COLLABORATE15

TECHNOLOGY AND APPLICATIONS FORUM FOR THE ORACLE COMMUNITY

Architectural Considerations for Data Warehousing with Hadoop

Session ID#: 10251

github.com/hadooparchitecturebook/hadoop-arch-book/tree/master/ch11-data-warehousing

Prepared by:
Mark Grover, Software Engineer
Jonathan Seidman, Solutions Architect
Cloudera, Inc.

@mark_grover

@jseidman

REMINDER

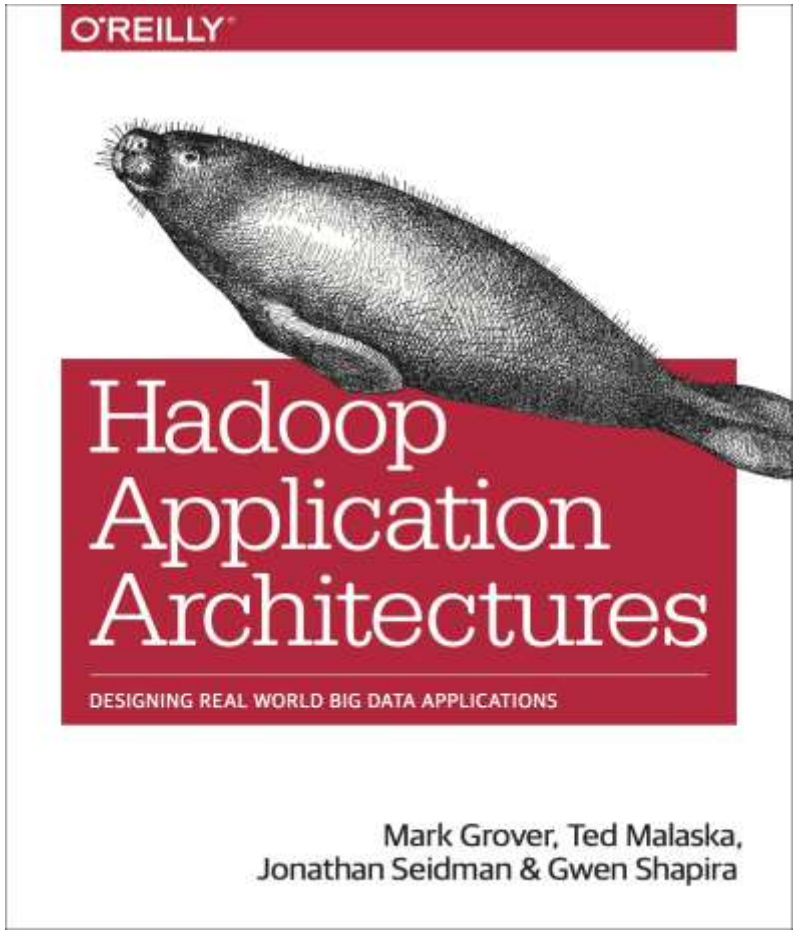
Check in on the COLLABORATE mobile app

About Us

- Mark
  - Software Engineer at Cloudera
  - Committer on Apache Bigtop, PMC member on Apache Sentry (incubating)
  - Contributor to Apache Hadoop, Spark, Hive, Sqoop, Pig and Flume
- Jonathan
  - Senior Solutions Architect/Partner Engineering at Cloudera
  - Previously, Technical Lead on the big data team at Orbitz Worldwide
  - Co-founder of the Chicago Hadoop User Group and Chicago Big Data

About the Book

- @hadooparchbook
- hadooparchitecturebook.com
- github.com/hadooparchitecturebook
- slideshare.com/hadooparchbook

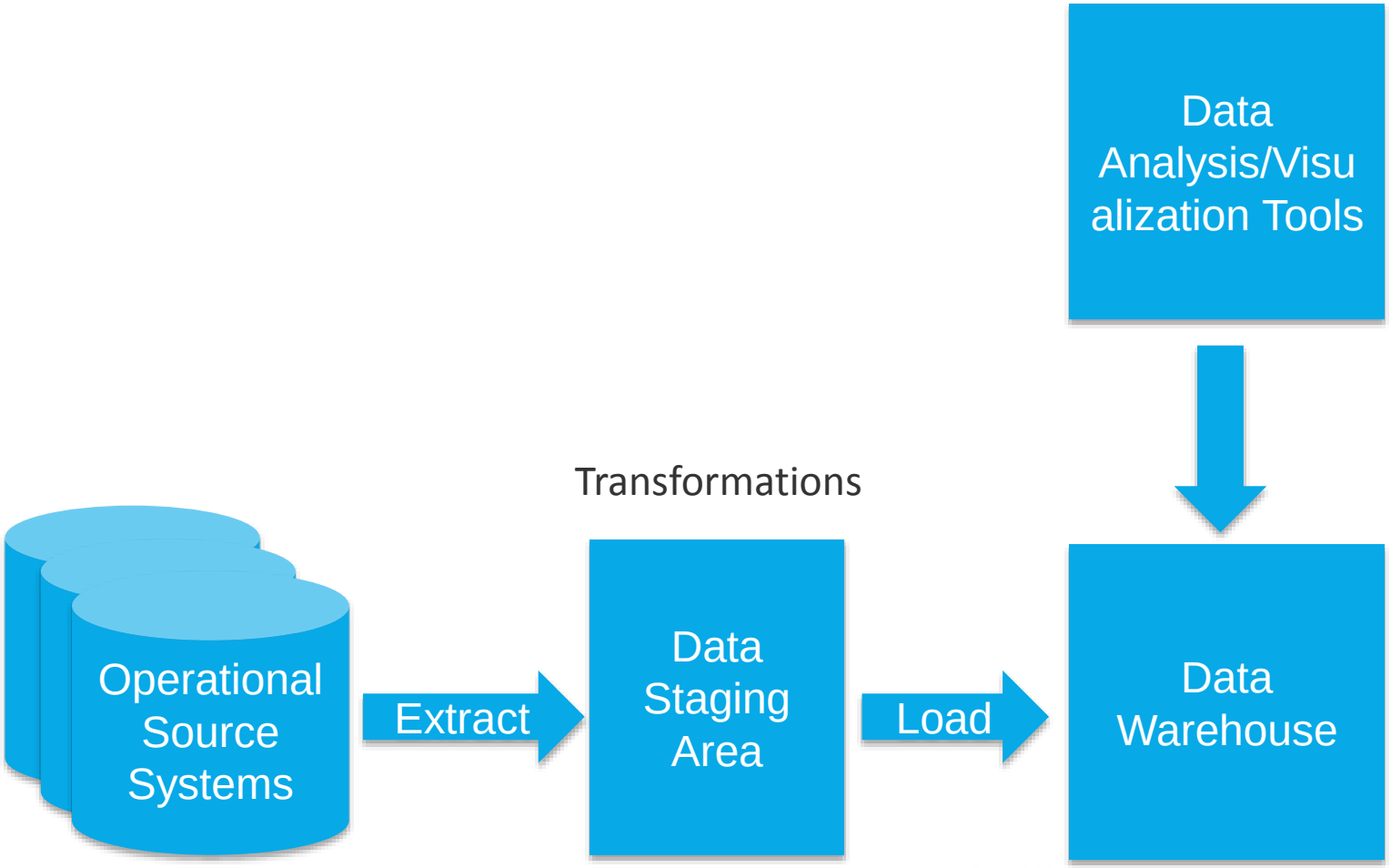


Agenda

- Typical data warehouse architecture.
- Challenges with the existing data warehouse architecture.
- How Hadoop complements an existing data warehouse architecture.
- (Very) Brief intro to Hadoop.
- Example use case.
- Walkthrough of example use case implementation.

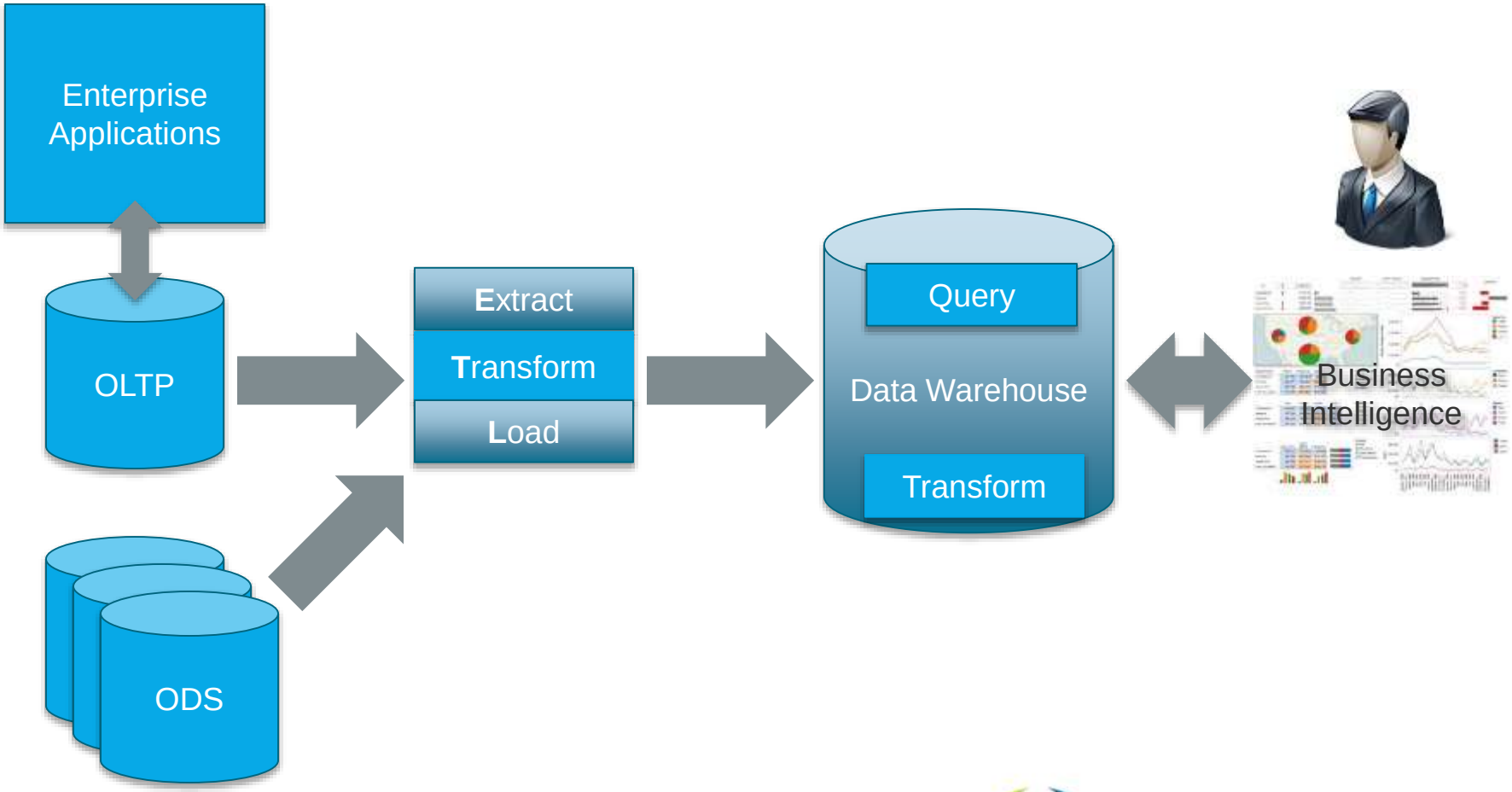
Typical Data Warehouse Architecture

Example High Level Data Warehouse Architecture



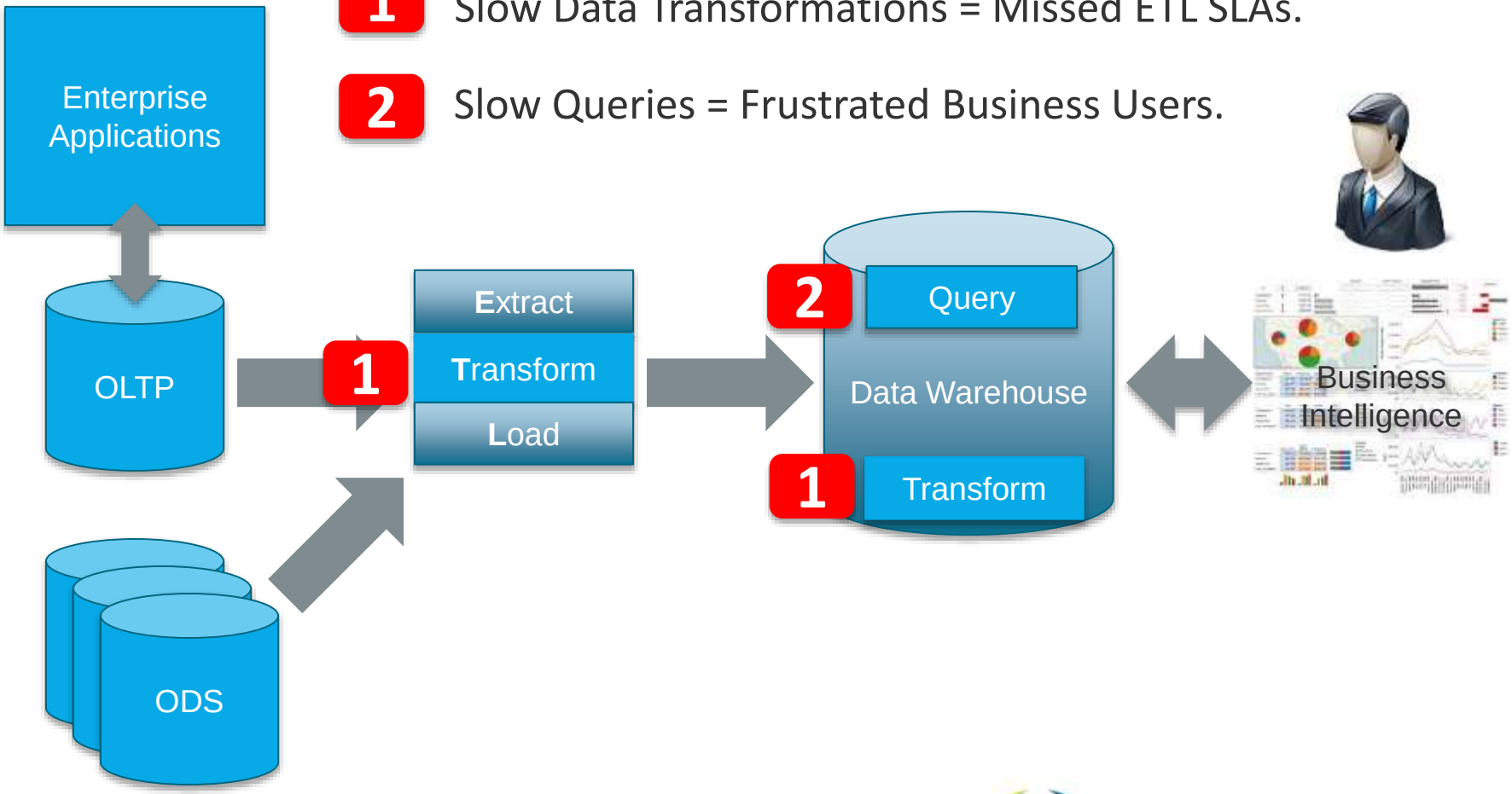
Challenges with the Data Warehouse Architecture

Challenge – ETL/ELT Processing

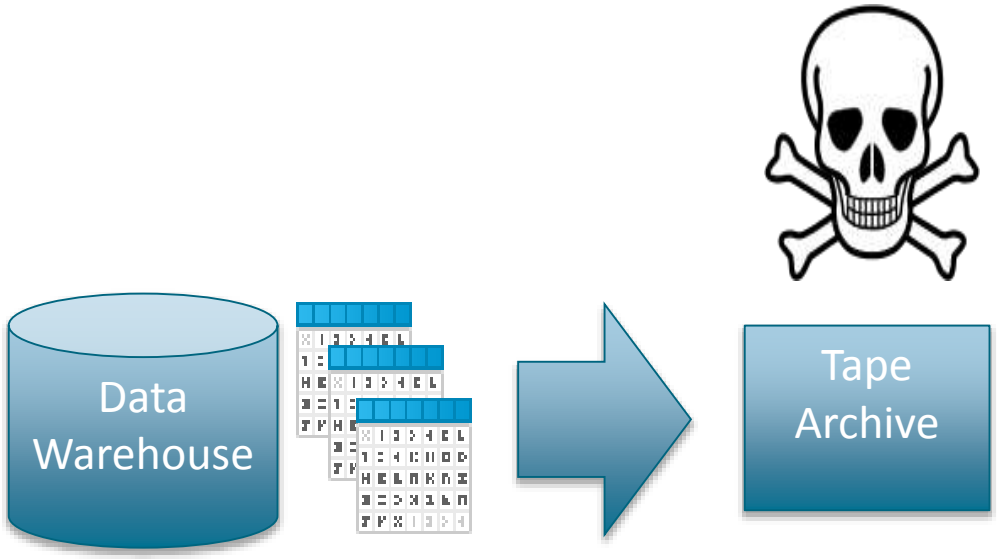


Challenges – ETL/ELT Processing

1. Slow Data Transformations = Missed ETL SLAs.
2. Slow Queries = Frustrated Business Users.

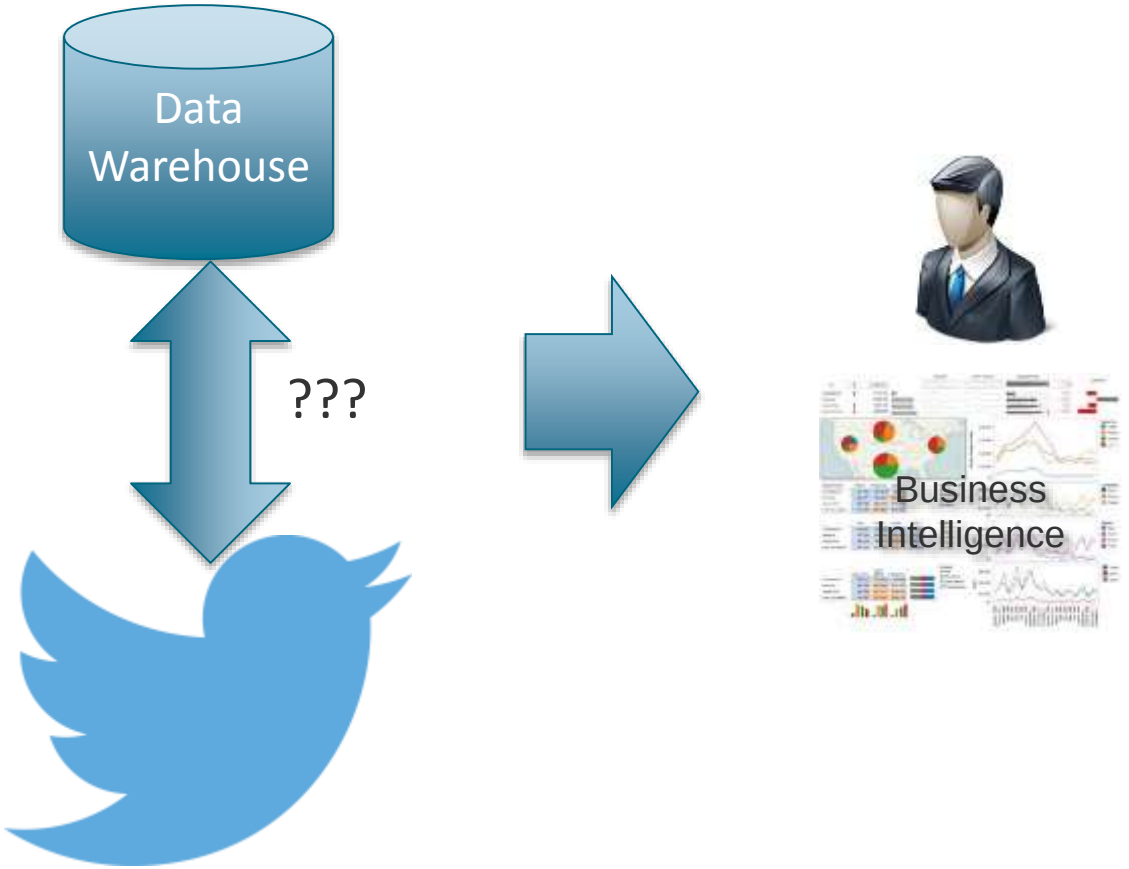


Challenges – Data Archiving



- Full-fidelity data only kept for a short duration
- Expensive or sometimes impossible to look at historical raw data

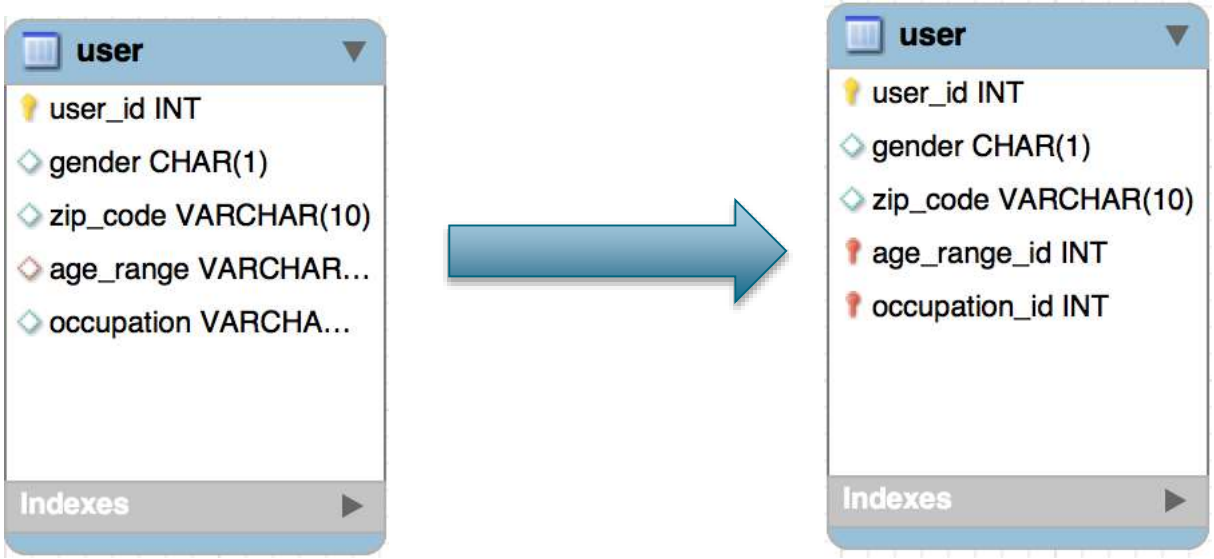
Challenge – Disparate Data Sources



- How do you join data from disparate sources with EDW?

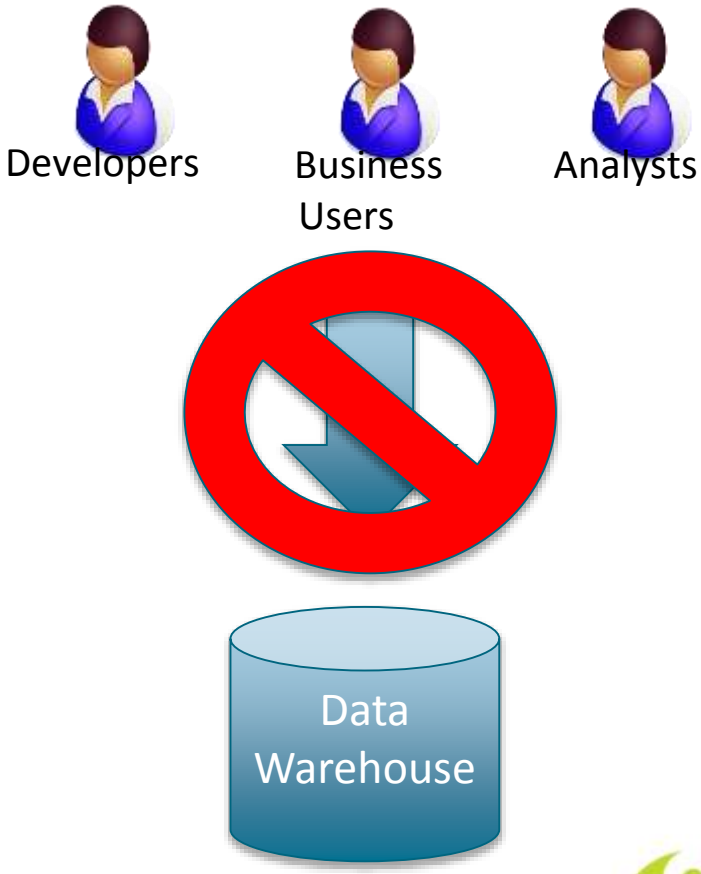
Challenge – Lack of Agility

- Responding to changing requirements, mistakes, etc. requires lengthy processes.



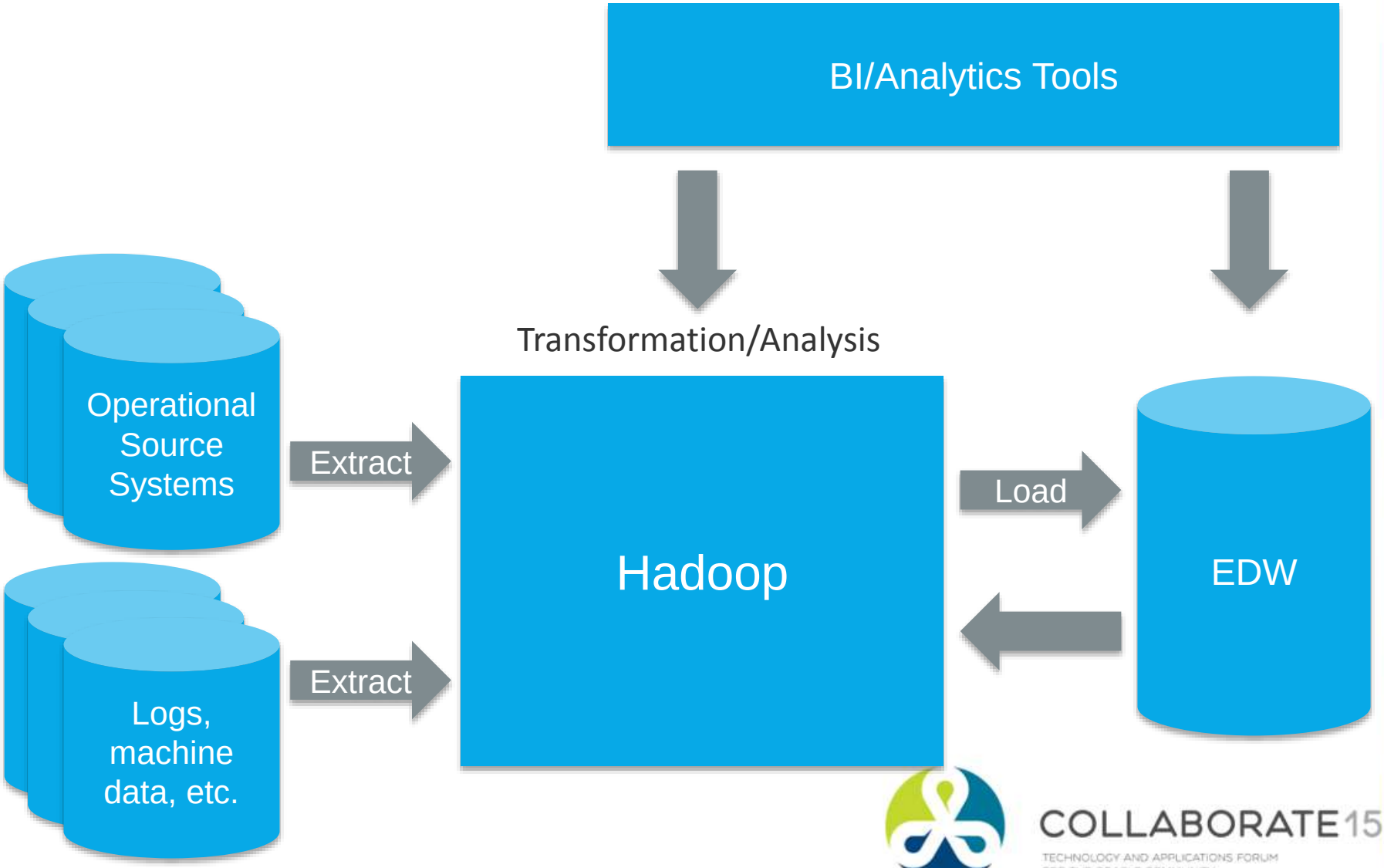
Challenge – Exploratory Analysis in the EDW

- Difficult for users to do exploratory analysis of data in the data warehouse.

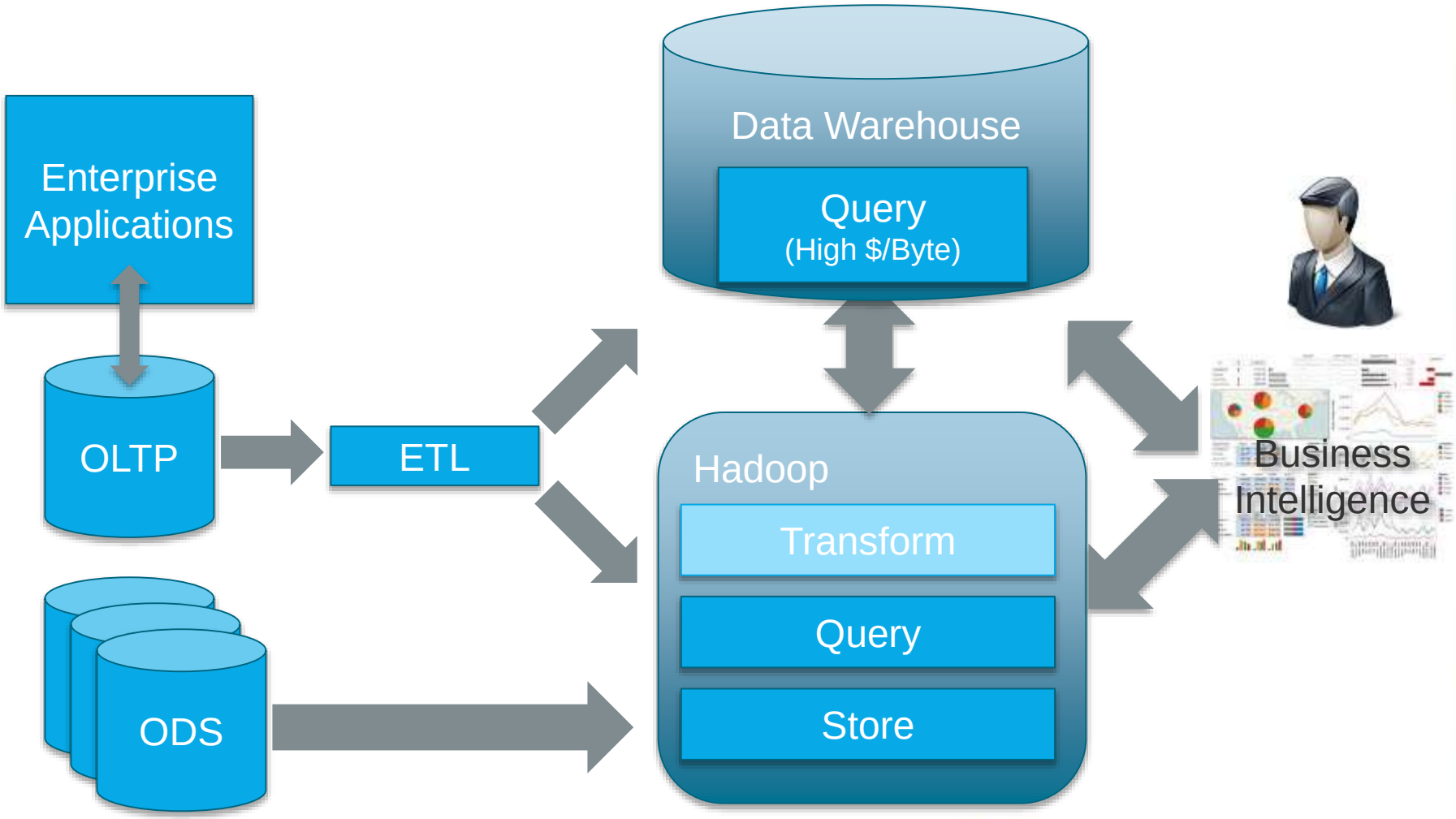


Complementing the EDW with Hadoop

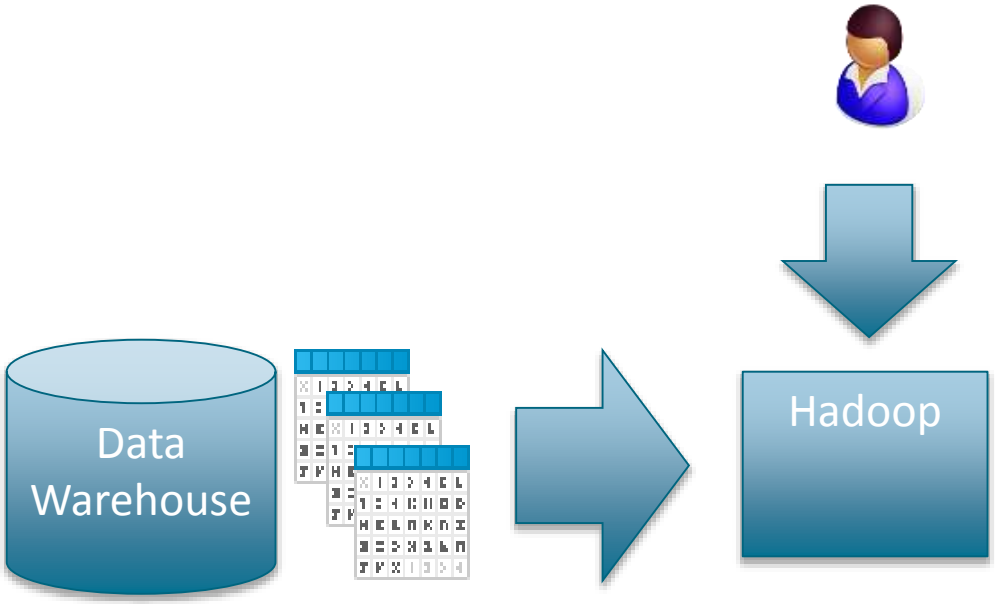
Data Warehouse Architecture with Hadoop



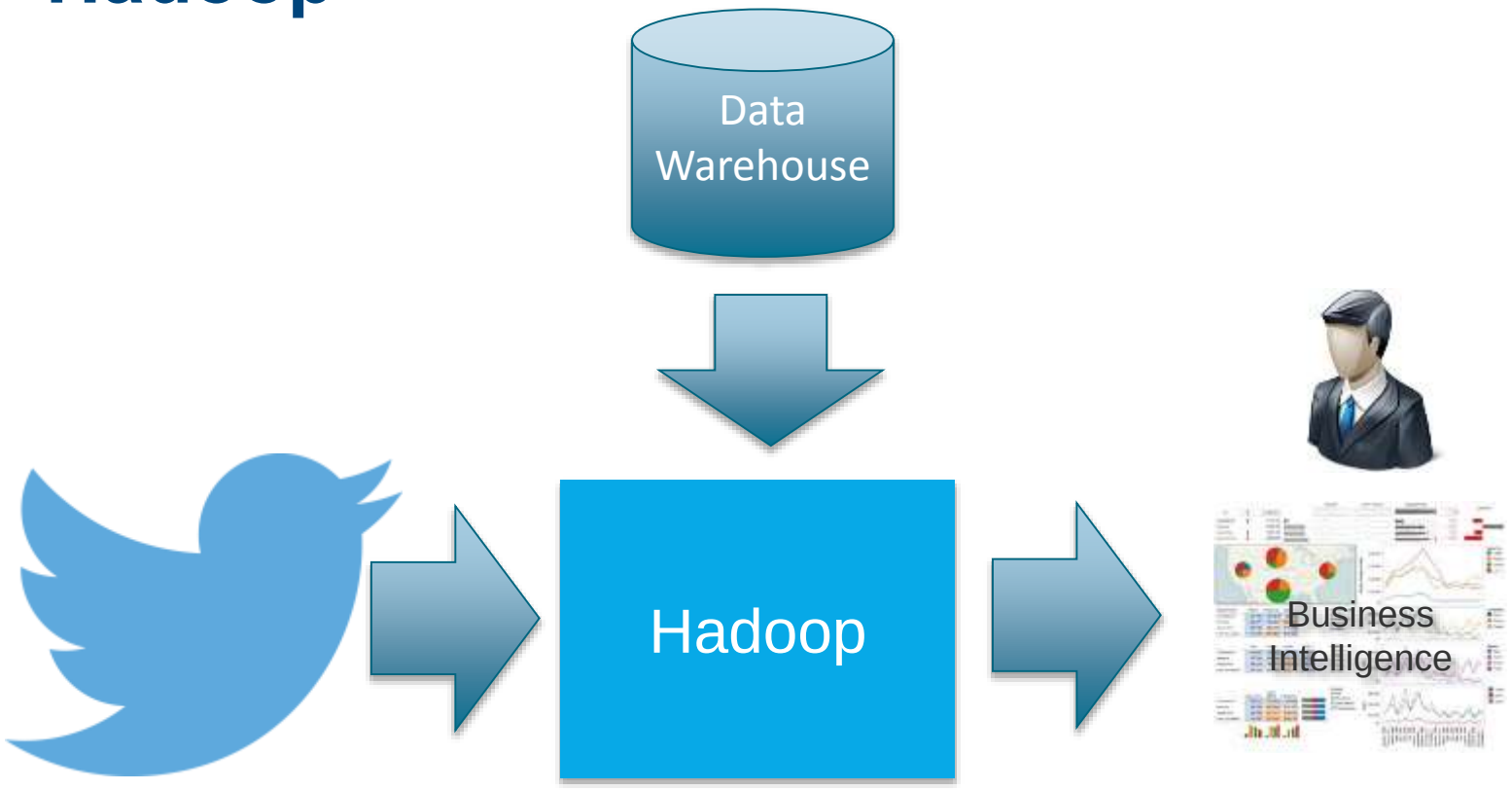
ETL/ELT Optimization with Hadoop



Active Archiving with Hadoop



Joining Disparate Data Sources with Hadoop



Agile Data Access with Hadoop

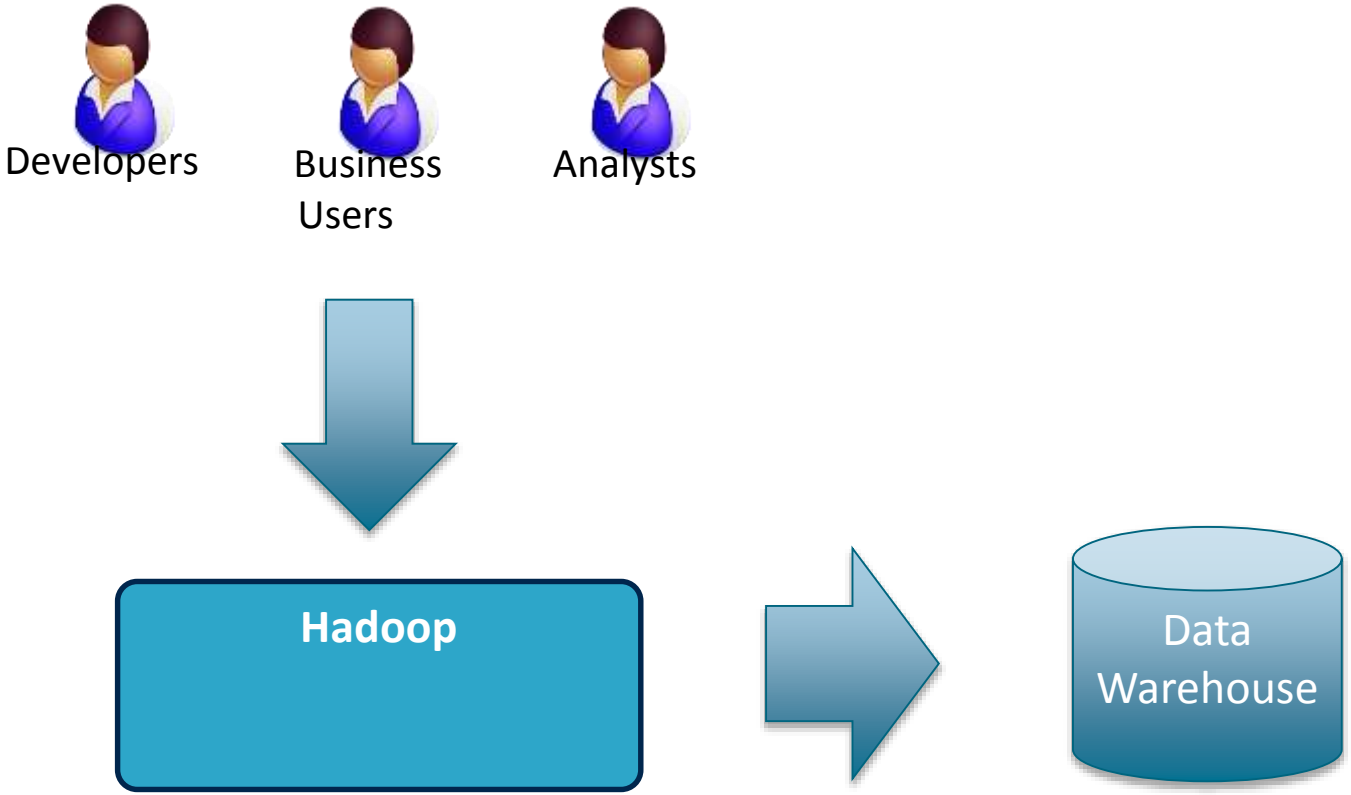
Schema-on-Write (RDBMS):

- **Prescriptive Data Modeling:**
  - Create static DB schema
  - Transform data into RDBMS
  - Query data in RDBMS format
- New columns must be added explicitly before new data can propagate into the system.
- **Good for Known Unknowns (Repetition)**

Schema-on-Read (Hadoop):

- **Descriptive Data Modeling:**
  - Copy data in its native format
  - Create schema + parser
  - Query Data in its native format
- New data can start flowing any time and will appear retroactively once the schema/parser properly describes it.
- **Good for Unknown Unknowns (Exploration)**

Exploratory Analysis with Hadoop



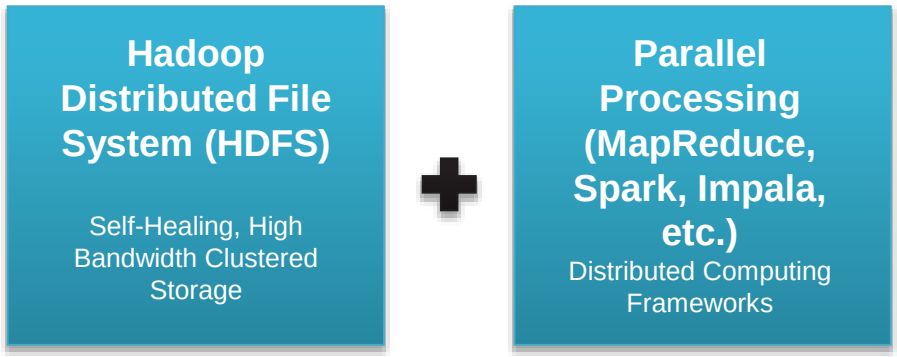
A Very Brief Intro to Hadoop

What is Apache Hadoop?

Apache Hadoop is an open source platform for data storage and processing that is…

- ✓ Scalable
- ✓ Fault tolerant
- ✓ Distributed

CORE HADOOP SYSTEM COMPONENTS



Has the Flexibility to Store and Mine Any Type of Data

- Ask questions across structured and unstructured data that were previously impossible to ask or solve
- Not bound by a single schema

Excels at Processing Complex Data

- Scale-out architecture divides workloads across multiple nodes
- Flexible file system eliminates ETL bottlenecks

Scales Economically

- Can be deployed on commodity hardware
- Open source platform guards against vendor lock

Oracle Big Data Appliance

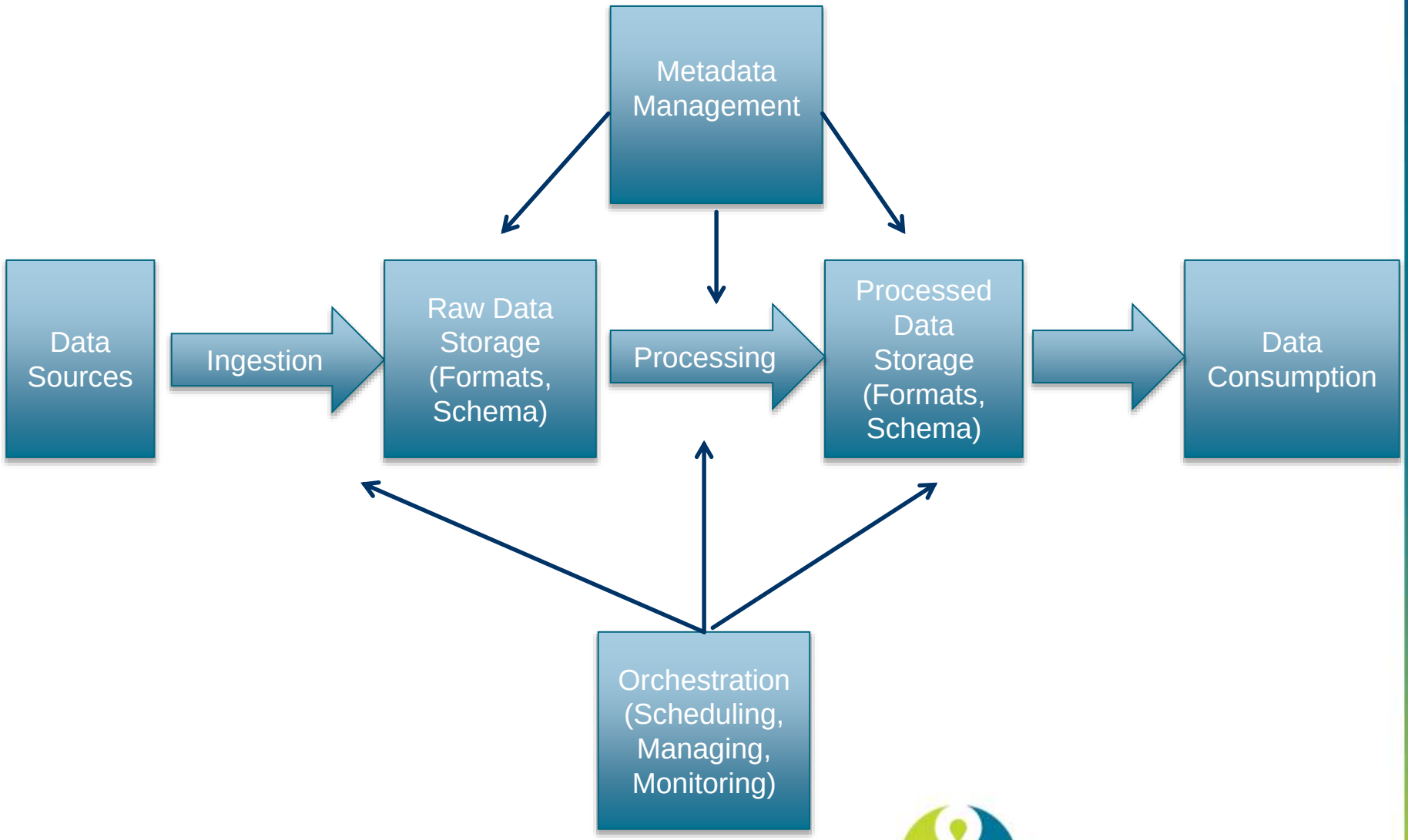
- All of the capabilities we're talking about here are available as part of the Oracle BDA.

Challenges of Hadoop Implementation

Challenges of Hadoop Implementation

COLLABORATE15
TECHNOLOGY AND APPLICATIONS FORUM FOR THE ORACLE COMMUNITY

Other Challenges – Architectural Considerations



Hadoop Third Party Ecosystem

Applications

Operational Tools

Data Systems

Infrastructure

Walkthrough of Example Use Case

Use-case

- Movielens dataset
- Users register by entering some demographic information
  - Users can update demographic information later on
- Rate movies
  - Ratings can be updated later on
- Auxillary information about movies available
  - e.g. release date, IMDB URL, etc.

Movielens data set

u.user

```
user id | age | gender | occupation | zip code
1|24|M|technician|85711
2|53|F|other|94043
3|23|M|writer|32067
4|24|M|technician|43537
5|33|F|other|15213
6|42|M|executive|98101
7|57|M|administrator|91344
```

Movielens data set

u.item

```
movie id | movie title | release date | video release date
| IMDb URL | unknown | Action | Adventure | Animation
|Children's | Comedy | Crime | Documentary | Drama |
Fantasy | Film-Noir | Horror | Musical | Mystery | Romance
| Sci-Fi | Thriller | War | Western |

1|Toy Story (1995)|01-Jan-
1995||http://us.imdb.com/M/title-
exact?Toy%20Story%20(1995)|0|0|0|1|1|1|0|0|0|0|0|0|0|0|0
|0|0|0
2|GoldenEye (1995)|01-Jan-
1995||http://us.imdb.com/M/title-
exact?GoldenEye%20(1995)|0|1|1|0|0|0|0|0|0|0|0|0|0|0|1
|0|0
3|Four Rooms (1995)|01-Jan-
1995||http://us.imdb.com/M/title-
exact?Four%20Rooms%20(1995)|0|0|0|0|0|0|0|0|0|0|0|0|0|
0|1|0|0
```

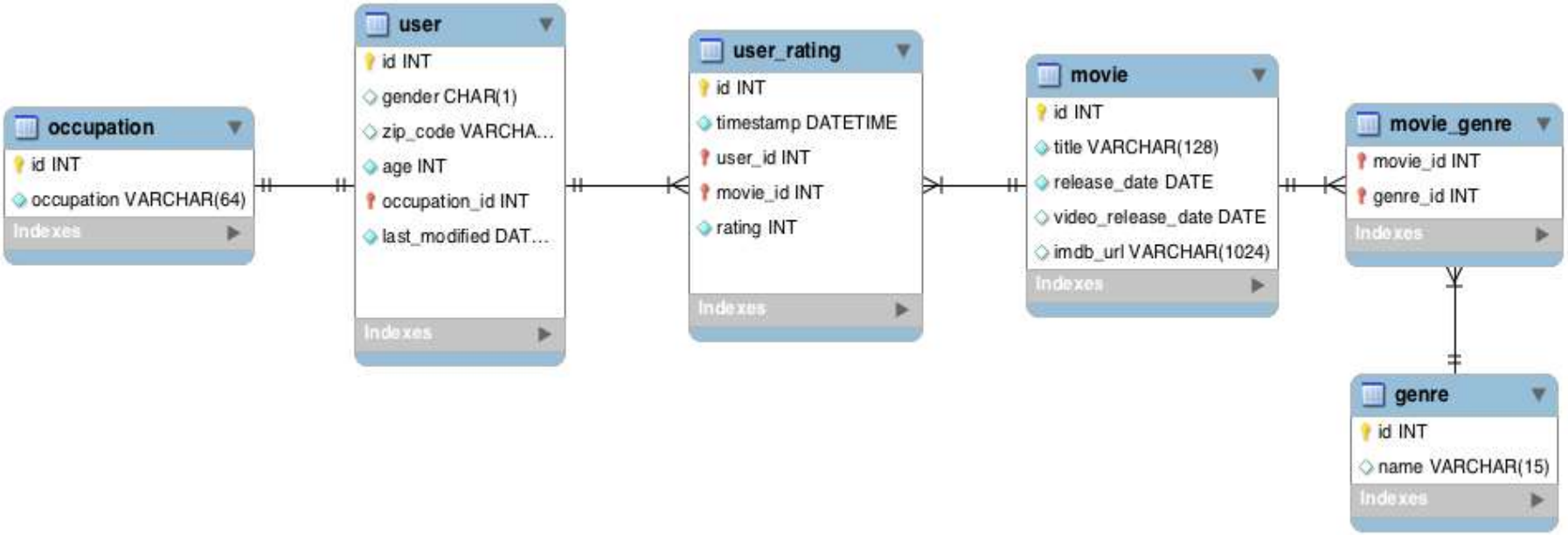
Movielens data set

u.data

```
user id | item id | rating | timestamp
196|242|3|881250949
186|302|3|891717742
22|377|1|878887116
244 51|2|880606923
166|346|1|886397596
```

OLTP schema

Movielens data set - OLTP



Data Modeling

Data Modeling Considerations

- We need to consider the following in our architecture:
  - Storage layer – HDFS? HBase? Etc.
  - File system schemas – how will we lay out the data?
  - File formats – what storage formats to use for our data, both raw and processed data?
  - Data compression formats?
- Hadoop is not a database, so these considerations will be different from an RDBMS.

Denormalization

- Why denormalize?
- When to do denormalize?
- How much to denormalize?

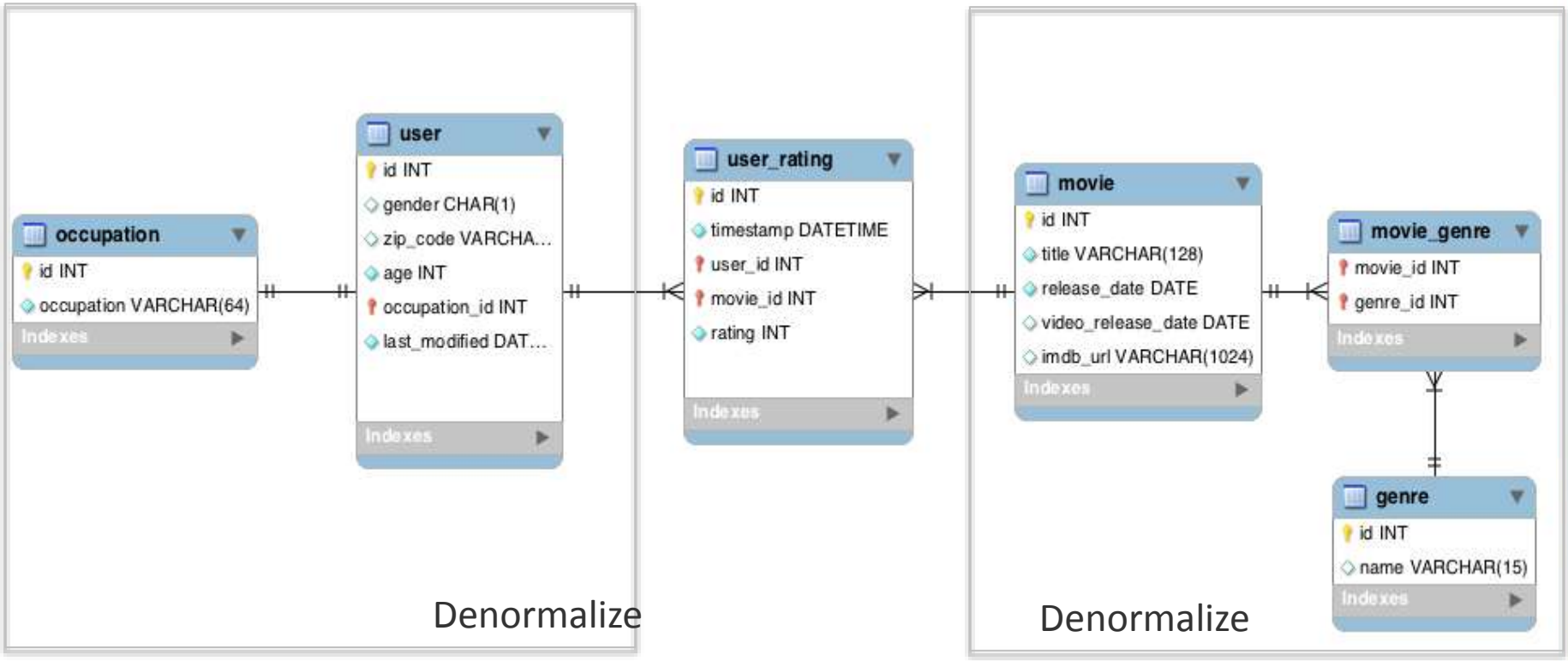
Why Denormalize?

- Regular joins are expensive in Hadoop
- When you have 2 data sets, no guarantees that corresponding records will be present on the same
- Such a guarantee exists when storing such data in a single data set

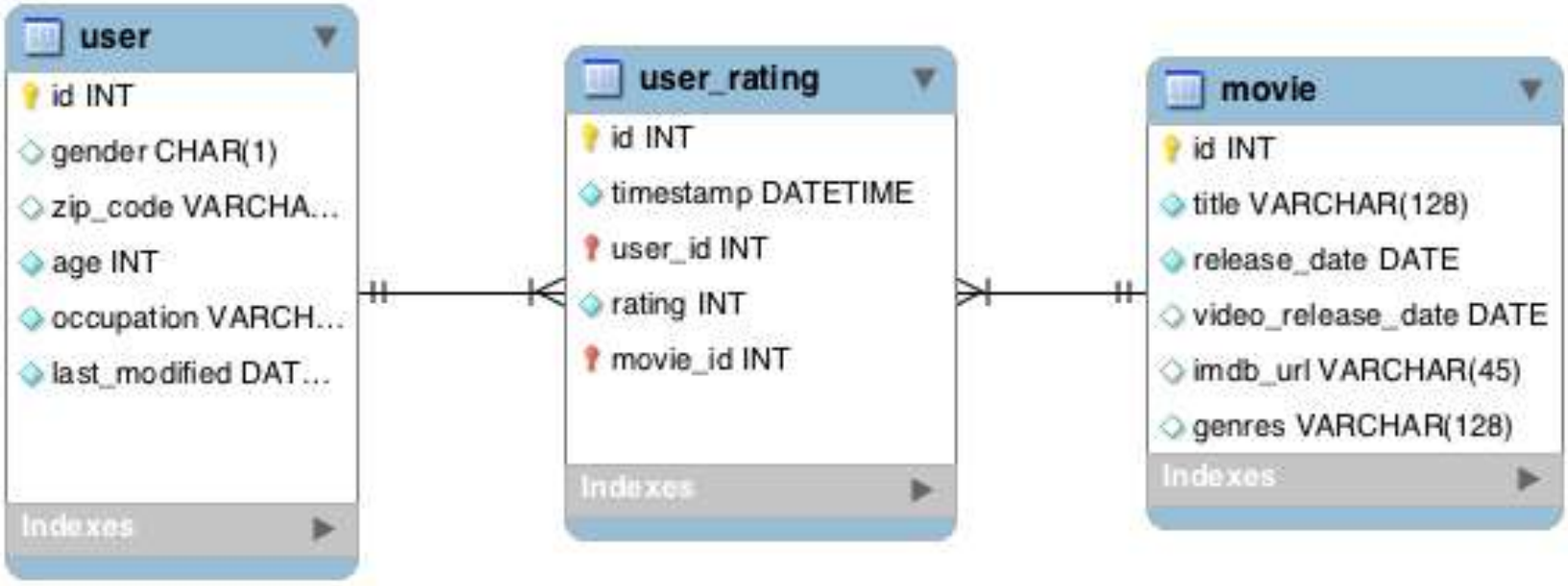
When to Denormalize?

- ~~Well, it's difficult to say~~
- It depends

Movielens Data Set - Denormalization



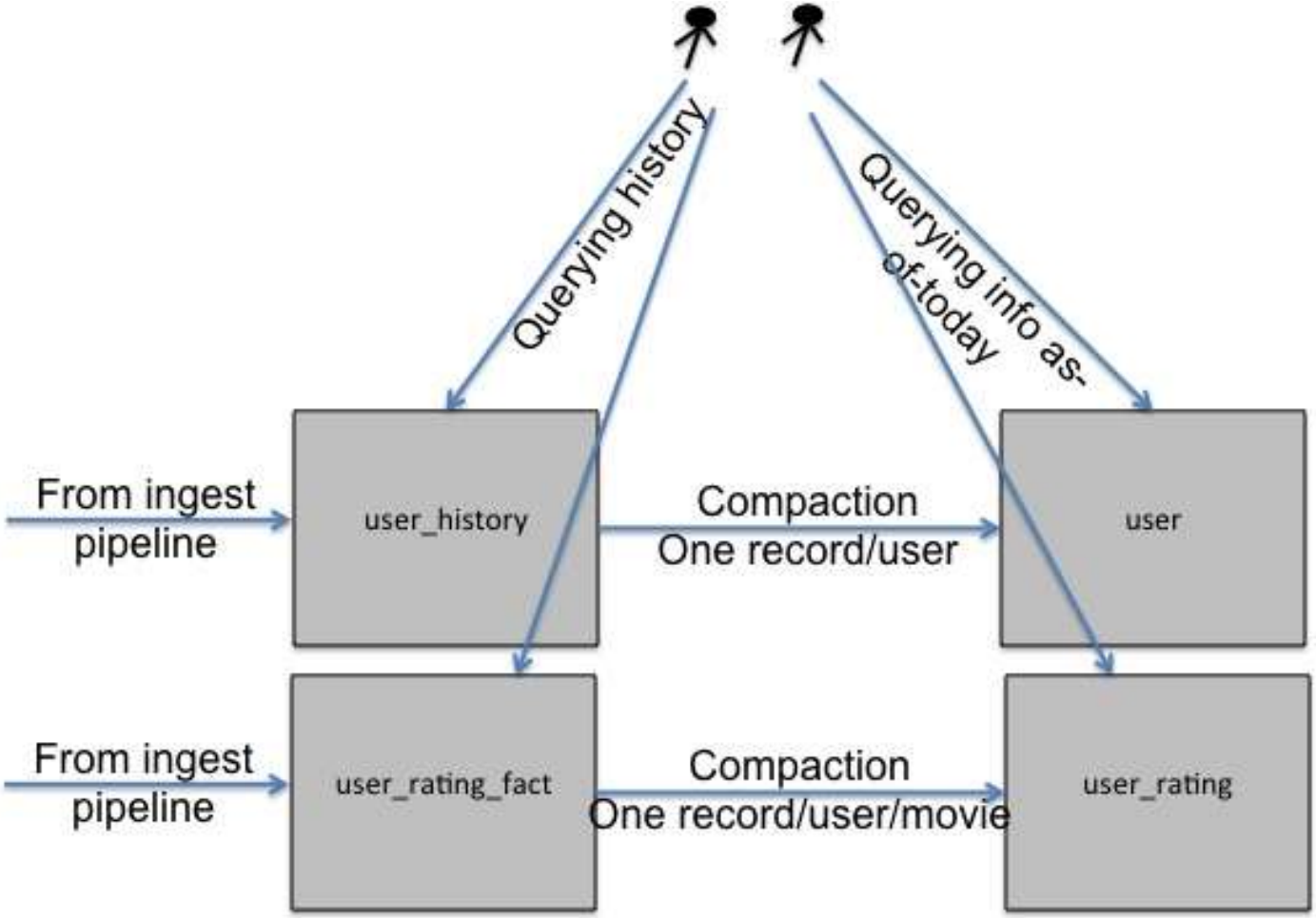
Data Set in Hadoop



Tracking Updates (CDC)

- Can’t update data in-place in HDFS
- HDFS is append-only filesystem
- We have to track all updates

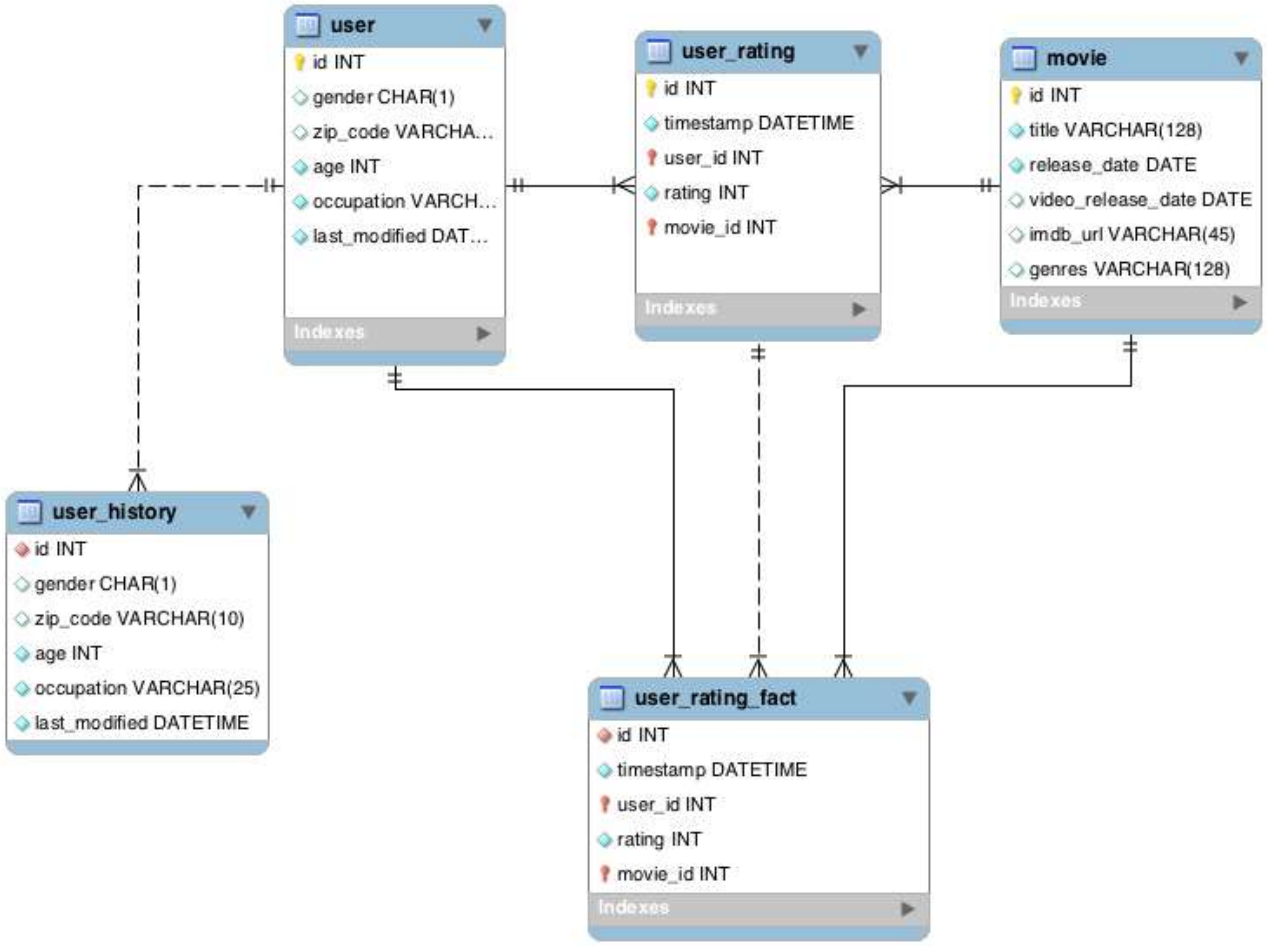
Tracking Updates in Hadoop



Hadoop File Types

- Formats designed specifically to store and process data on Hadoop:
  - File based – SequenceFile
  - Serialization formats – Thrift, Protocol Buffers, Avro
  - Columnar formats – RCFile, ORC, Parquet

Final Schema in Hadoop

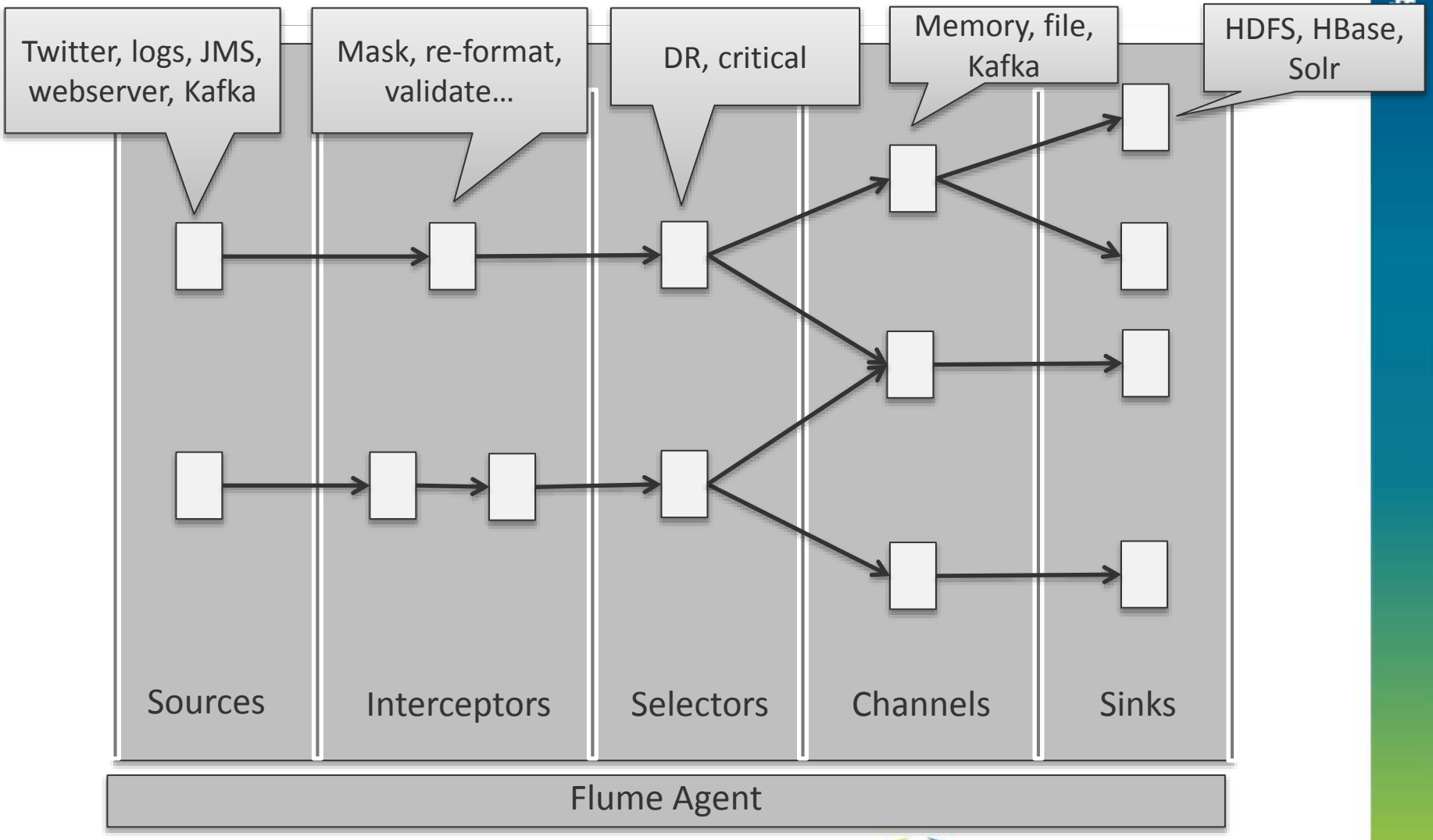


Our Storage Format Recommendation

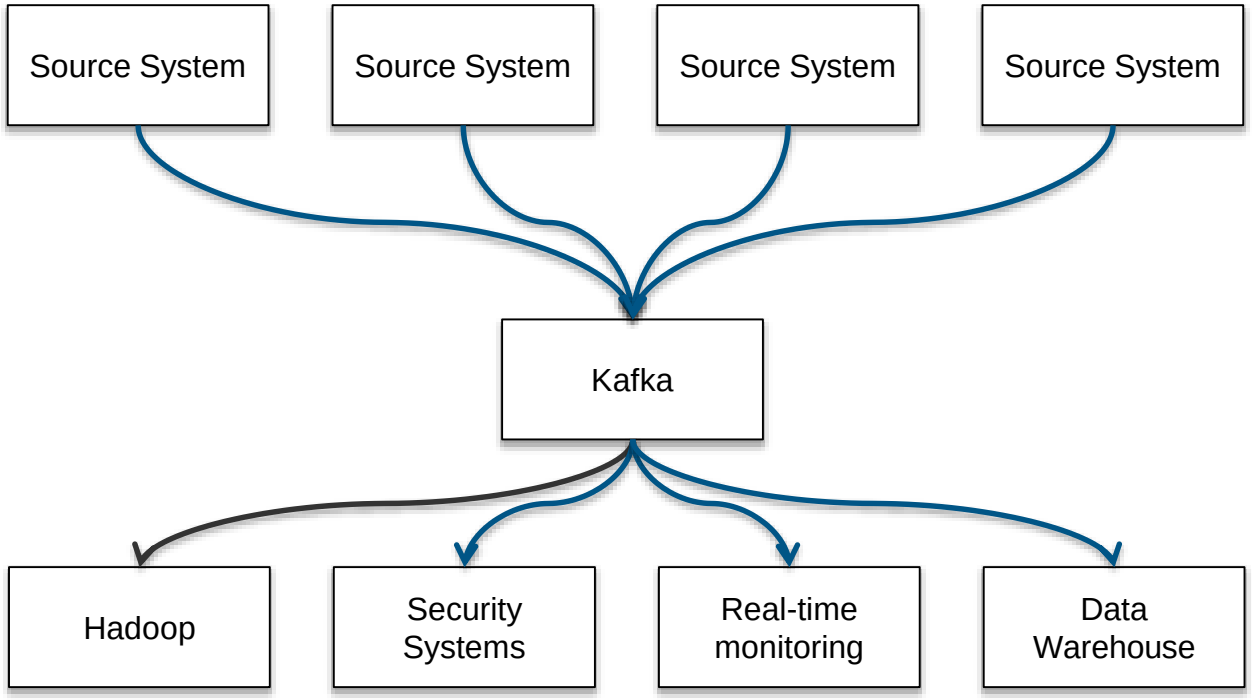
- Columnar format (Parquet) for merged/compacted data sets
  - user, user_rating, movie
- Row format (Avro) for history/append-only data sets
  - user_history, user_rating_fact

Ingestion

Ingestion – Apache Flume

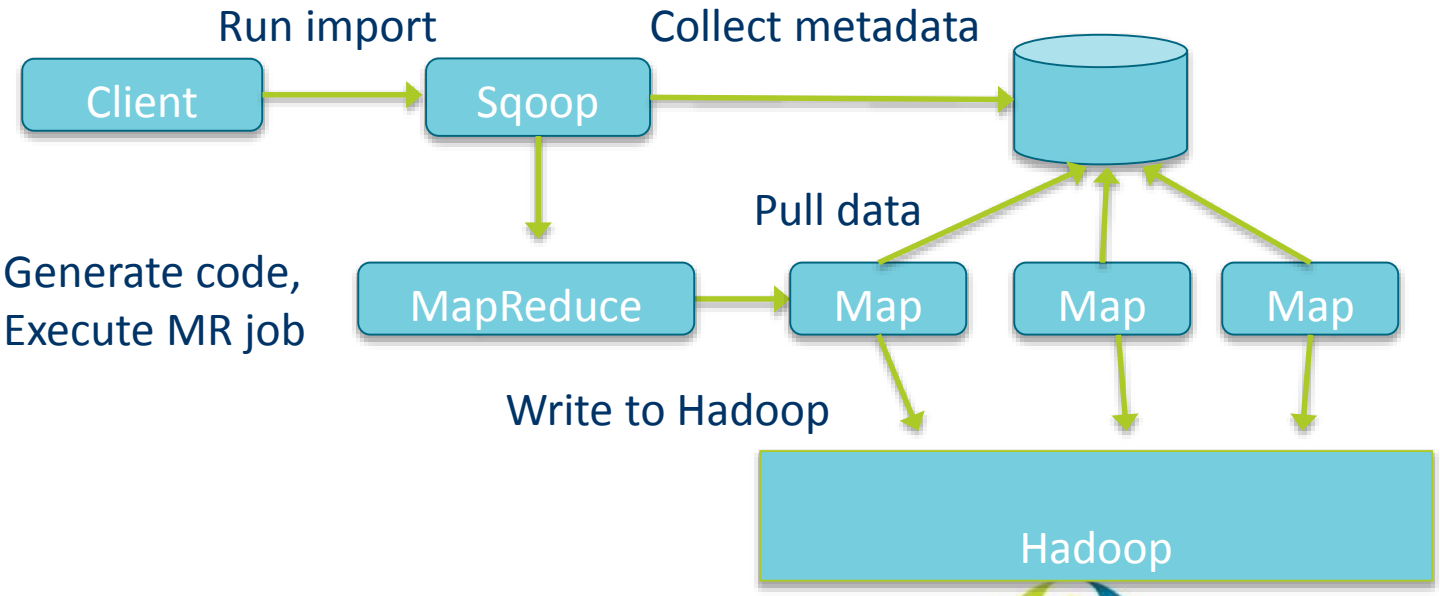


Ingestion – Apache Kafka



Ingestion – Apache Sqoop

- Apache project designed to ease import and export of data between Hadoop and external data stores such as an RDBMS.
- Provides functionality to do bulk imports and exports of data.
- Leverages MapReduce to transfer data in parallel.



Sqoop Import Example – Movie

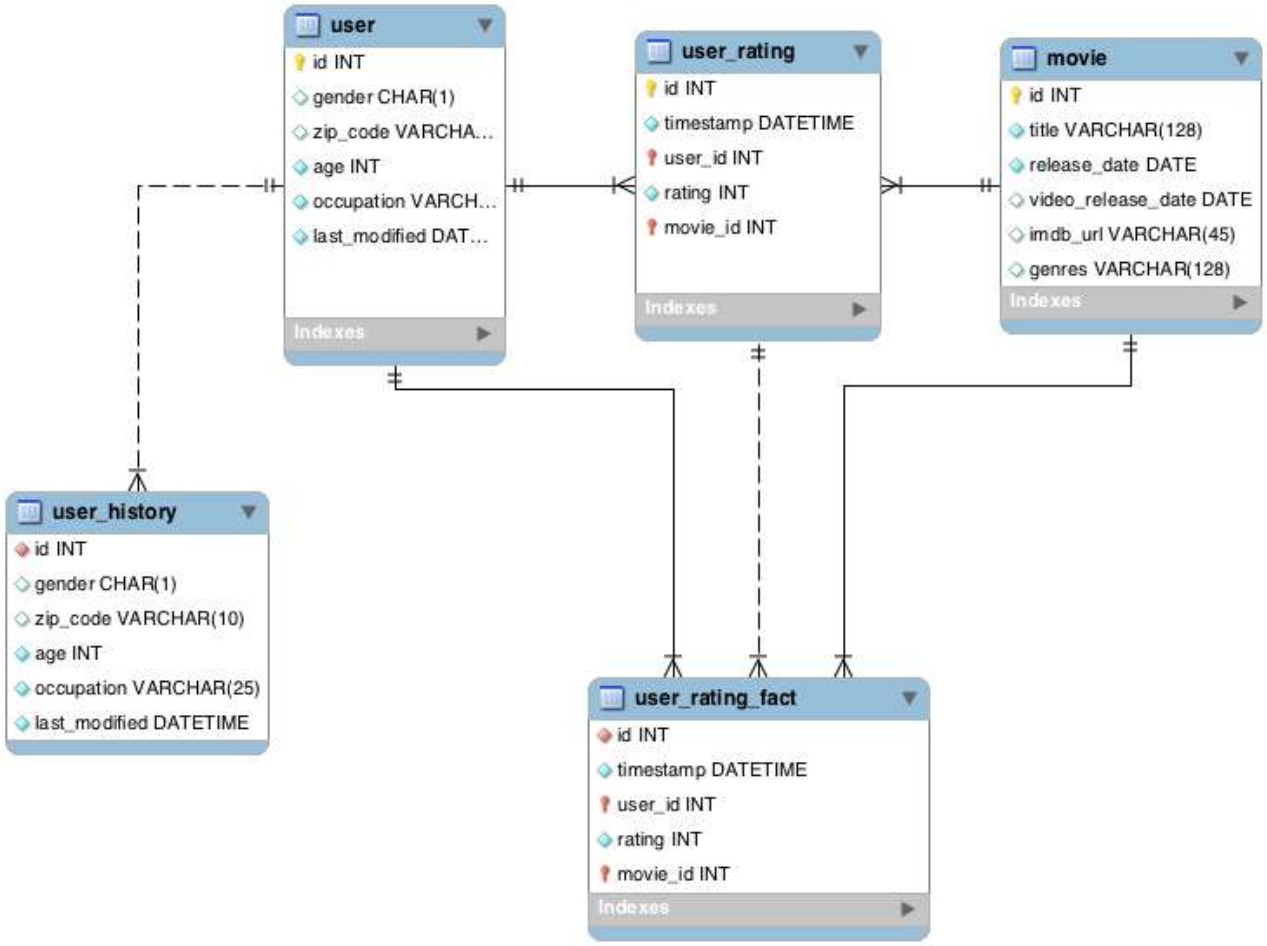
```
sqoop import --connect \
jdbc:mysql://mysql_server:3306/movielens \
--username myuser --password mypass --query \
'SELECT movie.*, group_concat(genre.name)
FROM movie
JOIN movie_genre ON (movie.id =
movie_genre.movie_id)
JOIN genre ON (movie_genre.genre_id = genre.id)
WHERE ${CONDITIONS}
GROUP BY movie.id' \
--split-by movie.id --as-avrodatafile \
--target-dir /data/movielens/movie
```


Data Processing

Popular Processing Engines

- MapReduce
  - Programming paradigm
- Pig
  - Workflow language based
- Hive
  - Batch SQL-engine
- Impala
  - Near real-time concurrent SQL engine
- Spark
  - DAG engine

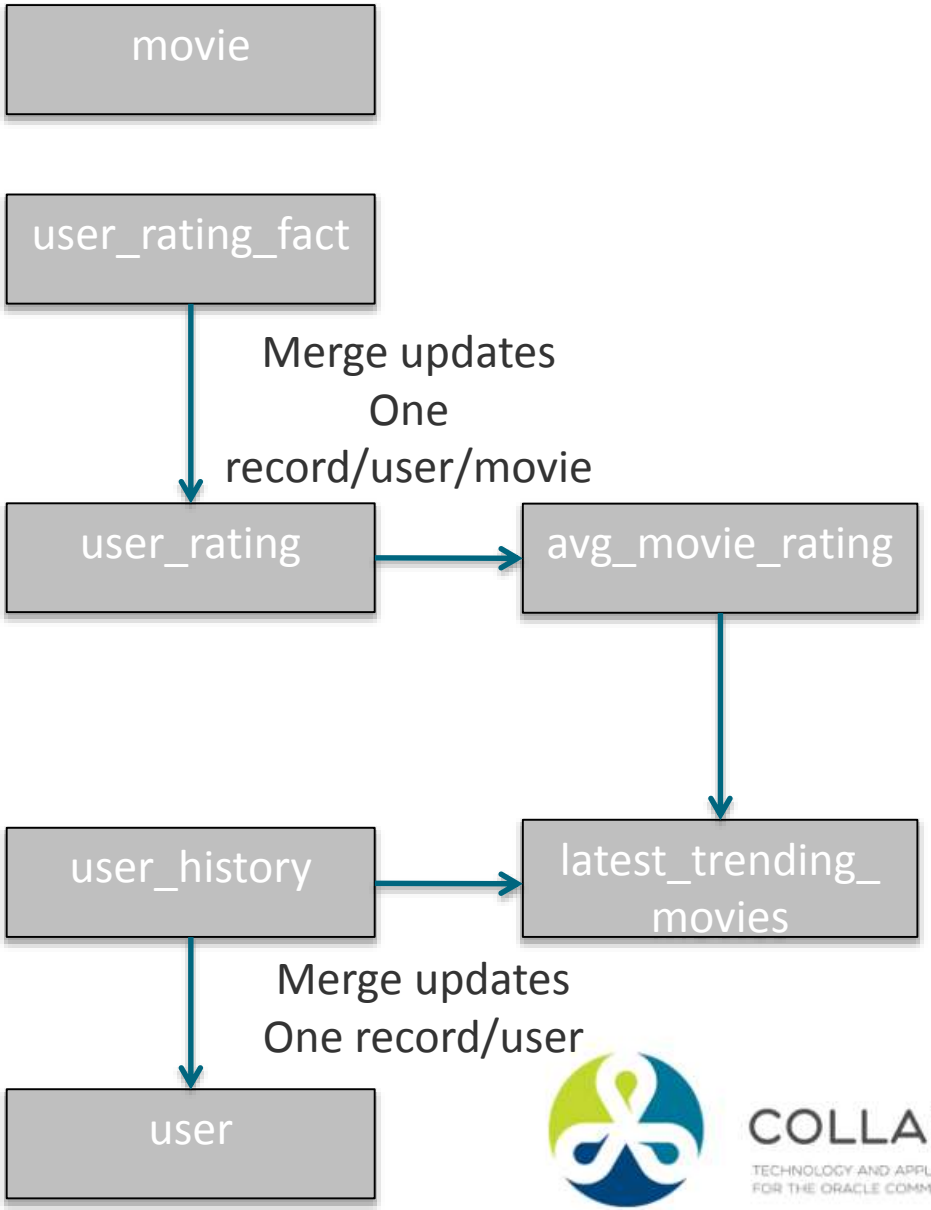
Final Schema in Hadoop



Merge Updates

```
hive>INSERT OVERWRITE TABLE user_tmp
  SELECT user.*
  FROM user
    LEFT OUTER JOIN user_upserts
    ON (user.id = user_upserts.id)
  WHERE
    user_upserts.id IS NULL
  UNION ALL
  SELECT
    id, age, occupation, zipcode,
TIMESTAMP(last_modified)
  FROM user_upserts;
```

Aggregations

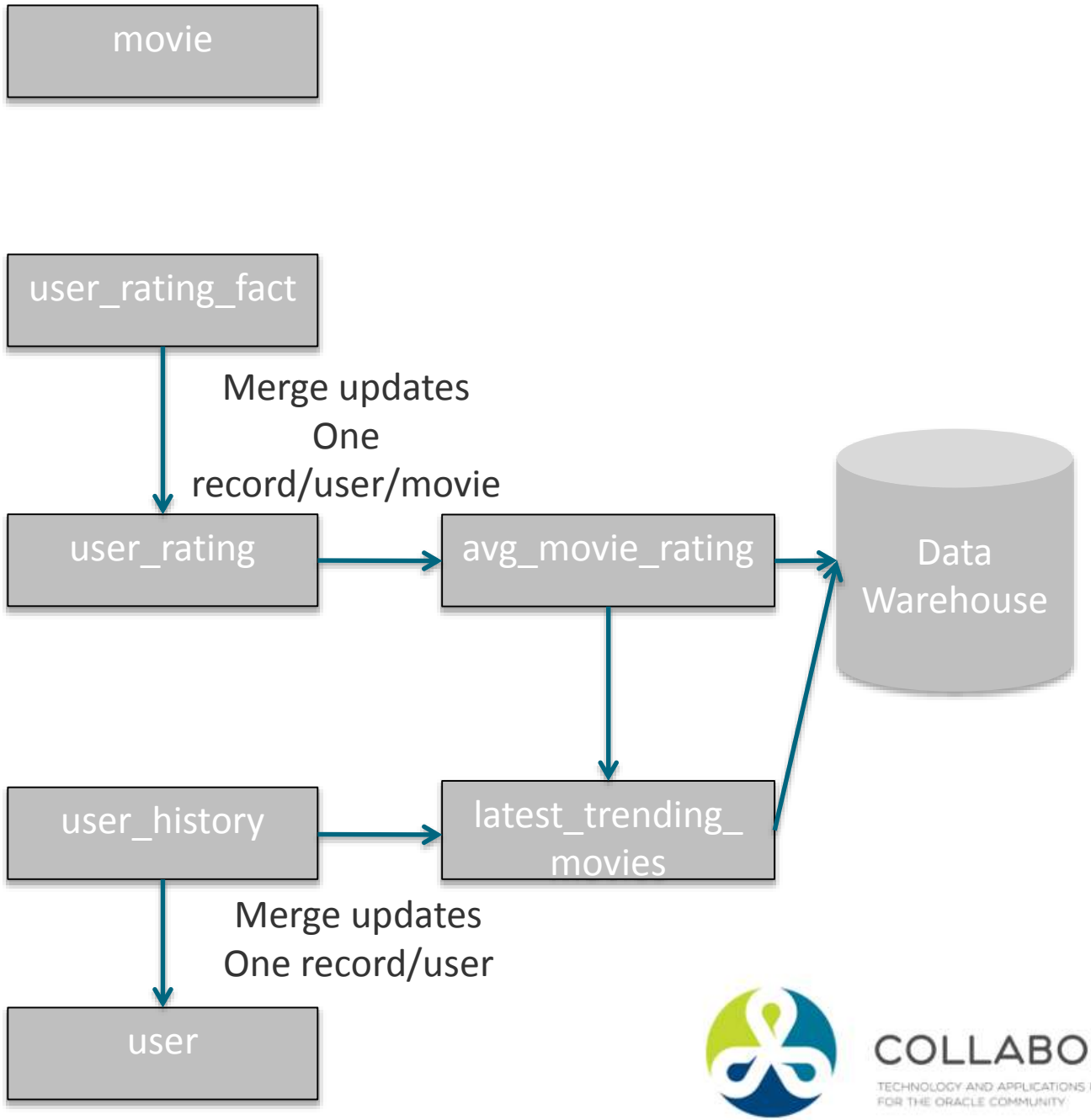


Aggregations

```
hive>CREATE TABLE avg_movie_rating AS
SELECT
  movie_id,
  ROUND(AVG(rating), 1) AS rating
FROM
  user_rating
GROUP BY
  movie_id;
```

Export to Data Warehouse

Export

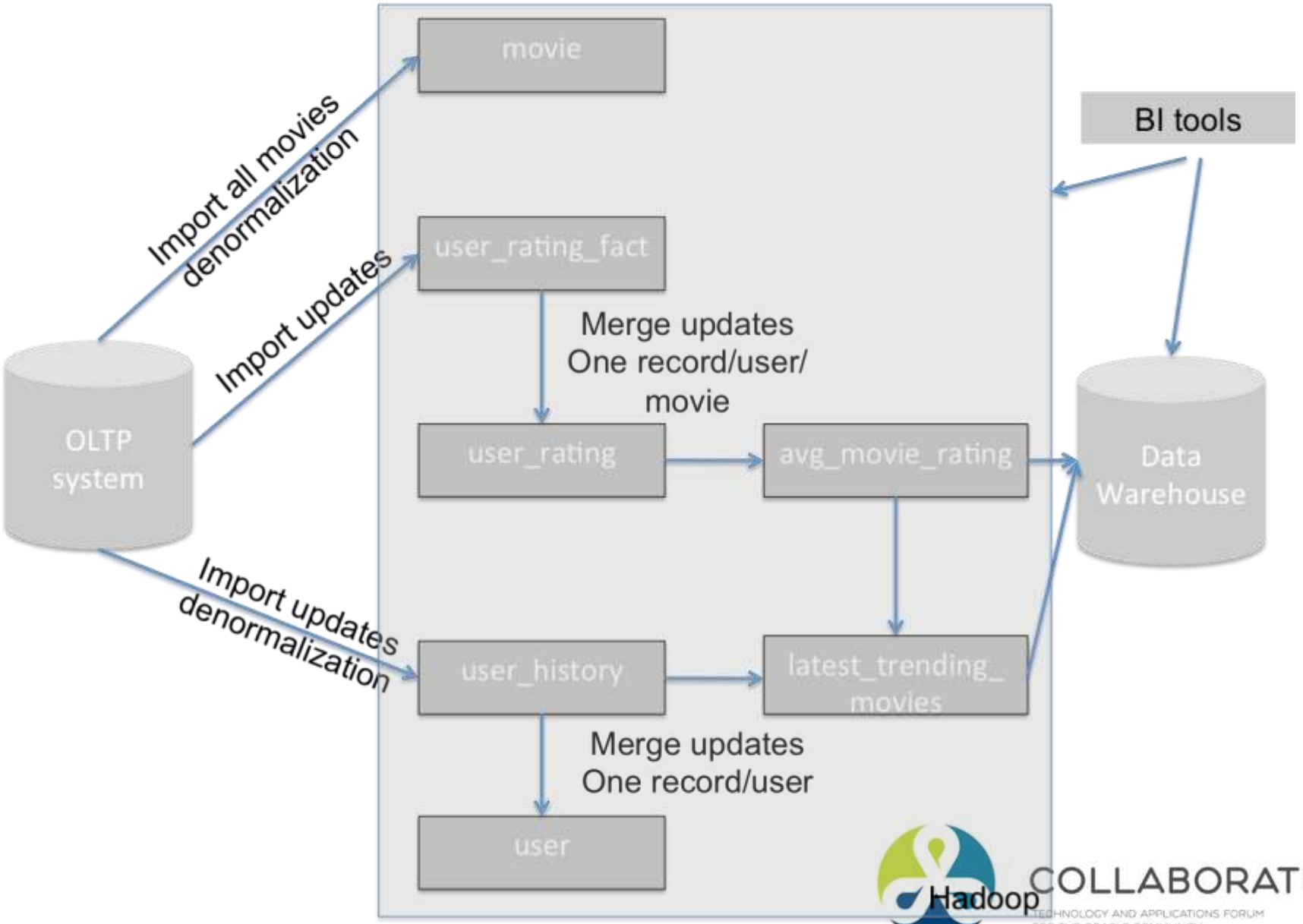

movie
user_rating_fact
Merge updates
One
record/user/movie
user_rating
avg_movie_rating
Data
Warehouse
user_history
latest_trending_
movies
Merge updates
One record/user
user

Sqoop Export

```
sqoop export --connect \
jdbc:mysql:/mysql_server:3306/movie_dwh \
--username myuser --password mypass \
--table avg_movie_rating --export-dir \
/user/hive/warehouse/avg_movie_rating \
-m 16 --update-key movie_id --update-mode \
allowinsert --input-fields-terminated-by \
'\001' --lines-terminated-by '\n'
```

Final Architecture

Final Architecture



Please complete the session evaluation

Thank you!

@hadooparchbook

You may complete the session evaluation either on paper or online via the mobile app

COLLABORATE 15

TECHNOLOGY AND APPLICATIONS FORUM FOR THE ORACLE COMMUNITY

#C15LV

This is a slide title that can be up to two lines of text without losing readability

- This is the first bullet of text
- This is the second bullet of text at the same level
  - This is a sub-bullet of text
    - This is a secondary sub-bullet of text (and should be as far sub-bullets indent)
      - This tertiary sub-bullet will be seldom used, but available
  - This is another sub-bullet of text
- And this is the third bullet of text

This is a slide title (one or two lines)

- This is the first bullet of text
  - This is a sub-bullet of text
- This is the second bullet of text at the same level
  - This is a sub-bullet of text
    - This is a secondary sub-bullet of text
      - This tertiary sub-bullet that can be use
  - This is another sub-bullet of text

- Senior Solutions Architect/Partner Enablement at Cloudera
- Previously, Technical Lead on the big data team at Orbitz Worldwide

This is a slide title (one or two lines)

Subject number one

- This is the first bullet of text
  - This is a sub-bullet of text
- This is the second bullet of text at the same level
  - This is a sub-bullet of text
    - This is a secondary sub-bullet of text
      - This tertiary sub-bullet that can be use
  - This is another sub-bullet of text

Subject number two

- This is the first bullet of text
- This is the second bullet of text at the same level
  - This is a sub-bullet of text
  - This is a sub-bullet of text
    - This secondary sub-bullet
  - This is another sub-bullet of text
- And this is another bullet of text

This is a slide title for a slide with just the title line (e.g., images/diagrams below)

What is Hadoop?

Hadoop is an open-source system designed
To store and process petabyte scale data.

That's pretty much what you need to know.
Well almost…

Compression Codecs

Well, maybe.
Not splittable.

Splittable.
Getting
better…

Splittable,
but no…

Very good
choice

Our Compression Codec Recommendation

- Snappy for all data sets (columnar as well as row based)

File Format Choices

Data set	Storage format	Compression Codec
movie	Parquet	Snappy
user_history	Avro	Snappy
user	Parquet	Snappy
user_rating_fact	Avro	Snappy
user_rating	Parquet	Snappy