**GUIDA CREAZIONE ACCOUNT CINECA E UTILIZZO CLUSTER LEONARDO PER LA GENERAZIONE DI FOTO DA TESTO**

La presente è una guida passo passo per la lavorazione e creazione di **immagini sequenziali partendo da un testo**.

L'intero processo è reso possibile grazie all'utilizzo delle risorse di **High Performance Computing (HPC)** fornite dal sistema **Leonardo del CINECA**. Questo cluster offre una potenza di calcolo nettamente superiore rispetto ai sistemi casalinghi, necessaria per gestire modelli complessi come quelli utilizzati per la generazione di immagini da testo.

La guida copre tutti i passaggi, dalla **creazione dell'account CINECA** fino all'effettivo **utilizzo del codice** per la generazione di immagini tramite testo.

**Panoramica dei Passaggi Principali:**

1. **Creazione dell'Account CINECA** sul portale UserDB.
2. **Configurazione dell'Accesso al Cluster Leonardo** (tramite client Smallstep, agent SSH e autenticazione a due fattori - 2FA).
3. **Preparazione dell'Ambiente di Sviluppo** su Leonardo (caricamento moduli e creazione ambiente virtuale).
4. **Installazione di Stable Diffusion** e delle relative dipendenze (utilizzando PyTorch con supporto CUDA e librerie come diffusers e transformers).
5. **Trasferimento dei File** (modelli e script Python) tra il computer locale e il cluster tramite comandi SCP.
6. **Esecuzione dello Script di Generazione Immagini** sui nodi di calcolo di Leonardo tramite il sistema di gestione code SLURM (.sh file).

***Creazione account cineca***

* Accedere al seguente link e creare il proprio account <https://userdb.hpc.cineca.it>

Immagine che contiene testo, schermata, software, Pagina Web

Il contenuto generato dall'IA potrebbe non essere corretto.

* Aperto il link selezionare “ Create new user “ per poter procedere alla registrazione
* Compilare il form in ogni sezione con dati corretti e veritieri
* Completata la registrazione controllare la mail per poter continuare la registrazione, tramite l’impostazione di una password da associare all’account
  + In caso di problemi con il link , richiederne un altro
* Procedere ora con l’accesso inserendo i propri dati personali

Immagine che contiene testo, schermata, Pagina Web, Sito Web

Il contenuto generato dall'IA potrebbe non essere corretto.

* Completare il proprio profilo utente compilando **TUTTE** e quattro le sezioni presenti

Immagine che contiene testo, schermata, numero, software

Il contenuto generato dall'IA potrebbe non essere corretto.

* Fatto ciò bisogna attendere la convalida dei dati inseriti (24-72 ore) e recarsi nella sezione “HPC Access” e cliccare sul tasto “[Sumbit]”

Immagine che contiene testo, schermata, Carattere, numero

Il contenuto generato dall'IA potrebbe non essere corretto.

Immagine che contiene testo, Carattere, schermata

Il contenuto generato dall'IA potrebbe non essere corretto.

* Non è possibile compiere tale azione se le quattro voci [Personal Data], [Institution], [HPC Docs] e [Ask for authorisation] non sono state eseguite correttamente da un comando “[OK]”
  + Nel caso in cui tutte le voci siano state compilate correttamente e non risulta ancora il tasto [Sumbit] attendere ancora
* Finiti i passaggi precedenti , bisognerà attendere, arriverà successivamente una mail da Cineca nella quale verranno comunicati i passaggi da eseguire con il nostro relativo username
* Una seconda mail con un link , con validità di 24h, attraverso il quale sarà possibile completare il nostro profilo
* Sarà necessario definire un codice unico , necessario per la procedura a due fattori 2FA

Immagine che contiene testo, schermata, grafica, Elementi grafici

Il contenuto generato dall'IA potrebbe non essere corretto.

* Verifichiamo la mail inserita in fase di registrazione, per poi impostare la password per Cineca HPC

Immagine che contiene testo, grafica, software, aqua

Il contenuto generato dall'IA potrebbe non essere corretto.

* Bisognerà definire un codice OTP, utilizzando un software esterno (Google Authenticator o Microsoft Authenticator)

Immagine che contiene testo, schermata, grafica, Sito Web

Il contenuto generato dall'IA potrebbe non essere corretto.

* Verrà poi mostrata una schermata con i codici di recupero , **VISUALIZZABILI SONO IN QUESTO PASSAGGIO**, da utilizzare in caso di errori nell’uso di OTP.
  + **SALVARI IN UN LUOGO SICURO PER NON PERDERLI**
  + **NON AVERE FRETTA IN QUESTO PASSAGGIO**

***Utilizzo Cluster Cineca***

* Esistono due guide presenti al seguente link (Windows o Linux/MacOS) https://wiki.u-gov.it/confluence/display/SCAIUS/2.1%3A+How+to+install+the+smallstep+client+and+configure+2FA+and+OTP
* LINUX/MACOS GUIDA
  + Eseguire nella shell i seguenti comandi
    - $ step ca bootstrap --ca-url=https://sshproxy.hpc.cineca.it --fingerprint 2ae1543202304d3f434bdc1a2c92eff2cd2b02110206ef06317e70c1c1735ecd
  + Per usare il certificato , l’utente deve attivare l’agente ssh
    - $ eval $(ssh-agent)
  + Per ottenere il certificato
    - $ step ssh login '<user-email>' --provisioner cineca-hpc

Immagine che contiene testo, schermata, Carattere, informazione

Il contenuto generato dall'IA potrebbe non essere corretto.

* + - Il risultato ottenuto dovrebbe essere il seguente
  + Si aprirà automaticamente il browser e ti chiederà di accedere

Immagine che contiene testo, schermata, grafica, Elementi grafici

Il contenuto generato dall'IA potrebbe non essere corretto.

* + Verrà richiesto il codice OTP tramite l’Authenticator utilizzato in fase di registrazione

Immagine che contiene testo, schermata, Carattere, biglietto da visita

Il contenuto generato dall'IA potrebbe non essere corretto.

* + Ottenuta questa schermata vuol dire che il certificato è stato generato correttamente
  + Per verificare la correttezza del certificato via ssh-agent tramite i seguenti comandi
    - $ ssh-add -L
    - $ step ssh list
  + Per esaminare la validità del certificato eseguire
    - $ step ssh list --raw  '<user\_email>' | step ssh inspect
  + Per loggarsi inserire
    - ssh [usernameCluster@login.leonardo.cineca.it](mailto:usernameCluster@login.leonardo.cineca.it)
      * Nel caso di errori digitare yes

Immagine che contiene testo, schermata, Carattere, informazione

Il contenuto generato dall'IA potrebbe non essere corretto.

* + - In questo caso si è presentato un problema per quanto riguarda le host keys, e per risolvere, bisognerà eliminare il certificato non valido
      * $ sudo nano -w /home/user/.ssh/known\_hosts
      * Mostrerà il file interessato, e sarà necessario eliminare le righe presenti nel file. Per poterle eliminare basterà usare la combinazione ‘ CTRL + K ‘ , che eliminerà la riga, poi si procede al salvataggio delle modifiche con ‘ CTRL + X ‘, poi confermare con S e poi premere ‘ Invio ‘.
    - Spunterà infine questa schermata

Immagine che contiene testo, schermata, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

* WINDOWS
  + Aprire la powershell come amministratore
  + Inserire il comando per installare scoop
    - iwr **-useb** get.scoop.sh | **iex**
  + Test scoop
    - Scoop help
  + Installare git support per scoop
    - Scoop install git
  + Installare smallstep
    - scoop bucket add smallstep <https://github.com/smallstep/scoop>**-bucket**.git
    - scoop install smallstep/step
  + Verifica smallstep
    - Step
  + Configurazione di step client
    - step ca bootstrap -**-ca-url**=[https://sshproxy.hpc.cineca.it](https://sshproxy.hpc.cineca.it/) -**-fingerprint** 2ae1543202304d3f434bdc1a2c92eff2cd2b02110206ef06317e70c1c1735ecd
  + Attivazione di ssh-agent
    - Get-Service -Name ssh-agent
    - Start-Service -Name ssh-agent
  + Sequenze di utilizzo
    - step ssh login <your**-email**> -**-provisioner** cineca**-hpc**
    - step ssh list

Immagine che contiene testo, schermata, Carattere, Blu elettrico

Il contenuto generato dall'IA potrebbe non essere corretto.

* + - step ssh list -**-raw**  <user\_email> | step ssh inspect

Immagine che contiene testo, schermata, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

* + Ora puoi accedere al cluster
  + Per loggarsi inserire
    - ssh [usernameCluster@login.leonardo.cineca.it](mailto:usernameCluster@login.leonardo.cineca.it)
      * Nel caso di errori digitare yes

Immagine che contiene testo, schermata, Carattere, informazione

Il contenuto generato dall'IA potrebbe non essere corretto.

* + - * In questo caso si è presentato un problema per quanto riguarda le host keys, e per risolvere, bisognerà eliminare il certificato non valido
        + Andare nella directory C:\Users\nomeutente\.ssh
        + Eliminare il file presente o svuotare il contenuto una volta aperto come file txt
      * Spunterà infine questa schermata

Immagine che contiene testo, schermata, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

***EXTRA: Creazione chiave all’interno dell’ambiente Leonardo***

* Date nuove politiche nel 2025 è diventato necessario creare una chiave per evitare limitazioni interne al sistema all’interno del cluster Leonardo.
* Si crea la chiave quindi tramite l’apposito comando (dopo i passaggi eseguiti precedentemente)
  + ***ssh-keygen***
* Eseguire subito dopo il comando per autenticare le connessioni interne tra i nodi del cluster
  + cat ~/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys

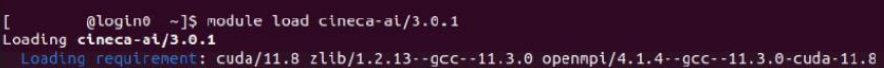
***Creazione ambiente virtuale***

* Appena effettuato l’accesso al Cluster, abbiamo un ambiente di sviluppo asettico; è quindi necessario caricare sul cluster i moduli di sviluppo con le relative librerie. Per fare questo è sufficiente seguire questi prompt:

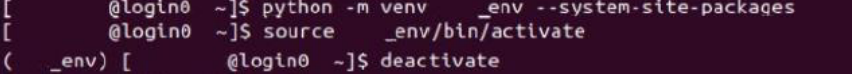
Immagine che contiene testo, schermata, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

* Per caricare il modulo è sufficiente inserire il seguente prompt



* Fatto ciò un passo importante è la definizione di un ambiente virtuale, in questo caso tramite i seguenti comandi



* Con il comando $ python -m venv “nome ambiente” –system-site-packages andiamo a creare il nostro ambiente virtuale che verrà salvato nel nostro cluster
* Per potervi accedere una volta creato basta inserire $source nome\_abiemnte/bin/activate.
* Per uscire $ deactivate

***Trasferimento File Su Cluster***

Importante per il funzionamento del sistema è possibile tramite comandi SCP , capaci di offrire possibilità di trasferimento file dal nostro personal computer al nostro cluster e viceversa, permettendoci di utilizzare software non eseguibili su hardware casalinghi con bassa potenza di calcolo. Nello specifico verrà usato il passaggio di file python e verranno utilizzati per test generativi eseguendoli con Leonardo.

**NB: Se vogliamo trasferire da leonardo al nostro pc dobbiamo eseguire i comandi scp sul nostro terminale e non su leonardo**

* Usiamo i comandi SCP permettendo di trasferire i file (tramite console)
  + $scp file.txt remote\_username@IP:/remote/directory
    - Dove remote\_username è il cluster di destinazione, mentre IP è l’indirizzo IP oppure il Nome dell’host
  + Un esempio di trasfermiento con un file dal cluster al nostro pc:
    - scp [nomeutente@login.leonardo.cineca.it:~/stable-diffusion/generated\_image.png](mailto:nomeutente@login.leonardo.cineca.it:~/stable-diffusion/generated_image.png) .
    - Come nomeutente si utilizzerà la stessa riga usata per loggarsi con ssh
    - Se vogliamo specificare la directory attuale usiamo il “.” altrimenti potremmo specificare la directory
      * Esempio
      * scp [nomeutente@login.leonardo.cineca.it:~/stable-diffusion/generated\_image.png](mailto:nomeutente@login.leonardo.cineca.it:~/stable-diffusion/generated_image.png) C:\Users\nomeutente\Desktop
    - Se volessimo vedere il file path usare $pwd , per il percorso utilizzato per il trasferimento dati
* Provvediamo alla verifica del trasferimento del file accediamo al cluster (sempre tramite ssh e fare accesso alla directory “Prove”) per visualizzare il contenuto e che tutto sia andato a buon fine
  + $ cd Prove/
  + $ ls oppure ls -la
  + $ cat Prova.txt (se volessimo vedere il contenuto)

***Perché Cineca e Hugging Face?***

Data la grande complessità nell’utilizzo dei modelli LLM o VLM , composti da miliardi di parametri, risulta necessario l’utilizzo di hardware nettamente superiore rispetto ad un normale pc casalingo. Si è reso necessario utilizzare una delle migliori strutture hardware presenti ad oggi, ovvero Cineca.

L’utilizzo del sistema risulta tuttavia più complesso rispetto ad un regolare personal computer, gestibile solamente tramite comandi tramite terminale, con una assenza quasi totale di interfaccia grafica , non permettendo l’accesso all’esterno del sistema.

Non potendo quindi accedere ad internet, si è dovuto ricorrere ad un collegamento con dei nodi esterni reso possibile grazie ad Huggingface, indispensabile per soluzioni locali.

* Azioni necessarie per l’utilizzo di modelli sul cluster sono:
  + Scaricare in locale sul nostro personal computer i modelli che desideriamo utilizzare
  + Tramite comandi Scp trasferirli su Cineca
  + Poter utilizzare Llama con Leonardo sfruttando la potenza hardawre di Cineca

I**NSTALLAZIONE DIPENDENZE E STABLE DIFFUSION**

* Aggiornare pip ed eseguire: pip install –upgrade pip
* Ora passiamo all’installazione di PyTorch con supporto CUDA per poter usufruire delle risorse grafiche (GPU) di Leonardo:
* pip install torch torchvision torchaudio --extra-index-url <https://download.pytorch.org/whl/cu11>
* Fatto questo procediamo a installare Stable Diffusion e altre librerie:
* pip install diffusers transformers accelerate safetensors scipy numpy

**HUGGING FACE**

* Per poter scaricare Stable Diffusione abbiamo bisogno di un modello pre addestrato. Bisogna quindi accedere ad Hugging Face e crearsi un’account
  + <https://huggingface.co/>
* Una volta creato un account bisogna ottenere una chiave di accesso su Hugging face
* Per fare ciò una creato il proprio account ed essersi loggati andare su settings

Immagine che contiene testo, schermata, software, Software multimediale

Il contenuto generato dall'IA potrebbe non essere corretto.

* Selezionare Access Token e Create New Token

Immagine che contiene testo, schermata, software, Software multimediale

Il contenuto generato dall'IA potrebbe non essere corretto.

* Selezionare tipologia Write e scegliere il nome

Immagine che contiene testo, schermata, software, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

* Fatto ciò è **MOLTO** importante salvarsi il tutto in un luogo sicuro

**STABLE DIFFUSION 1.5 INSTALLAZIONE**

* Ora possiamo tornare nuovamente sul nostro terminale per autenticarci tramite Hugging face:
  + huggingface-cli login
  + hf auth login (se il primo dovesse risultare deprecato)
* Qui verrà richiesto di inserire un Token di Accesso, inserire quello precedentemente salvato con cura

Immagine che contiene schermata, linea

Il contenuto generato dall'IA potrebbe non essere corretto.

* + In caso di problemi andare su Hugging Face e aggiornare il token o crearne un altro
  + In caso del seguente errore eseguire il comando mostrato per salvare le credenziali

Immagine che contiene schermata, testo, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

* Procediamo ora all’installazione del modello Stable Diffusion 1.5:
  + mkdir -p stable-diffusion/models
  + cd stable-diffusion/models
  + wget <https://huggingface.co/runwayml/stable-diffusion-v1-5/resolve/main/v1-5-pruned.ckpt>
* Procediamo ora all’installazione di VAE similmente a Stable Diffuson 1.5
  + wget <https://huggingface.co/stabilityai/sd-vae-ft-mse-original/resolve/main/vae-ft-mse-840000-ema-pruned.safetensors>  
    OPPURE
  + wget https://huggingface.co/stabilityai/sd-vae-ft-mse-original/resolve/main/vae-ft-mse-840000-ema-pruned.ckpt
* Finita l’installazione possiamo ora passare alla creazione di uno **SCRIPT** per l’esecuzione di Stable Diffusion
  + Siccome i nodi di Leonardo impediscono le esecuzioni dirette sul login node, dobbiamo utilizzare file sh per eseguire lo script
* Creare quindi un file chiamato sd\_generate.sh tramite console (“nano sd\_generatre.sh”):

#!/bin/bash

#SBATCH --job-name=stable\_diffusion

#SBATCH --partition=boost\_usr\_prod # Usa la coda booster per le GPU

#SBATCH --nodes=1

#SBATCH --ntasks-per-node=1

#SBATCH --gpus-per-node=1

#SBATCH --time=01:00:00

#SBATCH --output=output.txt

* Creiamo quindi anche file chiamto generate.py:
  + Creiamo un file python per l’utilizzo del file sh e la generazione dell’immagine:

from diffusers import StableDiffusionPipeline

import torch

# Carica il modello

pipe = StableDiffusionPipeline.from\_pretrained("runwayml/stable-diffusion-v1-5").to("cuda")

# Prompt dell'immagine

prompt = "A futuristic city at sunset, cyberpunk style, ultra-detailed"

# Genera l'immagine

image = pipe(prompt).images[0]

# Salva l'immagine

image.save("generated\_image.png")

print("Immagine generata con successo!")

**STABLE DIFFUSION DLX INSTALLAZIONE**

* Nel caso in cui volessimo usare una versione più potente per la generazione di immagini con una qualità migliore
* **NB: FARE MOLTA ATTENZIONE AL PESO DEL FILE E DELLO SPAZIO DISPONIBILE SU CINECA** 
  + **Eseguire il comando tramite console cineca:”** cinQuota “ per vedere lo spazio disponibile, in caso di superamento dello spazio, i file verranno cancellati automaticamente dopo qualche ora
* Per prima cosa sul **proprio computer locale** aprire Visual Studio Code oppure PyCharm o un idee a scelta per python
* Tramite console inserire:
  + pip install hf\_xet #NECESSARIO PER VELOCIZZARE IL DOWNLOAD
  + pip install huggingface\_hub
* Copiare il seguente codice dentro l’idee e salvare il file .py in base a dove si vuole scaricare Stable Diffusion DLX

from huggingface\_hub import snapshot\_download

MODEL\_ID = "stabilityai/stable-diffusion-xl-base-1.0"

LOCAL\_DIR = "./SDXL\_COMPLETE\_MODEL"

print(f"Inizio download di {MODEL\_ID}...")

download\_path = snapshot\_download(

repo\_id=MODEL\_ID,

local\_dir=LOCAL\_DIR,

repo\_type="model"

)

print(f"Download completato. I file sono in: {download\_path}")

* Copiare tutto quello installato nella cartella SDXL\_COMPLETE MODEL nel nostro cluster usando i comandi SCP
  + Esempio: scp -r C:\Users\mariorossi\Downloads\SDXL\_COMPLETE\_MODEL [mariorossi@login.leonardo.cineca.it:~/cartella/di/interesse/in/cineca](mailto:mariorossi@login.leonardo.cineca.it:~/cartella/di/interesse/in/cineca)

**NLTK INSTALLAZIONE**

* N**LTK** (acronimo di **Natural Language Tool Kit**) è una potente e popolare **libreria Python** progettata per lavorare con il linguaggio umano, supportando la **Natural Language Processing (NLP)**, ovvero l'elaborazione del linguaggio naturale. NLTK fornisce strumenti e risorse per eseguire una vasta gamma di operazioni di **analisi e pre-elaborazione del testo**. È ampiamente utilizzato nella ricerca, nell'insegnamento e nelle applicazioni di data science per comprendere e dare un senso ai dati testuali.
* Per installarlo bisogna scrivere tramite console:
  + pip install nltk
  + python
  + import nltk
  + nltk.download('punkt\_tab') OPPURE nltk.download('punkt\_tab', '/percorso/su/leonardo')
  + exit()
* Per usarlo nel proprio codice python basterà semplicemente specificare la location
  + NLTK\_DATA\_PATH = '/leonardo/home/userexternal/mariorossi/nltk'
* EXTRA: Se ci dovessero essere problemi con Hugging Face , tornare ai paragrafi precedenti e vedere i passaggi

**ESECUZIONE CODICE SH E PYTHON SU CLUSTER**

* Per poter quindi eseguire effettivamente un codice bisogna in primis eseguire un file bash tramite il comando :
  + sbatch file.sh
  + Un esempio di file sh potrebbe essere il seguente

#!/bin/bash

#SBATCH --job-name=nome\_job

#SBATCH --partition=boost\_usr\_prod # Usato questo personalmente

#SBATCH --nodes=1

#SBATCH --ntasks-per-node=1

#SBATCH --gpus-per-node=1

#SBATCH --time=01:00:00

#SBATCH --output=output.txt

# Attiviamo l'ambiente virtuale

source ~/stable-diffusion-env/bin/activate

# Eseguiamo il codice per generare un'immagine

python ~/cartella /file\_da\_eseguire.py

* + Fare attenzione alla “partition” utilizzata, tramite il comando “sinfo” è possibile consultare quali sono disponibili.
* Il file sh avvierà il nostro relativo file python ad esempio:

from diffusers import StableDiffusionPipeline

import torch

# Carica il modello

pipe = StableDiffusionPipeline.from\_pretrained("runwayml/stable-diffusion-v1-5").to("cuda")

# Prompt dell'immagine

prompt = "A futuristic city at sunset, cyberpunk style, ultra-detailed"

# Genera l'immagine

image = pipe(prompt).images[0]

# Salva l'immagine

image.save("generated\_image.png")

print("Immagine generata con successo!")

* Per vedere infine lo stato del nostro file basterà usare il comando:
  + squeue -u mario\_rossi

**LIBRERIE RICHIESTE DA INSTALLARE**

* Tramite console installa le varie librerie:
  + pip install torch torchvision
  + pip install torch torchvision torchaudio --index-url <https://download.pytorch.org/whl/cu121>
  + pip install diffusers transformers accelerate
  + pip install Pillow lpips tqdm
  + pip install diffusers transformers accelerate Pillow lpips tqdm

**METRICHE CLIP E LPIPS INSTALLAZIONE**

* Per installare le due metriche seguire i seguenti passaggi:
  + pip install lpips
  + pip install transformers
  + pip install diffusers transformers lpips tqdm Pillow nltk
  + wget https://download.pytorch.org/models/alexnet-owt-7be5be79.pth

**COMANDI RAPIDI E PROMEMORIA**

* step ssh login mariorossi@studio.unibo.it --provisioner cineca-hpc
* Get-Service -Name ssh-agent
* Start-Service -Name ssh-agent
* ssh mariorossi@login.leonardo.cineca.it
* scp mariorossi@login.leonardo.cineca.it:~/cartella1/generated\_image.png .
  + Singolo file
* scp -r lmedici0@login.leonardo.cineca.it:~/cartella1/generated\_image.png .
  + Cartella
* sinfo
  + Usato per controllare lo stato delle partizioni
* cinQuota
  + Usato per controllare la quota utilizzata e disponibile di Cineca
* file.sh cosi composto per far funzionare il tutto:

#!/bin/bash

#SBATCH --job-name=nome\_job

#SBATCH --partition=boost\_usr\_prod # Usato questo personalmente

#SBATCH --nodes=1

#SBATCH --ntasks-per-node=1

#SBATCH --gpus-per-node=1

#SBATCH --time=01:00:00

#SBATCH --output=output.txt

# Attiviamo l'ambiente virtuale

source ~/stable-diffusion-env/bin/activate

# Eseguiamo il codice per generare un'immagine

python ~/cartella /file\_da\_eseguire.py

* sbatch file.sh
  + File sh per avviare il file python
* squeue -u mario\_rossi
  + Serve per vedere lo stato del file in esecuzione