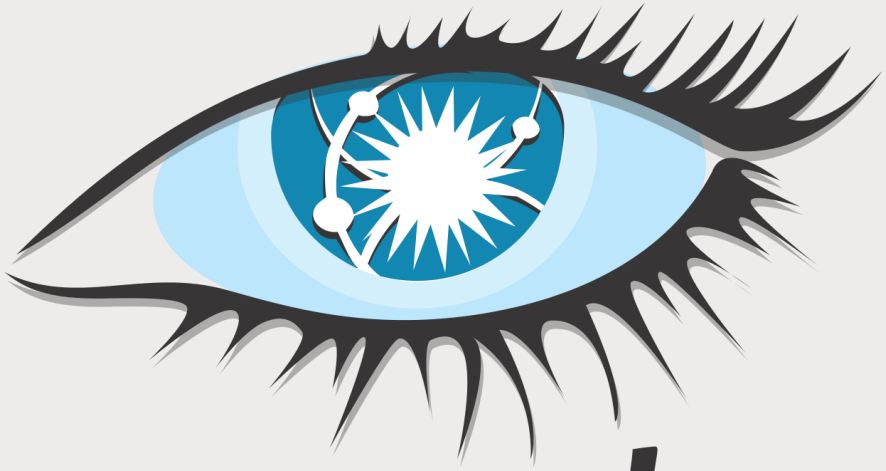
CASSANDRA

A distributed database with no single point of failure

Cassandra – NoSQL with a twist

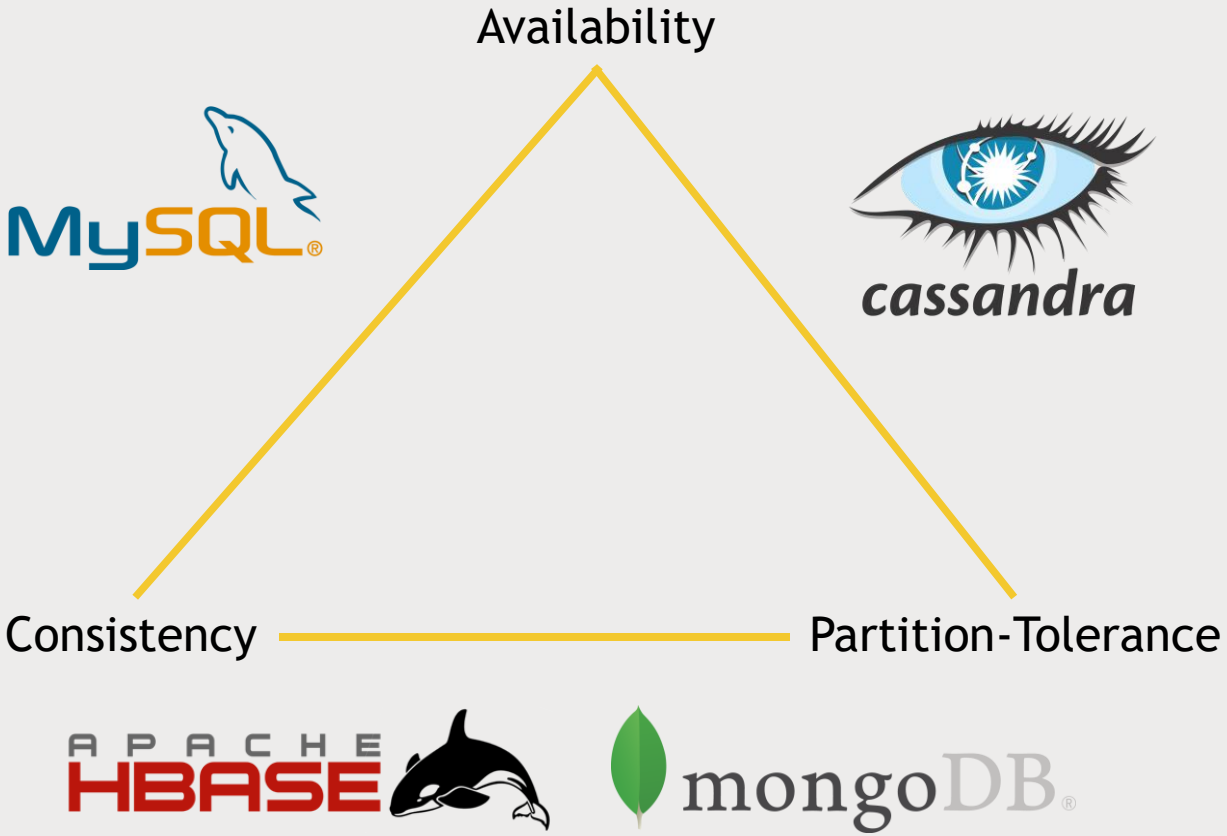
- Unlike HBase, there is no master node at all – every node runs exactly the same software and performs the same functions
- Data model is similar to BigTable / Hbase
- It's non-relational, but has a limited CQL query language as its interface



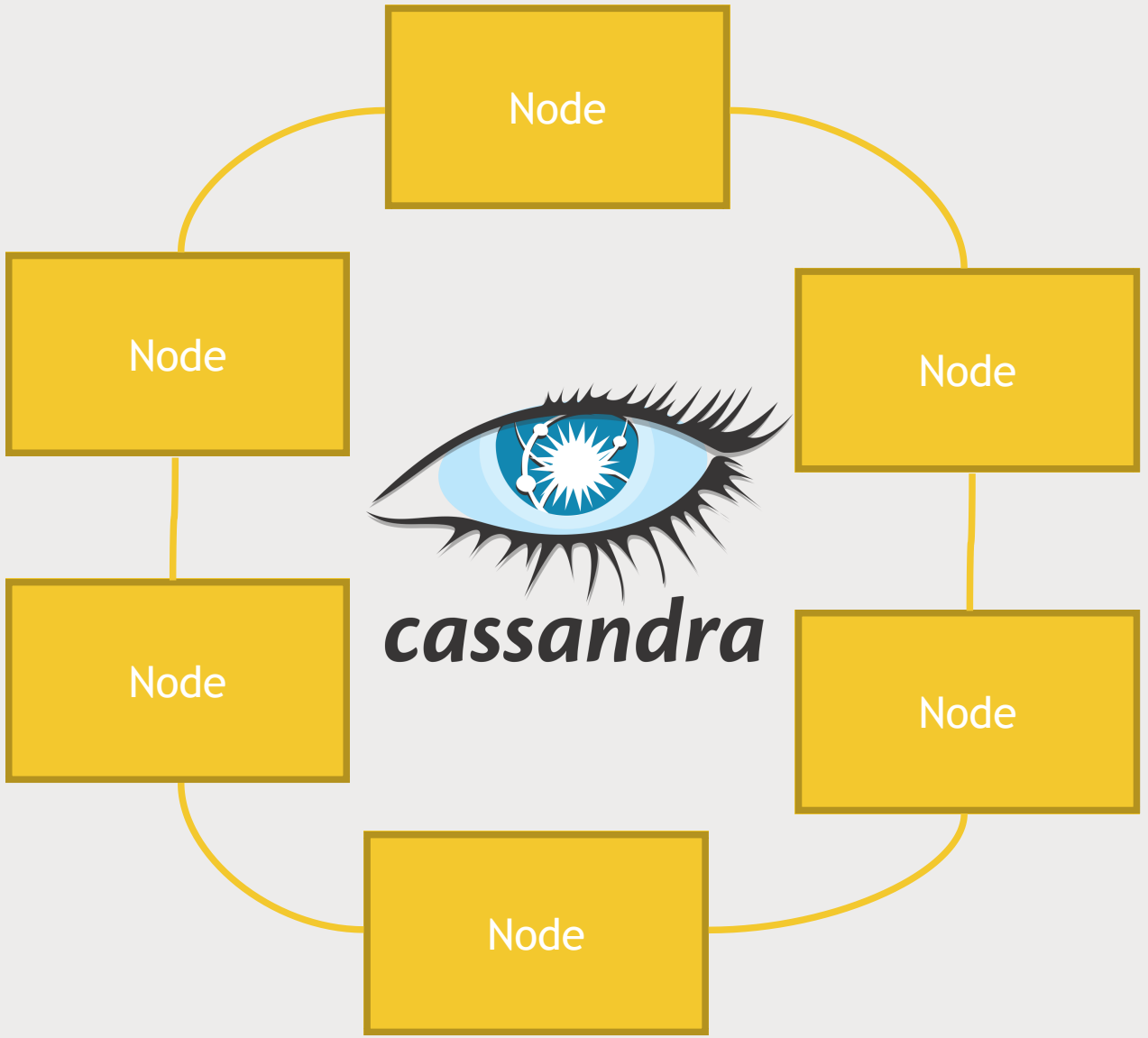
Cassandra's Design Choices

- The CAP Theorem says you can only have 2 out of 3: consistency, availability, partition-tolerance
  - *And partition-tolerance is a requirement with "big data," so you really only get to choose between consistency and availability*
- Cassandra favors availability over consistency
  - *It is "eventually consistent"*
  - *But you can specify your consistency requirements as part of your requests. So really it's "tunable consistency"*

Where Cassandra Fits in CAP tradeoffs

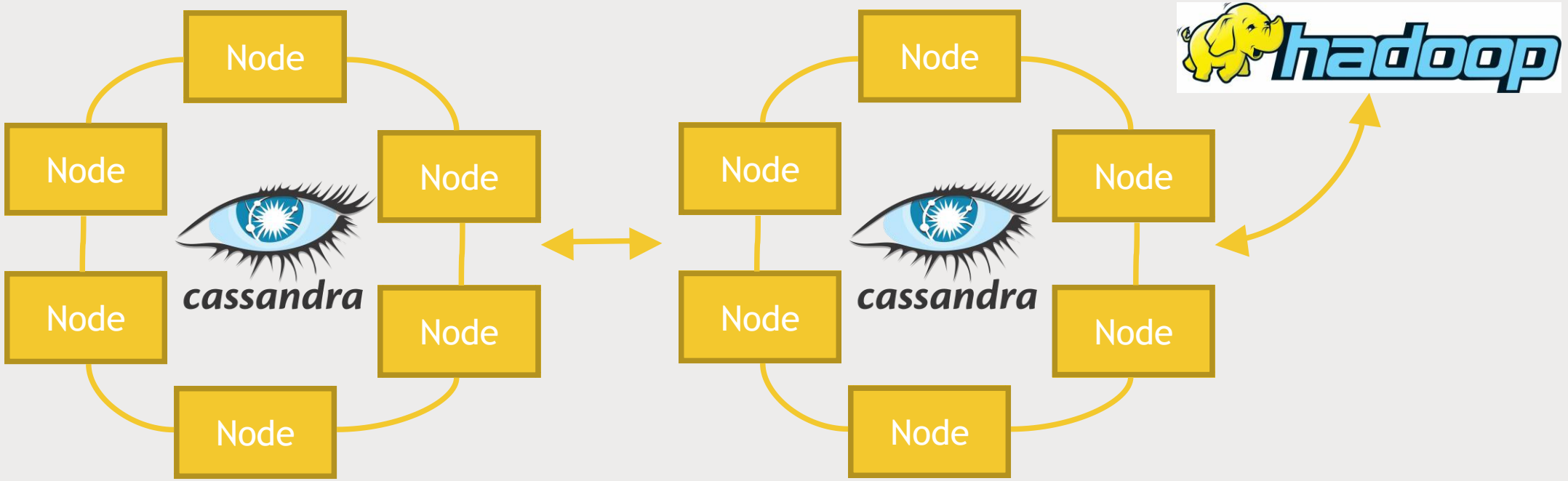


Cassandra architecture



Cassandra and your cluster

- Cassandra's great for fast access to rows of information
- Get the best of both worlds – replicate Cassandra to a another ring that is used for analytics and Spark integration



CQL (Wait, I thought this was NoSQL!)

- Cassandra's API is CQL, which makes it easy to look like existing database drivers to applications.
- CQL is like SQL, but with some big limitations!
  - *NO JOINS*
    - Your data must be de-normalized
    - So, it's still non-relational
  - *All queries must be on some primary key*
    - Secondary indices are supported, but…
- CQLSH can be used on the command line to create tables, etc.
- All tables must be in a *keyspace* – keyspaces are like databases

Cassandra and Spark

- DataStax offers a Spark-Cassandra connector
- Allows you to read and write Cassandra tables as DataFrames
- Is smart about passing queries on those DataFrames down to the appropriate level
- Use cases:
  - *Use Spark for analytics on data stored in Cassandra*
  - *Use Spark to transform data and store it into Cassandra for transactional use*

Let’s Play

- Install Cassandra on our virtual Hadoop node
- Set up a table for MovieLens users
- Write into that table and query it from Spark!