DataStax Meetups

DATASTAX

Make your data speak

Hands-on Codelab

cassandra

Apache Cassandra

3 hours, intermediate level + dinner break

An initiative by:

ITALIAN ASSOCIATION FOR MACHINE LEARNING

Powered by:

SOURCESENSE
Open Solutions for your Value

Live on Twitch !

Agenda

A – What is Apache Cassandra and why do you care ?

1. Getting starting with Apache Cassandra™ and use Cases
2. CodeLab : *Getting Started with Apache Cassandra*
3. Apache Spark™ and DataStax Enterprise Analytics

B – Machine Learning with DataStax Enterprise

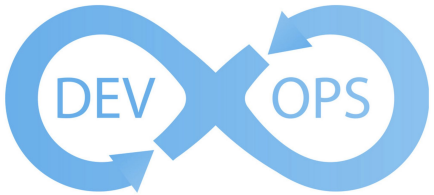
1. CodeLab : *Clustering with K-means*
2. CodeLab : *Classification with Naïve Bayes*
3. CodeLab : *Regression and Classific. with RandomForest*
4. CodeLab : *Recommendation with FP-Growth*
5. CodeLab : *Recommendation with Collaborating Filtering*

C – Resources and next steps

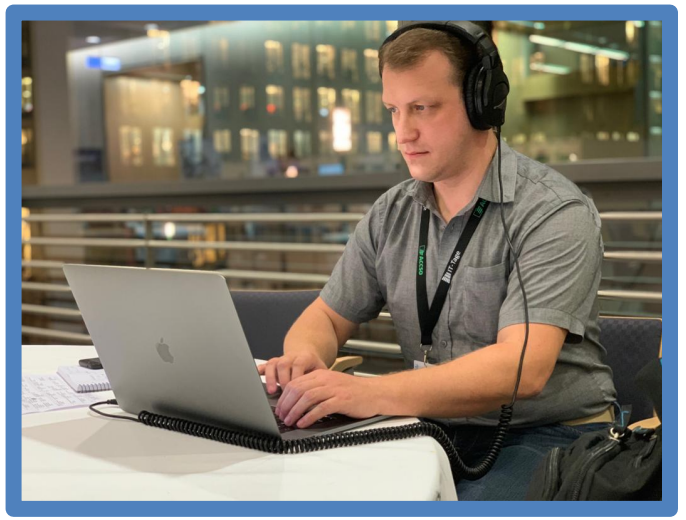
DATASTAX

Your Instructors

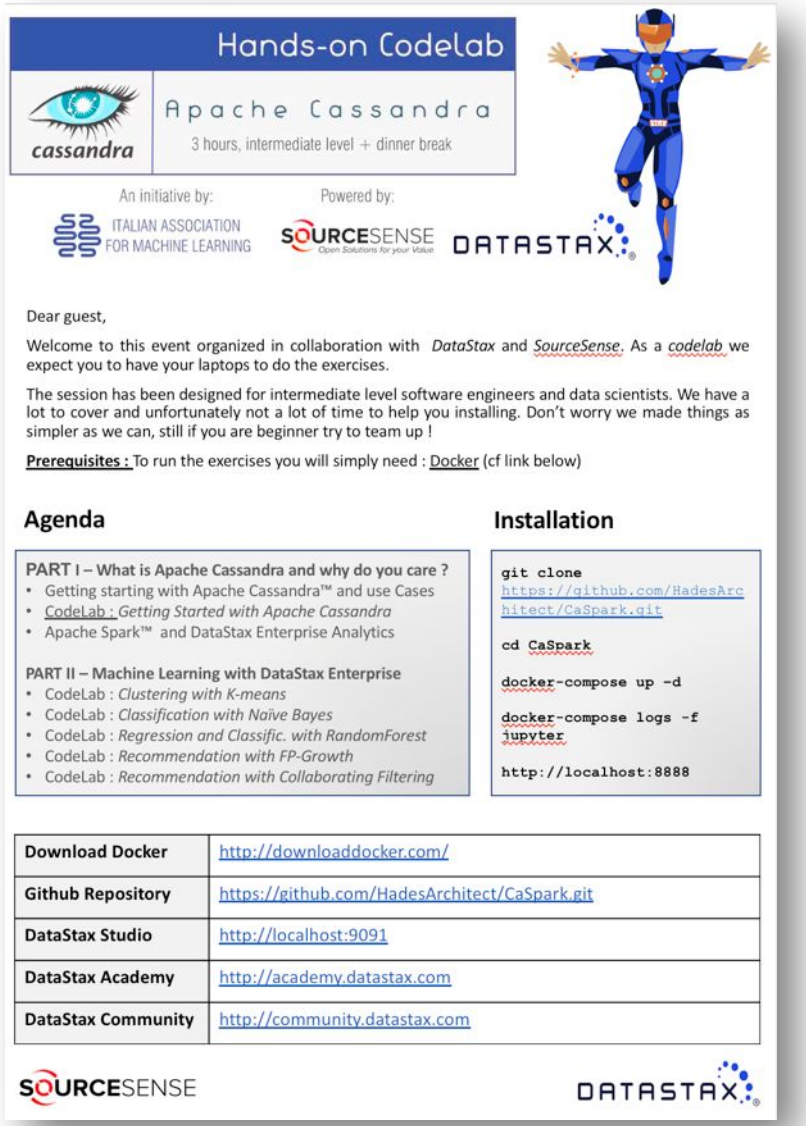
Cedrick Lunven



Aleks Volochnev



Before Starting



Hands-on Codelab

Apache Cassandra

3 hours, intermediate level + dinner break

An initiative by: ITALIAN ASSOCIATION FOR MACHINE LEARNING

Powered by: SOURCESENSE DATASTAX

Dear guest,

Welcome to this event organized in collaboration with *DataStax* and *SourceSense*. As a *codelab* we expect you to have your laptops to do the exercises.

The session has been designed for intermediate level software engineers and data scientists. We have a lot to cover and unfortunately not a lot of time to help you installing. Don't worry we made things as simpler as we can, still if you are beginner try to team up !

Prerequisites : To run the exercises you will simply need : Docker (cf link below)

Agenda

PART I – What is Apache Cassandra and why do you care ?
- Getting starting with Apache Cassandra™ and use Cases
- CodeLab : *Getting Started with Apache Cassandra*
- Apache Spark™ and DataStax Enterprise Analytics

PART II – Machine Learning with DataStax Enterprise
- CodeLab : *Clustering with K-means*
- CodeLab : *Classification with Naïve Bayes*
- CodeLab : *Regression and Classific. with RandomForest*
- CodeLab : *Recommendation with FP-Growth*
- CodeLab : *Recommendation with Collaborating Filtering*

Installation

```
git clone
https://github.com/HadesArchitect/CaSpark.git

cd CaSpark

docker-compose up -d

docker-compose logs -f
jupyter

http://localhost:8888
```

Download Docker	http://downloaddocker.com/
Github Repository	https://github.com/HadesArchitect/CaSpark.git
DataStax Studio	http://localhost:9091
DataStax Academy	http://academy.datastax.com
DataStax Community	http://community.datastax.com

SOURCESENSE DATASTAX

- You should have one of those sheet.
- Please execute the Installation steps **as soon as possible.** This will download a few docker images that can take some time !

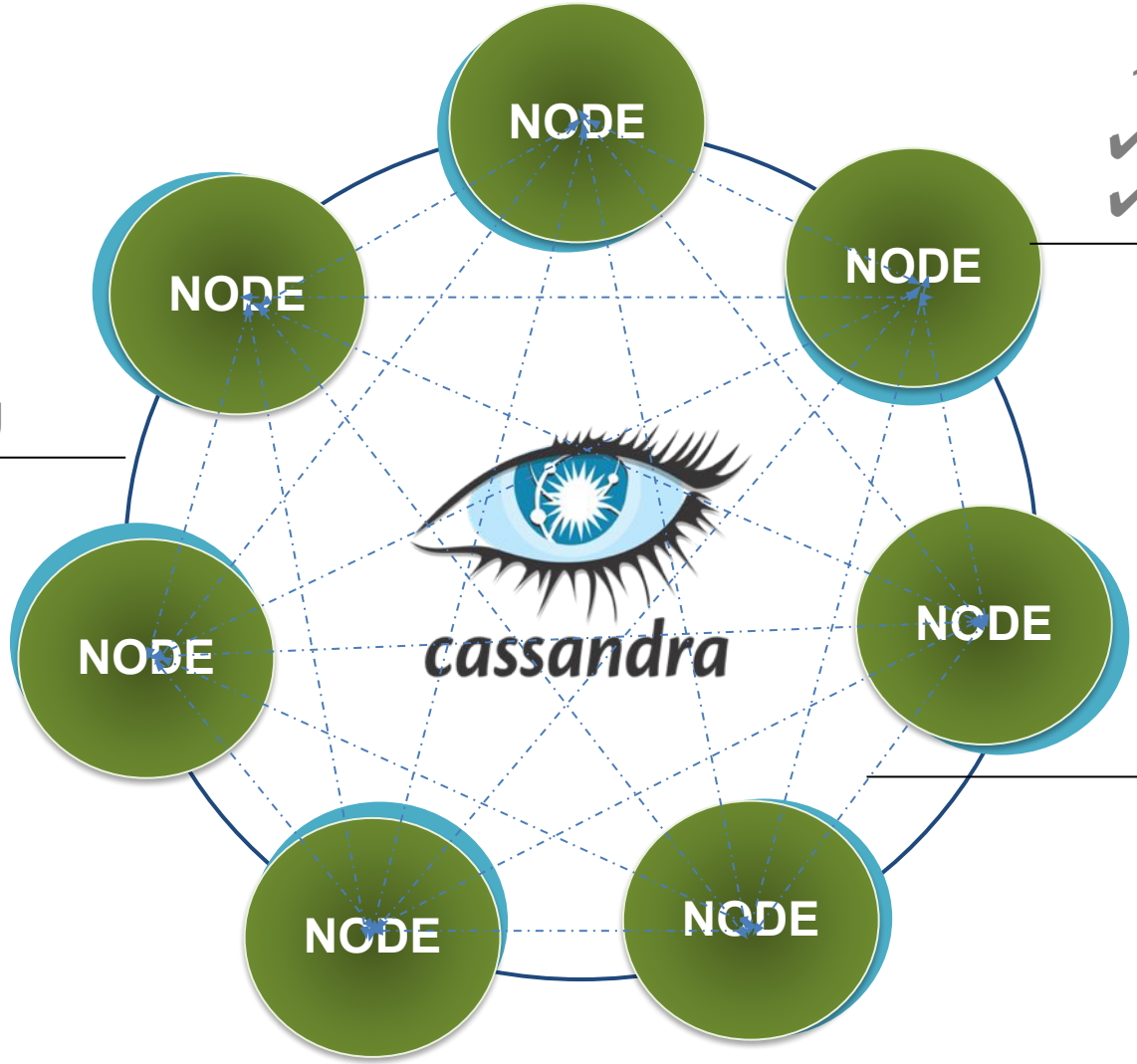
DATASTAX

DataStax Meetup

DATASTAX

Getting Started with Apache Cassandra

Apache Cassandra™ = Distributed NoSQL Database



1 Installation = 1 NODE
- ✔ Capacity: ± 1TB
- ✔ Throughput: 3000 Tx/sec/core

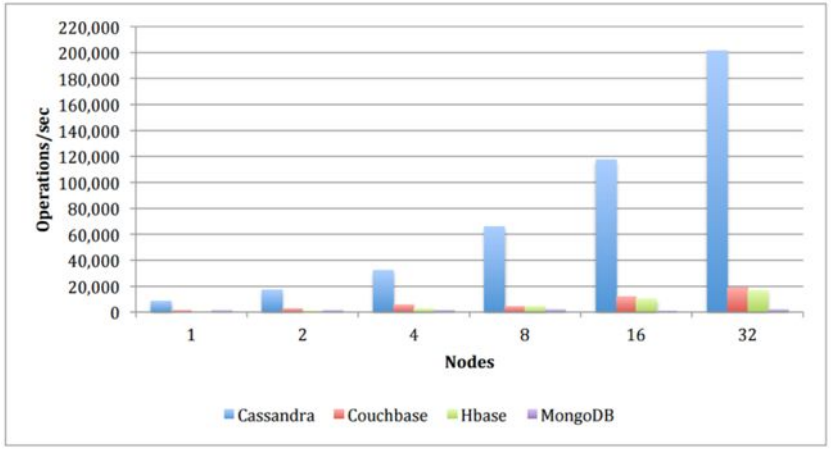
DataCenter | Ring

Communication:
- ✔ Gossiping

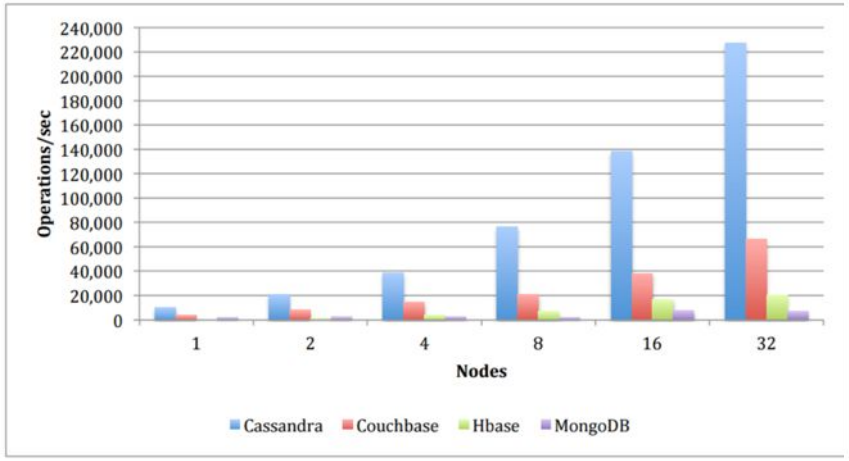
DATASTAX

Linear Scalability

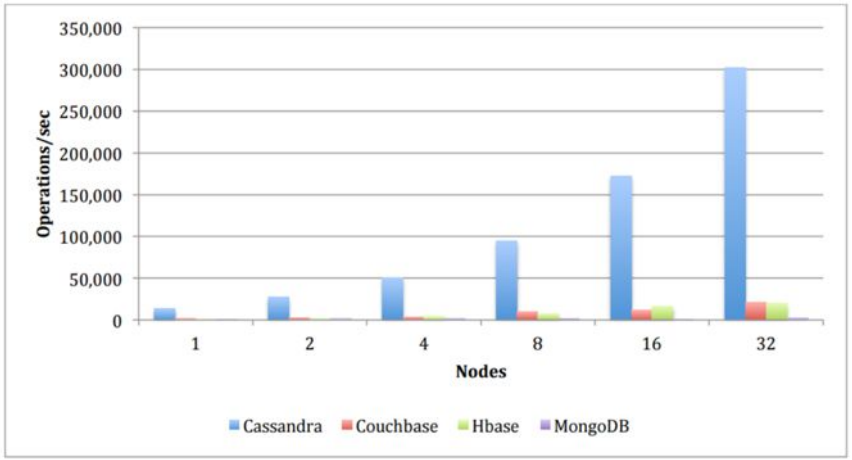
Read-Modify-Write Workload



Read-mostly Workload



Balanced Read/Write Mix



- **Need More Capacity ? Add new nodes**
- **Need more Throughput ? Add new nodes**

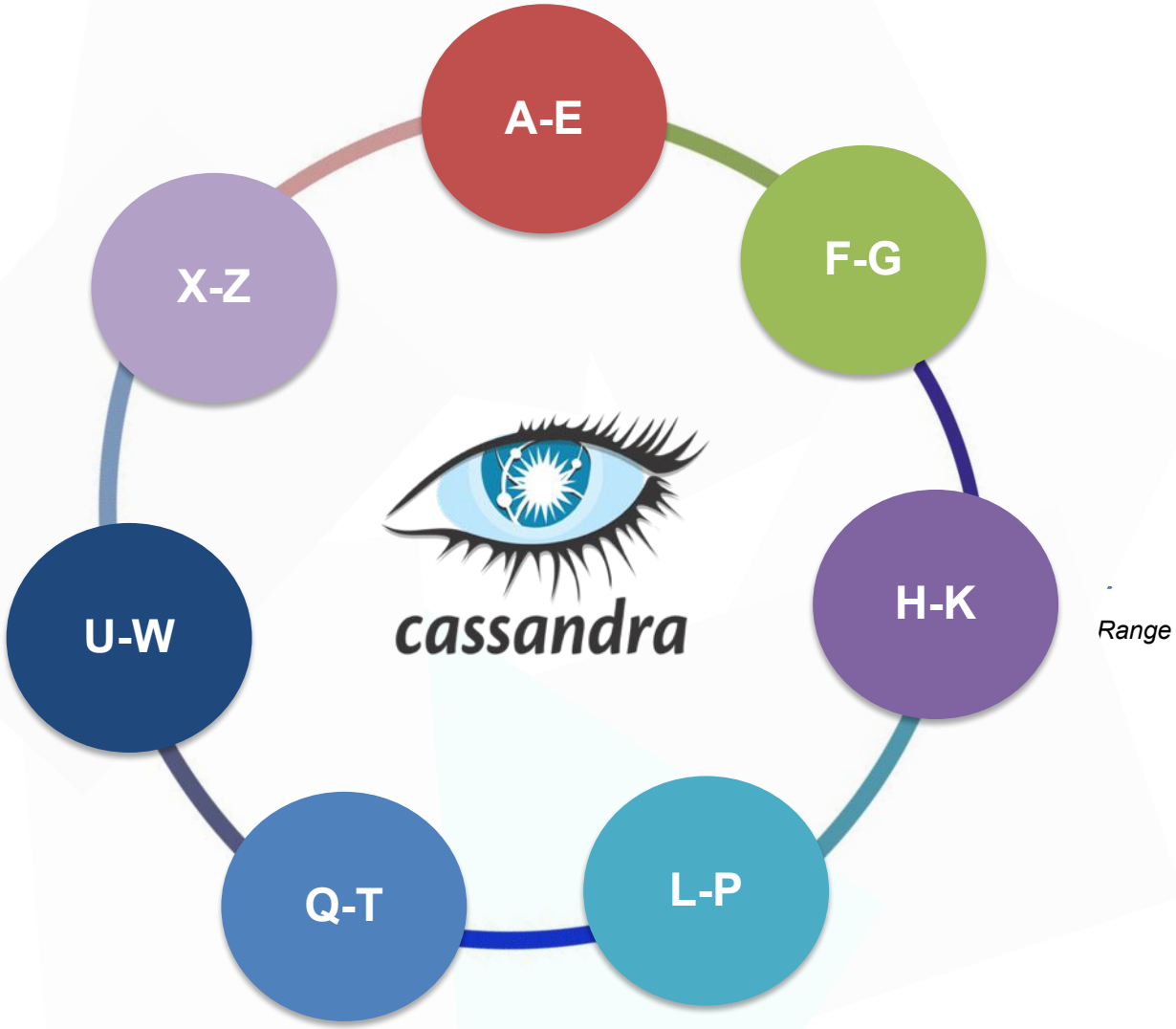
DATASTAX

Data is Distributed

Country	City	Habitant
USA	New York	8.000.000
USA	Los Angeles	4.000.000
FR	Paris	2.230.000
DE	Berlin	3.350.000
UK	London	9.200.000
AU	Sydney	4.900.000
DE	Nuremberg	500.000
CA	Toronto	6.200.000
CA	Montreal	4.200000
FR	Toulouse	1.100.000
JP	Tokyo	37.430.000
IN	Mumbai	20.200.000

Partition Key

Data is Distributed



Range

Data is *Evenly*-distributed

CO	City	Habitant
AU	Sydney	4.900.000
CA	Toronto	6.200.000
CA	Montreal	4.200.000
DE	Berlin	3.350.000
DE	Nuremberg	500.000

Partition Key

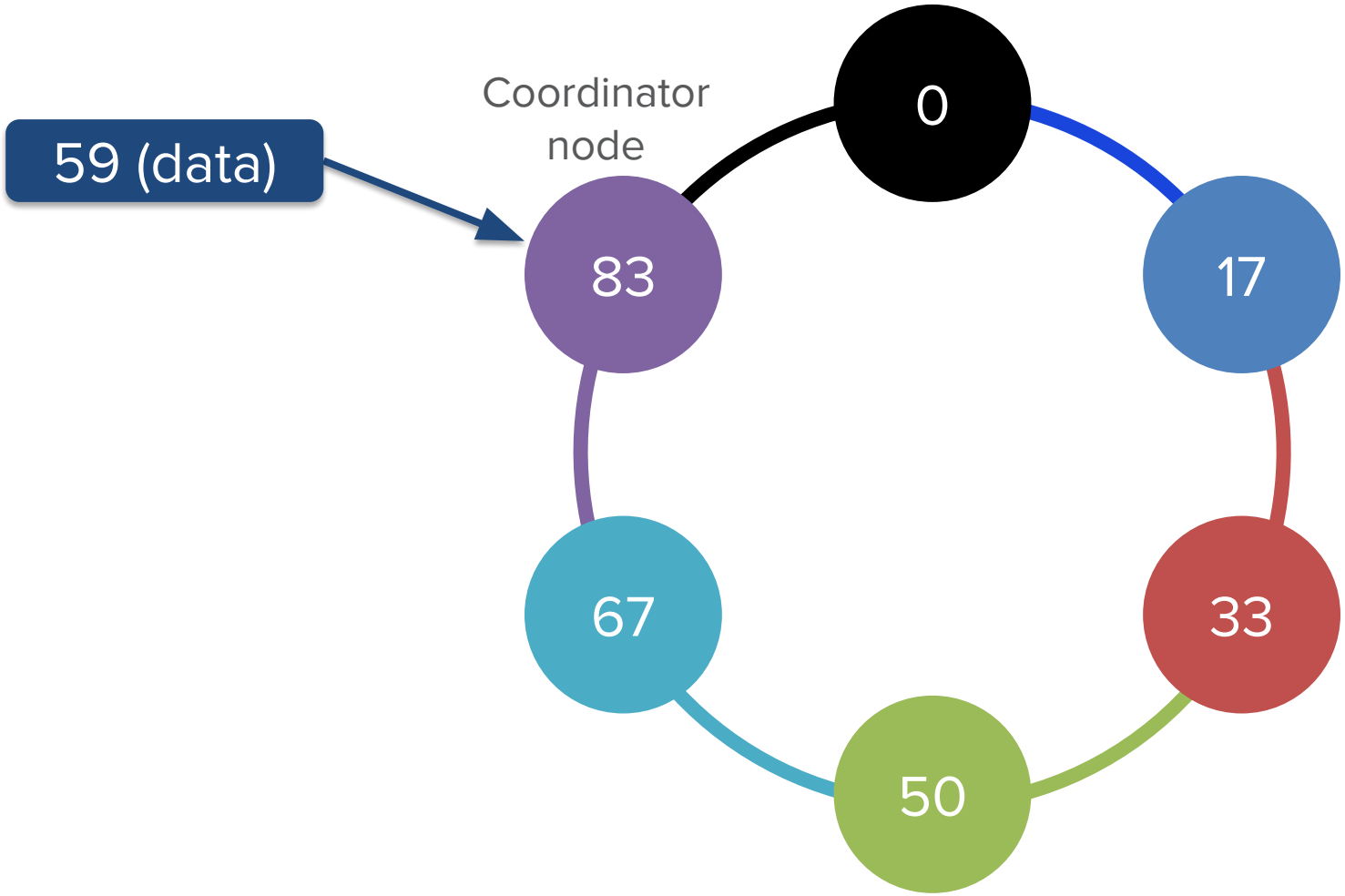
Partitioner

Hashing Function

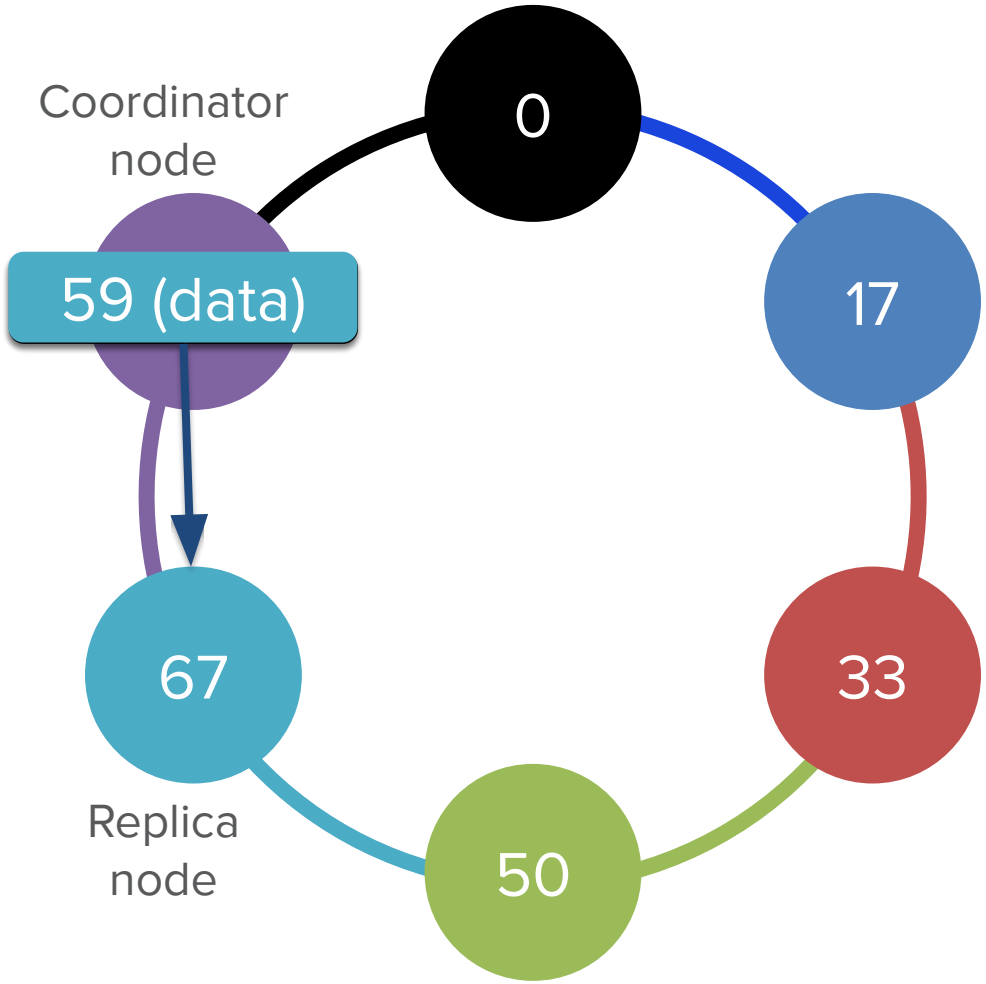
CO	City	Habitant
59	Sydney	4.900.000
12	Toronto	6.200.000
12	Montreal	4.200.000
45	Berlin	3.350.000
45	Nuremberg	500.000

Tokens

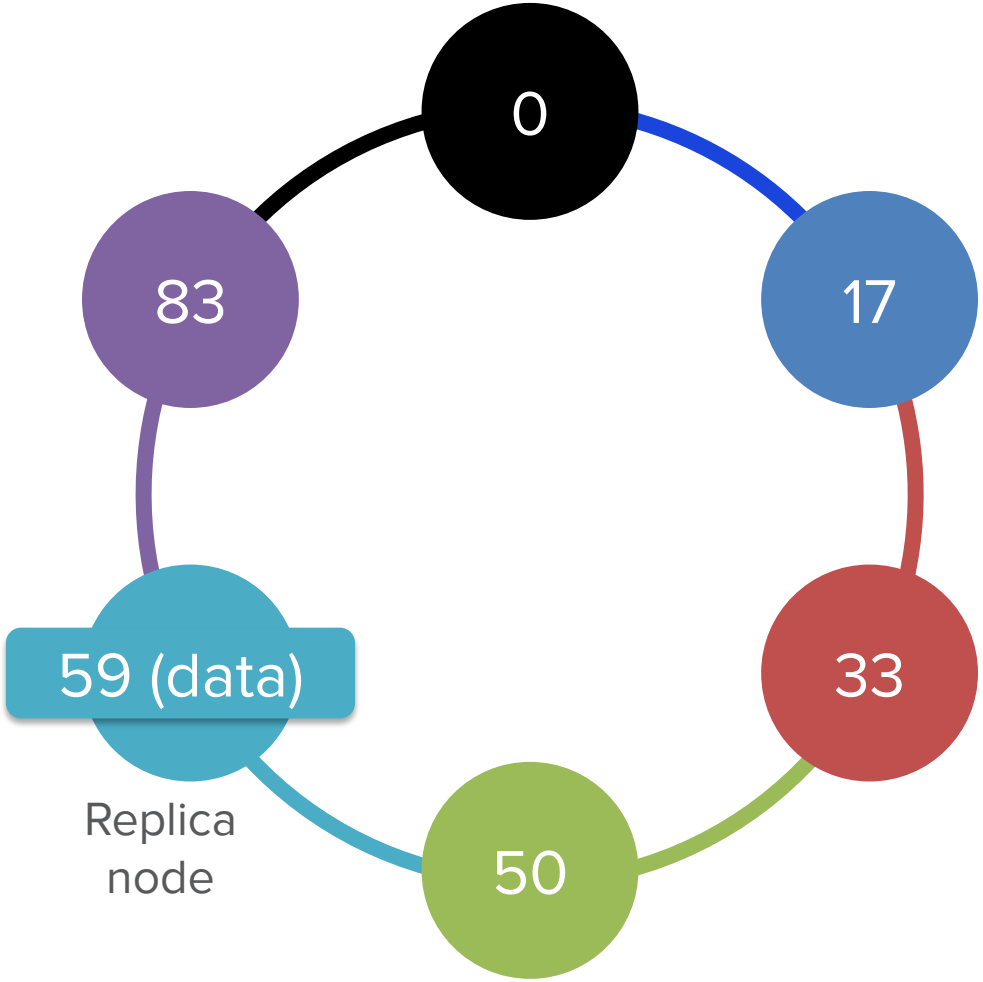
How the Ring Works



How the Ring Works

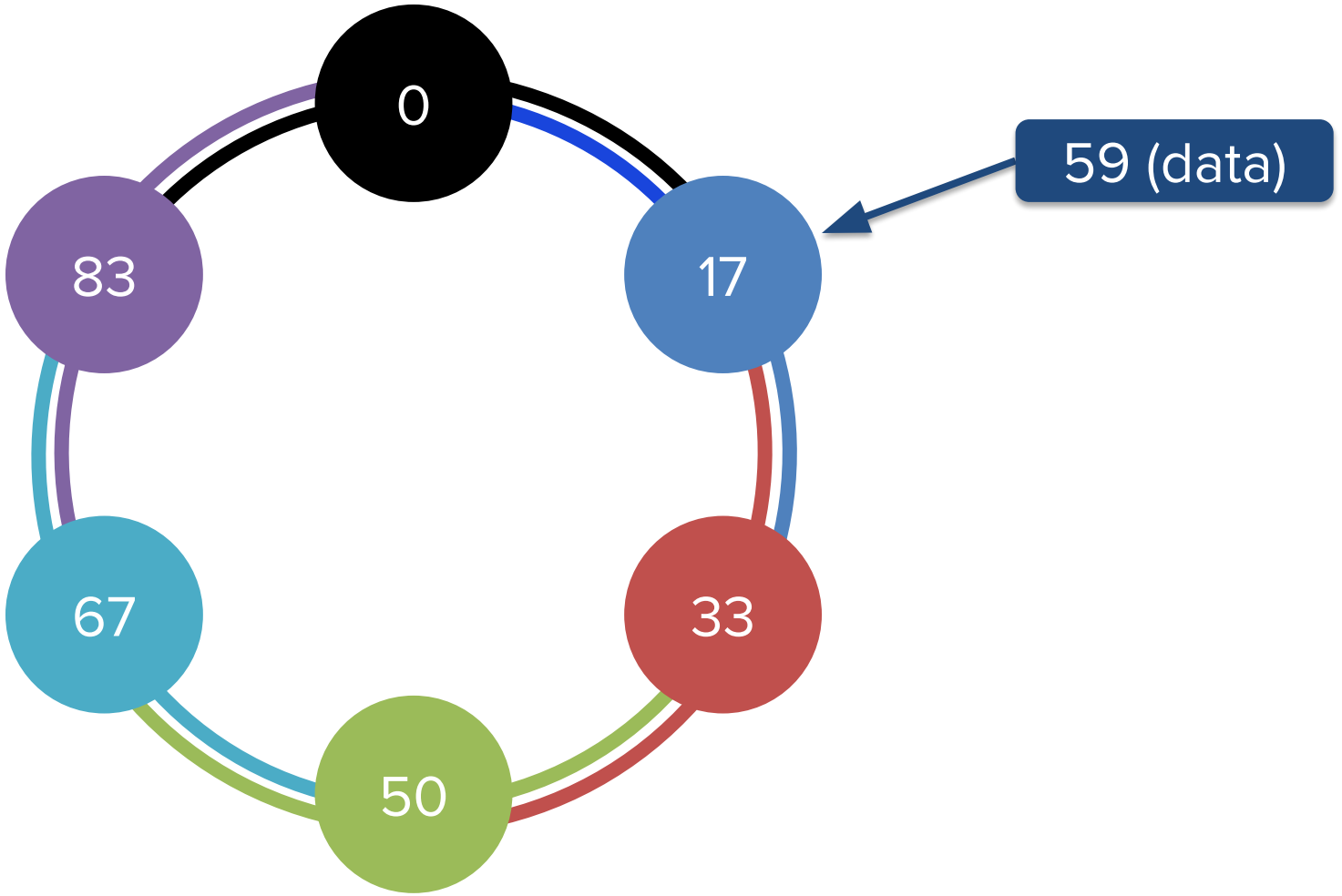


How the Ring Works



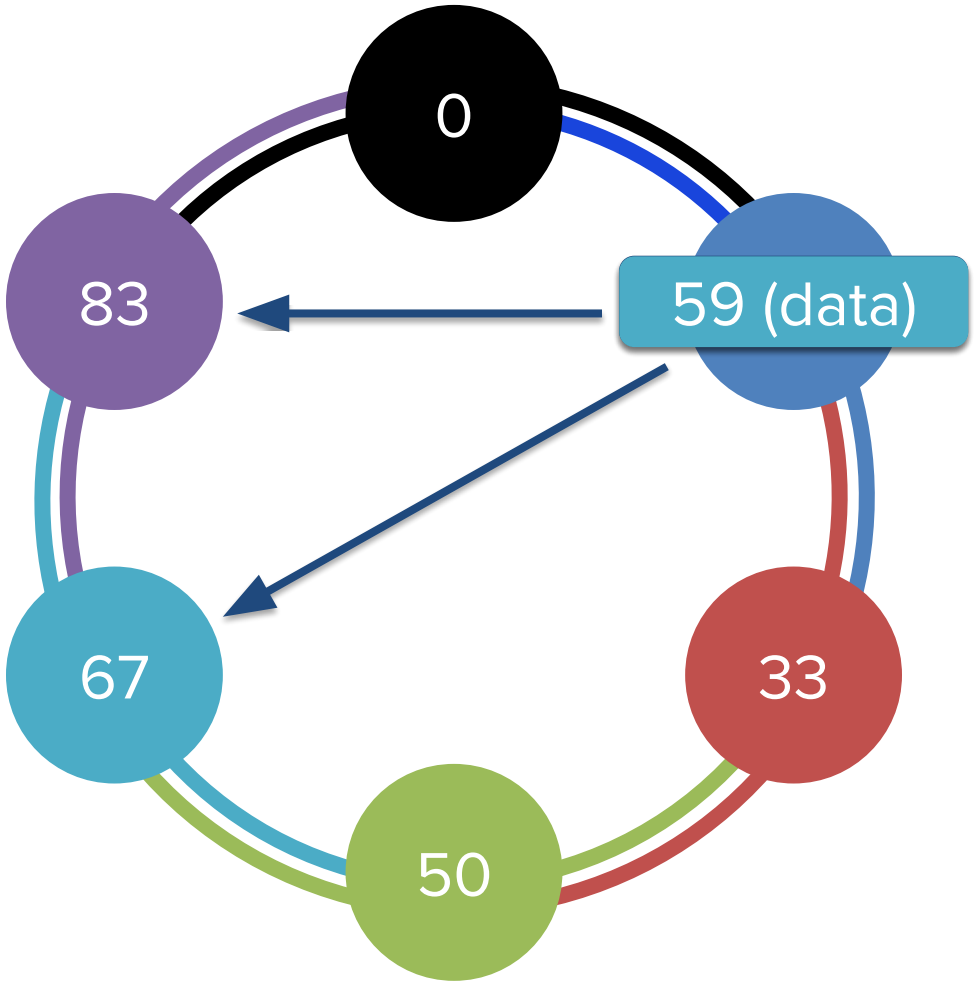
Replication within the Ring

RF = 2



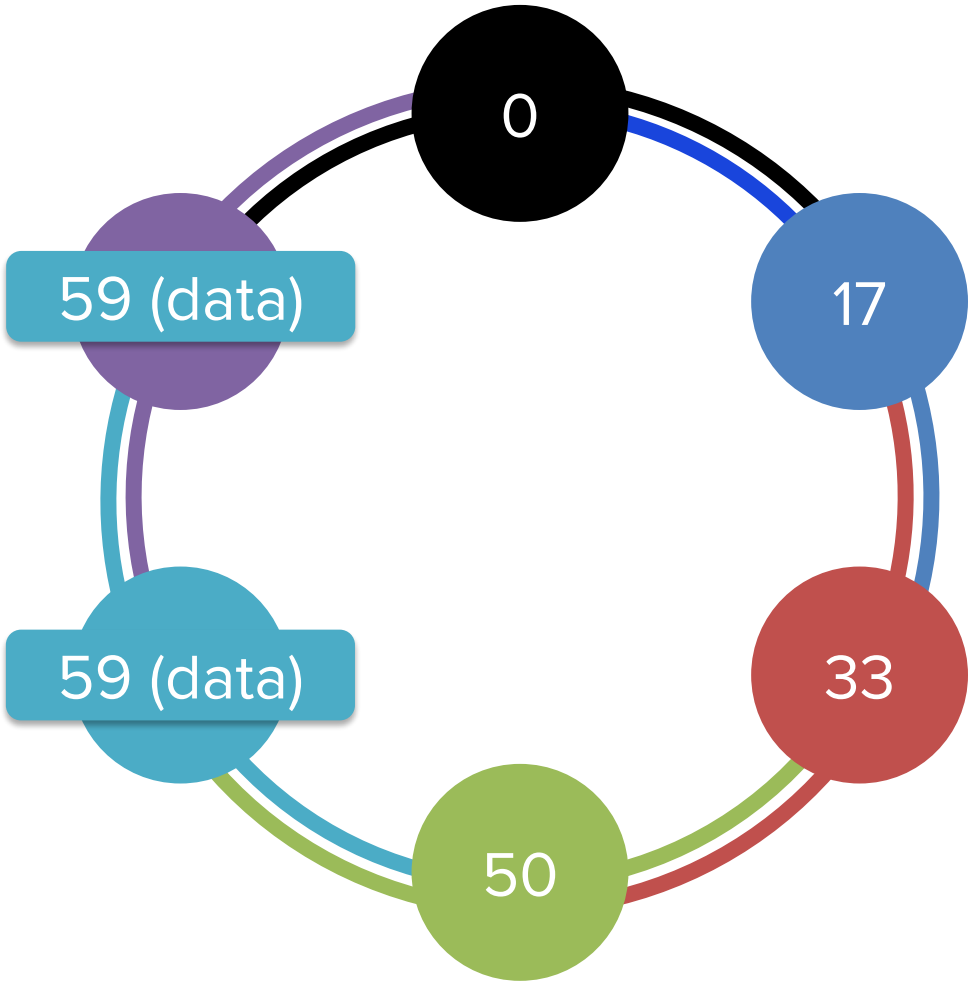
Replication within the Ring

RF = 2



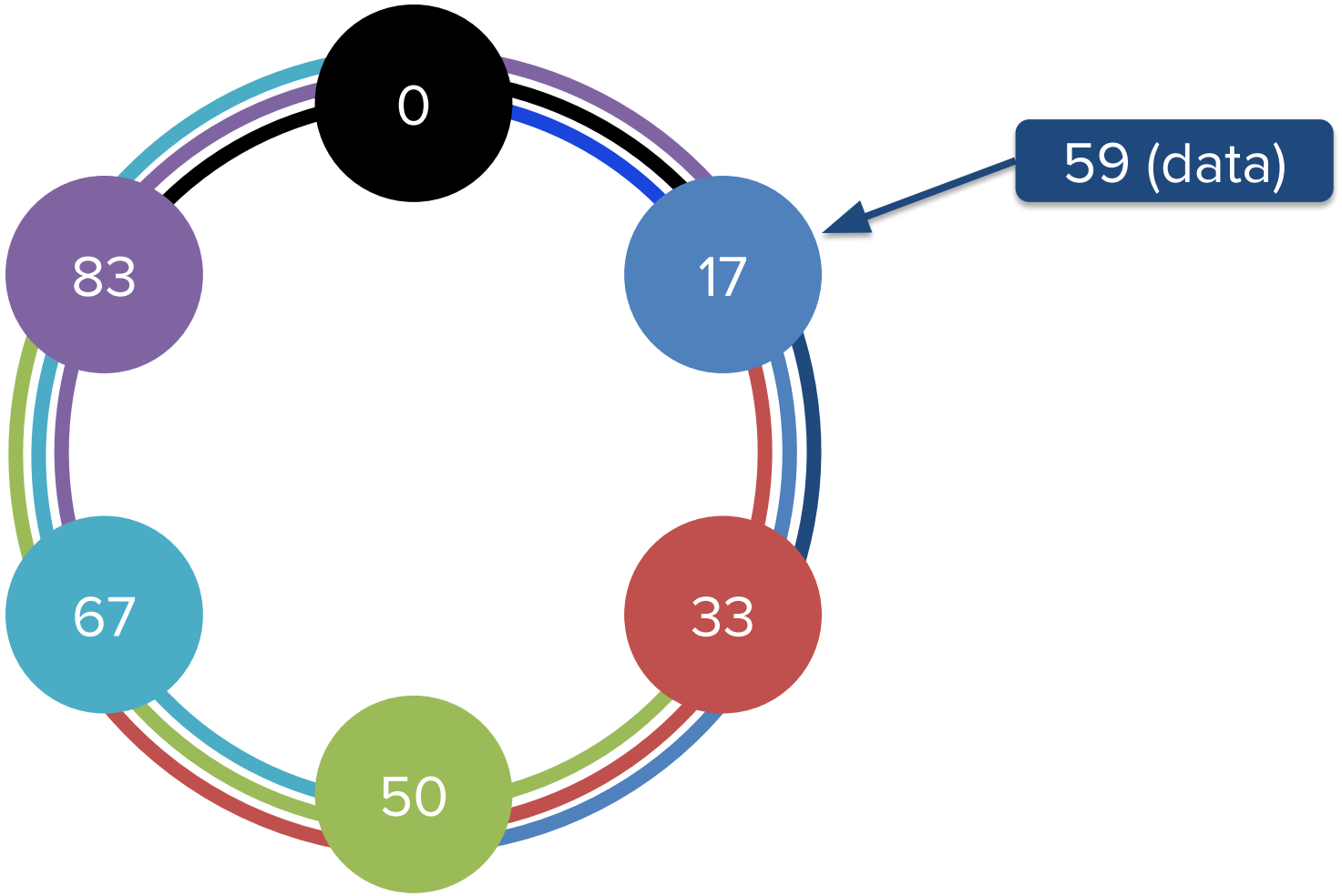
Replication within the Ring

RF = 2



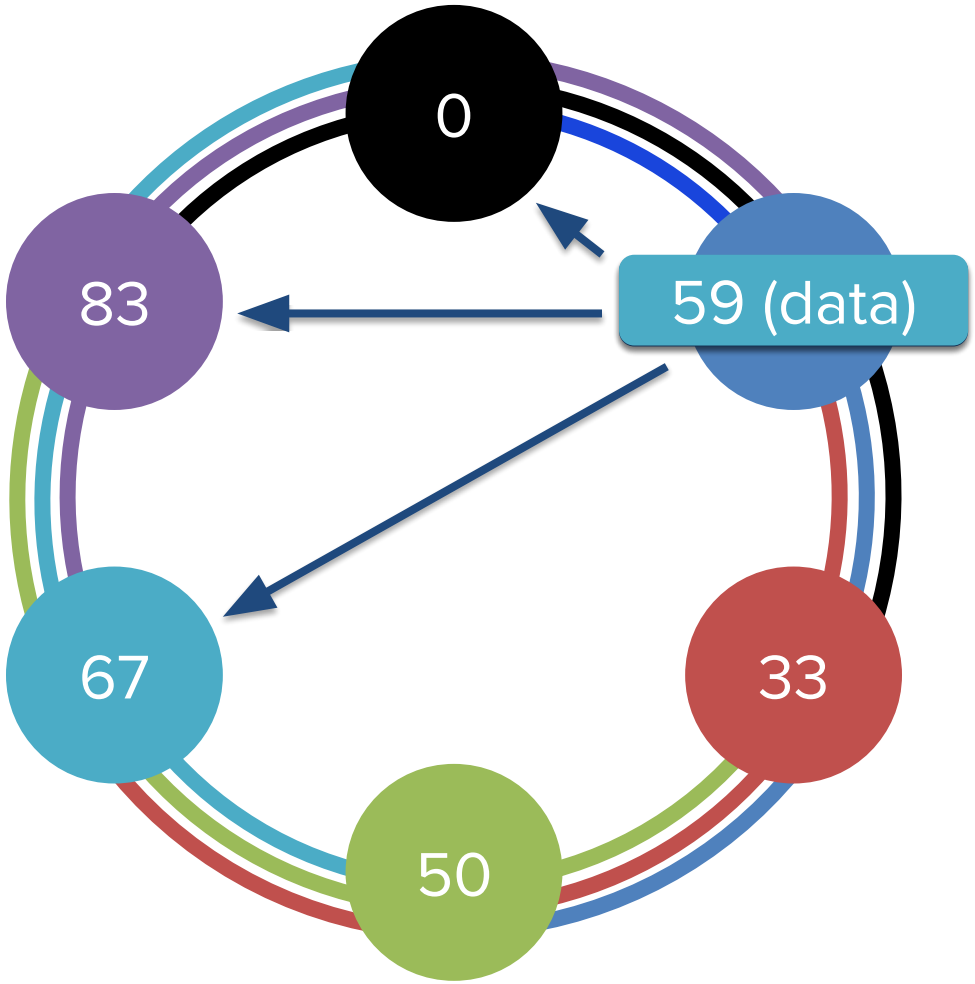
Replication within the Ring

RF = 3



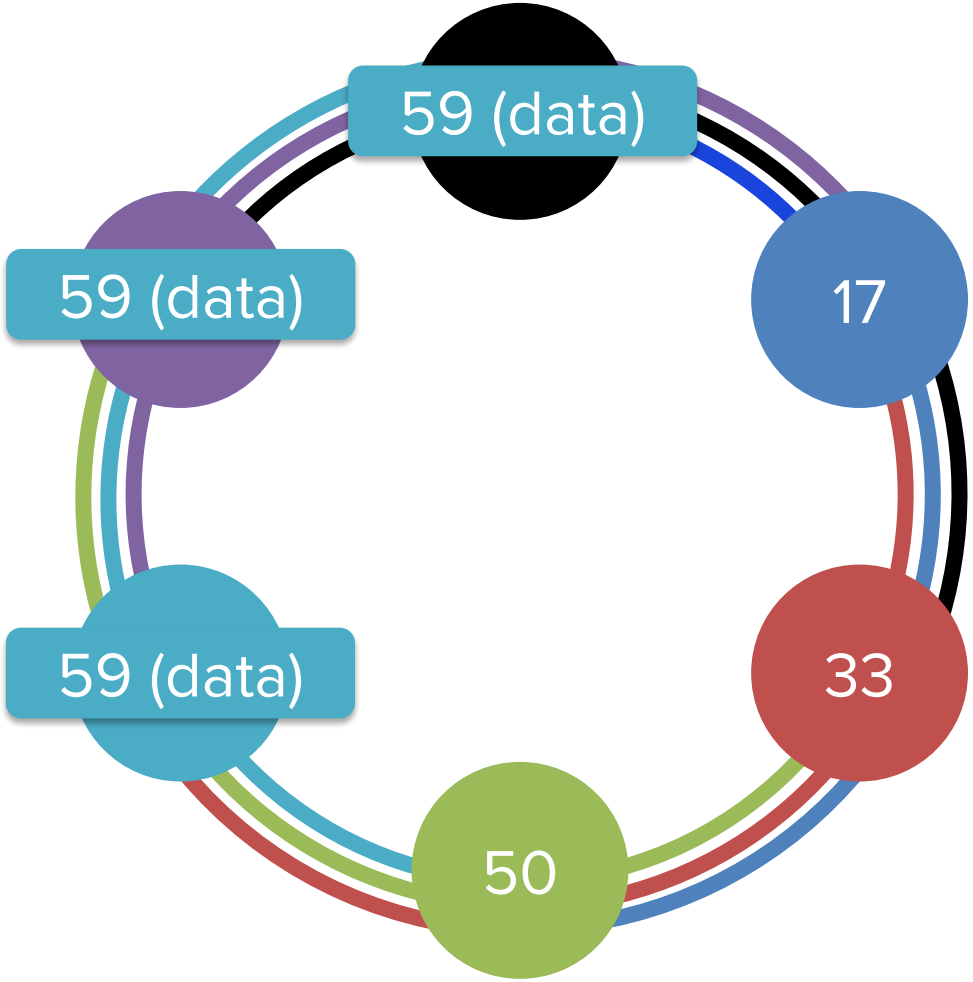
Replication within the Ring

RF = 3



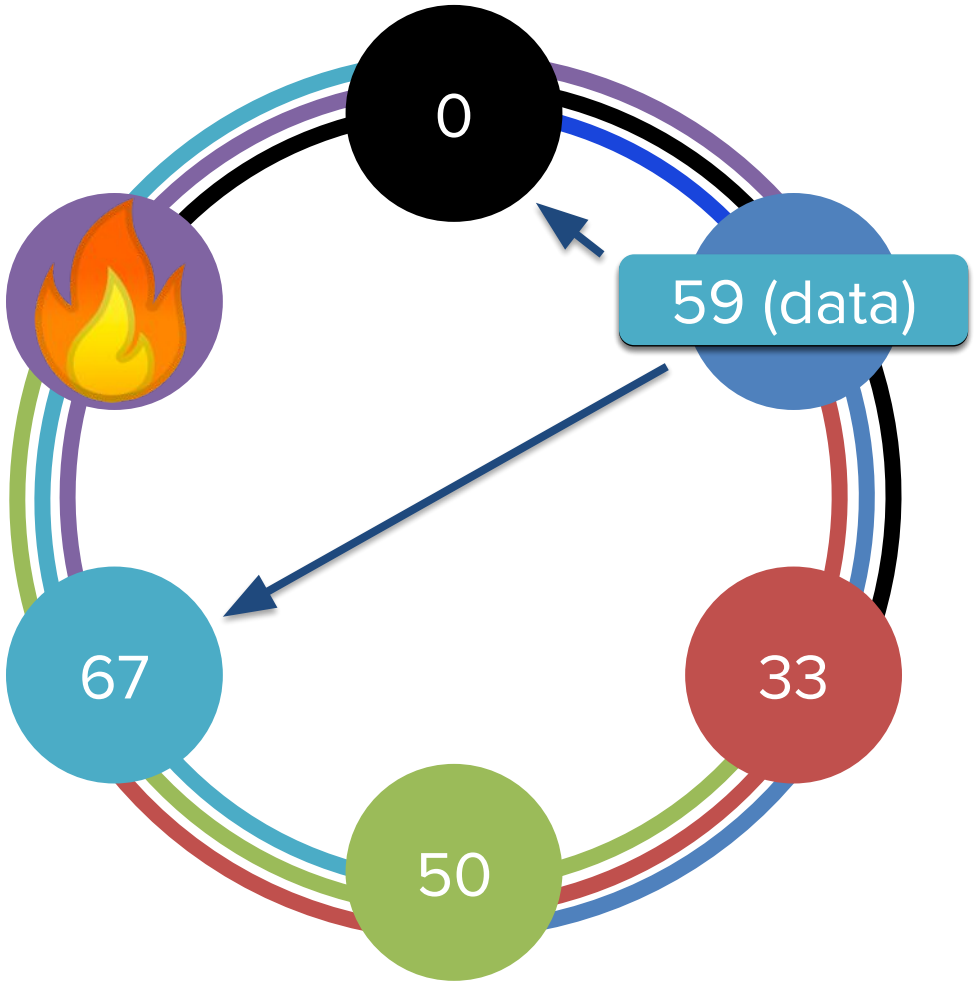
Replication within the Ring

RF = 3



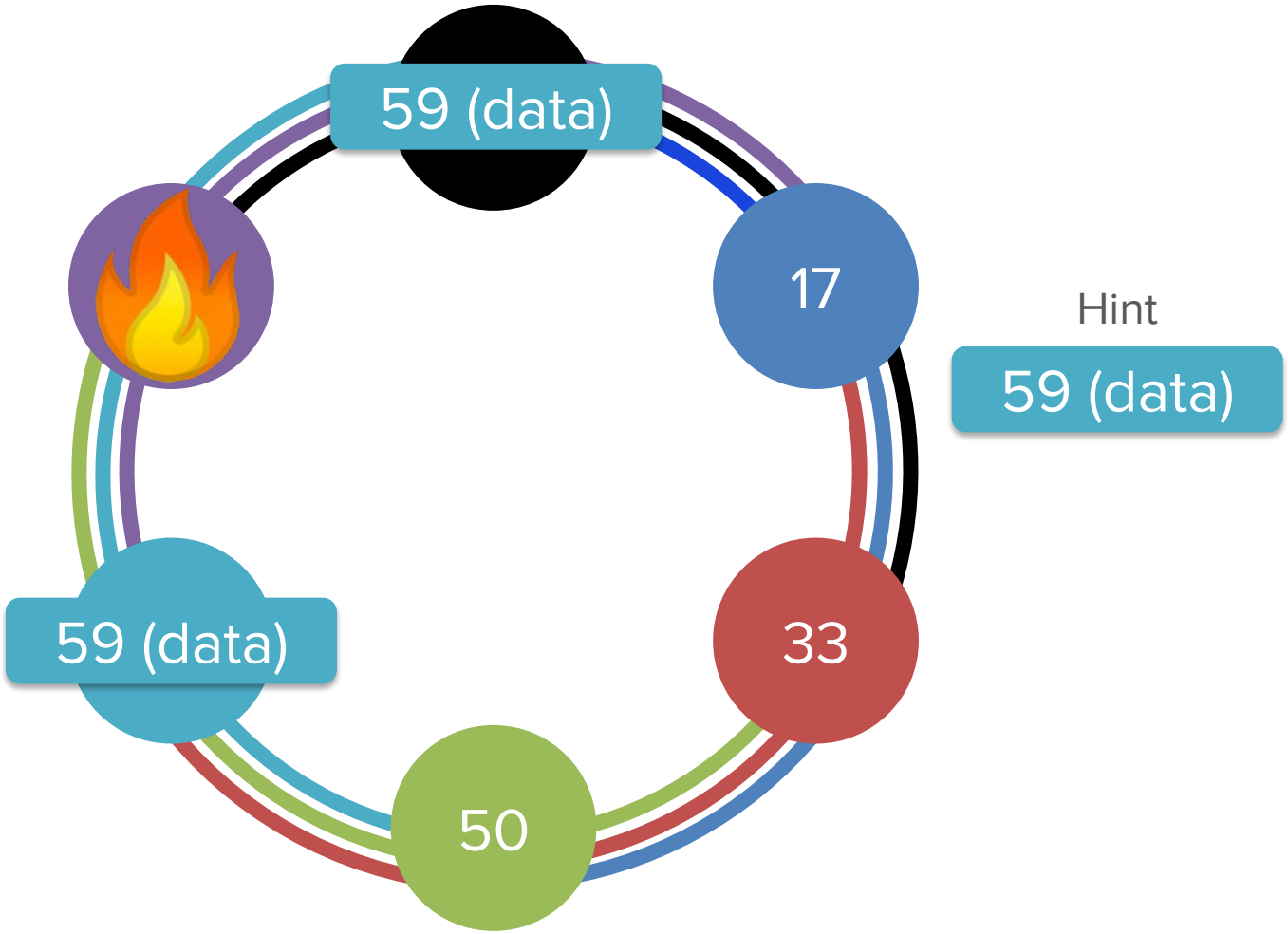
Node Failure

RF = 3



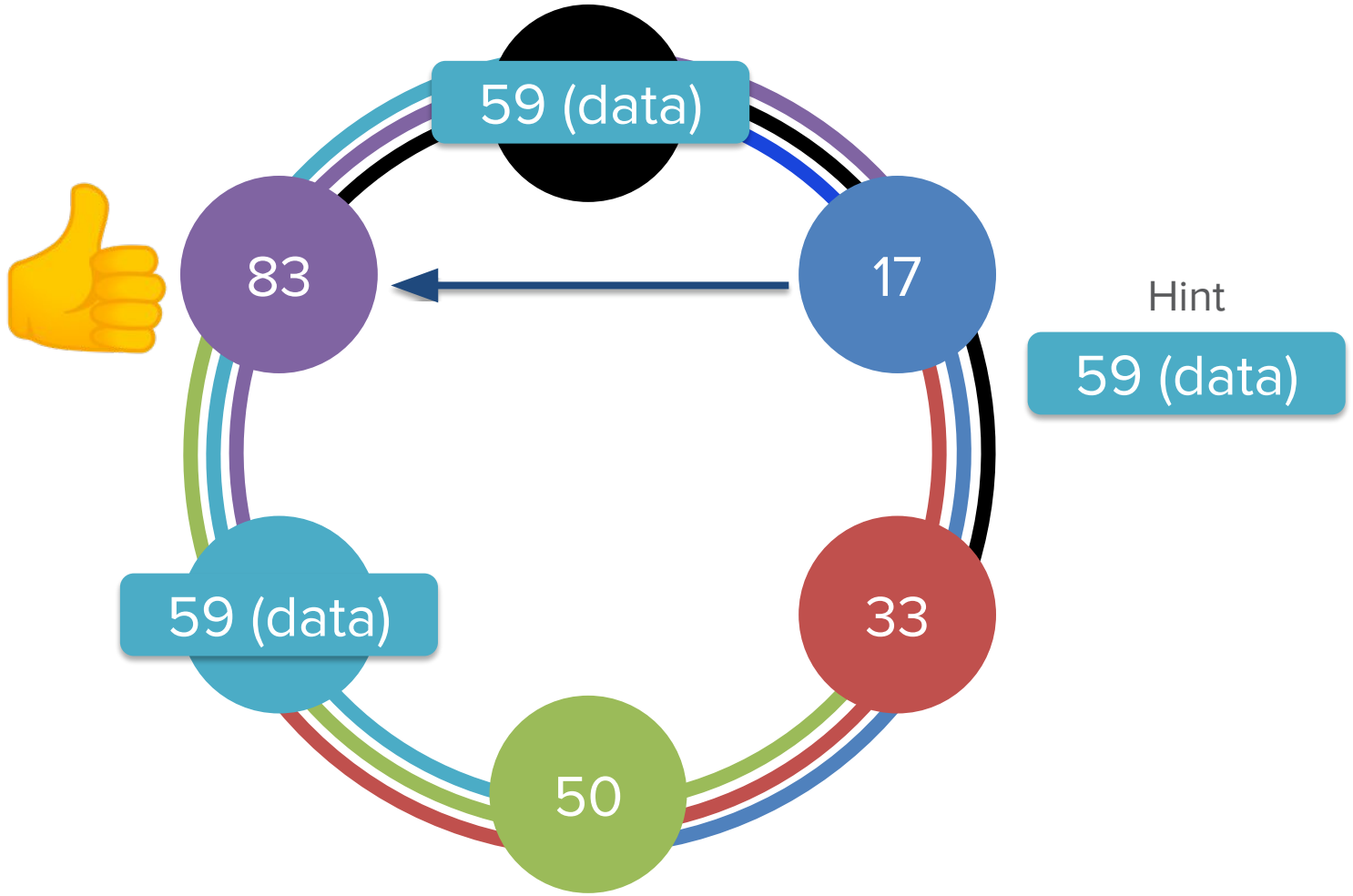
Node Failure

RF = 3



Node Failure

RF = 3

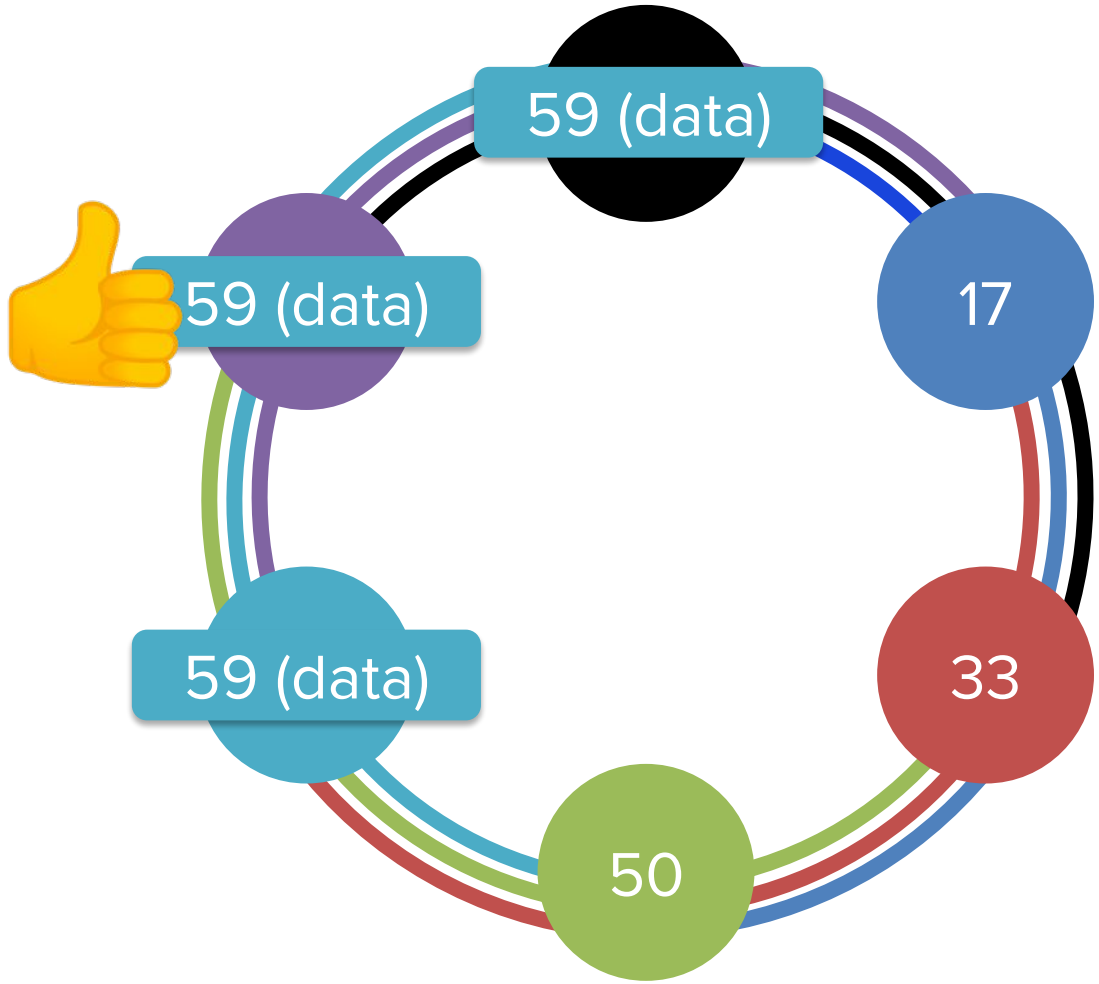


Hint

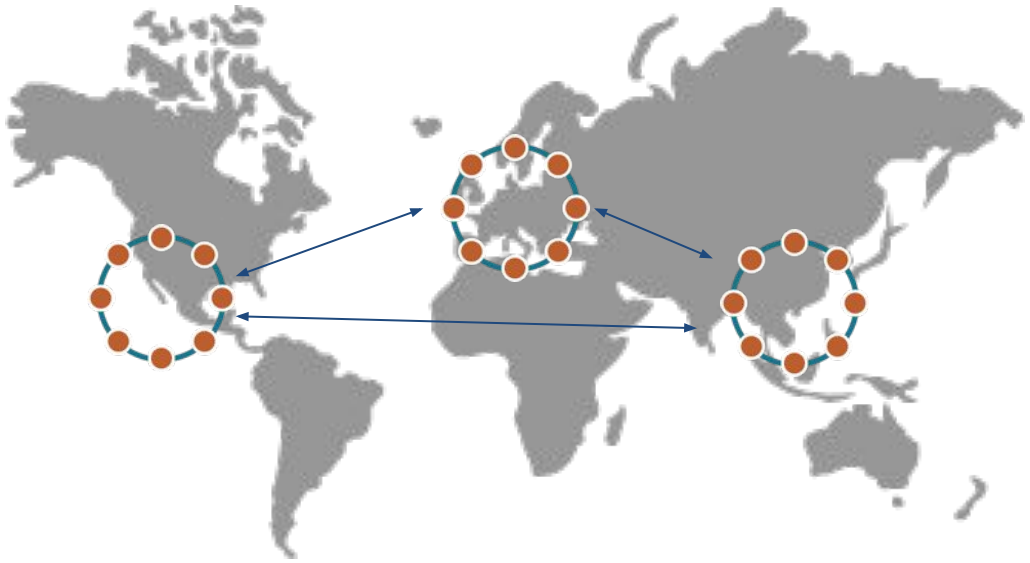
59 (data)

Node Failure – Recovered!

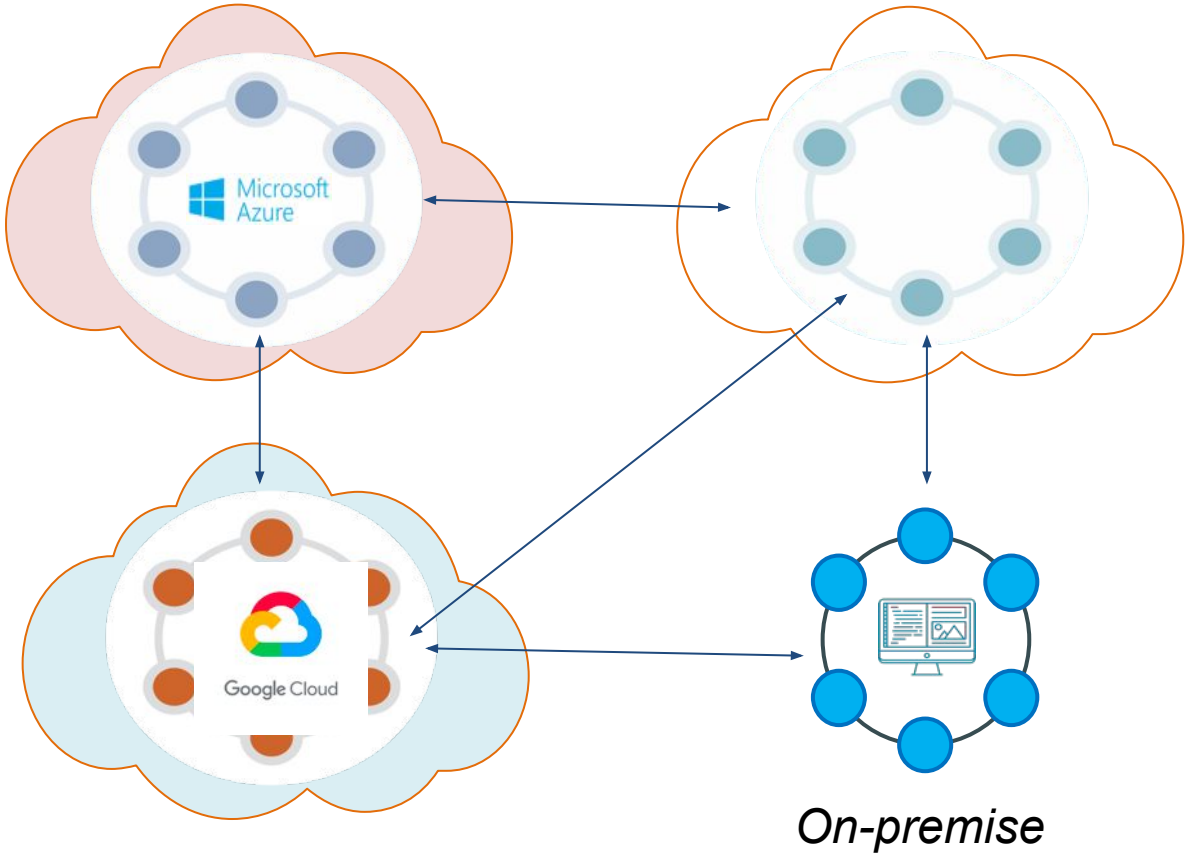
RF = 3



Data Distributed Everywhere



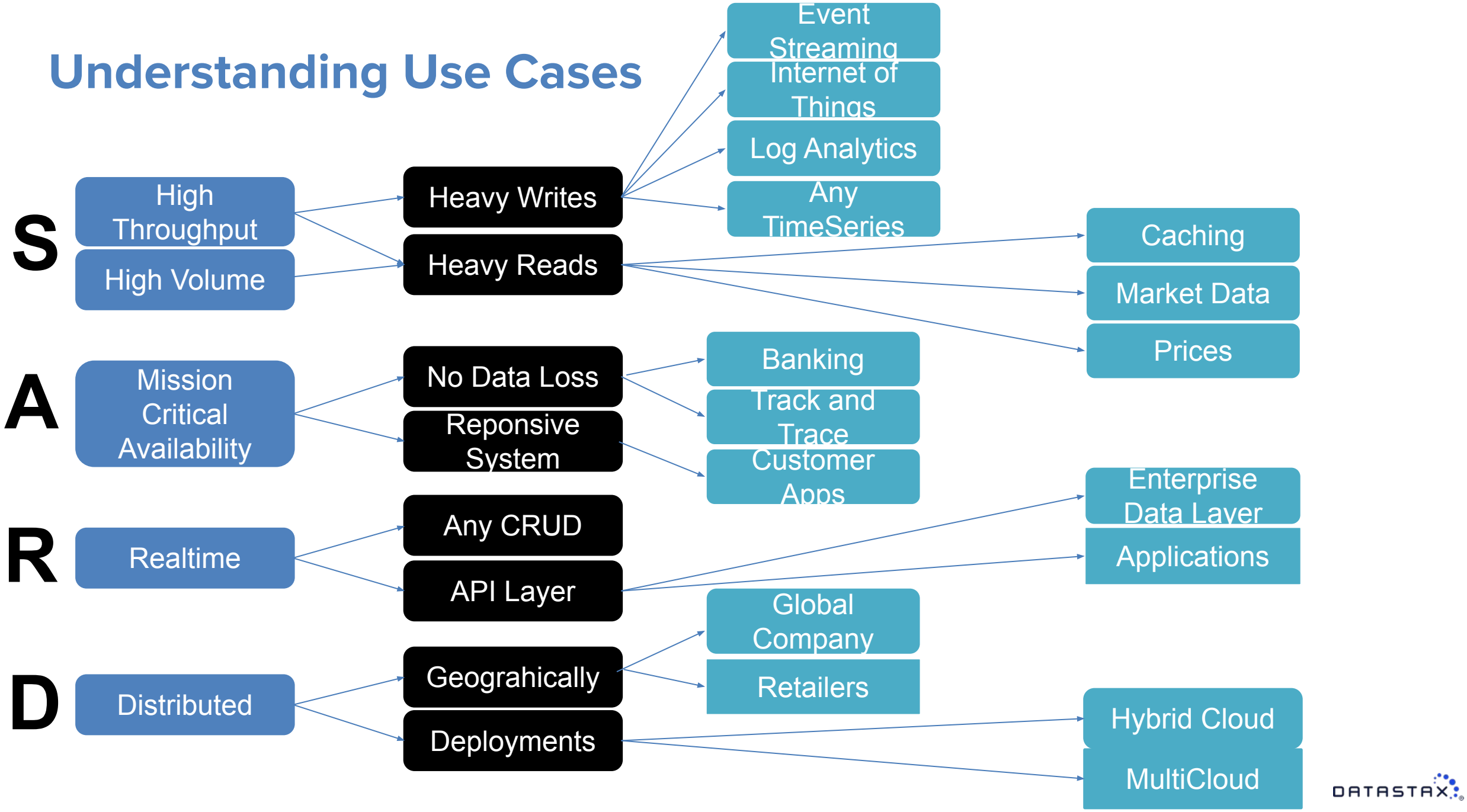
GEOGRAPHICALLY



HYBRID- MULTI CLOUD

DATASTAX

Understanding Use Cases



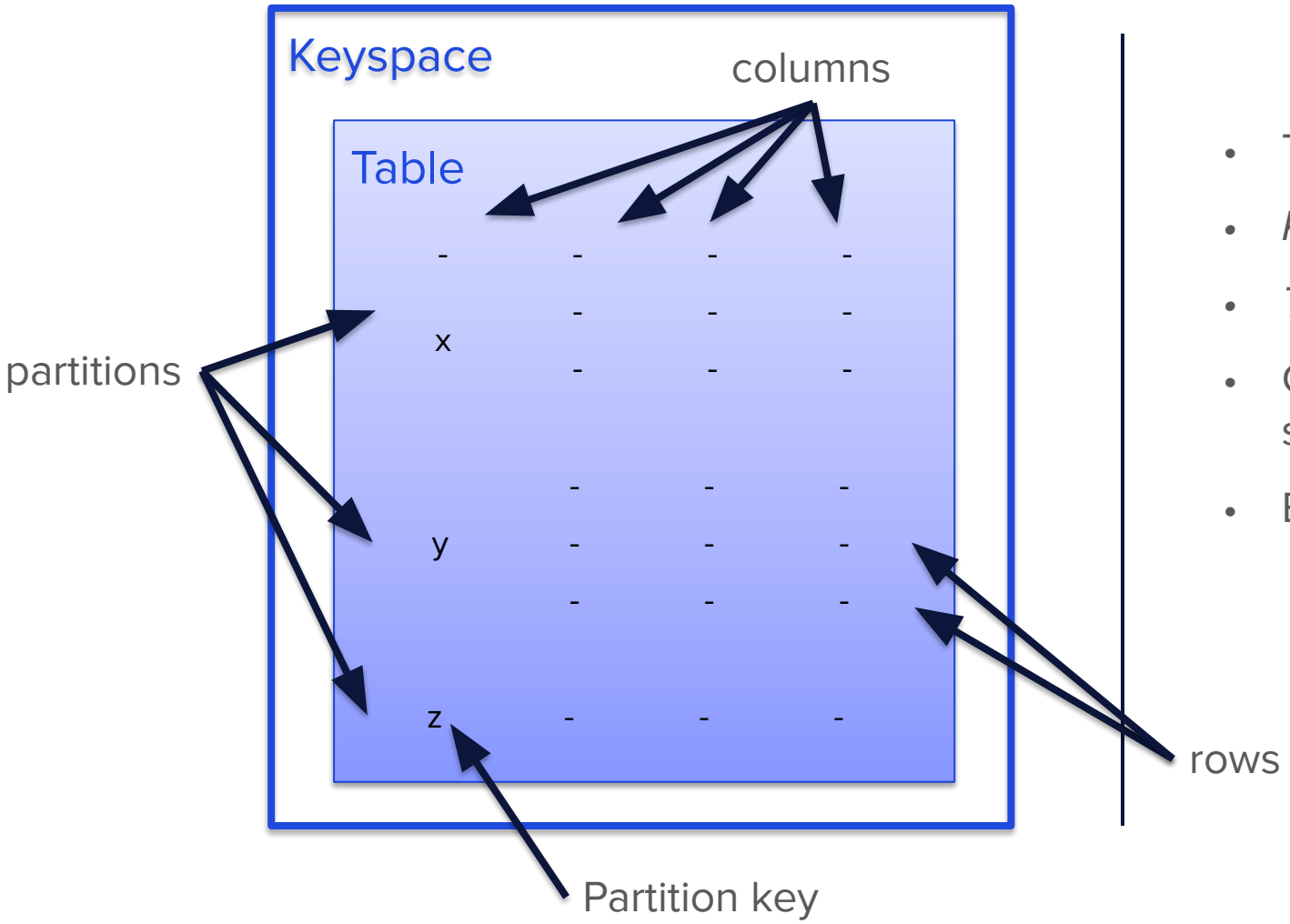
DATASTAX

DataStax Meetup

DATASTAX

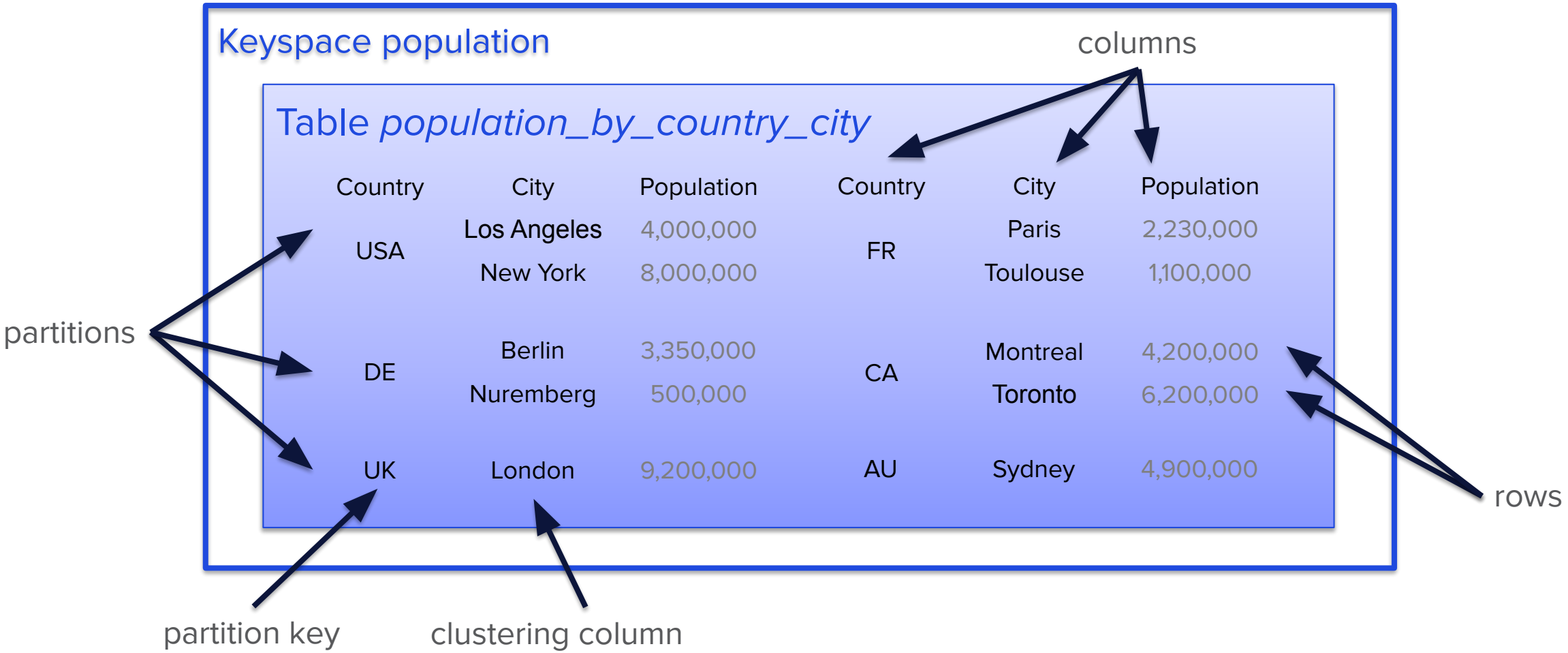
Cassandra's Data Model

How does Cassandra structure data?



- Tabular data model, with one twist
- *Keyspaces* contain *tables*
- *Tables* are organized in *rows* and *columns*
- Groups of related rows, called *partitions*, are stored together on the same node (or nodes)
- Each row contains a *partition key*
  - One or more columns that are hashed to determine which node(s) store that data

Example Data – City populations organized by country



Example Data – City populations organized by country

Keyspace population

Table *population_by_country_city*

Country	City	Population
USA	Los Angeles	4,000,000
	New York	8,000,000
DE	Berlin	3,350,000
	Nuremberg	500,000
UK	London	9,200,000

CQL Equivalent:

```
CREATE TABLE population_by_country_city (
  country text,
  city text,
  population int,
  PRIMARY KEY((country), city)
);
```

partition key

clustering column

Time for an exercise!

DATASTAX

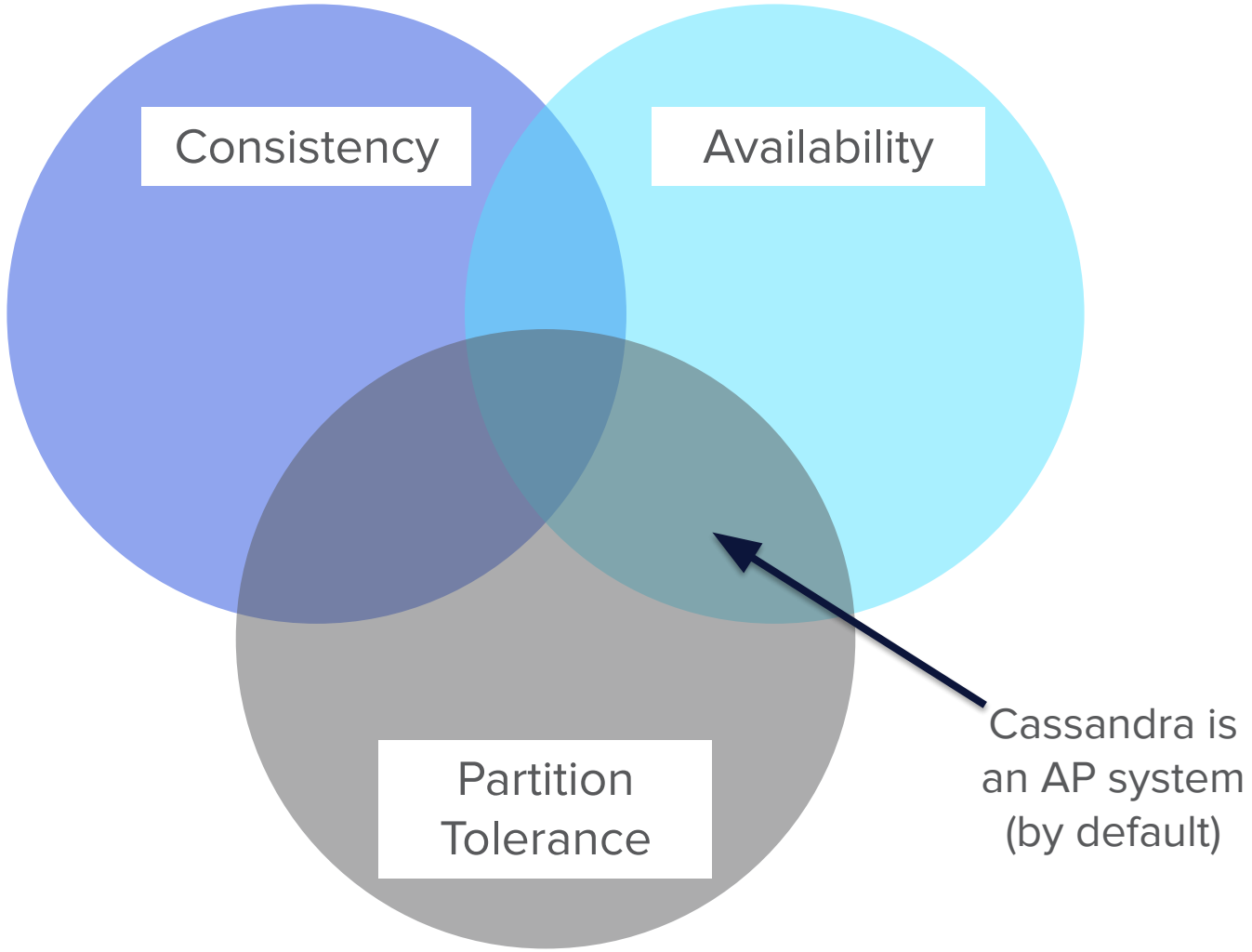
“Getting Started with Apache Cassandra™” Notebook

DataStax Developer Day

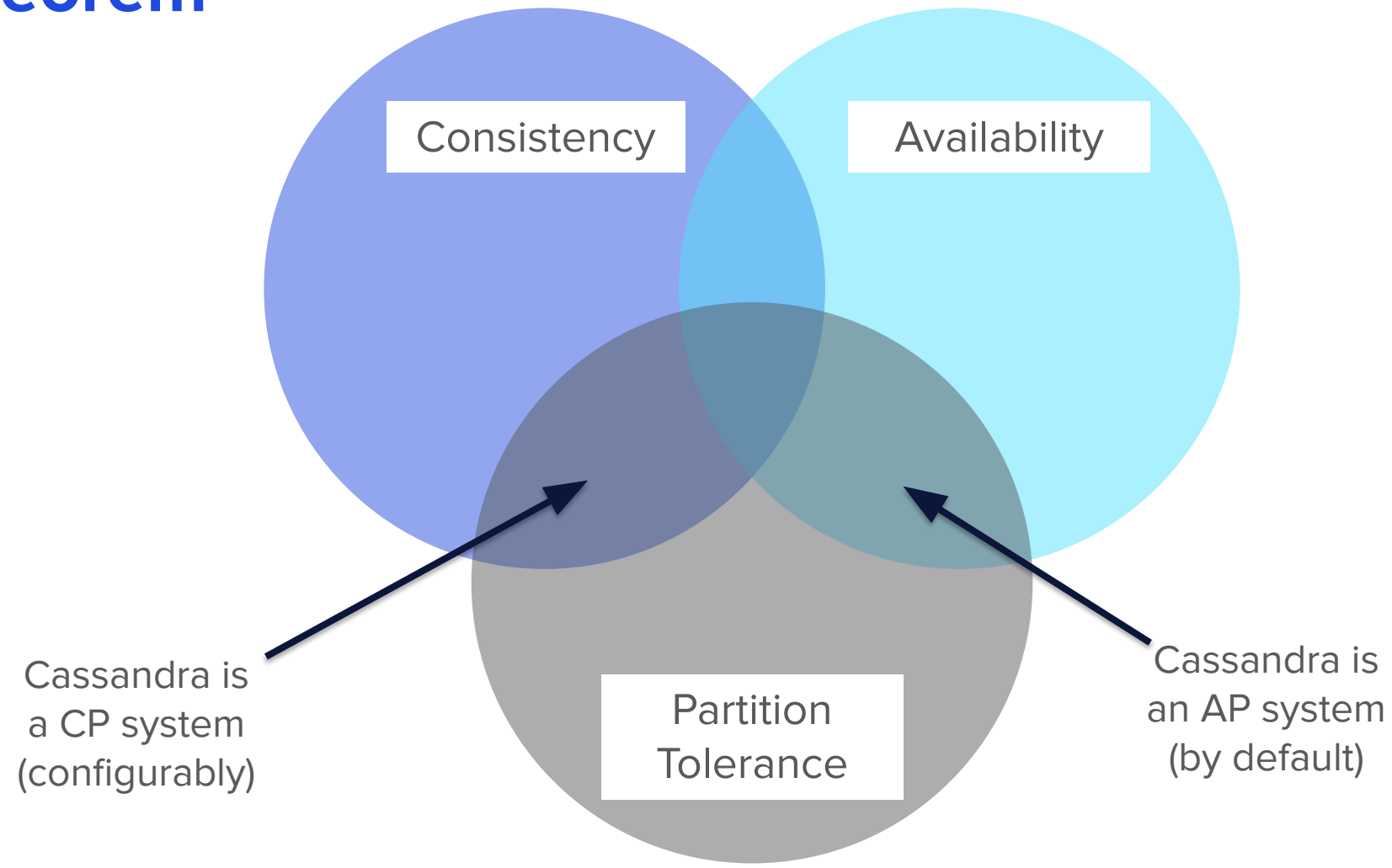
DATASTAX

Cassandra's Consistency

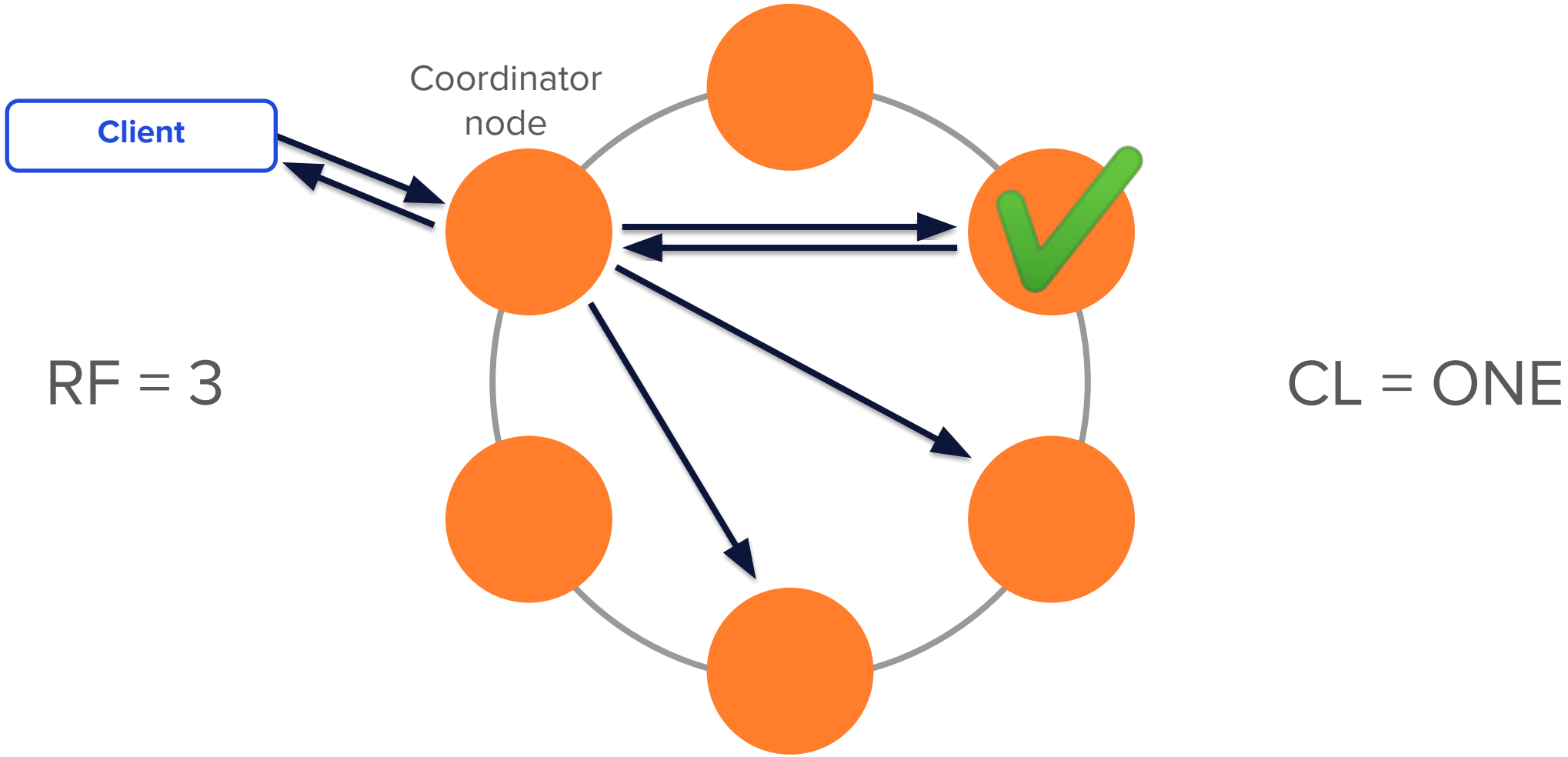
CAP Theorem


Consistency
Availability
Partition Tolerance
Cassandra is an AP system (by default)

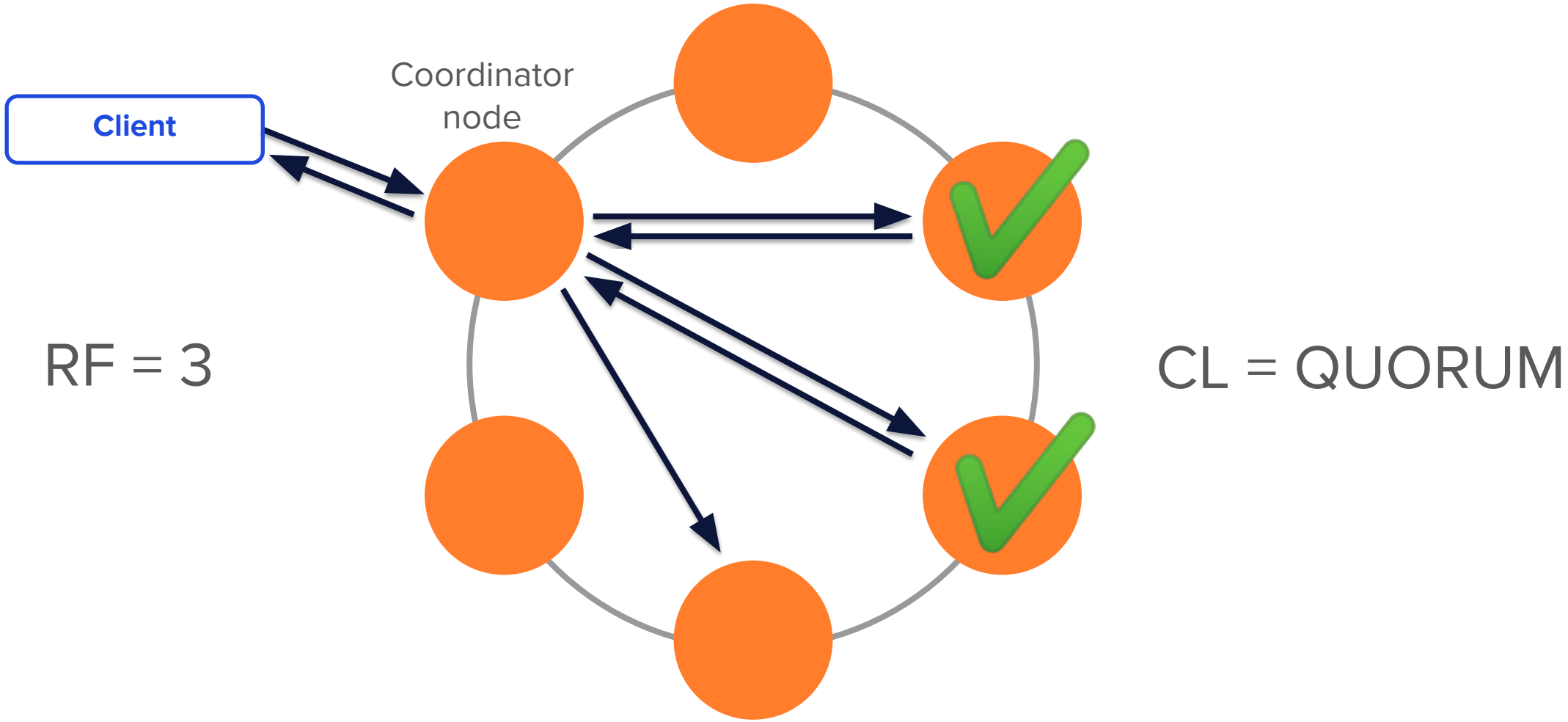
CAP Theorem


Consistency
Availability
Partition Tolerance
Cassandra is a CP system (configurably)
Cassandra is an AP system (by default)

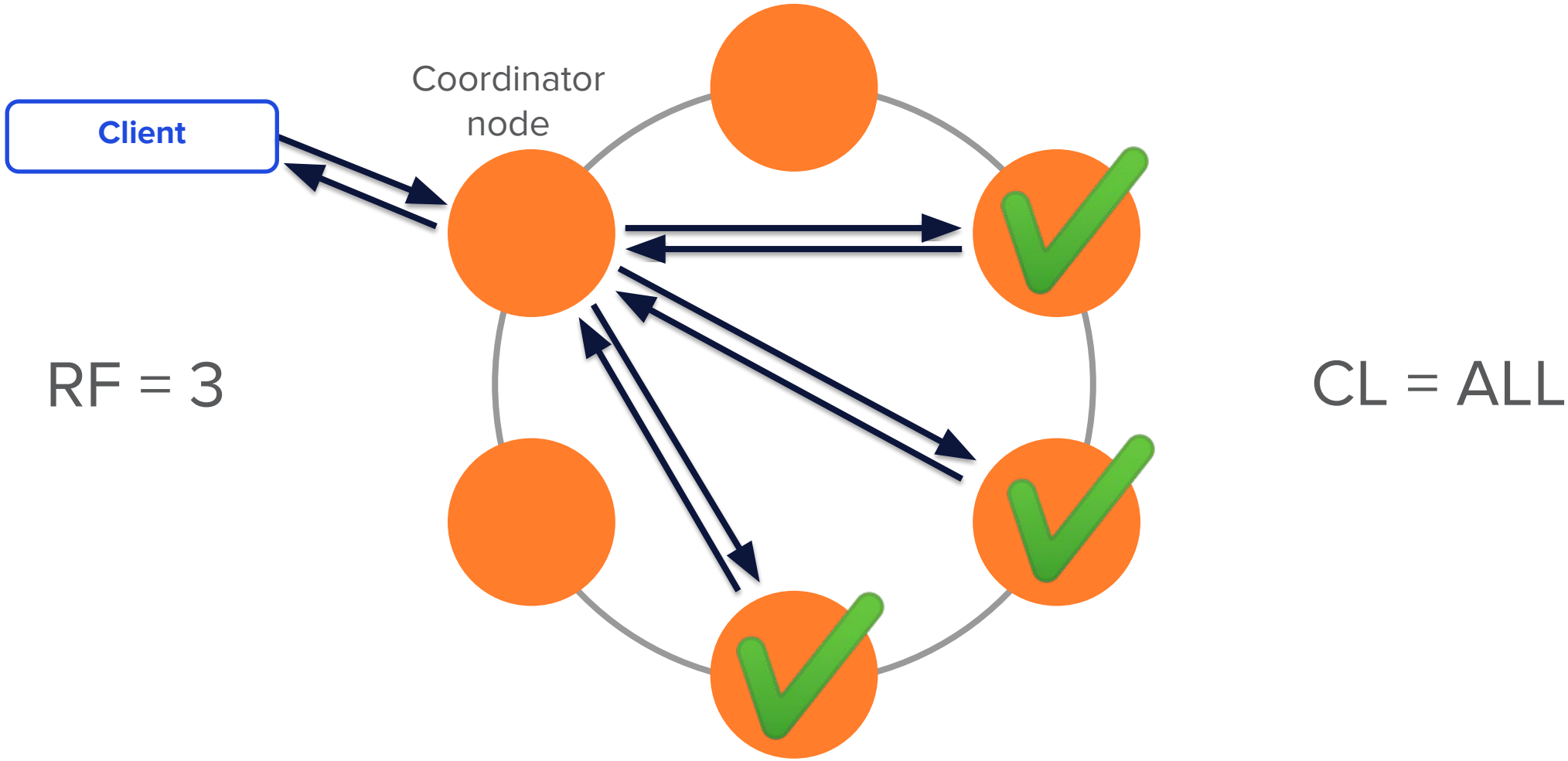
Consistency Levels



Consistency Levels



Consistency Levels



DataStax Drivers

- **DataStax Cassandra Drivers (OSS)**
  - CQL Support
  - Sync / Async API
  - Address Translation
  - Load Balancing Policies
  - Retry Policies
  - Reconnection Policies
  - Connection Pooling
  - Auto Node Discovery
  - SSL
  - Compression
  - Query Builder
  - Object Mapper
- **DataStax Enterprise Drivers**
  - OSS Drivers capabilities plus Enterprise improvements for
    - Performance, Usability, Scalability, Ecosystem
  - DSE Advanced Security, Unified Authentication
  - DSE Graph Fluent API
  - DSE Geometric Types





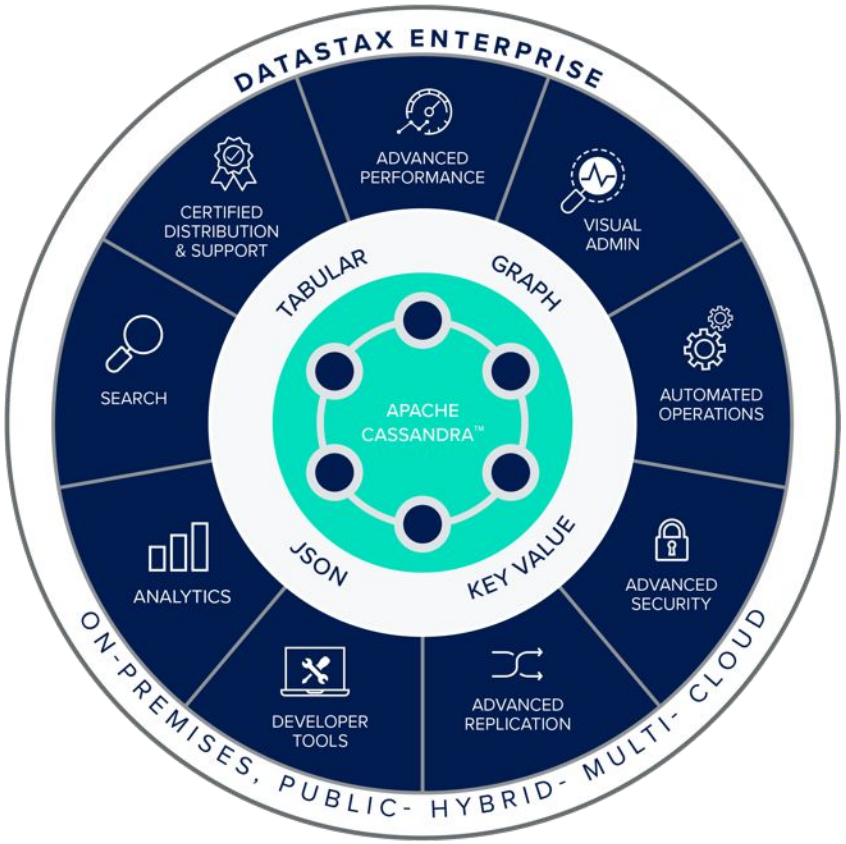

- ODBC
- JDBC

DataStax Meetup

DATASTAX

Getting Started with Apache Spark

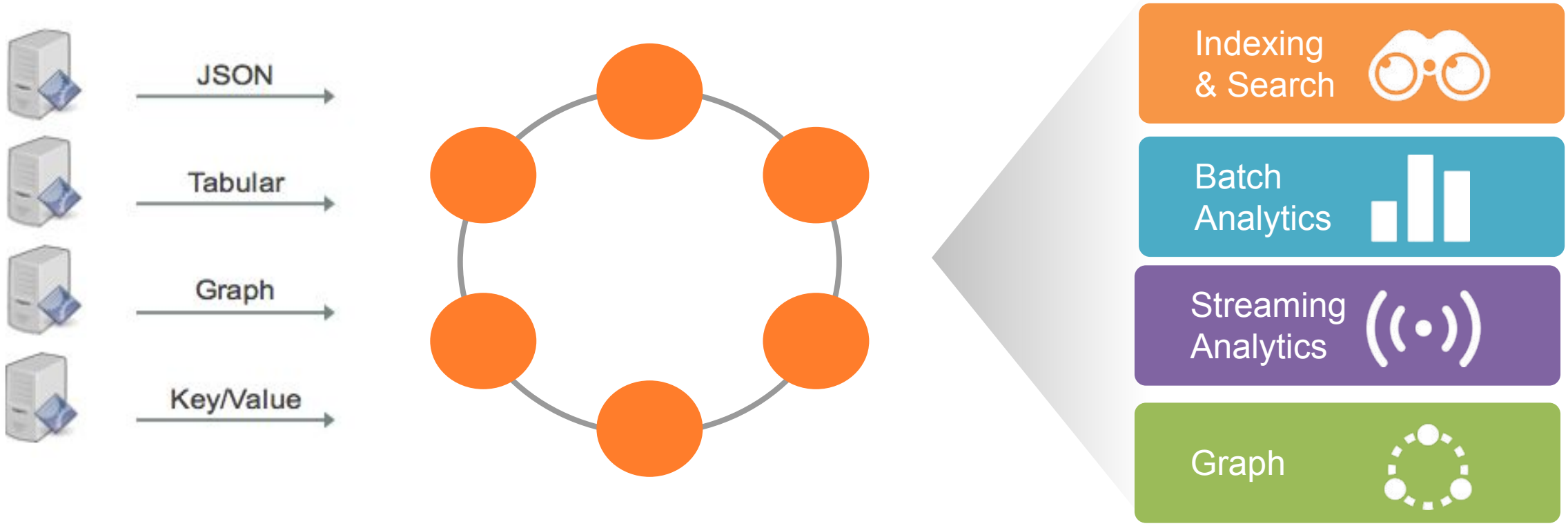
Our solution DataStax Enterprise (DSE)



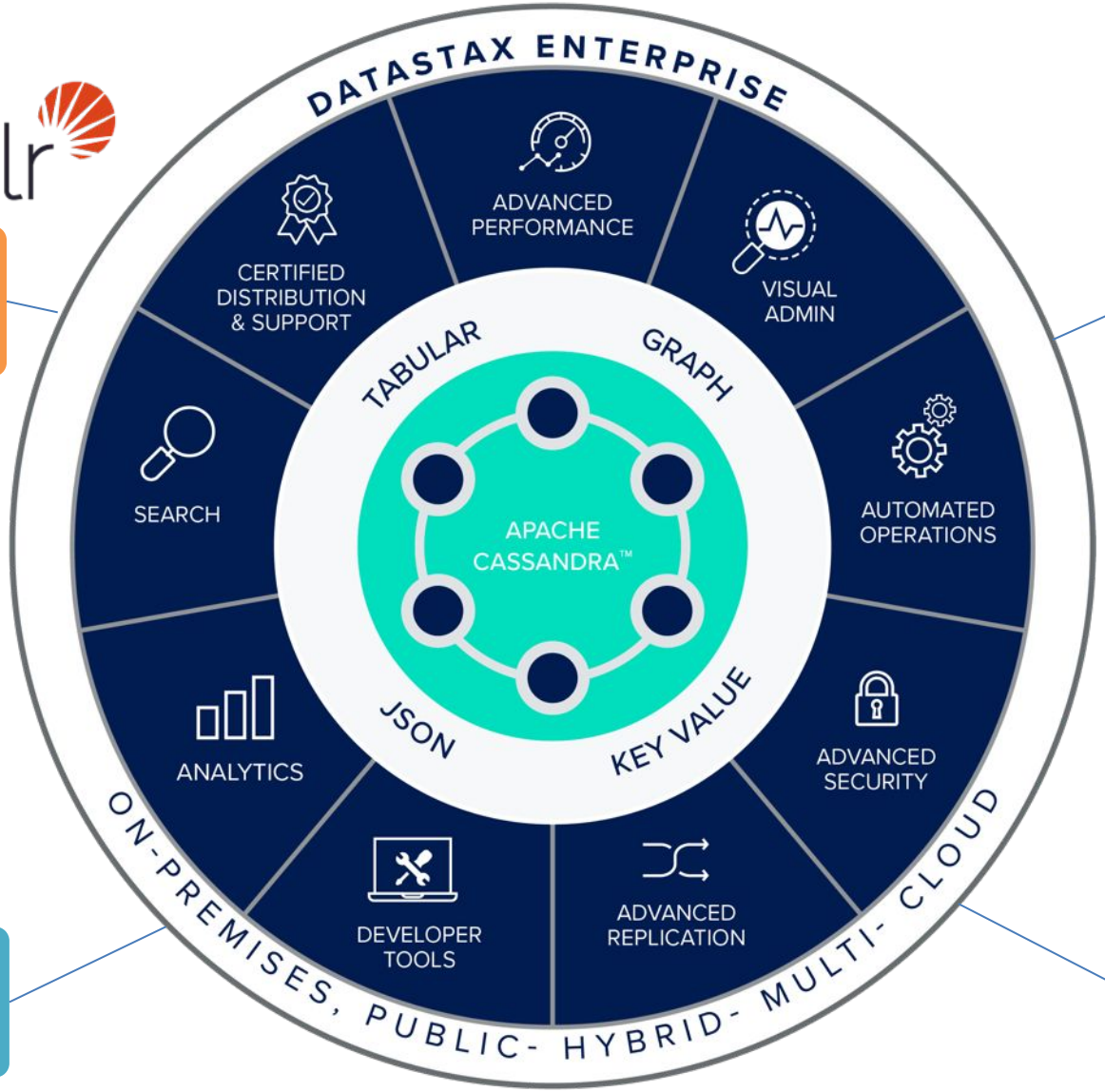
- A **unified data layer** of database, search, and analytics, all independent of the public cloud provider and portable
- Consistent data management built for on-prem, hybrid-, and/or multi-cloud
- Consistent security **model** across entire data layer
  - Row and columnar level control of your data helps you achieve data governance and compliance

DATASTAX

Integrated multi-model/mixed workload platform



DataStax Enterprise



Lucene Solr

Indexing & Search

Apache TinkerPop

Graph

Spark

Batch Analytics

cassandra

OLTP

DATASTAX

Data Analytics

- **Definition**

  Science and craft of building applications from data analysis steps to discover useful information and support data-driven decision making

- **Use cases**
  - Recommendations
  - Fraud detection
  - Social networks and Web link analysis
  - Marketing and advertising decisions
  - Customer 360
  - Sales and stock market analytics
  - IoT analytics

- **Analysing Steps**
  - Statistical analysis
  - Classification
  - Clustering
  - Regression
  - Similarity matching
  - Collaborative filtering
  - Profiling
  - Dimensionality reduction
  - Feature extraction

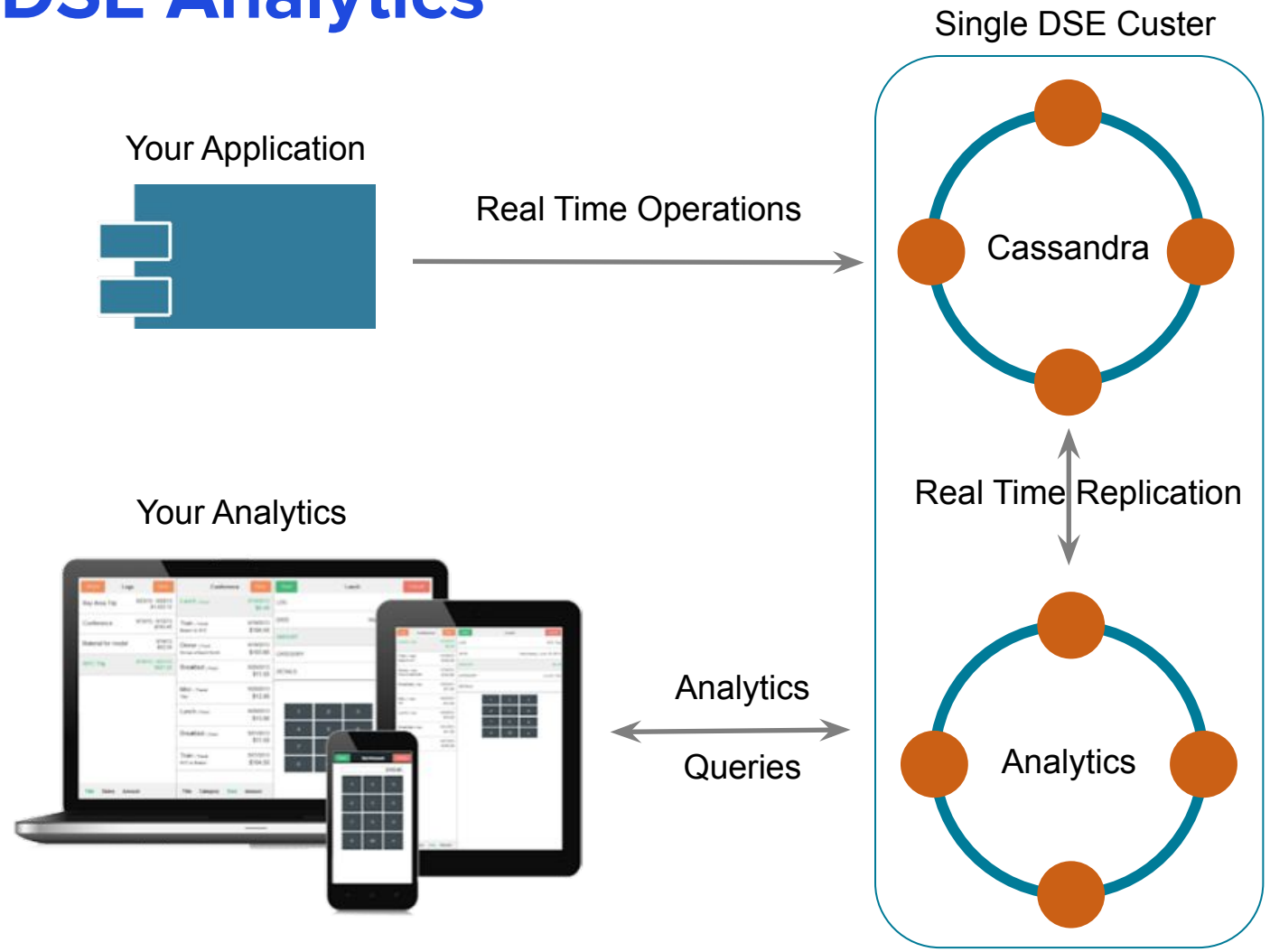
Apache Spark™

Distributed computation engine
designed for **big data** and in-memory processing

- Interactive and batch analytics
- Up to 100x faster than Hadoop
- 5-10x less code than Hadoop
- Efficiency and scalability
- Fault-tolerance

DATASTAX

DSE Analytics



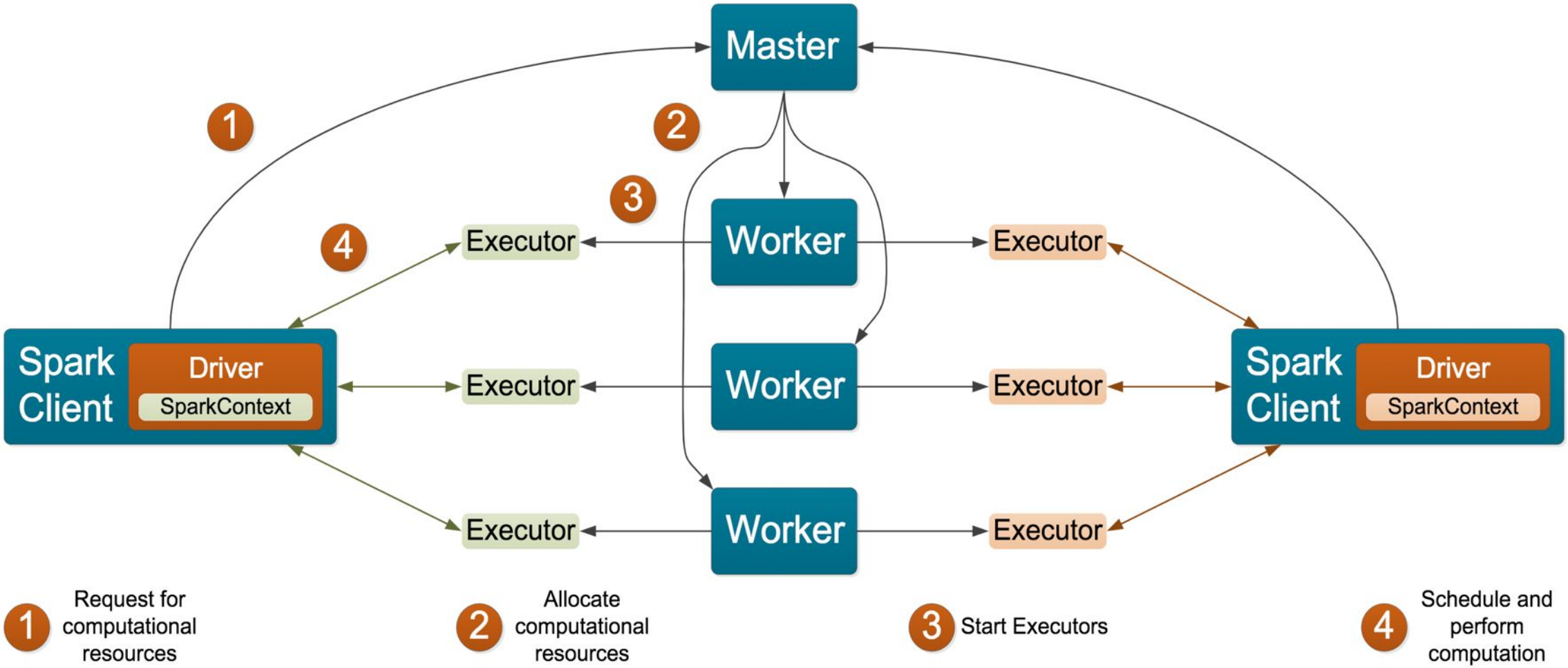
Streaming, ad-hoc, and batch

- High-performance
- High availability
- Workload management
- SQL reporting

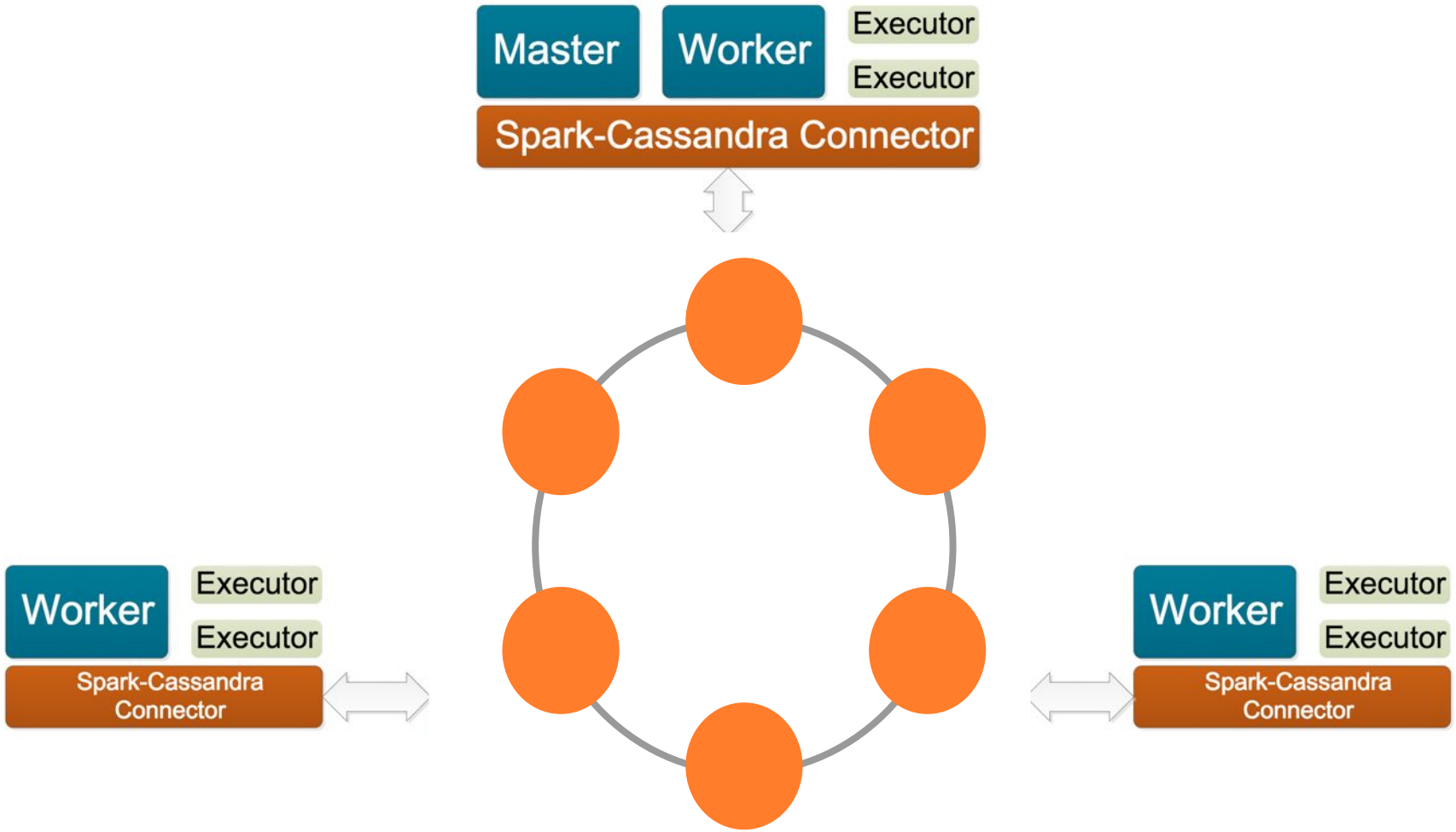
Compared to self-managed:

- No ETL
- True HA without Zookeeper

Spark Architecture



Architecture with Spark DSE Driver



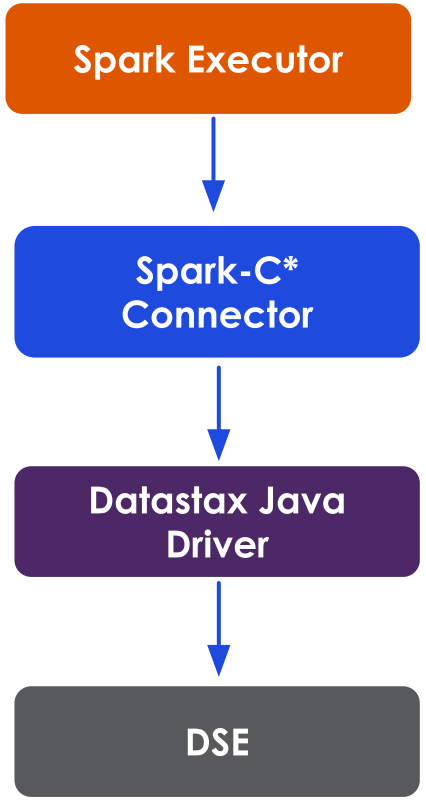
Database Access with DataStax Driver

- DataStax Cassandra Spark driver
  - Implemented mostly in Scala
  - Scala + Java APIs
  - Does automatic type conversions

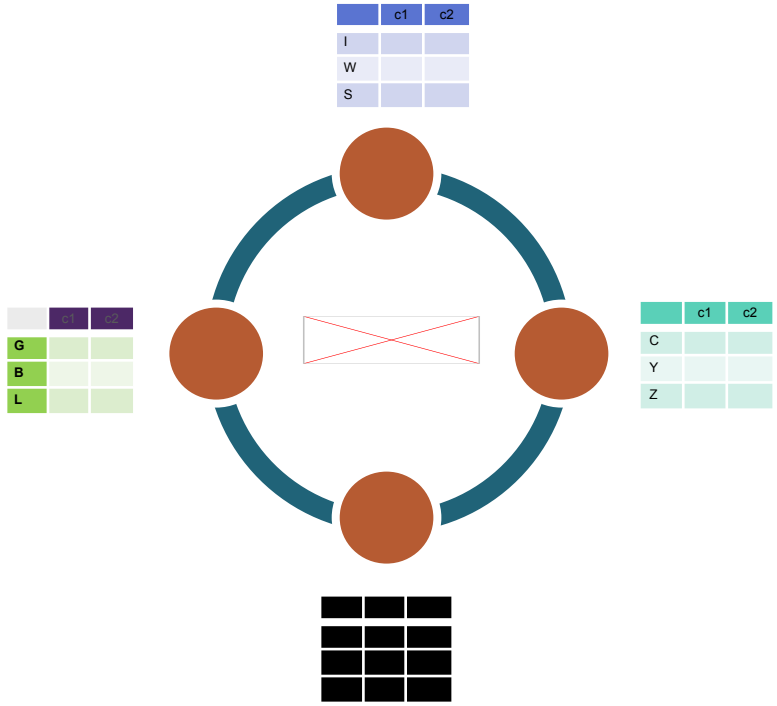
```
// Spark connection options
val conf = new SparkConf(true)
.set("spark.cassandra.connection.host", "127.0.0.1")
.set("spark.cassandra.auth.username", "cassandra")
.set("spark.cassandra.auth.password", "cassandra")
val sc = new SparkContext("spark://127.0.0.1:7077", "myapp", conf)


// Read from DSE and add partitioner with primary key
val rdd = sc.cassandraTable("my_keyspace", "my_table").byKey("pk","cc")


// Save to DSE
rdd.saveToCassandra("my_keyspace", "my_table", SomeColumns("key",
"value"))
```



Data Locality

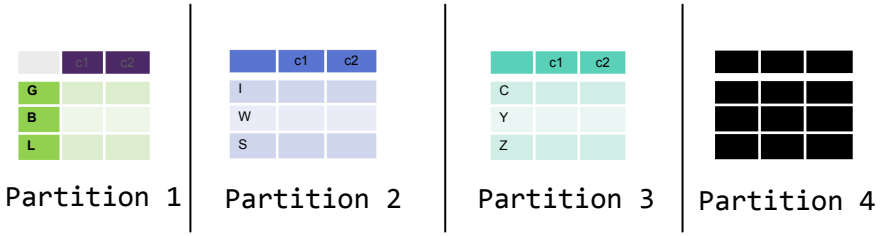


- DSE Analytics respects data locality
- No need for ETL between separated clusters
- Spark Master HA

Every Spark task uses a CQL-like query to fetch data for a given token range:

```
SELECT "key", "value"FROM "keyspace"."table“
    WHERE
        token("key") >  384023840238403 AND
        token("key") <= 38402992849280
ALLOW FILTERING
```

In Memory: Distributed on all available nodes



DATASTAX

Integrating Analytics into your application

```
dse cassandra -k
```

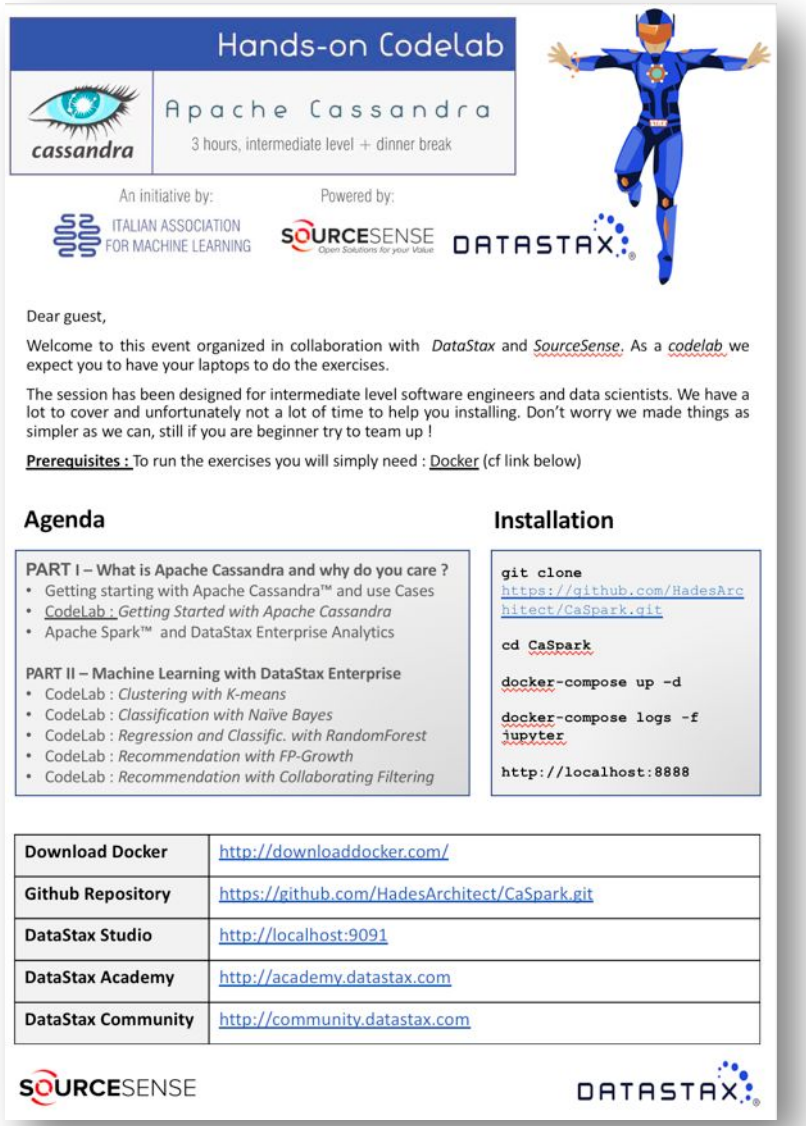
- Aaaaand you're done

DataStax Meetup

DATASTAX

Exercises Bootstrap

Before Starting



Hands-on Codelab

Apache Cassandra

3 hours, intermediate level + dinner break

An initiative by: ITALIAN ASSOCIATION FOR MACHINE LEARNING

Powered by: SOURCESENSE DATASTAX

Dear guest,

Welcome to this event organized in collaboration with *DataStax* and *SourceSense*. As a *codelab* we expect you to have your laptops to do the exercises.

The session has been designed for intermediate level software engineers and data scientists. We have a lot to cover and unfortunately not a lot of time to help you installing. Don't worry we made things as simpler as we can, still if you are beginner try to team up !

Prerequisites : To run the exercises you will simply need : Docker (cf link below)

Agenda

PART I – What is Apache Cassandra and why do you care ?
- Getting starting with Apache Cassandra™ and use Cases
- CodeLab : *Getting Started with Apache Cassandra*
- Apache Spark™ and DataStax Enterprise Analytics

PART II – Machine Learning with DataStax Enterprise
- CodeLab : *Clustering with K-means*
- CodeLab : *Classification with Naïve Bayes*
- CodeLab : *Regression and Classific. with RandomForest*
- CodeLab : *Recommendation with FP-Growth*
- CodeLab : *Recommendation with Collaborating Filtering*

Installation

```
git clone
https://github.com/HadesArc
hitect/CaSpark.git

cd CaSpark

docker-compose up -d

docker-compose logs -f
jupyter

http://localhost:8888
```

Download Docker	http://downloaddocker.com/
Github Repository	https://github.com/HadesArchitect/CaSpark.git
DataStax Studio	http://localhost:9091
DataStax Academy	http://academy.datastax.com
DataStax Community	http://community.datastax.com

SOURCESENSE DATASTAX

- You should have one of those sheet.
- Please execute the Installation steps **as soon as possible.** This will download a few docker images that can take some time !

DATASTAX

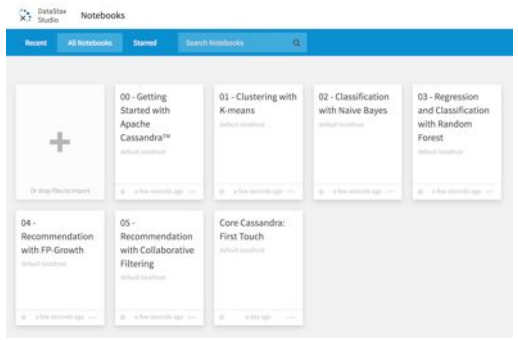
Your environment

```
docker-compose up -d
```

DOCKER

9091

DataStax Studio



DataStax Studio

DataStax Enterprise (DSE)

CASSANDRA

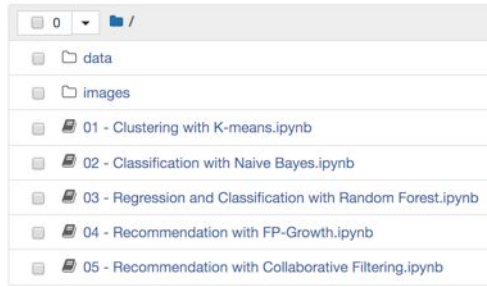
SEARCH

ANALYTICS

DATASTAX ENTERPRISE

8888

Jupyter Notebook



Visual development tool for DSE

Time for an exercise!

DATASTAX

“Getting Started with Apache Spark™” Notebook

DataStax Meetup

DATASTAX

Machine Learning

DataStax Meetup

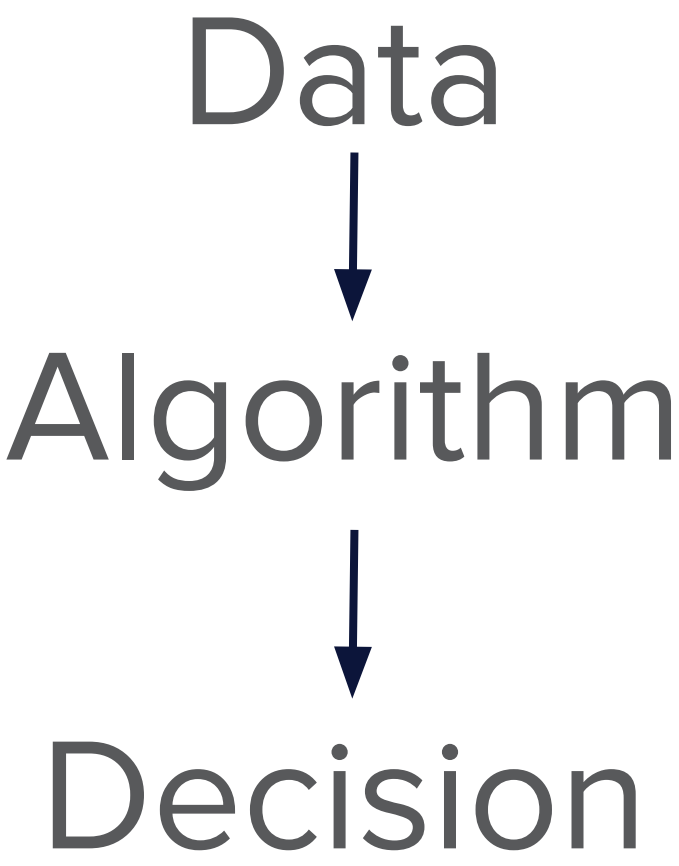
DATASTAX

Machine Learning

"Machine Learning is a science of
drawing circles and colorizing them"

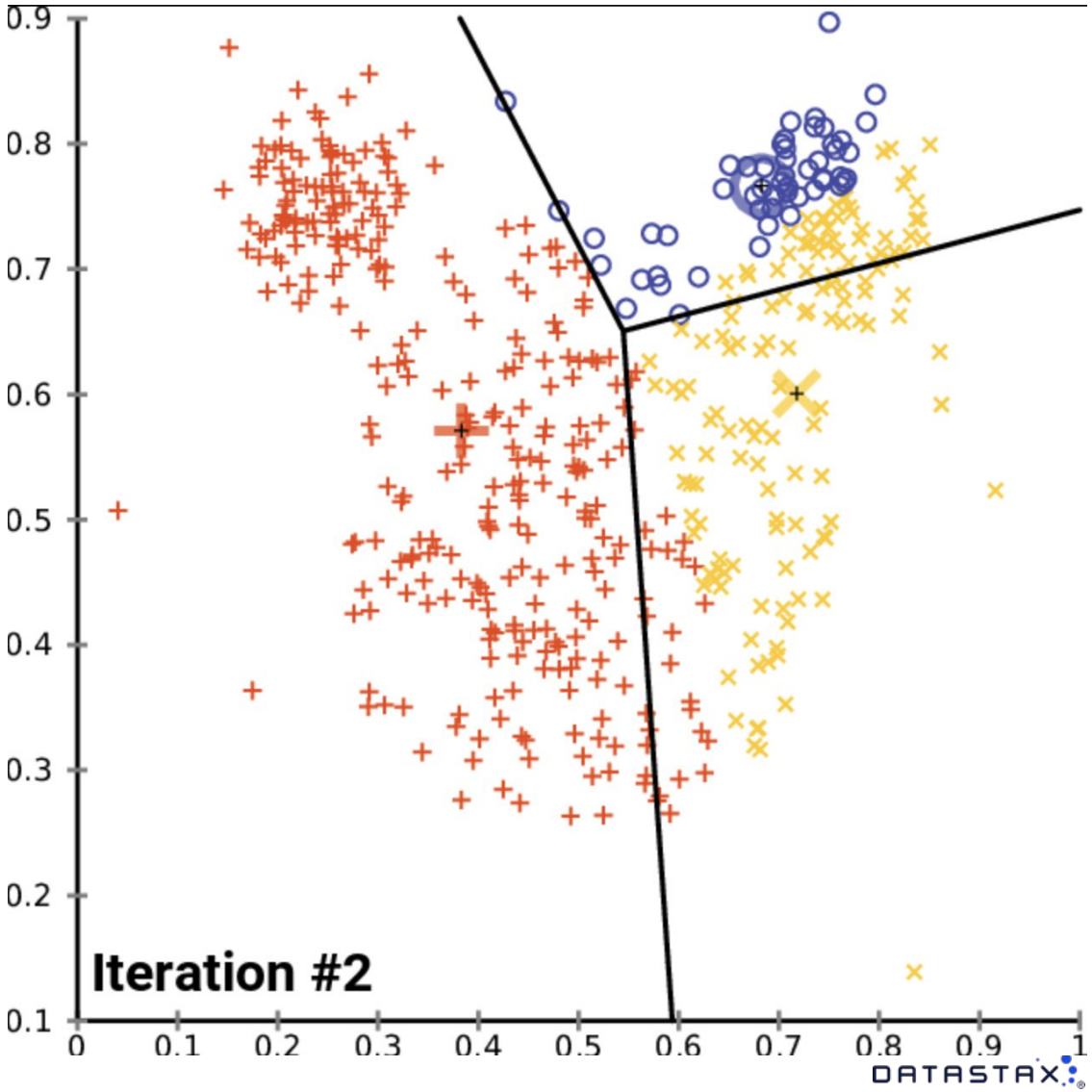
A. Volochnev

Machine Learning



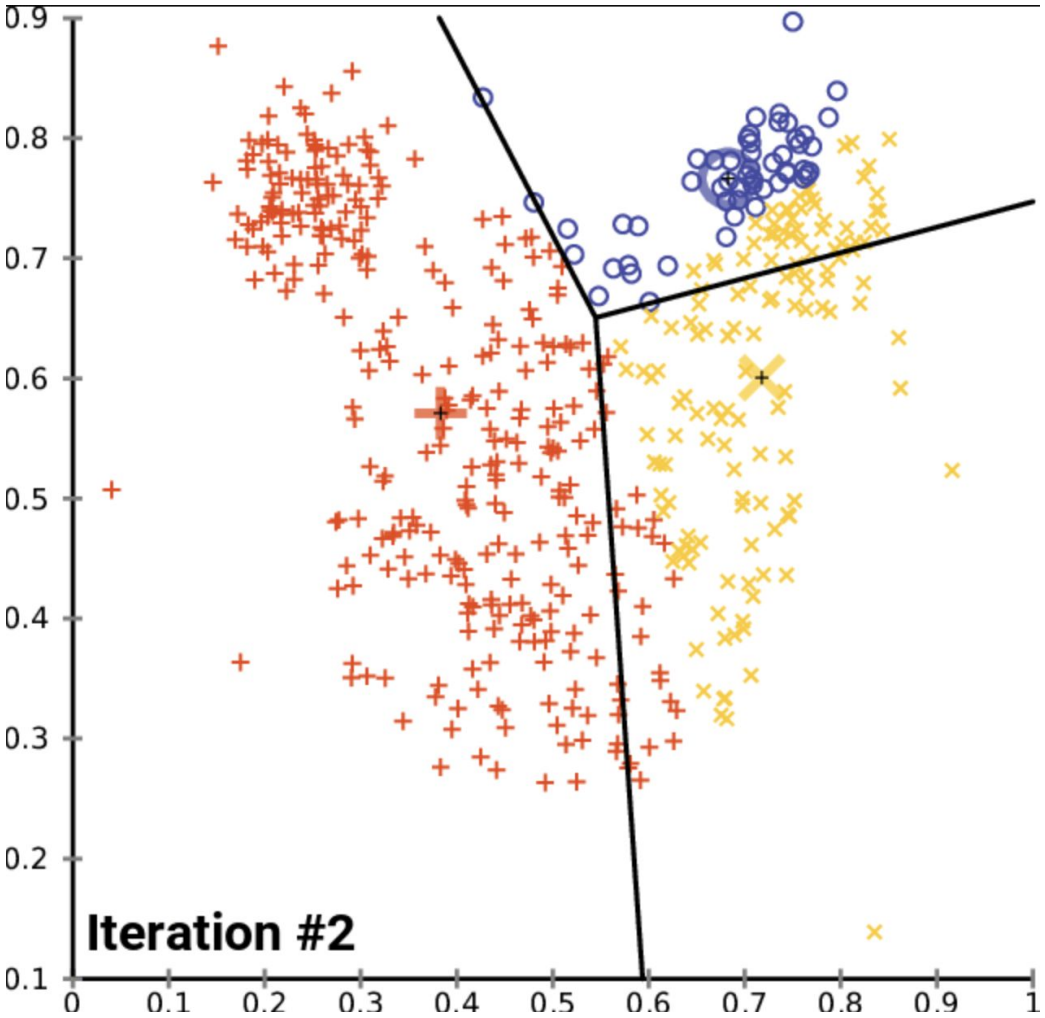
K-Means Clustering

“K-means clustering is a method of vector quantization, originally from signal processing, that is popular for cluster analysis in data mining. K-means clustering aims to partition observations into clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. This results in a partitioning of the data space into Voronoi cells. K-Means minimizes within-cluster variances (squared Euclidean distances), but not regular Euclidean distances, which would be the more difficult Weber problem: the mean optimizes squared errors, whereas only the geometric median minimizes Euclidean distances.”



K-Means Clustering Lab

- https://github.com/HadesArchitect/CaSpark
- http://localhost:8888
- password: datastax
- **K-Means Notebook**



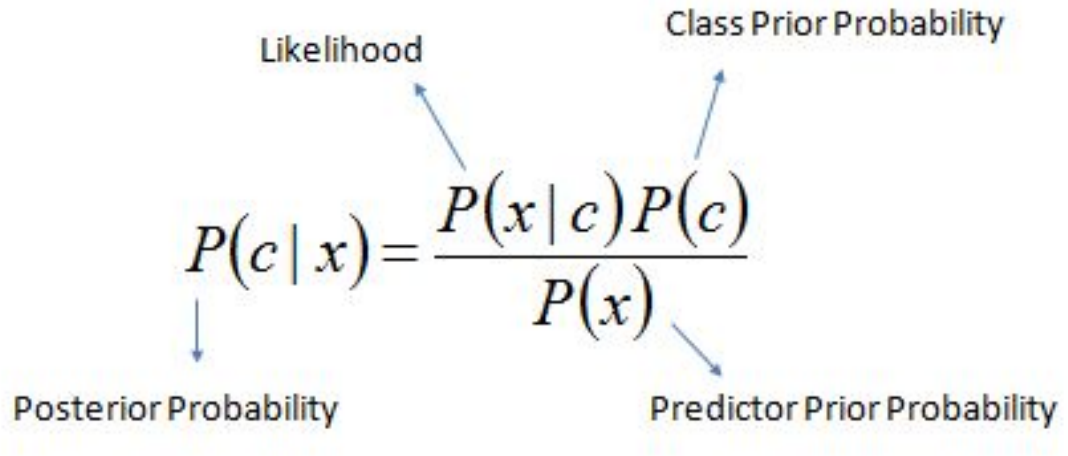
DATASTAX

Naïve Bayes

“Naïve Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong “naïve” independence assumptions between the features.

Naïve Bayes is a popular method for text categorization, the problem of judging documents as belonging to one category or the other (such as spam or legitimate, etc.) with word frequencies as the features. With appropriate pre-processing, it is competitive in this domain with more advanced methods including support vector machines. It also finds application in automatic medical diagnosis.

Naïve Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables in a learning problem. Maximum-likelihood training can be done by evaluating a closed-form expression, which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers.”

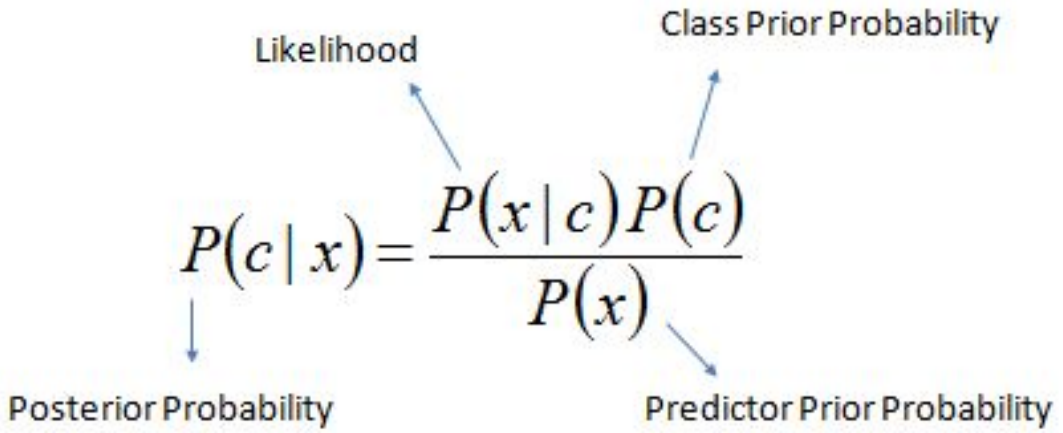


$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

DATASTAX

Naïve Bayes Lab

- https://github.com/HadesArchitect/CaSpark
- http://localhost:8888
- password: datastax
- **Naïve Bayes Notebook**

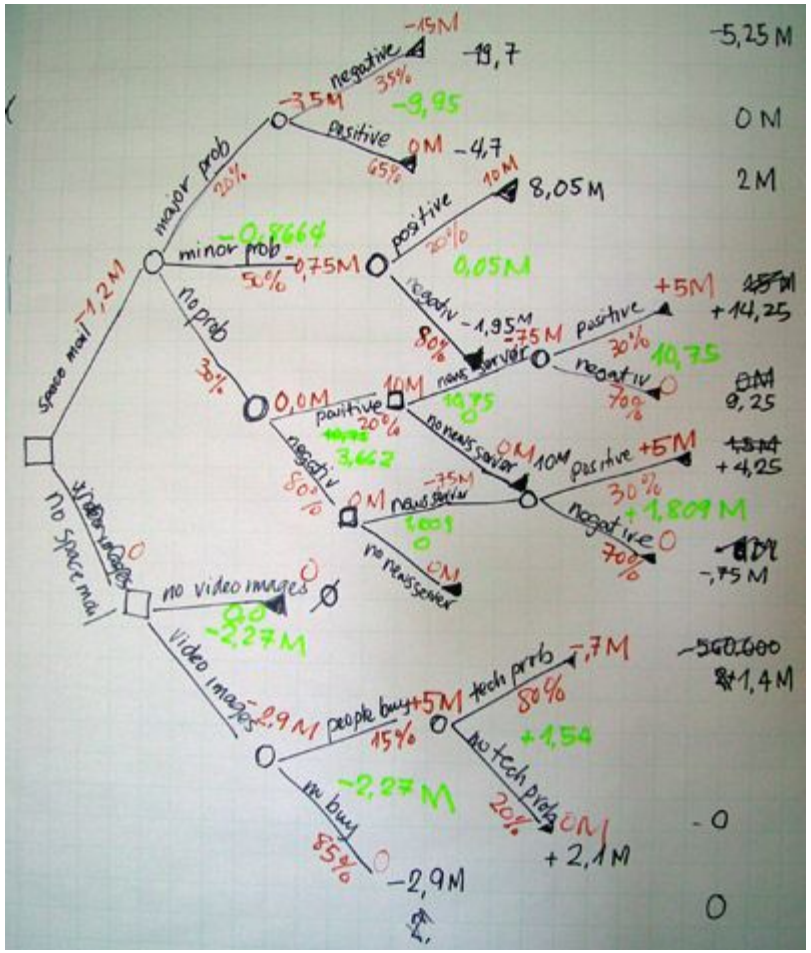


$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Random Forest

“Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set.

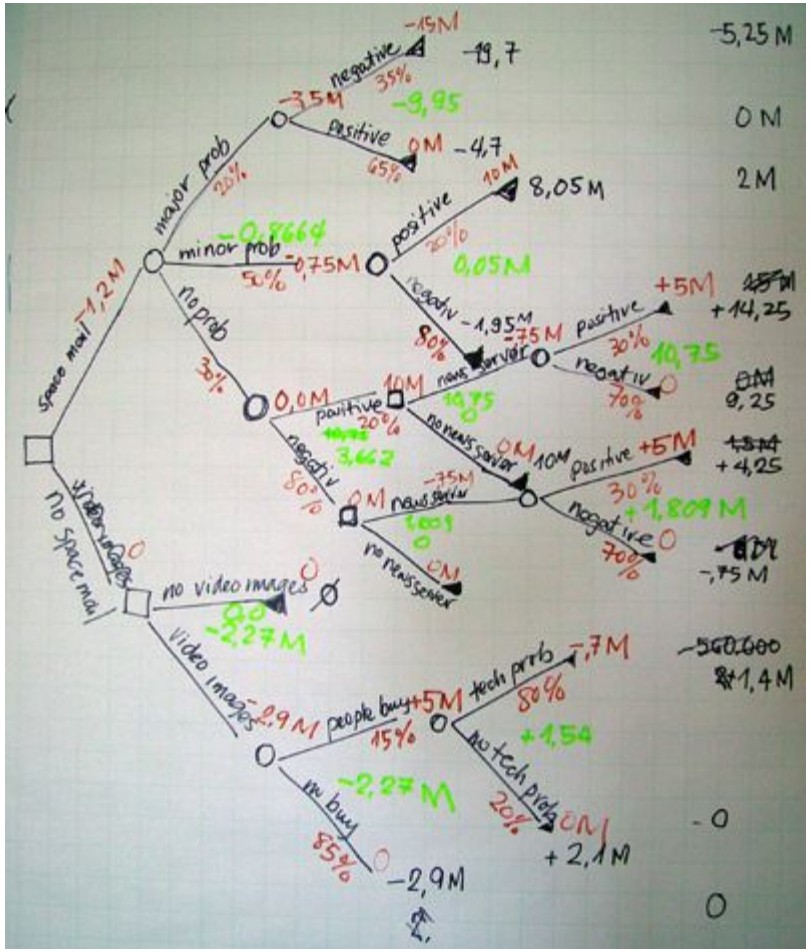
The first algorithm for random decision forests was created by Tin Kam Ho using the random subspace method, which, in Ho's formulation, is a way to implement the "stochastic discrimination" approach to classification proposed by Eugene Kleinberg.”



DATASTAX

Random Forest Lab

- https://github.com/HadesArchitect/CaSpark
- http://localhost:8888
- password: datastax
- **Random Forest Notebook**

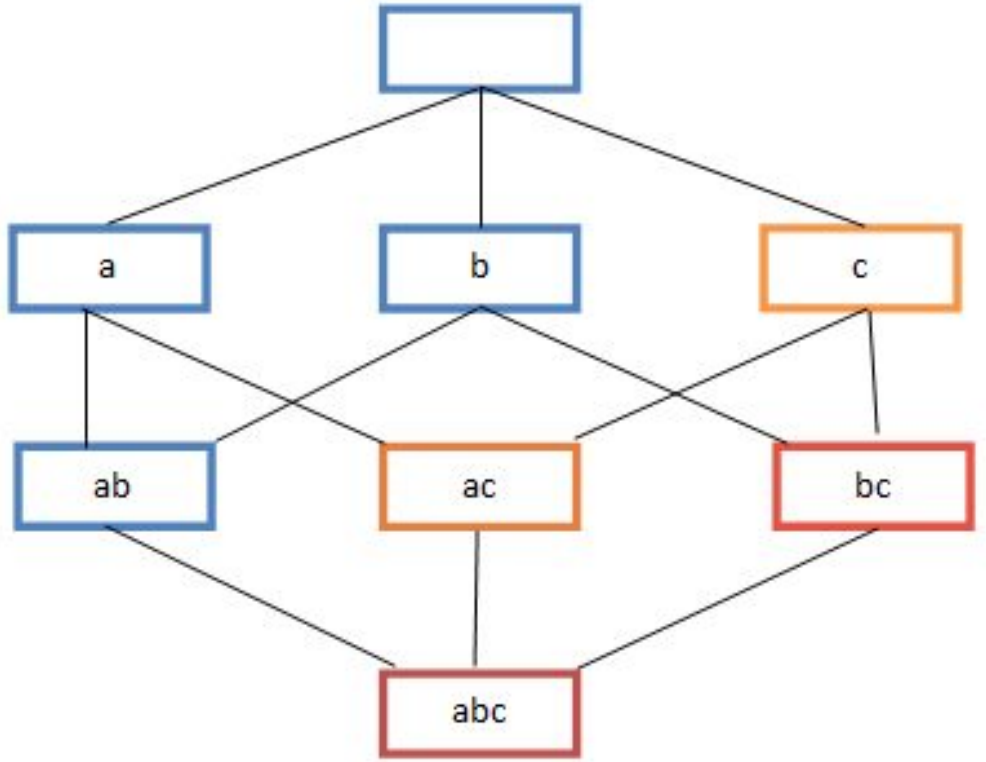


FP-Growth (Frequent Pattern Growth)

“In the first pass, the algorithm counts the occurrences of items (attribute-value pairs) in the dataset of transactions, and stores these counts in a header table. In the second pass, it builds the FP-tree structure by inserting transactions into a trie.

Items in each transaction have to be sorted by descending order of their frequency in the dataset before being inserted so that the tree can be processed quickly. Items in each transaction that do not meet the minimum support requirement are discarded. If many transactions share most frequent items, the FP-tree provides high compression close to tree root.

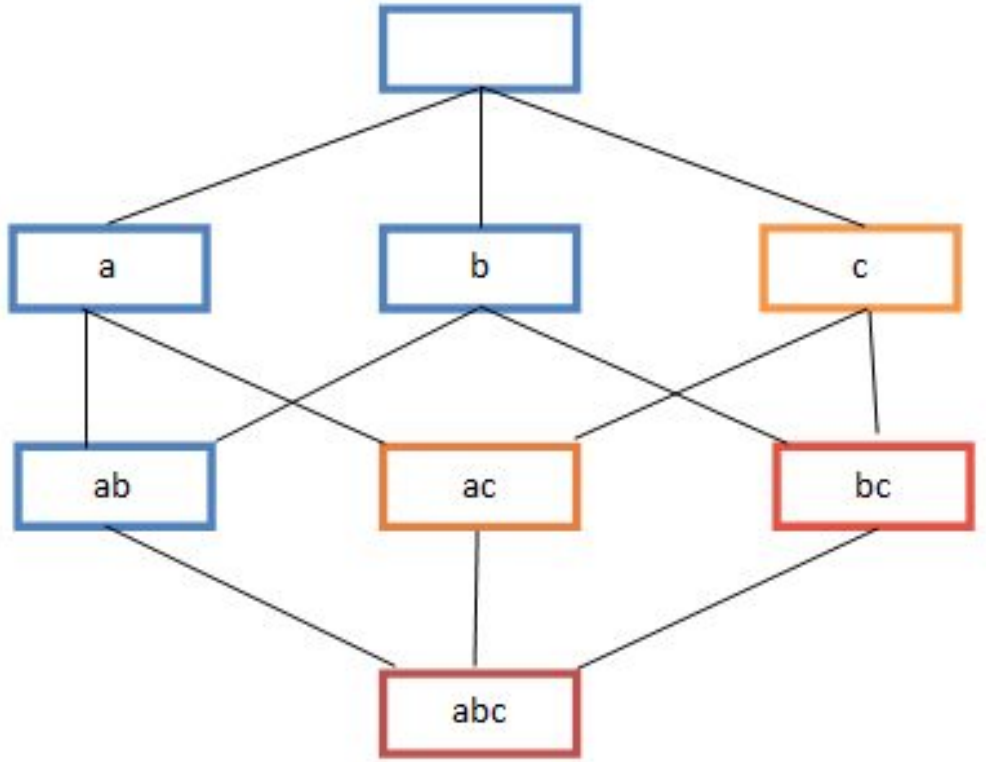
Recursive processing of this compressed version of the main dataset grows frequent item sets directly, instead of generating candidate items and testing them against the entire database”



DATASTAX

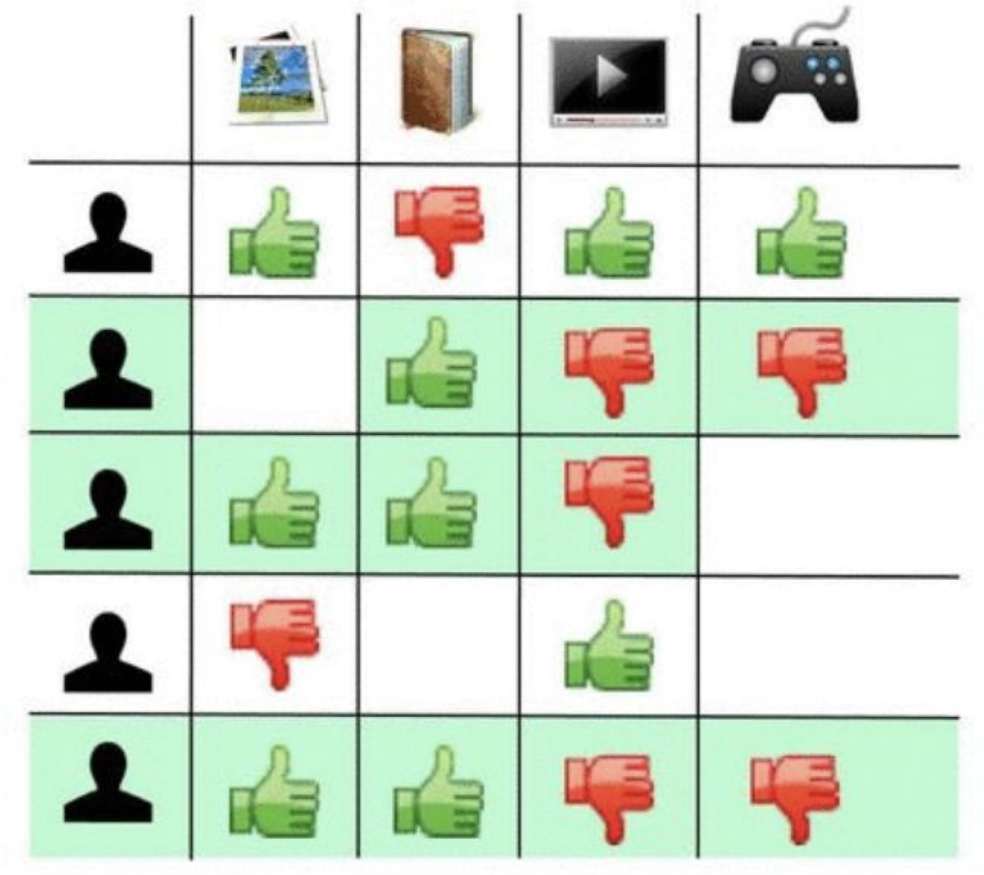
FP-Growth Lab

- https://github.com/HadesArchitect/CaSpark
- http://localhost:8888
- password: datastax
- **FP-Growth Notebook**



Collaborative Filtering

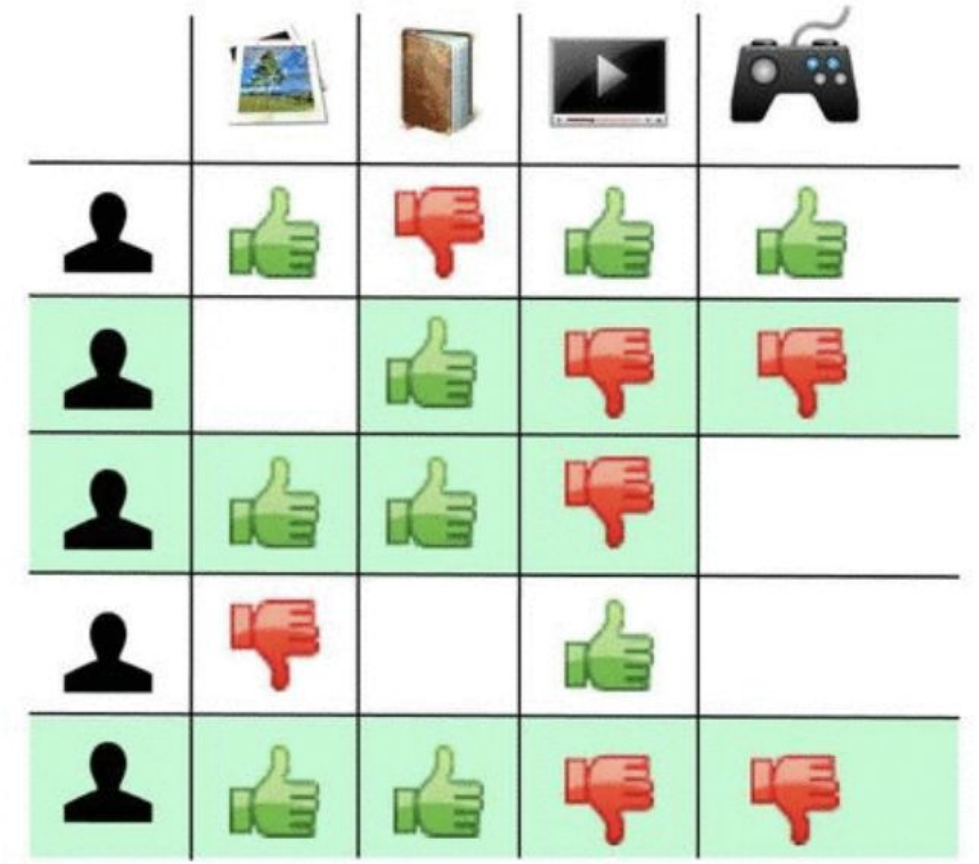
“Collaborative filtering is a method of making automatic predictions about the interests of a user by collecting preferences or taste information from many users. The underlying assumption of the collaborative filtering approach is that if a person A has the same opinion as a person B on an issue, A is more likely to have B's opinion on a different issue than that of a randomly chosen person. For example, a collaborative filtering recommendation system for television tastes could make predictions about which television show a user should like given a partial list of that user's tastes. Note that these predictions are specific to the user, but use information gleaned from many users. This differs from the simpler approach of giving an average score for each item of interest, for example based on its number of votes.”



DATASTAX

Collaborative Filtering Lab

- https://github.com/HadesArchitect/CaSpark
- http://localhost:8888
- password: datastax
- **Collaborative Filtering Notebook**



DATASTAX

DataStax Developer Day

DATASTAX

Developer Resources

Resources for Developers

- DataStax Academy
  - Training Courses and Certifications
  - Developer Blog
  - Distributed Data Show podcast
- Live Events
  - Developer Day
  - Meetups
- More Content
  - YouTube Channel
  - Live coding on Twitch
  - KillrVideo reference application
  - DataStax Academy Slack

KillrVideo

YouTube twitch

DATASTAX

Developer Day

A day of hands-on learning about DataStax Enterprise and Apache Cassandra™

- Use Cases
- Core Cassandra
- Cassandra Data Modeling
- Application Development
- Search, Analytics and Graph
- Operations and Security

Network with experts, Developer Advocates and peers

Open to the public, or schedule a private event

DATASTAX

Training Courses at DataStax Academy

- Free self-paced DSE 6 courses
  - DS201: DataStax Enterprise 6 Foundations of Apache Cassandra™
  - DS210: DataStax Enterprise 6 Operations with Apache Cassandra™
  - DS220: DataStax Enterprise 6 Practical Application Data Modeling with Apache Cassandra™
  - DS330: DataStax Enterprise 6 Graph
  - DS332: DataStax Enterprise 6 Graph Analytics (NEW)

https://academy.datastax.com

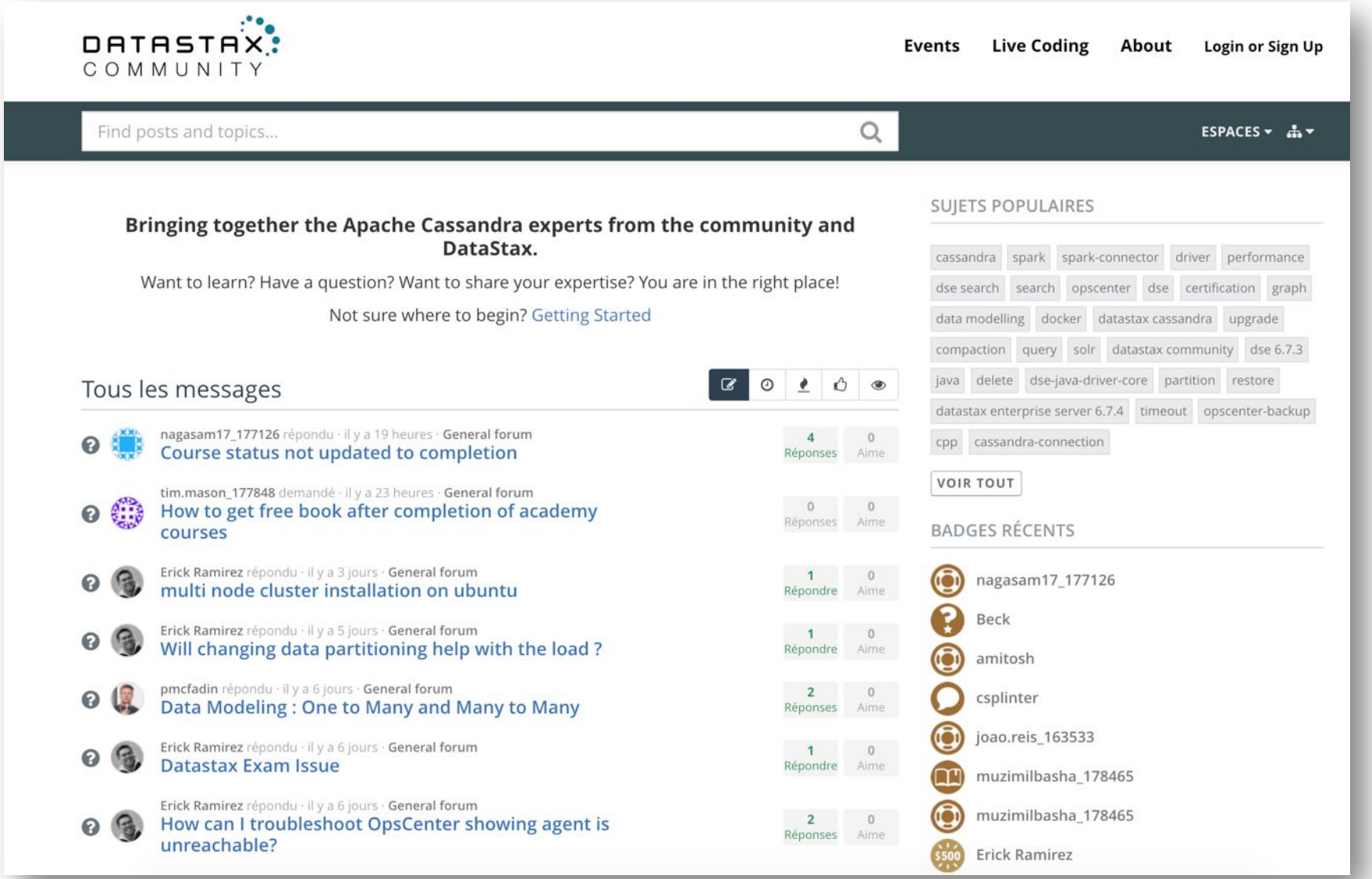
DATASTAX

Learning Paths on DataStax Academy

- Unsure where to start?
- Follow a learning path to learn about topics related to your role.
  - Administrator
  - Analytics Specialist
  - Architect
  - Developer
  - Graph Specialist
  - Search Specialist

https://academy.datastax.com/paths

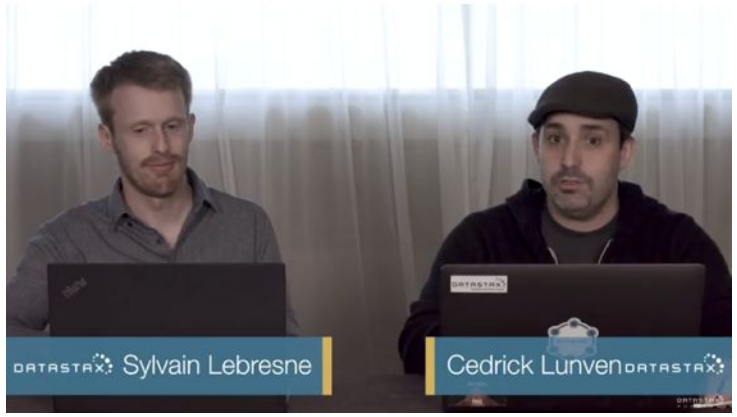
community.datastax.com



DATASTAX

Distributed Data Show

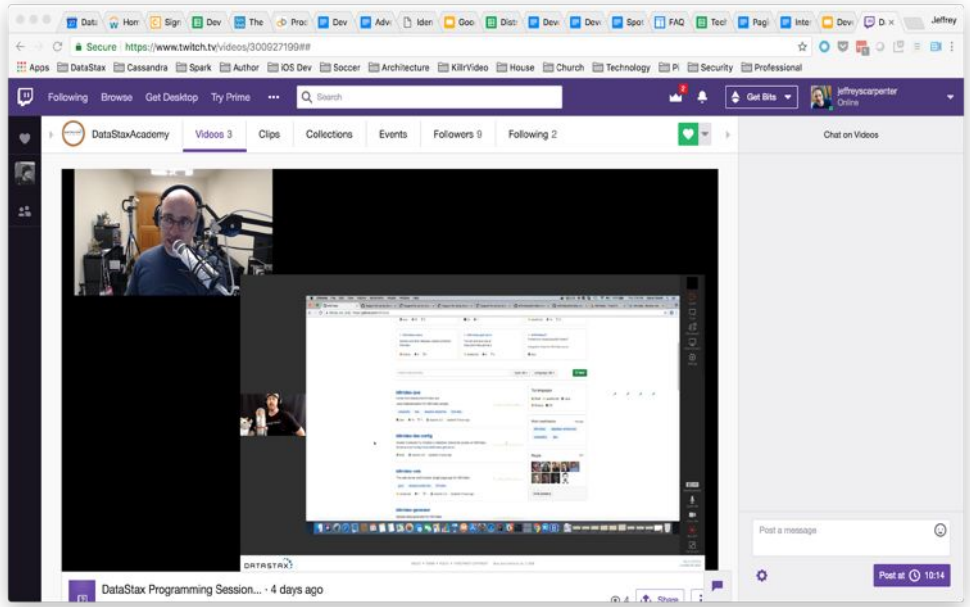
- Interview-style show featuring a mix of DataStax and industry guests
- We go in-depth on the technology and challenges of data in large-scale distributed systems
- Released weekly on DataStax Academy YouTube channel and as a podcast
- Send us your suggestions for topics and guests – we love customer use cases



DATASTAX

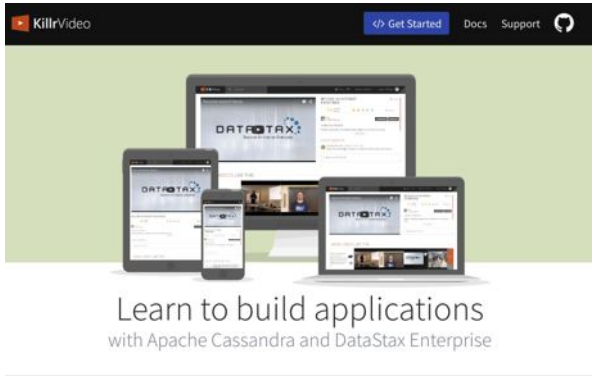
Live Coding on Twitch

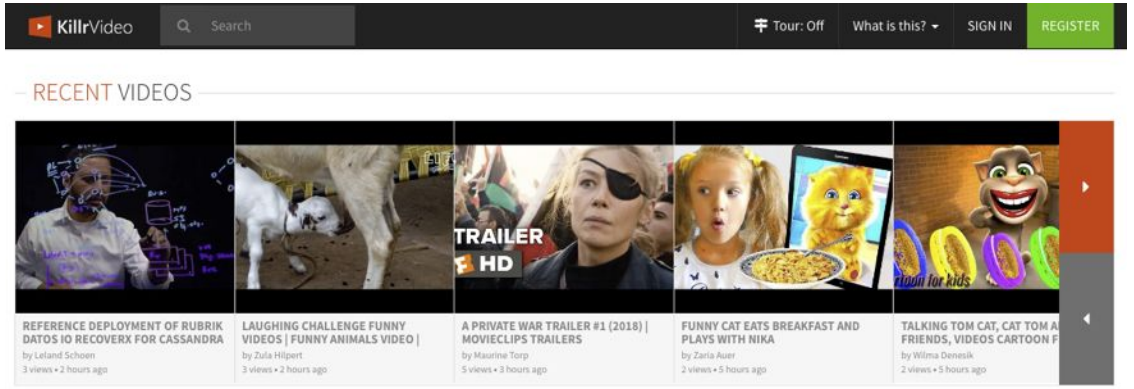
- Live coding sessions with advocates and guests each Thursday
  - https://www.twitch.tv/datastaxacademy
- Working through the challenges of building distributed systems
- Join the conversation and ask questions
- Some advocates also do streaming on personal channels



KillrVideo Reference Application

- Reference application for learning how to use Apache Cassandra and DataStax Enterprise
  - DataStax Drivers
  - Docker images
- Source code freely available
  - https://github.com/killrvideo
- Live version
  - http://killrvideo.com
- Download, test, modify, contribute!





DATASTAX

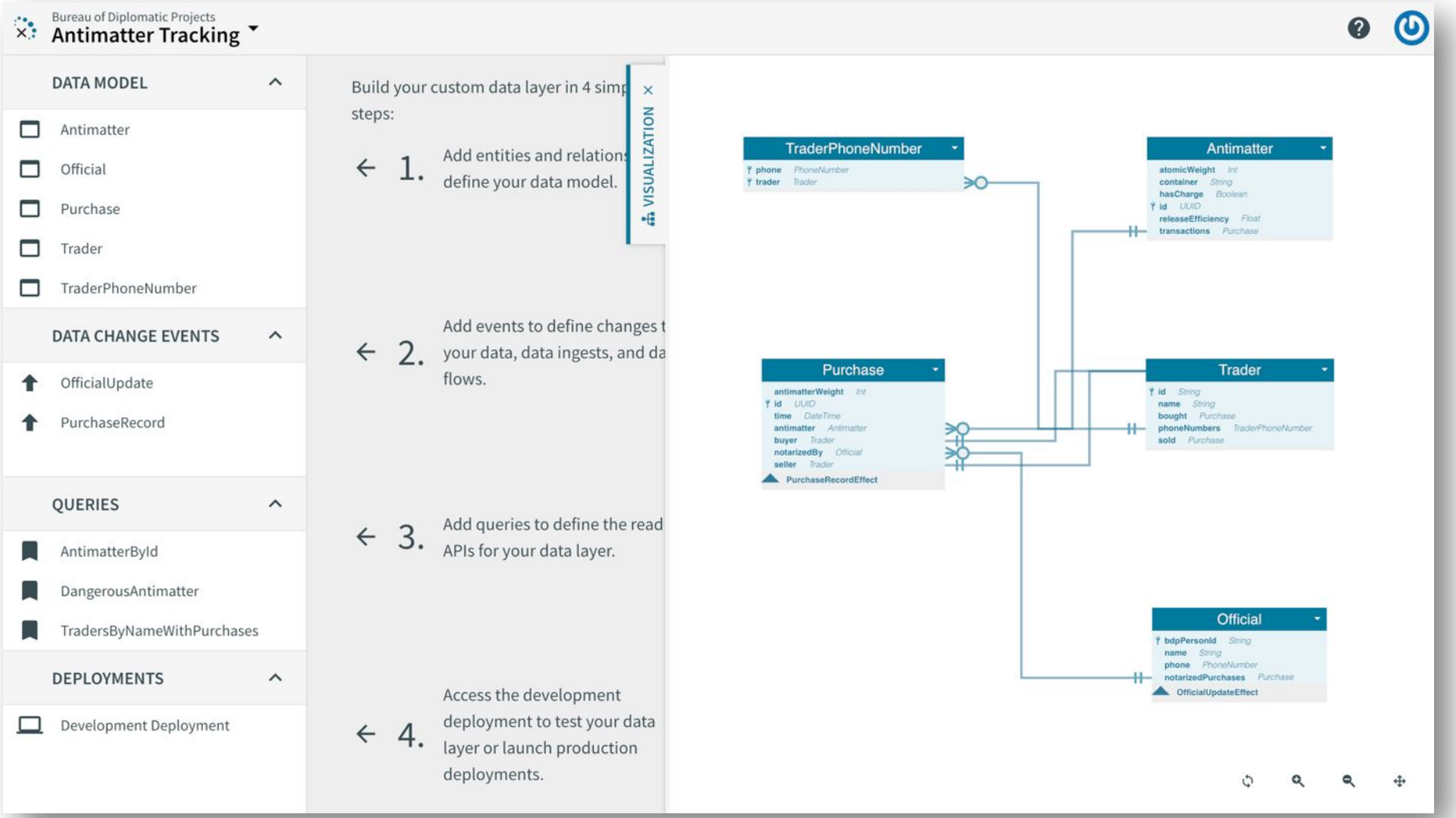
DataStax Meetup

DATASTAX

We need you

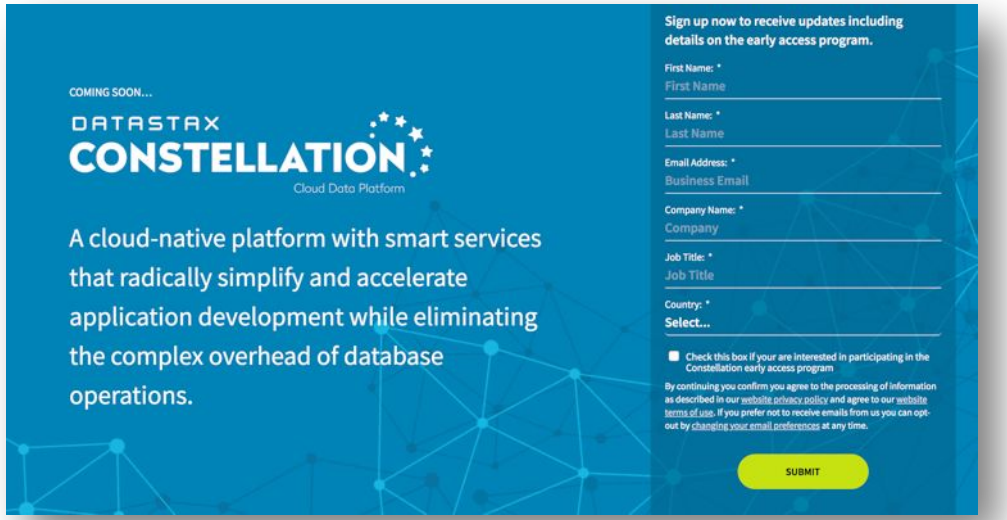
AppStax

http://run.appstax.io

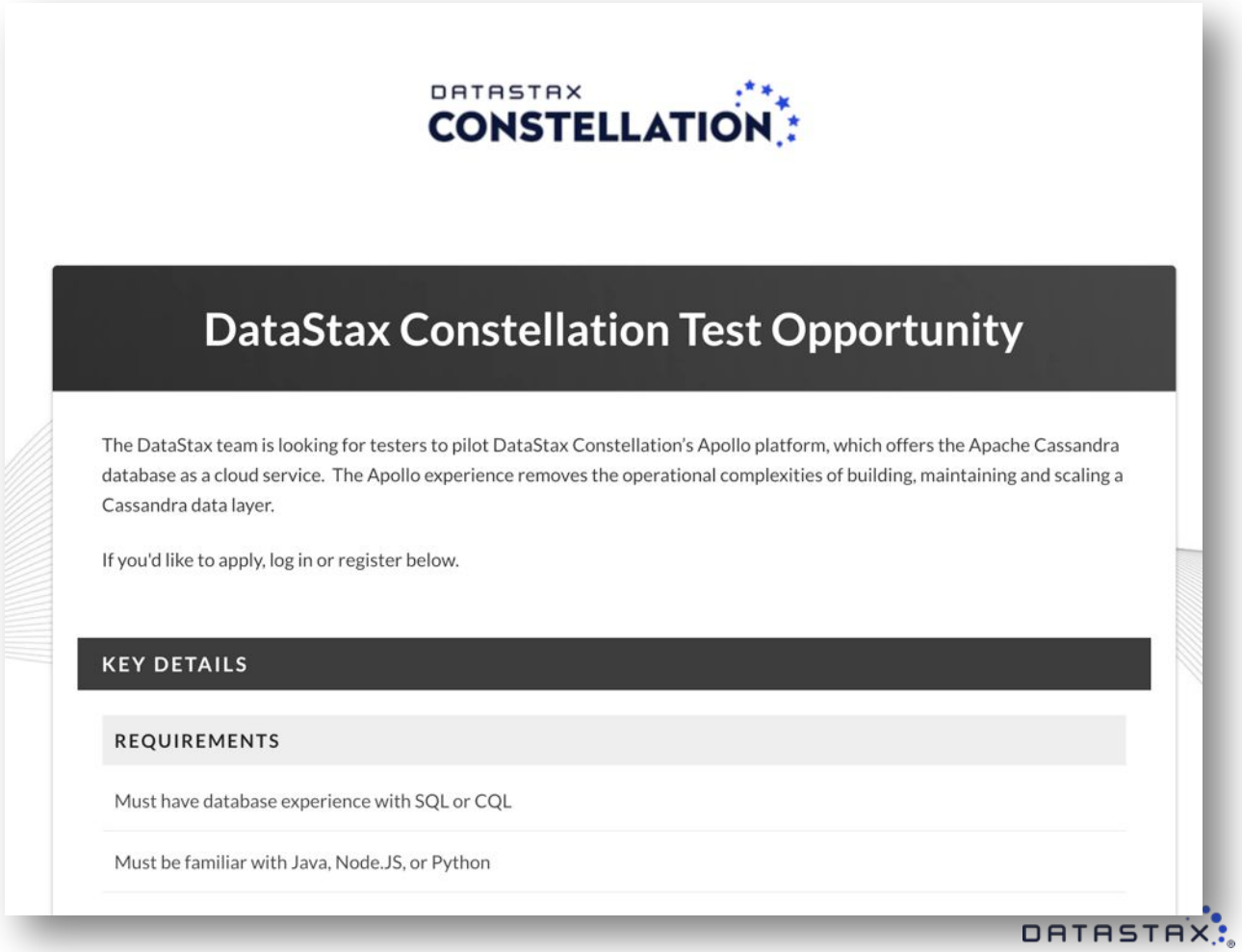


Constellation

http://constellation.datastax.com

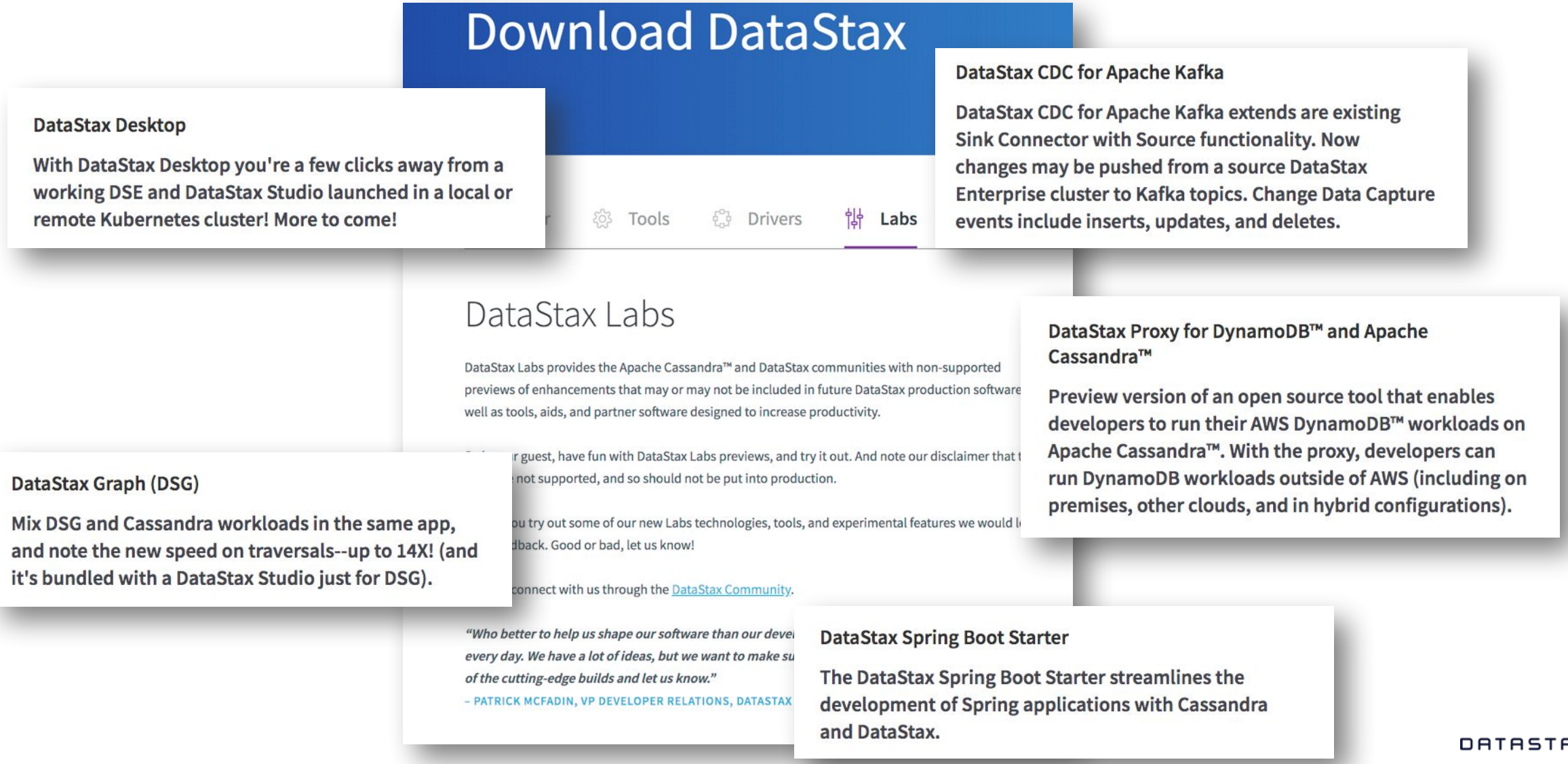


https://datastax.centercode.com/key/opportunity



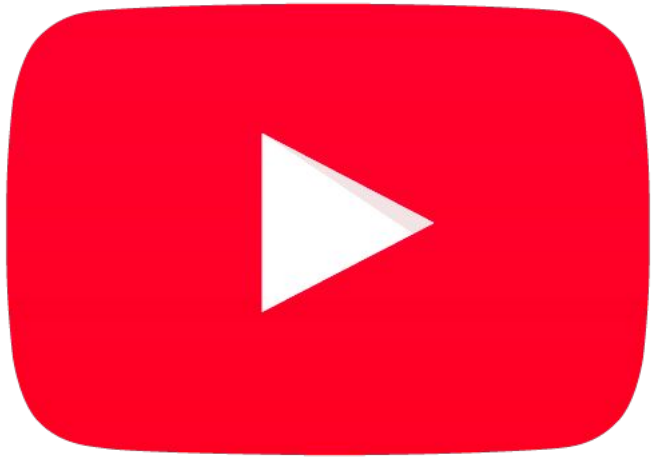
DATASTAX

DataStax Labs (https://downloads.datastax.com/#labs)



DATASTAX

Insights (https://www.datastax.com/products/datastax-insights)



https://youtu.be/iZ47rrKENuc

DATASTAX

Thank You

@CLUNVEN
@HadesArchitect