

CS 349

XWindows Setup

Using C++

Working remotely in Linux

Working Remotely with XWindows

X Windows is designed to

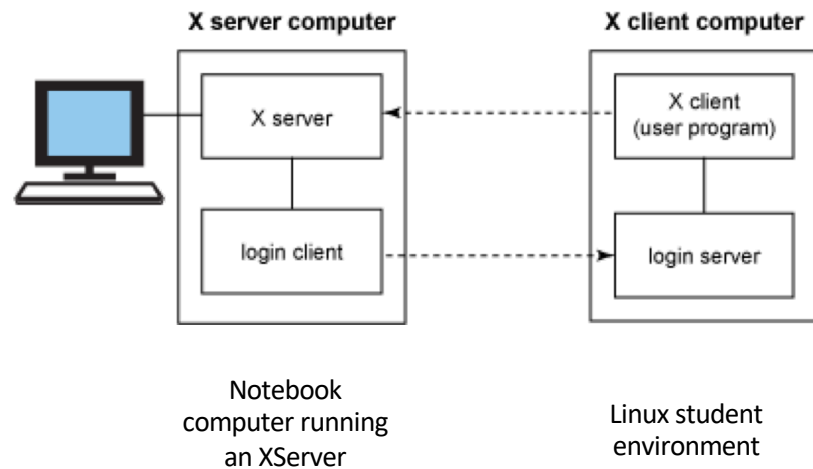
- compile code and run code on one machine (the X Client) and
- display the output on a second machine (the X Server).

X Client

- Must have X Windows installed.
- Must be a Unix-based system (Linux or Mac). Cannot run on Windows.
- The student environment (e.g. `linux.student.cs.uwaterloo.ca`) is already setup as an XClient.

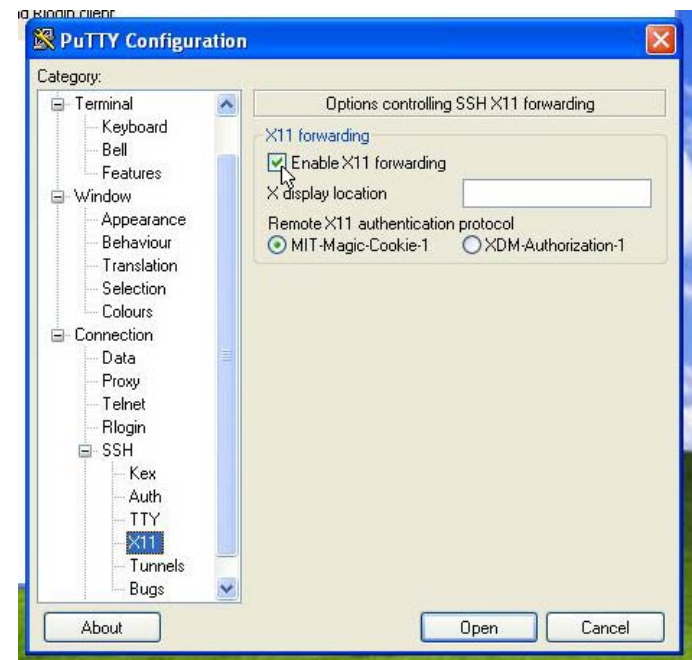
X Server

- Doesn't need to have X Windows.
- Just requires X Server software.
 - Linux: already installed
 - Mac: install XQuartz
 - Windows: install Xming.



Setup: Microsoft Windows

- Install an Xserver
 - Recommend Xming (<https://sourceforge.net/projects/xming/>)
 - Launch before proceeding.
- Install an SSH client
 - Install Putty (<https://www.putty.org>)
 - Connection/SSH/X11
 - Enable X11 forwarding
- Connect
 - Username: <your WatID>
 - Password: <your password>
 - Server: linux.student.cs.uwaterloo.ca



Setup: macOS

- Install an Xserver
 - Recommend XQuartz (<http://www.xquartz.org>).
 - Don't need to launch it! (will launch automatically)
- Install an SSH client
 - SSH is built-in, you can just use the terminal.
- Connect from the terminal:
 - `ssh -Y jdoe@ linux.student.cs.uwaterloo.ca`

Setup: Linux

- You should already have everything installed
 - You probably already have an XServer running
 - You *may* need to install `g++` and `libx11-dev` packages
- Connect from the terminal:
 - `ssh -Y jdoe@ linux.student.cs.uwaterloo.ca`

Compile and run programs

You need to compile on the X client system that has X Windows installed (since you need libraries, headers etc.). These steps assume you're using the Linux student environment to compile and execute.

Connect using the instructions on the previous page.

Compile with appropriate flags, then run executable.

```
$ g++ -o main main.cpp -L/usr/X11R6/lib -lX11 -lstdc++  
./main
```

The output should appear on your local machine.