Breaking the 1.5MB Barrier: Running large AI/ML Metaflow flows on Argo

Saurabh Garg

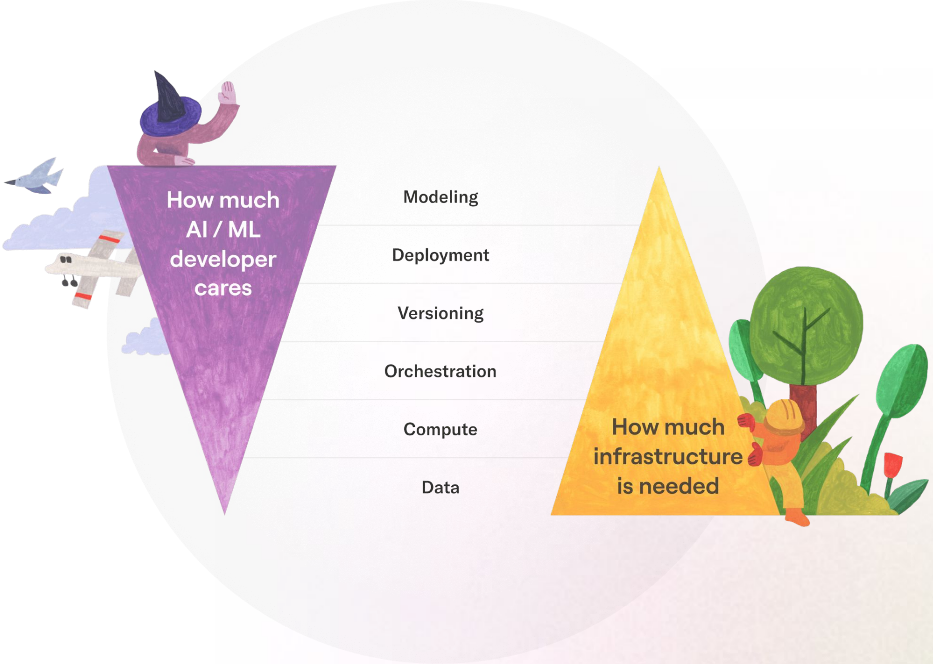
Outerbounds

Agenda

- Intro to Metaflow and Outerbounds platform
- AI/ML/DS workloads & Argo Workflows on k8s
- Breaking the 1.5MiB barrier
- Future work
- Q & A

What is Metaflow?

- Human friendly Python library to develop, deploy and operate data science/AI/ML applications
- Production grade deployments via Argo
- Track all flows/experiments and artifacts automatically
- Easy workflow construction, scale workflows via elastic cloud compute
- Access data from anywhere


How much AI / ML developer cares
Modeling
Deployment
Versioning
Orchestration
Compute
Data
How much infrastructure is needed

The Outerbounds Platform

All the building blocks required by real-world ML/AI systems

Compute

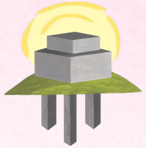
Versioning

Modeling

Data

Orchestration

Deployments



Customers

prime video · Goldman Sachs · NETFLIX · J.P.Morgan · amazon · intel.

Zillow · S&P Global · GE HealthCare · 23andMe · WARNER BROS. DISCOVERY · SIEMENS

Adobe · DELL · moz://a · TALA · Black Crow AI · Disney+

ESK · ARTERA · TRADE REPUBLIC · deliveroo · ramp · zendesk · A

carta · MERCK · MoneyLion · flexport. · deeptrust · Delivery Hero

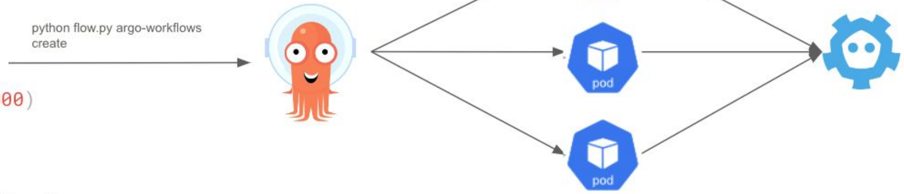
AI/ML/DS workloads can use many computers

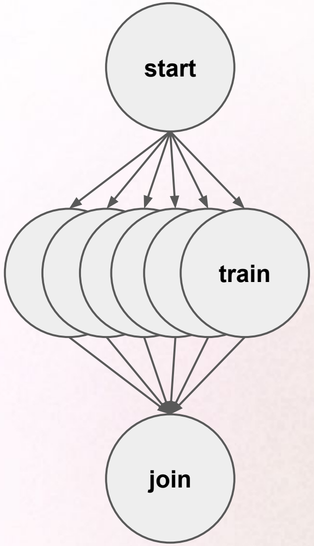
- Metaflow’s foreach construct allows you to process a cohort of different data points
- Many parallel copies of a single metaflow step are created
- Each copy of the train step gets mapped into a separate k8s container

```
@step
def start(self):
    self.params = list(range(100))
    self.next(self.train,
foreach='params')


@kubernetes
@resources(memory=128000)
@step
def train(self):
    self.model = train(...)
    self.next(self.join)

@step
def join(self, inputs):
```



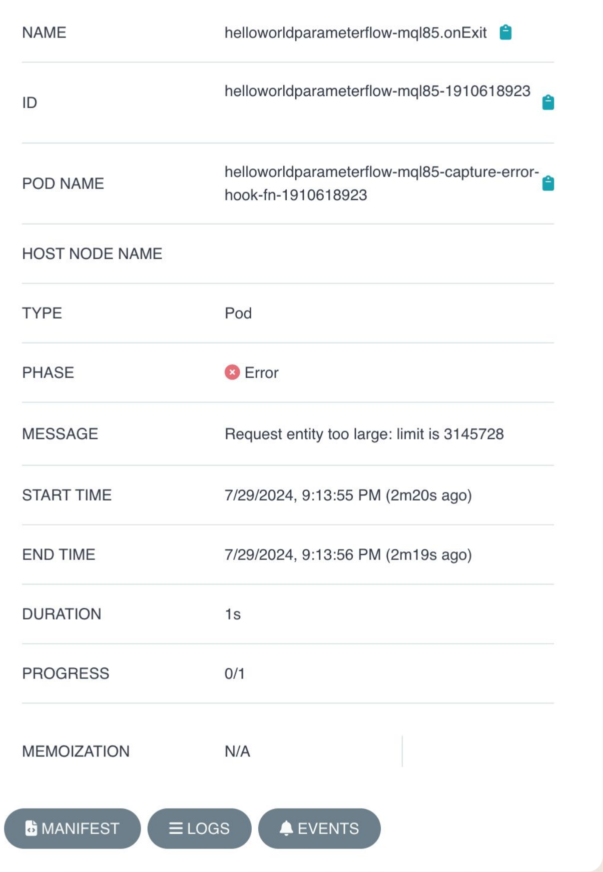


AI/ML/DS Large Argo Workloads on K8s is HARD!!

- Metaflow DAGs can be arbitrary in size but etcd limits max request size to 1.5MiB by default
    - Larger size of etcd requests will degrade latencies for other requests
- High throughput
    - Minimize e2e latency workloads experience in queues, webhook executions (validation/mutation)
- Equitable Resource Sharing
    - No single workload should be able to cannibalize all the resources
- Error Handling
    - Workloads can fail randomly (Network / IO / User / Infra etc).
    - Slow error detection/visualization/healing leads to wasted/repeated computations
- Distributed Training
    - Gang Scheduling (nested foreach's with parallel) is all or nothing

AI/ML/DS workloads with Argo Workflows on K8s

- Argo workflow.status stores the status of every node in the DAG
- If you utilize the foreach construct - you can potentially create thousands of nodes in your DAG
- If the request size* > 1.5MiB, argo will fail to update the workflow object - Request Entity too large
- Loss of work, wasted compute

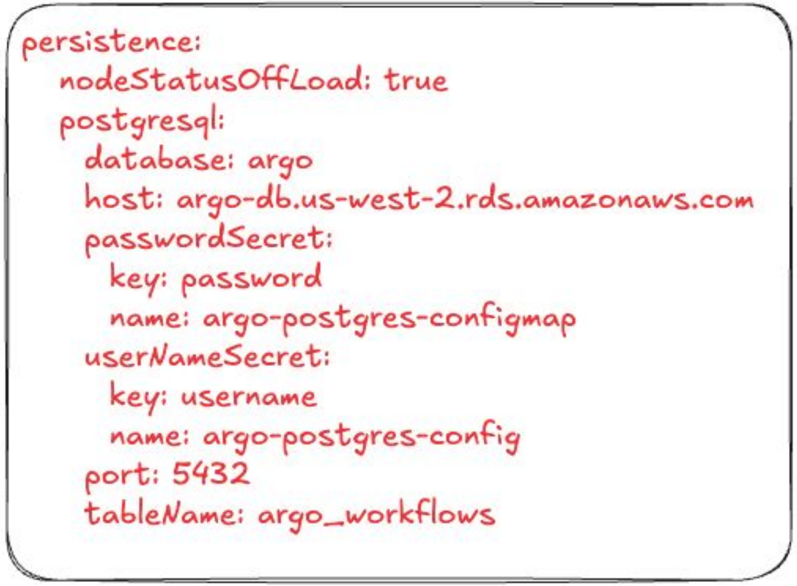


NAME	helloworldparameterflow-mql85.onExit
ID	helloworldparameterflow-mql85-1910618923
POD NAME	helloworldparameterflow-mql85-capture-error-hook-fn-1910618923
HOST NODE NAME	
TYPE	Pod
PHASE	✕ Error
MESSAGE	Request entity too large: limit is 3145728
START TIME	7/29/2024, 9:13:55 PM (2m20s ago)
END TIME	7/29/2024, 9:13:56 PM (2m19s ago)
DURATION	1s
PROGRESS	0/1
MEMOIZATION	N/A

Breaking the 1.5MB barrier

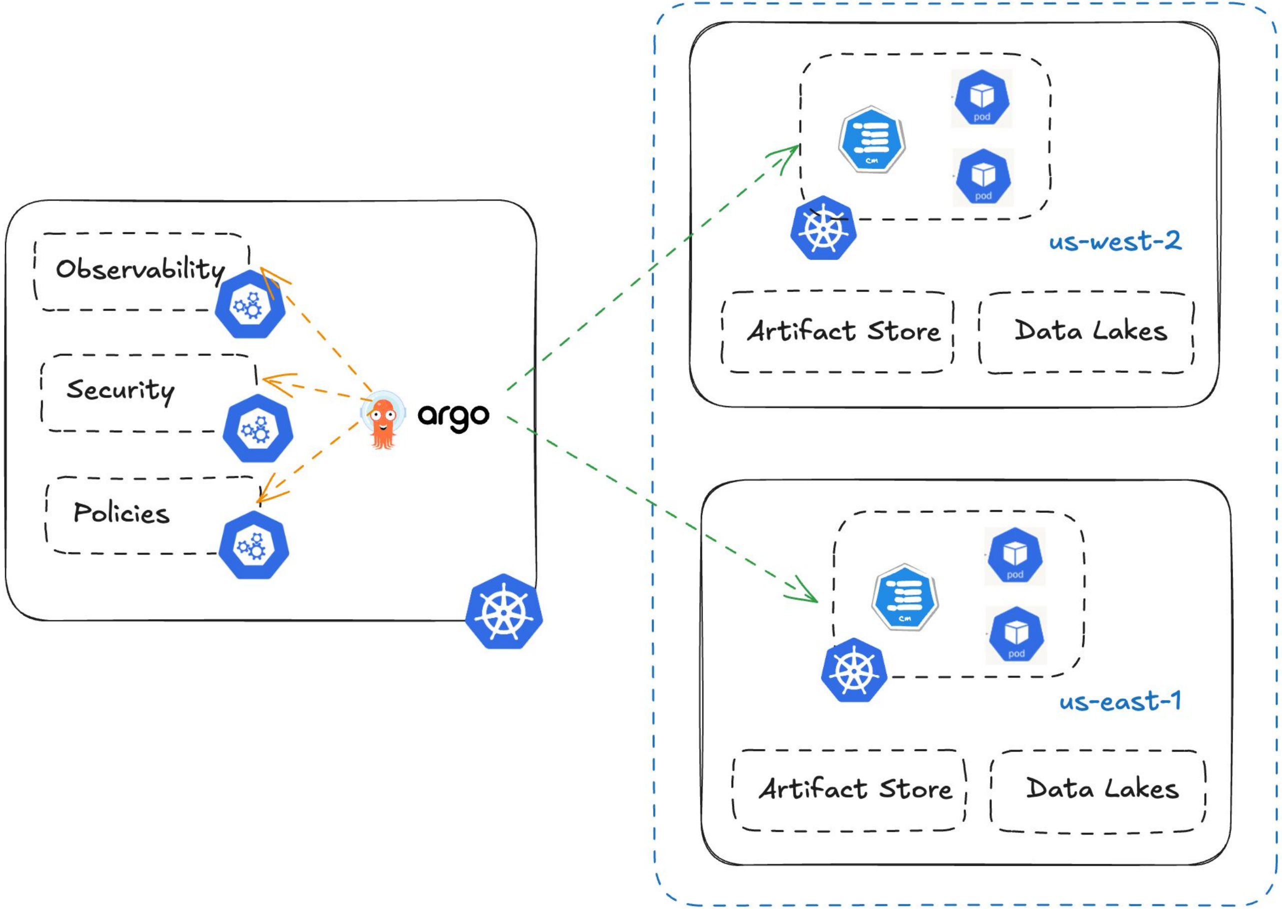
- Argo supports node status offloading via persistence to MySQL/Postgres
- Configure a configmap with a username/password to the DB
- Wire the configmap to the argo workflow controller via

```
--configmap argo-workflows-controller-configmap
```



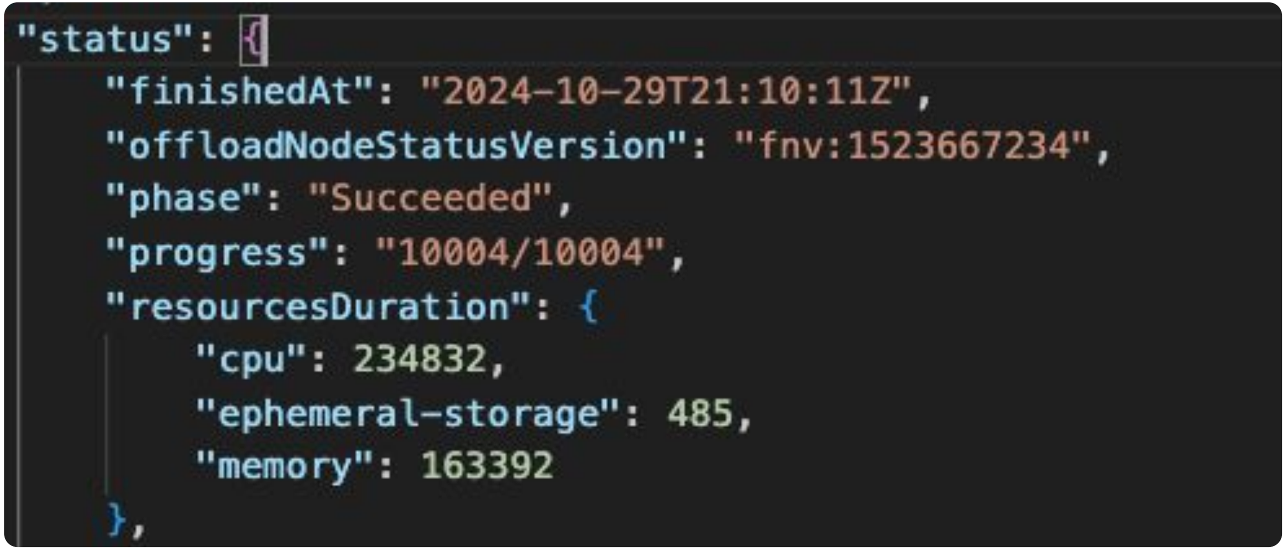
```
persistence:
  nodeStatusOffLoad: true
  postgresql:
    database: argo
    host: argo-db.us-west-2.rds.amazonaws.com
    passwordSecret:
      key: password
      name: argo-postgres-configmap
    userNameSecret:
      key: username
      name: argo-postgres-config
    port: 5432
    tableName: argo_workflows
```

Breaking the 1.5MB barrier



Breaking the 1.5MB barrier

- The Argo Integration on the Outerbounds platform, can now execute DAG’s that are as wide as 10K nodes
- Outerbounds platform scales seamlessly with your elastic cloud provider (x-cloud/x-regions)



```
"status": {
    "finishedAt": "2024-10-29T21:10:11Z",
    "offloadNodeStatusVersion": "fnv:1523667234",
    "phase": "Succeeded",
    "progress": "10004/10004",
    "resourcesDuration": {
        "cpu": 234832,
        "ephemeral-storage": 485,
        "memory": 163392
    },
```

Future work!!

- Supporting IAM based Auth instead of username/password auth in the Outerbounds platform for security conscious customers
- More options to configure large workload scheduling/resource sharing in the Outerbounds platform

& so much more...

Come talk to us @R41 or at any of the below social events

KubeCon

AI After Hours at Quarters

Nov 12, 6.00 - 9.00pm

Quarters Arcade Bar - Downtown
Salt Lake City, Utah

KubeCon

The Future of AI: SLC Edition

Nov 13, 5.30 - 9.00pm

Squatters Pub Brewery
Salt Lake City, Utah

Thank you!!!

Any Questions?