Usability and Debuggability
Airflow at Pinterest

Ace Haidrey, Dinghang Yu, Yulei Li
Workflow Platform

2022/05/25
Agenda

- Workflow Discoverability and Searchability
- Workflow Debuggability
- Workflow Profiling
Pinterest - a one-stop shop for inspiration, new products and new possibilities.
Scale

- **1.3K+** Active Users
- **14K+** Daily DAG Runs
- **131K+** Daily Task Instances
- **160K+** Daily UI/API Requests
Discovering and Searching
Recently Visited Dags

Where is my workflow? I can’t find it, and I just had it open!
Is there a way I can save a subset of dags to display first?
Is there a filter I can setup for my team to access dags we care for?
Cross Cluster Search

There are 3 clusters. How do I find the dag I need to view without checking each?
Tour Guides

Spinner has all these cool features, but how to know when to use and what to use?
Tour Guides Demo
Debuggability
Unified Testing

How can I test changes end to end?

- Single script to bring up a local testing container
Managed Testing Envs

How can I test experimental changes in prod env without affecting product dags?

- A/B testing env
Managed Testing Envs

How can I test in development FGAC dags?

- FGAC testing env

support indexing in development FGAC dags to run in kubernetes Pods with limited access to ensure data access separation for dev and prod
Next Deployment Time

When will my changes be deployed?

2022-05-24, 16:28:16 UTC

Deployment Schedule

Next estimated deployment time:
- spinner-workflows: in 28 mins
- pinboard (migrated): in 2 mins
- pinboard (native): in 1 min

Last deployment finish time:
- spinner-workflows: 15 mins ago
- pinboard (migrated): 18 hrs 20 mins ago
- pinboard (native): 22 hrs 24 mins ago
Non Scheduled Reason

My dag did not have an error message, and it still didn’t have a run. What can be happening?
I see the logs but it is too verbose. Is there an easier way to identify the error messages?
Debuggability features demoed in Airflow Summit 2021

- Dag Parsing Errors
- Dag Critical Path
- Build Information table
Dag Parsing Errors

This dag was supposed to have a run today but it didn’t, anything went wrong?

I see an error message saying my dag is broken, what happened?
Were changes deployed at the time my task ran?

Task Instance: spark_sql_test_task on 2022-05-23T18:30:00+00:00

Log by attempts

Try 1 Execution Stages

Execution pipeline shows the status and time taken for each stage of your task. You can hover over the circles to see stage details and failure root cause analysis.

Artifacts information

Workflows have a lot of moving parts, and this table is to inform you what the commit sha for each major component was at the time of your execution. This will help you understand if any change made were in at the time of the run or not.

<table>
<thead>
<tr>
<th>Artifact</th>
<th>Build Link</th>
<th>Deploy Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pinarrow Build</td>
<td>f88116bdec02bce497d10df9e66a09aa399f5f6e</td>
<td>Lupa Link</td>
</tr>
<tr>
<td>Pinboard Build</td>
<td>f027eab8cd24b6a7554a2a27efab4c294829aba5d</td>
<td>Lupa Link</td>
</tr>
<tr>
<td>Spinner Workflows</td>
<td>b8c24d15473f5d54950441a29ea36b9f27b1d9</td>
<td>Lupa Link</td>
</tr>
</tbody>
</table>
Critical Path Finder

This dag run took 5 hours, why is it so slow?
Workflow Profiling
Each task is executed in a dedicated k8s pod

Provide default resource allocation and require users to manually override

- No one-size-fit-all solution
- Cause resource underutilization or issues like OOM KILL

How much memory should I allocate to my task?

My task failed with OOM error, how do I fix it?

How much memory should I allocate to my task?

My task failed with OOM error, how do I fix it?
Workflow Cost Display

How much computing resources do each of my workflow run use?

- Collect resource usage data from yarn cluster
- Offline pipeline to convert the resource usage data into $

Choose to view cost by vcore, memory or the two combined

- A measurable metric to take actions on
- Bring awareness to users
- Target wasted resources
- Drive workflow optimization
Workflow Cost Demo
I have a few workflow jobs suddenly start to fail today, is there something wrong in Spinner?
## Incident Management Admin Tooling

### All Incidents

<table>
<thead>
<tr>
<th>Id</th>
<th>Status</th>
<th>Reporter</th>
<th>Name</th>
<th>Description</th>
<th>Jira</th>
<th>Severity</th>
<th>Allow Auto Reply</th>
<th>Created At</th>
<th>Updated At</th>
</tr>
</thead>
<tbody>
<tr>
<td>377</td>
<td>closed</td>
<td>eucacheen</td>
<td>root</td>
<td>Incident: Task failed with GOM error (test code 9) in Spinnier. We have just bumped the default memory settings for Spinnier native tasks, you can RETRY your previously failed tasks and the new memory settings should be sufficient.</td>
<td>IM-864</td>
<td>1</td>
<td>Yes</td>
<td>2021-05-06</td>
<td>2021-05-06</td>
</tr>
<tr>
<td>376</td>
<td>closed</td>
<td>eucacheen</td>
<td>root</td>
<td>Changes made to Eucalypsee script in effect 81.</td>
<td>1</td>
<td>No</td>
<td></td>
<td>2021-05-05</td>
<td>2021-05-05</td>
</tr>
<tr>
<td>375</td>
<td>closed</td>
<td>eucacheen</td>
<td>root</td>
<td>Incident to be closed in Spinnier</td>
<td>1</td>
<td>No</td>
<td></td>
<td>2021-05-05</td>
<td>2021-05-05</td>
</tr>
<tr>
<td>311</td>
<td>closed</td>
<td>eucacheen</td>
<td>root</td>
<td>Spinnier web is down due to endpoint registration in canary</td>
<td>IM-8253</td>
<td>3</td>
<td>No</td>
<td>2021-04-26</td>
<td>2021-05-04</td>
</tr>
<tr>
<td>309</td>
<td>closed</td>
<td>eucacheen</td>
<td>root</td>
<td>[DHT] Service discovery for MySQL; problem blocked spin scheduling</td>
<td>IM-8118</td>
<td>1</td>
<td>No</td>
<td>2021-04-22</td>
<td>2021-05-05</td>
</tr>
</tbody>
</table>

*Note: The table shows a list of incidents with details such as status, reporter, name, description, and timestamps for creation and update.*
Incident Management Smart Retries

Is there a way for retrying all failed jobs if the platform had an issue?

<table>
<thead>
<tr>
<th>Allow Auto Retry *</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow failed tasks to retry or not, 0 means disallow, 1 means allow</td>
<td></td>
</tr>
</tbody>
</table>
Triggering Dag Clarification

I try to trigger my dag manually but the conf doesn’t work right?

Click here to learn how to use the Configuration JSON option.

If you want to pass extra configurations to the DAG run, you can put the configuration in the form. Please note the type of your configuration must be a dictionary, and other formats are disallowed. You can take look at the below examples or read on the open source doc to better understand the usage.

A valid configuration example:

```json
{
  "input_key": "input_value"
}
```

Both the key and the value are enclosed in double quotes (" ") and everything is wrapped in curly braces ({}).

An invalid configuration example:

```json
"input_key": "input_value"
```

How can I access this configuration in my DAG?

Configuration JSON:

```json
{
  "input_key": "input_value"
}
```

Using input_key in the code:

```python
from airflow import DAG
from airflow import operators
from airflow import BashOperator
from airflow import dates

dag = DAG("example_parameterized_dag", schedule_interval=ScheduleInterval.None, start_date=days_ago(2))

parameterized_task = BashOperator(
    task_id='parameterized_task',
    bash_command="echo value: {{ dag_run.conf["input_key"] }}", # => using the given "input_key"
    dag=dag,
)
```

Note

The parameters from ```{ { dag_run.conf } } ``` can only be used in a template field of an operator.
Wrap Up
Identifying What Users Want

User Surveys

- Workflow testing
- Backfill
- Building new workflows
- System performance (e.g., response time, system load)
- Role/Access Control
- Debugging/Troubleshooting
- Resource management (e.g., I/O, memory, CPU)
- Build/deployment
- Better ResourceSpec support
- Spinner UI constructs
- Glitch when there are new changes
- Blocking Review is kind of broken
- support presto queries in...
- Documentation more easily
- ML/Ops, native integration
- PII spinner
- may be using terraform to...

Request Forms

- Sending alerts when a dagrun or workflow can’t finish within its SLO
- Support for retrying a job with new parameters
- Paste ability for a comma separated task list into the backfill form
- Add an Option to Turn off Alerts for backfill runs
- Ensure compatible artifacts are used in jobs that depend on multiple deployment pipelines
- Workflow team will create JIRA tickets for the requests more upvotes! (>=5)
- Differentiate ‘Success’ with ‘Marked as Success’
- Support send_dag_level_fail_email param similar to send_dag_level_failure_message ...
- Frequent pinboard deploys to both migrated & native environments
- Differentiate the PROD Spinner UI and devapp Spinner UI
- support multi-partition sensor
- Add a [DAG Final] node in Graph View
- add Callback hook for native spinner when job is marked failed/success
- Run access control scripts automatically for spinner users
Wrap Up

● Our goal is to make the Spinner workflow system more usable
  ○ Improve the workflow discoverability for our multi-environment setup
  ○ Streamline the debugging experience
  ○ Profile workflow and surface data on UI
  ○ ...

● Meanwhile, free the system admins from heavy & repetitive user support routine
  ○ Drop in our user support load

![Number of user questions graph from Jul 2021 to Mar 2022]
Future Work

● Improvements to the cost report to include other infra costs
● Search by the task_id, not just the dag_id
● Features to further reduce noise and identify errors in the logs
  ○ Improvement in log filtering tooling
  ○ Link source code in error messages
● Upgrade Airflow Version to 2.0+
● Blog post regarding UI feature development
● And More!
Additional Resources

● Airflow Summit 2021:
  ○ [Pinterest’s Migration Journey](#)
  ○ [Usability Improvements for Debugging](#)

● Airflow Summit 2020:
  ○ [Airflow as the next gen of workflow system at Pinterest](#)

● Tech Blog
  ○ [Spinner: Pinterest’s Workflow Platform](#)
  ○ [Spinner: The Mass Migration to Pinterest’s New Workflow Platform](#)
Thank you!

The Faces Behind The Work

Ace Haidrey
Software Engineer

Ashim Shrestha
Site Reliability Engineer

Apoorv Gaur
Software Engineer

Dinghang Yu
Software Engineer

Evan Li
Engineering Manager

Hannah Chen
Technical Program Manager

Hilda Johny
Software Engineer

Kuo-Kai Hsieh
Software Engineer

Numan Aral
Software Engineer

Yulei Li
Software Engineer