Pinball: Workflow System of Pinterest

Shuo Xiang / Jack Ye
Pinterest: The World’s Catalog of Ideas
Data and Workflow in Pinterest
Scale@Pinterest

Run on the Cloud (AWS)
Business Scale.
• 200M+ MAUs
• Hundred billions of Pins
• Billions of Boards
Data Scale
• 200+ PB @ S3
• 6000+ Hive/Hadoop nodes
• 400+ Presto nodes
Data@Pinterest

Workflow System (Pinball/Pinflow/Skyline)

Streaming System

Batch System

AdHoc System

Ingestion

Processing

Data Product

Web Fleet

Kafka

Merced

Tracker

WaterMill

MySQL

Singer
Workflow System

• Workload
  – 5000+ workflows
  – 30,000+ jobs

• Computation types
  – Hadoop, Hive, Spark ...
  – Cascading, Scalding ...
  – General script
Workflow use

- External/internal metrics reporting
- ETL for data warehouse
- Search, recommendation
- ML pipelines
Pinball: Pinterest’s Workflow System
Notations

workflow

dependency

job

Job 0

Job 1
Job 2
Job 3
Job 4

Job 5
Job 6
Job 7
Job 8

Job 9
Framework

- Composer to define workflows
- Scalable scheduler
- Multi-platform support
  - Local,
  - Hadoop/Yarn,
  - Presto
  - Mesos, K8s etc.
- Operate, report, monitor and alert
User Features

- Rich job templates
  - Hadoop/Hive/Spark/Local job template
  - Condition and waiter job
- Workflow overrun policy
- Job execution record
  - Automatic retry & backfill
  - Manual mark/erase
  - x-workflow and x-cluster dependency
Workflow Overrun

Decisions for multiple workflow instances

- skip
- delay
- abort
- start_new

<table>
<thead>
<tr>
<th>Time</th>
<th>Status</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1535599570053</td>
<td>RUNNING</td>
<td>20 hrs 1 min 24 secs</td>
</tr>
<tr>
<td>1535513177018</td>
<td>SUCCESS</td>
<td>20 hrs 13 mins 1 sec</td>
</tr>
<tr>
<td>1535426758978</td>
<td>SUCCESS</td>
<td>18 hrs 18 mins 9 secs</td>
</tr>
<tr>
<td>1535340351179</td>
<td>SUCCESS</td>
<td>19 hrs 2 mins 32 secs</td>
</tr>
<tr>
<td>1535253984604</td>
<td>ABORTED</td>
<td>1 day 8 secs</td>
</tr>
<tr>
<td>1535167542572</td>
<td>SUCCESS</td>
<td>19 hrs 37 mins 41 secs</td>
</tr>
</tbody>
</table>
Job Record

- External storage component
- Automatic retry & backfill
- Manual mark/erase
- X-workflow and X-cluster dependency
Design Choices

- Composer scheduler isolation
- Pluggable job/task
- Extensible
Architecture

Workflow Repository

Config Parser

UI, API, CLI

Master

Scheduler

Worker1

Worker2

WorkerN

Compute Infrastructure
Pinball Scheduler

Multi-composer

Parser

Config

Job

Hive	Hadoop	Python

Executor

Yarn	Local

JobTemplate

config.yaml

parser: parser params:
class ExamplePythonJob(data_job.PythonJob):
    _OWNERS = 'ldap_user_a'

    def _execute(self):
        print('this is a PythonJob')
class ExampleHiveJob(HiveJob):
    # True for date range, False for individual dates
    _MULTIDATE_EXECUTE = False

def _setup(self):
    super(data_job.HiveJob, self)._setup()
    self.params['k1'] = 'v1'
    self.params['k2'] = 'v2'

_QUERY_TEMPLATE = """SELECT ... %(k1)s %(k2)s"""
class ExampleHadoopJob(data_job.HadoopJob):
    def __init__(self):
        self._output_path = 's3://path/to/output/data'
        self.jobconf_args['mapred.job.name'] = str(self._get_class_name())
        self.jobconf_args['mapred.reduce.tasks'] = 200
        self.jobconf_args['cascading.combine_input_max_size'] = 3 * 1024 ** 3
        self.jobconf_args['mapred.output.compress'] = 'true'
        super(ExampleHadoopJob, self)._setup()

    def _get_class_name(self):
        return "com.pinterest.hadoop.businessA.JobDefA"

    def _execute(self):
        s3_utils.rm(self._output_path)
        super(ExampleHadoopJob, self)._execute()

    def _complete(self):
        # some post execution logic
EXAMPLE_WORKFLOW = {
    LocalConditionTemplate('ExamplePythonJob'): [],
    HadoopJobTemplate(
        name='ExampleHiveJob',
        job_full_class='pinterest.example.ExampleHiveJob'
    ): ['ExamplePythonJob'],
    HadoopJobTemplate(
        name='ExampleHadoopJob',
        job_full_class='pinterest.example.ExampleHadoopJob'
    ): ['ExamplePythonJob', 'ExampleHiveJob'],
}
**Example Composer #2**

### Workflow Composer

- **Workflow Name**: AdEngMetricsWorkflow
- **Job/Condition Name**: WaitForAdEngMetricsTable
- **Job/Condition Type**: Hive_query_condition
- **Owners**: edimarc@pinterest.com

```
1  set mapred.min.split.size.per.node=102400000000;
2  set mapred.min.split.size.per.node=1024000000000;
3  set mapred.max.split.size=1024000000000;
4  SELECT true FROM ad_ad_eng_metrics
5  WHERE the_dt='end_date' LIMIT 1.
```
Pinball Master

Service for

• composer-independent tokens
• workflow and job status
• scheduling status
• user interactions
Pinball Worker

• Claims runnable jobs from master
• Applies lease on job for fault-tolerance
• Runs general job submission command
• Scales up by increasing parallelism
Summary

- Flexible workflow system supports multiple composers and platforms
- Distributed master-worker design
- Rich user features and extensible
Upcoming features

• Enhanced composer
  – Release process (test/canary/prod)
  – Resource abstraction
  – Dynamic versionized workflow
• Improved execution isolation
We are hiring!

Jack Ye  wye@pinterest.com
Shuo Xiang  sxiang@pinterest.com