

An Overview Of the Tapis API Platform

Xpert Network Webinar, 8/19/2021

Joe Stubbs, PhD

Lead, Cloud and Interactive Computing

Texas Advanced Computing Center

University of Texas, Austin



Introduction

Joe Stubbs, PhD - *Research Associate, UT Austin.*
Lead, Cloud and Interactive Computing (CIC) at TACC.

- Joined TACC in 2013.
- Formed CIC in March, 2017, 4 full time staff.
- PI of Tapis NSF CSSI award

CIC Today

- 12 full-time staff (6 PhDs), plus students.
- Primarily funded by NSF, but also DARPA, NASA, and NIH.
- A mix of our own projects and collaborations with other groups.

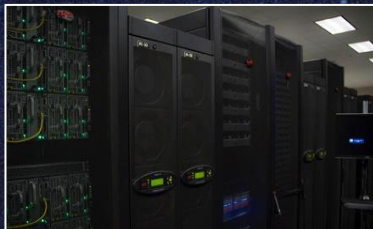
Ecosystem of Powerful Tools For Computational Research



HPC, HTC, Visualization, Large scale data storage, Cloud computing, Experimental architectures

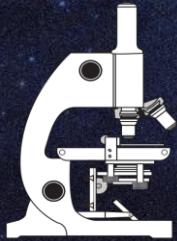


TACC Systems



Computing System	Specialization
<i>Frontera</i>	23.5 PF Intel Cascade Lake + NVIDIA V100 GPU
<i>Stampede 2</i>	18PF Intel Skylake and KNL – Capability HPC
<i>Lonestar 5</i>	2PF Intel Haswell – HPC & HTC
<i>Wrangler & Maverick</i>	600 TB Flash + 10 PB HDD – Data Intensive High-end NVIDIA GPU – ML, Analytics, Visualization
<i>Jetstream & Chameleon</i>	OpenStack Cloud – Usable, programmable infrastructure
<i>Rodeo</i>	VMware/OpenStack – Production hosting
<i>Stockyard</i>	20PB Lustre – Global, compute-optimized filesystem
<i>Corral</i>	15PB GPFS – High-integrity, performant HDD + Data services
<i>Ranch</i>	160PB SAMFS – Long-term archival storage

Challenges of Distributed Computational Research



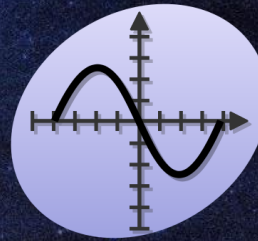
Instruments and Sensors

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, REST APIs, together with languages SDKs and CLI for securely managing data and executing code on HPC, HTC and cloud systems.



Why Use Tapis?

Three higher-level objectives

- **Programmable access to advanced resources** - Conduct analyses on cloud/high-throughput and HPC resources using a common API.
- **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 stack of technology



Tapis: Overview

Primary Capabilities - *version v2 (in Production)*

- Systems, Files, Apps, Jobs - *Interact with data and execute code on HPC and HTC systems.*
- Metadata - *Manage large collections of database/document objects.*
- Abaco Functions - *Execute containerized functions in response to messages.*
- Authentication Module - *OAuth2/OIDC compliant front-end.*



Tapis: Overview

Primary Capabilities - *Version V3 (in Production 9/1/2021)*

- Systems, Files, Apps, Jobs - *Interact with data and execute code on HPC and HTC systems.*
- Metadata - *Manage large collections of database/document objects.*
- Abaco Functions - *Execute containerized functions in response to messages.*
- Authentication Module - *OAuth2/OIDC compliant front-end.*
- Containerized Apps - *First class support for containers.*
- Streams - *Store and process sensor data in real-time.*
- Security Kernel - *Decentralized secrets store and authorization subsystem.*
- PgREST - *Managed, scalable, Postgres-as-a-service*

**New in v3*



Tapis API Catalog

Tenancy, Authentication and Security

- Tenants
- Sites
- Tokens
- Authenticator
- Security Kernel
- Postits

MetaData Management

- Meta
- PgREST

<https://tapis-project.github.io/live-docs>

Data Management and Code Executions

- Systems
- Files
- Apps
- Jobs

Streaming Data, Events and Functions

- Functions (Actors)
- Notifications
- Streams



How Do I Use Tapis?

- CURL can call the HTTP APIs

```
$ curl -H X-Tapis-Token $token https://api.tapis.io/v3/apps
```

- Command Line Interface (CLI)

```
$ files-list -s cloud.corral data/
```

- Official SDKs for Python (tapipy) and Java

```
>>> tp.jobs.submit(app_id='sail-fish.1.10', input_dir='data/raw/rnaseq')
```

- Web applications - such as Science Gateways



Tapis UI

Official Tapis project being developed by multiple individuals

- TypeScript library, tapis-typescript: <https://github.com/tapis-project/tapis-typescript>
- Full frontend application, tapis-ui: <https://github.com/tapis-project/tapis-ui>
- Pure static application, served out of github pages.

Use it in your own gateway project

- Fork the repo, configure tenancy, re-mix the components into your own UI
- Serverless application, no backend to maintain

Will demo Tapis UI later in the meeting

Who Is Using Tapis?

Science Gateways

- CyVerse
- DesignSafe
- VDJServer
- SD2E
- 3DEM
- iMicrobe
- `Ike Wai

Institutions

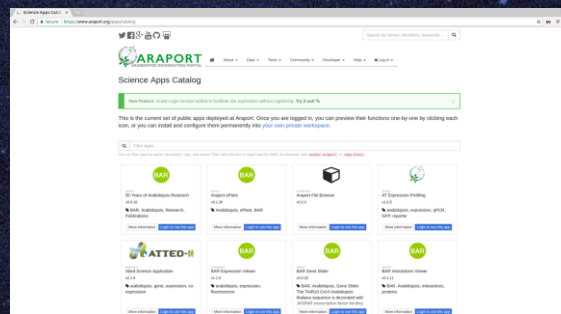
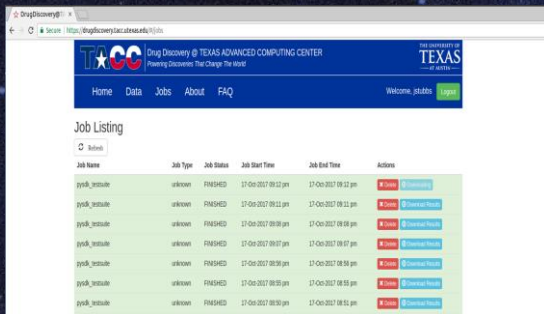
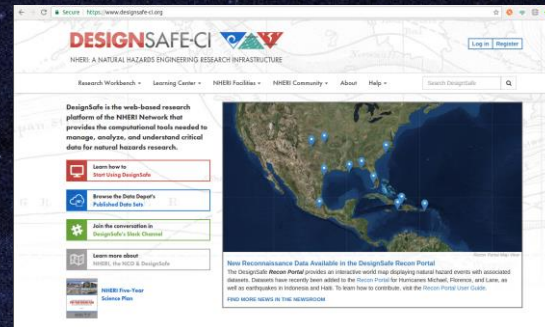
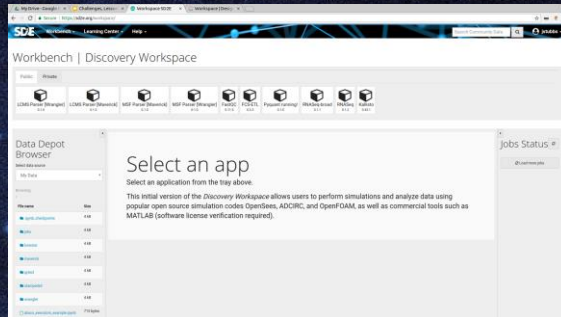
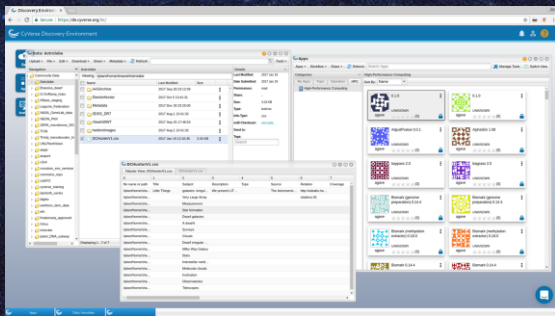
- TACC
- CDC
- UH
- NIH
- Compute Canada

Labs/Projects

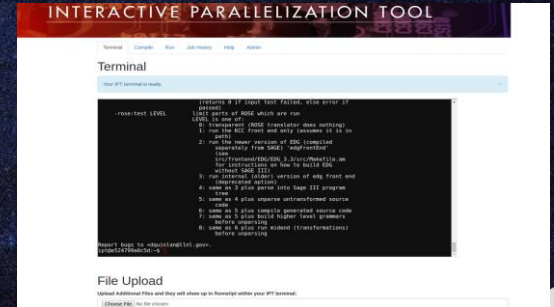
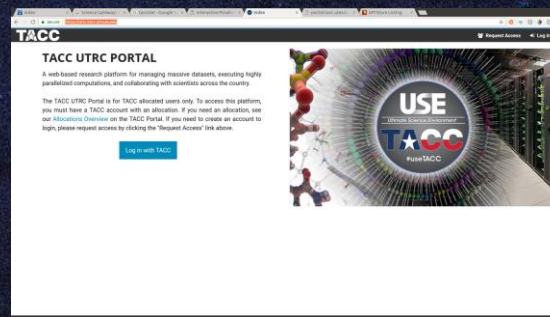
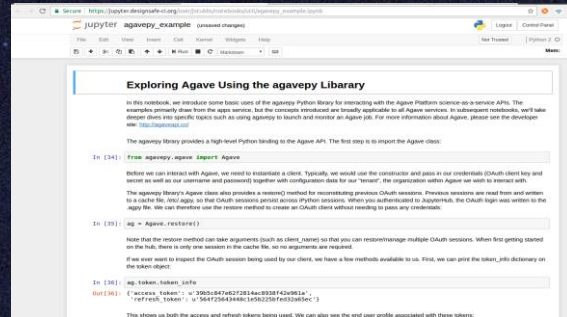
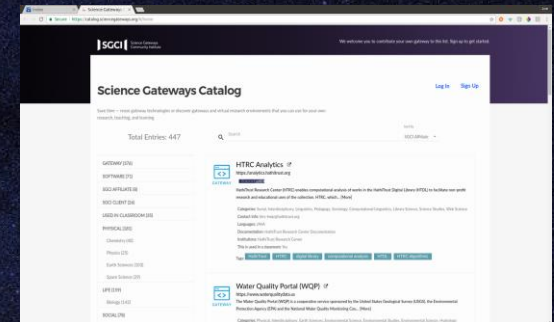
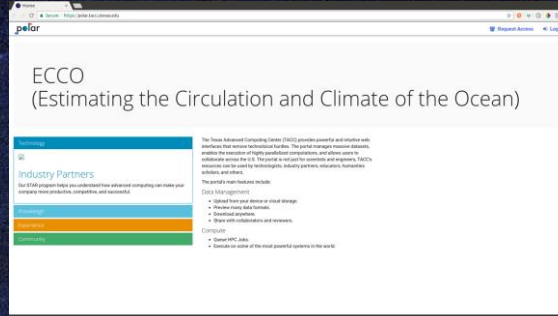
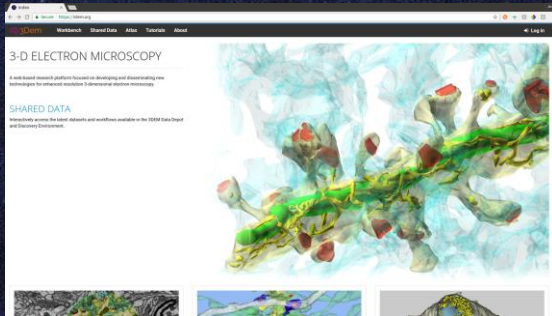
- Planet Texas 2050
- Hawaii Data Science Institute
- iReceptor+
- C-MAIKI
- ECCO
- GenApp
- Acute to Chronic Pain Signatures (A2CPS)

Additional collaborations starting soon...

Science Gateways



Across Various Domains



Types of Applications

Batch/HPC and Cloud/High-Throughput

- Astro/Physics - astropy data analysis pipelines
- Chem/Molecular Dynamics - Gromacs, LAMMPS, Quantum Espresso
- Engineering - ADCIRC, ANSYS, Clawpac, LS-DYNA, OpenFoam, Opensees,
- NGS - Blast, BWA, FastQC, TR Finder,
- Synthetic Design - Protein Design, XPlan,

Interactive/Data Analysis

- Jupyter notebook
- RStudio
- Custom apps - Shiny, Dash,



Tapis Roadmap

“V2” - Current Production version.

- In use by 15 independently funded projects.
- Much more informal usage: Approximately 50k OAuth clients.

“V3” - Initial development and Production Release

- 5 year NSF funded project, Sept 2019-Aug 2024
- Early Adopters Program - Jan 1, 2020 through Fall 2021
- Production Release Candidate -- available today
- Production 1.0.0 Release -- 9/1/2021



Join Us!

<https://tapis-project.org>

More information:

- PEARC 21 Tapis User Meeting: <https://tapis-project.github.io/pearc21-tapis-user-meeting-bof/>
- Developer Docs: <https://tacc-cloud.readthedocs.io>
- OpenAPI v3 Live docs: <https://tapis-project.github.io/live-docs/>
- Join our Slack team: <http://bit.ly/join-tapis>

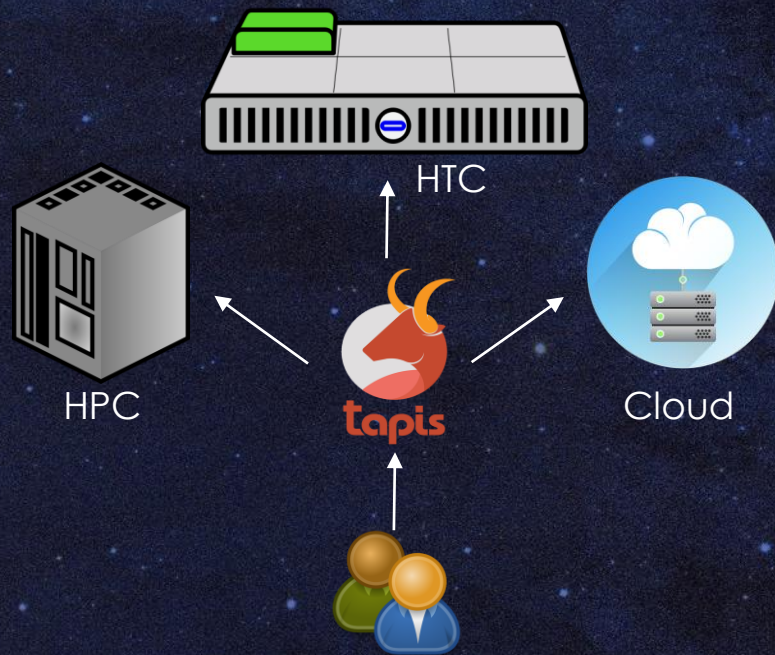


Extra Slides



Data Management and Code Execution APIs

/systems /files /apps /jobs



- Register storage and compute systems
- Ingest, move and transform data files and folders
- Register application containers on large systems
- Launch jobs to invoke applications
- Capture metadata about the workflow



Tapis Systems

- ▶ A **Tapis System** is an abstraction of a host identified by name or IP address.

- ▶ Uses of Tapis systems:
 - Storing and retrieving files and data.
 - Running a job, including:
 - Staging files to a system in preparation for running a job.
 - Executing a job on a system.
 - Archiving files and data on a remote storage system after job execution.

Registering a Tapis v3 System

At a high level a system represents the following information:

- *System ID*
- *Where the system is hosted*
- *Linux or S3 system*
- *Credentials to login to the system*
- *Owner, Effective User ID of the system, etc.*

A user can register Tapis v3 system either with Tapis UI, curl, Python/Java SDK using json



system.json

```
{  "id": "tapisv3-storage",
  "authnCredencial": {
    "privateKey": "",
    "publicKey": ""},
  "canExec": false,
  "defaultAuthnMethod": "PKI_KEYS",
  "description": "Systems for testing
large files transfers",
  "effectiveUserId": "${apiUserId}",
  "enabled": true,
  "host": "<\"host-ip>",
  "owner": "testuser2",
  "port": 22,
  "rootDir": "/home/testuser2",
  "systemType": "LINUX",
  "useProxy": false }
```


Tapis Apps

- ▶ A **Tapis App** is a versioned, containerized executable that runs on a specific execution system through Tapis Jobs service.
- ▶ Tapis apps are expected to evolve over time. Hence, app Id + version must be unique within a tenant.
- ▶ A Tapis app represents all the information required to run a Tapis job on a Tapis system and produce useful results.

Registering a Tapis v3 Application

At a high level a app represents the following information:

- App id
- Version
- App type: Batch or FORK
- App owner
- Runtime : Docker/Singularity
- Job Attributes

A user can register Tapis v3 app either with Tapis UI, curl, Python/Java SDK using json



app.json

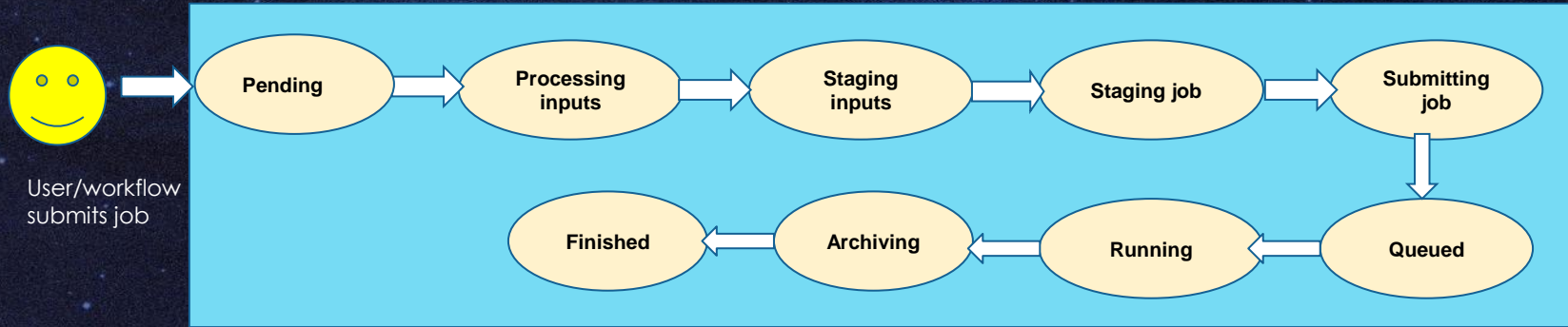
```
{
  "id": "demo.app",
  "version": "0.1",
  "appType": "FORK",
  "description": "My sample application",
  "runtime": "DOCKER",
  "containerImage": "docker.io/hello-
world:latest",
  "jobAttributes": {
    "description": "default job
description",
    "execSystemId": "execsystem1"
  }
}
```


Tapis Jobs

- ▶ Tapis Job service aims at launching applications directly on hosts or as job submitted to schedulers (currently only Slurm).
- ▶ The **Tapis v3 Jobs** service is specialized to run containerized applications on any host that supports container runtimes.
- ▶ Currently, Docker and Singularity containers are supported.
- ▶ The Jobs service uses the Systems, Apps, Files and Security Kernel services to process jobs.

A day in the life of Tapis v3 Job

Lifecycle of a successful job



Other jobs states:

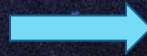
- Blocked
- Cancelled
- Failed
- Paused

Running Tapis v3 Job

At a high level a job represents the following information:

- *Job name*
- *App id*
- *AppVersion*
- *Parameter Set*
- *Input files*
- *Archive System*

A user can run Tapis v3 job either with Tapis UI, curl, Python/Java SDK using json



job.json

```
{
  "name": "demo.app",
  "appId": "demo.app",
  "appVersion": "0.0.1",
  "parameterSet": {
    "envVariables": [
      { "key": "JOBS_PARMS", "value": "15" }
    ],
    "archiveFilter": {
      "includes": ["Sleep*"],
      "includeLaunchFiles": true
    }
  },
  "fileInputs": [
    { "sourceUrl": "tapis://tapisv3-exec/sample1.txt",
      "targetPath": "sample1.txt" }
  ],
  "archiveSystemId": "demo.tapisv3..system"
}
```