接続されているディスクのパーティションの状況（バックアップデータ保存用ボリュームのパーティションが存在しないこと）を確認します。

```
# fdisk -l
```

```
[root@ta-rhel9 ~]# fdisk -l
Disk /dev/uda: 80 GiB, 85899345920 bytes, 167772160 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: D209C89E-EA5E-4FBD-B161-B461CCE297E0

Device        Start      End    Sectors  Size Type
/dev/uda1     2048       4095     2048    1M BIOS boot
/dev/uda2     4096     413695  409600  200M EFI System
/dev/uda3     413696   1437695 1024000  500M Linux extended boot
/dev/uda4    1437696 167772126 166334431 79.3G Linux filesystem

Disk /dev/udb: 100 GiB, 107374182400 bytes, 209715200 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
[root@ta-rhel9 ~]#
```

2. バックアップデータ保存用ボリュームの作成

下記コマンドを入力し、現在のファイルシステムのディスクの使用状況（バックアップデータ保存用ボリュームが存在しないこと）を確認します。

```
# df -Th
```

```
[root@ta-rhel9 ~]# df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
deutmpfs        deutmpfs  4.0M   0    4.0M  0%  /dev
tmpfs           tmpfs     1.8G   0    1.8G  0%  /dev/shm
tmpfs           tmpfs     733M   17M  716M  3%  /run
/dev/uda4       xfs       80G    2.4G  77G   4%  /
/dev/uda3       xfs       495M   154M  341M  32% /boot
/dev/uda2       ufat      200M   7.0M  193M  4%  /efi
tmpfs           tmpfs     367M   0    367M  0%  /run/user/0
```

2. バックアップデータ保存用ボリュームの作成

下記コマンドを入力し、バックアップデータ保存用ボリュームに対してfdiskの対話モードを起動します。

```
# fdisk /dev/vdb
```

下記を入力し、新しいパーティションを作成します。

Select (default p) : 「n」を入力してください。

```
[root@ta-rhel9 ~]# fdisk /dev/vdb
Welcome to fdisk (util-linux 2.37.4).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x2bd93e26.

Command (m for help): n
```

2. バックアップデータ保存用ボリュームの作成

パーティションタイプを選択します。

Select (default p) : p

※本手順ではデフォルトの「p」を入力します。

パーティション番号を設定します。

Partition number(1-4,default 1) : 1を入力してください

※本手順ではデフォルトの「1」を入力します。

パーティション値を設定します。

First sector : 何も入力せず、そのままEnter

Last sector : 何も入力せず、そのままEnter

```
Partition type
 p  primary (0 primary, 0 extended, 4 free)
 e  extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-209715199, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-209715199, default 209715199):
```

新しいパーティションが作成されたことを確認します。

```
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-209715199, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-209715199, default 209715199):
Created a new partition 1 of type 'Linux' and of size 100 GiB.
```

2. バックアップデータ保存用ボリュームの作成

パーティションのタイプを変更していきます。

Command (m for help) : 「t」を入力してください。

システムコード (Linux LVM) を選択します。

Hex code or alias (type L to list all) : 「8e」を入力してください。

```
Created a new partition 1 of type 'Linux' and of size 100 GiB.
```

```
Command (m for help): t
Selected partition 1
Hex code or alias (type L to list all): 8e
Changed type of partition 'Linux' to 'Linux LVM'.
```

パーティションのタイプが変更されたことを確認します。

Command (m for help) : 「p」を入力してください。

```
Command (m for help): p
Disk /dev/udb: 100 GiB, 107374182400 bytes, 209715200 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x2bd93e26

Device      Boot Start      End  Sectors  Size Id Type
/dev/udb1           2048 209715199 209713152  100G 8e Linux LVM
```

2. バックアップデータ保存用ボリュームの作成

変更内容を保存します。

Command (m for help) : 「w」を入力してください。

```
Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

2. バックアップデータ保存用ボリュームの作成

下記コマンドを入力し、バックアップデータ保存用ボリュームのパーティションが追加されたことを確認します。

```
# fdisk -l
```

```
[root@ta-rhel9 ~]# fdisk -l
Disk /dev/uda: 80 GiB, 85899345920 bytes, 167772160 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: D209C89E-EA5E-4FBD-B161-B461CCE297E0

Device            Start          End      Sectors  Size Type
/dev/uda1          2048           4095      2048     1M BIOS boot
/dev/uda2          4096          413695   409600   200M EFI System
/dev/uda3          413696        1437695  1024000  500M Linux extended boot
/dev/uda4          1437696       167772126 166334431 79.3G Linux filesystem

Disk /dev/udb: 100 GiB, 107374182400 bytes, 209715200 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x2bd93e26

Device            Boot Start          End      Sectors  Size Id Type
/dev/udb1          2048 209715199 209713152 100G 8e Linux LVM
```

2. バックアップデータ保存用ボリュームの作成

下記コマンドを入力し、上記で作成したパーティションにファイルシステムを作成します。

```
# mkfs -t xfs /dev/vdb1
```

```
[root@ta-rhel9 ~]# mkfs -t xfs /dev/vdb1
meta-data=/dev/vdb1          isize=512    agcount=4, agsize=6553536 blks
           =                  sectsz=512   attr=2, projid32bit=1
           =                  crc=1        finobt=1, sparse=1, rmapbt=0
           =                  reflink=1   bigtime=1 inobtcount=1
data      =                  bsize=4096  blocks=26214144, imaxpct=25
           =                  sunit=0     swidth=0 blks
naming    =version 2        bsize=4096  ascii-ci=0, ftype=1
log       =internal log    bsize=4096  blocks=12799, version=2
           =                  sectsz=512   sunit=0 blks, lazy-count=1
realtime  =none            extsz=4096  blocks=0, rtextents=0
```

マウントポイント用のディレクトリを作成していきます。下記コマンドを入力し、ディレクトリを移動します。

```
# pwd
# cd /
# pwd
```

```
[root@ta-rhel9 ~]# pwd
/root
[root@ta-rhel9 ~]#
[root@ta-rhel9 ~]# cd /
[root@ta-rhel9 /]#
[root@ta-rhel9 /]# pwd
/
```

2. バックアップデータ保存用ボリュームの作成

下記コマンドを入力し、現在のディレクトリ内の状況を確認します。

```
# ll
```

```
[root@ta-rhel9 /]# ll
total 24
dr-xr-xr-x.  2 root root    6 Aug  9  2021 afs
lrwxrwxrwx.  1 root root    7 Aug  9  2021 bin -> usr/bin
dr-xr-xr-x.  5 root root 4096 May 23 05:08 boot
drwxr-xr-x. 18 root root 3140 Jun  6 02:12 dev
drwxr-xr-x.  2 root root    0 Jun  6 02:06 efi
drwxr-xr-x. 92 root root 8192 Jun  6 02:04 etc
drwxr-xr-x.  3 root root   24 May 23 05:07 home
lrwxrwxrwx.  1 root root    7 Aug  9  2021 lib -> usr/lib
lrwxrwxrwx.  1 root root    9 Aug  9  2021 lib64 -> usr/lib64
drwxr-xr-x.  2 root root    6 Aug  9  2021 media
drwxr-xr-x.  2 root root    6 Aug  9  2021 mnt
drwxr-xr-x.  2 root root    6 Aug  9  2021 opt
dr-xr-xr-x. 165 root root    0 Jun  6 02:06 proc
dr-xr-x---.  3 root root  140 Jun  4 07:50 root
drwxr-xr-x. 32 root root   940 Jun  6 02:06 run
lrwxrwxrwx.  1 root root    8 Aug  9  2021 sbin -> usr/sbin
drwxr-xr-x.  2 root root    6 Aug  9  2021 srv
dr-xr-xr-x. 13 root root    0 Jun  6 02:06 sys
drwxrwxrwt.  9 root root 4096 Jun  6 02:07 tmp
drwxr-xr-x. 12 root root   144 May  3  2023 usr
drwxr-xr-x. 19 root root 4096 May 23 05:07 var
```

下記コマンドを入力し、マウントポイント用のディレクトリを作成します。

```
# mkdir data1
```

```
[root@ta-rhel9 /]# mkdir data1
```

2. バックアップデータ保存用ボリュームの作成

下記コマンドを入力し、マウントポイント用のディレクトリが作成されたことを確認します。

//

```
[root@ta-rhel9 ~]# ll
total 24
dr-xr-xr-x.  2 root root    6 Aug  9 2021 afs
lrwxrwxrwx.  1 root root    7 Aug  9 2021 bin -> usr/bin
dr-xr-xr-x.  5 root root 4096 May 23 05:08 boot
drwxr-xr-x  2 root root    6 Jun  6 02:28 data1
drwxr-xr-x 18 root root 3140 Jun  6 02:12 dev
drwxr-xr-x  2 root root    0 Jun  6 02:06 efi
drwxr-xr-x 92 root root 8192 Jun  6 02:04 etc
drwxr-xr-x  3 root root   24 May 23 05:07 home
lrwxrwxrwx.  1 root root    7 Aug  9 2021 lib -> usr/lib
lrwxrwxrwx.  1 root root    9 Aug  9 2021 lib64 -> usr/lib64
drwxr-xr-x  2 root root    6 Aug  9 2021 media
drwxr-xr-x  2 root root    6 Aug  9 2021 mnt
drwxr-xr-x  2 root root    6 Aug  9 2021 opt
dr-xr-xr-x 165 root root    0 Jun  6 02:06 proc
dr-xr-x---  3 root root   140 Jun  4 07:50 root
drwxr-xr-x 32 root root   940 Jun  6 02:06 run
lrwxrwxrwx.  1 root root    8 Aug  9 2021 sbin -> usr/sbin
drwxr-xr-x  2 root root    6 Aug  9 2021 sru
dr-xr-xr-x 13 root root    0 Jun  6 02:06 sys
drwxrwxrwt.  9 root root 4096 Jun  6 02:07 tmp
drwxr-xr-x 12 root root   144 May  3 2023 usr
drwxr-xr-x 19 root root 4096 May 23 05:07 var
```

2. バックアップデータ保存用ボリュームの作成

下記コマンドを入力し、元のディレクトリに戻ります。

```
# cd  
# pwd
```

```
[root@ta-rhel9 ~]# cd  
[root@ta-rhel9 ~]#  
[root@ta-rhel9 ~]# pwd  
/root
```

下記コマンドを入力し、バックアップデータ保存用ボリュームを、マウントポイント用のディレクトリへマウントします。

```
# mount デバイス マウントポイント
```

```
[root@ta-rhel9 ~]# mount /dev/vdb1 /data1
```

2. バックアップデータ保存用ボリュームの作成

下記コマンドを入力し、マウントされたことを確認します。

```
# df -Th
```

```
[root@ta-rhel9 ~]# df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
devtmpfs        devtmpfs  4.0M   0    4.0M  0% /dev
tmpfs           tmpfs     1.8G   0    1.8G  0% /dev/shm
tmpfs           tmpfs     733M   17M  716M  3% /run
/dev/uda4       xfs       80G    2.4G  77G   4% /
/dev/uda3       xfs       495M   154M  341M  32% /boot
/dev/uda2       ufat      200M   7.0M  193M  4% /boot/efi
tmpfs           tmpfs     367M   0    367M  0% /run/user/0
/dev/udb1       xfs       100G   746M  100G  1% /data1
```

3. バックアップ作業

LinuxOSのデータ

3. バックアップ作業

移行元サーバーで、バックアップデータを作成します。

「id ユーザー名」で出力される、移行するユーザーのID情報（ユーザーID・グループID・所属グループ）を控えてください。
※ 控えた情報は、後続「5. リストア作業(44ページ)」にて利用します。

コマンド例 : # id test

```
[root@rhel1818-ki ~]# id test
uid=1000(test) gid=1000(test) groups=1000(test)
```

以下のコマンドで、移行したいデータディレクトリをコピーします。

cp -auv (コピー元ディレクトリ) (コピー先ディレクトリ)

コマンド例 : cp -auv /home/test/migration_data /data1/
-a 元ファイルの属性を保持
-u 保存先ファイルより新しい、ファイルが存在しない場合にのみファイルをコピー
-v 実行内容を表示

```
[root@rhel1818-ki ~]# cp -auv /home/test/migration_data /data1
'/home/test/migration_data' -> '/mnt/data1/migration_data'
'/home/test/migration_data/testtext1.txt' -> '/mnt/data1/migration_data/testtext1.txt'
'/home/test/migration_data/testtext2.txt' -> '/mnt/data1/migration_data/testtext2.txt'
```

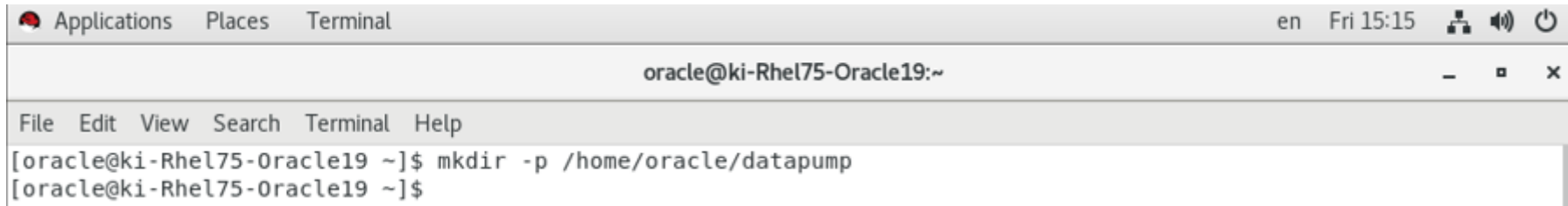
3. バックアップ作業

OracleDBのデータ

3. バックアップ作業

移行対象のデータの保存先となるディレクトリを作成します。

コマンド例 : `$ mkdir -p /home/oracle/datapump`

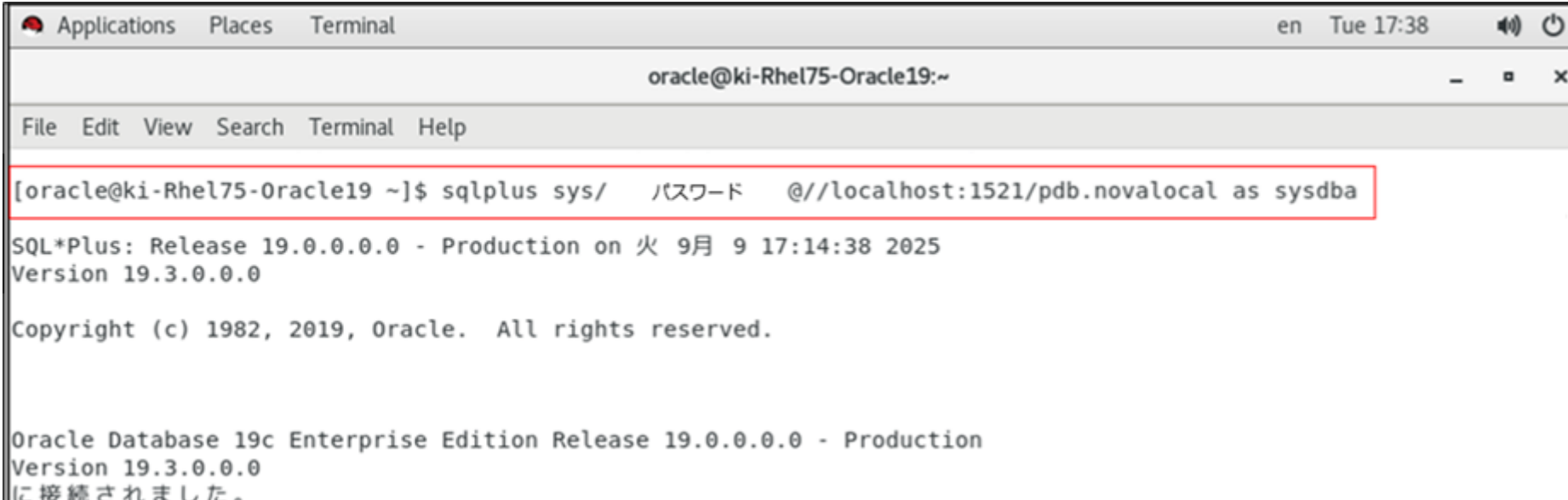


```
Applications Places Terminal en Fri 15:15 [oracle@ki-Rhel75-Oracle19:~]
oracle@ki-Rhel75-Oracle19:~
File Edit View Search Terminal Help
[oracle@ki-Rhel75-Oracle19 ~]$ mkdir -p /home/oracle/datapump
[oracle@ki-Rhel75-Oracle19 ~]$
```

3. バックアップ作業

以下のコマンドで、sysユーザーでデータベースに接続します。

```
$ sqlplus ユーザー名/パスワード@//ホスト名(localhost または ホスト名 または IPアドレス):1521/移行対象データのあるPDB as sysdba
```



```
Applications Places Terminal en Tue 17:38
oracle@ki-Rhel75-Oracle19:~
File Edit View Search Terminal Help
[oracle@ki-Rhel75-Oracle19 ~]$ sqlplus sys/ パスワード @//localhost:1521/pdb.novalocal as sysdba
SQL*Plus: Release 19.0.0.0.0 - Production on 火 9月 9 17:14:38 2025
Version 19.3.0.0.0
Copyright (c) 1982, 2019, Oracle. All rights reserved.
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
に接続されました。
```

3. バックアップ作業

以下のコマンドで、指定したパスへのディレクトリ・オブジェクトを作成します。

SQL> CREATE OR REPLACE DIRECTORY ディレクトリ名 AS 'OS上の絶対パス';

```
SQL> CREATE OR REPLACE DIRECTORY MIGRATION_DIR AS '/home/oracle/datapump';
```

ディレクトリが作成されました。

以下のコマンドで、ユーザーにディレクトリへの読み書き権限を付与します。

SQL> GRANT READ, WRITE ON DIRECTORY ディレクトリ名 TO Oracleデータベースで作成したユーザー名;

※pumpファイルの出力等を行うため、任意のユーザに権限を付与します。

本手順では、既存ユーザー（testuser）に対して、権限を付与しております。

```
SQL> GRANT READ, WRITE ON DIRECTORY MIGRATION_DIR TO testuser;
```

権限付与が成功しました。

3. バックアップ作業

以下のコマンドで、ユーザーにデータポンプのエクスポート（expdb）の実行権限を付与します。

```
SQL> GRANT DATAPUMP_EXP_FULL_DATABASE TO ユーザー名;
```

```
SQL> GRANT DATAPUMP_EXP_FULL_DATABASE TO testuser;
```

```
権限付与が成功しました。
```

以下のコマンドで、ユーザーにデータポンプのインポート（impdb）の実行権限を付与します。完了したら、exitします。

```
SQL> GRANT DATAPUMP_IMP_FULL_DATABASE TO ユーザー名;
```

```
SQL> exit;
```

```
SQL> GRANT DATAPUMP_IMP_FULL_DATABASE TO testuser;
```

```
権限付与が成功しました。
```

```
SQL> exit
```

3. バックアップ作業

以下コマンドで、DBをエクスポートします。

```
$ expdp ユーザー名/パスワード@//ホスト名(localhost または ホスト名 または IPアドレス):1521/移行対象データのあるPDB SCHEMAS=エクスポートしたいスキーマ DIRECTORY=ディレクトリ名 DUMPFILE=作成したいファイル名.dmp LOGFILE=作成したいログファイル名.log
```

```
[oracle@ ~]$  
[oracle@ ~]$  
[oracle@ ~]$ expdp testuser/ //localhost:1521/pdb.example.com SCHEMAS=testuser DUMPFILE=testuser_export.dmp LOGFILE=testuser_export.log DIRECTORY=MIGRATION_DIR  
ON_DIR  
Export: Release 19.0.0.0.0 - Production on 月 1月 26 19:06:04 2026  
Version 19.3.0.0.0  
Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.  
接続先: Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production  
"TESTUSER"."SYS_EXPORT_SCHEMA_01"を起動しています: testuser/*****@//localhost:1521/pdb.example.com SCHEMAS=testuser DUMPFILE=testuser_export.dmp LOGFILE=testuser_export.log  
og DIRECTORY=MIGRATION_DIR  
オブジェクト型 SCHEMA_EXPORT/TABLE/TABLE_DATAの処理中です  
オブジェクト型 SCHEMA_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICSの処理中です  
オブジェクト型 SCHEMA_EXPORT/TABLE/STATISTICS/TABLE_STATISTICSの処理中です  
オブジェクト型 SCHEMA_EXPORT/STATISTICS/MARKERの処理中です  
オブジェクト型 SCHEMA_EXPORT/USERの処理中です  
オブジェクト型 SCHEMA_EXPORT/SYSTEM_GRANTの処理中です  
オブジェクト型 SCHEMA_EXPORT/ROLE_GRANTの処理中です  
オブジェクト型 SCHEMA_EXPORT/DEFAULT_ROLEの処理中です  
オブジェクト型 SCHEMA_EXPORT/TABLESPACE_QUOTAの処理中です  
オブジェクト型 SCHEMA_EXPORT/PRE_SCHEMA/PROCACT_SCHEMAの処理中です  
オブジェクト型 SCHEMA_EXPORT/TABLE/TABLEの処理中です  
オブジェクト型 SCHEMA_EXPORT/TABLE/COMMENTの処理中です  
オブジェクト型 SCHEMA_EXPORT/TABLE/INDEX/INDEXの処理中です  
オブジェクト型 SCHEMA_EXPORT/TABLE/CONSTRAINT/CONSTRAINTの処理中です  
... "TESTUSER"."EMP_TABLE" 5.976 KB 1行がエクスポートされました  
マスター表 "TESTUSER"."SYS_EXPORT_SCHEMA_01"は正常にロード/アンロードされました  
*****  
TESTUSER.SYS_EXPORT_SCHEMA_01に設定されたダンプ・ファイルは次のとおりです:  
/home/oracle/datapump/testuser_export.dmp  
ジョブ "TESTUSER"."SYS_EXPORT_SCHEMA_01"が月 1月 26 19:07:08 2026 elapsed 0 00:00:59で正常に完了しました
```

出力したファイルはSCPコマンド等でバックアップデータ保存用ボリュームに転送します。

3. バックアップ作業

バックアップデータ保存用ストレージ上にバックアップデータが正しく移行されていることを確認して、バックアップ作業は完了です。

4. バックアップデータ保存用ボリュームの移行

下記コマンドを入力し、バックアップデータ保存用ボリュームをアンマウントします。

```
# umount /data1
```

下記コマンドを入力し、アンマウントされたことを確認します。

```
# df -Th
```

```
[root@20250804-ky-rhel7 ~]# df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
devtmpfs        devtmpfs  920M   0  920M   0% /dev
tmpfs           tmpfs     936M   0  936M   0% /dev/shm
tmpfs           tmpfs     936M  9.0M  927M   1% /run
tmpfs           tmpfs     936M   0  936M   0% /sys/fs/cgroup
/dev/mapper/rhel-root xfs       22G   15G   6.9G  68% /
/dev/mapper/rhel-var  xfs       10G   2.3G   7.8G  23% /var
/dev/vdb1        xfs       1.0T   33M   1.0T   1% /data1
/dev/vda1        xfs       497M  158M   340M  32% /boot
tmpfs           tmpfs     188M   0   188M   0% /run/user/0
```

4. バックアップデータ保存用ボリュームの移行

SDPFポータル画面を開き、「仮想サーバー」→「サーバーインスタンス」→「ボリューム」画面で、バックアップデータ保存用ボリュームを選択し、「接続の管理」を選択します。

※「接続の管理」からボリュームの切断（デタッチ）を行います。下記URL事象が発生する可能性があるため、デタッチ実施の際は、仮想サーバーインスタンスを停止した状態で実施してください。

<https://sdpf.ntt.com/faq/virtual-server-35/>

名前	説明	ゾーングループ	サイズ	種別	接続先	起動可能	ステータス	アクション
<input type="checkbox"/>	-	zone1-group	100GB	-		はい	利用可能	ボリュームの編集
<input type="checkbox"/>	-	zone1-group	80GB	Type-A		はい	reserved	ボリュームの編集
<input type="checkbox"/>	-	zone1-group	80GB	Type-B		はい	利用可能	ボリュームの編集
<input type="checkbox"/>	-	zone1-group	1024GB	Type-B		はい	利用可能	ボリュームの編集
<input type="checkbox"/>	-	zone1-group	1024GB	Type-B	の /dev/sdi に接続中	いいえ	使用中	ボリュームの編集
<input type="checkbox"/>	-	zone1-group	1024GB	Type-B	の /dev/sdi に接続中	いいえ	使用中	接続の管理
<input type="checkbox"/>	-	zone1-group	1024GB	Type-B	の /dev/sdi に接続中	いいえ	使用中	ボリュームの切断
<input type="checkbox"/>	-	zone1-group	1024GB	Type-B	の /dev/sdi に接続中	いいえ	使用中	イメージをアップロード
<input type="checkbox"/>	-	zone1-group	1024GB	Type-B	の /dev/sdi に接続中	いいえ	使用中	ボリューム種別の変更
<input type="checkbox"/>	-	zone1-group	1024GB	Type-B	の /dev/sdi に接続中	いいえ	使用中	ボリュームの削除

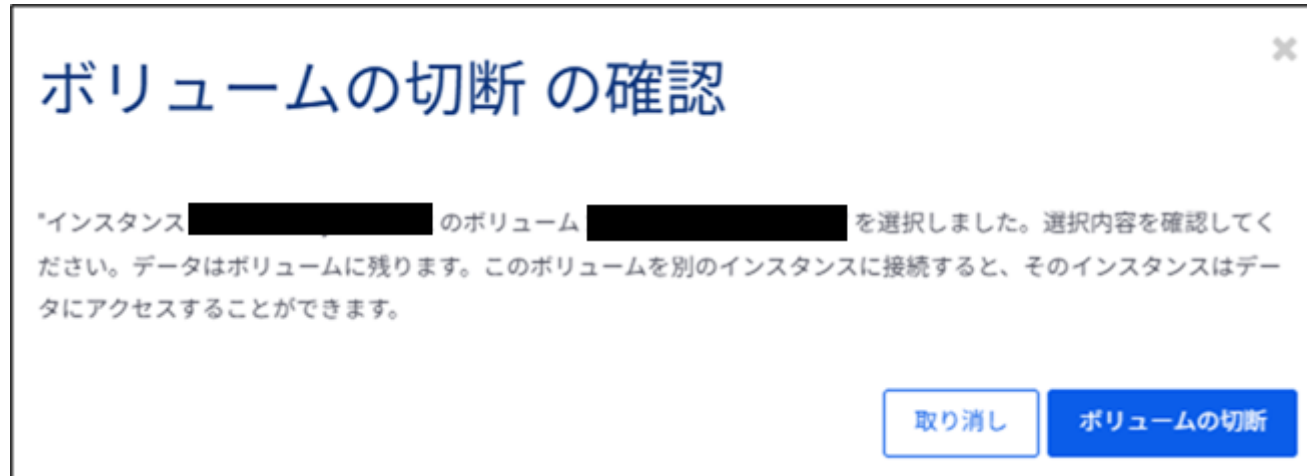
4. バックアップデータ保存用ボリュームの移行

「ボリュームの接続の管理」画面で、「ボリュームの切断」を押下します。



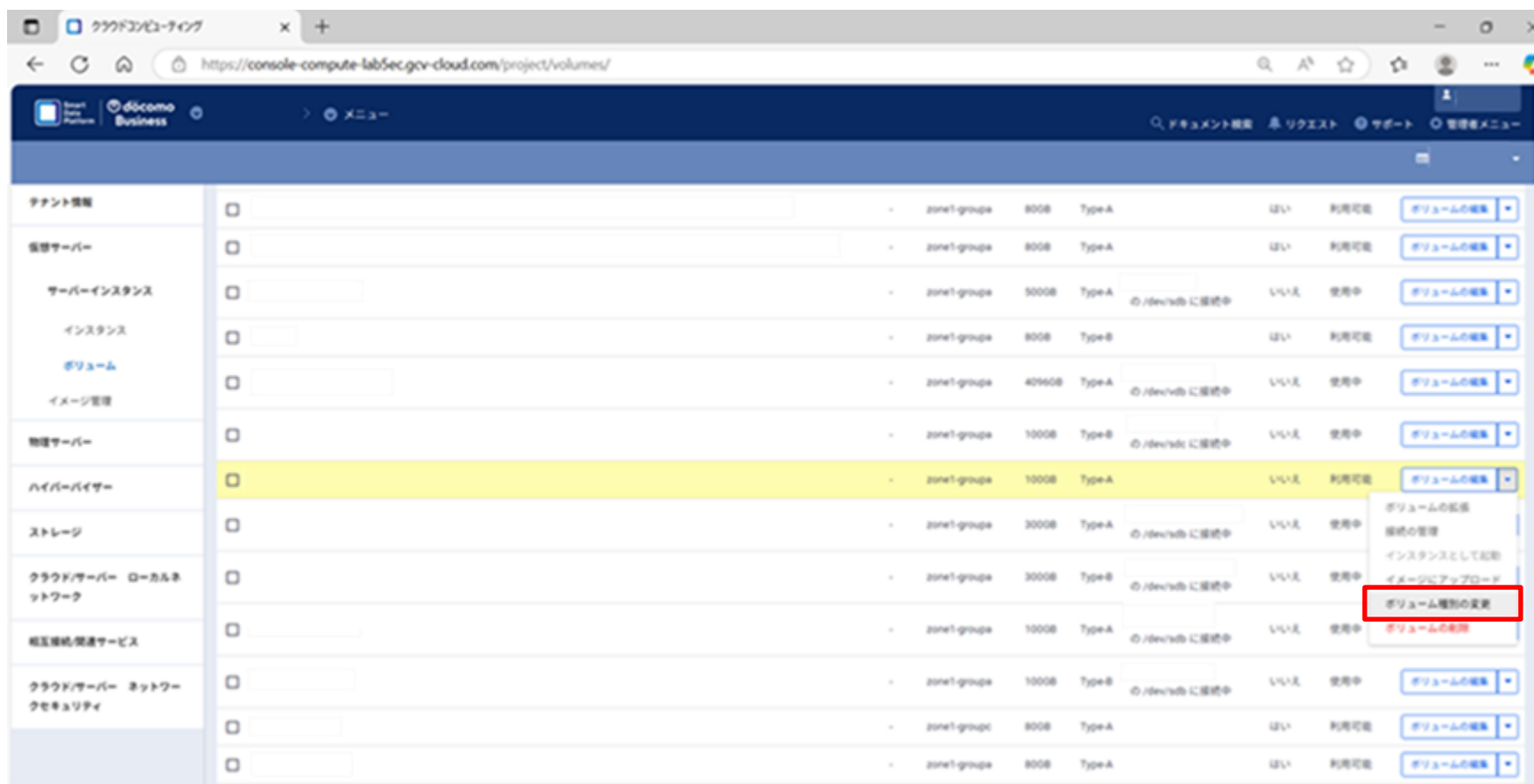
4. バックアップデータ保存用ボリュームの移行

「ボリュームの切断の確認」画面で、「ボリュームの切断」を押下します。



4. バックアップデータ保存用ボリュームの移行

先ほど作成したバックアップデータ保存用ボリュームを選択し、アクションリストから「ボリューム種別の変更」を選択します。



The screenshot shows the NTT Docomo Business console interface. A table lists various volumes with columns for name, zone, size, type, and status. The 'ボリューム種別の変更' (Change Volume Type) option is highlighted in the dropdown menu for a selected volume.

テナント情報	名前	ゾーン	サイズ	タイプ	接続状況	ステータス	アクション	
		zone1-group	800B	Type-A	はい	利用可能	ボリュームの編集	
		zone1-group	800B	Type-A	はい	利用可能	ボリュームの編集	
		zone1-group	5000B	Type-A	の /dev/sdb に接続中	いいえ	使用中	ボリュームの編集
		zone1-group	800B	Type-B	はい	利用可能	ボリュームの編集	
		zone1-group	40960B	Type-A	の /dev/sdb に接続中	いいえ	使用中	ボリュームの編集
		zone1-group	1000B	Type-B	の /dev/sdb に接続中	いいえ	使用中	ボリュームの編集
		zone1-group	1000B	Type-A	いいえ	利用可能	ボリュームの編集	
		zone1-group	3000B	Type-A	の /dev/sdb に接続中	いいえ	使用中	ボリュームの編集
		zone1-group	3000B	Type-B	の /dev/sdb に接続中	いいえ	使用中	ボリュームの編集
		zone1-group	1000B	Type-A	の /dev/sdb に接続中	いいえ	使用中	ボリュームの編集
		zone1-group	1000B	Type-B	の /dev/sdb に接続中	いいえ	使用中	ボリュームの編集
		zone1-group	800B	Type-A	はい	利用可能	ボリュームの編集	
		zone1-group	800B	Type-A	はい	利用可能	ボリュームの編集	

4. バックアップデータ保存用ボリュームの移行

ボリューム種別がType-Bになっていることを確認し、「ボリューム種別の変更」を選択します。

ボリューム種別の変更 ×

ボリューム種別

Type-B ▼

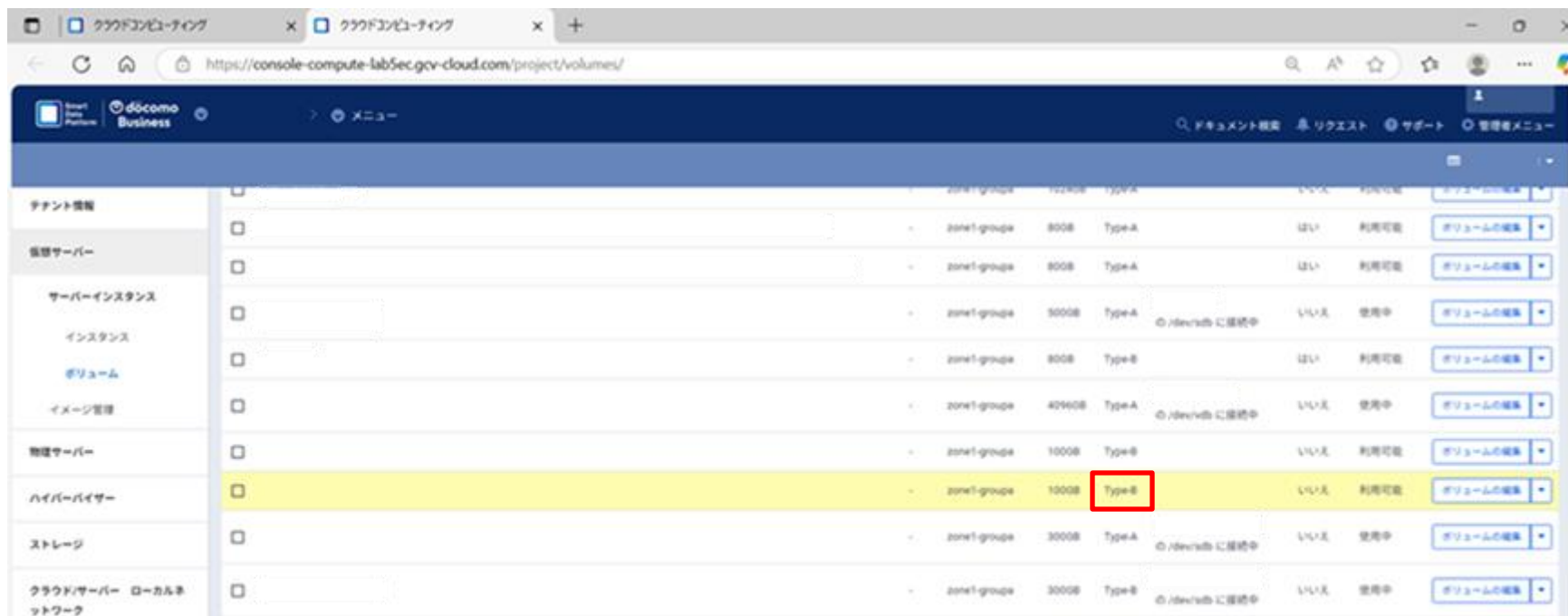
ドライランモードを有効にする

取り消し

ボリューム種別の変更

4. バックアップデータ保存用ボリュームの移行

バックアップデータ保存用ボリュームのタイプが“Type-B”になっていることを確認します。



テナント情報	ゾーン	容量	タイプ	接続状況	ステータス	操作	
	zone1-group	1024GB	Type-A	いいえ	利用可能	ボリュームの編集	
	zone1-group	800GB	Type-A	はい	利用可能	ボリュームの編集	
	zone1-group	800GB	Type-A	はい	利用可能	ボリュームの編集	
サーバーインスタンス							
インスタンス	zone1-group	3000GB	Type-A	の /dev/sdb に接続中	いいえ	使用中	ボリュームの編集
ボリューム	zone1-group	800GB	Type-B		はい	利用可能	ボリュームの編集
イメージ管理	zone1-group	4096GB	Type-A	の /dev/sdb に接続中	いいえ	使用中	ボリュームの編集
物理サーバー	zone1-group	1000GB	Type-B		いいえ	利用可能	ボリュームの編集
ハイパーバイザー	zone1-group	1000GB	Type-B		いいえ	利用可能	ボリュームの編集
ストレージ	zone1-group	3000GB	Type-A	の /dev/sdb に接続中	いいえ	使用中	ボリュームの編集
クラウド/サーバー ネットワーク	zone1-group	3000GB	Type-B	の /dev/sdb に接続中	いいえ	使用中	ボリュームの編集

4. バックアップデータ保存用ボリュームの移行

次に、バックアップデータ保存用ボリュームを移行先サーバーに接続します。
以下リンクの「4.1.2インスタンスへのデータボリュームのアタッチ」を参考に実施します。

[4.1.データボリュームの新規作成方法 - サーバーインスタンス チュートリアル](#)

※ボリュームのアタッチ/デタッチは仮想サーバーインスタンスを“停止”した状態で行ってください。
<https://sdpf.ntt.com/faq/virtual-server-35/>

4. バックアップデータ保存用ボリュームの移行

SDPFポータルで、[仮想サーバー]>[サーバーインスタンス]>[インスタンス]メニューを開きます。



4. バックアップデータ保存用ボリュームの移行

移行先サーバーを選択し、詳細画面を開き、[コンソール]タブを選択します。

The screenshot shows the 'Smart Data Platform' interface for 'docomo Business'. The main content area is titled 'インスタンスの詳細:' (Instance Details). On the left, there is a sidebar menu with options: 'テナント情報' (Tenant Information), '仮想サーバー' (Virtual Server), '物理サーバー' (Physical Server), and 'ハイパーバイザー' (Hypervisor). Below the title, there are three tabs: '概要' (Overview), 'セキュリティグループ' (Security Group), and 'コンソール' (Console). The 'コンソール' tab is highlighted with a red rectangular box.

4. バックアップデータ保存用ボリュームの移行

移行先サーバーにログインし、マウントポイント用のディレクトリを作成していきます。
下記コマンドを入力し、バックアップデータ保存用ボリュームが接続されていることを確認します。

```
# lsblk
```

```
File Edit View Search Terminal Help
[root@20250804-ky-rhel8 ~]# lsblk
NAME MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
vda   252:0    0   40G  0 disk
├─vda1 252:1    0    1M  0 part
├─vda2 252:2    0  100M  0 part /boot/efi
├─vda3 252:3    0 39.9G  0 part /
vdb   252:16   0    1T   0 disk
└─vdb1 252:17   0 1024G  0 part /data1
vdc   252:32   0    1T   0 disk
└─vdc1 252:33   0 1024G  0 part
```

下記コマンドを入力し、ディレクトリを移動します。

```
# pwd
# cd /
# pwd
```

```
[root@ta-rhel19 ~]# pwd
/root
[root@ta-rhel19 ~]#
[root@ta-rhel19 ~]# cd /
[root@ta-rhel19 /]#
[root@ta-rhel19 /]# pwd
/
```

4. バックアップデータ保存用ボリュームの移行

下記コマンドを入力し、現在のディレクトリ内の状況を確認します。

//

```
[root@ta-rhel9 ~]# ll
total 24
dr-xr-xr-x.  2 root root   6 Aug  9  2021 afs
lrwxrwxrwx.  1 root root   7 Aug  9  2021 bin -> usr/bin
dr-xr-xr-x.  5 root root 4096 May 23 05:08 boot
drwxr-xr-x  18 root root 3140 Jun  6 02:12 dev
drwxr-xr-x   2 root root   0 Jun  6 02:06 efi
drwxr-xr-x. 92 root root 8192 Jun  6 02:04 etc
drwxr-xr-x.  3 root root  24 May 23 05:07 home
lrwxrwxrwx.  1 root root   7 Aug  9  2021 lib -> usr/lib
lrwxrwxrwx.  1 root root   9 Aug  9  2021 lib64 -> usr/lib64
drwxr-xr-x.  2 root root   6 Aug  9  2021 media
drwxr-xr-x.  2 root root   6 Aug  9  2021 mnt
drwxr-xr-x.  2 root root   6 Aug  9  2021 opt
dr-xr-xr-x 165 root root   0 Jun  6 02:06 proc
dr-xr-x---.  3 root root  140 Jun  4 07:50 root
drwxr-xr-x  32 root root  940 Jun  6 02:06 run
lrwxrwxrwx.  1 root root   8 Aug  9  2021 sbin -> usr/sbin
drwxr-xr-x.  2 root root   6 Aug  9  2021 srv
dr-xr-xr-x  13 root root   0 Jun  6 02:06 sys
drwxrwxrwt.  9 root root 4096 Jun  6 02:07 tmp
drwxr-xr-x.  12 root root  144 May  3  2023 usr
drwxr-xr-x.  19 root root 4096 May 23 05:07 var
[root@ta-rhel9 ~]#
```

4. バックアップデータ保存用ボリュームの移行

下記コマンドを入力し、マウントポイント用のディレクトリを作成します。

```
# mkdir data1
```

```
[root@ta-rhel9 /]# mkdir data1
```

下記コマンドを入力し、マウントポイント用のディレクトリが作成されたことを確認します。

```
# ll
```

```
[root@ta-rhel9 /]# ll
total 24
dr-xr-xr-x.  2 root root    6 Aug  9 2021 afs
lrwxrwxrwx.  1 root root    7 Aug  9 2021 bin -> usr/bin
dr-xr-xr-x.  5 root root 4096 May 23 05:08 boot
drwxr-xr-x   2 root root    6 Jun  6 02:28 data1
drwxr-xr-x  18 root root 3140 Jun  6 02:12 dev
drwxr-xr-x   2 root root    0 Jun  6 02:06 efi
drwxr-xr-x. 92 root root 8192 Jun  6 02:04 etc
drwxr-xr-x.  3 root root   24 May 23 05:07 home
lrwxrwxrwx.  1 root root    7 Aug  9 2021 lib -> usr/lib
lrwxrwxrwx.  1 root root    9 Aug  9 2021 lib64 -> usr/lib64
drwxr-xr-x.  2 root root    6 Aug  9 2021 media
drwxr-xr-x.  2 root root    6 Aug  9 2021 mnt
drwxr-xr-x.  2 root root    6 Aug  9 2021 opt
dr-xr-xr-x 165 root root    0 Jun  6 02:06 proc
dr-xr-x---.  3 root root   140 Jun  4 07:50 root
drwxr-xr-x  32 root root   940 Jun  6 02:06 run
lrwxrwxrwx.  1 root root    8 Aug  9 2021 sbin -> usr/sbin
drwxr-xr-x.  2 root root    6 Aug  9 2021 sru
dr-xr-xr-x  13 root root    0 Jun  6 02:06 sys
drwxrwxrwt.  9 root root 4096 Jun  6 02:07 tmp
drwxr-xr-x. 12 root root   144 May  3 2023 usr
drwxr-xr-x. 19 root root 4096 May 23 05:07 var
```

4. バックアップデータ保存用ボリュームの移行

下記コマンドを入力し、元のディレクトリに戻ります。

```
# cd  
# pwd
```

```
[root@ta-rhel9 ~]# cd  
[root@ta-rhel9 ~]#  
[root@ta-rhel9 ~]# pwd  
/root
```

4. バックアップデータ保存用ボリュームの移行

下記コマンドを入力し、バックアップデータ保存用ボリュームを、マウントポイント用のディレクトリへマウントします。

```
# mount (デバイス) (マウントポイント)
```

下記コマンドを入力し、マウントされたことを確認します。

```
# df -Th
```

```
[root@20250804-ky-rhel8 ~]# mount /dev/vdc1 /backupdata
[root@20250804-ky-rhel8 ~]# df -Th
```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
devtmpfs	devtmpfs	3.7G	0	3.7G	0%	/dev
tmpfs	tmpfs	3.8G	0	3.8G	0%	/dev/shm
tmpfs	tmpfs	3.8G	18M	3.8G	1%	/run
tmpfs	tmpfs	3.8G	0	3.8G	0%	/sys/fs/cgroup
/dev/vda3	xfs	40G	17G	23G	43%	/
/dev/vda2	vfat	100M	5.8M	95M	6%	/boot/efi
/dev/vdb1	xfs	1.0T	7.2G	1017G	1%	/data1
tmpfs	tmpfs	770M	20K	770M	1%	/run/user/0
/dev/vdc1	xfs	1.0T	27G	997G	3%	/backupdata

5. リストア作業

つながろう。驚きを。幸せを。



LinuxOSのデータ

5. リストア作業

移行先サーバーで、リストア作業を実施します。

移行先となるユーザーを作成します。

```
コマンド # groupadd -g (GID) (グループ名)
          # useradd -u (UID) -g (GID) (ユーザー名)
          # passwd (ユーザー名)
```

※「3. バックアップ作業(20ページ)」で控えた移行するユーザーのID情報(ユーザーID・グループID・所属グループ)を指定してください。
移行先サーバーの作成にて、移行先となるユーザーも作成済の場合、本手順の実施は不要です。

```
[root@rhel194-ki ~]# groupadd -g 1000 test
[root@rhel194-ki ~]# useradd -u 1000 -g 1000 test
[root@rhel194-ki ~]# passwd test
Changing password for user test.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@rhel194-ki ~]#
```

「mkdir ディレクトリ」で、移行先となるユーザーディレクトリ配下に移行データ用ディレクトリを作成します。

コマンド例 : `mkdir /home/test/migration_data_from_oldrhel`

```
[root@rhel194-ki ~]# mkdir /home/test/migration_data_from_oldrhel
```

5. リストア作業

以下のコマンドで、バックアップデータを移行先ディレクトリにコピーします。

`cp -auv` (コピー元ディレクトリ) (コピー先ディレクトリ)

コマンド例 : `cp -auv /data1 /home/test/migration_data_from_oldrhel`
-a 元ファイルの属性を保持
-u 保存先ファイルより新しい、ファイルが存在しない場合にのみファイルをコピー
-v 実行内容を表示

```
[root@rhel194-ki ~]# cp -auv /mnt/data1 /home/test/migration_data_from_oldrhel/  
'/data1' -> '/home/test/migration_data_from_oldrhel/migration_data'  
'/data1/testtext1.txt' -> '/home/test/migration_data_from_oldrhel/migration_data/testtext1.txt'  
'/data1/testtext2.txt' -> '/home/test/migration_data_from_oldrhel/migration_data/testtext2.txt'  
[root@rhel194-ki ~]#
```

以下のコマンドで、差分を確認します。何も表示されなければ成功です。

`diff -r` (コピー元) (コピー先)

コマンド例 : `diff -r /home/test/migration_data_from_oldrhel/ migration_data /data1`

```
[root@rhel194-ki ~]# diff -r /home/test/migration_data_from_oldrhel/migration_data /data1  
[root@rhel194-ki ~]#
```

5. リストア作業

以下のコマンドで、ファイルの中身が読み取れるか確認します。

```
# cd (リストア先のディレクトリ)  
  cat (ファイル名)
```

コマンド例 : cd migration_data_from_oldrhel/ migration_data
 cat testtext1.txt

```
[test@rhel194-ki ~]$ cd migration_data_from_oldrhel/migration_data/  
[test@rhel194-ki migration_data]$ ls -l  
total 8  
-rw-rw-r--. 1 test test 24 Aug 26 03:15 testtext1.txt  
-rw-rw-r--. 1 test test 24 Aug 26 03:15 testtext2.txt  
[test@rhel194-ki migration_data]$ cat testtext1.txt  
testtext1_025 0826 1215  
[test@rhel194-ki migration_data]$ cat testtext2.txt  
testtext2_025 0826 1215
```

5. リストア作業

つながる。驚きを。幸せを。



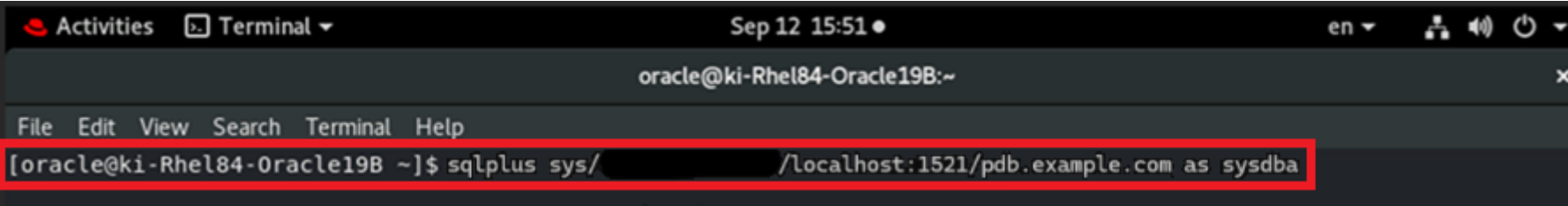
OracleDBのデータ

5. リストア作業

移行先サーバーで、リストア作業を実施します。

以下のコマンドで、データベースに接続します。

```
$ sqlplus ユーザー名/パスワード@//ホスト名(localhost または ホスト名 または IPアドレス):1521/移行対象データのあるPDB as sysdba
```



A terminal window screenshot showing the execution of the sqlplus command. The terminal title bar indicates the user is 'oracle' on a machine named 'ki-Rhel84-Oracle19B'. The command entered is 'sqlplus sys/ /localhost:1521/pdb.example.com as sysdba', which is highlighted with a red box. The terminal also shows a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'.

5. リストア作業

以下のコマンドで、移行対象のDBがあるPDBにセッションを切り替えます。

```
SQL> ALTER SESSION SET CONTAINER = DBの移行先となるPDB;
```

```
SQL> ALTER SESSION SET CONTAINER = ORCLPDB;
```

セッションが変更されました。

以下のコマンドで、移行元にてエクスポートしたdmpファイルが保存されているパスへのディレクトリ・オブジェクトを作成します。

```
SQL> CREATE OR REPLACE DIRECTORY ディレクトリ名 AS '/DB_create';
```

```
SQL> CREATE OR REPLACE DIRECTORY DATA_PUMP_DIR AS '/DB_create';
```

ディレクトリが作成されました。

5. リストア作業

以下のコマンドで、PDB内にインポートデータの受け皿となるユーザーを作成し、権限(データベース利用権限、読み書き権限)を付与します。

```
SQL> CREATE USER ユーザー名 IDENTIFIED BY パスワード;  
SQL> GRANT CONNECT, RESOURCE TO ユーザー名;  
SQL> GRANT READ, WRITE ON DIRECTORY ディレクトリ名 TO ユーザー名;
```

```
SQL> CREATE USER testuser IDENTIFIED BY " パスワード ";
```

ユーザーが作成されました。

```
SQL> GRANT CONNECT, RESOURCE TO testuser;
```

権限付与が成功しました。

```
SQL> GRANT READ, WRITE ON DIRECTORY DATA_PUMP_DIR TO testuser;
```

権限付与が成功しました。

5. リストア作業

以下のコマンドで、作成した表領域への書き込み権限を指定ユーザーに付与します。
完了したら、exitコマンドでデータベースを終了します。

SQL> ALTER USER ユーザー名 QUOTA UNLIMITED ON 表領域名;

```
SQL> ALTER USER testuser QUOTA UNLIMITED ON USERS;  
ユーザーが変更されました。  
SQL> exit  
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production  
Version 19.13.0.0.0との接続が切断されました。
```

5. リストア作業

以下コマンドで、DBをインポートします。

`$ impdp ユーザー名(system or sys)/パスワード@//ホスト名(localhost または ホスト名 または IPアドレス):1521/移行対象データのあるPDB schemas=DBを使用するスキーマ名 directory=指定したディレクトリ名 dumpfile=移行元で作成したdmpファイル logfile=作成したいログファイル名.log`

```
[oracle@ki-Rhel84-Oracle19B ~]$ impdp SYSTEM/ パスワード @//localhost:1521/orclpdb.novalocal schemas=testuser directory=DATA_PUMP_DIR dumpfile=testuser_export.dmp logfile=import.log

Import: Release 19.0.0.0.0 - Production on 金 9月 12 16:53:31 2025
Version 19.13.0.0.0

Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.

接続先: Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
マスター表 "SYSTEM"."SYS_IMPORT_SCHEMA_01"は正常にロード/アンロードされました
"SYSTEM"."SYS_IMPORT_SCHEMA_01"を起動しています: SYSTEM/*****@//localhost:1521/orclpdb.novalocal schemas=testuser directory=DATA_PUMP_DIR dumpfile=testuser_export.dmp logfile=import.log
オブジェクト型 SCHEMA_EXPORT/USERの処理中です
ORA-31684: オブジェクト型 USER: "TESTUSER"はすでに存在します

オブジェクト型 SCHEMA_EXPORT/SYSTEM_GRANTの処理中です
オブジェクト型 SCHEMA_EXPORT/ROLE_GRANTの処理中です
オブジェクト型 SCHEMA_EXPORT/DEFAULT_ROLEの処理中です
オブジェクト型 SCHEMA_EXPORT/PRE_SCHEMA/PROCACT_SCHEMAの処理中です
オブジェクト型 SCHEMA_EXPORT/TABLE/TABLEの処理中です
オブジェクト型 SCHEMA_EXPORT/TABLE/TABLE_DATAの処理中です
. "TESTUSER"."PRODUCTS"          7.078 KB          5行がインポートされました
オブジェクト型 SCHEMA_EXPORT/TABLE/CONSTRAINT/CONSTRAINTの処理中です
オブジェクト型 SCHEMA_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICSの処理中です
オブジェクト型 SCHEMA_EXPORT/TABLE/STATISTICS/TABLE_STATISTICSの処理中です
オブジェクト型 SCHEMA_EXPORT/STATISTICS/MARKERの処理中です
ジョブ "SYSTEM"."SYS_IMPORT_SCHEMA_01"が完了しましたが、1エラーが金 9月 12 16:53:48 2025 elapsed 0 00:00:16で発生しています

[oracle@ki-Rhel84-Oracle19B ~]$
```

6. バックアップデータ保存用ボリューム・移行元サーバーの削除

新環境へのデータ移行完了しており、アプリケーションが問題なく稼働することをご確認いただけましたら、バックアップデータ保存用ボリュームと移行元サーバーを削除していきます。

6. バックアップデータ保存用ボリューム・移行元サーバーの削除

つながろう。驚きを。幸せを。



【ボリュームの削除】

6. バックアップデータ保存用ボリューム・移行元サーバーの削除

つながり。驚きを。幸せを。



移行先サーバーで、下記コマンドを入力し、バックアップデータ保存用ボリュームをアンマウントします。

コマンド例 : # *umount /data1*

```
[root@20250804-ky-rhel8~]#umount /data1
```

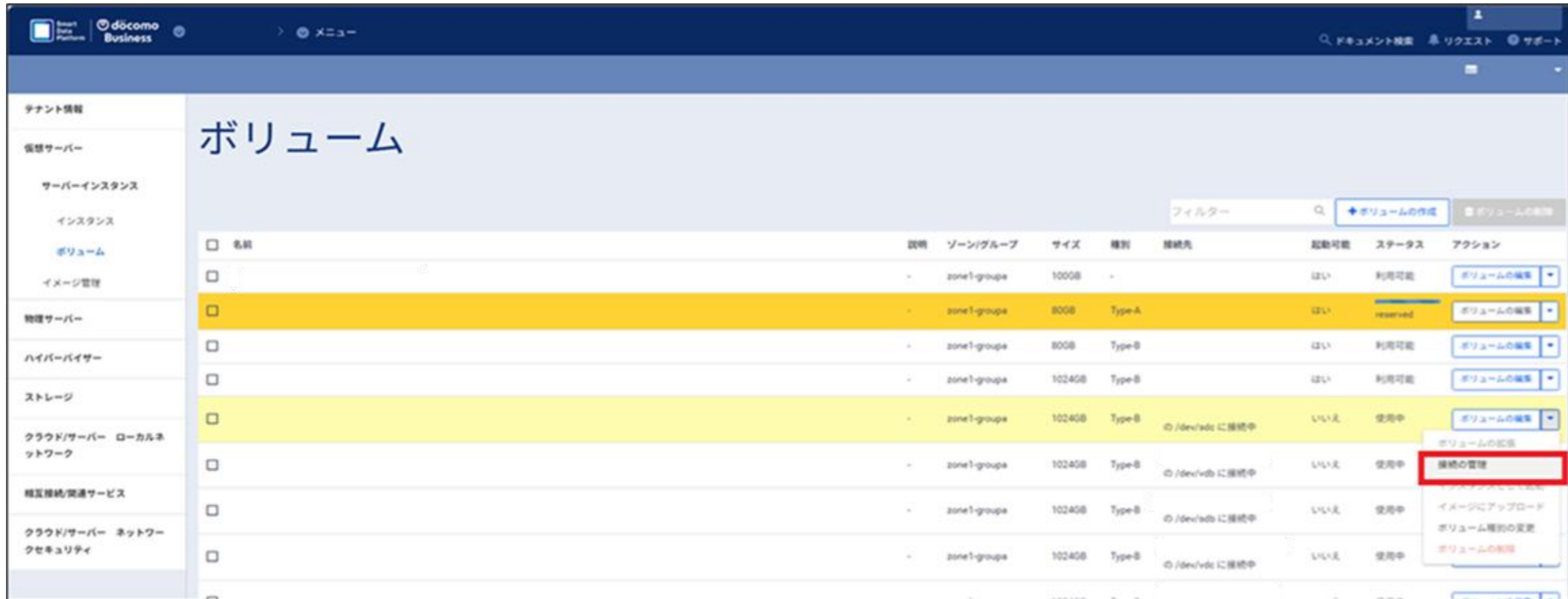
下記コマンドを入力し、アンマウントされたことを確認します。

df -Th

```
[root@20250804-ky-rhel8~]#df -Th
```

6. バックアップデータ保存用ボリューム・移行元サーバーの削除

SDPFポータルで、「仮想サーバー」→「サーバーインスタンス」→「ボリューム」画面で、バックアップデータ保存用ボリュームを選択し、「接続の管理」を選択します。



6. バックアップデータ保存用ボリューム・移行元サーバの削除

「ボリュームの接続の管理」画面で、「ボリュームの切断」を押下します。



「ボリュームの切断の確認」画面が表示されるので、「ボリュームの切断」を押下します。



6. バックアップデータ保存用ボリューム・移行元サーバーの削除

つながり。驚きを。幸せを。



以下を参照して、コントロールパネルのボリューム一覧画面から、削除したいボリュームを選択し、「ボリュームの削除」を実行します。

ボリュームの削除方法

6. バックアップデータ保存用ボリューム・移行元サーバーの削除

つながり。驚きを。幸せを。



【移行元サーバーの削除】

6. バックアップデータ保存用ボリューム・移行元サーバーの削除

以下を参照して、移行元サーバーを削除します。

インスタンスの削除

つながろう。驚きを。幸せを。

 NTT docomo Business

1.1.6 仮想サーバーインスタンス新環境移行（RHEL+Oracle DB）データ移行手順

- APIを利用した移行作業内容および詳細
 - APIを利用した手順について、現時点での準備はございません。

■ 切り戻し方法

- 「手順5. 移行元サーバーの削除」前であれば、移行元環境が残っているため、切り戻しが可能です。
- 「手順5. 移行元サーバーの削除」後の切り戻しはできません。
- 必要に応じて、以下手順を参考にサーバーインスタンスのスナップショット作成をご検討ください。
[2.6.インスタンスのスナップショットを作成する - サーバーインスタンス チュートリアル](#)
[5.2.ボリュームをイメージ管理へアップロードする - サーバーインスタンス チュートリアル](#)

1.1.6 仮想サーバーインスタンス新環境移行（RHEL+Oracle DB）データ移行手順

■ 手順通り進まない場合の対処方法

- エラー等が発生した場合や手順に関する不明点がございましたら、チケットシステムよりチケット起票いただきお問い合わせください。

※チケット起票手順につきましては、「[SDPFクラウド/サーバー関連チケット起票方法](#)」をご参照ください。
お問い合わせ内容によって下記区分をご選択ください。

Incident Submission	:	故障・不具合に関するお問い合わせ
General Inquiry	:	設備更改のガイドラインおよび手順書に関するお問い合わせ