Virtual Desktops for HPC

2021-02-11

New Zealand eScience Infrastructure

Is this you?

I want to visualise a dataset too large for my local machine!

My HPC workflow requires lots of uploading/downloading!

I need to monitor my HPC job in real time!

Is this you?

I don't like command line.

I really don't want to learn command line, please don't make me learn command line, it looks really hard. Oh gosh please no, what does any of that mean, what even is a linux, I'm a busy PhD student, I haven't slept in two years, it's all too much, I don't want to hack any mainframes I just want to finish this simulation before I die, I just need a nice comfortable windows environment that doesn't also melt my laptop, the university issued this laptop and I don't know what they'd do if I made it into a plastic puddle, maybe

Challenges

Provide a Desktop Environment Suitable for use on a HPC Virtual DeskTop (VDT)

Performance
- High bandwidth.
- Low latency.

Accessibility
- Shouldn't require (too much) setup.
- Familiar Desktop Environment.

Utility
- Maintainable.
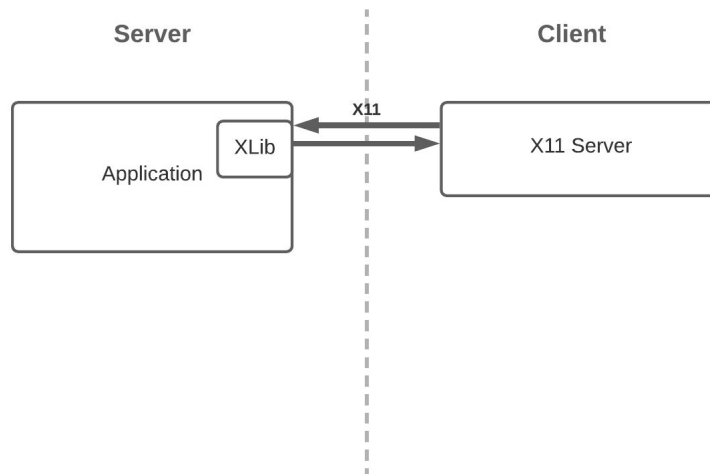- Customisable.

Scheduling
- Persistence, disconnecting, reconnecting.
- Use any resources available to SLURM

Challenges
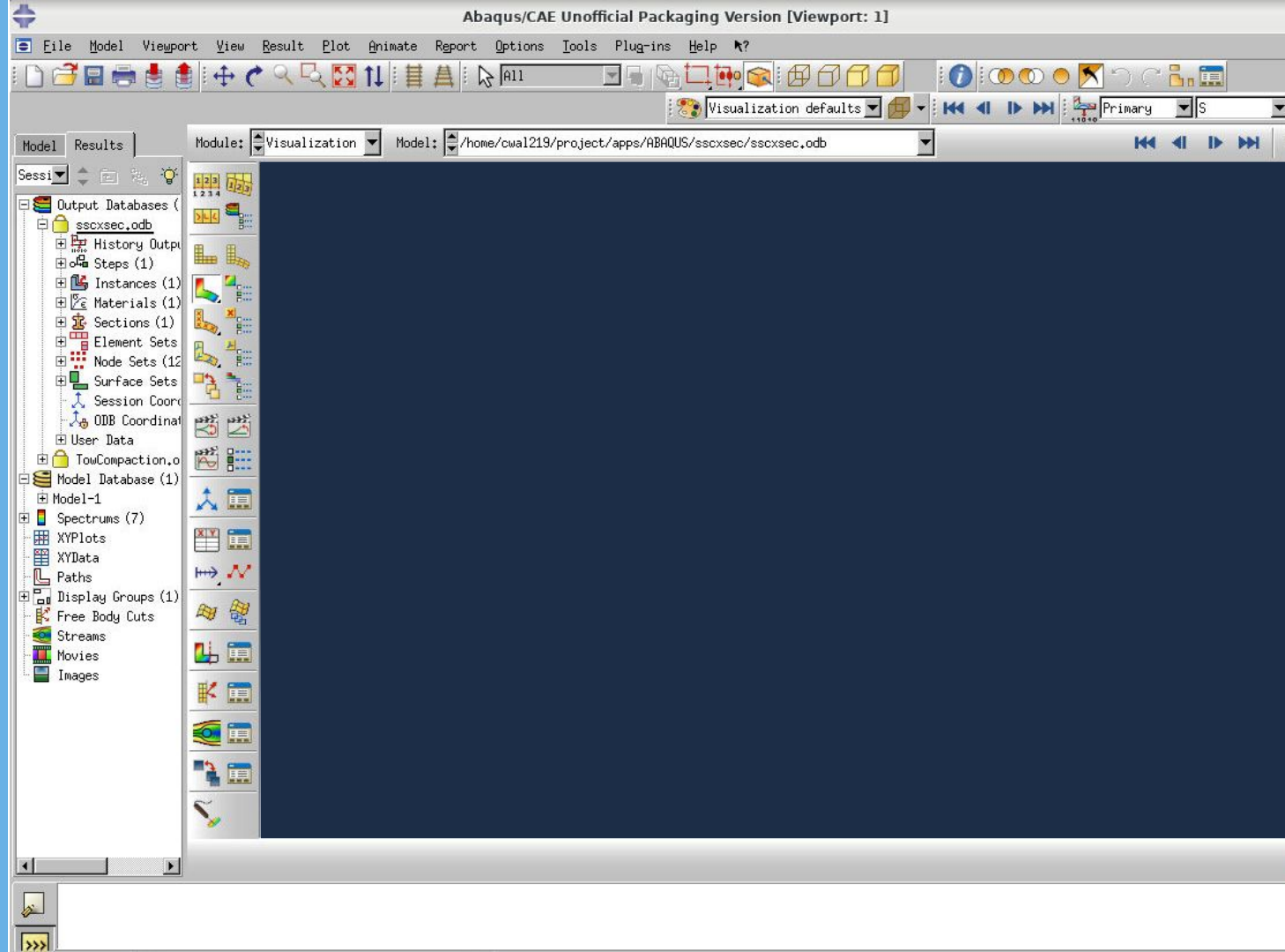
- Performance

Regular X11

X11 server sends user input across network, receives instructions for drawing GUI.

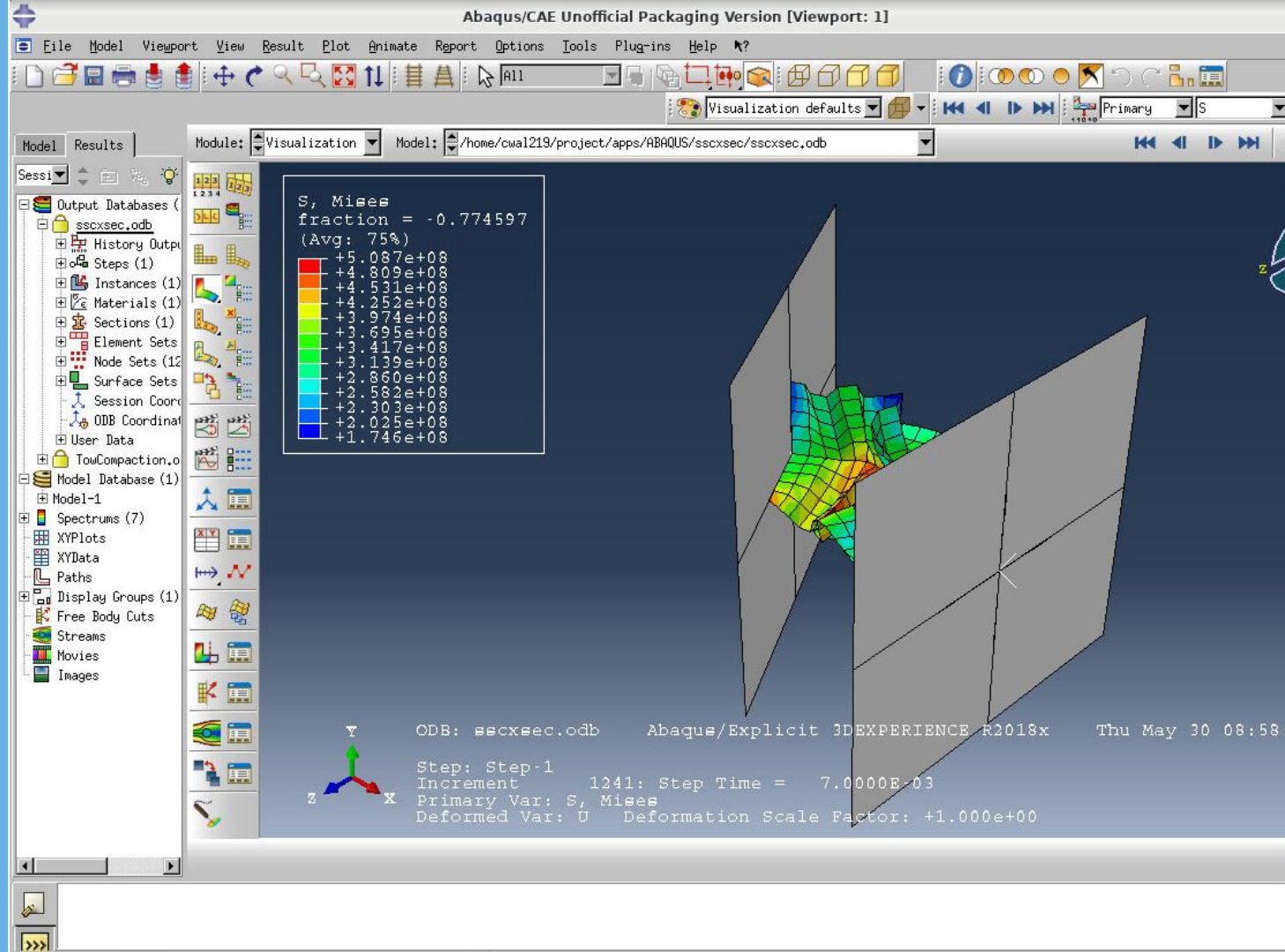- Heavy use of bandwidth, high latency.

Challenges

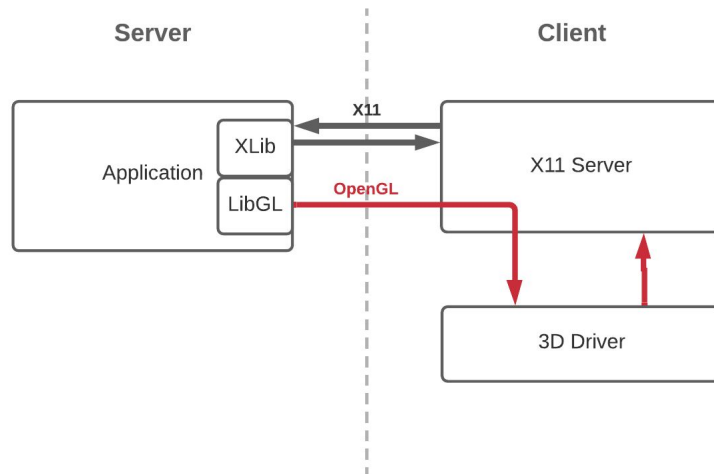- Performance

Challenges

- Performance

Challenges

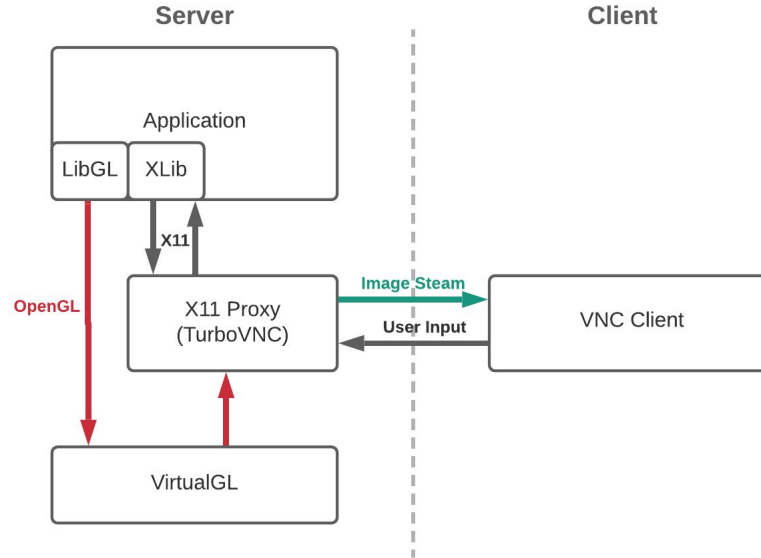- Performance

X11 with Indirect Rendering

OpenGL rendering instructions wrapped by X11 protocol sent across network to be rendered local graphics driver.

- Client machine must have the resources to render the OpenGL commands.
- Even more use of bandwidth, higher latency.

Challenges

- Performance

TurboVNC + VirtualGL

Server   **Client**

Application

| LibGL | XLib |

X11

OpenGL

X11 Proxy
(TurboVNC)

Image Steam →

User Input ←
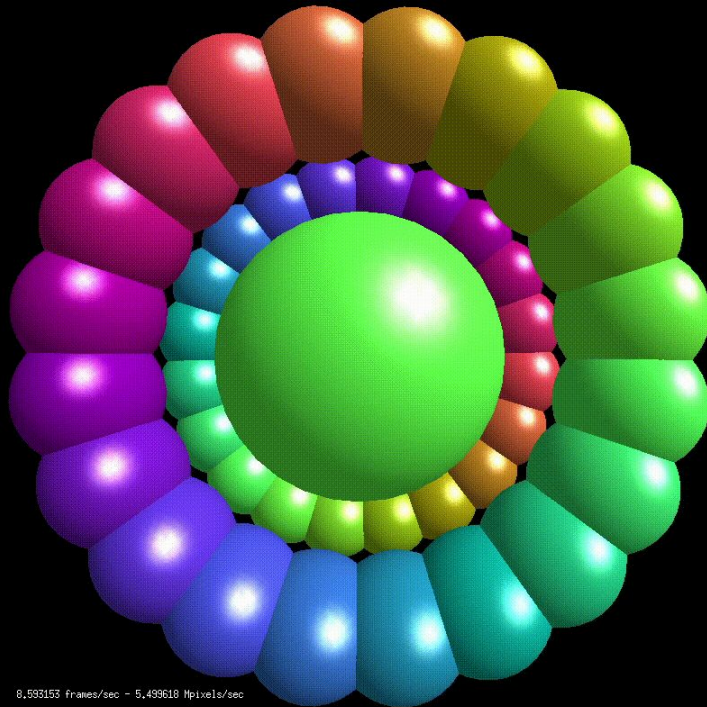
VNC Client

VirtualGL

X11 Drawing and OpenGL rendering done server side

Allows fast communication between Hardware/Drivers, Application and Display

Output from X11 proxy downsampled and compressed before being sent across network.

Challenges

- Performance

8.593153 frames/sec - 5.499618 Mpixels/sec

28.247675 frames/sec - 18.055914 Mpixels/sec

≈ 8 f/s   <   ≈ 32 f/s

Challenges
- Accessibility

Application

LibGL   XLib

X11

OpenGL

X11 Proxy (TurboVNC)

Image Steam

User Input

VirtualGL

Lite Server (Websockify)

VNC Client (noVNC)

Web Browser
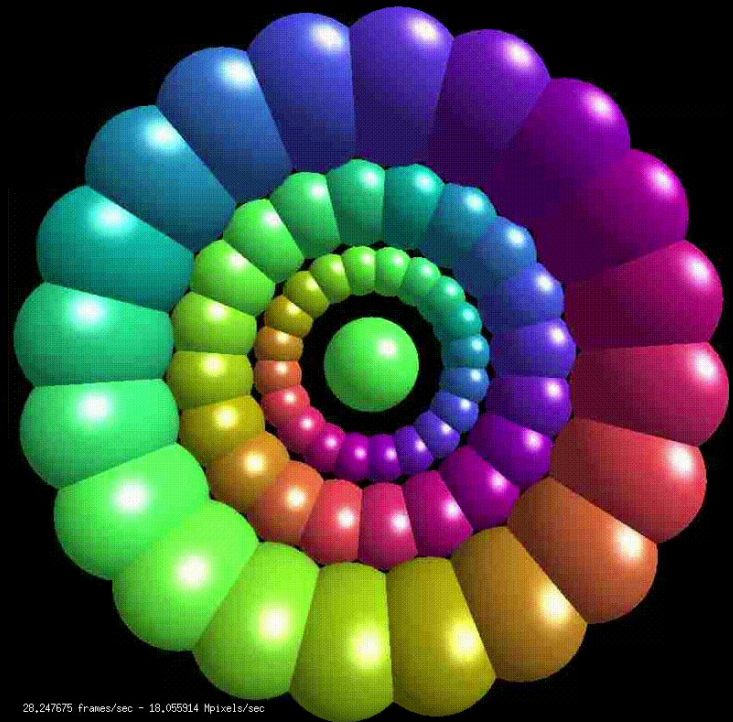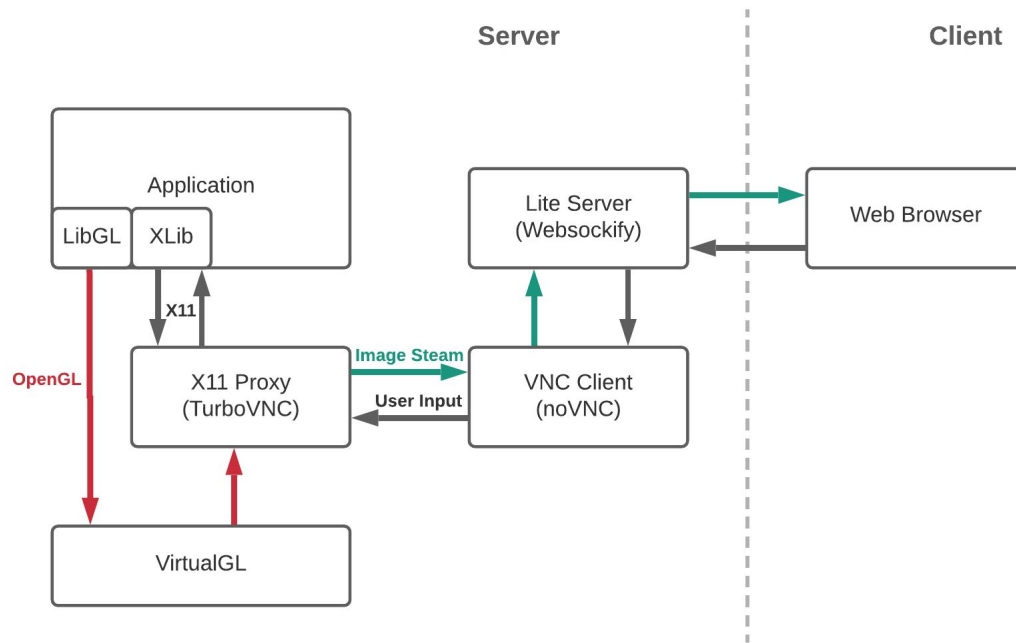
X11 Drawing and OpenGL rendering done server side

Challenges

- Accessibility

Applications   13:25   Callum W

XFCE Window Manager

Not a VM!

Set of Familiar Tools

ANSYS System Cou...

Creates Symlinks to Project Directories

COMSOL 5.5

CFX

Setup Paths and Icons for Commonly used Software
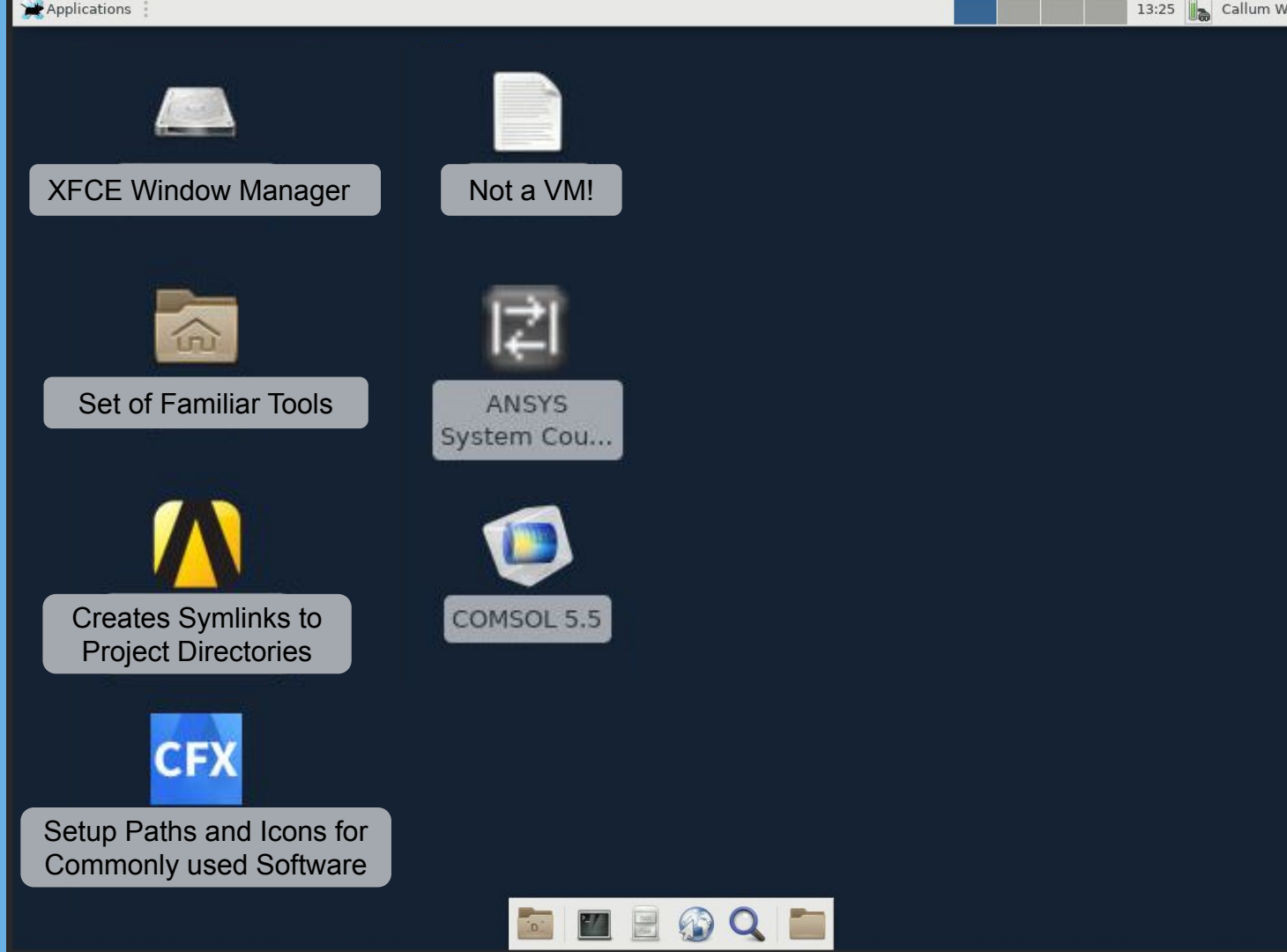
Challenges

- Utility

Things are getting complicated...

- Lots of dependencies
- Fragile
- Difficult to extend
- Not especially portable

Challenges

- Utility

Containerisation!

- Allows full(ish) control of environment. (No root required).

- Can use your own software install, or use NeSI software stack using lmod.

- VDT container can be used as base image.
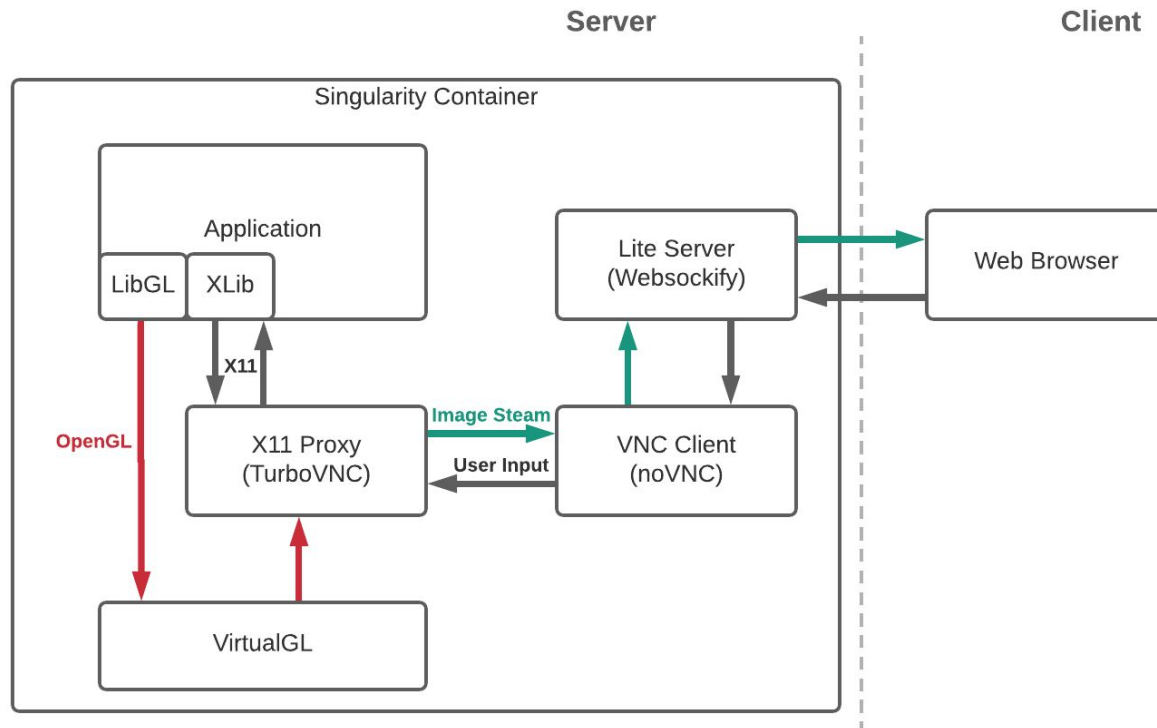
- Integrated with SLURM.
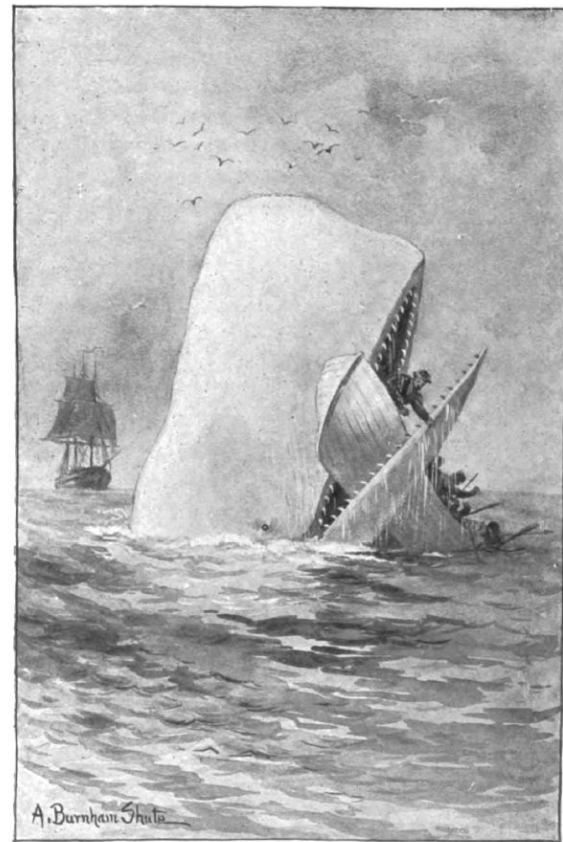
" Both jaws, like enormous shears, bit the craft completely in twain."

—Page 510.

A. Burnham Shute

Challenges
- Utility

Challenges
- Scheduling

- Launch on login node / gateway
- Launch within existing SLURM job
- Launch within new SLURM job
- Launch within Jupyter Lab

- Die when your connection closes
- Don't do that

Challenges

- Scheduling

Server

Client

Login Node / Gateway

Singularity Container

Application
Lib & util

Image Steam

X11
OpenGL

VirtualGL
(TurboVNC)

Lite
VNC

User Input

VNC
Client
(noVNC)

(Websockify)

Web Browser

```
vdt start <port>
```

Challenges

- Scheduling

Server

Client

Login Node / Gateway

Singularity Container

Application

Lib GL update Steam

OpenGL

X11

User Input

VirtualGL
(TurboVNC)

Lite
VNC

Client
(noVNC)

(Websockify)

Web Browser
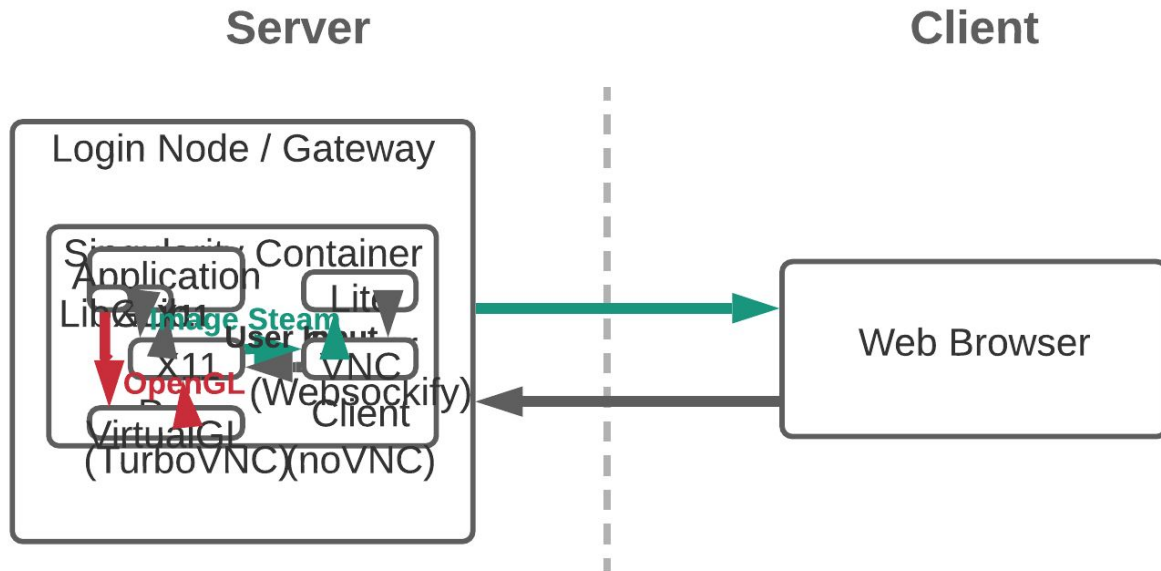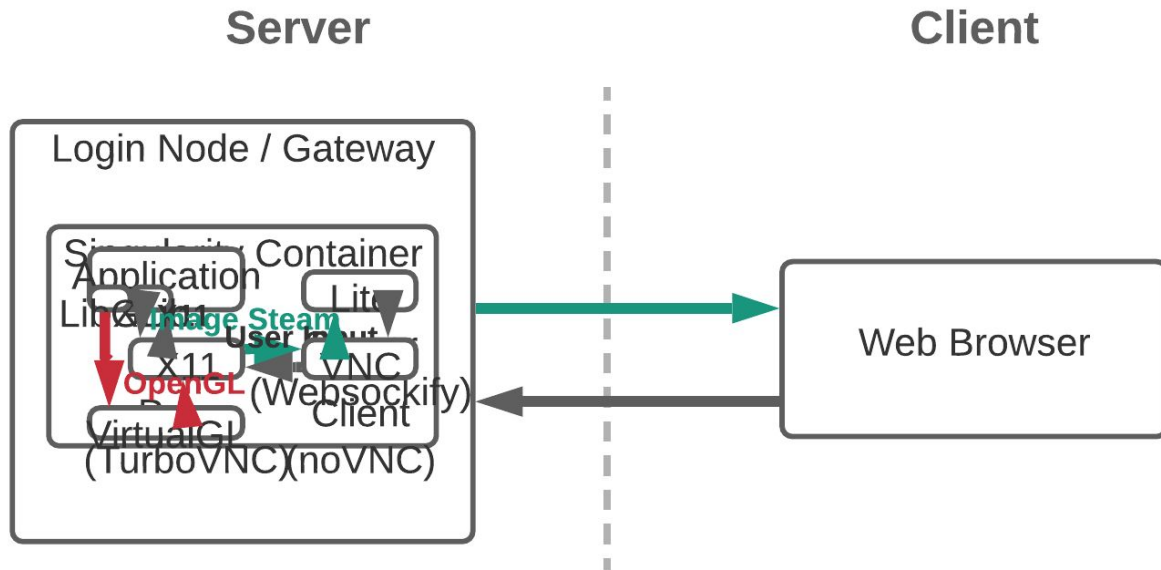
```
vdt start <port> &
vdt list
```

Challenges

- Scheduling

Server

Client

Login Node / Gateway

Web Browser

SLURM

Reverse SSH tunnel

Compute Node

Singularity Container
Application
Lib & image **Steam**
x11 **OpenGL**
VirtualGL
(TurboVNC)

Lite
User input
VNC
(Websockify)
Client
(noVNC)

```
salloc <slurm-flags> vdt start <port>
```
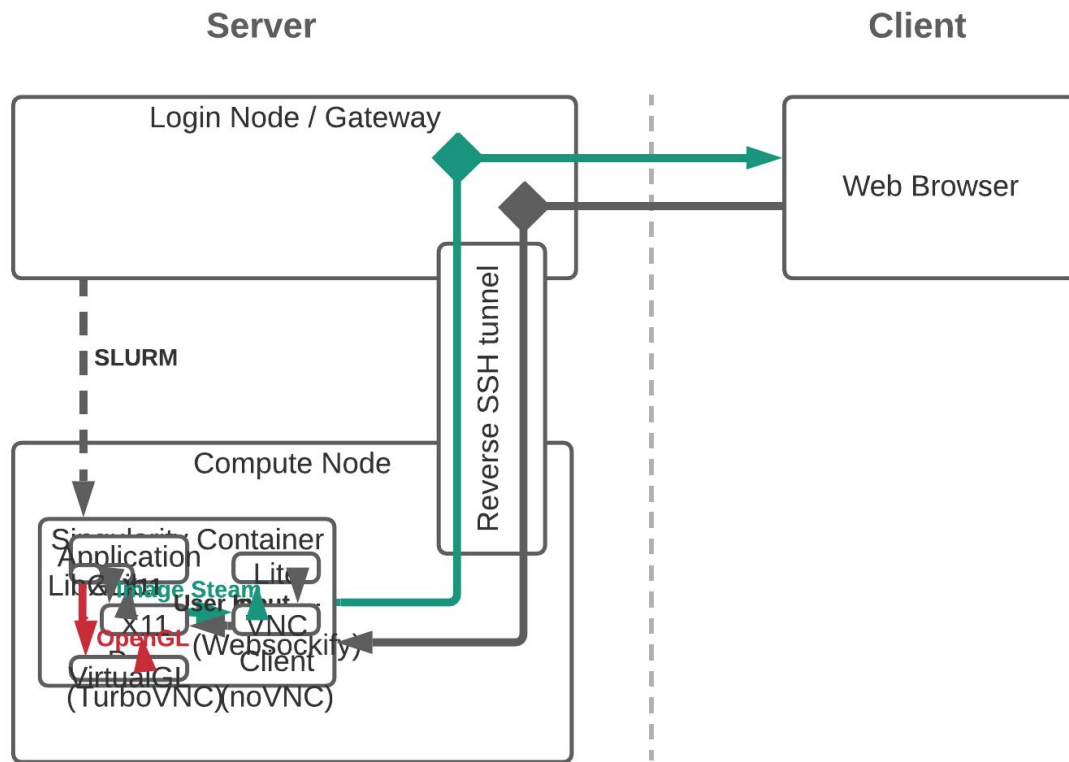
Challenges
- Scheduling


```
salloc <slurm-flags> vdt start <port> &
```

Challenges
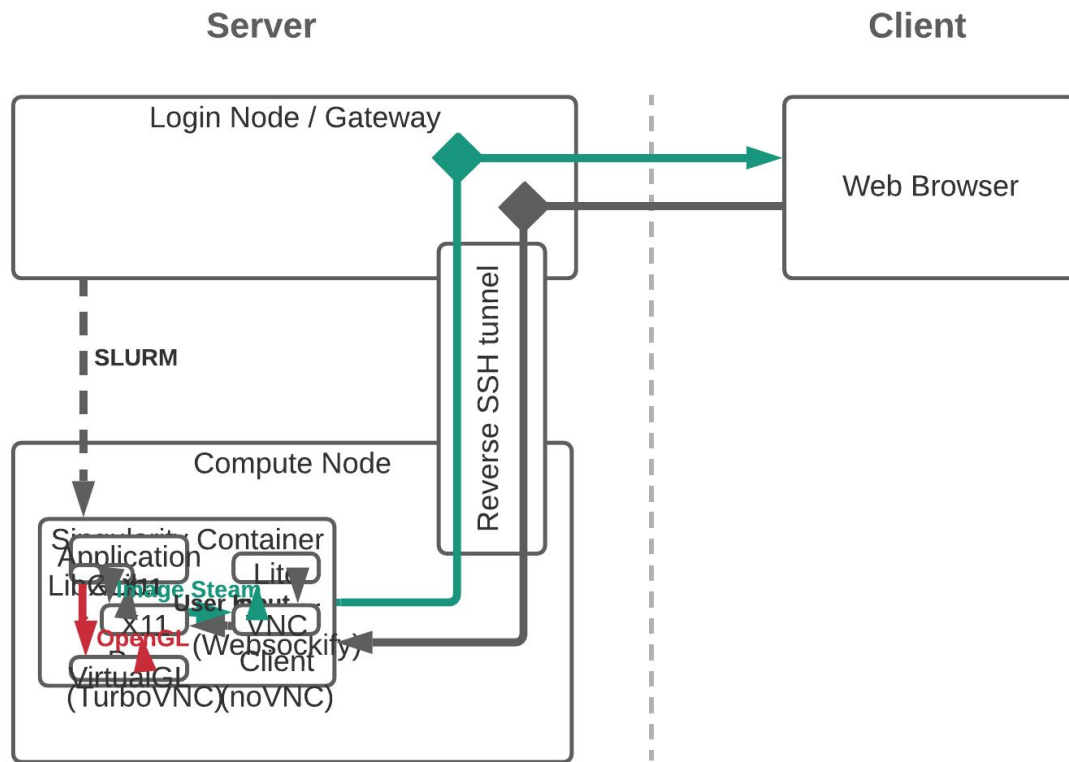
- Scheduling

What Next?

- Faster!
- GPGPUs for rendering purposes.
- Tweak popular software so that it works with hardware graphics straight away.
- Better integrate with other NeSI services.

- Still requires CMD line and an SSH connection for full functionality.
- Fragile, messy (can leave behind processes)
- Documentation

aaahh

Glossary

- Remote : A computer somewhere else (as opposed to Local)
- VNC: Virtual Network Computing,
  - Sending your keyboard/mouse inputs to remote computer.
  - Receiving images back.
- OpenGL: API for rendering 2D and 3D vectors.
- X11: Framework for providing window based GUIs, specifically designed to work over networks.
- GLX: Interface between OpenGL and X11.
- Mesa: An open source software implementation of OpenGL.
- VirtualGL: Program that redirects OpenGL commands to a proxy display.
- TurboVNC: VirtualGL implementation that makes use of image compression.
- noVNC: VNC client that runs in a web browser.
- Singularity: OCI compliant container software, popular on HPCs as it works without root.
- OCI: Open container initiative.
- SLURM: Scheduling system in use on NeSI

NZ Research Software Engineers Conference

Spring 2021

Help us plan the programme!
Email events@nesi.org.nz to get involved.

Who attends:
- Researchers and academics who code
- Software engineers & system admins working in the research domain
- Generalists who bring together the research and technical domains
- Developers, IT managers, coding enthusiasts, and big data analysts from Crown Research Institutes, universities, and other public sector organisations

NeSI @ eResearch NZ - Talks & Workshops:

Wednesday 10 Feb

13:00 - 17:00 - **Maxime Rio -** Machine Learning on NeSI 101

13:20 - 13:40 - **Jun Huh -** Taonga: building a data repository for genomics research in New Zealand

13:20 - 13:40 - **Dinindu Senanayake** - Paving the way for Bioinformatics excellence in New Zealand

14:20 -15:00 - **Brian Flaherty** - Moving data: getting up to speed with Globus and Science DMZ

15:50 - 16:50 - **Jana Makar** - Challenge Accepted: Responding to community feedback for supporting diversity in HPC & eResearch

Thursday 11 Feb

11:00 - 11:20 - **Maxime Rio** - Data science consultancies at NeSI: A whirlwind tour of case studies

13:30 - 13:50 - **Chris Scott** - GPUs on NeSI

13:50 - 14:10 - **Georgina Rae** - Building Partnerships for eResearch

14:10 - 14:30 - **Wolfgang Hayek** - NeSI Consultancies - Evolving a Scientific Programming Service

14:40 - 15:00 - **Albert Savary** - Software on NeSI

15:00 - 15:20 - **Jeff Zais** - Taking Advantage of Technology Innovations in the Next Generation of NeSI HPC Infrastructure

15:20 - 15:40 - **Callum Walley** - Virtual Desktops for HPC

Thursday 11 Feb (cont.)

15:20 - 15:40 - **Robin Bensley** - Staying connected in an evolving eResearch ecosystem

16:00 - 17:00 - **Megan Guidry** - Sowing the seeds of capability: Experience what Carpentries instructor training is all about

Friday 12 Feb

11:20 - 12:30 - **Nick Jones** - Future of eResearch

12:20 - 12:30 - **José Filipe Gonçalves Higino** - Coaching great practices of describing a problem

13:30 - 14:30 - **Blair Bethwaite** - Embracing cloud-native architectures

13:30 - 14:30 - **Alexander Pletzer and Nooriyah Lohani** - Who needs research software engineers?

13:30 - 14:30 - **Georgina Rae** - FAIR for Research Software

www.nesi.org.nz   support@nesi.org.nz

@NeSI_NZ   https://www.nesi.org.nz/apply

New Zealand eScience Infrastructure