

Monitoring Airflow using Prometheus

- Spinup **Statsd** Docker container, with following mappings volume bind
vi /opt/statsd/statsd_mapping.yml (in host VM)

```
---
mappings:
  # Map dot separated stats to labels
  - match: airflow.dagrun.dependency-check.*.*
    name: "airflow_dagrun_dependency_check"
    labels:
      dag_id: "$1"

  - match: airflow.operator_successes_(.*)
    match_type: regex
    name: "airflow_operator_successes"
    labels:
      operator: "$1"

  - match: airflow.operator_failures_(.*)
    match_type: regex
    name: "airflow_operator_failures"
    labels:
      operator: "$1"

  - match: airflow.scheduler_heartbeat
    match_type: regex
    name: "airflow_scheduler_heartbeat"
    labels:
      type: counter

  - match: airflow.dag.*.*.duration
    name: "airflow_task_duration"
    labels:
      dag_id: "$1"
      task_id: "$2"

  - match: airflow.dagrun.duration.success.*
    name: "airflow_dagrun_duration"
    labels:
      dag_id: "$1"

  - match: airflow.dagrun.duration.failed.*
    name: "airflow_dagrun_failed"
    labels:
      dag_id: "$1"

  - match: airflow.dagrun.schedule_delay.*
    name: "airflow_dagrun_schedule_delay"
    labels:
```

```

    dag_id: "$1"

- match: airflow.dag_processing.last_runtime.*
  name: "airflow_dag_processing_last_runtime"
  labels:
    dag_file: "$1"

- match: airflow.dag_processing.last_run.seconds_ago.*
  name: "airflow_dag_processing_last_run_seconds_ago"
  labels:
    dag_file: "$1"

- match: airflow.pool.open_slots.*
  name: "airflow_pool_open_slots"
  labels:
    pool: "$1"

- match: airflow.pool.used_slots.*
  name: "airflow_pool_used_slots"
  labels:
    pool: "$1"

- match: airflow.pool.starving_tasks.*
  name: "airflow_pool_starving_tasks"
  labels:
    pool: "$1"

- match: airflow.zombies_killed
  name: "airflow_zombies_killed"
  labels: {}

- match: airflow.executor.running_tasks
  name: "airflow_executor_running_tasks"
  labels: {}

- match: airflow.executor.queued_tasks
  name: "airflow_executor_queued_tasks"
  labels: {}

```

Running statsd docker container

```

sudo docker run -d --name statsd-exporter -p 9102:9102 -p 9125:9125 -p 9125:9125/udp -v /opt/statsd
/statsd_mapping.yml:/tmp/statsd_mapping.yml prom/statsd-exporter --statsd.mapping-config=/tmp
/statsd_mapping.yml

```

- Install stasd inside pythonvirtual env
source to python3 virtual env

```

pip install statsd

```

- Enabling **Statsd** metrics in Airflow

Enable the statsd metrics by setting `statsd_on = True` in configuration file `airflow.cfg`. Before saving the changes, the statsd configuration should look like as follows:

```
statsd_on = True
statsd_host = 10.0.0.19
statsd_port = 9125
statsd_prefix = airflow
statsd_allow_list = scheduler,executor,dagrun
```

```
[metrics]
# StatsD (https://github.com/etsy/statsd) integration settings.
# Enables sending metrics to StatsD.
statsd_on = True
statsd_host = 10.0.0.19
statsd_port = 9125
statsd_prefix = airflow

# If you want to avoid sending all the available metrics to StatsD,
# you can configure an allow list of prefixes (comma separated) to send only the metrics that
# start with the elements of the list (e.g: "scheduler,executor,dagrun")
statsd_allow_list = scheduler,executor,dagrun

# A function that validate the statsd stat name, apply changes to the stat name if necessary and return
```

Restart Airflow webserver and scheduler services

- Login to Prometheus VM, and add **statsd** scraping point in `prometheus.yml`

```
- job_name: airflow_100019
  static_configs:
    - targets: ['10.0.0.19:9102']
```

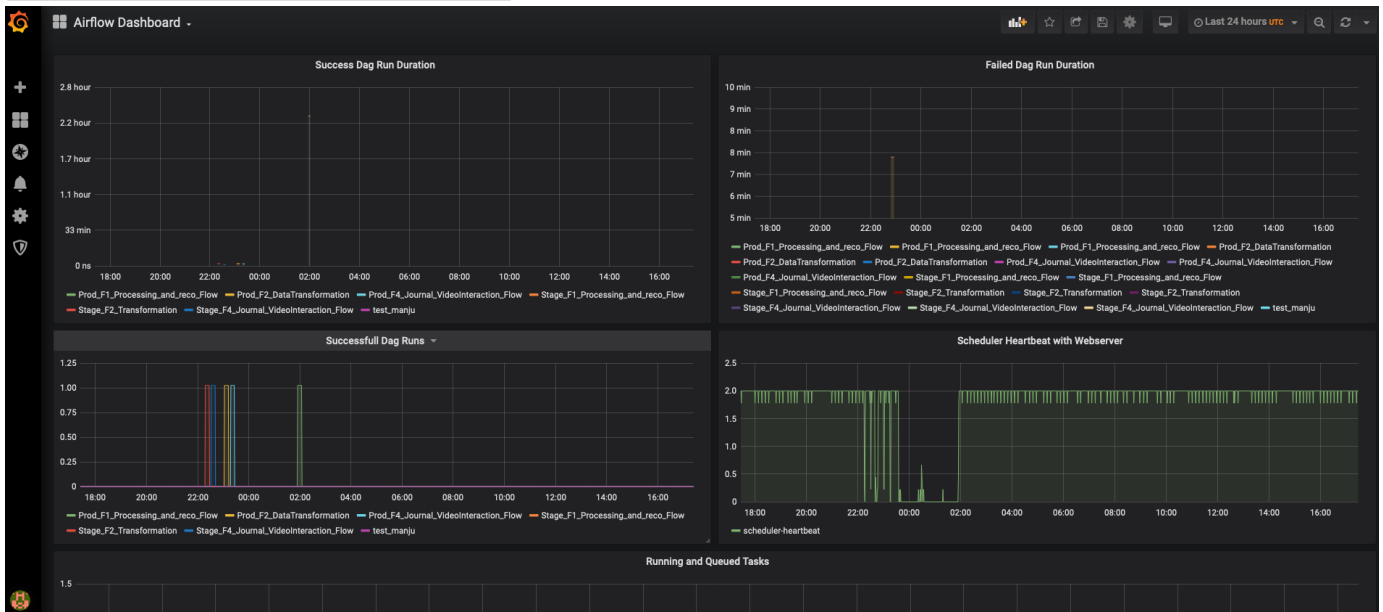
```
static_configs:
- targets: ['10.0.0.27:40023']
```

```
- job_name: airflow_100019
  static_configs:
    - targets: ['10.0.0.19:9102']
```

- Displaying metrics in Grafana
Upload the attached JSON file



Airflow Dashboa...2395684089.json



- Add Alert rules upon **dag_id** run status

```

groups:
- name: airflow.rules
  rules:
    - alert: Prod WARNING Airflow Job did not succeed or started yet
      expr: increase(airflow_dagrun_duration_count{dag_id=~"Prod.*"}
[26h]) * 10 < 8
      for: 1h
      labels:
        severity: "warning"
      annotations:
        summary: " {{ $labels.dag_id }} job has not run at least from
past 24 hours"
        description: "Grafana URL http://10.0.0.21:3000/d/qBWJRGPug
/airflow-dag-dashboard-1?orgId=1&from=now-24h&to=now "

    - alert: Stage WARNING Airflow Job did not succeed or started yet
      expr: increase(airflow_dagrun_duration_count{dag_id=~"Stage.*"}
[26h]) * 10 < 8
      for: 1h
      labels:
        severity: "warning"
      annotations:
        summary: " {{ $labels.dag_id }} job has not run at least from
past 24 hours"
        description: "Grafana URL http://10.0.0.21:3000/d/qBWJRGPug
/airflow-dag-dashboard-1?orgId=1&from=now-24h&to=now "

```