Using Tapis: APIs for Portable, Reproducible High Performance Computing in the Cloud

Julia Looney and Richard Cardone
Cloud and Interactive Computing
Texas Advanced Computing Center
University of Texas, Austin

January 8, 2020
HPC, HTC, Visualization, Large scale data storage, Cloud computing, Experimental architectures
Challenges of Computational Research

Instruments and Sensors
- Sequencers
- Telescopes
- Shake Tables
- Wind Tunnels
- Lasers

Data Processing
- Quality Assurance
- ETL scripts
- Machine Learning

Model Simulation
- OpenMP
- MPI
- CUDA

Sharing & Publishing
- Permissions
- History
- Provenance
- Reproducibility
The Tapis Project

NSF funded, Web-based Computing Framework To Support Research

- Generally - A framework to support computational research, in any domain, that enables investigators to leverage computing resources across various institutions.

- More Technically- A set of hosted, multi-datacenter REST APIs, together with languages SDKs and CLI for securely managing data and executing code on HPC, HTC and cloud systems.

[NFS awards 1931439 and 1931575]


Why Use Tapis?

Tapis Gives You The Ability to...

- **Track your analysis provenance** - Tapis records your input and output data along with application used and settings - so you know what you have done every time.

- **Reproduce your analysis** - Tapis records all your inputs/outputs/parameters etc. so you can re-run an analysis.

- **Share your data, workflows/applications, computational resources** with collaborators or your lab - Tapis enables sharing with access controls for all your data/resources/applications within Tapis.

Without having to install or support a complicated technology stack
### Who Is Using Tapis?

<table>
<thead>
<tr>
<th>Science Gateways</th>
<th>Labs/Projects</th>
</tr>
</thead>
<tbody>
<tr>
<td>CyVerse</td>
<td>Planet Texas 2050</td>
</tr>
<tr>
<td>DesignSafe</td>
<td>Hawaii Data Science Institute</td>
</tr>
<tr>
<td>VDJServer</td>
<td>iReceptor+</td>
</tr>
<tr>
<td>SD2E</td>
<td>C-MAIKI</td>
</tr>
<tr>
<td>3DEM</td>
<td>ECCO</td>
</tr>
<tr>
<td>iMicrobe</td>
<td>GenApp</td>
</tr>
<tr>
<td><code>Ike Wai</code></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Institutions</th>
<th>Additional collaborations starting soon...</th>
</tr>
</thead>
<tbody>
<tr>
<td>TACC</td>
<td></td>
</tr>
<tr>
<td>CDC</td>
<td></td>
</tr>
<tr>
<td>UH</td>
<td></td>
</tr>
<tr>
<td>Compute Canada</td>
<td></td>
</tr>
</tbody>
</table>
Science Gateways
DesignSafe is the web-based research platform of the NHERI Network that provides the computational tools needed to manage, analyze, and understand critical data for natural hazards research.

- What is the NHERI Network?
- What is the role of the Network Coordination Office?
- How can I start using DesignSafe?
DesignSafe - Authentication

DesignSafe Authentication leverages Tapis OAuth

TACC LDAP integration easily configured.
DesignSafe Research Workbench
DesignSafe - Data Depot

DesignSafe Data Depot leverages:
* /systems
* /files
* /meta

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Last modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>jstubbs</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>jstubbs</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>__output</td>
<td>--</td>
<td>5/5/17 7:07 PM</td>
</tr>
<tr>
<td>agavepy token checks.ipynb</td>
<td>192.0 kB</td>
<td>6/20/17 4:00 PM</td>
</tr>
<tr>
<td>archive</td>
<td>--</td>
<td>8/18/16 7:29 AM</td>
</tr>
<tr>
<td>clawpack test.ipynb</td>
<td>3.0 kB</td>
<td>5/23/17 3:48 PM</td>
</tr>
<tr>
<td>data.txt</td>
<td>68.0 bytes</td>
<td>4/11/17 12:17 PM</td>
</tr>
<tr>
<td>ffmpeg_test.ipynb</td>
<td>873.0 bytes</td>
<td>3/17/17 9:50 AM</td>
</tr>
</tbody>
</table>
DesignSafe - Workspace

DesignSafe Workspace leverages:
* /systems
* /apps
* /jobs
* /meta
Launch OpenSees
Primary Capabilities - current production version (previously Agave):

- Systems, Files, Apps, Jobs - Resources to manage/share data, run jobs
- Metadata - Associate JSON documents with Tapis resources
- Abaco Functions - Provide actor-based function execution
- Authentication Module - OAuth2/OIDC, tenant-specific user authentication
Tapis: V3 Overview

Primary Capabilities - Future Version:

- Systems, Files, Apps, Jobs - Resources to manage/share data, run jobs*
- Metadata - Associate JSON documents with Tapis resources*
- Abaco Functions - Provide actor-based function execution
- Authentication Module - OAuth2/OIDC, tenant-specific user authentication*
- Containerized Apps - First class support for containers
- Streams - Store and process sensor data in real-time
- Security Kernel - Decentralized secrets store and authorization subsystem

* New in v3
Tapis V3 Goals

1. Use containers to maximize job portability and reproducibility
   a. All applications are delivered as images and run in containers
   b. Portability: Match system capabilities with application runtime requirements
   c. Reproducibility: Include many runtime dependencies in container image

2. Support sensor input and streaming data
   a. Real-time, event driven workflows
   b. CHORDS time-series data facility
   c. Integrate with actor and job computational engines

3. Extend multiple data center support
   a. One primary and zero or more associate sites run Tapis components
   b. Decentralize authz/secrets across data centers by allowing multiple Security Kernels
   c. Isolate authentication, authorization and secrets administration by tenant
Architectural Overview
Multi-Data Center Challenges

- **Security**
  - Establishing trust between geographically dispersed services
  - Allowing local administration of secrets and permissions

- **Configuration**
  - Determining what components will run at each site
  - Routing requests and work to proper service instance

- **Management**
  - Multiple orchestrators (e.g., separate Kubernetes at each site)
  - Coordinating updates and maintenance
  - Starting and stopping services
  - Monitoring, logging, auditing, debugging
Decentralized Portable Security

Security Kernel

Vault Cluster w/Consul Backend (HA)

Security Library

REST

DB

Apache Shiro Authorization

Permission Checker

Cache

Java
Security Kernel Challenges

- **Fast**, fine-grained authorization checking
  - Extend Apache Shiro permission model
  - Postgres backend
  - Server caching (client caching if needed)

- Comprehensive secrets management
  - Hashicorp Vault backend
  - All non-user secrets used by system kept in Vault
    - SSH keys, DB credentials, passwords, etc.
  - Automate key rotation
  - Simplify key distribution using VaultCA

- Portable deployment
  - Each tenant can have its own Security Kernel instance
    - Ease of deployment is paramount
  - Manage directly by site administrators
  - Secrets never leave site
Smart Scheduling Challenges

- Support for Docker and Singularity container images and runtimes
- Systems advertise their hardware and software capabilities
- Workloads can be scheduled to run on qualifying systems
- Scheduling can take resource availability into account to minimize time-to-solution
Container Challenges

- Achieve greater reproducibility of results
  - Encapsulate most application dependencies in container images
- Enable greater application portability
  - Enable an application to run on different execution systems
  - Dynamically match system capabilities and application requirements
  - Develop algorithms that dynamically minimize time-to-solution
- Efficient image distribution on many-node clusters
Functions-as-a-Service with Abaco

1. Users define computational primitives called Actors with a Docker image.

2. Abaco assigns each actor a unique URL over which it can receive messages.

3. Users send the actor a message by making an HTTP POST request to the URL.

4. Abaco launches a container from the associated image, injecting the message into the container.
Real-time Complex Workflows

- Events can trigger workflow processing in one data center
- Results from initial steps informs subsequent steps in the workflow
- Workflow processing can leverage resources in other datacenters
- Workflows can mix realtime, batch, cloud, HTC and HPC resources
Streams API

- Integrating with CHORDS (NSF Earth Cube)
- Stream data from geo-distributed sensors
- Search/slice data using geospatial and temporal indexes
- Trigger data processing or workflows based on user defined conditions
- Process data streams with
  - Batch jobs via Tapis apps
  - Relay streams to 3rd party engines
- Real time data processing
- Invoke notifications, jobs on thresholds
- Uses an influxdb, mongodb backend with a REST Interface
Tapis Roadmap

“V2” - Current Production version.
● In use by 15 independently funded projects.
● Much more informal usage: Approximately 20k OAuth clients.

“V3” - In development;
● 5 year NSF funded project, Sept 2019-Aug 2024
● Early Adopters Program - Jan 1, 2020
● Beta Release Early Summer 2020
● Early Adopters Workshop July/August 2020.
Tutorials and Training

We will be providing training and tutorials throughout the year. Check the following website for updates:

https://tapis-project.org/training/
Join Us!

https://tapis-project.org

More information:

- Developer Docs: https://tacc-cloud.readthedocs.io
- OpenAPI v3 Live docs: https://tapis-project.github.io/live-docs/
Thanks!

https://tapis-project.org