Sell Cron, Buy Airflow!
Modern Data Pipelines in Finance

James Meickle
Site Reliability Engineer, Quantopian

Velocity New York, 10/02/2018
About James

- Site Reliability Engineer, Quantopian
- Site Reliability Engineer, Harvard Center for Brain Science
- Sales Engineer, AppNeta
- Developer, Romney for President 2012
- Formerly academia & public policy

@jmeickle

Is this a data pipeline?

cron
DISCLAIMER 1: I am an SRE professional at a financial institution, not a finance professional!

@jmeickle
DISCLAIMER 2: Opinions expressed in this presentation are my own, not Quantopian's!
DISCLAIMER 3: This presentation is not investment advice!
SEC guidelines are extremely serious.

Elon Musk is being sued by the SEC over 'Tesla 420' tweet

Blunt accusations could see Musk's career go to pot
PART 1:
Data pipelines and data problems
Become an Expert in Quant Finance

Quantopian provides free education, data, and tools so anyone can pursue quantitative finance. Select members license their algorithms and share in the profits.

https://www.quantopian.com/about  @quantopian

Community Achievements

All numbers are as of June 1, 2018

$52.9MM  Peak allocation level to a single algorithm

$155MM  Cumulative sum of capital allocated to licensed algorithms
SRE Team

- AWS Infra
- Kubernetes
- Architecture
- Monitoring
- Logging
- Build
- Deploy
Black Team

- Data infrastructure
- Investment automation
- Productizing data science
Community data

- Data from commercial data vendors
- Cleaned and reconciled
- Packaged into high-performance formats
- Aware of lookahead bias
- Portfolio risk model
Internal data

- Site analytics
- Calculating contest performance
- Simulating algorithm portfolios
- Experimental datasets and data science
- Risk management
- Allocation decisions
- Portfolio returns
- Compliance reporting
High reliability
Unreliable data
Brittle crontabs
Uncoordinated logic
SRE bottleneck
Of course we wanted to "fix our data pipelines", but...
What were our business goals?
PART 2: Business goals and tool selection
Reliability
Observability
Velocity
Our data problems are unique, just like everyone else's.
Democratization
NOT big data
NOT streaming data
There is no cloud
it's just someone else's computer
<table>
<thead>
<tr>
<th>DATE</th>
<th>HOLIDAY</th>
<th>STOCK MARKET (NYSE STATUS)</th>
<th>BOND MARKET (SIFMA RECOMMENDATION)</th>
<th>US EQUITY FUTURES (CME GLOBEX STATUS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>January 1, 2016</td>
<td>New Years Day</td>
<td>Closed</td>
<td>Closed</td>
<td>Closed</td>
</tr>
<tr>
<td>January 18, 2016</td>
<td>Martin Luther King, Jr. Day</td>
<td>Closed</td>
<td>Closed</td>
<td>Open until 1:00 PM Reopen at 6:00 PM</td>
</tr>
<tr>
<td>February 15, 2016</td>
<td>Washington's Birthday/Presidents Day</td>
<td>Closed</td>
<td>Closed</td>
<td>Open until 1:00 PM Reopen at 6:00 PM</td>
</tr>
<tr>
<td>March 24, 2016</td>
<td>Day before Good Friday</td>
<td>Open</td>
<td>Open until 2:00 PM</td>
<td>Open</td>
</tr>
<tr>
<td>March 25, 2016</td>
<td>Good Friday</td>
<td>Closed</td>
<td>Closed</td>
<td>Closed</td>
</tr>
<tr>
<td>May 27, 2016</td>
<td>Friday before Memorial Day</td>
<td>Open</td>
<td>Open until 2:00 PM</td>
<td>Open</td>
</tr>
<tr>
<td>May 30, 2016</td>
<td>Memorial Day</td>
<td>Closed</td>
<td>Closed</td>
<td>Open until 1:00 PM Reopen at 6:00 PM</td>
</tr>
<tr>
<td>July 1, 2016</td>
<td>Friday before Independence Day</td>
<td>Open</td>
<td>Open until 2:00 PM</td>
<td>Open</td>
</tr>
<tr>
<td>July 4, 2016</td>
<td>Independence Day</td>
<td>Closed</td>
<td>Closed</td>
<td>Open until 1:00 PM Reopen at 6:00 PM</td>
</tr>
</tbody>
</table>
And the winner is...
Apache Airflow 1.10.0 is out ❤️🎉!!

Highlights:
- New RBAC web interface in beta
- First class kubernetes operator
- Experimental kubernetes executor
- Timezone support
- Performance optimizations for large DAGs
- Many GCP and S3 integration improvements
- Tons of Bug Fixes

9:53 AM - 27 Aug 2018

96 Retweets 182 Likes
Self-explaining systems are more observable.
import airflow
PART 3: The one with the code samples
class IsTradingHolidayOperator(ConditionalTradingCalendarOperator):
    
    Returns true (and executes downstream tasks) if the session timestamp includes any regular trading holidays. Otherwise, returns false (and skips downstream tasks). Does not take into account sessions with irregular opens or closes, as those are not regular holidays (even if they are "holidays").
    
def condition(self, **context):
        session_timestamp = self.get_session_timestamp(context)
        holidays = trading_calendar.holidays_at_time(
            self.calendar.regular_holidays,
            session_timestamp,  # Start day
            session_timestamp,  # End day (the same day!)
            session_timestamp,  # Time of day (date part not used)
            'UTC'
        )

        is_trading_holiday = (len(holidays) > 0)

        if is_trading_holiday:
            self.log.info("%s is a trading holiday.", session_timestamp)
            return True
        else:
            self.log.info("%s is not a trading holiday.", session_timestamp)
            return False
# Expected around 9 AM.
wait_until_9 = TimeSensor(
    task_id="wait_until_9",
    # Wait until 9 AM Eastern.
    target_time=east_to_utctime('09:00'),
    retries=3,
)

def is_company_meeting_check(**context):
    ""
    This function will get executed on the worker
    ""
    ex_date = context["next_execution_date"]
    ts = Timestamp(ex_date, tz='UTC')
    if ts.dayofweek == WEDNESDAY:
        return "company_meeting"
    else:
        return "order_mealpal"

is_company_meeting = BranchPythonOperator(
    task_id='is_company_meeting',
    python_callable=is_company_meeting_check,
    provide_context=True
)

company_meeting = SlackAPIPostOperator(
    task_id="company_meeting",
    text="Company meeting! Don't order today."
    **slack_kwargs
)

order_mealpal = SlackAPIPostOperator(
    task_id="order_mealpal",
    text="<!here> It's that time! 🍔: Order here: "
    "https://secure.mealpal.com/lunch",
    **slack_kwargs
)
Mealpal AI  APP  9:02 AM
Company meeting! Don't order today.

trhodes 🐓  9:16 AM

@jmeickle
when 2 airflow??
Airflow Staging (core)  APP  4:13 AM
⚠ nightly_dataload.ingest_factset_fundamentals_v3 [2018-05-14T00:00:00] has retried (1/2 attempts).

View DAG (Graph)  View DAG (Tree)  View Task

Airflow Staging (core)  APP  4:32 AM
❌ nightly_dataload.ingest_factset_fundamentals_v3 [2018-05-14T00:00:00] has failed!

View DAG (Graph)  View DAG (Tree)  View Task

✅ nightly_dataload.ingest_fundamentals_v3 [2018-05-14T00:00:00] is complete.
✅ nightly_dataload.daily_fundamentals [2018-05-14T00:00:00] is complete.
✅ nightly_dataload.queue_risk_metrics_job [2018-05-14T00:00:00] is complete.
✅ nightly_dataload.queue_fund_risk_model_job [2018-05-14T00:00:00] is complete.
✅ nightly_dataload.queue_public_risk_model_job [2018-05-14T00:00:00] is complete.
✅ nightly_dataload.queue_psim_risk_metrics_job [2018-05-14T00:00:00] is complete.

Airflow Staging (core)  APP  4:59 AM
✅ nightly_dataload.daily_fund_risk_model [2018-05-14T00:00:00] is complete.

Airflow Staging (core)  APP  5:44 AM
✅ nightly_dataload.daily_public_risk_model [2018-05-14T00:00:00] is complete.
✅ nightly_dataload.queue_contest [2018-05-14T00:00:00] is complete.

Airflow Staging (core)  APP  6:00 AM
✅ nightly_dataload.backtest_replacer [2018-05-14T00:00:00] is complete.
Summary Metrics

Number of High-Urgency Incidents

253
filemover deploy safety check one-liner:

```bash
$ sudo crontab -l -u filemover |
grep -v '#' | awk '{print $2, $1}' | sort -n
```

0 1
4 30
4 30
4 30
4 30
4 30
4 35
4 5
4 5
4 5
7 5
8 0
8 0
8 10
8 30
8 35
8 5
9 0
9 10
9 5
11 35
13 35
<table>
<thead>
<tr>
<th>Created</th>
<th>Assigned</th>
<th>Mentioned</th>
<th>Review requests</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>3 Open</td>
<td>68 Closed</td>
</tr>
</tbody>
</table>

Pull requests

@jmeickle
I think the startup is move algo of starting at 6am.

Airflow Production (core) APP 5:07 AM
nightly_dataload.queue_contest [2018-07-24T00:00:00] is complete.

trhodes 5:08 AM
oh shit that’s also on airflow? That’s awesome.

^ cleared the dag.

Airflow Production (core) APP 5:12 PM
nightly_dataload.daily_assetdb [2018-09-28T00:00:00+00:00] is complete.
nightly_dataload.daily_adjustmentsdb [2018-09-28T00:00:00+00:00] is complete.
nightly_dataload.daily_minute_bars [2018-09-28T00:00:00+00:00] is complete.

jd 5:12 PM
I freaking love airflow.
Appendix 1: Why Airflow is winning
Apache Airflow (Incubating)  https://airflow.apache.org

- 5,383 commits
- 7 branches
- 95 releases
- 580 contributors
- Apache-2.0
<functional architecture image>
Appendix 2: Future Development
<mock up of DAG simulation>
1. $\mathcal{O} = \{ x \in t_1 | \exists (a, b) \in M \cdot a = x \}$

2. $\mathcal{N} = \{ y \in t_2 | \exists (a, b) \in M \cdot b = y \}$

3. $\mathcal{P} = \{ (a, b) | a \in \mathcal{O}, b \in \mathcal{N} \cdot (p(a), p(b)) \in M \lor (p(a), p(b)) \in \mathcal{P} \}$

4. $\forall (a, b) \in \mathcal{P}$ in traversal order:
   (a) if $\text{likelyRenamed}(a, b) = \text{true}$
      i. $\mathcal{P} = \mathcal{P} \setminus \{ (a, b) \}$, $L = L \cup \{ (a, b) \}$
      ii. $\forall x \in t_1 | (x, b) \in \mathcal{P} \cdot \mathcal{P} = \mathcal{P} \setminus \{ (x, b) \}$, $U = U \cup \{ (x, b) \}$
      iii. $\forall y \in t_2 | (a, y) \in \mathcal{P} \cdot \mathcal{P} = \mathcal{P} \setminus \{ (a, y) \}$, $U = U \cup \{ (a, y) \}$
   (b) if $\text{likelyRenamed}(a, b) = \text{false}$
      i. $\mathcal{P} = \mathcal{P} \setminus \{ (a, b) \}$, $U = U \cup \{ (a, b) \}$

5. Let the user acknowledge all real renamings and insert them to $R$

6. $\forall (a, b) \in R$
   (a) Generate operation $\text{REN}(a, l(b))$
   (b) $M = M \cup \{ (a, b) \}$
Does your team need data engineering training?

Talk to me at the reception!

@jmeickle
Thank you!
This presentation is for informational purposes only and does not constitute an offer to sell, a solicitation to buy, or a recommendation for any security; nor does it constitute an offer to provide investment advisory or other services by Quantopian, Inc. ("Quantopian"). Nothing contained herein constitutes investment advice or offers any opinion with respect to the suitability of any security, and any views expressed herein should not be taken as advice to buy, sell, or hold any security or as an endorsement of any security or company. In preparing the information contained herein, Quantopian, Inc. has not taken into account the investment needs, objectives, and financial circumstances of any particular investor. Any views expressed and data illustrated herein were prepared based upon information, believed to be reliable, available to Quantopian, Inc. at the time of publication. Quantopian makes no guarantees as to their accuracy or completeness. All information is subject to change and may quickly become unreliable for various reasons, including changes in market conditions or economic circumstances.