ESIP Lab Project

**Geoweaver: a web-based system for managing compound geospatial workflows of large-scale distributed deep networks**

October Progress Report

Prepared by

Ziheng Sun, Liping Di

Center for Spatial Information Science and Systems

George Mason University

20181031

Contents

[Executive Summary 2](#_Toc528771903)

[Project Actions 3](#_Toc528771904)

[System Architecture 3](#_Toc528771905)

[Geoweaver workflow designer 3](#_Toc528771906)

[Geoweaver bridge assembly 4](#_Toc528771907)

[Geoweaver Data Renderer 6](#_Toc528771908)

[Geoweaver GitHub repository 6](#_Toc528771909)

[Evaluation 8](#_Toc528771910)

[Next Steps 9](#_Toc528771911)

Executive Summary

This month Geoweaver team has carried out the project passing its half-way milestone as scheduled. Our progresses in October are listed in the following table:

|  |  |  |
| --- | --- | --- |
| **Milestones** | **Progress** | **Actions** |
| July 31: kick off, set up the development environment, develop web wrapper on top of open sourced deep learning/high performance computing library | 100% |  |
| Sep 30: develop workflow designer and data producer, complete bridge assembly between Geoweaver and data/function resources | 80% | * Geoweaver workflow interface has been intensively * The web ssh in bridge assembly has been developed. |
| Nov 30: complete module integration, create and conduct LSTM experiment | 40% | * A LSTM traing workflow is created in Geoweaver and being debugged to work through. |
| Jan 31: complete source code wrap-up, upload demonstration video, snapshot cloud instance, finish the GitHub final report and demonstrate it in ESIP winter 2019 | 40% | * Source code has been initially uploaded to Github. * The documents are attached as well. * Solved the vulnerable library security issues what are alerted by Github. |

Project Actions

System Architecture

The architecture design is put here to remind us to follow the original design and remember our goal during the development.

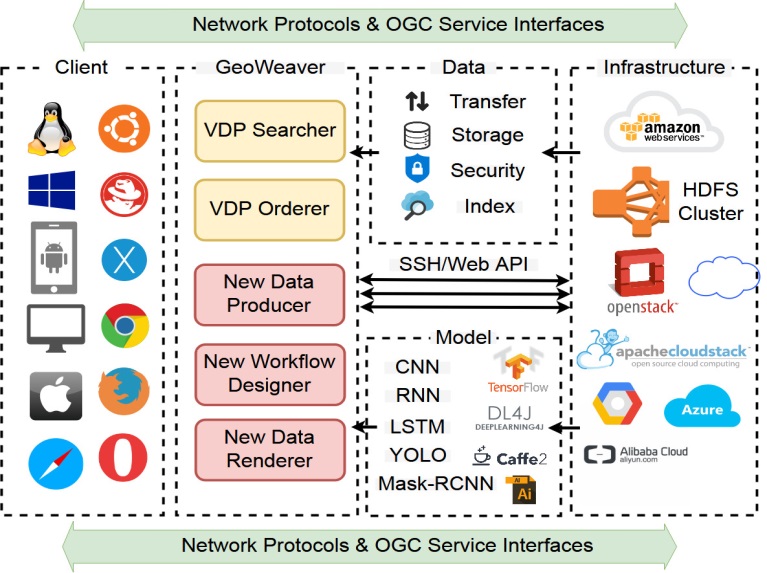


Figure 1. Architecutre design of Geweaver

Geoweaver workflow designer

The development of Geoweaver workflow designer has been at half-way point. It contains two major areas: workspace and side menu.

The side menu has three resource categories: host, process, and workflow.

* A host is an infrastructure which could be a PC, a linux, a rack server, a virtual machine or even a cluster.
* A process is an executable which could be a Shell script, a c/c++ program, a python code, a web service, etc. Process represents the atomic unit in business logic. Process must run on host and could run on various host as long as the environment was already set up.
* Workflow is a chain or just a graph of composition of processes.

The workspace is a graphic intuitive interface for users to visualize the three kinds of virtual resources. Fig. 2 shows an initial workflow of our LSTM experiment. It consists of:

* rescaling cropland data layers,
* downloading landsat band images,
* filtering cloud, shadow, bad pixels,
* reprojecting them into the same projection,
* aligning them to match by pixels,
* normalizing the pixel values,
* mapping CDL classes to numeric continuous values, and
* transforming the coordinates to a consistent form.

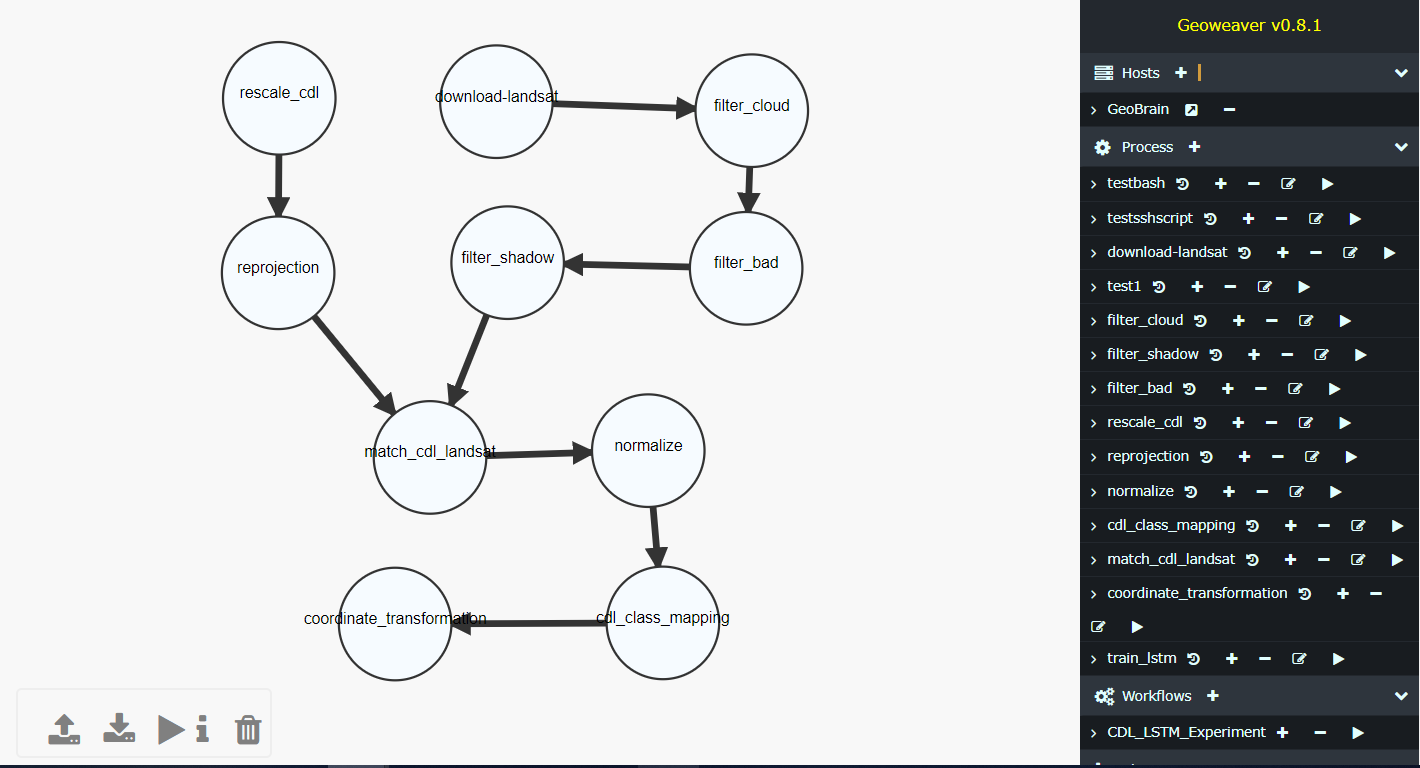


Figure 2. The initial training workflow of our LSTM experiment in Geoweaver



Figure 3. Workspace toolbar menu (from left to right: upload workflow, download workflow, execute workflow, show details of selected process, delete selected/all processes)

Geoweaver bridge assembly

To communicate with the involved hosts in geoweaver, we established a web ssh within Geoweaver. Scientists can use this entry to access their servers like they did via SSH secure client or putty. One advantage of having this web ssh is that scientists can easily capture their routine data preprocessing scripts and turn them into managed processes in Geoweaver. Geoweaver will take care of these steps and allow scientists to accomplish data processing steps by only one single click.

Fig. 3 shows the interface of adding a new host (server or virtual machine instances). Figure 4 shows the dialog which allows users to create customized processes like Shell scripts. Fig. 5 shows the interface of running the created processes/workflow on selected host.

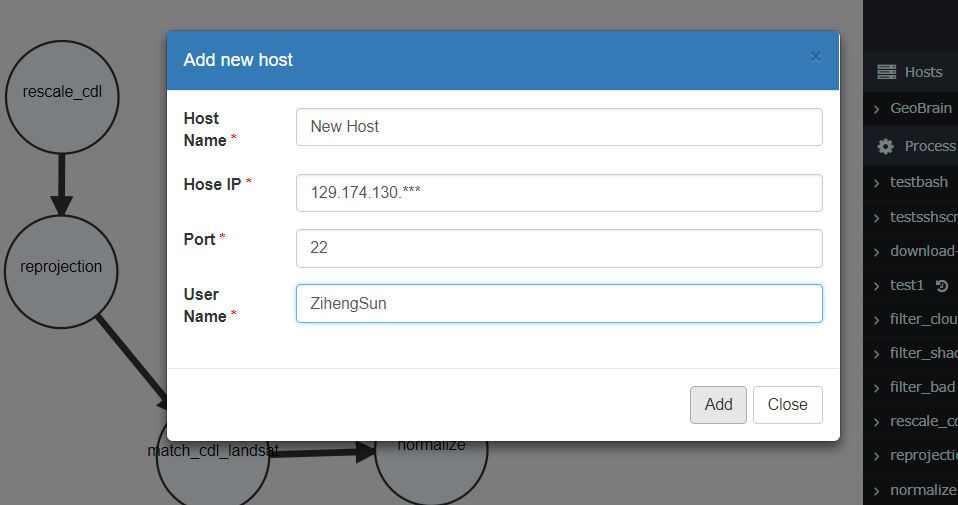


Figure 4. Add a new server to the host list

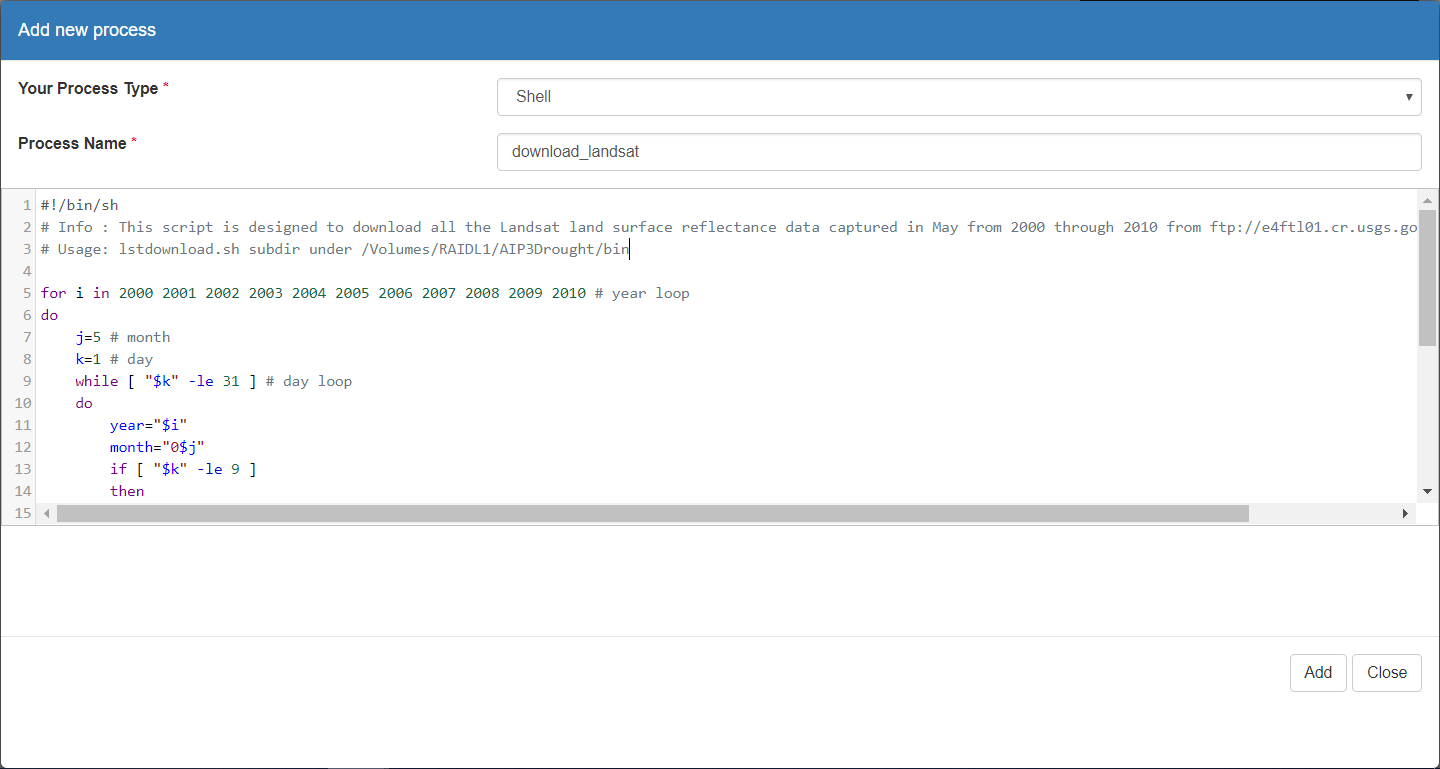


Figure 5. A dialog for creating the process of download\_landsat

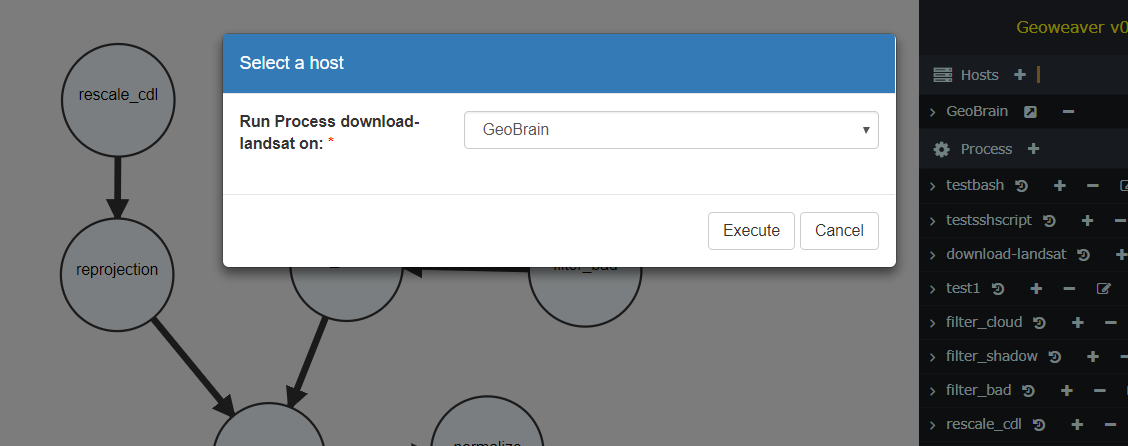


Figure 6. Run the processes and workflows on the selected host

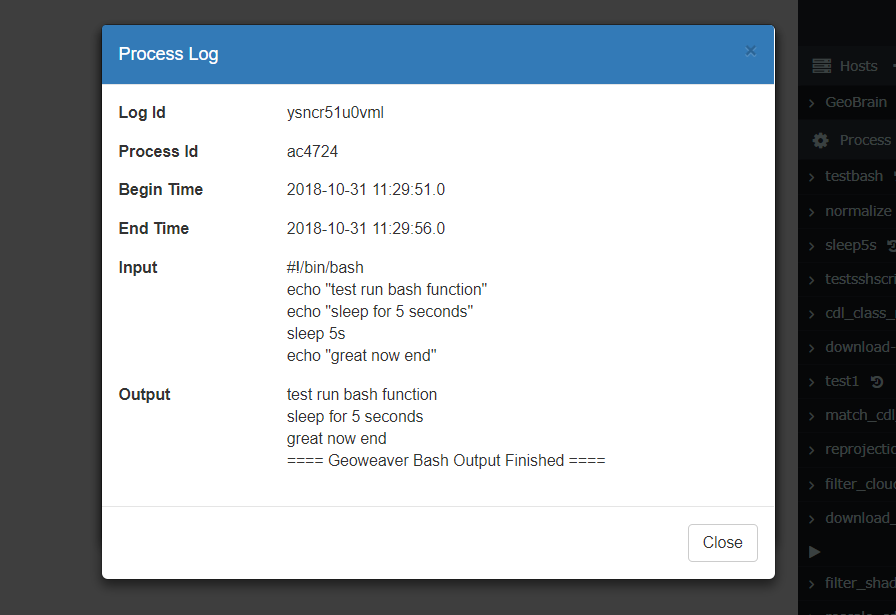


Figure 7. Process execution log

Geoweaver Data Renderer

Rendering module will reuse the module from CyberConnector.

Geoweaver GitHub repository

The github repository has been updated with source code and documents (<https://github.com/ESIPFed/Geoweaver>).

A Github page (<https://esipfed.github.io/Geoweaver/>) has been deployed online.

Evaluation

Not yet.

Next Steps

* Continue the development of Geoweaver workflow designer and bridge assembly.
* Complete the development of the LSTM experiment in a deployed Geoweaver instance.
* Create a cloud snapshot image or docker image of Geoweaver.
* Evaluate the performance and future of Geoweaver.
* Present Geoweaver in ESIP cluster telecons and winter conferences in Bethesda.
* Join in ESIP bootcamp to be trained on how to make Geoweaver into a sustainable success.