Scientific Computing Without the Command Line: Enabling Any HPC Code to Run Anywhere through a Web Interface with the Agave API

April 1, 2020

Kathy Traxler, Steven R. Brandt
Department of Computer Science, Center for Computation and Technology Louisiana State University

https://togo.agaveplatform.org
- Kathy Traxler: ktraxler@lsu.edu
- Cell: 225-454-7409
- If you want a tutorial for your group, etc. just call and we’ll set it up
What is a Science Gateway?

Normally, it means a web interface used to run a complex scientific application on a high performance computer.

Designing a GUI or web interface to resources and making it robust is a long complex process that can take many hours.

Today we are going to learn how to make a science gateway from your application in an hour!

More formally:

https://www.xsede.org/gateways-overview
What is Agave?

- The Agave Platform (LINK: http://agaveplatform.org) is an open source, science-as-a-service API platform for powering your digital lab.

- Agave allows you to bring together your public, private and shared high performance computing (HPC), high throughput computing (HTC), cloud and Big Data resources under a single friendly REST API.
How does Agave work

Under the hood, Agave works by using a REST api and JSON to format data.

What do the words JSON and REST mean?

1. What is JSON? JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate.

2. What is REST? REST stands for REpresentational State Transfer. REST is a stateless, cacheable, uniform way of using URL's to communicate between a client and server.
What Capabilities does Agave provide?

- Run code
- Manage data
- Collaborate meaningfully
- Integrate anywhere
- Provenance chain
https://togo.agaveplatform.org

- This is the URI you will use after the tutorial is over
- To use this you must have an Agave account and an HPC account
Everything on this screen if you were to create it from scratch would be a HUGE job requiring hours of coding. With ToGo your Gateway’s users can login securely and get to work.

This is the Create account and login page.
Finish Agave Account

- Finish creating your account by filling in required information and submitting
- Go to the email app you use for the email address you gave Agave and click on the link to verify and activate your account.
Setup HPC Account

- Already have an XSEDE or other supercomputer center (HPC) account? You are set.

- No HPC account? You must get one before you can proceed. Ask your group leader, major professor or other researchers around you where to get your account. HPC account: https://accounts.hpc.lsu.edu/login_request.php

- After acquiring your account and password you are ready to go.
Get the code and wrapper files

- To follow along with this tutorial you need several files. These files will be downloaded from my web page in the following slides.
  
  - `input.txt` (input parameter file)
  - `drawgau.cpp` (source code)
  - `drawgau-wrapper` (batch file script to run the job)
#!/bin/bash

echo 'Executing Draw Gau Code'

# Setting the x flag will echo every command onto stderr. This is
# for debugging, so we can see what's going on.
set -x

echo ==ENV==============

# The env command prints out the entire execution environment. This
# is also present for debugging purposes.
env > env.txt

echo ==PWD==============

# We also print out the execution directory. Again, for debugging purposes.
pwd > pwd.txt

echo ==JOB===============

EXE_DIR=/home/$USER/agave-deployment

if [ "${PBS_NODEFILE}" = "" ]
then
    # When running on a system managed by Torque
    # this variable should be set. If it's not,
    # that's a problem.
    echo "The PBS_NODEFILE was not set"
    exit 2
fi

# Strip off the .par from the end of the parfile name.
# Create a directory with the name of the parfile.
# More debugging.
echo PARFILE=$parfile
echo NODEFILE=$PBS_NODEFILE

# By default, the PBS_NODEFILE lists nodes multiple times, once for each MPI process that should run there. We only want one MPI process per node, so we create a new file with "sort -u".
LOCAL_NODEFILE=nodefile.txt
sort -u < $PBS_NODEFILE > $LOCAL_NODEFILE
PROCS=$(wc -l <$LOCAL_NODEFILE>)

echo "${xmin} ${xmax} ${deltax}" > input.txt

# Execute our MPI command.
mpirun -np $PROCS -machinefile $LOCAL_NODEFILE $EXE_DIR/drawgau input.txt

# Create a plot
gnuplot drawgau.py drawgau[0-9]*.txt
[ktraxler@shelob ]$
Get Files

• You need the files and the best way to get them is:

  • log into your personal hpc account and type the following statements:

    • curl -LO https://cct.lsu.edu/~sbrandt/agave-deployment.tgz
    • tar xzvf agave-deployment.tgz
    • cd agave-deployment
    • make
Setup Agave Storage and Execution Systems
Introducing the new Agave ToGo!

Agave ToGo v2 is a full-featured web application designed to show off core functionality you are familiar with in the Agave Platform as well as demonstrate some of the advanced use cases which are possible leveraging the core Agave Core Science APIs.

This application is meant to serve as a reference from which you can build your own application. Feel free to fork this repository and edit as needed. To contribute back enhancement and bug fixes, please fork the repository and submit a pull request.

26 Jobs

12,5TB Data Moved

6 Apps

+25% Activity Increase

The right side of the dashboard.
The left side of the dashboard

This is where we will do our work!
Create the Systems Needed

- We will first create the Storage System description
- Then create the Execution System description
- Use the Dashboard and choose the Systems item
The left side of the dashboard
Click on the “Systems” link
This will take you to the page where you will create your execution and storage systems
Choose storage from the window above.

Click on menu and choose storage for your system.
SYSTEM EDITOR WIZARD - STEP 2 OF 2

1. Type
2. Details
3. Connectivity

ID
shelb-storage-hpctrn20

Name
Shelob at LSU

Status
UP

Description
The Shelob supercomputer at LSU

Site
hpc.lsu.edu

Code:
```
1
"owner": "ktraxler",
2
"available": true,
3
"description": "The Shelob supercomputer at LSU",
4
"storage": {
5
  "proxy": null,
6
  "protocol": "SFTP",
7
  "mirror": false,
8
  "port": 22,
9
  "auth": {
10
    "type": "PASSWORD"
11
  },
12
  "publicAppsDir": null,
13
  "host": "shelob.hpc.lsu.edu",
14
  "rootDir": "/",
15
  "homeDir": "/home/hpctrn20",
16
  "proxyTunnel": "NO"
17
}```
```json
{
  "owner": "ktraxler",
  "available": true,
  "description": "The Shelob supercomputer at LSU",
  "storage": {
    "proxy": null,
    "protocol": "SFTP",
    "mirror": false,
    "port": 22,
    "auth": {
      "type": "PASSWORD",
      "username": "ktraxler",
      "password": "#SSdalejr#88amp2017"
    },
    "publicAppsDir": null,
    "host": "shelob.hpc.lsu.edu",
    "rootDir": "/",
    "homeDir": "/home/hpctrn20",
    "proxyTunnel": "NO"
  },
  "type": "STORAGE",
  "site": "hpc.lsu.edu",
  "default": false,
  "public": false,
  "globalDefault": false,
  "name": "Shelob at LSU",
  "id": "shelob-storage-hpctrn20",
  "status": "UP"
}
```
<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Auth Server Port</td>
<td>22</td>
</tr>
<tr>
<td>Root Directory</td>
<td>/</td>
</tr>
<tr>
<td>Home Directory</td>
<td>/home/hpctrn20</td>
</tr>
<tr>
<td>Proxy Tunnel</td>
<td>NO</td>
</tr>
<tr>
<td>Storage Authentication Type</td>
<td>PASSWORD</td>
</tr>
<tr>
<td>Username</td>
<td>hpctrn20</td>
</tr>
<tr>
<td>Password</td>
<td>***********************</td>
</tr>
</tbody>
</table>

JavaScript code:
```javascript
},
  "type": "STORAGE",
  "site": "hpc.lsu.edu",
  "default": false,
  "public": false,
  "globalDefault": false,
  "name": "Shelob at LSU",
  "id": "shelob-storage-hpctrn20",
  "status": "UP"
```
System Builder Wizard

You have successfully created your system

ID
shelob-storage-hpctrn20

Name
Shelob at LSU

Status
UP

Type
STORAGE

Description
The Shelob supercomputer at LSU
Systems Management

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Type</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>condor.opensciencegrid.org</td>
<td>Open Science Grid</td>
<td>EXECUTION</td>
<td>Actions</td>
</tr>
<tr>
<td>data.agaveapi.co</td>
<td>Agave Cloud Storage</td>
<td>STORAGE</td>
<td>Actions</td>
</tr>
<tr>
<td>docker.tacc.utexas.edu</td>
<td>Demo Docker VM</td>
<td>EXECUTION</td>
<td>Actions</td>
</tr>
<tr>
<td>ktraxler-qb</td>
<td>QB at LONI</td>
<td>EXECUTION</td>
<td>Actions</td>
</tr>
<tr>
<td>qb-ktraxler</td>
<td>qb (ktraxler)</td>
<td>EXECUTION</td>
<td>Actions</td>
</tr>
<tr>
<td>qb-storage-ktraxler</td>
<td>qb storage (ktraxler)</td>
<td>STORAGE</td>
<td>Actions</td>
</tr>
<tr>
<td>shelob-sbrandt</td>
<td>Shelob LSU (sbrandt)</td>
<td>EXECUTION</td>
<td>Actions</td>
</tr>
</tbody>
</table>
Choose Execution this time
```json
{
    "maxSystemJobs": 50,
    "executionType": "HPC",
    "available": true,
    "description": "Shelob supercomputer at LSU",
    "storage": {
        "proxy": null,
        "protocol": "SFTP",
        "mirror": false,
        "port": 10,
        "auth": {
            "type": "PASSWORD"
        },
        "host": "shelob.hpc.lsu.edu",
        "rootDir": "/",
        "homeDir": "/home/hpctrn20",
        "proxyTunnel": "NO"
    }
}
```
Scheduler
TORQUE

Maximum System Jobs
50

Scratch Directory
/work/hpctrm20/

Work Directory
/work/hpctrm20/

Environment

Startup Script

Queues

Name
shelob

Arbitrary name for the queue. This will be used in the job submission process, so it should line up with the name of an actual queue on the execution system.

Maximum Jobs
20
Queues

Name
shelob

Arbitrary name for the queue. This will be used in the job submission process, so it should line up with the name of an actual queue on the execution system.

Maximum Jobs
20

Maximum number of jobs that can be queued or running within this queue at a given time. Defaults to 10. -1 for no limit.

Maximum Nodes
203

Maximum number of nodes that can be requested for any job in this queue. -1 for no limit.

Maximum Memory Per Node
64GB

Maximum memory per node for jobs submitted to this queue in ###.###EIPITG format.

Maximum Processors Per Node
Maximum Memory Per Node

64GB

Maximum memory per node for jobs submitted to this queue in #.###.#EIP$ITI$G format

Maximum Processors Per Node

16

Maximum number of processors per node that can be requested for any job in this queue. -1 for no limit

Maximum Requested Time

72:00:00

Maximum run time for any job in this queue given in hh:mm:ss format

Custom directive

Arbitrary text that will be appended to the end of the scheduler directives in a batch submit script. This could include a project number, system-specific directives, etc

Default

True if this is the default job for the queue (*otherwise false*)
Type following lines into the “Custom Directives” box

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>#PBS</td>
<td>-A</td>
<td>hpc_cmr (your allocation)</td>
</tr>
<tr>
<td>#PBS</td>
<td>-q</td>
<td>checkpoint</td>
</tr>
<tr>
<td>#PBS</td>
<td>-l</td>
<td>nodes=${AGAVE_JOB_NODE_COUNT}:ppn=16</td>
</tr>
</tbody>
</table>
can be requested for any job in this queue. -1 for no limit

Maximum Requested Time

72:00:00

Maximum run time for any job in this queue given in hh:mm:ss format

Custom directive

Arbitrary text that will be appended to the end of the scheduler directives in a batch submit script. This could include a project number, system-specific directives, etc.

Default

True if this is the default queue for the system, false otherwise

+ Add

Previous  Next  Submit

Back to Systems  Use Definition
Login Authentication

**Type**
- PASSWORD

**Username**
- hpctrl20

**Password**
- [Redacted]

Storage

**Protocol**
- SFTP

**Host**
- shelob.hpc.lsu.edu

**System Auth Server Port**
- 10

**Root Directory**
- /

**Home Directory**
- /home/hpctrl20

**Proxy Tunnel**
- NO

```json
{
    "type": "TOMCRED",
    "username": "hpctrl20",
    "password": "f=m&one23four",
    "host": "shelob.hpc.lsu.edu",
    "proxyTunnel": "NO",
    "startupScript": null,
    "scheduler": "TORQUE",
    "default": false,
    "public": false,
    "maxSystemJobsPerUser": 10,
    "id": "shelob-execu-hpctrl20",
    "workDir": "/work/hpctrl20/",
    "owner": "ktraxler",
    "site": "hpc.lsu.edu",
    "environment": "",
    "queues": [
        {
            "maxJobs": 20,
            "maxMemoryPerNode": "64GB",
            "default": false,
            "maxRequestedTime": "72:00:00",
            "name": "shelob",
            "description": null,
            "maxNodes": 203,
            "maxProcessorsPerNode": 16,
            "mappedName": null,
            "maxUserJobs": -1,
            "customDirectives": null
        }
    ],
    "globalDefault": false,
    "name": "Shelob supercomputer at LSU",
    "status": "UP",
    "scratchDir": "/work/hpctrl20/"
}
```
System Auth Server Port
10
Root Directory
/
Home Directory
/home/hpctrn20
Proxy Tunnel
NO

Storage Authentication
Type
PASSWORD
Username
hpctrn20
Password

{'maxNodes': 203,
'maxProcessorsPerNode': 16,
'mappedName': null,
'maxUserJobs': -1,
'customDirectives': null
}

"globalDefault": false,
"name": "Shelob supercomputer at LSU",
"status": "UP",
"scratchDir": "/work/hpctrn20/"
**System Builder Wizard**

You have successfully created/updated your system

<table>
<thead>
<tr>
<th>ID</th>
<th>shelob-execu-hpctrn20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Shelob supercomputer at LSU</td>
</tr>
<tr>
<td>Status</td>
<td>UP</td>
</tr>
<tr>
<td>Type</td>
<td>EXECUTION</td>
</tr>
<tr>
<td>Description</td>
<td>Shelob supercomputer at LSU</td>
</tr>
</tbody>
</table>
## File Browser

### Shelob at LSU:

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>.gnome2</td>
<td>4KB</td>
<td>3 days ago</td>
</tr>
<tr>
<td>.mozilla</td>
<td>4KB</td>
<td>3 days ago</td>
</tr>
<tr>
<td>.pki</td>
<td>4KB</td>
<td>8 hours ago</td>
</tr>
<tr>
<td>.ssh</td>
<td>4KB</td>
<td>9 hours ago</td>
</tr>
<tr>
<td>.subversion</td>
<td>4KB</td>
<td>3 days ago</td>
</tr>
<tr>
<td>agave-deployment</td>
<td>4KB</td>
<td>2 hours ago</td>
</tr>
<tr>
<td>drawgau</td>
<td>4KB</td>
<td>7 hours ago</td>
</tr>
<tr>
<td>.bash_history</td>
<td>28.5KB</td>
<td>a few seconds ago</td>
</tr>
<tr>
<td>.bash_history3</td>
<td>364B</td>
<td>7 hours ago</td>
</tr>
<tr>
<td>.bash_logout</td>
<td>18B</td>
<td>3 days ago</td>
</tr>
<tr>
<td>.bash_profile</td>
<td>176B</td>
<td>3 days ago</td>
</tr>
<tr>
<td>.bashrc</td>
<td>124B</td>
<td>3 days ago</td>
</tr>
<tr>
<td>.emacs</td>
<td>500B</td>
<td>3 days ago</td>
</tr>
</tbody>
</table>
Create an Agave App

- Back to working with Agave TOGO to run our job
## Apps Management

Manage your collection of apps

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Version</th>
<th>Label</th>
<th>Short Description</th>
<th>Exec. System</th>
<th>Public</th>
<th>Revision</th>
<th>Last Modified</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>drawgauktraxler-0.1.0</td>
<td>drawgauktraxler</td>
<td>0.1.0</td>
<td>draw curve</td>
<td>gaussian curve</td>
<td>qb-exec-ktraxler</td>
<td>false</td>
<td>1</td>
<td>2 days ago</td>
<td>Actions</td>
</tr>
<tr>
<td>shell-runner-0.1.0</td>
<td>shell-runner</td>
<td>0.1.0</td>
<td>Execute a command at a shell</td>
<td>This will execute whatever command you give in the command parameter</td>
<td>qb-exec-ktraxler</td>
<td>false</td>
<td>4</td>
<td>2 days ago</td>
<td>Actions</td>
</tr>
<tr>
<td>cloud-runner-0.1.0u1</td>
<td>cloud-runner</td>
<td>0.1.0</td>
<td>Run your code in the cloud</td>
<td>Generic template for running arbitrary code in Agave's Dockerized cloud.</td>
<td>docker.tacc.utexas.edu</td>
<td>true</td>
<td>1</td>
<td>5 months ago</td>
<td>Actions</td>
</tr>
<tr>
<td>jfonner-fork-1.0</td>
<td>jfonner-fork</td>
<td>1.0</td>
<td>Remote command execution script</td>
<td>Simple app for running a user-defined command on a remote system</td>
<td>stampede-fonner</td>
<td>false</td>
<td>1</td>
<td>5 months ago</td>
<td>Actions</td>
</tr>
<tr>
<td>cactus-</td>
<td>cactus-</td>
<td>1.0</td>
<td>Cactus</td>
<td>Solves PDEs</td>
<td>shelob-</td>
<td>false</td>
<td>3</td>
<td>5 months ago</td>
<td>Actions</td>
</tr>
</tbody>
</table>
App Builder Wizard

You can re-use an existing App definition template and provide your own Name and Dependencies:

**APP BUILDER WIZARD - STEP 1 OF 6**

1. Basics
2. Dependencies
3. Environment
4. Parameters

Name
shell-runner

The name of the application. The name does not have to be unique, but the combination of name and version does.

Version
0.1.0

The version of the application in `#.##.#` format. While the version does not need to be unique, the combination of name and version does have to be unique.

Label
Execute a command at a shell

Label for use in forms generated by the jobs service

```json
{
    "name": "shell-runner",
    "version": "0.1.0",
    "helpURI": "http://developer.agaveapi.co/",
    "label": "Execute a command at a shell",
    "defaultNodeCount": 1,
    "defaultMaxRunTime": "01:00:00",
    "shortDescription": "This will execute whatever command you give in the command parameter",
    "longDescription": "This will execute whatever command you give in the command parameter",
    "executionSystem": "",
    "executionType": "CLI",
    "parallelism": "SERIAL",
    "deploymentPath": "ktraxler/apps/shell-runner-0.1.0",
    "deploymentSystem": "storage.example.com",
    "templatePath": "wrapper.sh",
    "testPath": "test/test.sh",
    "tags": [
        "execute",
        "awesome",
        "demo"
    ]
}
```
The name of the application. The name does not have to be unique, but the combination of name and version does.

Version

The version of the application in #.#.# format. While the version does not need to be unique, the combination of name and version does have to be unique.

Label

Label for use in forms generated by the jobs service

Short description

Short description of this app

Long description

Draw Gaussian Curve (points only)
draw curve

Draw Gaussian Curve (points only)

execute

awesome
demo

Array of terms you may associate with this app

Help URL
http://developer.agaveapi.co/

The URL where users can go for more information about the app.

Ontology

"awesome",
"demo",
"executionType": "HPC",
"executionSystem": "shelob-execution-hpctrn20",
"deploymentPath": "agave-deployment",
"deploymentSystem": "shelob-storage-hpctrn20",
"templatePath": "drawgau-wrapper.txt",
"testPath": "test.txt",
"checkpointable": false,
"modules": [],
"inputs": [
{
"id": "parfile",
"value": {
"validator": "",
"visible": true,
"required": true,
"order": 0,
"enquote": false,
"default": "input.txt"
},
"details": {
"label": "input for the program",
"description": null,
"argument": "input.txt",
"showArgument": false,
"repeatArgument": false
},
"semantics": {
"minCardinality": 1,
"maxCardinality": 1,
"ontology": [],
"fileTypes": []
}
Array of terms you may associate with this app

Help URL

http://developer.agaveapi.co/

The URL where users can go for more information about the app.

Ontology

ontology

execute

ontology

awesome

ontology

demo

An array of ontology terms describing this app.

Previous  Next  Submit
Deployment path

**agave-deployment**

The path to the folder on the deployment system containing the application wrapper and dependencies.

Deployment system

**shelob-storage-hpctrn20**

The ID of the storage system where this app's assets should be stored.

Wrapper script

**drawgau-wraper.txt**

The path to the wrapper script relative to the deploymentPath.

Test script

**test.txt**

The path to the test script relative to the deploymentPath.

```
1 {
2  "id": "drawgau2-shelob-hpctrn20-0.1.0",
3  "name": "drawgau2-shelob-hpctrn20",
4  "icon": null,
5  "parallelism": "SERIAL",
6  "defaultProcessorsPerNode": 2,
7  "defaultMemoryPerNode": 64,
8  "defaultNodeCount": 1,
9  "defaultMaxRunTime": "00:10:00",
10  "defaultQueue": "shelob",
11  "version": "0.1.0",
12  "isPublic": false,
13  "helpURI": "http://developer.agaveapi.co/",
14  "label": "Drawgau",
15  "owner": "ktraxlex",
16  "shortDescription": "draw curve",
17  "longDescription": "Draw Gaussian Curve (points only)",
18  "tags": [
19     "execute",
20     "awesome",
21     "demo"
22  ],
23  "ontology": [
24     "execute",
25     "awesome",
26     "demo"
27  ],
28  "executionType": "HPC",
29  "executionSystem": "shelob-exec-hpctrn20",
30  "deploymentPath": "agave-deployment",
31  }
```
Deployment path

```
agave-deployment
```

The path to the folder on the deployment system containing the application wrapper and dependencies.

Deployment system

```
shelob-storage-hpctrn20
```

The ID of the storage system where this app's assets should be stored.

Wrapper script

```
drawgauwrapper.txt
```

The path to the wrapper script relative to the deploymentPath.

Test script

```
test.txt
```

The path to the test script relative to the deploymentPath.

An array of modules to load prior to the execution of the application. This is only relevant when you use the unix Modules or LMOD utilities to manage dependencies on the app execution system.
agave-deployment

The path to the folder on the deployment system containing the application wrapper and dependencies.

Deployment system

shelob-storage-hpctrn20

The ID of the storage system where this app's assets should be stored.

Wrapper script

drawgau-wrapper.txt

The path to the wrapper script relative to the deploymentPath.

Test script

test.txt

The path to the test script relative to the deploymentPath.

An array of modules to load prior to the execution of the application. This is only relevant when you use the unix Modules or LMOD utilities to manage dependencies on the app execution system.
App Edit Wizard

Execution type
HPC

The execution type of the application. If you're unsure, it's probably HPC.

Execution system
shelob-execu-hpctrn20

The ID of the execution system where this app should run.

Default queue

Default queue to use when submitting this job if none is provided in the job request. Can be left blank and a queue will be determined at run time.

Default node count
1

Default number of nodes to be used when running this app if no node count is given in the job request.

```json
{
    "id": "drawgau2-shelob-hpctrn20-0.1.0",
    "name": "drawgau2-shelob-hpctrn20",
    "icon": null,
    "parallelism": "SERIAL",
    "defaultProcessorsPerNode": 2,
    "defaultMemoryPerNode": 64,
    "defaultNodeCount": 1,
    "defaultMaxRunTime": "00:10:00",
    "defaultQueue": "shelob",
    "version": "0.1.0",
    "isPublic": false,
    "helpURI": "http://developer.agaveapi.co/",
    "label": "Drawgau",
    "owner": "ktaxler",
    "shortDescription": "draw curve",
    "longDescription": "Draw Gaussian Curve (points only)",
    "tags": [
        "executes",
        "awesome",
        "demo"
    ],
    "ontology": [
        "execute",
        "awesome",
        "demo"
    ],
    "executionType": "HPC",
    "executionSystem": "shelob-execu-hpctrn20",
    "deploymentPath": "agave-"
}
```
Default node count

1

Default number of nodes to be used when running this app if no node count is given in the job request

Default memory (GB)

64

Default memory in GB to be used when running this app if no memory is given in the job request

Default processor count

2

Default number of processors per node to be used when running this app if no processor count is given in the job request

Default run time

00:10:00

Default max run time to be used when running this app if no requested run time is given in the job request

Parallelism

SERIAL

The parallelism type of the application. If you're unsure, it's probably SERIAL.

Checkpointable

True  False

Does this app support checkpointing?

"demo",
"executionType": "HFC",
"executionSystem": "shelob-exec-hptrn20",
"deploymentPath": "agave-deployment",
"deploymentSystem": "shelob-storage-hptrn20",
"templatePath": "drggaue-wrapper.txt",
"testPath": "test.txt",
"checkpointable": false,
"modules": [],
"inputs": [
  {
    "id": "param",
    "value": {
      "validator": "",
      "default": "param",
      "description": "Input parameter name."
    }
  },
  {
    "id": "param2",
    "value": {
      "validator": "",
      "default": "param2",
      "description": "Input parameter description."
    }
  }
]
{
  "id": "drawgau2-shelob-hpctmn20-0.1.0",
  "name": "drawgau2-shelob-hpctmn20",
  "icon": null,
  "parallelism": "SERIAL",
  "defaultProcessorsPerNode": 2,
  "defaultMemoryPerNode": 64,
  "defaultNodeCount": 1,
  "defaultMaxRunTime": "00:10:00",
  "defaultQueue": "Shelob",
  "version": "0.1.0",
  "isPublic": false,
  "helpURI": "http://developer.agaveapi.co/",
  "label": "Drawgau",
  "owner": "ktraxler",
  "shortDescription": "draw curve",
  "longDescription": "Draw Gaussian Curve (points only)",
  "tags": [
    "execute",
    "awesome",
    "demo"
  ],
  "ontology": [
    "execute",
    "awesome",
    "demo"
  ],
  "executionType": "HPC",
  "executionSystem": "shelob-exec-hpctmn20"
}
App Edit Wizard

ID
parfile

The unique identifier for this input file. This will be referenced in the wrapper script.

Details

Descriptive details about this app inputs used in form generation.

Label
input for the program

The label displayed for this input.

Prepend command line argument?
True  False

Should this command line argument be injected into the command line?

```json
1 {
2   "id": "drawgau2-shelob-hpctrn20-0.1.0",
3   "name": "drawgau2-shelob-hpctrn20",
4   "icon": null,
5   "parallelism": "SERIAL",
6   "defaultProcessorsPerNode": 2,
7   "defaultMemoryPerNode": 64,
8   "defaultNodeCount": 1,
9   "defaultMaxRunTime": "00:10:00",
10  "defaultQueue": "shelob",
11  "version": "0.1.0",
12  "isPublic": false,
13  "helpURI": "http://developer.agaveapi.co/",
14  "label": "Drawgau",
15  "owner": "ktraxler",
16  "shortDescription": "draw curve",
17  "longDescription": "Draw Gaussian Curve (points only)",
18  "tags": [
19    "execute",
20    "awesome",
21    "demo"
22  ],
23  "ontology": [
24    "execute",
25    "awesome",
26    "demo"
27  ],
28  "executionType": "HPC",
29  "executionSystem": "shelob-exuc-hpctrn20",
```
Semantics

Semantic information about the input field.

Ontology

Array of ontology terms describing this input.

Min cardinality

1

Max number of instances of this input per job.

Max cardinality

1

Values

Default value and validations for the input field.

Default value

input.txt

Validator regex

The regular expression used to validate this parameter value.

Visible

Yes

Deployment

deploymentSystem: "shelob-storage-hpctrn20",

templatePath: "drawgau-wrapper.txt",

testPath: "test.txt",

checkpointable: false,

modules: [],

inputs: [

{"id": "parfile",

"value": {

"validator": ",",

"visible": true,

"required": true,

"order": 0,

"enquote": false,

"default": "input.txt"

},

"details": {

"label": "input for the program",

"description": null,

"argument": "input.txt",}]}
Prepend command line argument?
- True
- False

Should this command line argument be injected into the submit script preceding the input?

**Semantics**

Semantic information about the input field.

**Ontology**

Array of ontology terms describing this input.

**Min cardinality**

1

Minimum number of instances of this input per job.

**Max cardinality**

1

Max number of instances of this input per job.

**Values**

Default value and validations for the input field.

**Default value**

Input.txt

Validator regex
Default value and validations for the input field.

Default value

input.txt

Default value

Validator regex

The regular expression used to validate this parameter value.

Visible

Yes  No

Should this parameter be visible? If not, there must be a default and it will be required.

Required

Yes  No

Is this parameter required? If visible is false, this must be true.

Order

0

The order in which this parameter should be printed when generating an execution command for forked execution. This will also be the order in which parameters are returned in the response json.
App Edit Wizard

Basics

1

drawgau0.txt  + Add output

ID

drawgau0.txt  ✓

The unique identifier for this input file. This will be referenced in the wrapper script.

Details

Descriptive details about this app outputs used in form generation.

Label

output file  ✓

The label displayed for this input.

Prepend command line argument?

True  False

Should this command line argument be injected into the command.

Environment

Dependencies

Dependencies

Parallelism

0

Default number of processes per node.

Default processors per node

2

Default memory per node

64

Default node count

1

Default max run time

00:10:00

Default queue

[drawgau2-shelob-hpctrn20]

Version

0.1.0

Is public

false

Help URI

http://developer.agaveapi.co/

Label

Drawgau

Owner

ktraxler

Short description

Draw Gaussian Curve (points only)

Tags

execute, awesome, demo

Ontology

execute, awesome, demo

Execution type

HPC

Execution system

shelob-hpctrn20
Prepend command line argument?

True  False

Should this command line argument be injected into the submit script preceding the input?

Semantics

Semantic information about the output field.

Ontology

Array of ontology terms describing this input.

Min cardinality

1

Minimum number of instances of this input per job.

Max cardinality

-1

Max number of instances of this input per job.

Values

Default value and validations for the output field.

Default value

drawgau0.txt

Validator regex


Values

Default value and validations for the output field.

Default value

**drawgau0.txt**

Validator regex

The regular expression used to validate this parameter value.

Visible

Yes  No

Should this parameter be visible? If not, there must be a default and it will be required.

Order

1

The order in which this parameter should be printed when generating an execution command for forked execution. This will also be the order in which parameters are returned in the response json.

---

```json
"label": "input for the program",
"description": null,
"argument": "input.txt",
"showArgument": false,
"repeatArgument": false
",
"semantics": {
"minCardinality": 1,
"maxCardinality": 1,
"ontology": [],
"fileTypes": []
}
",
"parameters": [],
"outputs": [
{
"id": "drawgau0.txt",
"value": {
"validator": "",
"order": 1,
"default": "drawgau0.txt"
},
"details": {
"label": "output file",
"description": null
}
},
"semantics": {
"minCardinality": 1,
"maxCardinality": -1,
```
You have successfully created/updated your app

<table>
<thead>
<tr>
<th>Name</th>
<th>drawgau2-shelob-hpctrn20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>0.1.0</td>
</tr>
<tr>
<td>Label</td>
<td>Drawgau</td>
</tr>
<tr>
<td>Description</td>
<td>draw curve</td>
</tr>
<tr>
<td>Execution Type</td>
<td>HPC</td>
</tr>
<tr>
<td>Execution System</td>
<td>shelob-execu-hpctrn20</td>
</tr>
<tr>
<td>Parallelism</td>
<td>SERIAL</td>
</tr>
<tr>
<td>Deployment System</td>
<td>shelob-storage-hpctrn20</td>
</tr>
<tr>
<td>Deployment Path</td>
<td>agave-deployment</td>
</tr>
</tbody>
</table>
### File Browser

**Agave Cloud Storage**: `ktraxler/archive/jobs/job-964575848630709785-242ac11c-0001-007`

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>.agave.log</code></td>
<td>398B</td>
<td>2 minutes ago</td>
</tr>
<tr>
<td><code>drawgau-0612-0827-964575848630709785-242ac11c-0001-007.err</code></td>
<td>1.1KB</td>
<td>2 minutes ago</td>
</tr>
<tr>
<td><code>drawgau-0612-0827-964575848630709785-242ac11c-0001-007.out</code></td>
<td>5.8KB</td>
<td>2 minutes ago</td>
</tr>
<tr>
<td><code>nodefile.txt</code></td>
<td>10B</td>
<td>2 minutes ago</td>
</tr>
</tbody>
</table>
### Apps Management

**Manage your collection of apps**

#### Apps Management

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Label</th>
<th>Short Description</th>
<th>Exec. System</th>
<th>Last Modified</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>drawgau2-shelob-hpctrn20-0.1.0</td>
<td>drawgau2-shelob-hpctrn20</td>
<td>Drawgau</td>
<td>draw curve</td>
<td>shelob-exocu-hpctrn20</td>
<td>a minute ago</td>
<td></td>
</tr>
<tr>
<td>drawgau-shelob-hpctrn20-0.10</td>
<td>drawgau-shelob-hpctrn20</td>
<td>draw curve</td>
<td>draw curve</td>
<td>shelob-exocu-hpctrn20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>drawgau-shelob-hpctrn15-0.1.0</td>
<td>drawgau-shelob-hpctrn15</td>
<td>draw curve</td>
<td>draw curve</td>
<td>shelob-exocu-hpctrn15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>drawgau-lsu-tutorial-0.1.0</td>
<td>drawgau-lsu-tutorial</td>
<td>draw curve</td>
<td>gaussian curve</td>
<td>shelob-hpctrn14</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**Drawgau**

Draw Gaussian Curve (points only)

**Drawgau Documentation**

**Inputs**

input for the program

```plaintext
Select: agave://data.agaveapi.co/ktraxler/agave/input.txt
```

**Job details**

**Maximum job runtime**

00:10:00

In HH:MM:SS format. The maximum time you expect this job to run for. After this amount of time your job will be killed by the job scheduler. Shorter run times result in shorter queue wait times. Maximum possible time is 48:00:00 (48 hours).

**Job name**

```
drawgau_0611-728
```

A recognizable name for this job

**Batch Queue**

System queue to which the job should be submitted

- **Archive output**
  - Should the output be archived
Uncheck the “Archive output” box
### Successfully submitted job

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>964575848630709785-242ac11c-0001-007</td>
</tr>
<tr>
<td>Name</td>
<td>drawgau-0612-0827</td>
</tr>
<tr>
<td>Owner</td>
<td>ktraxler</td>
</tr>
<tr>
<td>Status</td>
<td>PENDING</td>
</tr>
<tr>
<td>Execution System</td>
<td>shelob-hpctm20-exec</td>
</tr>
<tr>
<td>Archive</td>
<td>true</td>
</tr>
<tr>
<td>Archive System</td>
<td>data.agaveapi.co</td>
</tr>
<tr>
<td>Archive Path</td>
<td>ktraxler/archive/jobs/job-964575848630709785-242ac11c-0001-007</td>
</tr>
<tr>
<td>Batch Queue</td>
<td>shelob</td>
</tr>
<tr>
<td>Application ID</td>
<td>drawgau2-shelob-hpctm20-0.1.0</td>
</tr>
</tbody>
</table>
Output from Job Run

- Log into your HPC account
- cd into /work/username
- ls -l
Conclusion

- Once you define your storage system, execution system, and app you have a working science gateway!
- We have successfully run a job without logging into our hardware system and using the command line.
- Save development time by using Agave.
Final thoughts

- If you want a demo or hands on for your group let us know:
  - ktraxler@lsu.edu
  - sbrandt@cct.lsu.edu