How Asurion Scaled and Simplified Airflow for Workload Orchestration

Rajesh Gundugollu
Karthik Raparthi
asurion is a tech care company

We eliminate the fears and frustrations associated with technology, to ensure our 286 million customers get the most out of their devices, appliances and connections.

WE OFFER:
- Protection & insurance plans
- Repair & replacement services
- 24/7 live tech help
- Setup & installation

FOR:
- Phones
- Connected home tech
- Major appliances
Our purpose

We care for people and their tech — whenever, wherever, however.

PEOPLE
We support close to 300 million people all over the world

TECH
We care for thousands of tech devices, including your favorite phones, laptops, tablets, TVs and gaming consoles
We operate on a global scale

- Founded in 1994
- 11 Call centers
- 17 Countries
- 19,000 Employees
- 29 Offices
- 700+ Retail stores
- 121 Offices
- 17.8M Asurion customer reviews
- Nearly 286M Customer relationships
- 4.5 Stars out of five
- Clients including the top carriers in US, Japan, Latin America and Korea

Supply chain operations across the globe

1 based on 17.8M Asurion customer reviews
How we help our customers:

10,000+ experts

We’ve created the largest tech care network:

- **Virtual**
  Call or chat with us 24/7

- **In-Store**
  700+ stores nationwide

- **On the road**
  We make house calls for 88% of US Population
We have built long-term partnerships
Why do we need a robust Workload Orchestration solution?

A Data Driven Future

Workload Orchestration is at the heart of a Data Platform Founded on Data.

Gravity of the Problem

1000s of Job runs that process 100s of Billions of rows per day with 1,000s of complex dependencies.
What’s our Solution?

Orca

Its Asurion’s flavor of Apache Airflow that’s at the heart of our Data Platform.

With unparalleled reliability, accuracy, stability, and resiliency, Orca is the linchpin of our data-driven future—a future that embodies our vision.

# DAGs
1000s

# Job Runs
Millions

Availability
99.5%
How was the solution approached? Principles

**Founding Principles**

Separation of Concerns
Metadata First

**Evolved Principles**

Universal Operator
Shared nothing Infrastructure
Metadata Rich
Automation
How was the solution approached? **Founding Principles**

**Separation of Concerns**
- **Airflow**
  - Orchestration
  - Data Integration

**Metadata First**
- **Airflow**
  - Orchestration Metadata
  - Data Integration Metadata
  - Job(s)
    - Configuration
    - Metrics
  - DAG(s)
    - Configuration
    - Metrics
  - Job to Job Relations
  - DAG to DAG Relations
### Stress Testing over 250 Days

<table>
<thead>
<tr>
<th># DAGs</th>
<th>Peak Concurrency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>700</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th># Job Runs</th>
<th>Job Runs per Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.2M</td>
<td>22K</td>
</tr>
</tbody>
</table>

### Results

- **Sizing of Worker nodes/concurrency**
- **Auto scaling metrics**

**How was the solution approached?** Experiment
How was the solution approached? **Evolved Principles**

**Single Operator**
- Airflow
- Universal Operator
- Container
- Data Integration Code

**Shared Nothing Infrastructure**
- Compute
- Compute
- Compute
- Compute
- Scheduler
- WebServer
- Queues
- Workers
- Storage
- Storage
- Storage
- Storage

“Simple is Powerful”
How was the solution approached? Architecture

Orca 3.0

RDS – Postgres (Orca DB, Airflow meta DB)

AWS S3 (logs)

Webserver service

Airflow binaries, Logs

DAGFileProcessor

DAGs, Plugins

WebServer Node

Users

ASG: Scheduler

DAGs, Plugins

Airflow binaries, Logs

Scheduler service

Scheduler 1 Node

Scheduler 2 Node

ASG 1: Queue 1

DAGs, Plugins

Airflow binaries, Logs

Celery executor

Worker Node 11

Worker Node 12

Scheduler service

Code sync process (DAGs, Plugins)

Broker (RabbitMQ) Node

DAGs, Plugins

Airflow binaries, Logs

Broker service

Flower service

Broker service

1

2

3

4

5

6

Scheduler

Broker

Webserver

Code sync

Worker

ECS Fargate

DynamoDB

Lambda

SQS
Resiliency?

Metadata Rich

- Comprehensive Metadata
- Real-time Metadata Events

Automation

- Code Generation
- Deployments
- Monitoring and Recovery
- Incident Management
User Accelerators

1- Metadata Validations
2- Metadata Registration

3- Intelligent Orchestration code embeds
4- DAG File creation
5- Event Generation

DAG Deploy

Search for Dag: [Search for Dag...]

dag: [dag]

Tue, 11 Apr 2023 18:27:05 UTC: [Tue, 11 Apr 2023 18:27:05 UTC]

Gcf: [Gcf]
Download: [Download]
Clear: [Clear]

Publish: [Publish]
Visualise: [Visualise]
User Accelerators

DAG Edit

Range of Utilities, Command Center Operations to use while operating Airflow or Orca Menu.

- Edit Load Pointers
- Disable Tasks
- Edit Metadata
- Update Parameters
- Metadata Profile
- Disable Pods
- Disable Dag Notification
- Download Dag Image
- Delete Dag
- Sync Dag Metadata
- Create Dag file
# User Accelerators

## Queue Information

<table>
<thead>
<tr>
<th>Cluster</th>
<th>Running</th>
<th>Waiting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Queue1</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Queue2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Queue3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Queue4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Queue5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Queue7</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Queue8</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Queue9</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Updated: 2023-04-18 04:11:50 UTC
The Team

Raparthi, Karthik
Software Engineer 4

Grange, Arpita
Principal Project Manager

Butler, David Aaron
Software Engineer 2

Antony, Suresh
Director

Javvaji, Srinivas
Principal DevOps Engineer

Jeremy Dykes
DevOps Engineer 4

Sanjoy Das
Snr Software Engineer
Summary: Principles and Lessons Learned

- Separation of Concerns
- Metadata First
- Universal Operator
- Shared Nothing Infrastructure
- Metadata Rich
- Automation
Questions?

Rajesh.Gundugollu@Asurion.com
Karthik.Raparthi@Asurion.com