



# VOSS

## VOSS Migrate

# Increasing Disk Space in Migrate

Release 1.0

Jun 19, 2025

Copyright © 2025 VisionOSS Limited. All rights reserved.

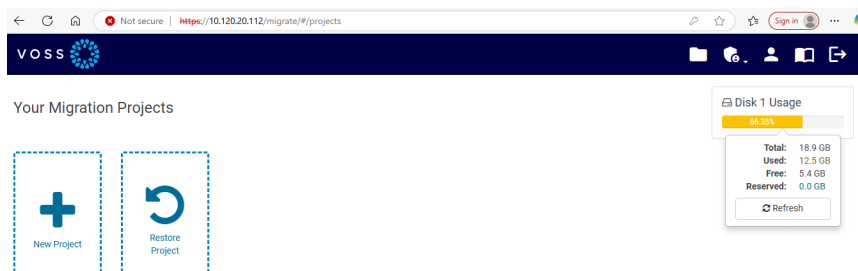
## Contents

<b>Introduction</b>	<b>1</b>
<b>Work Instructions</b>	<b>2</b>

## Introduction

This document describes the detailed steps for increasing disk space on the VOSS Migrate. When you start using the Migrate application, you will notice that you are running out of space for your migration project. The screenshot below illustrates the disk space currently being monitored.

To allow the Migrate application to handle more migration projects, you'll need to increase your disk size.



Sda1 is the root partition comes with standard OVA

```
platform@cpt-naz-migrate:~$ diag disk
Filesystem      Size  Used Avail Use% Mounted on
udev            7.8G     0  7.8G   0% /dev
tmpfs           3.2G  106M   3.1G   4% /run
/dev/sda1       18G   4.3G   13G  26% /
tmpfs           7.9G     0   7.9G   0% /dev/shm
tmpfs           5.0M     0   5.0M   0% /run/lock
tmpfs           7.9G     0   7.9G   0% /sys/fs/cgroup
none            7.9G     0   7.9G   0% /run/shm
none            7.9G    20K   7.9G   1% /tmp
/dev/sdb1       9.9G   37M   9.4G   1% /var/log
/dev/sdb2       40G   6.9G   31G  19% /opt/platform
tmpfs           1.6G     0   1.6G   0% /run/user/1401

platform@cpt-naz-migrate:~$
```

## Work Instructions

### 2.1 Shut down VOSS Migrate via the CLI

Type the command system shutdown in the VOSS CLI and create a snapshot of the VM before making any changes.

## 2.2 Add New Harddrive

Log in to the VMware console and select the VOSS Migrate VM.

Right-click on the VM and select **Edit Settings**.

Click **Add...** and select **Hard Disk**.

Follow the rest of the wizard and set the parameters - 300GB, thick provisioned in this case.

Once done, power on the VM and log in to the system as the platform user.

Run a disk listing using the command: `drives list`

```
drives add <disk> migrate:database
```

After adding the drive, below is the output from our lab *before* running the `drive add` command:

```
platform@cpt-naz-migrate:~$ drives list
Used disks and mountpoints:
    sde1 - insights-voss-sync:database
    dm-0 - mongodb:dbroot
    sdc1 - services:backups

Unused disks:
sdf
Unused mountpoints:
migrate:database
migrate:migrater-data
services:SWAPSPACE

Volume Groups
voss - 25 GB free, 250 GB total
Physical volumes:
sdd1
Logical volumes:
dbroot/dm-0 - 225 GB
platform@cpt-naz-migrate:~$
```

```
platform@cpt-naz-migrate:~$ drives add sdf migrate:database
```

```
platform@cpt-naz-migrate:~$ drives add sdf migrate:database
Configuration setting "devices/scan_lvs" unknown.
Configuration setting "devices/allow_mixed_block_sizes" unknown.
WARNING: Failed to connect to lvmetad. Falling back to device scanning.
cec54165-5b69-4b26-b682-08e15e6e431f
Application migrate processes stopped.
Migrating data to new drive - this can take several minutes
Data migration complete - reassigning drive
Checking that /dev/sdel is mounted
/opt/platform/apps/insights-voss-sync/database
Checking that /dev/sdf1 is mounted
Checking that /dev/dm-0 is mounted
/opt/platform/apps/mongodb/dbroot
Checking that /dev/sdcl is mounted
/backups

Application services:firewall processes stopped.
Reconfiguring applications...
Application migrate processes started.
platform@cpt-naz-migrate:~$
```

```
platform@cpt-naz-migrate:~$ drives list
```

```
platform@cpt-naz-migrate:~$ drives list
Used disks and mountpoints:
  sdel - insights-voss-sync:database
  sdf1 - migrate:database
  dm-0 - mongodb:dbroot
  sdcl - services:backups

Unused disks:
  none - if disks have been hot-mounted, it may be necessary to reboot the system

Unused mountpoints:
  migrate:migrater-data
  services:SWAPSPACE

Volume Groups
  voss - 25 GB free, 250 GB total
  Physical volumes:
    sdd1
  Logical volumes:
    dbroot/dm-0 - 225 GB
platform@cpt-naz-migrate:~$
```

## 2.3 Allow the Migrate application to use the added disk

Log in to the VOSS Migrate VM via the GUI as user udt, click on **Config Change**, and make the modifications as outlined below.

<https://10.120.20.112/migrate/admin/>

The screenshot shows the VOSS Migrate Admin GUI. On the left is a sidebar with navigation links: AUTHENTICATION AND AUTHORIZATION, CONFIG (highlighted), DATA, LOGS, and PERIODIC TASKS. The main content area displays configuration settings for disk usage and project name.

**DISK\_USAGE\_MOUNT\_POINTS**

NAME	DEFAULT	VALUE
DISK_USAGE_MOUNT_POINTS	/	/opt/platform/apps/migrate/database

**DISK\_USAGE\_DEFAULT\_PSQLE\_FREE\_SPACE\_MOUNT\_POINT**

NAME	DEFAULT	VALUE
DISK_USAGE_DEFAULT_PSQLE_FREE_SPACE_MOUNT_POINT	/	/opt/platform/apps/migrate/database

**PROJECT\_NAME**

NAME	DEFAULT	VALUE
PROJECT_NAME	VOSS Migrate	VOSS Migrate

**VOSS Automate Provisioning Tweaks**

NAME	DEFAULT	VALUE
PUT_ONLY_ROLES	relation device/skypeforbusinessonline	relation device/skypeforbusinessonline

## 2.4 Verify the increased disk space

Login to the VOSS Migrate VM via the GUI and click on **Disk Usage**.

The screenshot shows the VOSS Migrate VM GUI. The top navigation bar includes the VOSS logo and a 'Sign in' button. Below the navigation bar, there are two buttons: 'New Project' and 'Restore Project'. On the right side, there is a 'Disk Usage' widget showing the following information:

Category	Value
Total	300 GB
Used	15 GB
Free	285 GB
Reserved	0 GB

A 'Refresh' button is located below the disk usage table.

Confirm that you can see the new available space

Note: How the **migrate:database** is mapped to **/migrater-db**

```
root@cpt-naz-migrate:~# vi /opt/platform/apps/migrate/docker/migrate_backend_start
```

```
VOLUMES="$VOLUMES -v /opt/platform/apps/migrate/database/db-checks:migrater-db:ro"
```

```
platform@opt-naz-migrate: ~
$ /bin/bash

DAPHNE_PORT=8000
POSTGRES_PASSWORD=`/opt/platform/bin/config.py --app=migrate get /seed --value`
POSTGRES_USER=`/opt/platform/bin/config.py --app=migrate get /dbuser --value`
POSTGRES_DB=`/opt/platform/bin/config.py --app=migrate get /dbname --value`
POSTGRES_HOST=`/opt/platform/bin/config.py --app=migrate get /dbhost --value`
REDIS_HOST=`/opt/platform/bin/config.py --app=migrate get /redishost --value`

if [[ "$( /usr/bin/docker ps -aq -f name=migrater$ | wc -l )" -gt 0 ]]; then
    /usr/bin/docker rm -f migrater
fi

VOLUMES="-v /opt/platform/apps/migrate/migrater-data:/migrater-data"
VOLUMES="$VOLUMES -v /opt/platform/apps/migrate/migrater-secrets:/migrater-secrets"
VOLUMES="$VOLUMES -v /opt/platform/apps/migrate/database/db-checks:/migrater-db:ro"

# wait for postgres/redis to be ready
/opt/platform/apps/migrate/utils/services-ready pgreedy
/opt/platform/apps/migrate/utils/services-ready redisready

set -o pipefail
/usr/bin/docker run --rm --name migrater \
-h migrater \
$VOLUMES \
-e POSTGRES_HOST=${POSTGRES_HOST} \
-e POSTGRES_PASSWORD=${POSTGRES_PASSWORD} \
-e POSTGRES_USER=${POSTGRES_USER} \
-e POSTGRES_DB=${POSTGRES_DB} \
-e REDIS_HOST=${REDIS_HOST} \
-p 0.0.0.0:${DAPHNE_PORT}:${DAPHNE_PORT} \
--memory-swappiness 10 \
migrater:latest 2>&1|/usr/bin/logger -t m2uc.backend
```