



# INTEGRATING MYSQL & HADOOP

Fun with Sqoop

# What's MySQL?

- Popular, free relational database
- Generally monolithic in nature
- But, can be used for OLTP - so exporting data into MySQL can be useful
- Existing data may exist in MySQL that you want to import to Hadoop

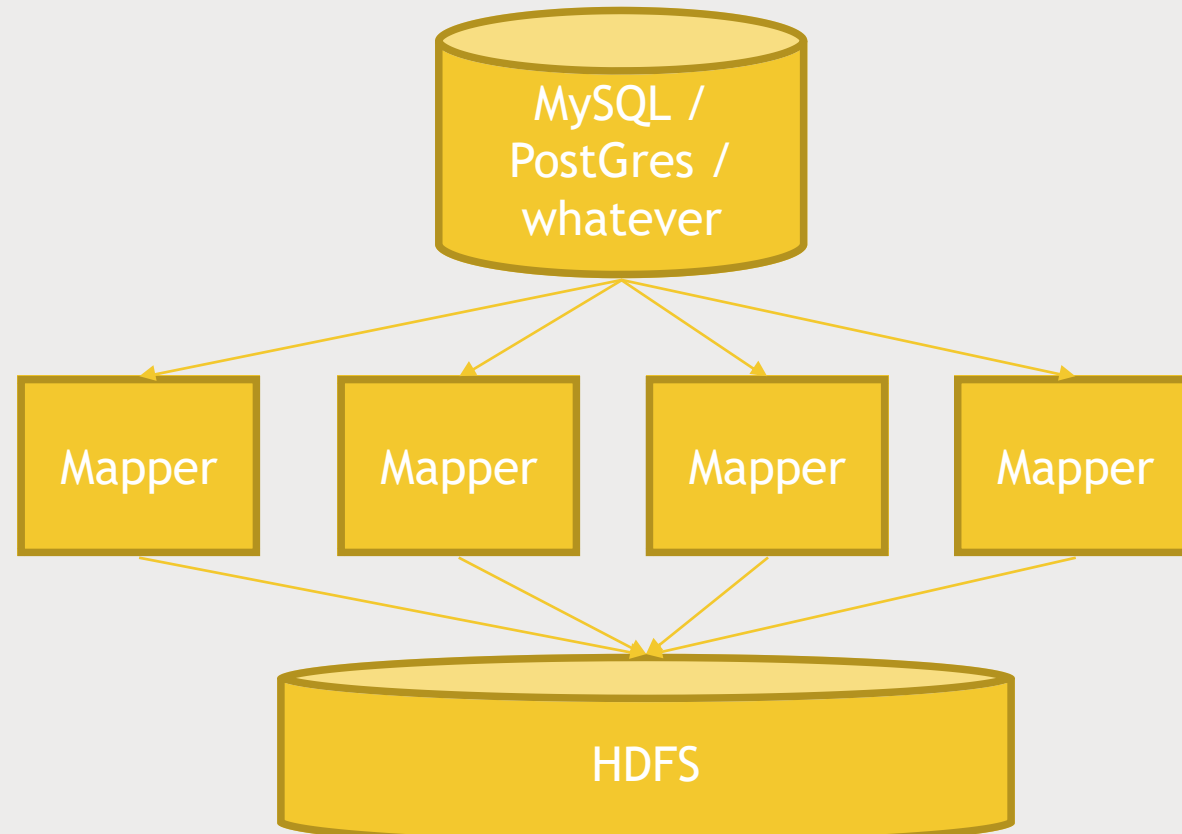


# Sqoop to the rescue



# Sqoop can handle BIG data

- Actually kicks off MapReduce jobs to handle importing or exporting your data!



# Sqoop: Import data from MySQL to HDFS

```
sqoop import --connect jdbc:mysql://localhost/movielens --driver  
com.mysql.jdbc.Driver --table movies
```



or any other jdbc connected database

# Sqoop: Import data from MySQL directly into Hive!

- `sqoop import --connect jdbc:mysql://localhost/movielens --driver com.mysql.jdbc.Driver --table movies --hive-import`



# Incremental imports

- You can keep your relational database and Hadoop in sync
- `--check-column` and `--last-value` for example: to import data later than a given timestamp



# Sqoop: Export data from Hive to MySQL

- `sqoop export --connect jdbc:mysql://localhost/movielens -m 1 --driver com.mysql.jdbc.Driver --table exported_movies --export-dir /apps/hive/warehouse/movies --input-fields-terminated-by '\0001'`
- Target table must already exist in MySQL, with columns in expected order



# Let's play with MySQL and Sqoop

- Import MovieLens data into a MySQL database
- Import the movies to HDFS
- Import the movies into Hive
- Export the movies back into MySQL

