Proxmox : utiliser Ollama dans un conteneur LXC avec des GPU Nvidia

Lien : https://shionn.github.io/draft/nvidia-proxmox-lxc-passthrought-ollama.html

Installer Ollama

- utiliser un conteneur avec les pilotes Nvidia
- installer les prérequis

```
apt install -y curl zstd pciutils
```

L'installateur à besoin de lspci (dans pciutils)

- installer Ollama

```
# wget https://ollama.com/install.sh
# bash ./install.sh
>>> Cleaning up old version at /usr/local/lib/ollama
>>> Installing ollama to /usr/local
>>> Downloading ollama-linux-amd64.tar.zst
######################################################################################################################
## 100.0%
>>> Creating ollama user...
>>> Adding ollama user to render group...
>>> Adding ollama user to video group...
>>> Adding current user to ollama group...
>>> Creating ollama systemd service...
>>> Enabling and starting ollama service...
Created symlink '/etc/systemd/system/default.target.wants/ollama.service' ->
'/etc/systemd/system/ollama.service'.
>>> NVIDIA GPU installed.
```

Tester le modèle en console

```
# ollama run qwen2.5-coder:7b
```

Le modèle Qwen2.5-Coder 7B dans Ollama en mode interactif est spécialisé pour le code.

Poser des questions en langage naturel

Il suffit simplement de taper une question comme :

- Comment créer une API REST en Python ?

Le modèle répond dans le terminal.

Générer du code (tous langages) Comme il s'agit d'un modèle coder, on peux lui demander :

- Écris une fonction en JavaScript qui trie une liste d'objets par date.

Ou même des projets complets :

- Génère un Dockerfile pour une application FastAPI.

Expliquer du code :

- Explique ce que fait ce script :

<ton code ici>

Il te donnera une explication détaillée.

Déboguer ou améliorer du code

- Voici mon code, il plante. Trouve l'erreur.

ou

- Optimise cette fonction pour la rendre plus rapide.

Travailler en conversation continue

- Ollama garde l'état de la conversation tant que le processus est lancé. Il est alors possible d'enchaîner les prompts :
- Maintenant rends le code compatible Python 3.12.

Quitter proprement

- /bye

ou simplement CTRL + C.

Utiliser le modèle dans un script (API locale Ollama)

- depuis une autre console, pour appeler l'API :

```
curl http://localhost:11434/api/generate \
  -d '{ "model": "qwen2.5-coder:7b", "prompt": "Écris une classe Python." }'
```

Utiliser dans VS Code ou un éditeur

Beaucoup d'extensions permettent de configurer Ollama comme LLM local. **Qwen2.5-Coder** peux alors être utilisé comme assistant de code directement dans l'IDE.

Lancer en mode serveur

```
ollama serve
```

Le modèle devient accessible à d'autres outils (LM Studio, Continue, Cursor, etc.).