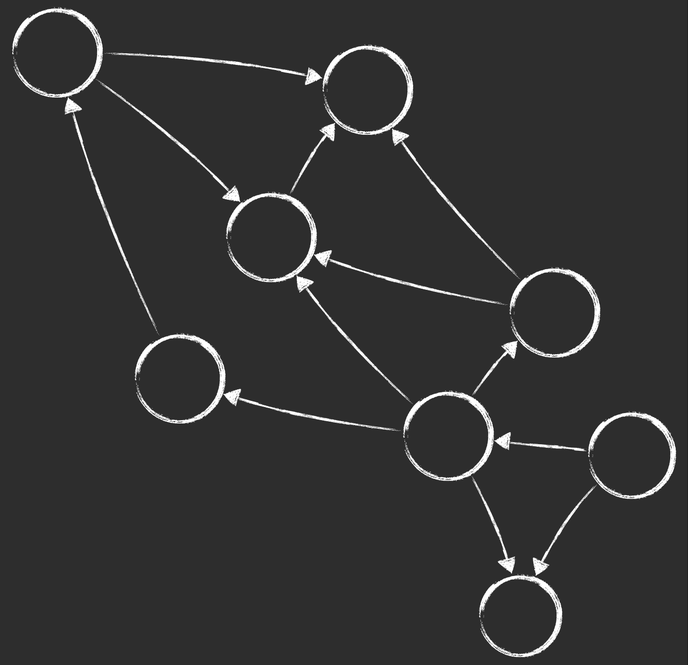
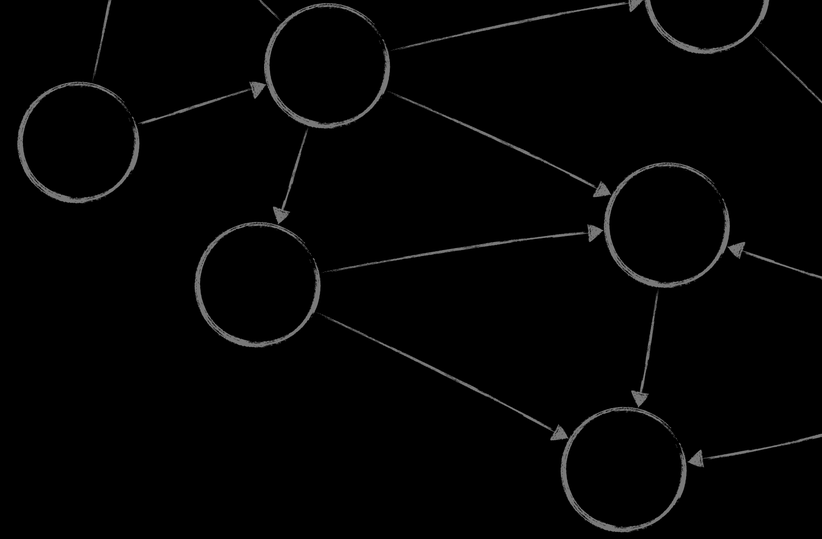
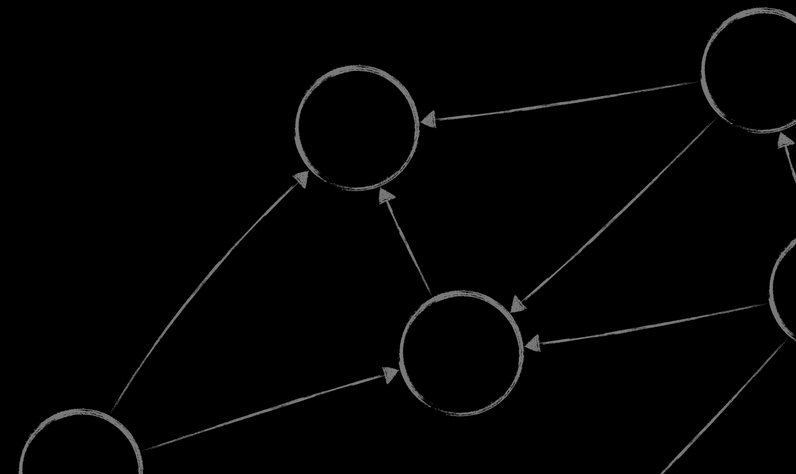
O'REILLY®
oscon



# Open Source Big Graph Analytics on Neo4j with Apache Spark

Kenny Bastani (@kennybastani)





oscon.com
#oscon

# Speaker Introduction

- Graph database enthusiast
- Building microservice architectures
- Lead developer at Digital Insight
- Co-author of upcoming O’Reilly book — Spring Boot Essentials



# The Problem

It's hard to analyze graphs at scale



# The importance of graph algorithms

- PageRank gave us Google
- Friend of a friend gave us Facebook
- Collaborative filtering makes Netflix recommendations awesome



# Why is it so hard to do this stuff?



# Enemy #1:

Relational databases store data in ways that make it difficult to extract graphs for analysis



# Enemy #2:

If you still think Big Data is a buzz word

You haven’t had to feel the pain of failing at it.



# When you hit a wall because your data is too big

You start to see what this big data thing is all about.



# Distributed File Systems

Distributed file systems are a foundational component of big data analytics

Chops things into manageable sized blocks, usually 64MB

Spreads those blocks out across a cluster of VM resources



# Hadoop MapReduce

Worth mentioning, Hadoop started this whole distributed MapReduce thing

You could translate the raw data from a CSV and turn it into a map of keys to values

Keys are distributed per node and used to reduce the values into a partitioned analysis



# Graph algorithms can be evil at scale

It depends on the complexity of your graph

How many strongly connected components you have

But since some graph algorithms like PageRank are iterative

You have to iterate from one stage and use the results of the previous stage



# It doesn't matter how many nodes you have in your cluster

For iterative graph algorithms, the complexity of the graph will make you or break you

Graphs with high complexity need a lot of memory to be processed iteratively

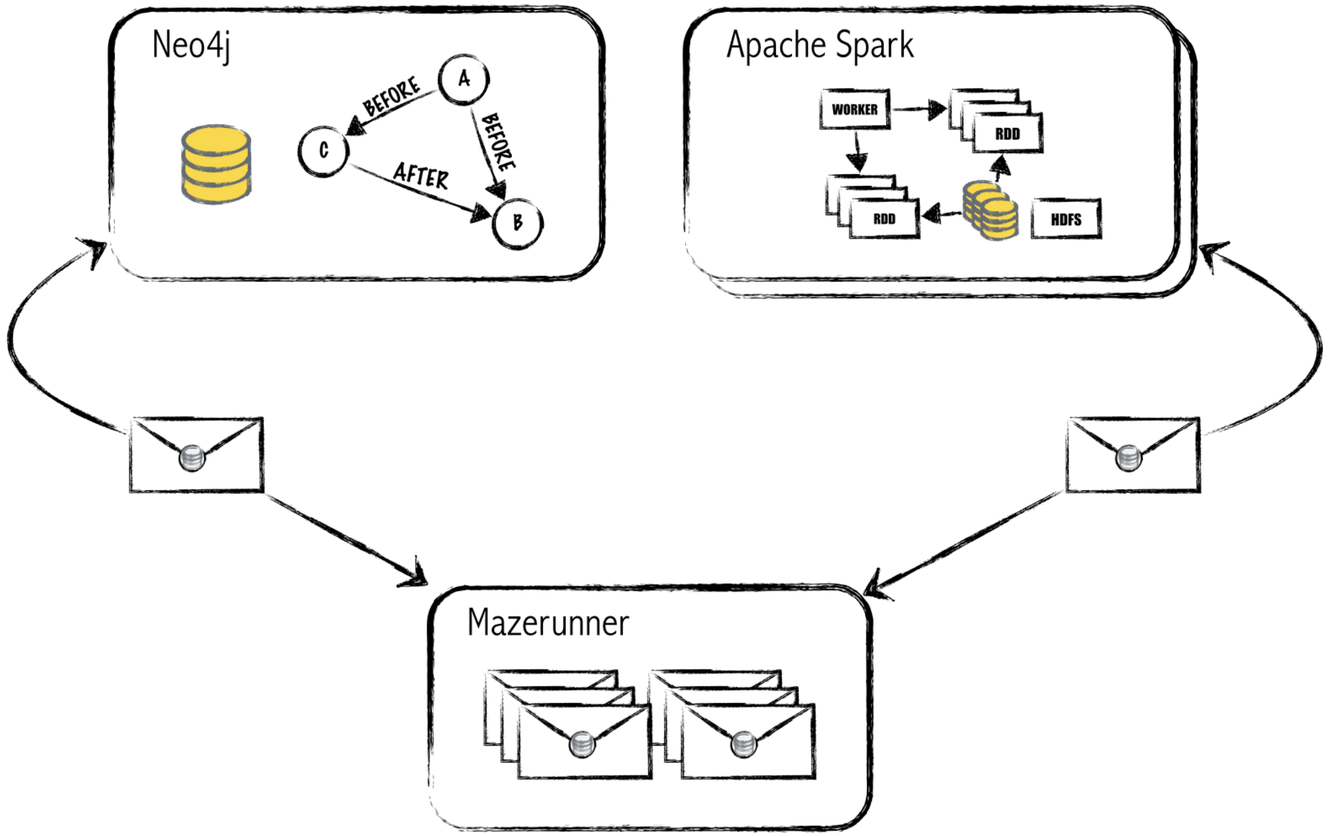


# Neo4j Mazerunner Project

2-way Graph ETL



# What is Neo4j Mazerunner?



# The basic idea is…

Graph databases need ETL so you can analyze your data and look it up later.



# Docker

If you’re not up on Docker, let me give you a quick intro.



# Docker

Docker is a VM framework that let’s you easily create a recipe for an image and deploy applications with ease.

The idea is that infrastructure and operational complexity makes it hard for agile development of new products.



# Why?

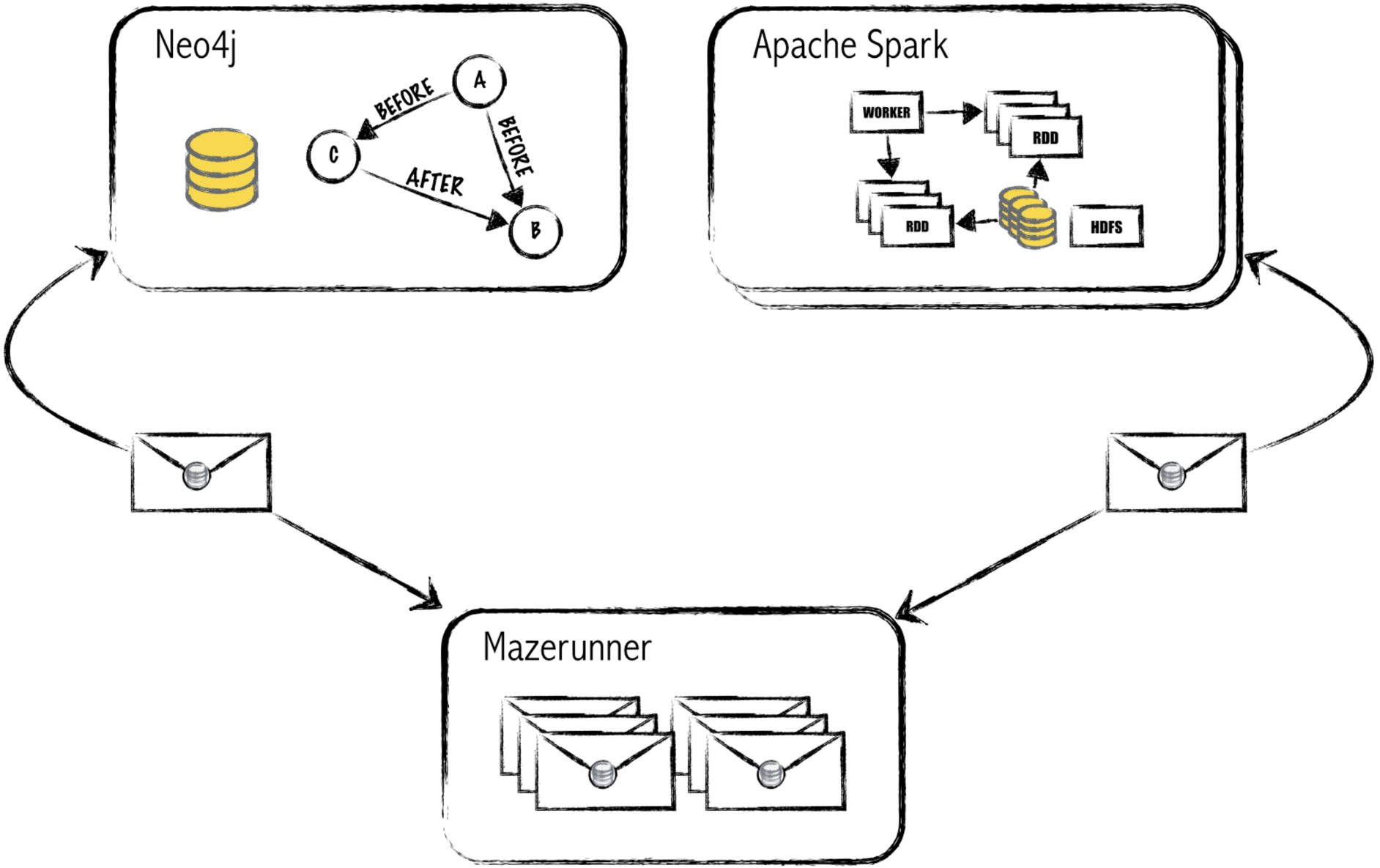
If I am an engineer on a product team, I want to choose my own software libraries and languages to solve problems.



# Microservices and Docker

- If want to build a new service, use whatever application framework you want. As long as you communicate over REST.
- Docker gives you the freedom to use Neo4j, or MySQL, or MongoDB or whatever application dependency you want inside your container.




Neo4j
A
BEFORE
BEFORE
C
AFTER
B
Apache Spark
WORKER
RDD
RDD
HDFS
Mazerunner

# Docker Compose

Docker Compose allows you to run multi-container applications

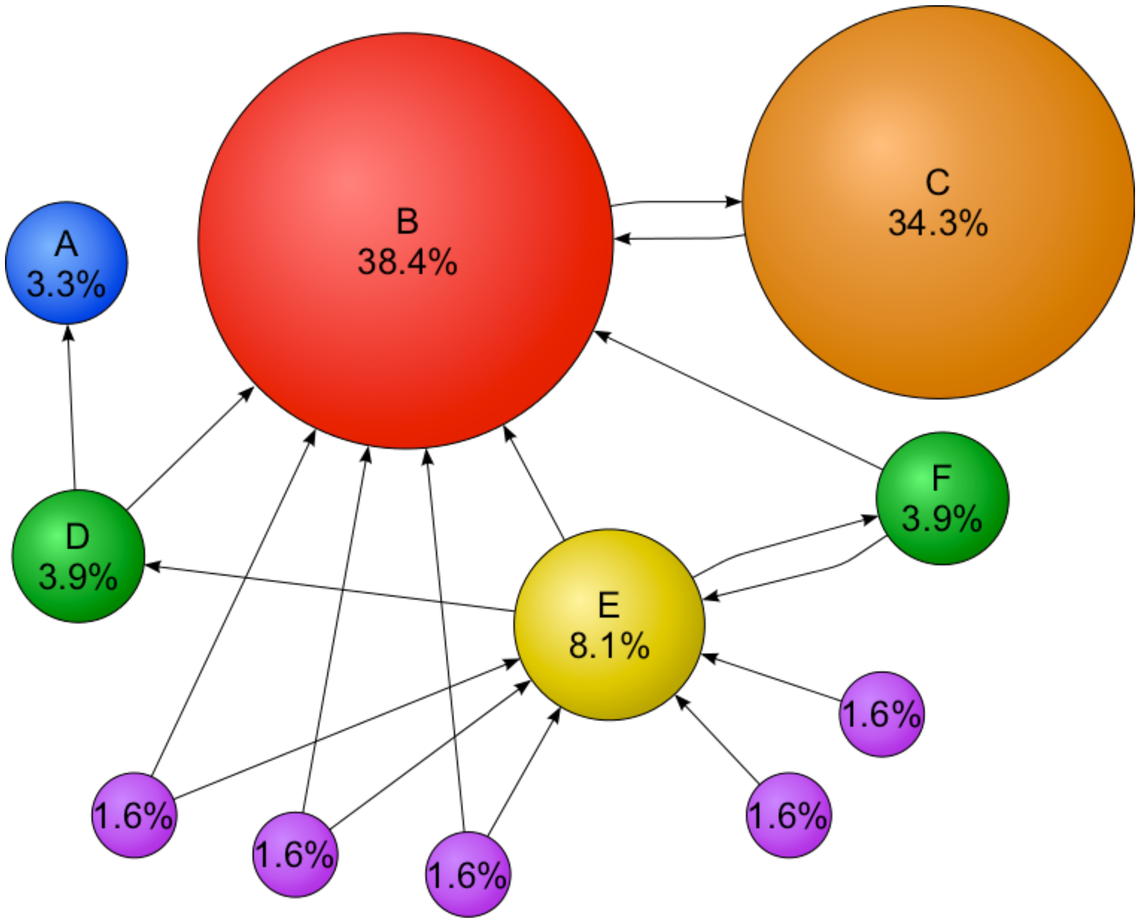
It uses a single YAML file



```
hdfs:
  image: sequenceiq/hadoop-docker:2.4.1
  command: /etc/bootstrap.sh -d -bash
mazerunner:
  image: kbastani/neo4j-graph-analytics:latest
  links:
   - hdfs
graphdb:
  image: kbastani/docker-neo4j:latest
  ports:
   - "7474:7474"
   - "1337:1337"
  volumes:
   - /opt/data
  links:
   - mazerunner
   - hdfs
```

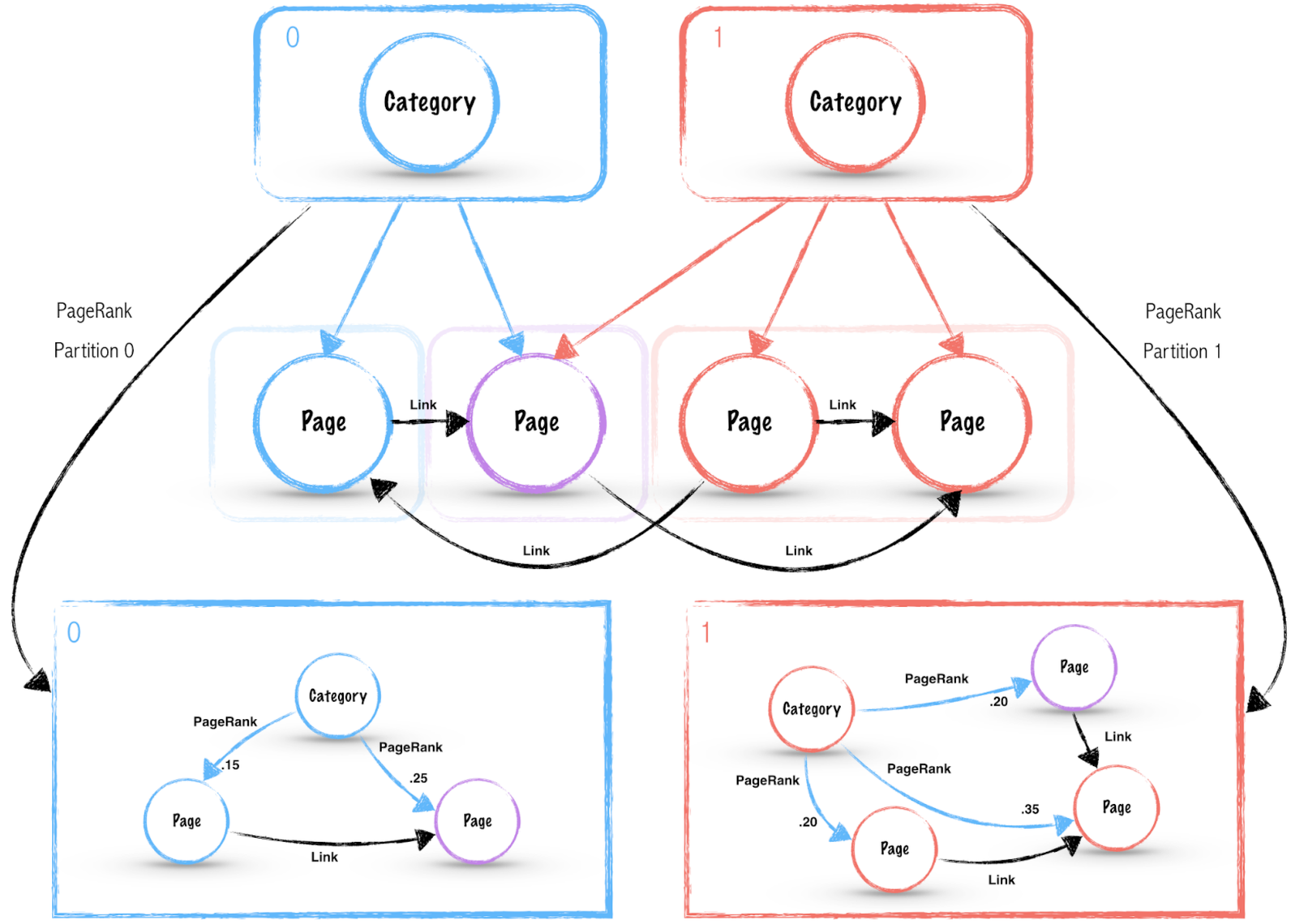


# PageRank



# Distributed PageRank




0
Category
1
Category
PageRank
Partition 0
PageRank
Partition 1
Page
Link
Page
Page
Link
Page
Link
Link
0
Category
PageRank
.15
PageRank
.25
Page
Page
Link
1
Category
PageRank
.20
Page
Link
PageRank
PageRank
.35
Page
.20
Page
Link

# Questions?

kennybastani.com