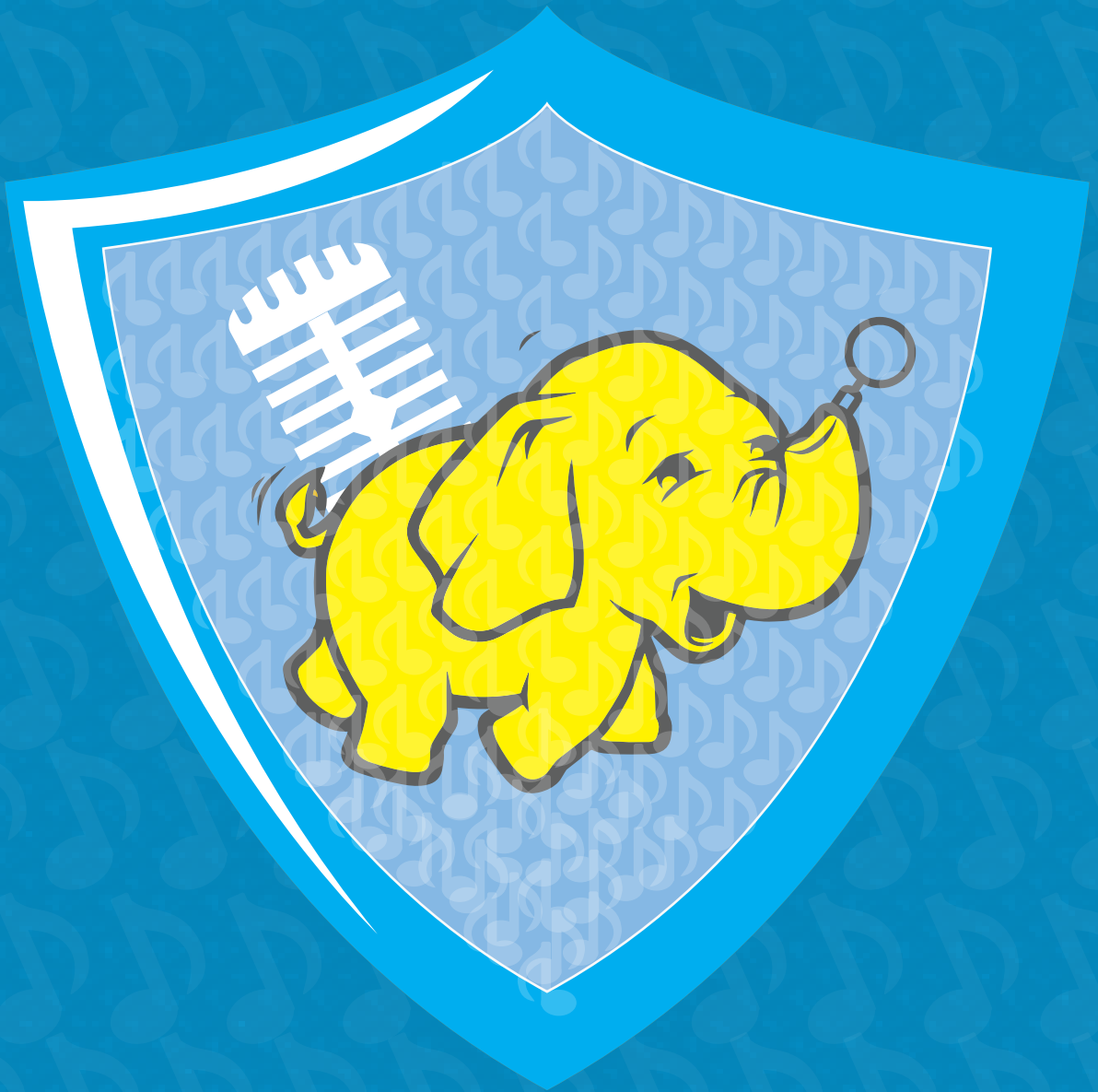


ACADGILD

MUSIC DATA ANALYSIS USING **HADOOP**



About ACADGILD

ACADGILD is a technology education startup that aims to create an ecosystem for skill development in which people can learn from mentors and from each other. We believe that software development requires highly specialized skills that are best learned with guidance from experienced practitioners. Online videos or classroom formats are poor substitutes for building real projects with help from a dedicated mentor. Our mission is to teach hands-on, job-ready software programming skills, globally, in small batches of 8 to 10 students, using industry experts.

ACADGILD offers courses in

Enroll in our programming course
& Boost your career



ANDROID
DEVELOPMENT



DIGITAL
MARKETING



MACHINE LEARNING
WITH R



BIG DATA
ANALYSIS



JAVA FOR
FRESHER



BIG DATA & HADOOP
ADMINISTRATION



FULL STACK WEB
DEVELOPMENT



NODE JS



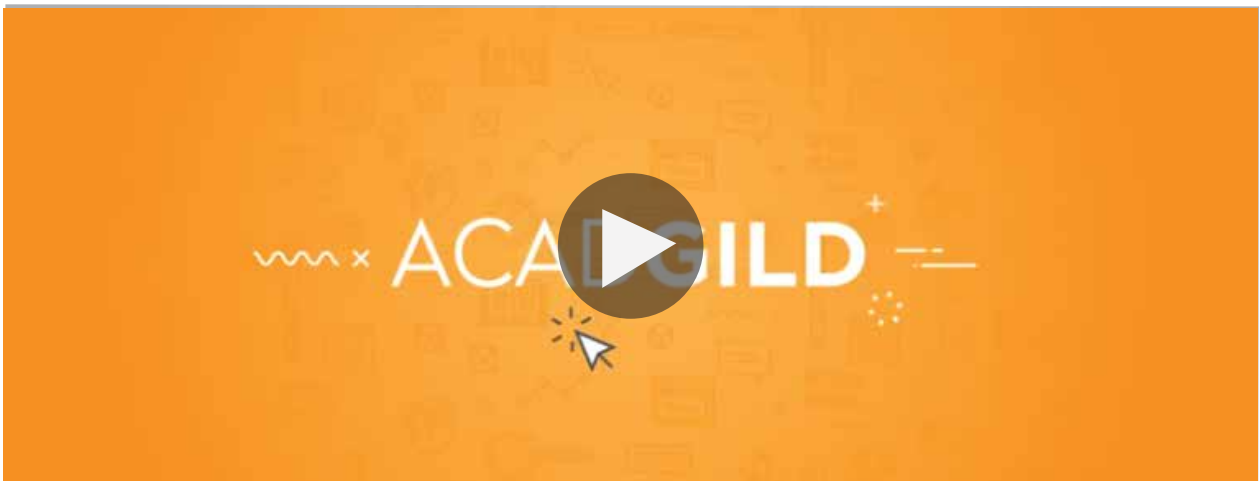
CLOUD
COMPUTING



FRONT END
DEVELOPMENT
(WITH ANGULARJS)



[Watch this short video to know more about ACADGILD.](#)



© 2016 ACADGILD. All rights reserved.

No part of this book may be reproduced, distributed, or transmitted in any form or by any means, electronic or mechanical methods, including photocopying, recording, or by any information storage retrieval system, without permission in writing from ACADGILD.

Disclaimer

This material is intended only for the learners and is not intended for any commercial purpose. If you are not the intended recipient, then you should not distribute or copy this material. Please notify the sender immediately or [click here to contact us](#).

Published by
ACADGILD,
support@acadgild.com



Table of Contents

1.0 Project Description

2.0 Data Files

2.1 Fields present in the data files

2.2 LookUp Tables

3.0 Data Ingestion and Initial Validation

3.1 Rules for data ingestion and data filtering

4.0 Data Enrichment

4.1 Rules for data enrichment

4.2 Post Enrichment

5.0 Data Analysis

5.1 Challenges and Optimisations

6.0 Challenges and Optimisations

7.0 Challenges and Optimisations



1.0

Project Description

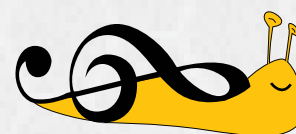
A leading music-catering company is planning to analyse large amount of data received from varieties of sources, namely mobile app and website to track the behaviour of users, classify users, calculate royalties associated with the song and make appropriate business strategies. The file server receives data files periodically after every 3 hours.



2.0

Data Files

Data set consists of user information , song details like song_id , Artist_id, and number of likes and dislikes received for each song.

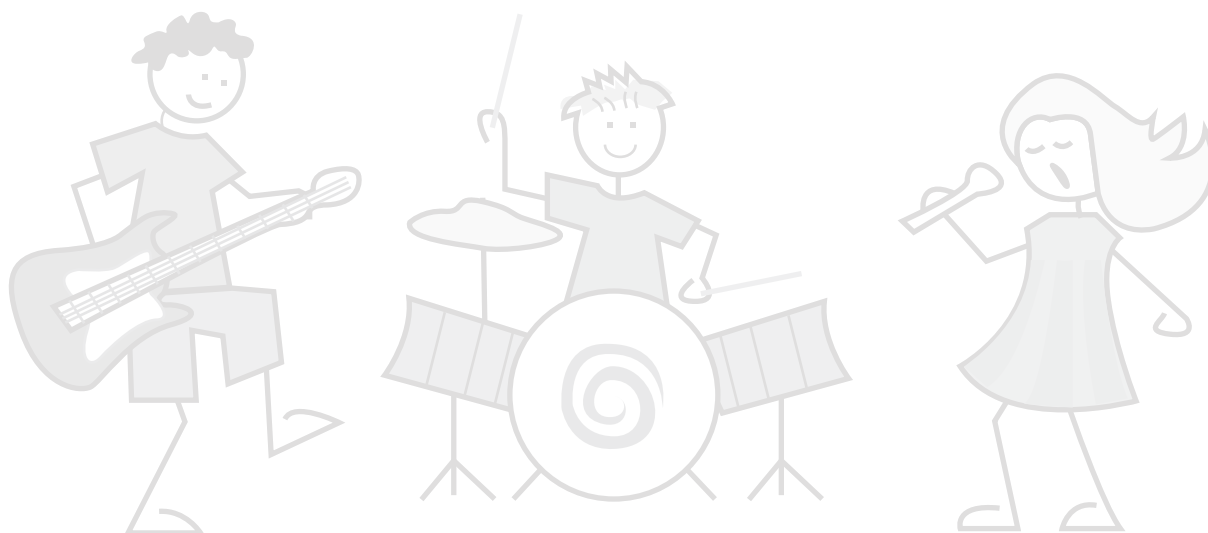


2.1

Project Description

Data files contain below fields.

| Column Name/Field Name | Column Description/Field Description |
|------------------------|--|
| User_id | Unique identifier of every user |
| Song_id | Unique identifier of every song |
| Artist_id | Unique identifier of the lead artist of the song |
| Timestamp | Timestamp when the record was generated |
| Start_ts | Start timestamp when the song started to play |
| End_ts | End timestamp when the song was stopped |
| Geo_cd | Can be 'A' for USA region, 'AP' for asia pacific region, 'J' for Japan region, 'E' for europe and 'AU' for australia region |
| Station_id | Unique identifier of the station from where the song was played |
| Song_end_type | How the song was terminated. 0 means completed successfully 1 means song was skipped 2 means song was paused 3 means other type of failure like device issue, network error etc. |
| Like | 0 means song was not liked 1 means song was liked |
| Dislike | 0 means song was not disliked 1 means song was disliked |



2.2

LookUp Tables

There are some existing look up tables present in NoSQL databases. They play an important role in data enrichment and analysis.

| Table Name | Description |
|------------------|---|
| Station_Geo_Map | Contains mapping of a geo_cd with station_id |
| Subscribed_Users | Contains user_id, subscription_start_date and subscription_end_date. Contains details only for subscribed users |
| Song_Artist_Map | Contains mapping of song_id with artist_id alongwith royalty associated with each play of the song |
| User_Artist_Map | Contains an array of artist_id(s) followed by a user_id |



3.0

Data Ingestion and Initial Validation

3.1 Rules for data ingestion and data filtering

- ✓ Data coming from web applications reside in /data/web and has xml format.
- ✓ Data coming from mobile applications reside in /data/mob and has csv format.
- ✓ Data files come every 3 hours.
- ✓ All the timestamp fields in data coming from web application is of the format YYYY-MM-DD HH:MM:SS.
- ✓ All the timestamp fields in data coming from mobile application is a long integer interpreted as UNIX timestamps.
- ✓ Finally, all timestamps must have the format of a long integer to be interpreted as UNIX timestamps.
- ✓ If both *like* and *dislike* are 1, consider that record to be **invalid**.
- ✓ If any of the fields from User_id, Song_id, Timestamp, Start_ts, End_ts, Geo_cd is **NULL** or absent, consider that record to be invalid.
- ✓ If Song_end_type is **NULL** or absent, treat it to be 3
- ✓ Create a temporary identifier for all the data files received in the last 3 hours (may be an integer batch_id which is auto incremented or a string obtained after combining current date and current hour, to keep track of valid and invalid records per batch).



4.0

Data Enrichment

4.1 Rules for data enrichment

- ✓ If any of like or dislike is **NULL** or absent, consider it as 0.
- ✓ If fields like Geo_cd and Artist_id are **NULL** or absent, consult the lookup tables for fields Station_id and Song_id respectively to get the values of Geo_cd and Artist_id.
- ✓ If corresponding lookup entry is not found, consider that record to be invalid.

| NULL or absent field | Look up field | Look up table (Table from which record can be updated) |
|----------------------|---------------|--|
| Geo_cd | Station_id | Station_Geo_Map |
| Artist_id | Song_id | Song_Artist_Map |



4.2

Post Enrichment

- ✓ Move all valid records in /hadoop/processing_dir in HDFS and invalid records in Local File System at /usr/invalid directory.
- ✓ Maintain a copy of valid records in /usr/validated in Local File System. Run a cleaner everyday to clean validated files which are more than 7 days old.



5.0

Data Analysis

It is not only the data which is important, rather it is the insight it can be used to generate important. Once we have made the data ready for analysis, we have to perform below analysis on a daily basis.

- ✓ Determine top 10 station_id(s) where maximum number of songs were played, which were liked by unique users.
- ✓ Determine total duration of songs played by each type of user, where type of user can be '**subscribed**' or '**unsubscribed**'. An unsubscribed user is the one whose record is either not present in **Subscribed_users** lookup table or has *subscription_end_date* earlier than the *timestamp* of the song played by him.
- ✓ Determine top 10 connected artists. Connected artists are those whose songs are most listened by the unique users who follow them.
- ✓ Determine top 10 songs who have generated the maximum revenue. Royalty applies to a song only if it was *liked* or was *completed successfully* or both.
- ✓ 5. Determine top 10 unsubscribed users who listened to the songs for the longest duration.



5.1

Challenges and Optimisations

- ✓ LookUp tables are in NoSQL databases. Integrate them with the actual data flow.
- ✓ Try to make joins as less expensive as possible.
- ✓ Data Cleaning, Validation, Enrichment, Analysis and Post Analysis have to be automated. Try using schedulers.
- ✓ Appropriate logs have to be maintained to track the behaviour and overcome failures in the pipeline.



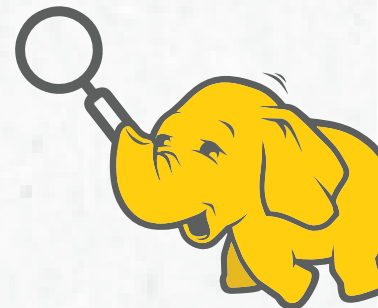
6.0

Post Analysis

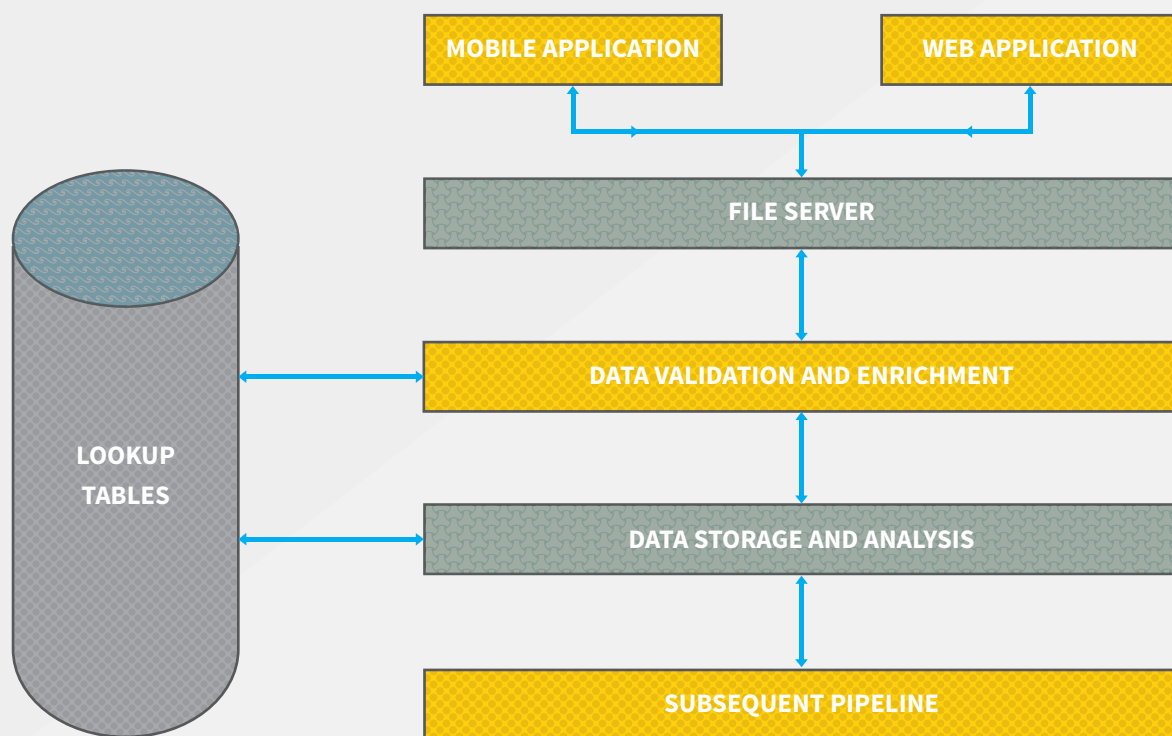
This part is mostly out of the scope of big data developer domain.

Once the analysis is complete, multiple actions can be taken place later on. It includes:

- ✓ 1. Moving result of analysis to the RDMS for data storage and quick retrieval.
- ✓ 2. Form visualisations on the top of analysed data.
- ✓ 3. Send data to data science or machine learning pipelines for further forecast.



This part is mostly out of the scope of big data developer domain.



ACADGILD

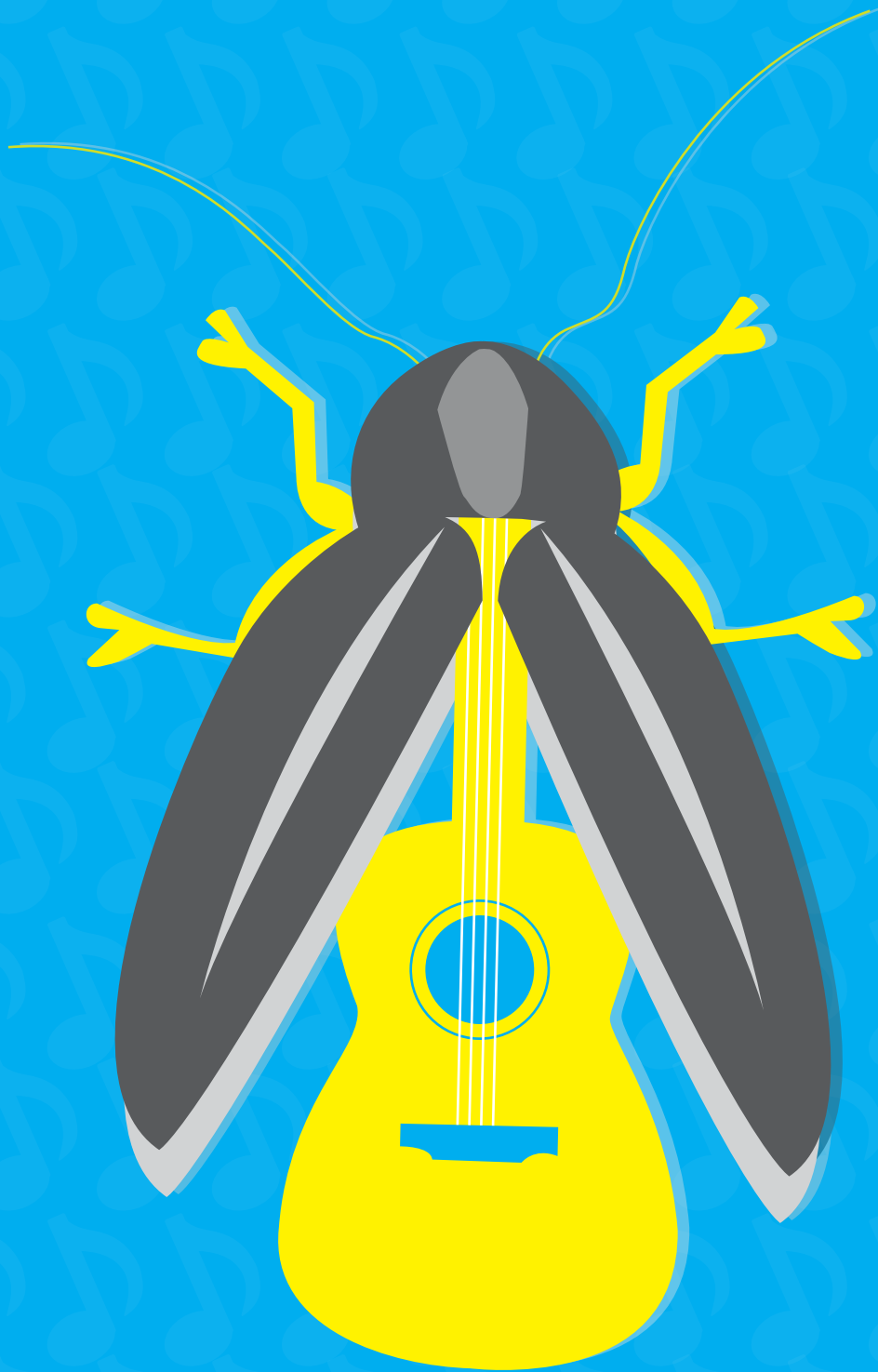
Check out these resources to enrich your skills in [Big Data and Hadoop](#).

We hope this eBook has been helpful in understanding the vital steps necessary to make your career in Big Data domain.

For a better understanding and in-depth learning of Big Data and Hadoop technology, enroll for our [Big Data and Hadoop Development course](#).

Keep visiting our website www.acadgild.com for more posts on Big Data and other technologies.





ACAD**GILD**