Multi-Cloud Resource Management using Apache Mesos for Planned Integration with Apache Airavata

Pankaj Saha, Madhusudhan Govindaraju

Department of Computer Science
Binghamton University
State University of New York (SUNY)

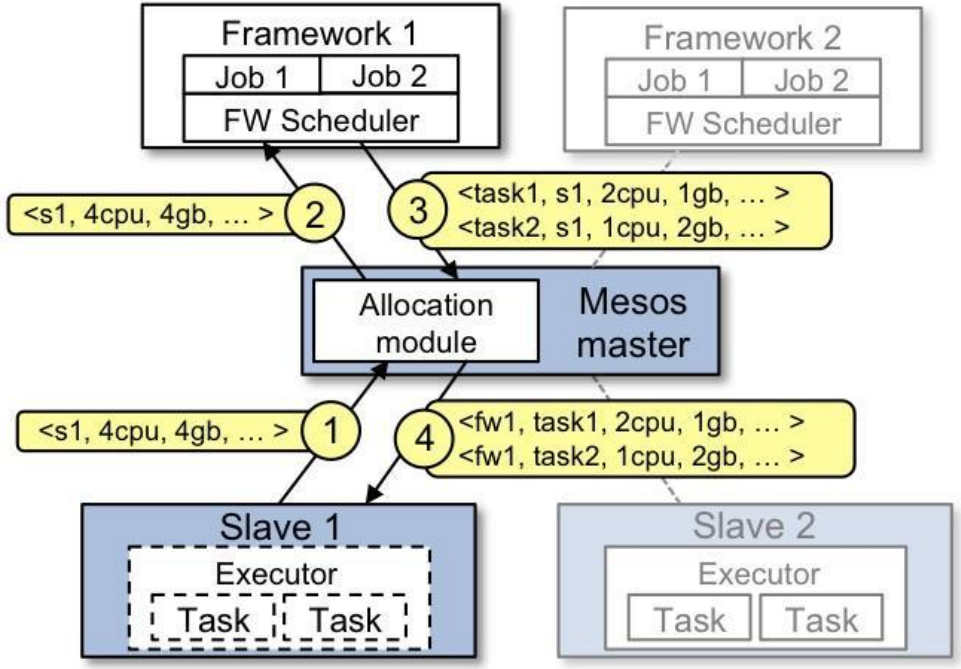
Suresh Marru, Marlon Pierce

Research Technologies
University Information Technology Services
Indiana University

Target Use Cases

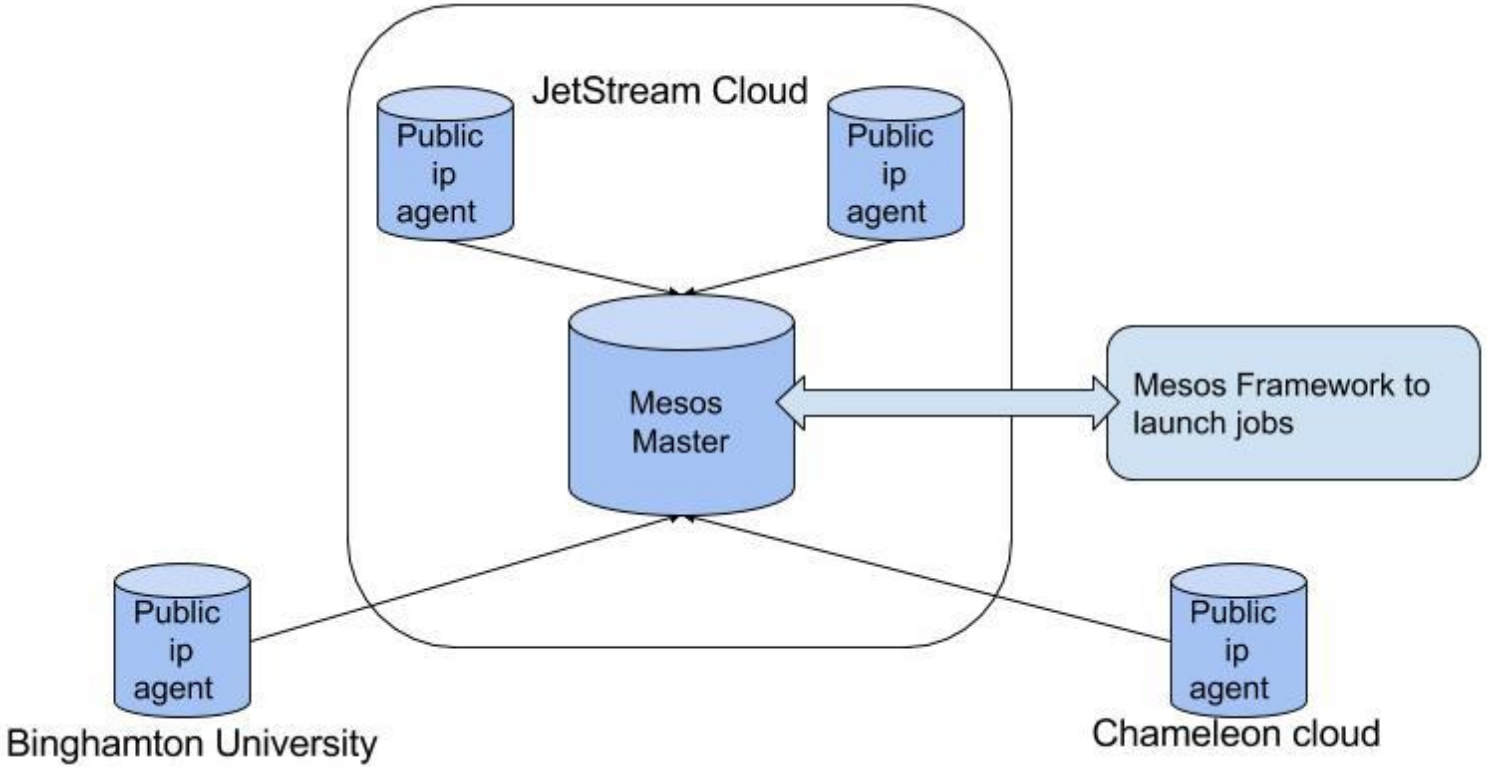
- Meta-Scheduling across clusters will help with the following cases
  - Large jobs gets stuck in job queue because scheduler needs enough resources available before launching it
  - Submitting many smaller jobs may have conflict with specific clusters policies and restrictions
  - Scientists having multiple cloud accounts wants to launch jobs on VMs across clouds.

Apache Mesos as a distributed kernel

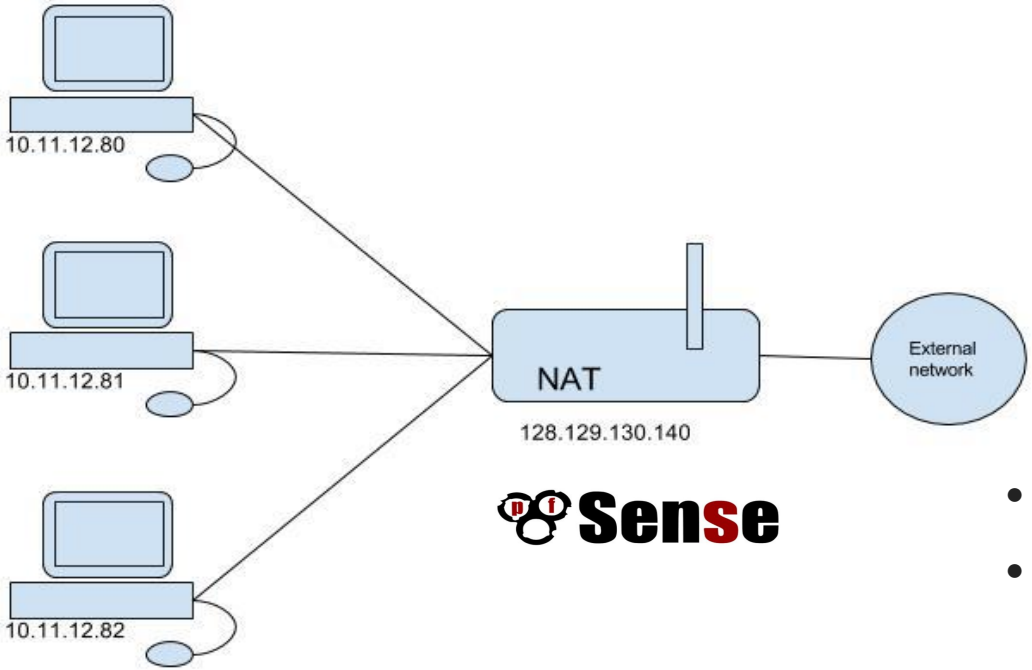


- Multi-resource scheduling
  - Abstracts CPU, memory, and storage to provide a global view for scheduling
- Proven scalability for 10K+ nodes in industry
  - Twitter, eBay, Apple, Groupon, Paypal, …
- Support for Docker

Multi cloud Mesos setup



pfSense as Network Address Translator

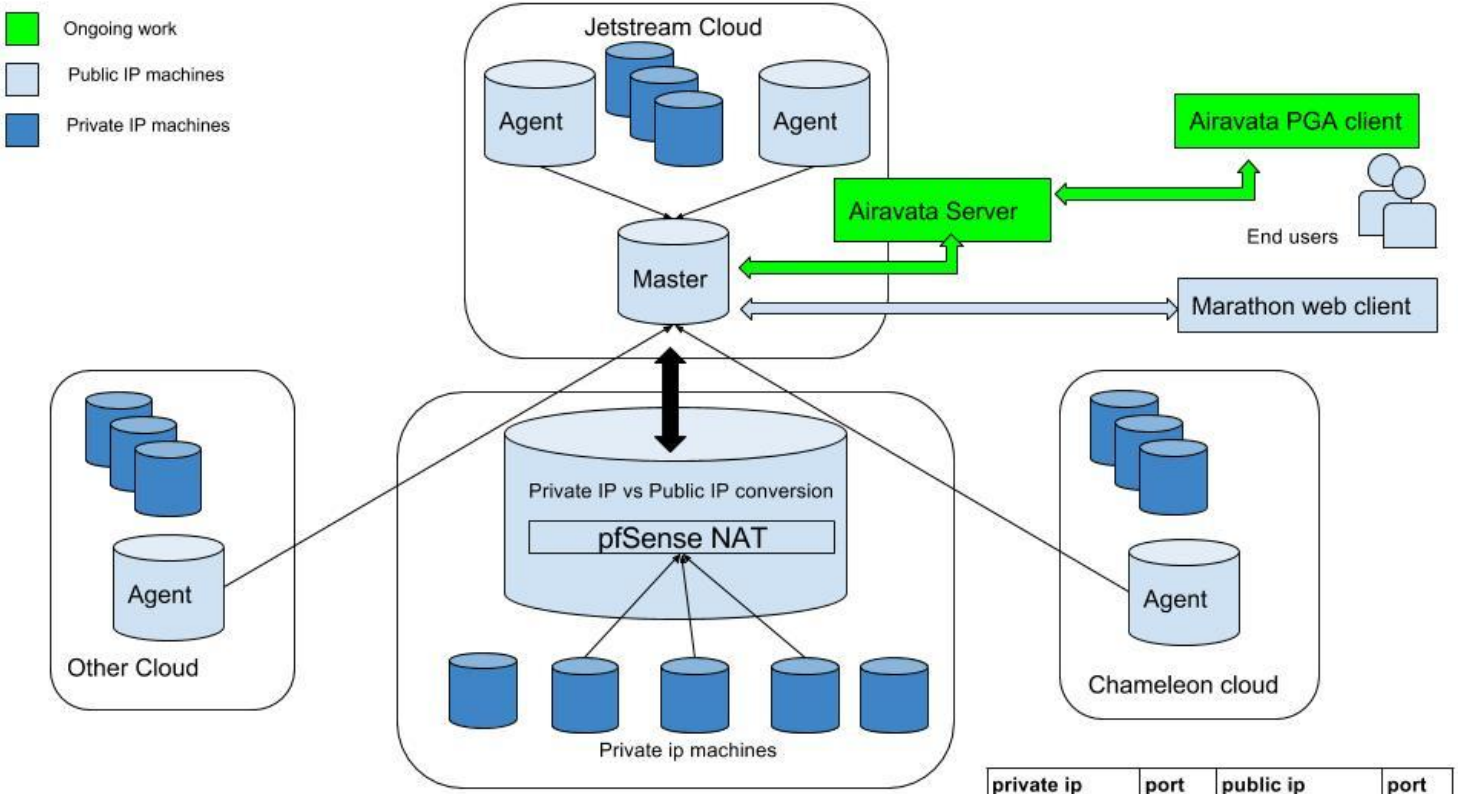


private ip	port	public ip	port
10.11.12.80	8080	128.129.130.140	8080
10.11.12.81	8081	128.129.130.140	8081
10.11.12.82	8082	128.129.130.140	8082

NAT forwarding Table

- pfSense is an open source software distribution based on **FreeBSD**.
- It can be installed on a physical computer or a virtual machine to make a dedicated firewall/router for a network

Private agents behind campus firewall



private ip	port	public ip	port
10.11.12.80	8080	128.129.130.140	8080
10.11.12.81	8081	128.129.130.140	8081
10.11.12.82	8082	128.129.130.140	8082

NAT Table

Initial Experience

Mesos Installation:

- Installation Mesos on ubuntu based cloud VM and campus bare metal was seamless

pfSense installation:

- Installing pfSense on Binghamton University campus machine was seamless
- We are working on how to install the similar setup on cloud based routers

Future work

- Replicate a software NAT on Jetstream and Chameleon cloud.
- User defined policies for scientific jobs can be implemented, instead of the default scheduling policies for jobs.
- Design custom backfilling algorithms that can run the jobs out of order.

Acknowledgement

- Appreciate Gateways 2016 committee for supporting my travel expenses.

Contact Us

Pankaj Saha (psaha4@binghamton.edu)
Madhu Govindaraju (mgovinda@binghamton.edu)
Marlon Pierce (marpierc@iu.edu)
Suresh Marru (smarru@iu.edu)

Apache Airavata: https://airavata.apache.org/community/mailing-lists.html
Apache Mesos: http://mesos.apache.org/community/