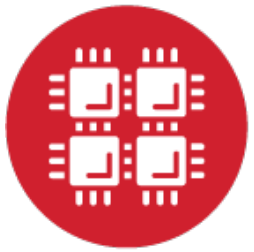


Ohio Supercomputer Center

An OH·TECH Consortium Member

SUG Breakout Session:
OSC OnDemand App Development

Basil Mohamed Gohar

Web and Interface Applications Manager

Eric Franz

Senior Engineer & Technical Lead

This work is supported by the National Science Foundation of the United States under the award NSF SI2-SSE-1534949.

Goals

1. Demystify OnDemand apps
2. Give an idea of how you can build them yourselves

Why build custom apps?

- Automate and streamline your use of HPC at OSC
- Reduce the learning time for new grad students by obviating the need to learn a lot about using OSC systems by providing a web interface instead
- Turn a multiple step process involving a single button click that can be initiated from any browser or even a phone

Open OnDemand Overview

- ✓ About Open OnDemand, OSC OnDemand, and AweSim
- How OnDemand and OnDemand Apps work
- App Development
  - System apps
  - Interactive apps
  - Gateway apps
- Deployment options
- Future support coming. Contact us to get started now.

About Open OnDemand

- Open source software project
- Installable for an HPC cluster (or clusters)
  - Standalone system, Lab, Department, University or National Resources
- More than just an “out of the box” solution
  - OnDemand is a platform that can be extended through additional and custom apps

Open OnDemand Features

- Single point of entry for HPC Center's services
- User needs three things
  - URL: ondemand.osc.edu or apps.awesim.org
  - Username
  - Password
- Zero install (Completely browser based)
- Single sign-on
- Firewall friendly (Keep traffic on https port)

OSC's OnDemand Installations

- OSC OnDemand: ondemand.osc.edu – has been focused on providing HPC access
- AweSim: apps.awesim.org – has been focused on custom OnDemand app development for HPC

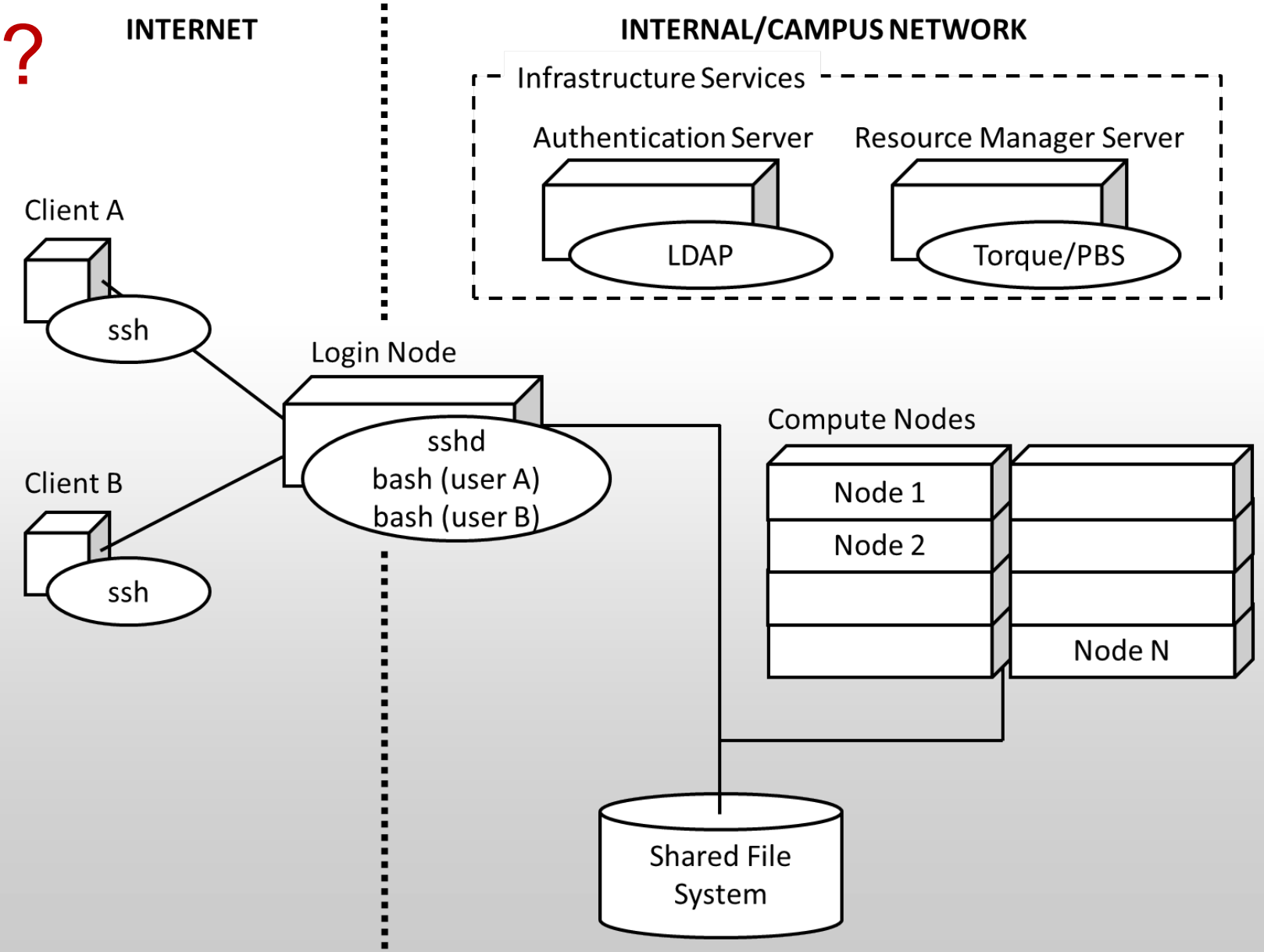
Open OnDemand Overview

- ✓ About Open OnDemand, OSC OnDemand, and AweSim
- ✓ How OnDemand and OnDemand Apps work
- App Development
  - System apps
  - Interactive apps
  - Gateway apps
- Deployment options
- Future support coming. Contact us to get started now.

How does it work?

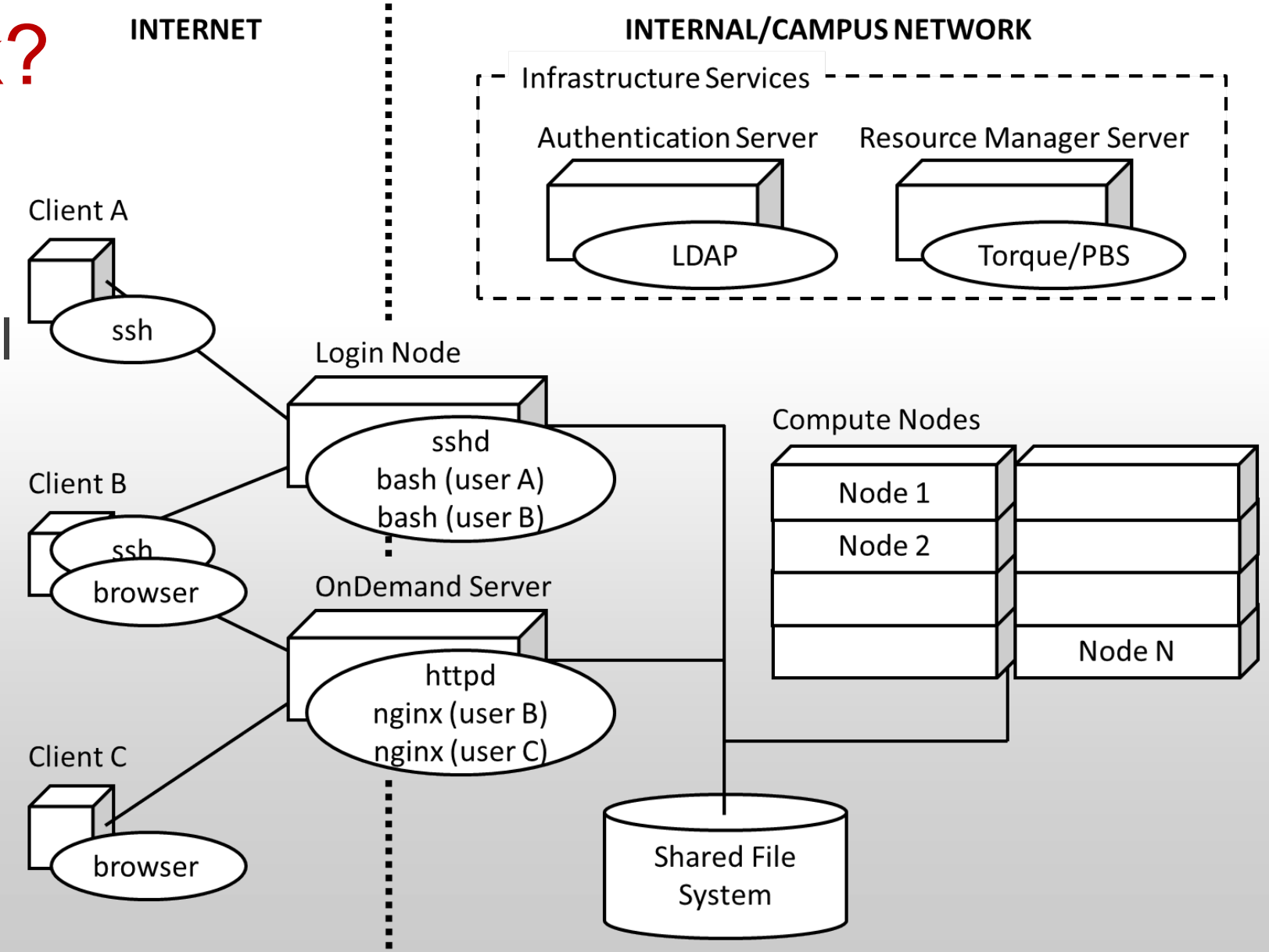
Our View of the World

- Clients connect to login node through SSH
- Interact with batch system and shared file system through the “bash” user process



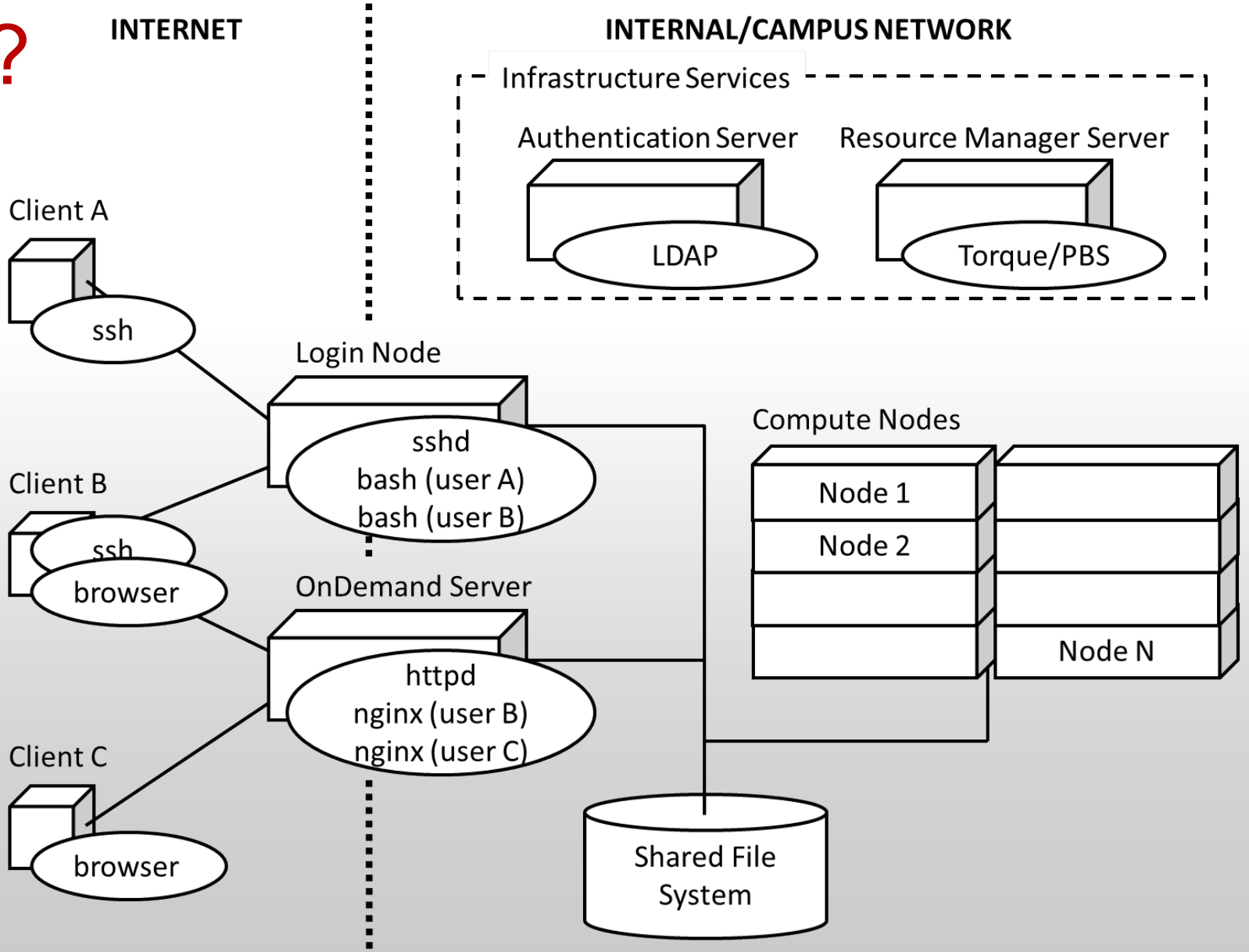
How does it work?

- We introduced an "OnDemand" server
  - Similar to a traditional login node
  - Same authentication
  - Talks to the same resource manager
  - Mounts the same shared file system



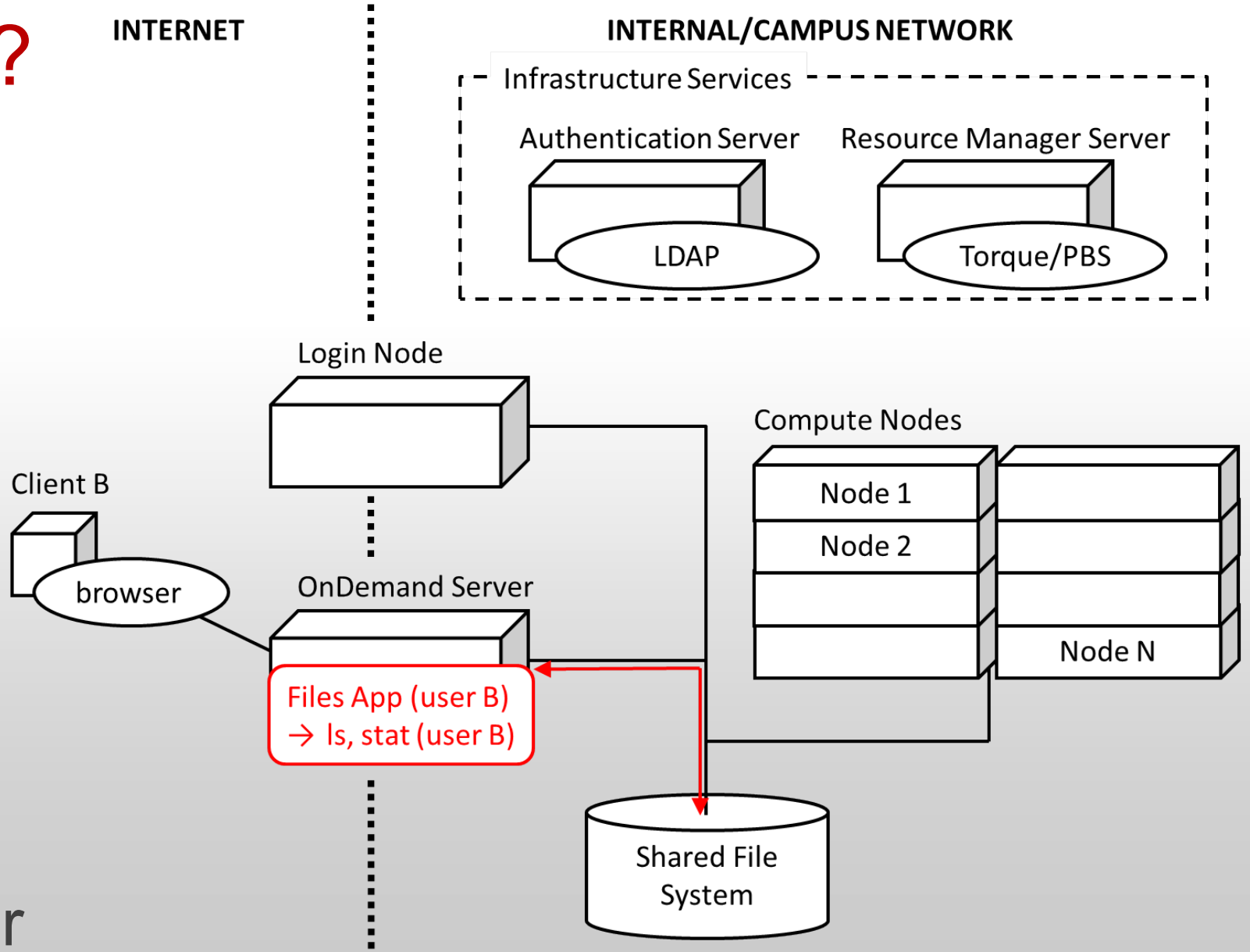
How does it work?

- Now clients may also connect to OnDemand server through browser
- The “httpd” proxy starts up a per-user nginx (PUN) process for each user
- The PUN launches apps as children processes



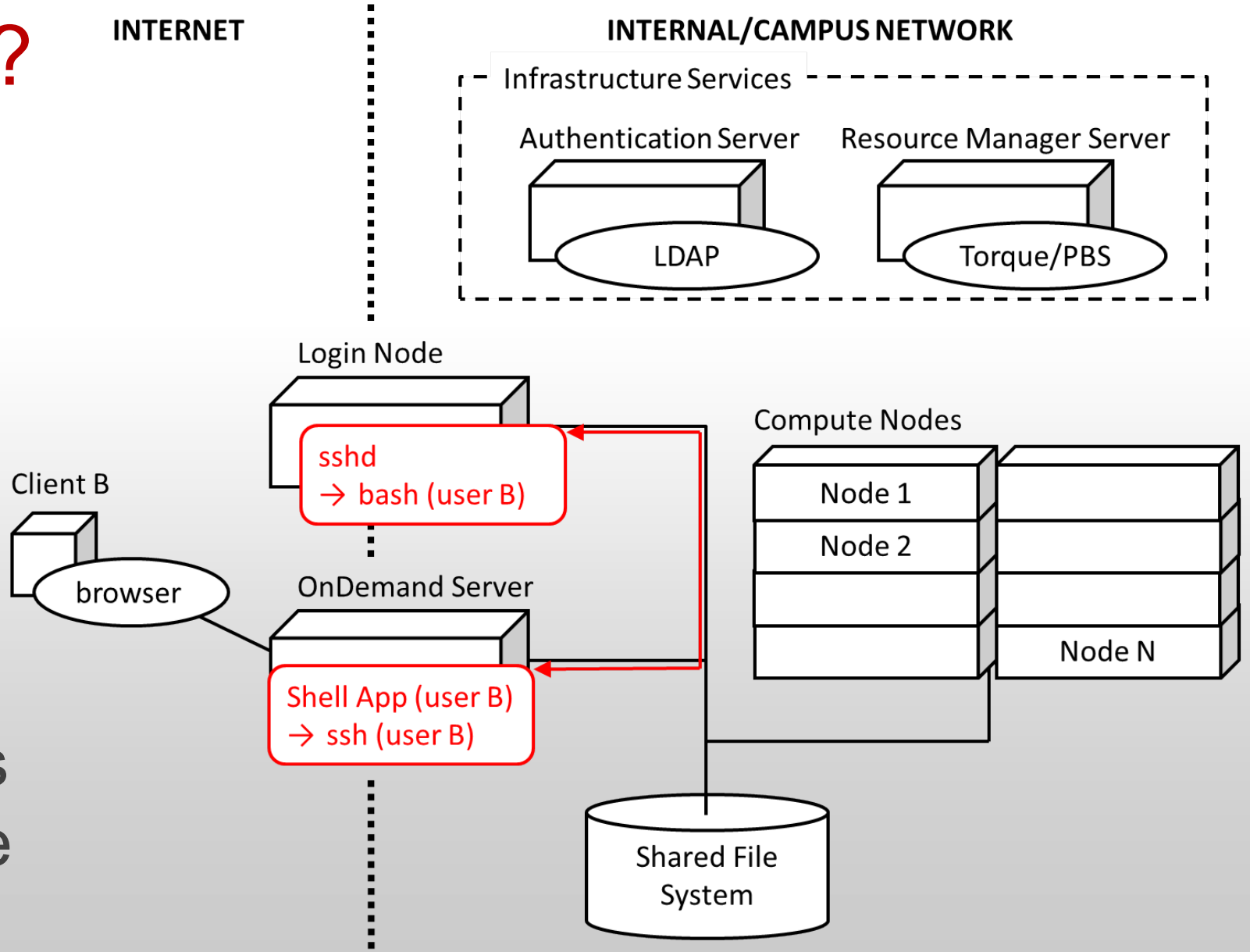
How does it work?

- The “Files App” launches as the user under the PUN
- Interacts with the shared file system through the Node.js “fs” core library
- File permissions are maintained as all processes run as user



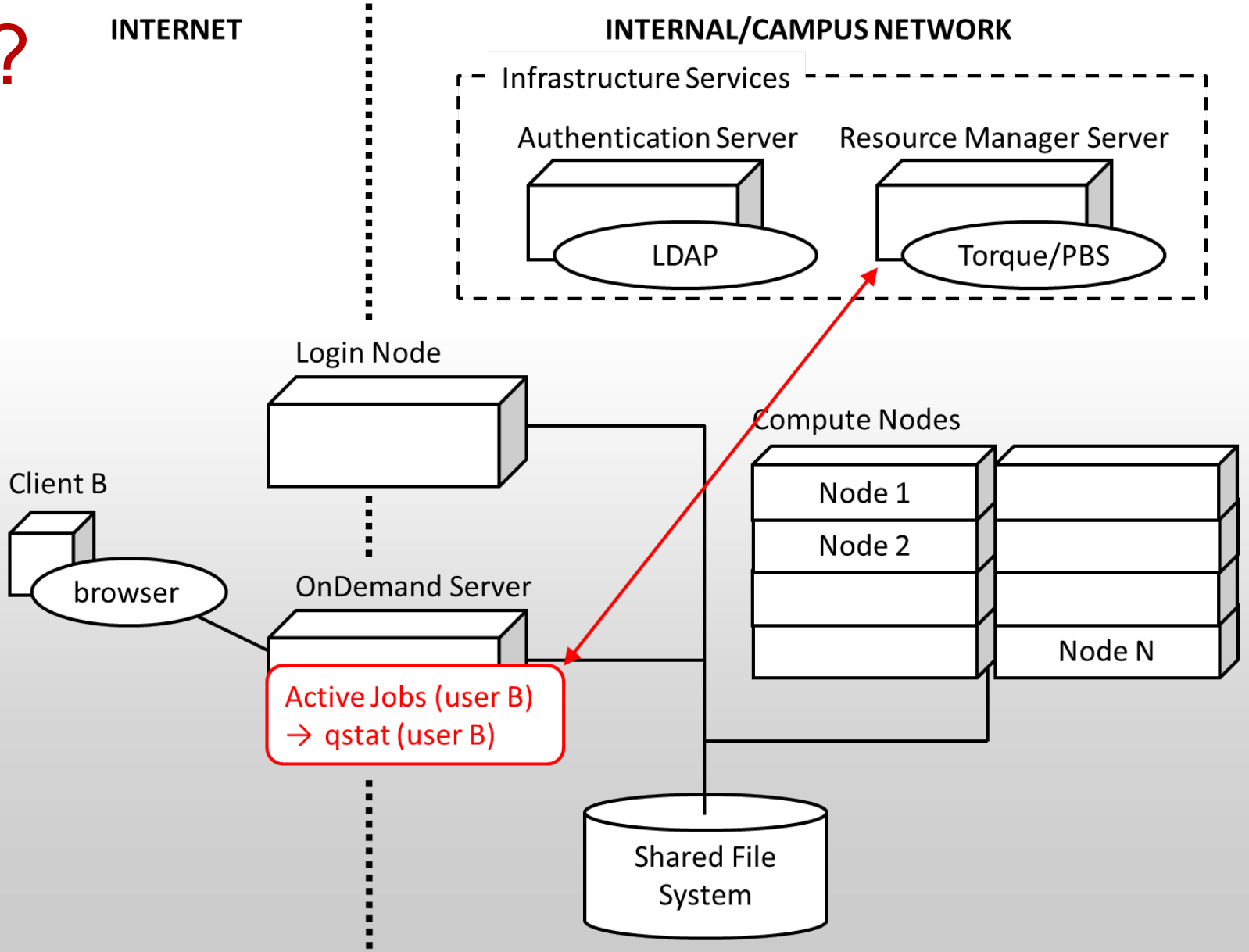
How does it work?

- The “Shell App” launches as the user under the PUN
- Within the shell app an “ssh” process is started connecting to the login node
- The browser behaves as the terminal for the “ssh” process



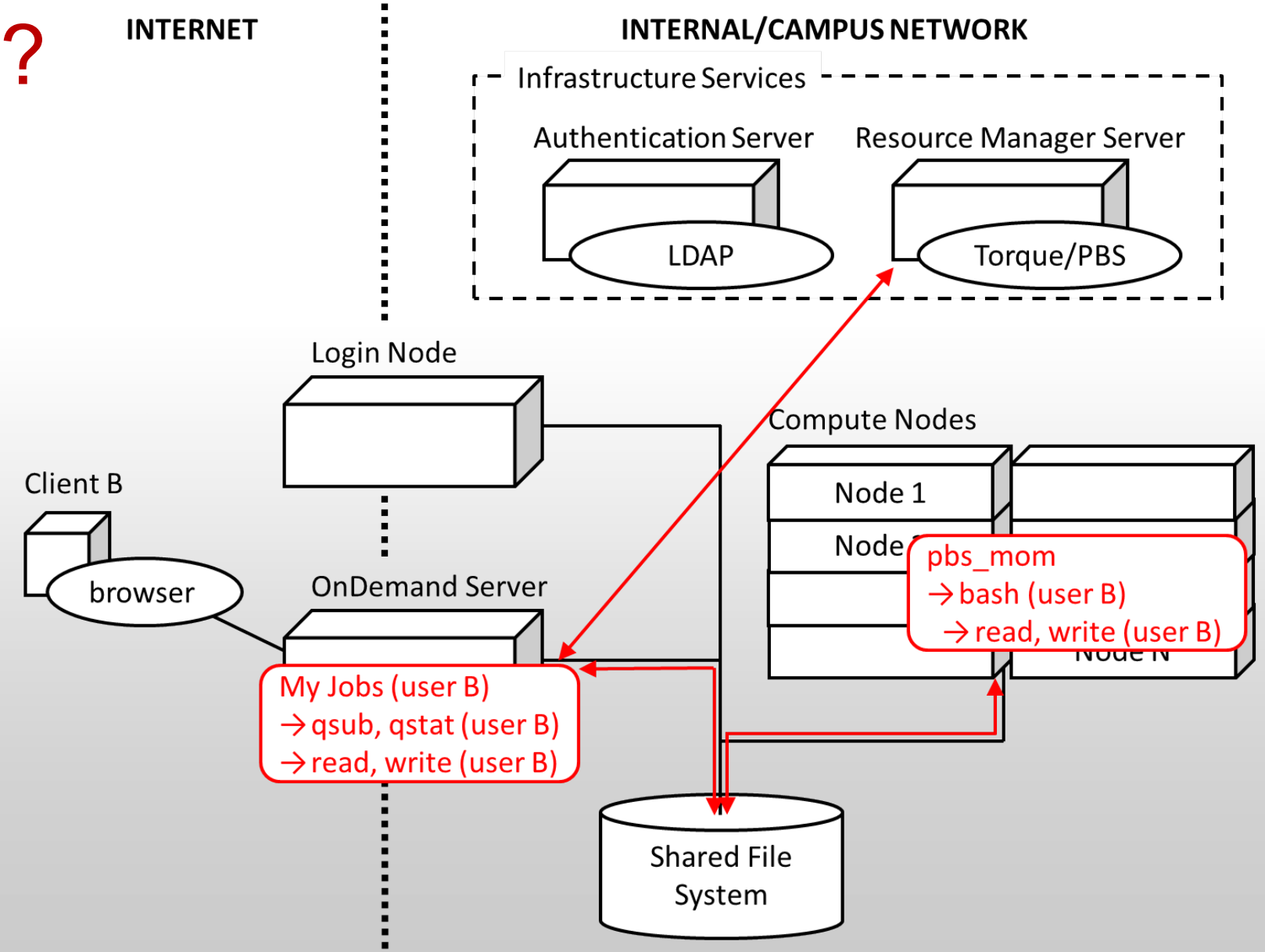
How does it work?

- The “Active Jobs App” launches as the user under the PUN
- Interacts with a Torque batch server
- Equivalent to a “qstat” command called by the user



How does it work?

- The “My Jobs App” launches as the user under the PUN
- Submits jobs to Torque batch server
- Writes input files and reads output files generated by batch jobs through shared file system



Open OnDemand Overview

- ✓ About Open OnDemand, OSC OnDemand, and AweSim
- ✓ How OnDemand and OnDemand Apps work
- ✓ App Development
  - System apps
  - Interactive apps
  - Gateway apps
- Deployment options
- Future support coming. Contact us to get started now.

System apps

- Examples include Shell, File Explorer, File Editor, Active Jobs, and System Status apps
- Provide cluster access, file access or editing, or live report on system activity
- They are web apps that Passenger application server can serve
  - Ruby, Python, or Node.js
  - Dependencies are installed in the app's deployment directory
- OnDemand provides a developer mode, currently accessible to some OSC users through apps.awesim.org, to assist in building these apps

Demo: ps to quota app

- Interactive Demo

Interactive Apps

- Examples include Desktop, Jupyter, Paraview, Matlab, RStudio
- All apps are “batch connect” plugins to the Dashboard Passenger app
- Workflow:
  - Submit a web form to start a batch job
  - Batch job starts server on compute node
  - User connects to server through OnDemand web interface
- 2 types of batch connect apps
  - VNC server (i.e. Desktop, Paraview, Matlab)
  - web server (i.e. Jupyter, RStudio)
- All OSC’s plugins code is publicly available
- They can be used as a starting point for making your own

Interactive Apps: Demo

- Matlab in OSC OnDemand
- Show plugin: https://github.com/OSC/bc_osc_matlab

Gateway Apps

- AKA Science Gateways
- Passenger web apps in Ruby, Python, or Node.js
- Submits batch jobs using Torque & Moab
- Web interface to
  - Submits batch jobs using Torque & Moab by using a web form
  - Track status
  - Access results
- We provide an “AppKit” that works with Ruby on Rails to speed up Gateway development
  - With knowledge of Rails and our “AppKit”, you can build a basic app in a day

Gateway App Demo

- Interactive Demo of PseudoFUN app

Gateway Apps: Build from scratch

- Starting with an example job template you want to build an app around
- Four steps:
  1. Run a terminal command to scaffold a gateway app using our custom Rails generators, specifying parameters you want users to modify
  2. Replace the default job template with your own job template
  3. Parameterize your job template to use the values submitted by users through the web form
  4. Use our library to specify in the code which batch job(s) to create during a simulation submission

Gateway Apps: Copy and modify

- We will provide example apps that you can copy and modify. This is an alternative to using the generators the App Kit provides.

Open OnDemand Overview

- ✓ About Open OnDemand, OSC OnDemand, and AweSim
- ✓ How OnDemand and OnDemand Apps work
- ✓ App Development
  - System apps
  - Interactive apps
  - Gateway apps
- ✓ Deployment options
- Future support coming. Contact us to get started now.

Deployment options: Possible audiences for your apps

- Create an app and have other members in your group run it
- Create an custom app for a subset of users outside of your group
- Create a generic app for everyone at OSC to use

Deployment options: Three mechanisms

- Run in Developer Sandbox:
  - Share app code through git
  - Other users clone app into their own sandbox and run it
- App Sharing:
  - Deploy production version to your home directory
  - Other users launch the app deployed in your home directory
  - Use file permissions to control who can launch app
- App Publishing:
  - OSC deploys app to web host's local disk
  - App appears in dropdowns with other apps for everybody to launch

Open OnDemand Overview

- ✓About Open OnDemand, OSC OnDemand, and AweSim
- ✓How OnDemand and OnDemand Apps work
- ✓App Development 3 types of apps
  - System apps (i.e. Active Jobs and System Status)
  - Interactive apps (i.e. Jupyter, Matlab, Paraview, Desktop)
  - Gateway apps (i.e. My Jobs, and custom AweSim apps)
- ✓Deployment options
- ✓Future support coming. Contact us to get started now.

Future support coming. Contact us to get started now.

- App Development support
  - Tutorials
  - Example apps
  - Documentation
- Developer mode enabled (or able to be enabled) for every OSC user

- If you want to get started before we roll this out to all OSC users, contact us.

Open Discussion

Thank you! Any questions?

Basil Mohamed Gohar
Web and Interface Applications Manager
Ohio Supercomputer Center
bgohar@osc.edu

Eric Franz
Web and Interface Applications Senior Engineer & Technical Lead
Ohio Supercomputer Center
efranz@osc.edu

https://osc.github.io/Open-OnDemand/ go.osu.edu/ood