Los Alamos NATIONAL LABORATORY

Integrating CONDUIT (Capacity ON Demand User Interaction Toolkit) into Open OnDemand

Christa Collins (HPC-DES)

Mentor: Kevin Pelzel

08/7/2025

LA-UR-25-28217

Overview

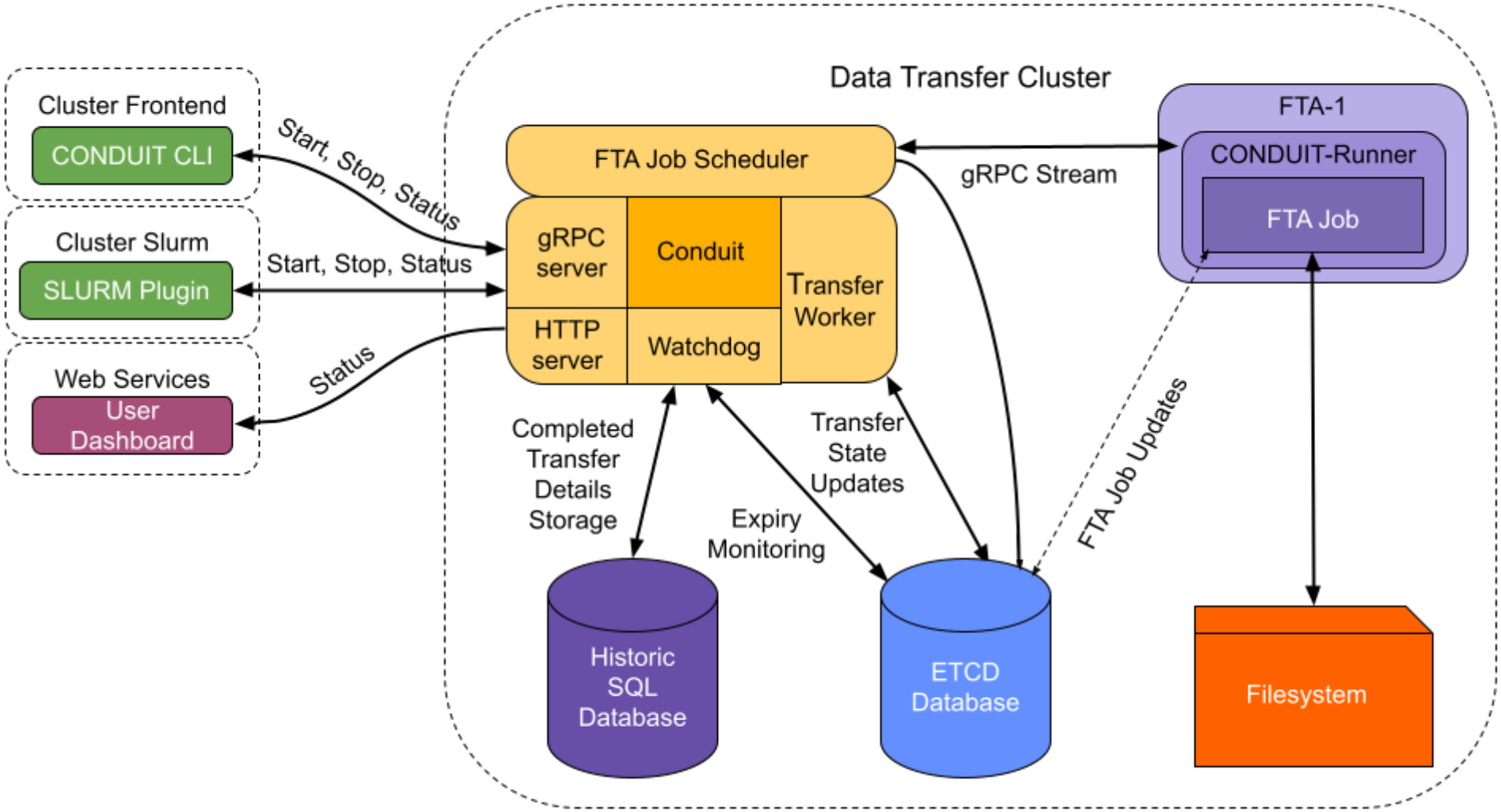
- CONDUIT Description and Overview
- Open OnDemand Overview
- Project Description and Overview
- CONDUIT Dashboard as a Passenger Open OnDemand Application
- Conclusion
- Future Work

CONDUIT Description and Overview

- CONDUIT (Capacity ON Demand User Interaction Toolkit) is a data transfer orchestration system that eases the burden on users when transferring data between LANL's networked storage systems

- By simplifying the file transfer process CONDUIT encourages the use of campaign storage. Allowing for:
  - Automation
  - Increased data integrity
  - Integration with the HPC systems workload manager

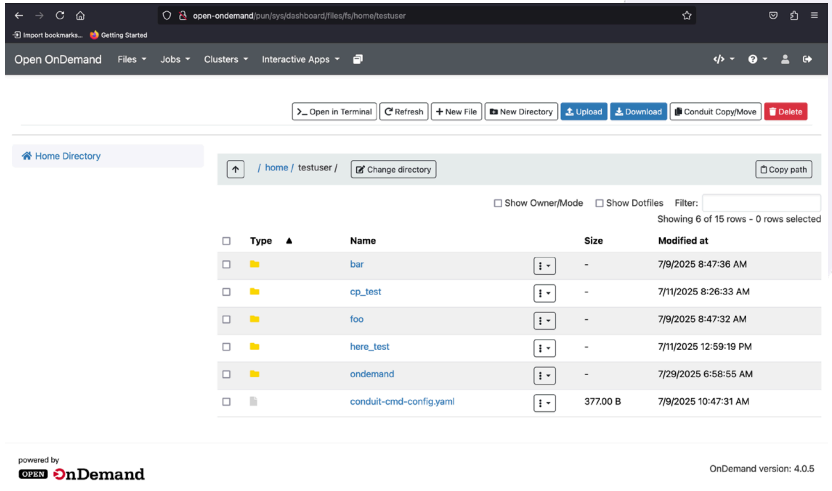
CONDUIT-In Depth



What is Open OnDemand?

- Open OnDemand is an open-source web platform which allows users to access HPC applications and resources from a web browser

- Purpose: Allows users to submit jobs, manage files, and run applications for supercomputers



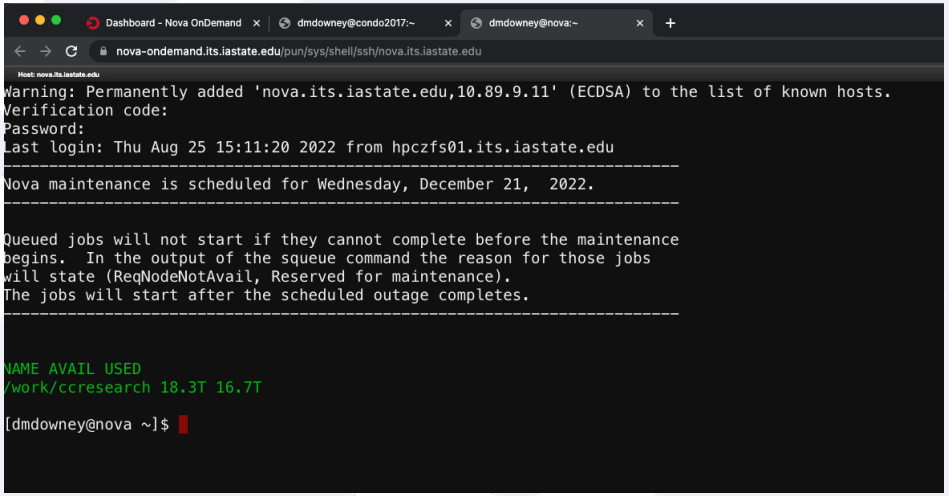
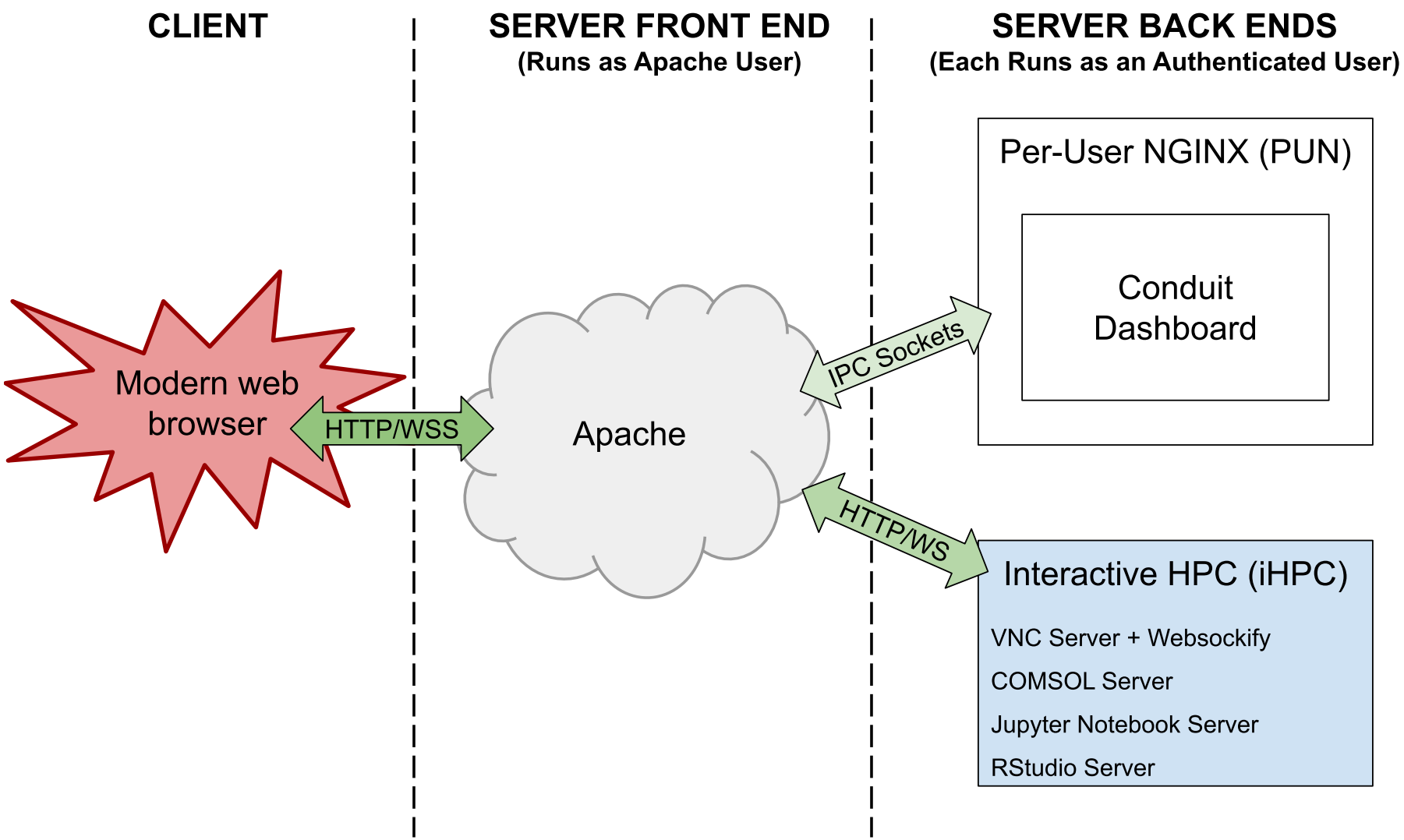


Image credit: https://research.it.iastate.edu/open-ondemand

Open OnDemand Overview



Project Description and Overview

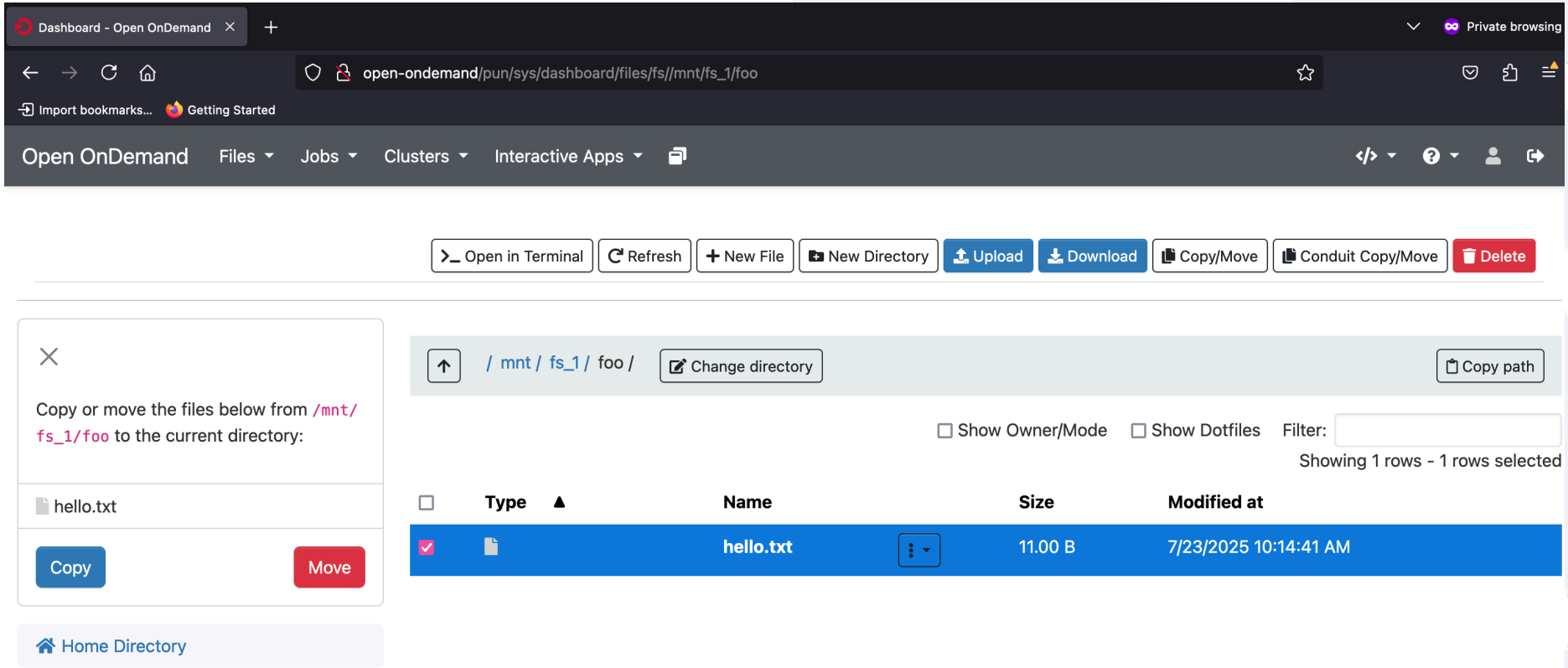
- Integrate CONDUIT into Open OnDemand
- Modify built-in Open OnDemand's File App to run CONDUIT commands
- Utilize OpenID Connect (OIDC) provided by Open OnDemand for user authentication
- Implement CONDUIT dashboard as a Passenger App in Open OnDemand

Running CONDUIT commands in Open OnDemand

Running CONDUIT Commands in Open OnDemand

- Added CONDUIT copy and move commands to the Files app
- Accomplished by modifying ruby code in Open OnDemand's built in files app
  - Modified code runs *conduit cp* as a shell command
  - Then copies or moves files from their source to their destination
- Initial approach of running CONDUIT CLI commands was suboptimal due to the need for a user Kerberos ticket
- Later approaches consist of building and sending http request directly to the CONDUIT server
- Allowing us to take advantage of preexisting OIDC tokens in Open OnDemand to authenticate users with CONDUIT

Running CONDUIT Commands in Open OnDemand



OIDC for User Authentication

What is OIDC?

- OpenID Connect (OIDC) is an identification protocol, built on top of the OAuth 2.0 framework, which allows verified users access to protected endpoints

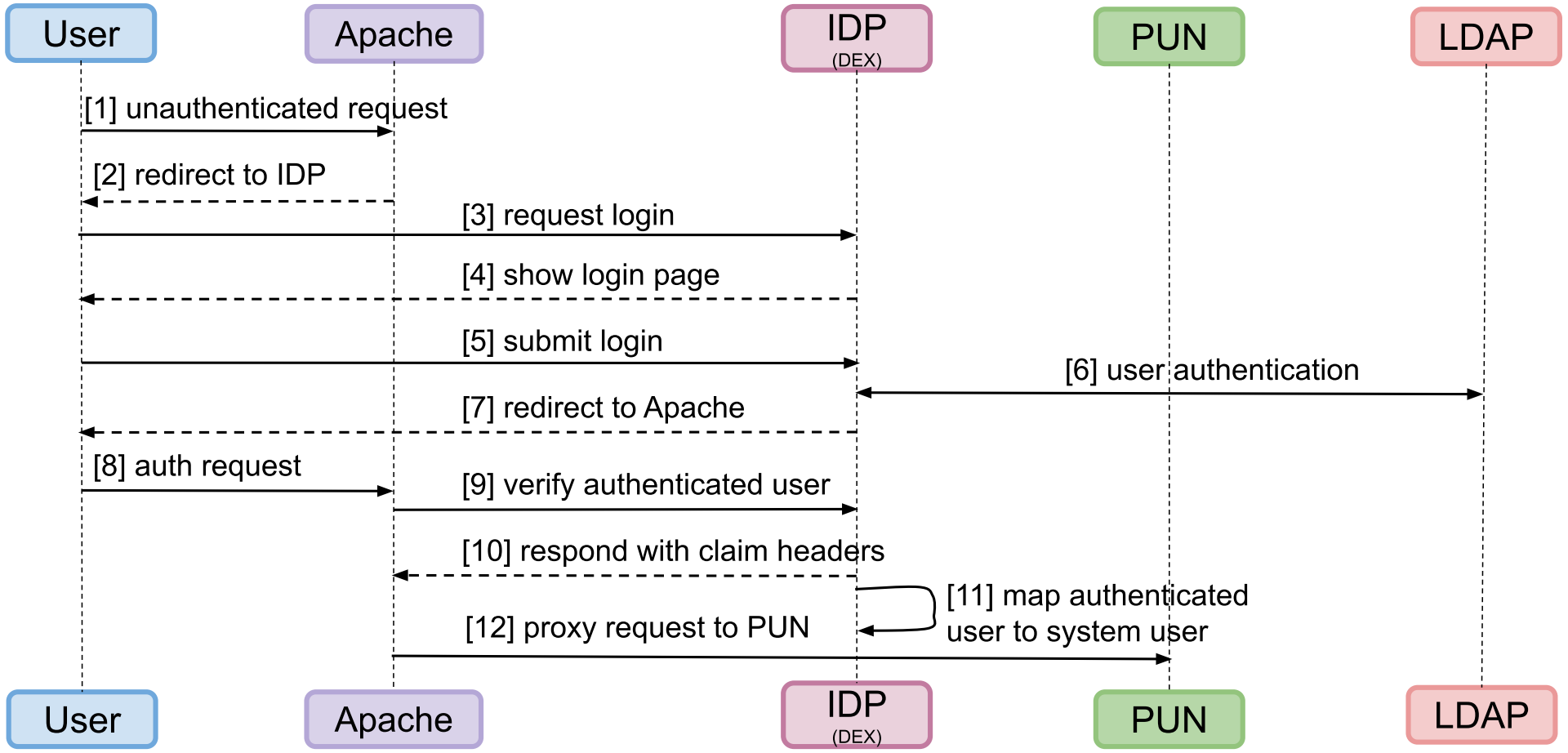
- Allows users to access applications through using Identity Providers (IDP), such as email, to authenticate their identities

- Provides users with one set of credentials to access multiple sites
  - When users sign into a service, they are redirected to their IDP
  - Once users are authenticated, they are redirected back to their application

OIDC Implementation

- Created a testing environment by creating an Open OnDemand instance
  - OIDC Provider: Dex backed by LDAP
  - Dex functioned as our substitute for LANL's current OIDC provider
- Utilized access token provided by the IDP for authentication
- Access tokens in Open OnDemand
  - OIDC tokens are not provided to Open OnDemand apps
  - Open OnDemand exposes access tokens to a pre-hook command during PUN startup
  - Used a pre-hook script to write the user's access token in the user's home directory
  - This access token is later used by conduit apps running in the PUN to authenticate with CONDUIT

OIDC Authentication Flow



Adding CONDUIT as an Open OnDemand Passenger App

Interactive vs Passenger Applications

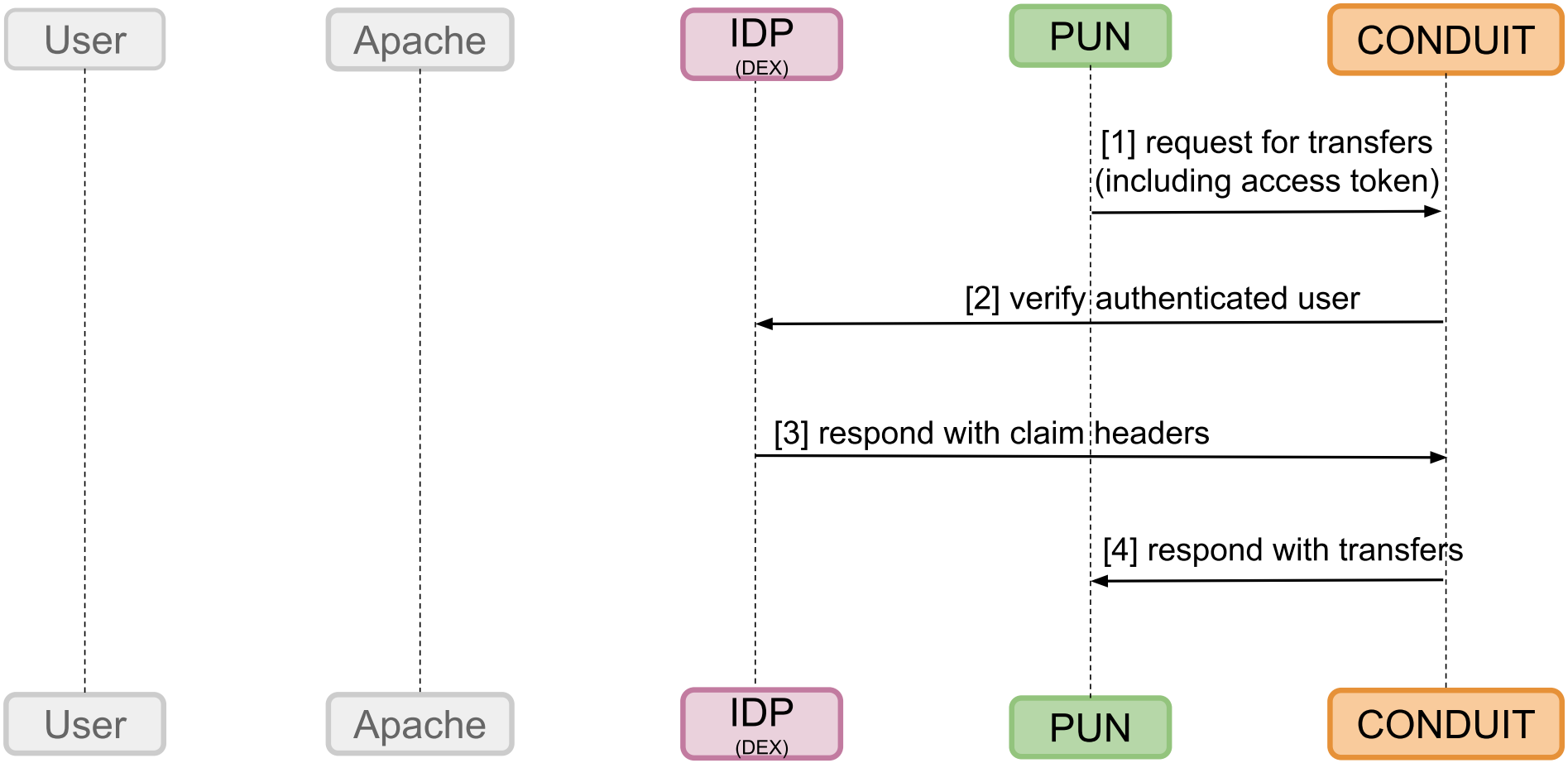
- Interactive Apps
  - Plugins containing configuration files and a job template for running a VNC or Web server application (i.e.Juypter Notebook, MATLAB)
  - Run on a compute node

- Passenger Apps
  - Racked based Ruby, Python, or Node.JS apps
  - Run within the user's NGINX (PUN) instance
  - Core apps in OnDemand such as the Dashboard and Shell app are Passenger apps
  - Developers can create their own passenger apps in Open OnDemand
  - Run on the same node as Open OnDemand

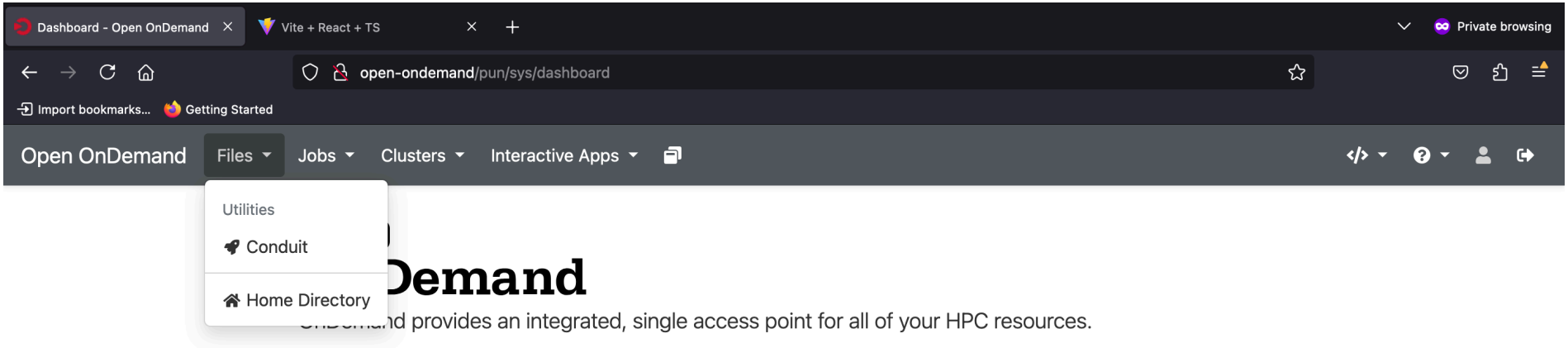
CONDUIT Dashboard Passenger App

- Implemented CONDUIT dashboard as a passenger app
  - Because the CONDUIT dashboard is a Node.js application, it is supported by Open OnDemand
- Previously the dashboard directly sent requests to the HTTP server
- Now we use Express, running within PUN, to proxy the requests from the dashboard to the CONDUIT server
- These requests are authenticated with the user's access token that we placed in their home directory
  - Attach access token to the request header on the client side
  - The CONDUIT server verifies the user's access token with the IDP

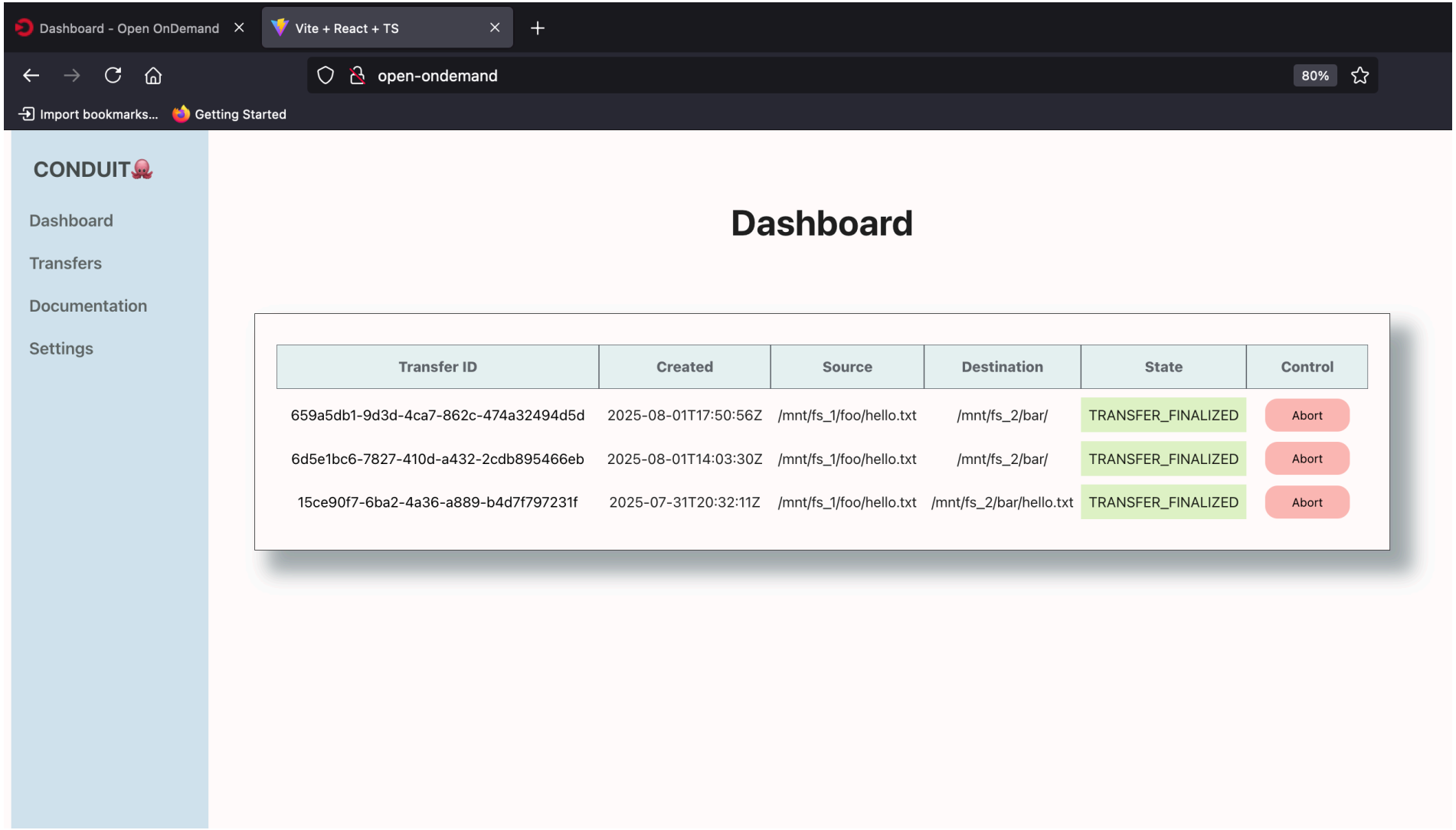
CONDUIT Passenger App Flow



CONDUIT Passenger App



CONDUIT Passenger App



Conclusion

- Provided another way for users to interact with CONDUIT
- Successfully integrated the CONDUIT dashboard into Open OnDemand
- Expanded CONDUIT's authentication functionality through the utilizing OIDC tokens

Future Work

- Continue upgrading the dashboard
- Work on making the files app requests to CONDUIT more robust
- Adding transfer progress functionality to Open OnDemand CONDUIT requests

Questions?

Christa Collins

ccollins@lanl.gov

B.S. Computer Engineering, 2025

Liberty University