

Web Architecture for Artificial Intelligence and Machine-First Uses

Designs to support the age of interoperability

John Consolati (LLNL), T. Mendoza (Matalino Software), C. Cook (LLNL), U. Mir (LLNL)

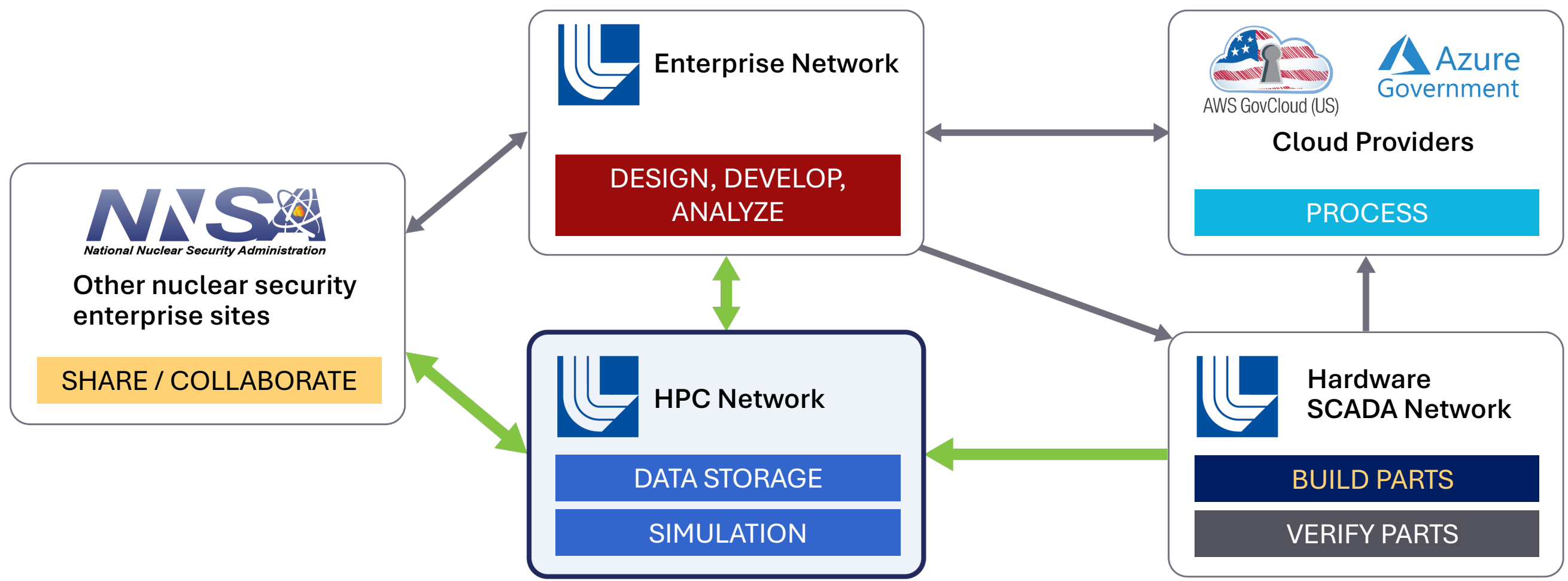
HPC users expect seamless connectivity across enterprise networks and cloud providers. Executing machine-first jobs using LC's HPC as part of composed workflows is critical for timely science. Demand is growing from Nuclear Security Enterprise efforts like Digital Transformation, requiring interoperability between historically isolated networks and sites. Additionally, AI is expected to expand across many services. We are modernizing web architecture to support greater automation and AI capabilities.

This is a 2-year project with 2 FTEs.

Drivers for Change

Both technology and the Nuclear Security Enterprise are changing rapidly and demand a more agile infrastructure:

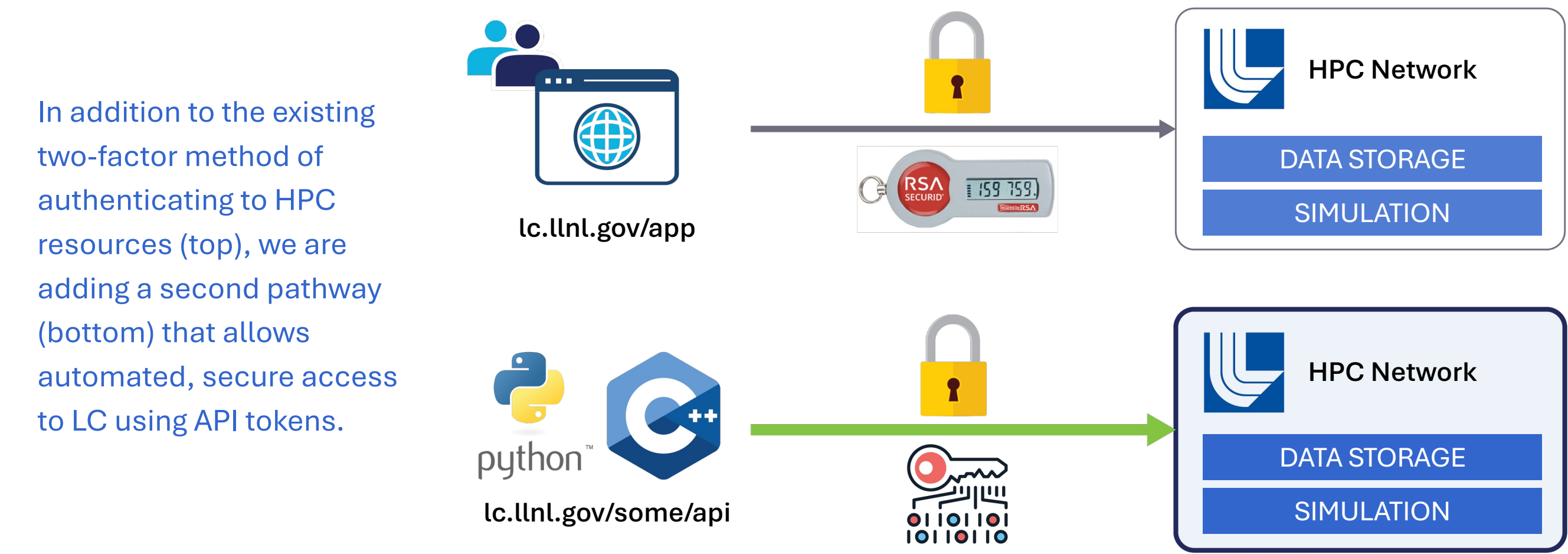
- Software iterates quickly and is deviating from infrequent monolithic deployments
- NNSA's Digital Engineering / Digital Transformation requires that HPC workflows consume and generate data across disparate security zones
- NNSA provides resources (e.g. ESN Hub) that need to integrate with HPC
- Workflows are increasingly complex and thus require automatable patterns



This representative workflow demonstrates the requirement to communicate between different segments of LLNL, commercial cloud providers, and other NNSA sites as part of a comprehensive HPC workflow. Approaches in this poster seek to optimize green arrows.

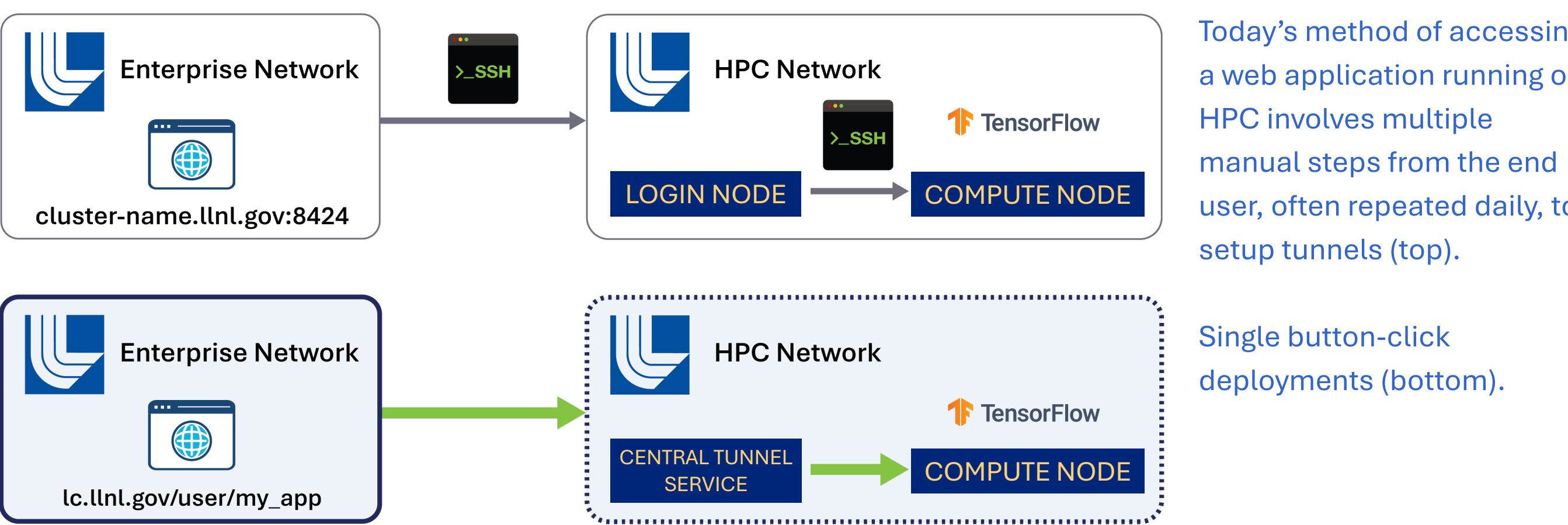
Automated Access to LC Resources

Applications increasingly require services to interoperate between network boundaries over HTTP APIs (e.g. REST, GraphQL, gRPC, webhooks). Some web services natively support API tokens to facilitate secure machine access, but many do not. We are creating a generic provisioning service to enable secure automatable connectivity between the broader LLNL environment and LLNL HPC. This approach brings LLNL closer to parity with the commercial cloud's ease of use while maintaining security.



Easier Web Access to HPC Jobs

Many open-source web tools are designed to be executed in single-user environments like cloud sandboxes or local desktops. As such, they often require complex network tunneling and Linux expertise to run securely on LC's shared HPC systems. This requirement ranges from burdensome to daunting for an end-user. We aim to provide secure, direct web access to user-deployed web tools running on compute clusters to facilitate the user's focus on science instead of DevOps integration.



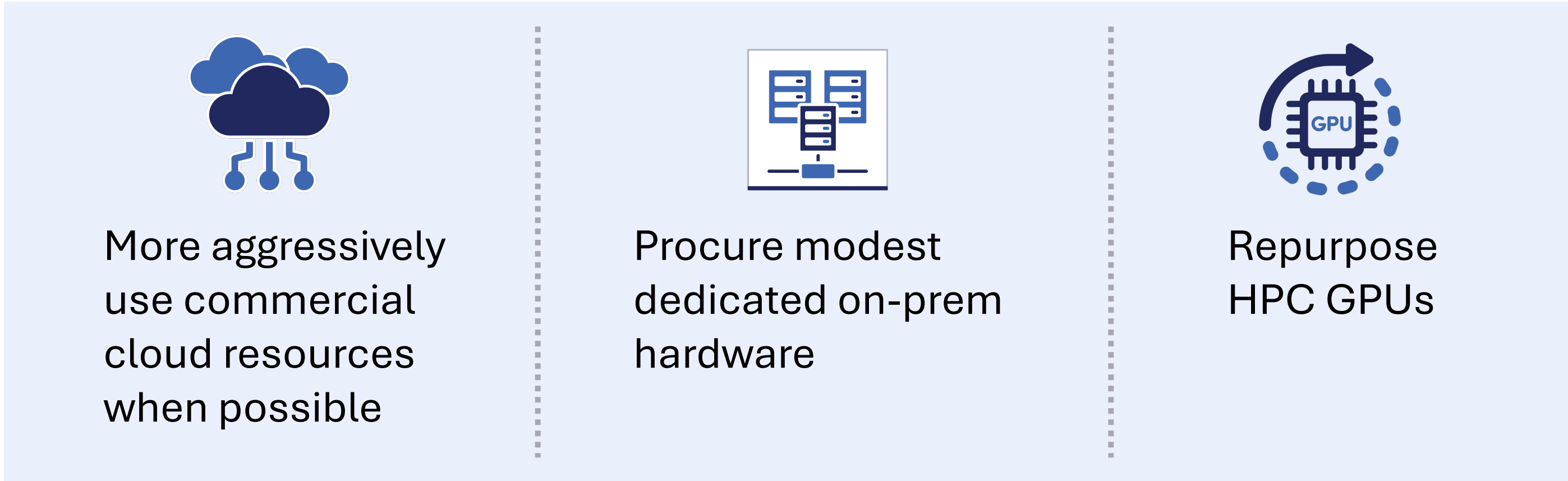
Effects of Integrating AI

LC is positioned extremely well for AI workloads that fit into a batch execution model but must deploy infrastructure (GPUs) to support persistent use cases:

- Web services demand continuous uptime but are not guaranteed to have high load at all times – wastes allocated batch resources if we use HPC GPUs
- AI workloads require expensive hardware – infrastructure budgets are not ready to absorb cost if we procure dedicated GPUs

Cloud pay-for-use pricing models are attractive, but commercial cloud services are not currently viable across all security enclaves.

We are responding to fast-moving needs in three ways:



Conclusions

Web architecture, like other pieces of the HPC center, must change to accommodate workflows that are increasingly decentralized across LLNL and NNSA. Architecture development priorities are shifting toward interoperability between networks and less on satisfying entire workflows within the HPC bubble. AI everywhere requires dedicated hardware and thoughtful investments.

Collaborators

- Los Alamos National Laboratory
- Sandia National Laboratories
- Oak Ridge National Laboratory
- GitLab, Inc.
- LLNL Enterprise Information Technology
- Matalino Software

Next Steps

- Interoperability between enterprise and HPC authentication systems
- Dedicated AI inference hardware for web, move pilot capabilities to prod
- Support dynamic authentication for user web apps running on HPC and LC persistent cloud

Enabling automatable and seamless interaction with HPC inputs and outputs