

# VMware vSphere Bitfusion 範例指南

2021 年 5 月 11 日

進行了更新，以包括 VMware vSphere Bitfusion 3.5  
VMware vSphere Bitfusion 3.0

您可以在 VMware 網站上找到最新的技術文件，網址如下：

<https://docs.vmware.com/tw/>

**VMware, Inc.**  
3401 Hillview Ave.  
Palo Alto, CA 94304  
[www.vmware.com](http://www.vmware.com)

Copyright © 2020-2021 VMware, Inc. 保留所有權利。 [版權與商標資訊](#)。

# 目錄

關於《vSphere Bitfusion 範例指南》	4
<b>1 將 TensorFlow 與 vSphere Bitfusion 搭配使用的簡介</b>	<b>5</b>
<b>2 使用 vSphere Bitfusion 安裝和執行 TensorFlow</b>	<b>6</b>
安裝 NVIDIA CUDA	6
在 Ubuntu 上安裝 NVIDIA CUDA	6
在 CentOS 或 Red Hat Linux 上安裝 NVIDIA CUDA	7
安裝 NVIDIA cuDNN	8
在 CentOS 和 Red Hat Linux 上安裝 Python	9
安裝 TensorFlow	9
安裝 TensorFlow 基準	10
執行 TensorFlow 基準	11

# 關於《vSphere Bitfusion 範例指南》

《vSphere Bitfusion 範例指南》提供使用 vSphere Bitfusion 在 VMware vSphere 上執行 TensorFlow 的相關資訊。

VMware 十分重視包含性。為了在我們的客戶、合作夥伴和內部社群中貫徹這一原則，我們將使用包含性語言建立內容。

《vSphere Bitfusion 範例指南》說明如何安裝 TensorFlow 和開放原始碼基準並使用 vSphere Bitfusion 執行基準。本指南可作為瞭解如何在 vSphere Bitfusion 下使用 TensorFlow 和其他人工智慧 (AI) 以及機器學習 (ML) 應用程式和架構的基礎。

## 預定對象

本資訊適用於想要將 vSphere Bitfusion 與機器學習平台結合使用的任何人。本資訊是針對熟悉使用 VMware vSphere 實作虛擬機器技術和資料中心作業且富有經驗的 Linux 系統管理員而撰寫。

# 將 TensorFlow 與 vSphere Bitfusion 搭配使用的簡介

# 1

若要將 TensorFlow 與 vSphere Bitfusion 搭配使用，您必須安裝和設定多個元件。

若要將 TensorFlow 與 vSphere Bitfusion 搭配使用，請完成下列工作。

- 1 安裝 vSphere Bitfusion。請參閱 VMware vSphere Bitfusion 安裝指南。
- 2 安裝 NVIDIA CUDA 11。
- 3 安裝 NVIDIA cuDNN 8。
- 4 如果您使用的是 CentOS 或 Red Hat Linux，則必須安裝 Python 3。
- 5 安裝 TensorFlow 2.4。
- 6 安裝 TensorFlow 基準。
- 7 執行 TensorFlow 基準以測量您的系統效能。

# 使用 vSphere Bitfusion 安裝和執行 TensorFlow

## 2

若要將 TensorFlow 與 vSphere Bitfusion 搭配使用，您必須安裝並設定數個軟體套件和程式設計架構。

本章節討論下列主題：

- 安裝 NVIDIA CUDA
- 安裝 NVIDIA cuDNN
- 在 CentOS 和 Red Hat Linux 上安裝 Python
- 安裝 TensorFlow
- 安裝 TensorFlow 基準
- 執行 TensorFlow 基準

## 安裝 NVIDIA CUDA

CUDA 是一種由 NVIDIA 開發的平行計算平台和程式設計模型，可在圖形處理單元 (GPU) 上進行一般計算。CUDA 使用 GPU 的處理能力大幅加快計算應用程式的速度。CUDA 將由 TensorFlow 基準使用。

## 在 Ubuntu 上安裝 NVIDIA CUDA

可以在 Ubuntu Linux 上安裝 CUDA。

確認已在 Ubuntu 作業系統上安裝 vSphere Bitfusion 用戶端。

### 程序

- 1 導覽至虛擬機器上要下載 NVIDIA CUDA 發行版的目錄。

```
cd <download_directory>
```

- 2 下載並移動 cuda-ubuntu2004.pin 檔案。

```
wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2004/x86_64/cuda-ubuntu2004.pin
sudo mv cuda-ubuntu2004.pin /etc/apt/preferences.d/cuda-repository-pin-600
```

- 3 使用 `wget` 命令下載適用於 Ubuntu 20.04 的 NVIDIA CUDA 發行版。

```
wget <https://developer.download.nvidia.com/compute/cuda/11.0.3/local_installers/cuda-repo-ubuntu2004-11-0-local_11.0.3-450.51.06-1_amd64.deb>
```

- 4 使用 `dpkg -i` 命令安裝適用於 Ubuntu 20.04 的 CUDA 11 套件。

```
sudo dpkg -i cuda-repo-ubuntu2004-11-0-local_11.0.3-450.51.06-1_amd64.deb
```

- 5 使用 `apt-key` 命令安裝金鑰以驗證軟體套件。

`apt-key` 命令會管理 `apt` 用來驗證套件的金鑰清單。已使用這些金鑰進行驗證的套件將視為受信任。

```
sudo apt-key add /var/cuda-repo-ubuntu2004-11-0-local/7fa2af80.pub
```

- 6 更新並安裝 CUDA 軟體套件。

```
sudo apt-get update
sudo apt-get install cuda
```

- 7 (選擇性) 若要確認 GPU 磁碟分割大小或驗證 vSphere Bitfusion 部署上可用的資源，請執行 NVIDIA 系統管理介面 (`nvidia-smi`) 監控應用程式。

```
bitfusion run -n 1 nvidia-smi
```

- 8 導覽至包含 CUDA 矩陣乘法 (`matrixMul`) 範例檔案的目錄。

```
cd /usr/local/cuda/samples/0_Simple/matrixMul
```

- 9 針對 `matrixMul` 範例檔案執行 `make` 和 `bitfusion run` 命令。

```
sudo make
bitfusion run -n 1 ./matrixMul
```

## 後續步驟

安裝並設定 NVIDIA cuDNN。請參閱[安裝 NVIDIA cuDNN](#)。

## 在 CentOS 或 Red Hat Linux 上安裝 NVIDIA CUDA

可以在 CentOS 8 或 Red Hat Linux 8 上安裝 CUDA 11。

### 程序

- 1 導覽至虛擬機器上要下載 NVIDIA CUDA 發行版的目錄。

```
cd <download_directory>
```

- 2 若要下載適用於 CentOS 8 或 Red Hat Linux 8 的 NVIDIA CUDA 11 套件，請執行 `wget` 命令。

```
wget https://developer.download.nvidia.com/compute/cuda/11.0.3/local_installers/cuda-repo-
rhel8-11-0-local-11.0.3_450.51.06-1.x86_64.rpm
```

- 3 若要安裝 CUDA 套件，請執行 `rpm -i` 命令。

```
sudo rpm -i cuda-repo-rhel8-11-0-local-11.0.3_450.51.06-1.x86_64.rpm
```

- 4 按如下所示執行 `yum clean all` 和 `yum -y install` 命令，以更新您的環境並安裝 CUDA 軟體套件。

```
sudo yum clean all
sudo yum -y install cuda
```

- 5 (選擇性) 若要確認 GPU 磁碟分割大小或驗證 vSphere Bitfusion 部署上可用的資源，請執行 NVIDIA 系統管理介面 (`nvidia-smi`) 監控應用程式。

```
bitfusion run -n 1 nvidia-smi
```

- 6 導覽至包含 CUDA 矩陣乘法 (`matrixMul`) 範例檔案的目錄。

```
cd /usr/local/cuda/samples/0_Simple/matrixMul
```

- 7 針對 `matrixMul` 範例檔案執行 `make` 和 `bitfusion run` 命令。

```
sudo make
bitfusion run -n 1 ./matrixMul
```

## 後續步驟

安裝並設定 NVIDIA cuDNN。請參閱[安裝 NVIDIA cuDNN](#)。

## 安裝 NVIDIA cuDNN

cuDNN 是一個 GPU 加速的基元庫，用於深度神經網路。

### 必要條件

建立一個 NVIDIA 開發人員帳戶，以從該帳戶下載與您的 NVIDIA CUDA 版本相符且適用於您的 Linux 發行版的 cuDNN 套件。請參閱 <https://developer.nvidia.com/cudnn>。



**程序**

- 1 透過執行適用於您的 Linux 發行版的命令順序來安裝 cuDNN 套件。

- ◆ Ubuntu 版本 20.04

```
sudo dpkg -i libcudnn8_8.0.5.39-1+cuda11.0_amd64.deb
```

- ◆ CentOS 8 和 Red Hat Linux 8

```
sudo rpm -ivh libcudnn8-8.0.5.39-1.cuda11.0.x86_64.rpm
```

- 2 若要確認 cuDNN 是否已安裝，請執行 `ldconfig -p | grep cudnn`。

**後續步驟**

- 如果您使用的是 CentOS 或 Red Hat Linux，則必須安裝 Python 3。請參閱在 [CentOS 和 Red Hat Linux 上安裝 Python](#)。
- 如果您使用的是 Ubuntu Linux，則可以安裝 TensorFlow。請參閱[安裝 TensorFlow](#)。

## 在 CentOS 和 Red Hat Linux 上安裝 Python

對於 CentOS 和 Red Hat Linux，必須安裝 Python 3。

如果您使用的是 Ubuntu，則無需執行此程序。Ubuntu 已預先安裝了 Python 3。

**程序**

- 1 透過執行 `yum update` 命令，更新所有目前已安裝的套件。

```
sudo yum update
```

- 2 若要安裝 Python 3，請執行 `dnf` 命令。

```
sudo dnf install python3
```

- 3 透過 `python3 -V` 命令，確認您使用的是 Python 3。

```
python3 -V
Python 3.6.8
```

- 4 (可選) 建立您環境的快照。

**後續步驟**

安裝 TensorFlow。請參閱[安裝 TensorFlow](#)。

## 安裝 TensorFlow

TensorFlow 是您搭配 Bitfusion 使用的機器學習 (ML) 架構。

使用適用於 Python 3 的套件安裝程式 `pip3` 安裝 TensorFlow。

**程序**

- 1 如果在 Ubuntu 20.04 上安裝了 TensorFlow，請安裝其他 Python 資源。

```
sudo apt-get -y install python3-testresources
```

- 2 透過執行適用於您的 Linux 發行版和版本的命令順序來安裝 pip3。

- Ubuntu 20.04

```
sudo apt-get install -y python3-pip
```

- CentOS 8 和 Red Hat Linux 8

```
sudo yum install -y python36-devel
sudo pip3 install -U pip setuptools
```

- 3 使用 pip3 install 命令安裝 TensorFlow。

```
sudo pip3 install tensorflow-gpu==2.4
```

**後續步驟**

可以執行 TensorFlow 基準以測試 vSphere Bitfusion 部署的效能。請參閱[安裝 TensorFlow 基準](#)。

## 安裝 TensorFlow 基準

TensorFlow 基準是一種開放原始碼 ML 應用程式，旨在測試 TensorFlow 架構的效能。

可以將 TensorFlow 基準建立分支並下載至本機環境。在 Git 中，分支是一條獨立的開發線。

**程序**

- 1 安裝 Git。

```
sudo yum -y update
sudo yum install git
```

- 2 建立 ~/bitfusion 並將其設為您的工作目錄。

```
mkdir bitfusion
cd ~/bitfusion
```

- 3 將 Tensorflow 基準的 Git 存放庫複製到您的本機環境。

```
git clone https://github.com/tensorflow/benchmarks.git
```

#### 4 導覽至基準目錄並列出存放庫的分支。

```
cd benchmarks
git branch -a

master
remotes/origin/HEAD -> origin/master
...
remotes/origin/cnn_tf_v1.13_compatible
...
remotes/origin/cnn_tf_v2.1_compatible
...
```

#### 5 執行 Git 簽出並列出 TensorFlow 基準存放庫。

```
git checkout cnn_tf_v2.1_compatible
```

```
Branch cnn_tf_v2.1_compatible set up to track remote branch cnn_tf_v2.1_compatible
from origin.
Switched to a new branch 'cnn_tf_v2.1_compatible'
```

```
git branch
```

```
cnn_tf_tf_v2.1_compatible
master
```

#### 後續步驟

可以執行 TensorFlow 基準以測試 vSphere Bitfusion 部署的效能。請參閱[執行 TensorFlow 基準](#)。

## 執行 TensorFlow 基準

可以執行 TensorFlow 基準以測試 vSphere Bitfusion 和 TensorFlow 部署的效能。

透過執行 TensorFlow 基準並使用各種組態，可以瞭解 ML 工作負載在 vSphere Bitfusion 環境中的回應方式。

#### 程序

- 若要導覽至 ~/bitfusion/ 目錄，請執行 `cd ~/bitfusion/`。
- 若要使用 `tf_cnn_benchmarks.py` 基準指令碼，請執行 `bitfusion run` 命令。

透過執行範例中的命令，可以使用單一 GPU 的全部記憶體和 /data 目錄中預先安裝的 ML 資料。

```
bitfusion run -n 1 -- python3 \
./benchmarks/scripts/tf_cnn_benchmarks/tf_cnn_benchmarks.py \
--data_format=NCHW \
--batch_size=64 \
```

```
--model=resnet50 \
--variable_update=replicated \
--local_parameter_device=gpu \
--nodistortions \
--num_gpus=1 \
--num_batches=100 \
--data_dir=/data \
--data_name=imagenet \
--use_fp16=False
```

- 3 若要使用 `tf_cnn_benchmarks.py` 基準指令碼，請執行具有 `-p 0.67` 參數的 `bitfusion run` 命令。透過執行範例中的命令，可以使用單一 GPU 的 67% 的記憶體和 `/data` 目錄中預先安裝的 ML 資料。`-p 0.67` 參數可讓您在其餘 33% 的 GPU 記憶體磁碟分割中執行其他工作。

```
bitfusion run -n 1 -p 0.67 -- python3 \
./benchmarks/scripts/tf_cnn_benchmarks/tf_cnn_benchmarks.py \
--data_format=NCHW \
--batch_size=64 \
--model=resnet50 \
--variable_update=replicated \
--local_parameter_device=gpu \
--nodistortions \
--num_gpus=1 \
--num_batches=100 \
--data_dir=/data \
--data_name=imagenet \
--use_fp16=False
```

- 4 若要使用 `tf_cnn_benchmarks.py` 基準指令碼，請使用整合資料執行 `bitfusion run` 命令。透過執行範例中的命令，可以使用單一 GPU 的全部記憶體，而不使用預先安裝的 ML 資料。TensorFlow 可以使用一組模擬映像建立整合資料。

```
bitfusion run -n 1 -- python3 \
./benchmarks/scripts/tf_cnn_benchmarks/tf_cnn_benchmarks.py \
--data_format=NCHW \
--batch_size=64 \
--model=resnet50 \
--variable_update=replicated \
--local_parameter_device=gpu \
--nodistortions \
--num_gpus=1 \
--num_batches=100 \
--use_fp16=False
```

## 結果

您現在可以透過 vSphere Bitfusion 使用遠端伺服器的共用 GPU 執行 TensorFlow 基準。基準支援許多模型和參數，可協助您探索機器學習學科中的廣闊空間。如需詳細資訊，請參閱 VMware vSphere Bitfusion 使用者指南。