

X Window System で遊んだ

～嗚呼、素晴らしきかな X Window System～

Endered

2023 年 06 月 23 日

自己紹介

- ▶ 名前: Endered
- ▶ 所属: 会津大学大学院 1 年
- ▶ 趣味: Lisp / 言語処理計
- ▶ 好きなもの: 関数型言語 / Linux
- ▶ その他: スライドを LATEX で書いてみました (楽しかった)

自己紹介

- ▶ 名前: Endered
- ▶ 所属: 会津大学大学院 1 年
- ▶ 趣味: Lisp / 言語処理計
- ▶ 好きなもの: 関数型言語 / **Linux** ← なぜ好きなのか？
- ▶ その他: スライドを LATEX で書いてみました (楽しかった)

なぜ Linux が好きなのか

色々理由はありますが、特筆するとしたら

なぜ Linux が好きなのか

色々理由はありますが、特筆するとしたら
X Window System があるから

X Window System とは

GUI に関する機能を提供するための通信規格 & その実装

- ▶ 提供している機能
 - ▶ 画面にウィンドウを描画
 - ▶ マウスを動かす & クリックの処理
 - ▶ キーボード入力を処理

X Window System とは

GUI に関する機能を提供するための**通信規格** & その実装

- ▶ 提供している機能
 - ▶ 画面にウィンドウを描画
 - ▶ マウスを動かす & クリックの処理
 - ▶ キーボード入力を処理
- ▶ 通信規格で定義されているので非常に便利

すごいぞ X Window System(X 転送について)

X 転送とは

- ▶ 我々の感覚としては画面転送 (VNC) が近いです
- ▶ ただし、描画命令をネットワーク越しに送っている

実際にX転送を使ってみよう

以下のコマンドを実行するだけ

emacs をローカルに転送する例

```
ssh -Y "適当なサーバ" emacs
```

X 転送の何がすごいのか

- ▶ X 転送は X Window System の**描画命令**を転送している
 - ▶ VNC などのソフトウェアは描画結果を転送している

X 転送の何がすごいのか

- ▶ X 転送は X Window System の**描画命令**を転送している
 - ▶ VNC などのソフトウェアは描画結果を転送している
- ▶ つまり、X 転送では (理論上)FPS が遊べる
 - ▶ VNC だと: マウスカーソルが上手に処理できなくて画面が吹っ飛ぶ
 - ▶ X 転送だと: 普通に遊べる (GPU で描画する系は遅い)

X 転送の何がすごいのか

- ▶ X 転送は X Window System の**描画命令**を転送している
 - ▶ VNC などのソフトウェアは描画結果を転送している
- ▶ つまり、X 転送では (理論上)FPS が遊べる
 - ▶ VNC だと: マウスカーソルが上手に処理できなくて画面が吹っ飛ぶ
 - ▶ マウスカーソルの位置を転送しているから
 - ▶ X 転送だと: 普通に遊べる (GPU で描画する系は遅い)
 - ▶ マウスカーソルの移動イベントを転送しているから

X 転送の何がすごいのか

- ▶ X 転送は X Window System の**描画命令**を転送している
 - ▶ VNC などのソフトウェアは描画結果を転送している
- ▶ つまり、X 転送では (理論上)FPS が遊べる
 - ▶ VNC だと: マウスカーソルが上手に処理できなくて画面が吹っ飛ぶ
 - ▶ マウスカーソルの位置を転送しているから
 - ▶ X 転送だと: 普通に遊べる (GPU で描画する系は遅い)
 - ▶ マウスカーソルの移動イベントを転送しているから
- ▶ リモートで起動したソフトウェアをローカルで動いているかのように使える

ssh -Y "サーバ名" がやっていること

注意

- ▶ ここではリモートの\$DISPLAYを x 、ローカルの\$DISPLAYを y だとします。
- ▶ また、環境によっては動かない場合があります。

ssh -Y "サーバ名" がやっていること

注意

- ▶ ここではリモートの\$DISPLAYを x 、ローカルの\$DISPLAYを y だとします。
- ▶ また、環境によっては動かない場合があります。

1. リモートのポート $6000 + x$ をローカルの $6000 + y$ にフォワーディングする

ssh -Y "サーバ名" がやっていること

注意

- ▶ ここではリモートの\$DISPLAYを x 、ローカルの\$DISPLAYを y だとします。
- ▶ また、環境によっては動かない場合があります。

1. リモートのポート $6000 + x$ をローカルの $6000 + y$ にフォワーディングする
2. ローカルのディスプレイ y へ描画するのに必要な鍵を取得する

ssh -Y "サーバ名" がやっていること

注意

- ▶ ここではリモートの\$DISPLAYを x 、ローカルの\$DISPLAYを y だとします。
- ▶ また、環境によっては動かない場合があります。

1. リモートのポート $6000 + x$ をローカルの $6000 + y$ にフォワーディングする
2. ローカルのディスプレイ y へ描画するのに必要な鍵を取得する
3. リモートの x へ描画する際に先程取得した鍵を使うようにする

ssh -Y "サーバ名" がやっていること

注意

- ▶ ここではリモートの\$DISPLAYを x 、ローカルの\$DISPLAYを y だとします。
- ▶ また、環境によっては動かない場合があります。

1. リモートのポート $6000 + x$ をローカルの $6000 + y$ にフォワーディングする
2. ローカルのディスプレイ y へ描画するのに必要な鍵を取得する
3. リモートの x へ描画する際に先程取得した鍵を使うようにする
4. 以上 (恐らく)

X 転送を自分でやってみる

実践してみましょう

- ▶ リモートのポート $6000 + x$ をローカルの $6000 + y$ にフォワーディングする
- ▶ ローカルのディスプレイ y へ描画するのに必要な鍵を取得する
- ▶ リモートの x へ描画する際に先程取得した鍵を使うようにする

X 転送を自分でやってみる

実践してみましょう

- ▶ リモートのポート $6000 + x$ をローカルの $6000 + y$ にフォワーディングする
 - ▶ `ssh -R 6123:/tmp/.X11-unix/X0`
 - ▶ ※ ここでは、フォワーディング先をポートではなくソケットにしている
- ▶ ローカルのディスプレイ y へ描画するのに必要な鍵を取得する
- ▶ リモートの x へ描画する際に先程取得した鍵を使うようにする

X 転送を自分でやってみる

実践してみましょう

- ▶ リモートのポート $6000 + x$ をローカルの $6000 + y$ にフォワーディングする
 - ▶ `ssh -R 6123:/tmp/.X11-unix/X0`
 - ▶ ※ ここでは、フォワーディング先をポートではなくソケットにしている
- ▶ ローカルのディスプレイ y へ描画するのに必要な鍵を取得する
 - ▶ `xauth list $DISPLAY | grep -oP '\w*$'`
- ▶ リモートの x へ描画する際に先程取得した鍵を使うようにする

X 転送を自分でやってみる

実践してみましょう

- ▶ リモートのポート $6000 + x$ をローカルの $6000 + y$ にフォワーディングする
 - ▶ `ssh -R 6123:/tmp/.X11-unix/X0`
 - ▶ ※ ここでは、フォワーディング先をポートではなくソケットにしている
- ▶ ローカルのディスプレイ y へ描画するのに必要な鍵を取得する
 - ▶ `xauth list $DISPLAY | grep -oP '\w*$'`
- ▶ リモートの x へ描画する際に先程取得した鍵を使うようにする
 - ▶ `xauth add $DISPLAY . "先程取得した鍵"`
 - ▶ この際、`$DISPLAY` を `:123` に設定する

動いた～

作ったものの紹介

X Window System を理解するためにちょっとしたソフトウェアを作ってみました

作ったものの紹介

X Window System を理解するためにちょっとしたソフトウェアを作ってみました
その名も *Space Ship Cursor*

操作説明

ソフトを起動すると宇宙船を模したカーソルが出現するよ

- ▶ WS で前進/後進、AD で左右に回転！
- ▶ Jでマウスクリック！

操作説明

ソフトを起動すると宇宙船を模したカーソルが出現するよ

- ▶ WS で前進/後進、AD で左右に回転！
- ▶ Jでマウスクリック！

ということでマウスを使用するゲームでも遊ぼうと思います