Building Data Products using Hadoop at Linkedin

Mitul Tiwari
Search, Network, and Analytics (SNA)
LinkedIn

Who am I?

Mitul Tiwari

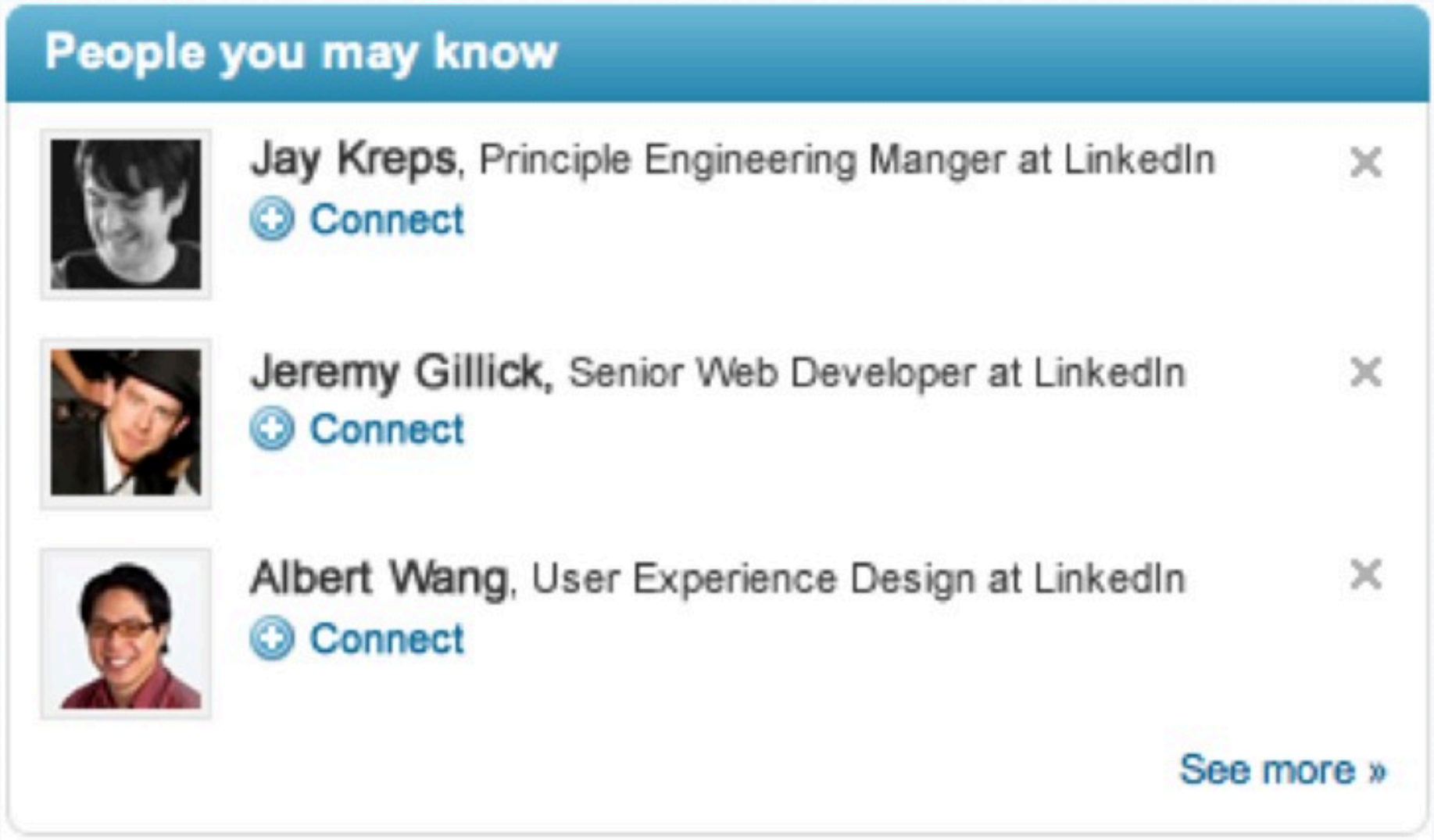
Senior Research Engineer at LinkedIn

San Francisco Bay Area | Computer Software

Current	**Senior Research Engineer** at **LinkedIn**
Past	Lead Member of Technical Staff at **Kosmix**
	Member of Technical Staff at **Kosmix**
	Summer Intern at **Google Inc.**
	see all
Education	The University of Texas at Austin
	The University of Texas at Austin
	Indian Institute of Technology, Bombay
Recommendations	**2** people have recommended Mitul
Connections	**500+** connections
Websites	Personal Website
	Blog
Twitter	mitultiwari
Public Profile	http://www.linkedin.com/in/mitultiwari

What do I mean by Data Products?

People You May Know



4

Profile Stats: WVMP

Profile Stats Pro

Last 90 Days May 10, 2011 – August 8, 2011 Settings

Who's Viewed Your Profile

LAST THREE MONTHS

Graduate Student at Stanford University

Someone in the Executive Leadership function in the Staffing and Recruiting industry from San Francisco Bay Area

Dj Das
Business Operations / Product Management / Big Data Evangelist
San Francisco Bay Area | Information Technology and Services
In Common: 8 shared connections 1 shared group

Software Developer at SAS

Ravikant D V S
Research Assistant at Cornell University
Austin, Texas Area | Computer Software
In Common: 5 shared connections

Doug Judd
CEO at Hypertable, Inc.
San Francisco Bay Area | Internet
In Common: 34 shared connections 2 shared groups

Bruce Ferguson
High Frequency Trader / Quant Strategist
Toronto, Canada Area | Investment Management
In Common: 1 shared connection

Trends

Views | Appearances in Search

Total Views **489**

Top Search Keywords

1.	**hadoop**	**6%**
2.	kosmix	6%
3.	mitul tiwari	3%
4.	data mining	3%
5.	linkedin	2%
6.	mapreduce	2%
7.	computer science	2%
8.	hadoop mapreduce	2%
9.	hadoop microsoft	1%
10.	tweets	1%
11.	ruby	1%
12.	machine learning	1%

Viewers of this profile also ...

Viewers of this profile also viewed...

Subramanyam (Manyam) Mallela
Director of Engineering, Online...

Digvijay Lamba
Chief Architect Categorization at Kosmix

Vijay Chittoor
Founder at Mertado

Anand Rajaraman
Founder, Kosmix; Founding Partner,...

Indrani Chakravarty
Member of Technical Staff at...

Ankur Jain
Director, Product Management at...

Sri Subramaniam
Engineering & Product Executive

Venky Harinarayan
SVP, Global eCommerce at Walmart

Gaurav Bhalotia
Director, at Retrevo

Skills

Search Skills & Expertise

Related Skills

- HBase
- MapReduce
- Nutch
- Solr
- Lucene
- AWS
- Collaborative Filtering
- EC2
- Guice
- Weka
- S3
- Recommender Systems
- MarkLogic
- Amazon Web Services
- Memcached
- RDFS
- HAProxy
- Web Crawling
- Berkeley DB
- CouchDB

Hadoop

▲46% y/y

Primary Industry: Internet

Apache Hadoop is a Java software framework that supports data-intensive distributed applications under a free license. It enables applications to work with thousands of nodes and petabytes of data. Hadoop was inspired by Google's MapReduce and Google File System (GFS) papers. Hadoop is a top-level Apache project, being built and used by a community of ...
More on 'Hadoop' at Wikipedia »

✔ Listed on your profile — Edit Your Skills

Hadoop Professionals

Tom White (2nd)
Engineer at Cloudera
Tom White has been an Apache Hadoop committer since February 2007, and is a member of the Apache Software...

Doug Cutting (2nd)
Me
I am a founder of the Lucene, Nutch, Hadoop and Avro open source projects.

Patrick Hunt (2nd)
Member at The Apache Software Foundation

Arun C Murthy (2nd)
Founder and Architect at Hortonworks Inc.
I am a Founder and Architect at Hortonworks Inc. Hortonworks was formed by the key architects and core Hadoop...

Devaraj Das (2nd)
Founder at HortonWorks
Distributed/Grid Computing. Security technologies (at all layers of the stack - application, tcp, IP, physical...

Jakob Homan (1st)
Senior Software Engineer at LinkedIn
Apache Hadoop Committer and PMC member. Trying to be everywhere at once on LinkedIn's SNA team - Hadoop, Hive,...

Koji Noguchi (2nd)
Hadoop Debugger

Chris Douglas (2nd)
Technical Yahoo!

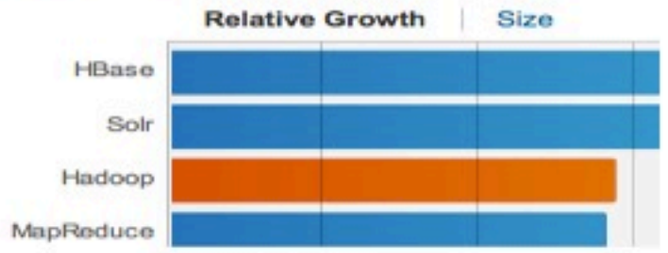
More Hadoop Professionals...

Hadoop Groups

Hadoop Users — You are a member
A group for Hadoop users.

Share Tweet

89th Fastest growing skill



Related Companies

The Apache Software Foundation
Computer Software, United States
Follow

Cloudera
Computer Software, San Francisco Bay Area
Follow

Yahoo!
Internet, San Francisco Bay Area
Follow

LinkedIn
Internet, San Francisco Bay Area
Stop following

eBay
Internet, San Francisco Bay Area
Follow

More companies...

Hadoop Jobs

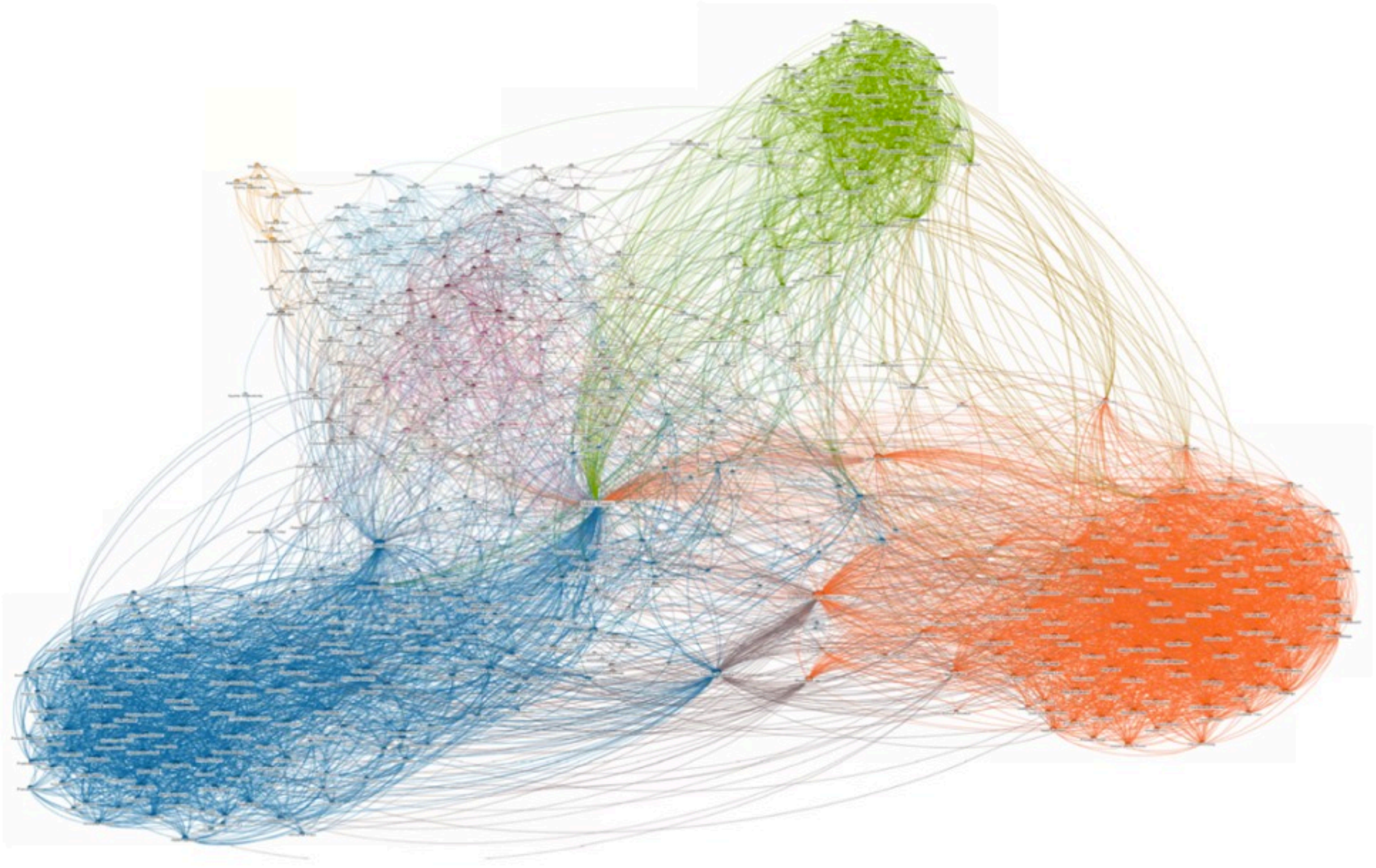
Senior Hadoop Systems Developer
Cloudera, Inc. - Palo Alto or San Francisco CA

Software Engineer – Hadoop Engineering
Greenplum - San Francisco Bay Area

Sr. Software Engineer - Hadoop
Apple Inc. - Cupertino, CA

Software Architect (Java, Hadoop)
NetApp - Sunnyvale CA

InMaps



Data Products: Key Ideas

- Recommendations
  - People You May Know, Viewers of this profile ...
- Analytics and Insight
  - Profile Stats: Who Viewed My Profile, Skills
- Visualization
  - InMaps

9

Data Products: Challenges

- LinkedIn: 2nd largest social network
- 120 million members on LinkedIn
- Billions of connections
- Billions of pageviews
- Terabytes of data to process

10

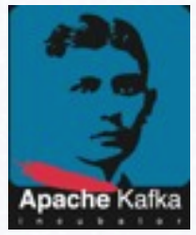
Outline

- What do I mean by Data Products?
- Systems and Tools we use
- Let's build "People You May Know"
- Managing workflow
- Serving data in production
- Data Quality
- Performance

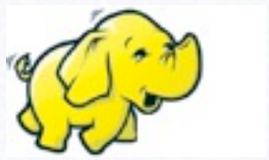
11

Systems and Tools

- Kafka (LinkedIn)



- Hadoop (Apache)



- Azkaban (LinkedIn)

- Voldemort (LinkedIn)

12

Systems and Tools

- Kafka
  - publish-subscribe messaging system
  - transfer data from production to HDFS
- Hadoop
- Azkaban
- Voldemort

13

Systems and Tools

- Kafka
- Hadoop
  - Java MapReduce and Pig
  - process data
- Azkaban
- Voldemort

14

Systems and Tools

- Kafka
- Hadoop
- Azkaban
  - Hadoop workflow management tool
  - to manage hundreds of Hadoop jobs
- Voldemort

15

Systems and Tools

- Kafka
- Hadoop
- Azkaban
- Voldemort
  - Key-value store
  - store output of Hadoop jobs and serve in production

16

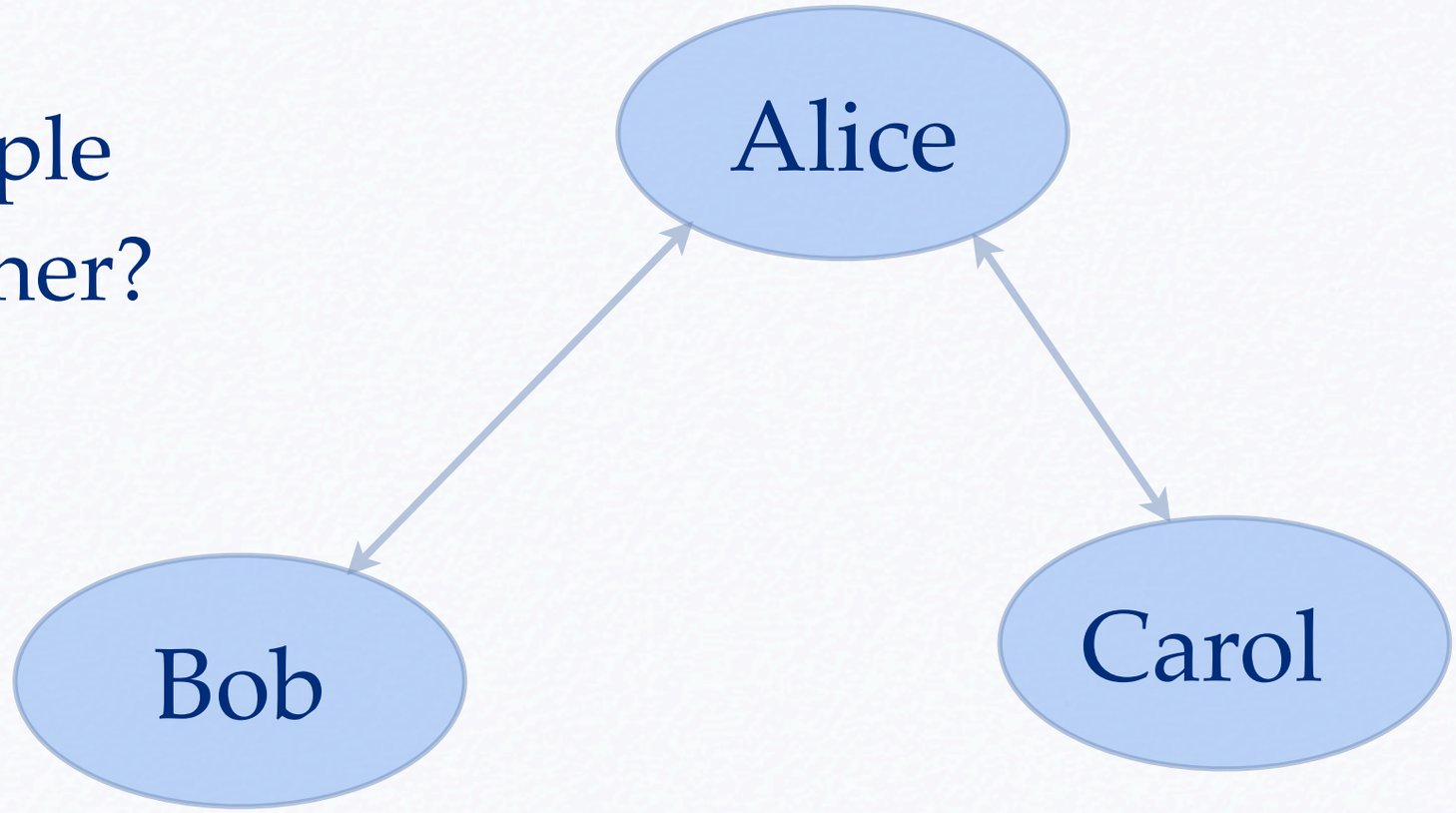
Outline

- What do I mean by Data Products?
- Systems and Tools we use
- Let's build "People You May Know"
- Managing workflow
- Serving data in production
- Data Quality
- Performance

17

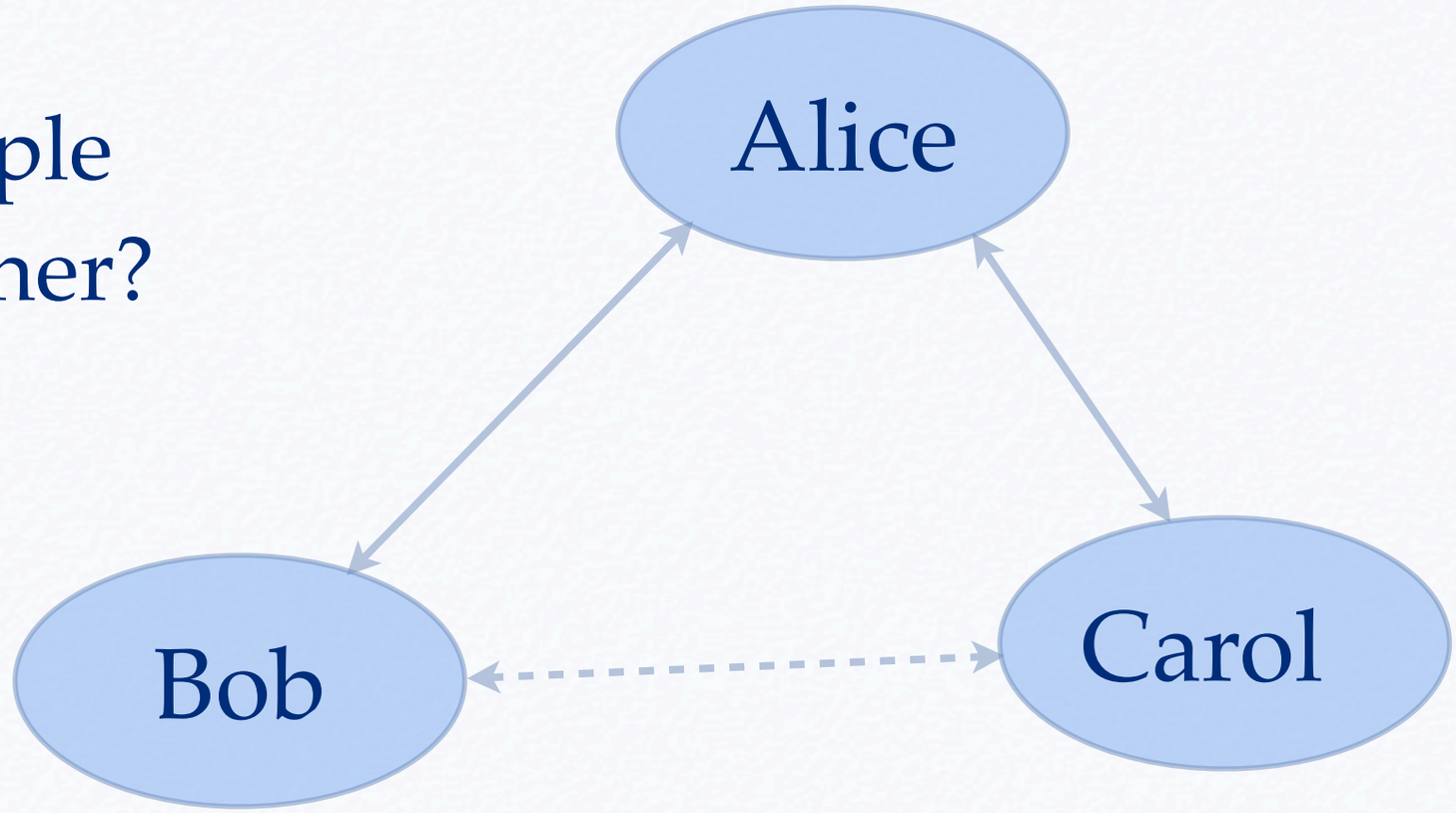
People You May Know

How do people know each other?



People You May Know

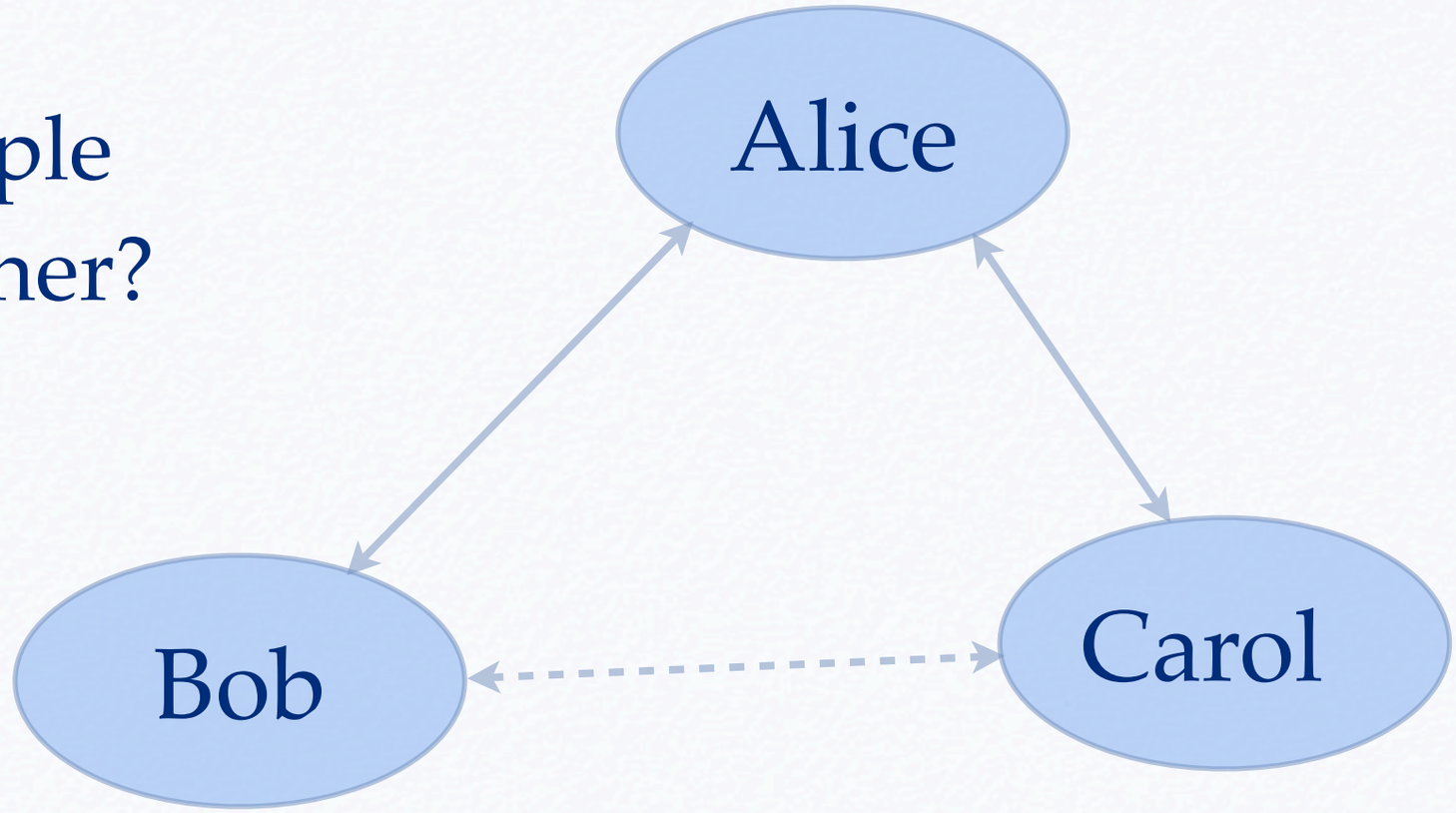
How do people know each other?



19

People You May Know

How do people know each other?

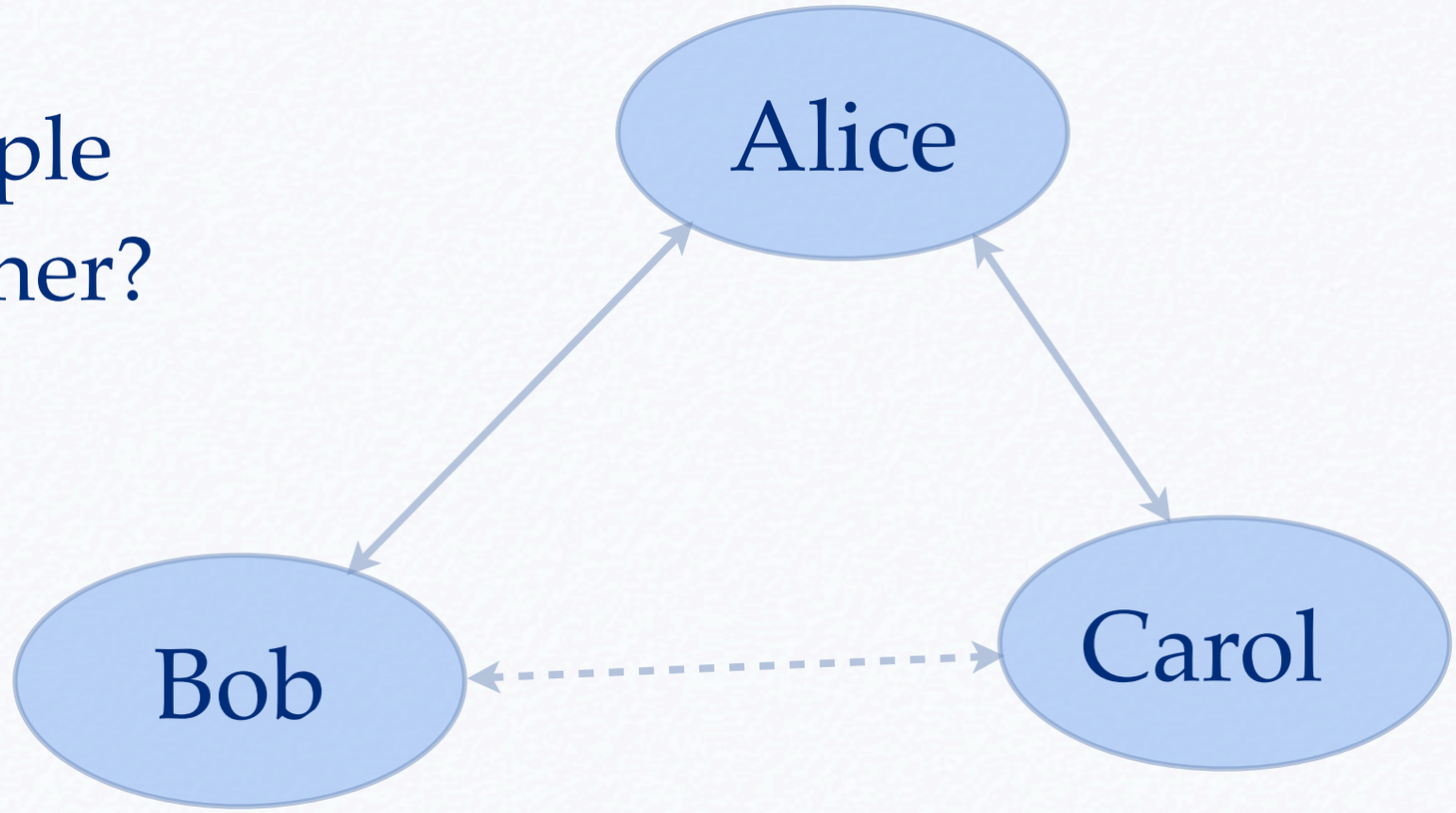


Triangle closing

20

People You May Know

How do people know each other?



Triangle closing

Prob(Bob knows Carol) ~ the # of common connections

21

Triangle Closing in Pig

```
-- connections in (source_id, dest_id) format in both directions
connections = LOAD `connections` USING PigStorage();
group_conn = GROUP connections BY source_id;
pairs = FOREACH group_conn GENERATE
        generatePair(connections.dest_id) as (id1, id2);

common_conn = GROUP pairs BY (id1, id2);
common_conn = FOREACH common_conn GENERATE
                flatten(group) as (source_id, dest_id),
                COUNT(pairs) as common_connections;
STORE common_conn INTO `common_conn` USING PigStorage();
```


22

Pig Overview

- Load: load data, specify format
- Store: store data, specify format
- Foreach, Generate: Projections, similar to select
- Group by: group by column(s)
- Join, Filter, Limit, Order, ...
- User Defined Functions (UDFs)

23

Triangle Closing in Pig

```
-- connections in (source_id, dest_id) format in both directions
connections = LOAD `connections` USING PigStorage();
group_conn = GROUP connections BY source_id;
pairs = FOREACH group_conn GENERATE
        generatePair(connections.dest_id) as (id1, id2);

common_conn = GROUP pairs BY (id1, id2);
common_conn = FOREACH common_conn GENERATE
                flatten(group) as (source_id, dest_id),
                COUNT(pairs) as common_connections;
STORE common_conn INTO `common_conn` USING PigStorage();
```


24

Triangle Closing in Pig

```
-- connections in (source_id, dest_id) format in both directions
connections = LOAD `connections` USING PigStorage();
group_conn = GROUP connections BY source_id;
pairs = FOREACH group_conn GENERATE
        generatePair(connections.dest_id) as (id1, id2);

common_conn = GROUP pairs BY (id1, id2);
common_conn = FOREACH common_conn GENERATE
                flatten(group) as (source_id, dest_id),
                COUNT(pairs) as common_connections;
STORE common_conn INTO `common_conn` USING PigStorage();
```


25

Triangle Closing in Pig

```
-- connections in (source_id, dest_id) format in both directions
connections = LOAD `connections` USING PigStorage();
group_conn = GROUP connections BY source_id;
pairs = FOREACH group_conn GENERATE
        generatePair(connections.dest_id) as (id1, id2);

common_conn = GROUP pairs BY (id1, id2);
common_conn = FOREACH common_conn GENERATE
                flatten(group) as (source_id, dest_id),
                COUNT(pairs) as common_connections;
STORE common_conn INTO `common_conn` USING PigStorage();
```


26

Triangle Closing in Pig

```
-- connections in (source_id, dest_id) format in both directions
connections = LOAD `connections` USING PigStorage();
group_conn = GROUP connections BY source_id;
pairs = FOREACH group_conn GENERATE
        generatePair(connections.dest_id) as (id1, id2);

common_conn = GROUP pairs BY (id1, id2);
common_conn = FOREACH common_conn GENERATE
                flatten(group) as (source_id, dest_id),
                COUNT(pairs) as common_connections;
STORE common_conn INTO `common_conn` USING PigStorage();
```


27

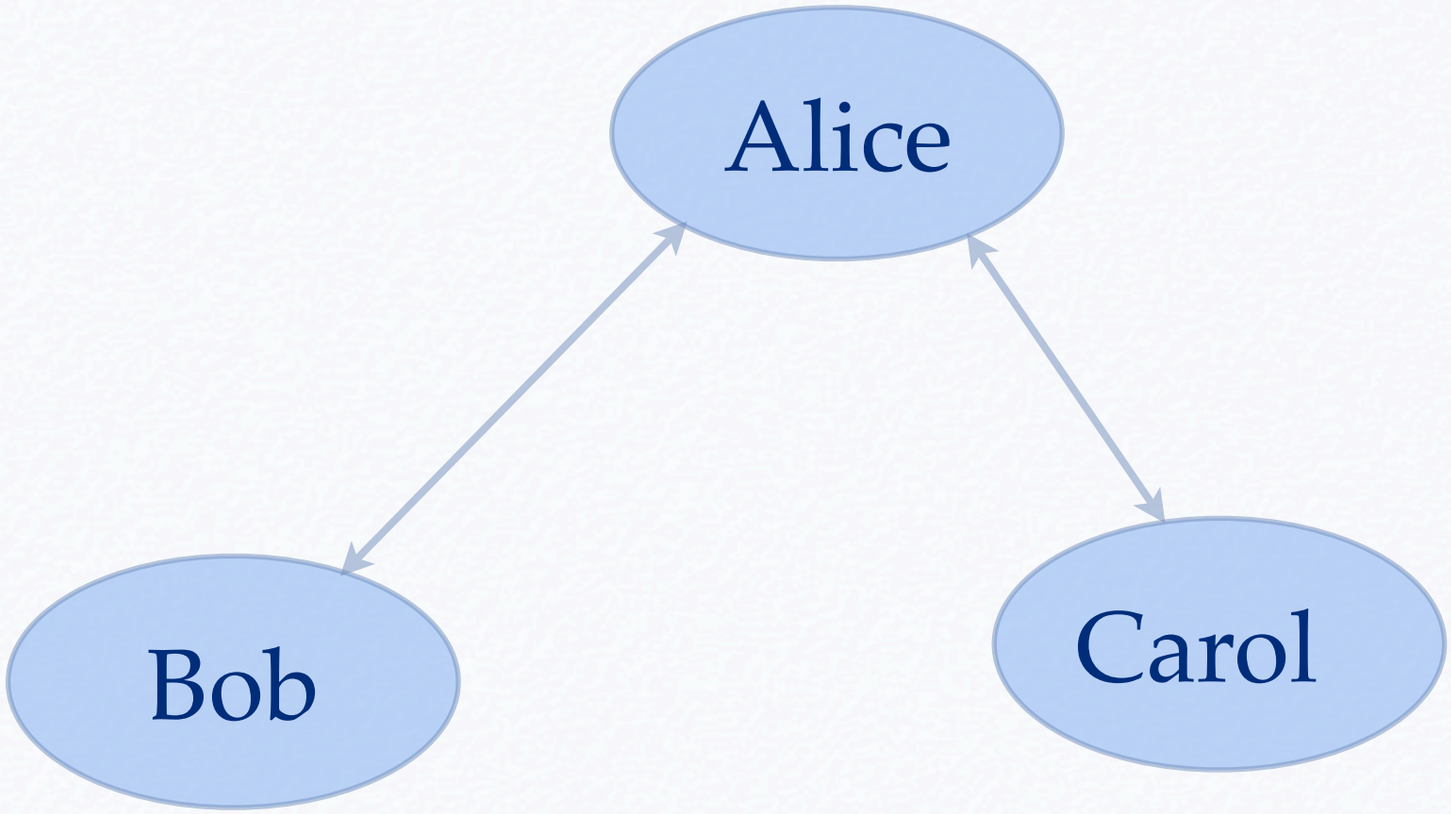
Triangle Closing in Pig

```
-- connections in (source_id, dest_id) format in both directions
connections = LOAD `connections` USING PigStorage();
group_conn = GROUP connections BY source_id;
pairs = FOREACH group_conn GENERATE
        generatePair(connections.dest_id) as (id1, id2);

common_conn = GROUP pairs BY (id1, id2);
common_conn = FOREACH common_conn GENERATE
                flatten(group) as (source_id, dest_id),
                COUNT(pairs) as common_connections;
STORE common_conn INTO `common_conn` USING PigStorage();
```


28

Triangle Closing Example



1.(A,B),(B,A),(A,C),(C,A)

2.(A,{B,C}),(B,{A}),(C,{A})

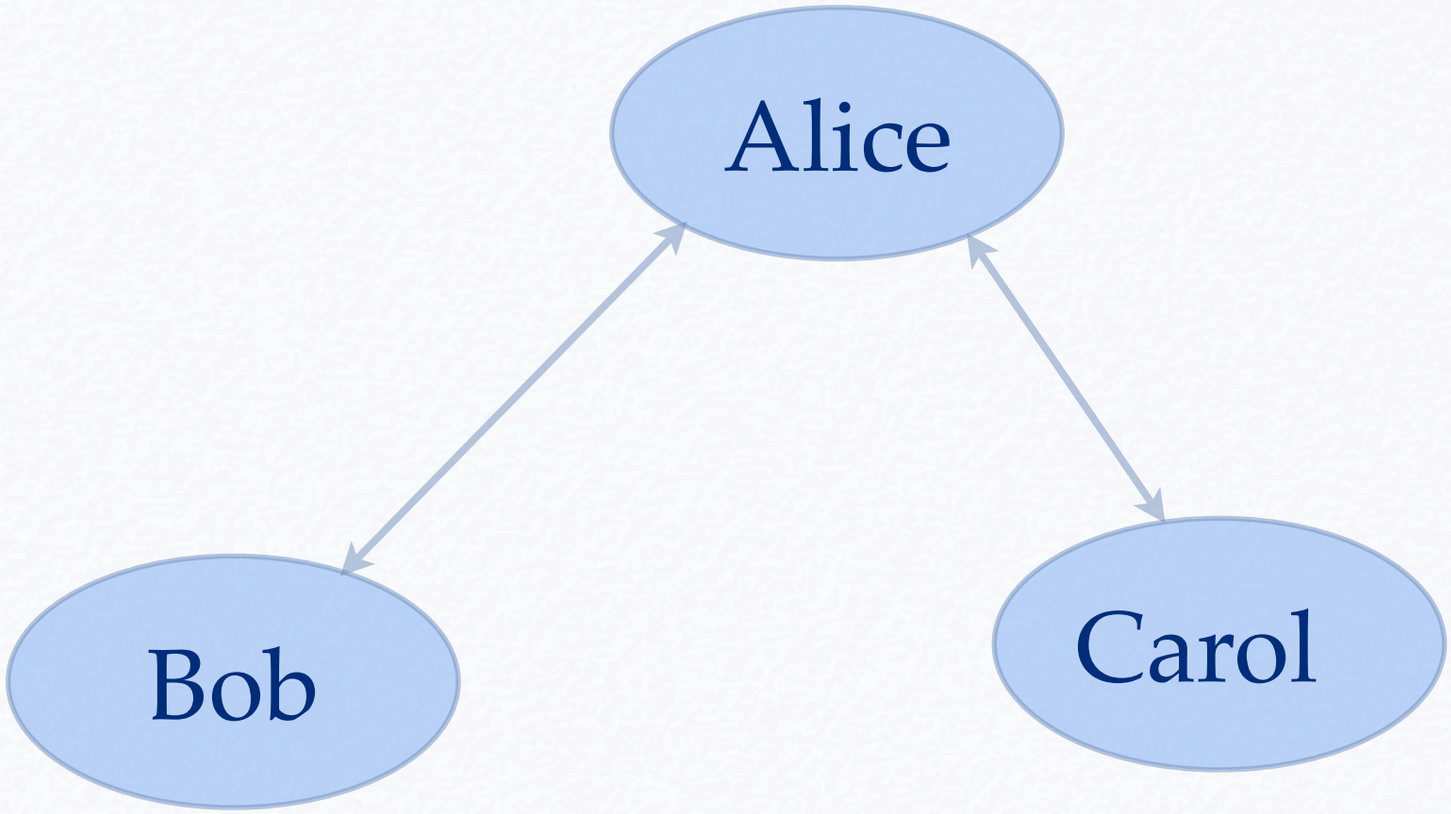
3.(A,{B,C}),(A,{C,B})

4.(B,C,1), (C,B,1)

```
connections = LOAD `connections` USING
PigStorage();
```


29

Triangle Closing Example



1.(A,B),(B,A),(A,C),(C,A)

2.(A,{B,C}),(B,{A}),(C,{A})

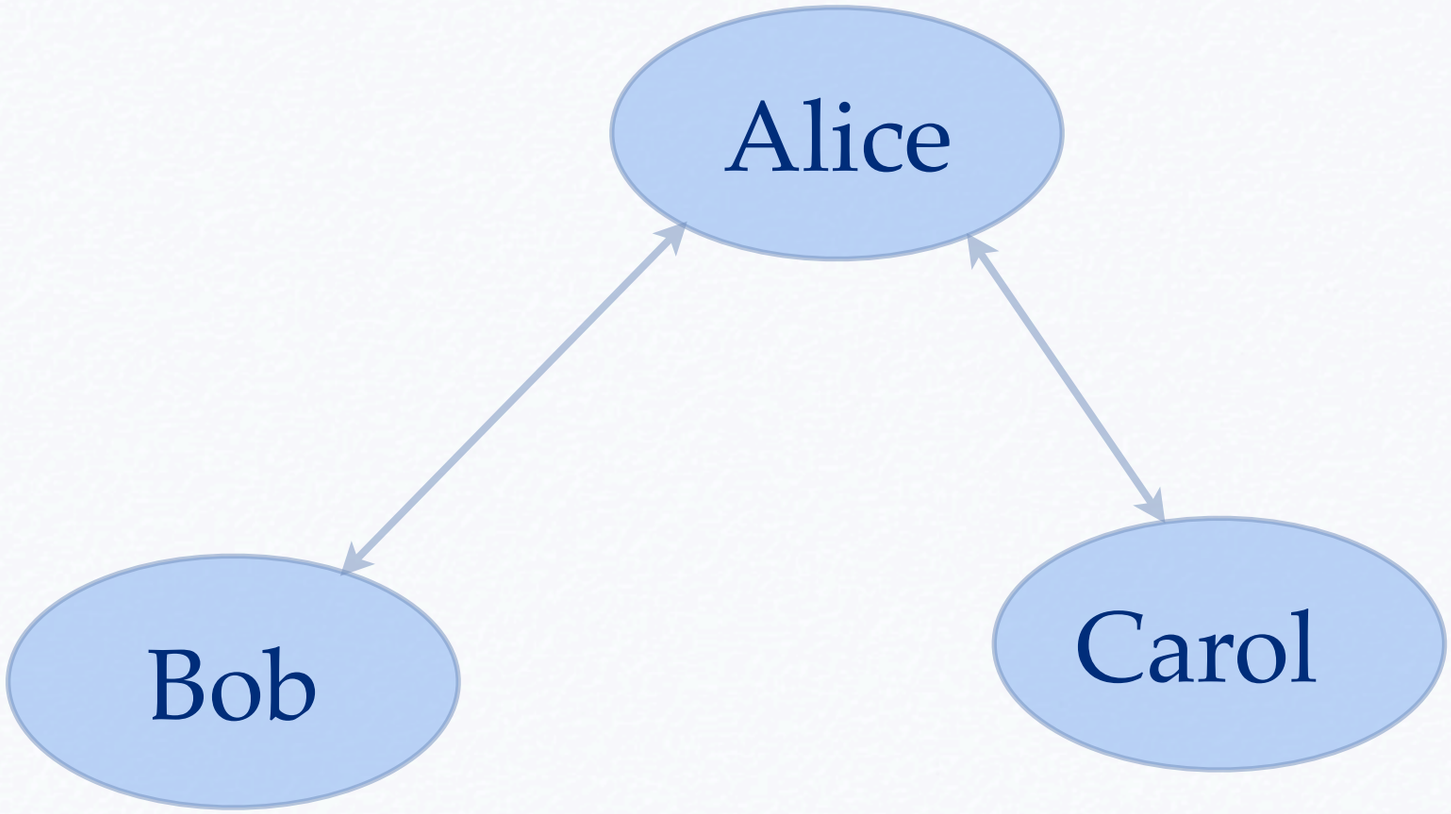
3.(A,{B,C}),(A,{C,B})

4.(B,C,1), (C,B,1)

```
group_conn = GROUP connections BY
source_id;
```


30

Triangle Closing Example



1.(A,B),(B,A),(A,C),(C,A)

2.(A,{B,C}),(B,{A}),(C,{A})

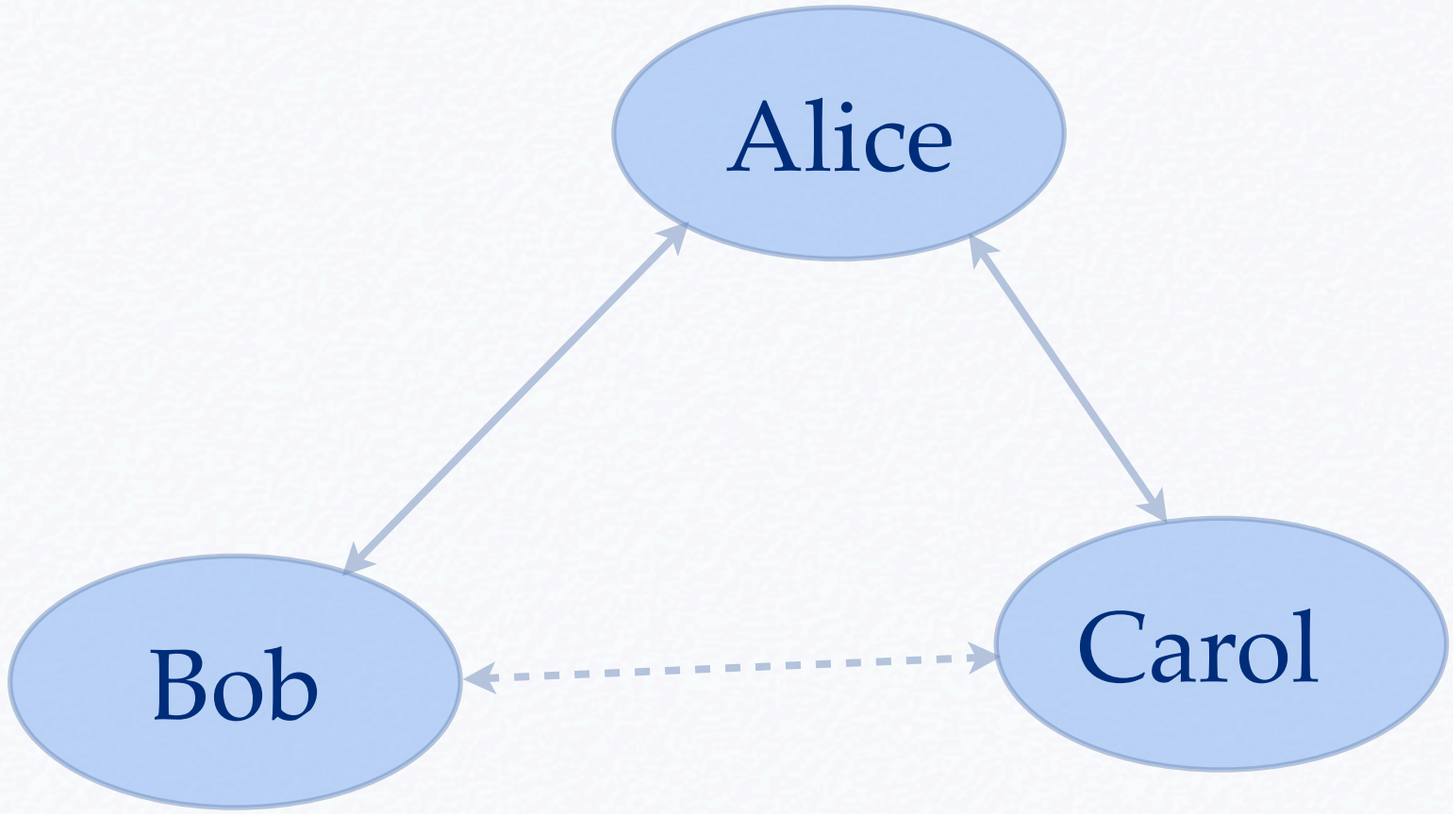
3.(A,{B,C}),(A,{C,B})

4.(B,C,1), (C,B,1)

```
pairs = FOREACH group_conn GENERATE
generatePair(connections.dest_id) as (id1, id2);
```


31

Triangle Closing Example



1.(A,B),(B,A),(A,C),(C,A)

2.(A,{B,C}),(B,{A}),(C,{A})

3.(A,{B,C}),(A,{C,B})

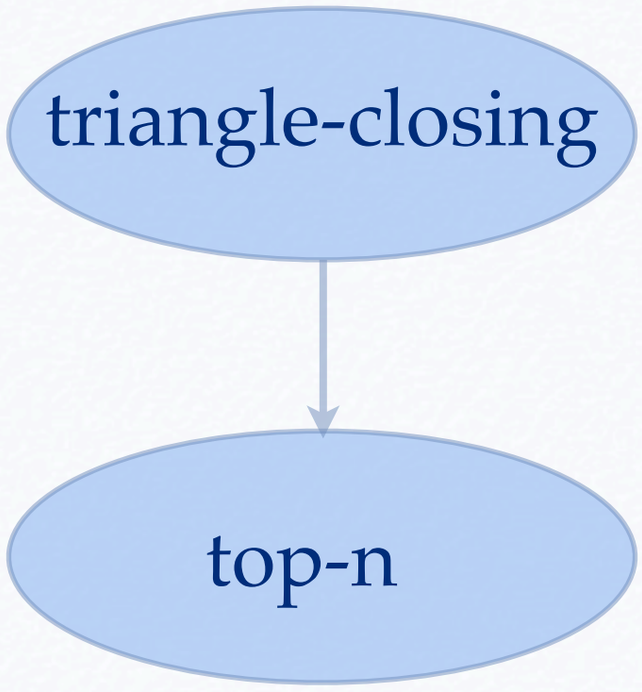
4.(B,C,1), (C,B,1)

```
common_conn = GROUP pairs BY (id1, id2);
common_conn = FOREACH common_conn
GENERATE flatten(group) as (source_id, dest_id),
COUNT(pairs) as common_connections;
```


32

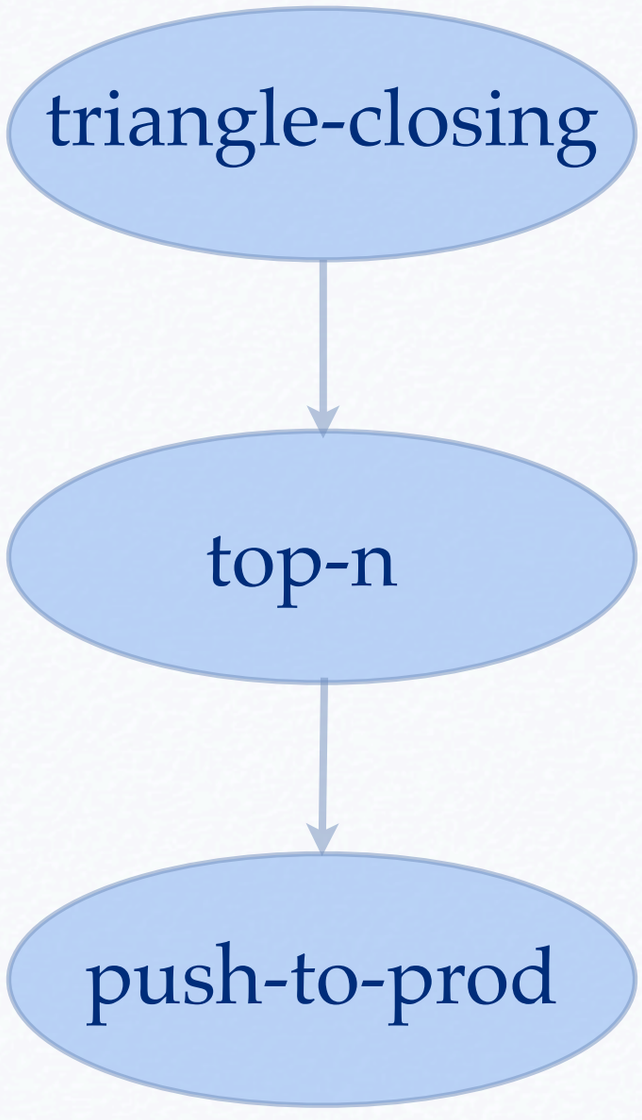
Our Workflow

Our Workflow



34

Our Workflow



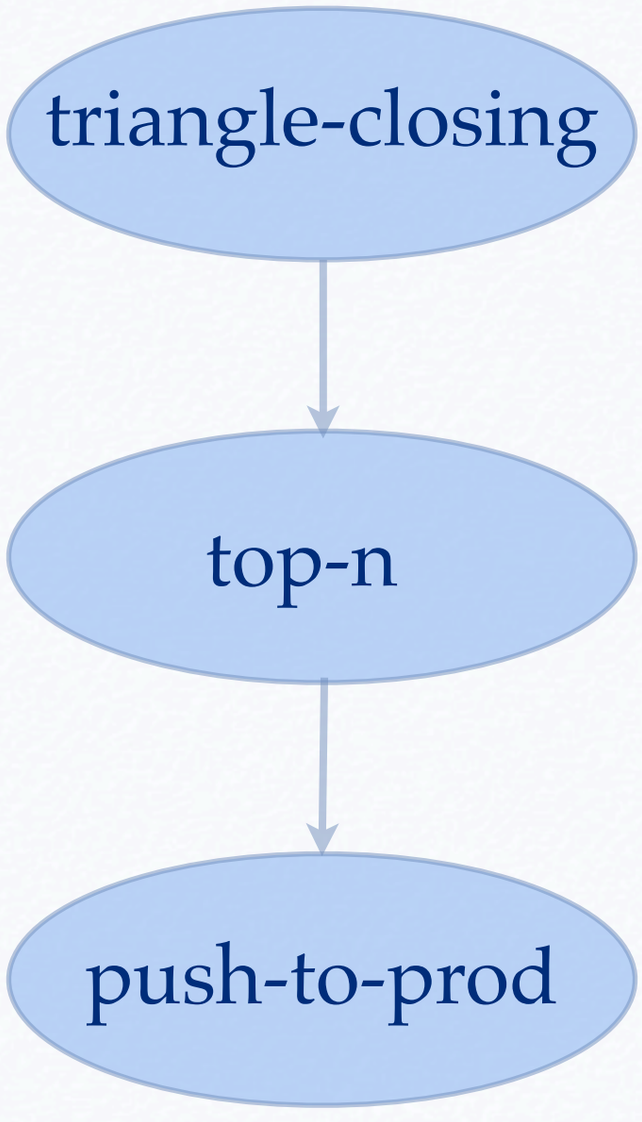
35

Outline

- What do I mean by Data Products?
- Systems and Tools we use
- Let's build "People You May Know"
- Managing workflow
- Serving data in production
- Data Quality
- Performance

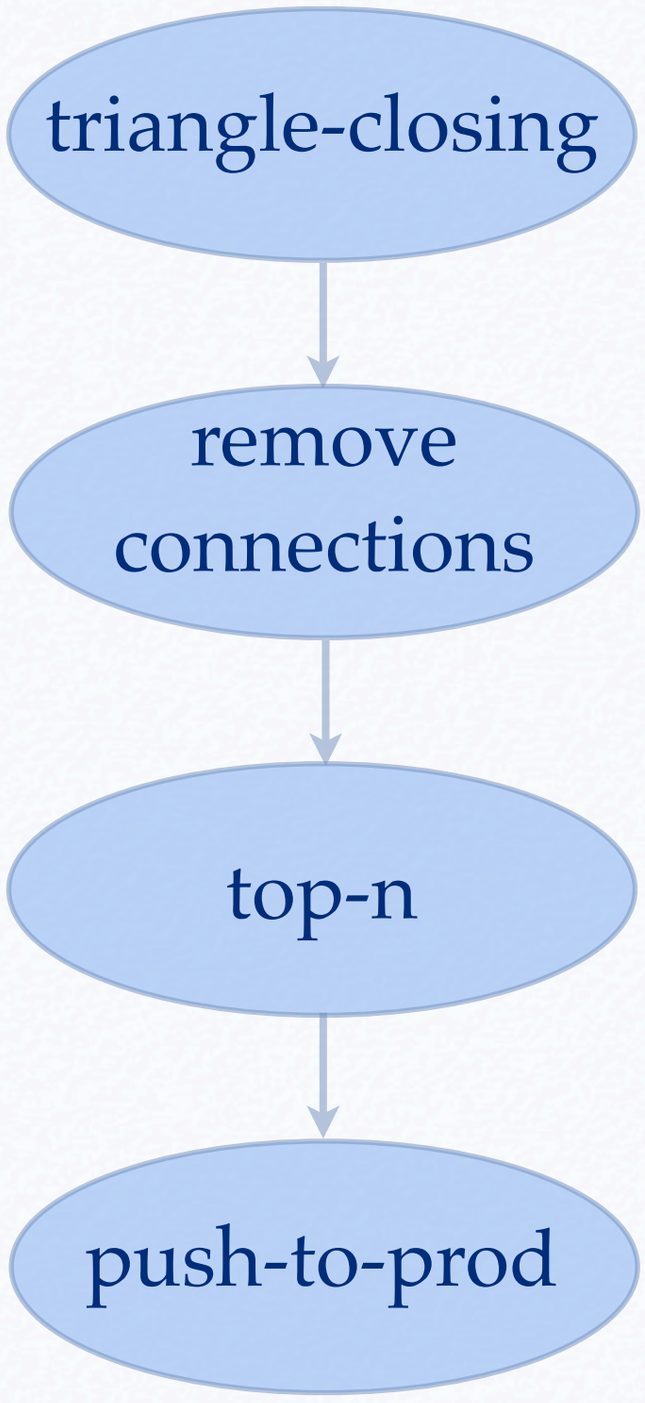
36

Our Workflow



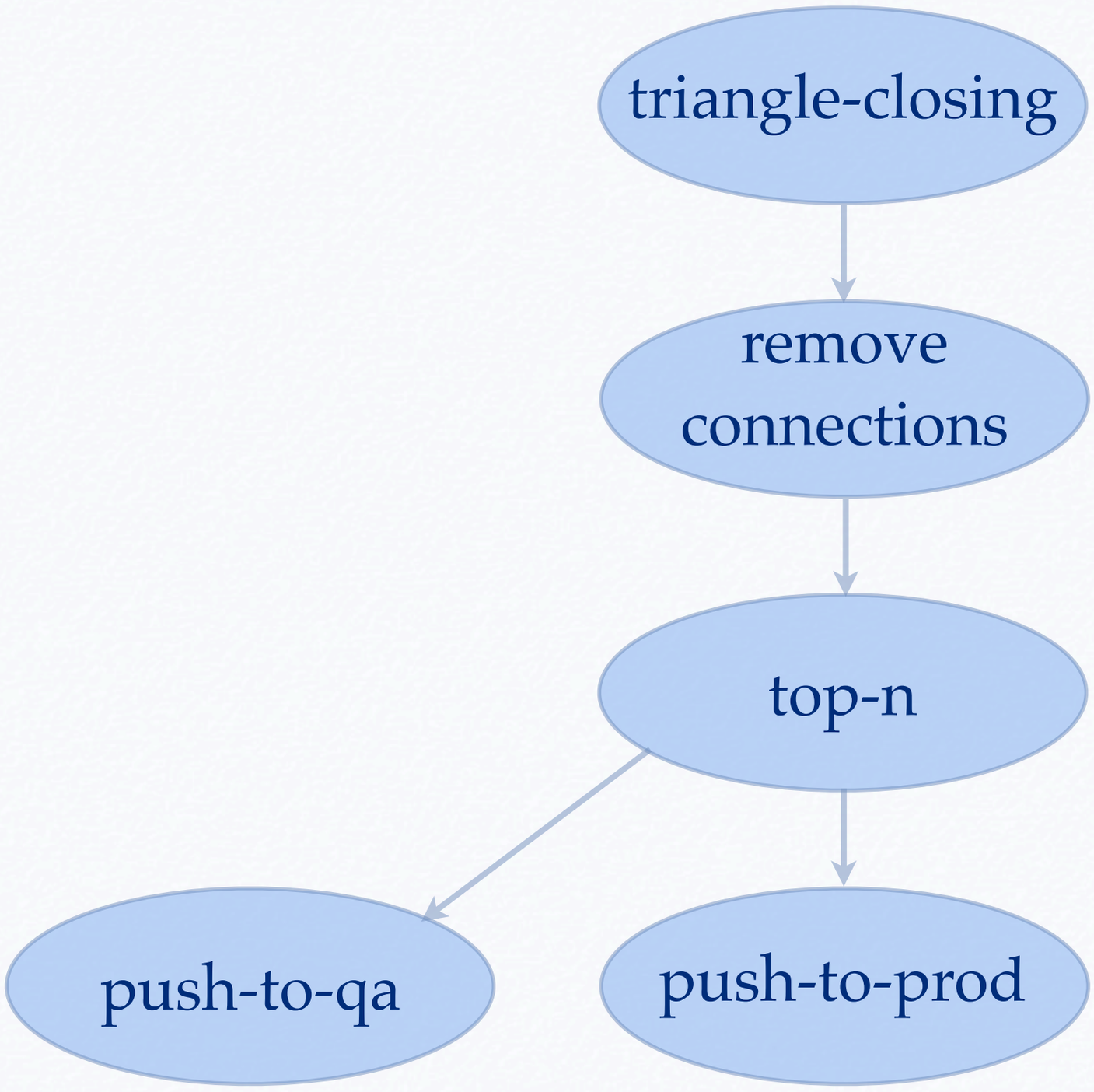
37

Our Workflow



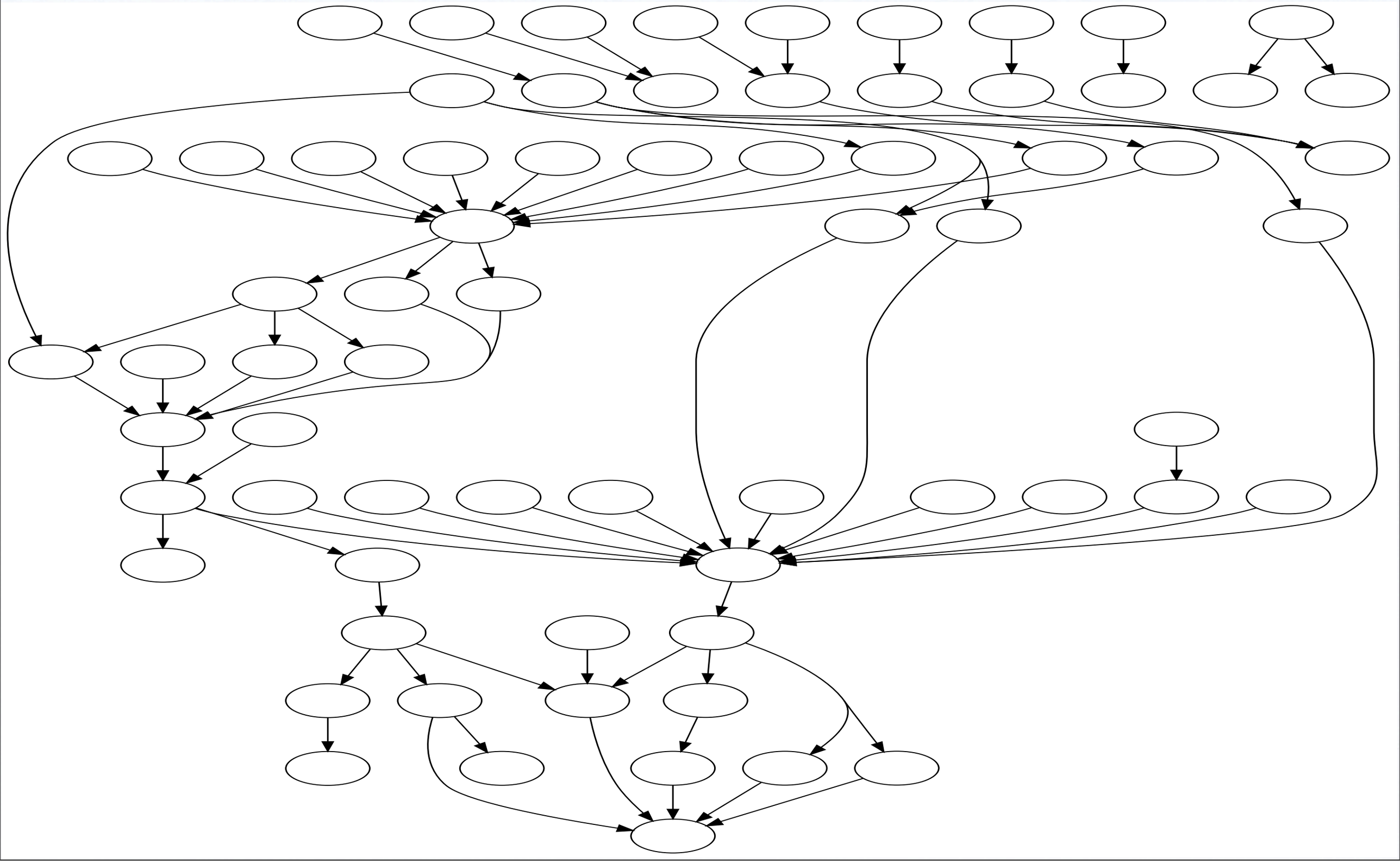
38

Our Workflow



39

PYMK Workflow



Workflow Requirements

- Dependency management
- Regular Scheduling
- Monitoring
- Diverse jobs: Java, Pig, Clojure
- Configuration/Parameters
- Resource control/locking
- Restart/Stop/Retry
- Visualization
- History
- Logs

41

Workflow Requirements

- Dependency management
- Regular Scheduling
- Monitoring
- Diverse jobs: Java, Pig, Clojure
- Configuration/Parameters
- Resource control/locking
- Restart/Stop/Retry
- Visualization
- History
- Logs

Azkaban

42

Sample Azkaban Job Spec

type=pig

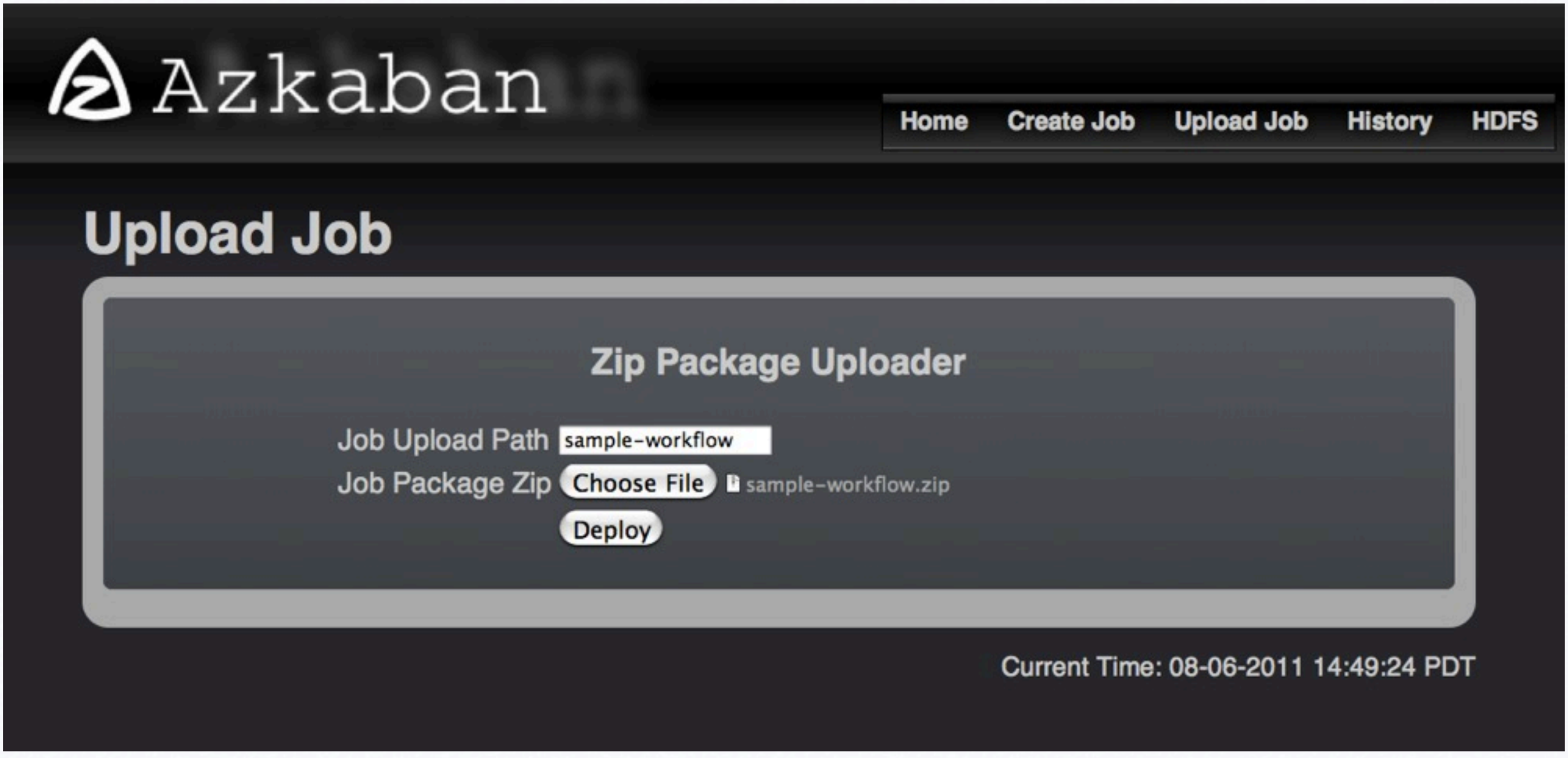
pig.script=top-n.pig

dependencies=remove-connections

top.n.size=100

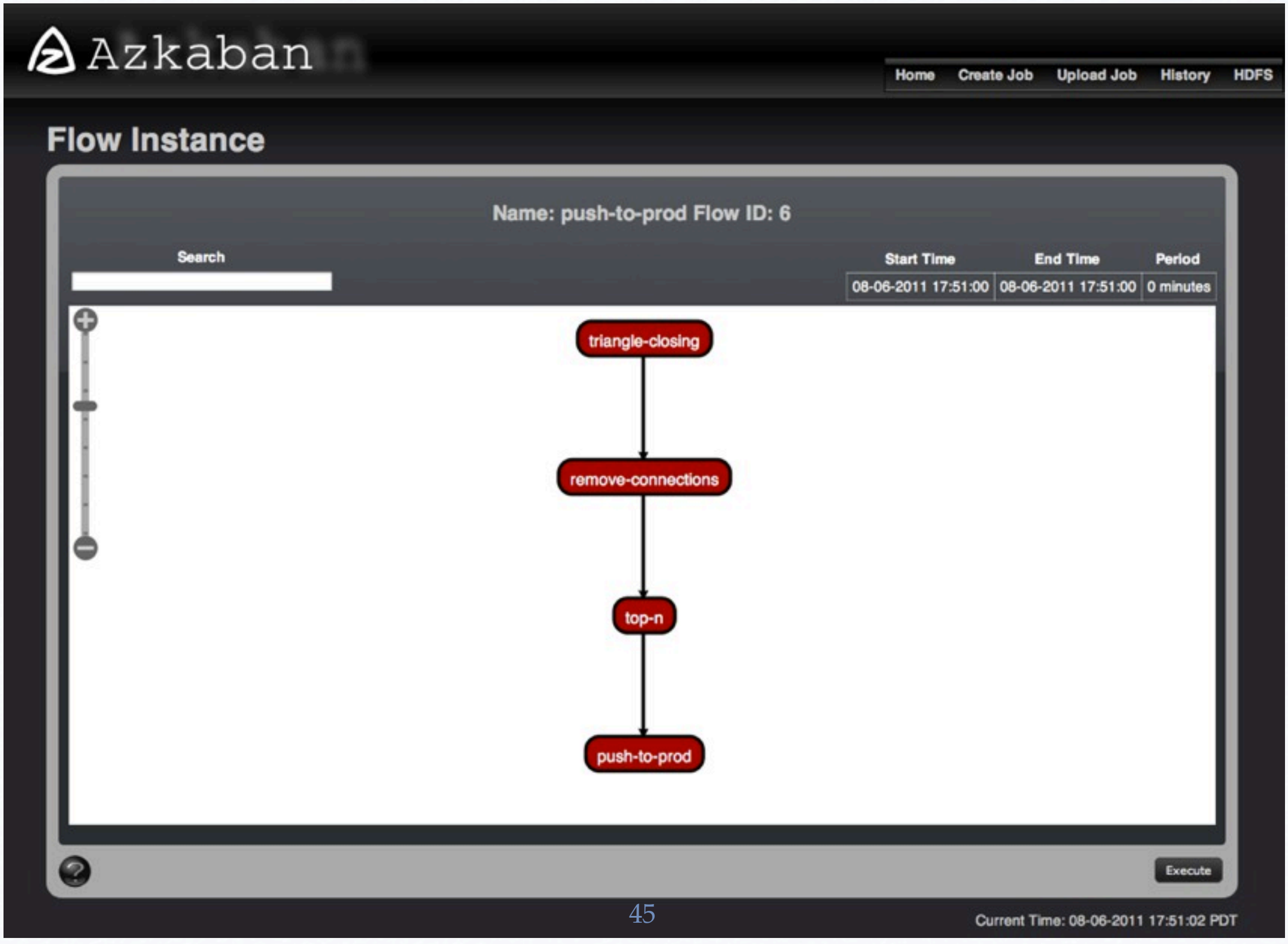
43

Azkaban Workflow



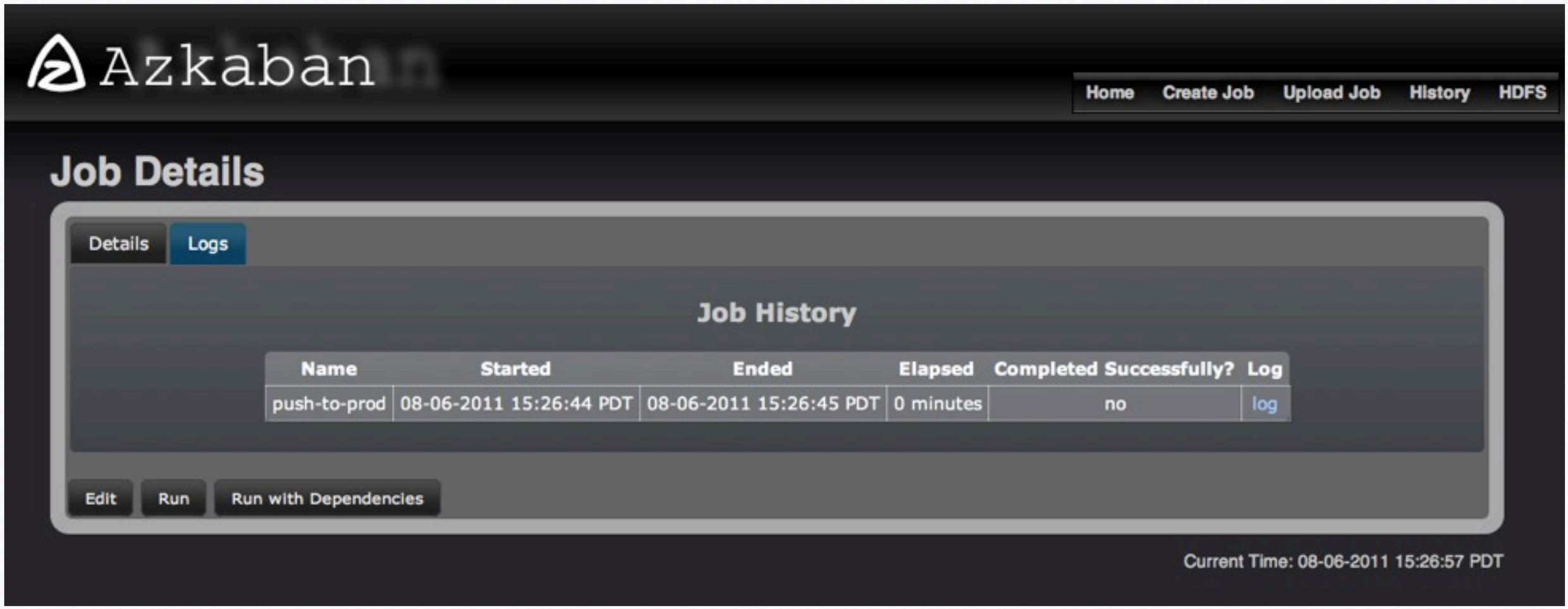
44

Azkaban Workflow



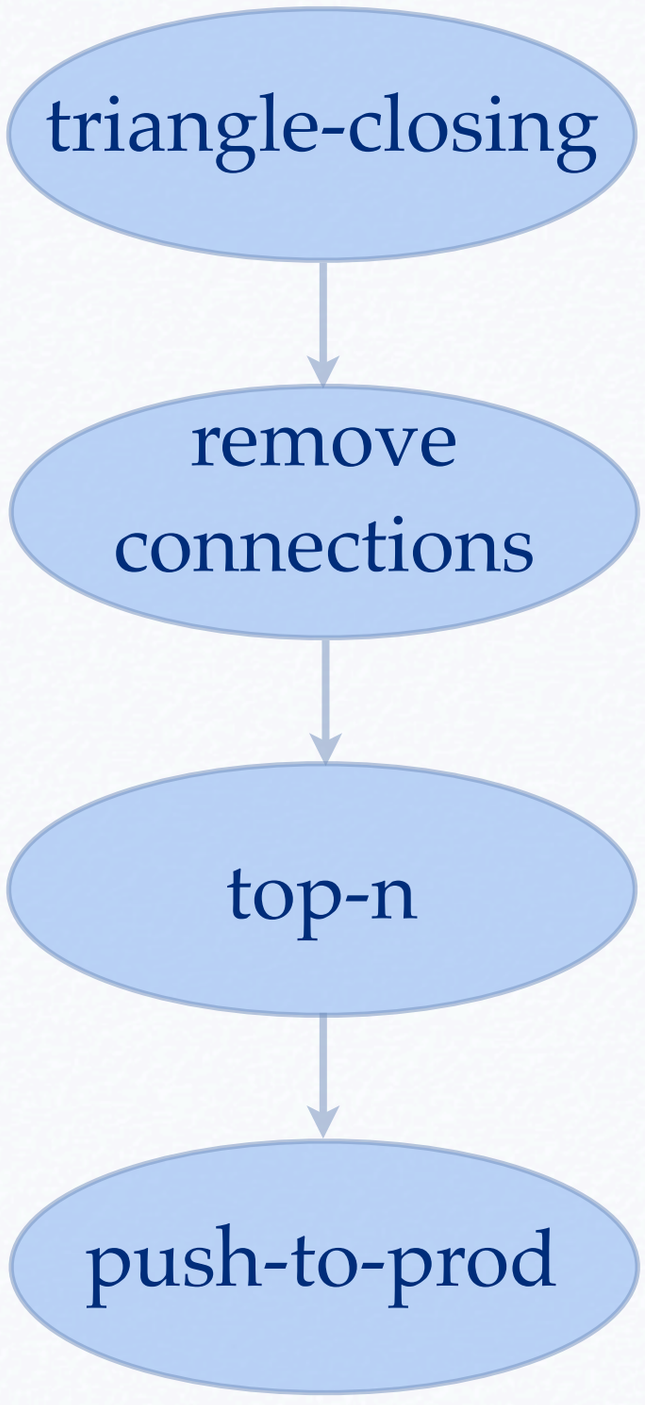
45

Azkaban Workflow



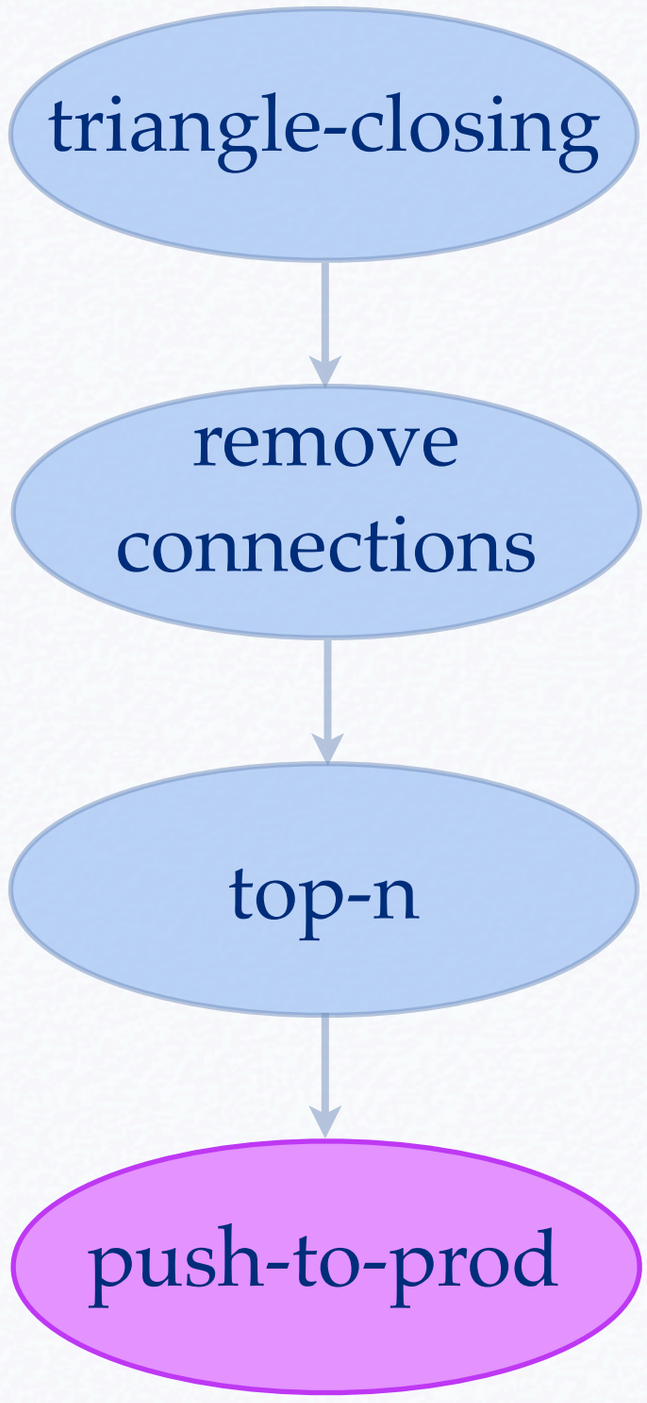
46

Our Workflow



47

Our Workflow



48

Outline

- What do I mean by Data Products?
- Systems and Tools we use
- Let's build "People You May Know"
- Managing workflow
- Serving data in production
- Data Quality
- Performance

49

Production Storage

- Requirements
  - Large amount of data/Scalable
  - Quick lookup/low latency
  - Versioning and Rollback
  - Fault tolerance
  - Offline index building

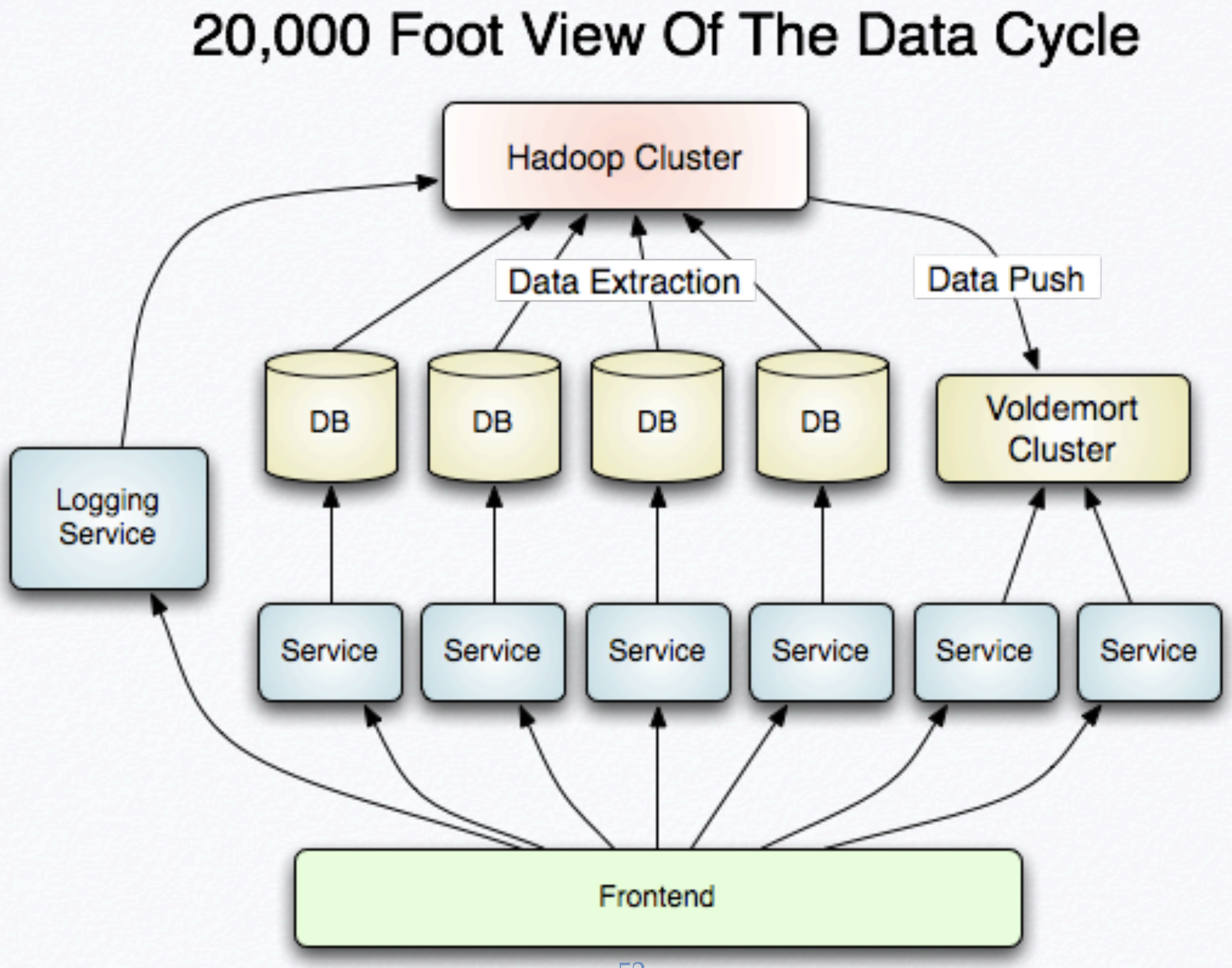
50

Voldemort Storage

- Large amount of data/Scalable
- Quick lookup/low latency
- Versioning and Rollback
- Fault tolerance through replication
- Read only
- Offline index building

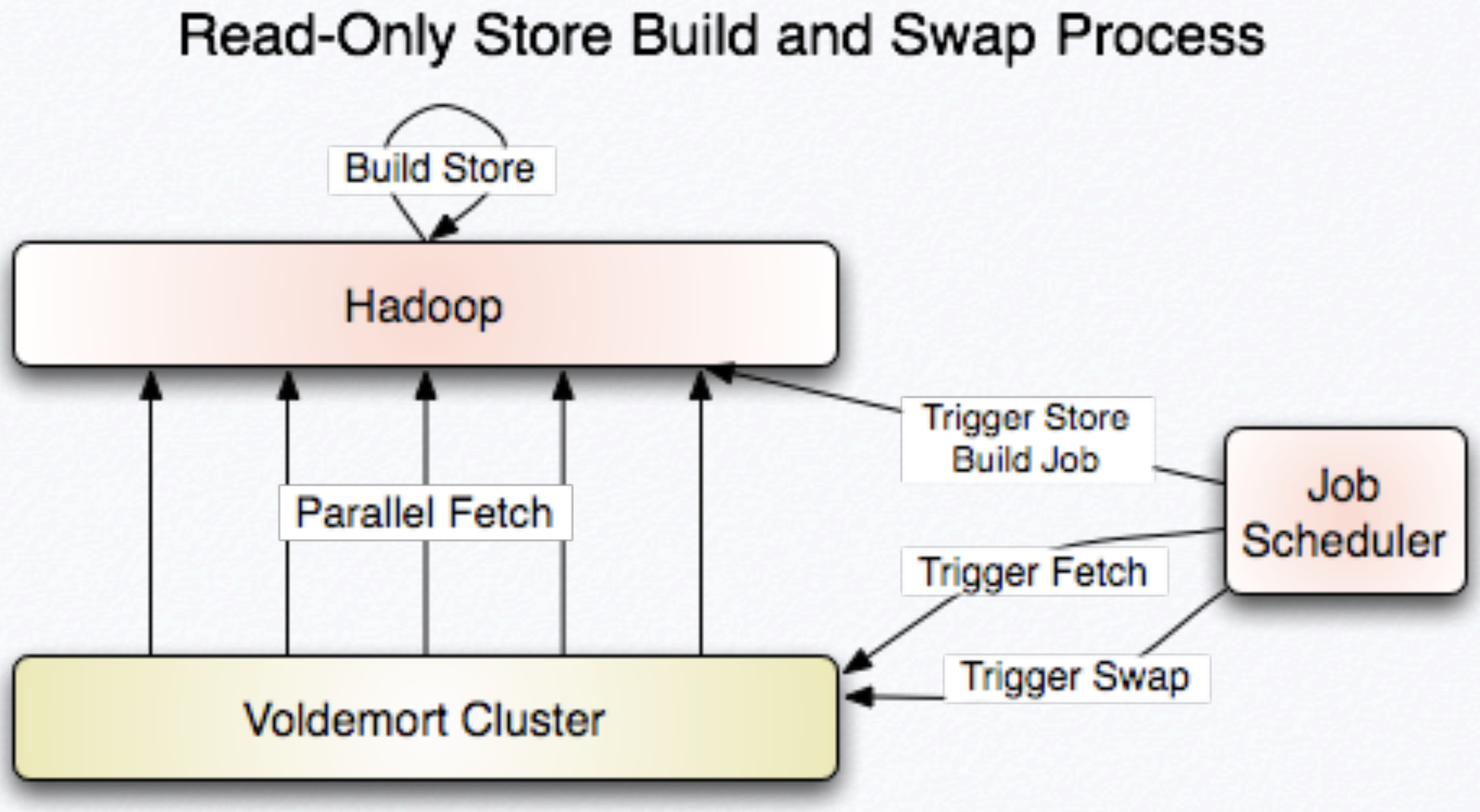
51

Data Cycle



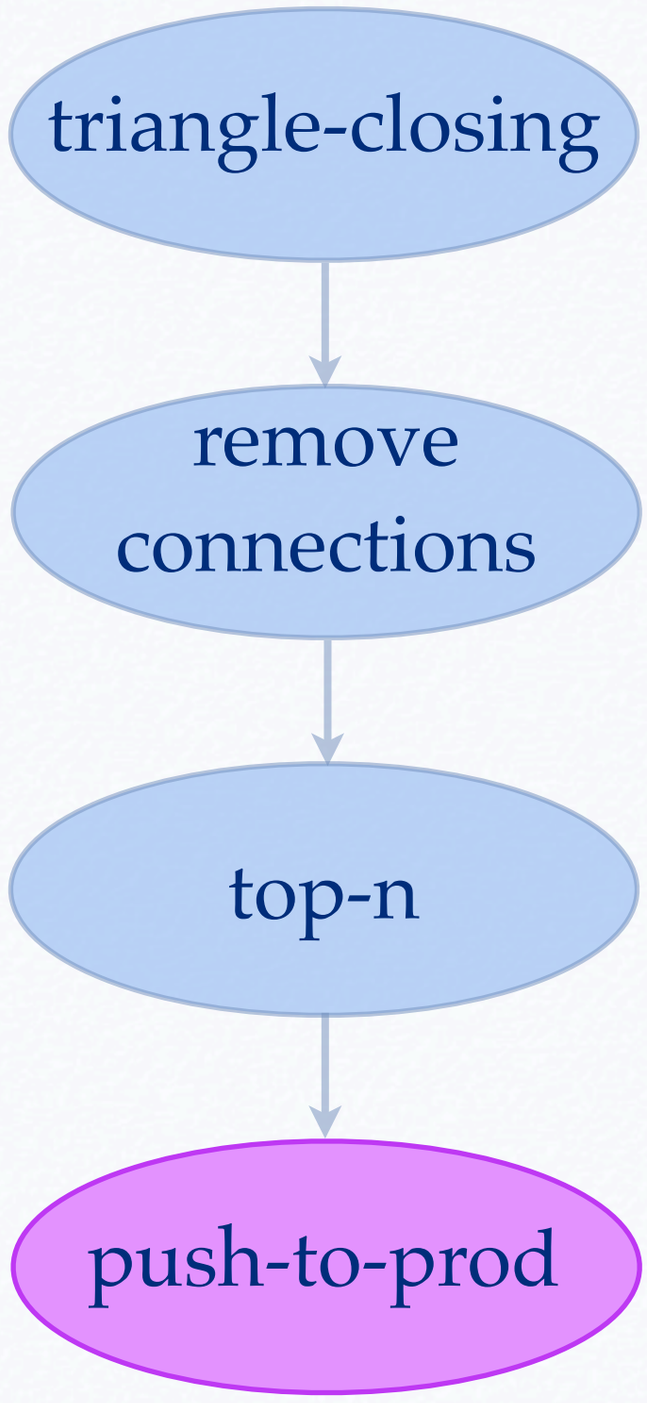
52

Voldemort RO Store



53

Our Workflow



54

Outline

- What do I mean by Data Products?
- Systems and Tools we use
- Let's build "People You May Know"
- Managing workflow
- Serving data in production
- Data Quality
- Performance

55

Data Quality

- Verification
- QA store with viewer
- Explain
- Versioning / Rollback
- Unit tests

56

Outline

- What do I mean by Data Products?
- Systems and Tools we use
- Let's build "People You May Know"
- Managing workflow
- Serving data in production
- Data Quality
- Performance

57

Performance

Performance

- Symmetry
  - Bob knows Carol then Carol knows Bob

Performance

- Symmetry
  - Bob knows Carol then Carol knows Bob
- Limit
  - Ignore members with > k connections

58

Performance

- Symmetry
  - Bob knows Carol then Carol knows Bob
- Limit
  - Ignore members with > k connections
- Sampling
  - Sample k-connections

58

Things Covered

- What do I mean by Data Products?
- Systems and Tools we use
- Let's build "People You May Know"
- Managing workflow
- Serving data in production
- Data Quality
- Performance

59

SNA Team

- Thanks to SNA Team at LinkedIn
- http://sna-projects.com
- We are hiring!

60

Questions?