

# WSLの環境構築

## 参考URL

今回の方法はwindows10 build 2004以降、windows11で有効  
windows10のビルド番号は、「設定、バージョン情報、OSビルド」で確認。

## WSLのインストール

powershellを管理権限で開き、下記のコマンドを実行する。  
(powershellはスタートボタンから検索。複数の方法あり)

```
wsl --install
```

＊必ずwindowsの更新プログラムがすべて実行されてから、インストールを行うこと。新しいパソコンは特に注意。

## WSLの起動

windowsターミナル（おすすめ）、またはPowerShellを起動する。必要ないかも？  
(windows10の場合はダウンロード:[url](#))

```
wsl
```

と入力するとwslが起動する。  
ユーザ名、パスワードを設定すると、Linuxのターミナルが起動する。

## WSLの初期設定

### リポジトリの変更

デフォルトではUSのリポジトリになっており、速度が遅いため国内のミラーリポジトリへ変更する。

```
sudo sed -i.org -e 's|archive.ubuntu.com|ftp.jaist.ac.jp/pub/Linux/ubuntu|g' /etc/apt/sc
```

パッケージを更新する。

```
sudo apt update && sudo apt upgrade -y
```

＊このコマンドは、定期的に行うこと。

これで `sudo apt install hoge hoge` で好きなパッケージをインストールできるようになった。

## 日本語フォントの表示

```
# fc-cache コマンド等をインストール
$ sudo apt install -y fontconfig

# Windows側のフォントをシンボリックリンクすることで日本語フォントを使用できるようにする
$ sudo ln -s /mnt/c/Windows/Fonts /usr/share/fonts/windows

# フォントキャッシュクリア
$ sudo fc-cache -fv

# 日本語言語パックのインストール
$ sudo apt -y install language-pack-ja

# ロケールを日本語に設定
$ echo 'export LANGUAGE=ja_JP.UTF-8' >> ~/.bashrc
$ echo 'export LC_ALL=ja_JP.UTF-8' >> ~/.bashrc
$ exec $SHELL -l
$ sudo update-locale LANG=ja_JP.UTF8

# いったん終了して再起動すればアプリケーションで日本語が使えるようになる
$ exit

# --- WSL2シェル再起動 ---

$ exec $SHELL -l

# タイムゾーンをJSTに設定
$ sudo dpkg-reconfigure tzdata
## TUI で設定: Asia > Tokyo

# 日本語 man (コマンドマニュアル) をインストール
$ sudo apt install -y manpages-ja manpages-ja-dev
```

## Windows Terminalの設定

色々設定できるが、開始ディレクトリをWSLのユーザーホーム（`C:\Users\USERNAME`）にすると便利。

# WSL2でGUIを使えるようにする

windows11ではwslgが使えるが、ここではVcXsrvを用いて環境構築を行う。  
参考サイト：

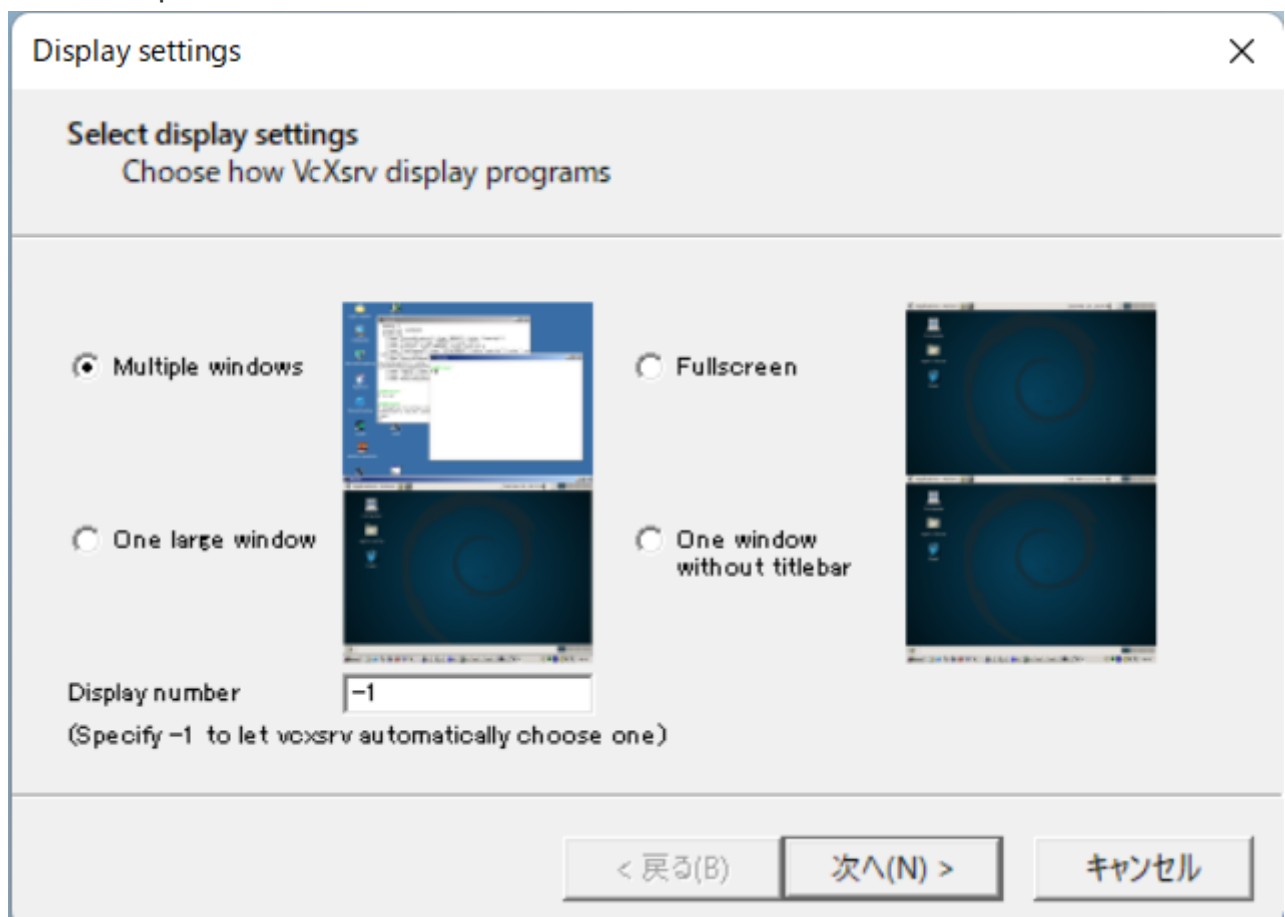
- [Windowsでputtyを使わずにX11 Forwardingをする。](#)
- [WSL2でGUIアプリケーションを使えるようにする](#)

## VcXsrvのインストール

[ダウンロードリンク](#)

## VcXsrvの起動時のダイアログ設定

- Select display settings: Multiple Display
- Select how to start clients: Start no client
- Extra settings:
  - [x]Clipboard (Primary Selection)
  - [x]Native opengl
  - [ ]Disable access control
  - Additional parameters for VcXsrv: -ac



Client startup

Select how to start clients

☒ Start no client

This will just start the xserver. You will be able to start local clients later.

☐ Start a program

This will start a local or remote program which will connect to the xserver. You will be able to start local clients later too. Remote programs are started using SSH.

☐ Open session via XDMCP

This will start a remote XDMCP session. Starting local clients later is limited. This option is not available with the "Multiple windows" mode.

< 戻る(B)

次へ(N) >

キャンセル

Extra settings

Extra settings

☒ Clipboard

Start the integrated clipboard manager

☒ Primary Selection

Also map the PRIMARY selection to the windows clipboard.

☒ Native opengl

Use the native windows opengl library (wgl). Make sure to export the LIBGL\_ALWAYS\_INDIRECT environment variable.

☐ Disable access control

Use this when you want vvxsrv to accept connections from all clients.

Additional parameters for VcXsrv

-ac

< 戻る(B)

次へ(N) >

キャンセル

ファイヤーウォールの設定→パブリックネットワーク

## WSL側の設定

Xorg GUI環境をインストール

```
sudo apt update && sudo apt install -y libgl1-mesa-dev xorg-dev
```

DISPLAY環境変数をWindows側のIPにする

```
sudo tee -a ~/.bashrc << \E0S
# WSL2 VcXsrv setting
export DISPLAY=${(cat /etc/resolv.conf | grep nameserver | awk '{print $2}'):0.0
E0S
```

ssh\_configを編集する。

```
sudo nano /etc/ssh_config
```

行末に下記を追加

```
# Host * の部分に追加
GSSAPIAuthentication yes
ForwardAgent yes
ForwardX11 yes
XAuthLocation /usr/bin/xauth
ForwardX11Trusted yes
```

シェルを再起動

```
exit
```

## VcXsrvのスタートアップへの登録

編集中

## 今までで発生したトラブル事例

- 0x800720efd→更新プログラムが実行中
- 0x800701bc→カーネルをアップデートする ([ダウンロード先](#))