Youtube Data Analytics

A cloud compatible data analytics project.

Git Link: https://github.com/subhajitsr/data-assignment-SPH/tree/main

Subhajit Maji [10 Aug, 2024]

Objective

To build an end to end Data pipeline to convert the raw data into meaningful insights. Below are the main parts of the solution.

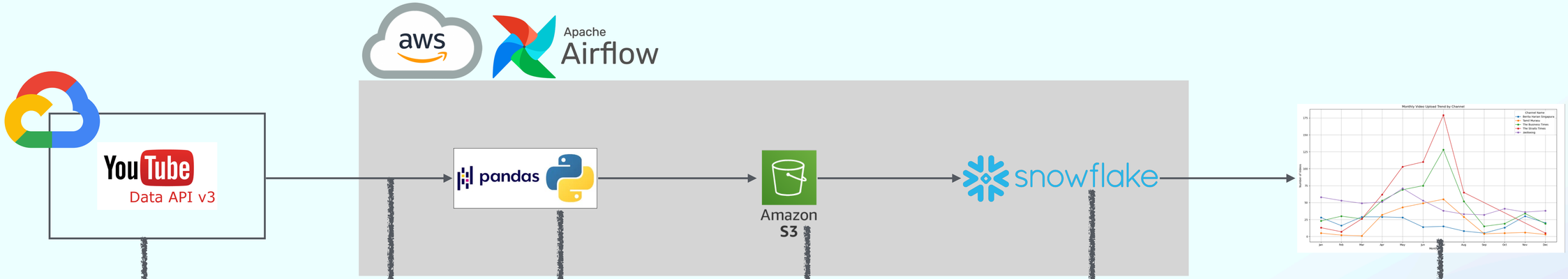
- **Extract** Channel/Video metadata, statistics from Youtube Data API.
- **Clean and prepare** the data into a data frame and upload to S3.
- Load into **Data Warehouse** from S3.
- Create **Data modelling** to define the Facts and Dimensions, build relations between them.
- Build Semantic to perform **logical transformations** on the data and produce calculated KPIs and Dimensions.
- Produce **meaningful insights** from the data.

Technologies at a glance

Below are the technologies used to develop the solution, most of them are either open source or running in free tier.

- **Scripting**: Python 3.9
- **Data Extraction**: Google Cloud python sdk with Youtube Data API V3.
- **Cleaning and Transformation**: Pandas DataFrame.
- **Staging and loading to Warehouse**: S3 Bucket and Snowflake
- **Analytical Query**: SQL
- **Graphical Insights**: Matplotlib
- **Pipeline Orchestration**: Apache Airflow

Architecture & Implementation



1. Created a Project in Google Cloud.
2. Enabled **Youtube Data API V3** for the project.
3. **Created Service account** and given access to the Youtube Data API.
4. **Generated the Secret** Key for the service account to be used in the Pipeline.

1. Created **<u>YoutubeChannel</u>** class, which can fetch a range of data points. This class can be enhanced or inherited in future to support additional data points.

1. Performed **data cleaning** and handled **duplicate data** removal based on the expected Key columns

1. Created a new **S3 bucket**.
2. Created a **IAM user** for the project.
3. Created a new **IAM policy** which gives Read/ Write access to the new bucket.
4. Created a new **IAM Group** and added the above policy to it.
5. Finally, Added the new user to the above group.
6. Created four different folders in the bucket to drop different files.

1. Created **Data model** to decide the tables and the relations. i.e. Dimensions and Facts.
2. Created S3 Stage objects and Core tables for loading data to Snowflake.
3. Created a **<u>SnowflakeLoader</u>** python class which can handle different types of data loading from S3 to Snowflake table. i.e. Full refresh, Incremental etc. This is can be easily **reused by the other applications** wherever S3-SF load is needed.
4. Created **Daily snap** tables for Videos and Channels to capture the KPIs.

1. Used Jupyter Notebook and **Matplotlib** to produce some basic visualisation on the data.

Data Definition

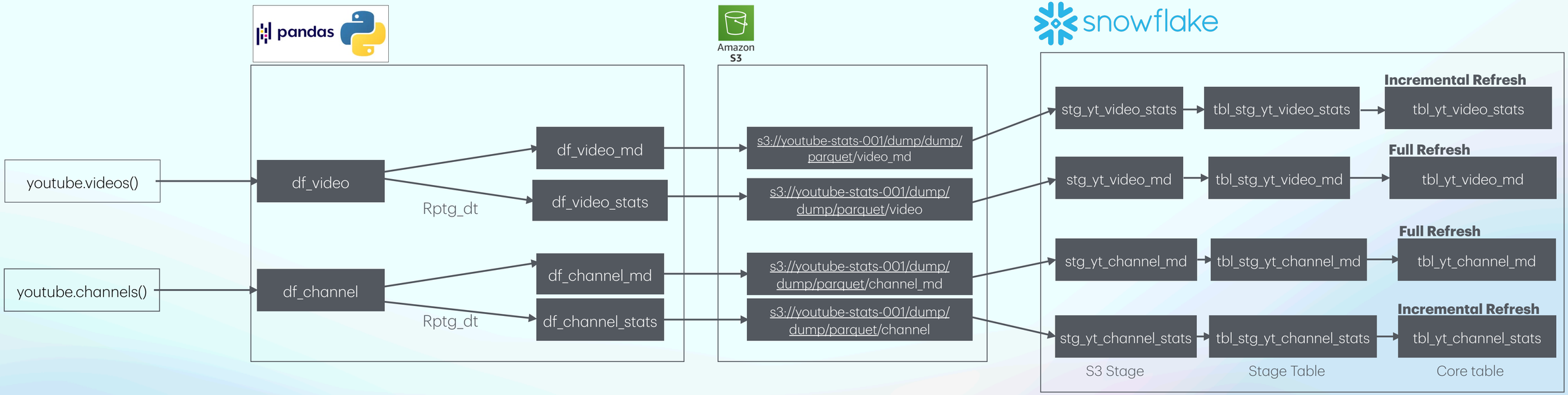
List of data points collected from the Youtube Data API V3 and definition for each of them.

- Collected from **youtube.channels()**
  - **channel_id** - Unique channel id
  - **channel_name** - Name/Username of the channel
  - **title** - Channel Title
  - **description** - Description of the channel
  - **customUrl** - This is a unique URL for a Channel
  - **publishedAt** - Time of the channel publish
  - **country** - Country from which the channel is originated
  - **viewCount** - Total number of views till date on all the videos published
  - **subscriberCount** - Total number of subscribers till date
  - **videoCount** - Total number of videos published till date

- Collected from **youtube.videos()**
  - **id** - Unique video id
  - **title** - Title of a video
  - **url** - Unique URL for the video
  - **views** - Count of views from the time of Publish
  - **likes** - Total likes till date
  - **dislikes** - Total dislikes till date
  - **comments** - Total comments till date
  - **publishedAt -** Time of the video publish.

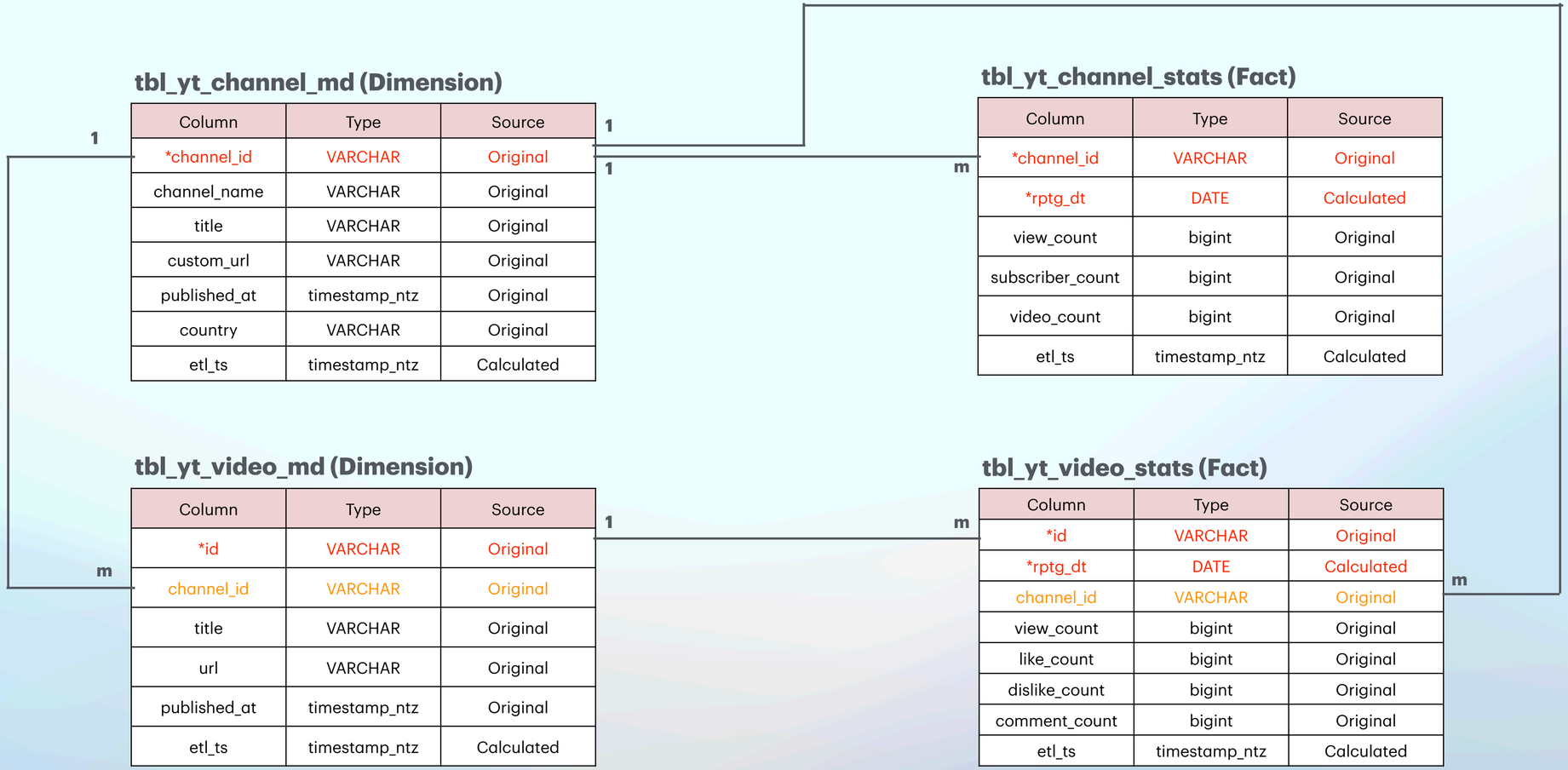
Data Flow

End-to-end data flow diagram



Data Model

All the Dimensions and Measures, and the relations between them.

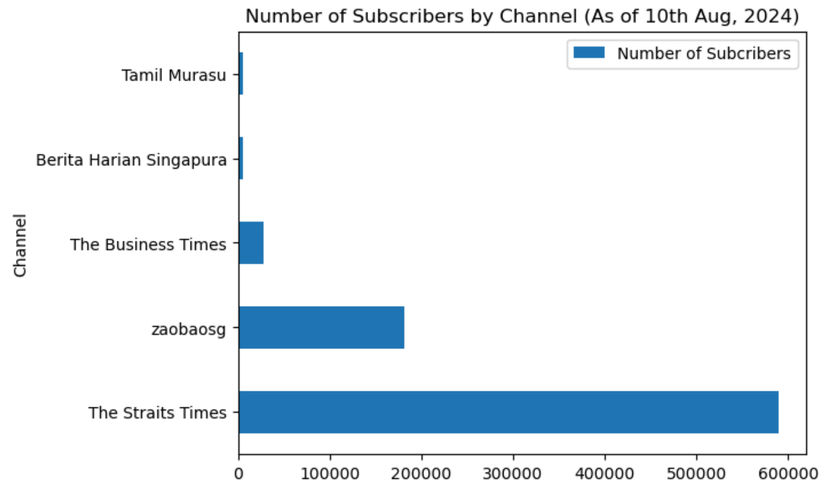


Insights (1/4)

How many subscribers are there for each channel?

```
data = pd.read_sql("""select
md.channel_name,
md.title as "Channel",
stat.subscriber_count as "Number of Subcribers"
from CORE.tbl_yt_channel_stats stat
inner join CORE.tbl_yt_channel_md md on md.channel_id = stat.channel_id
where stat.rptg_dt='2024-08-10'
order by 3 desc;""",dbcon)
display(data)
data.plot(x='Channel',y=['Number of Subcribers'],kind='barh',title='Number of Subscribers by Channel (As of 10th Aug
plt.show()
```

	CHANNEL_NAME	Channel	Number of Subcribers
0	straitstimesonline	The Straits Times	590000
1	zaobaodotsg	zaobaosg	182000
2	TheBusinessTimes	The Business Times	27200
3	BeritaHarianSG1957	Berita Harian Singapura	5010
4	Tamil_Murasu	Tamil Murasu	4860



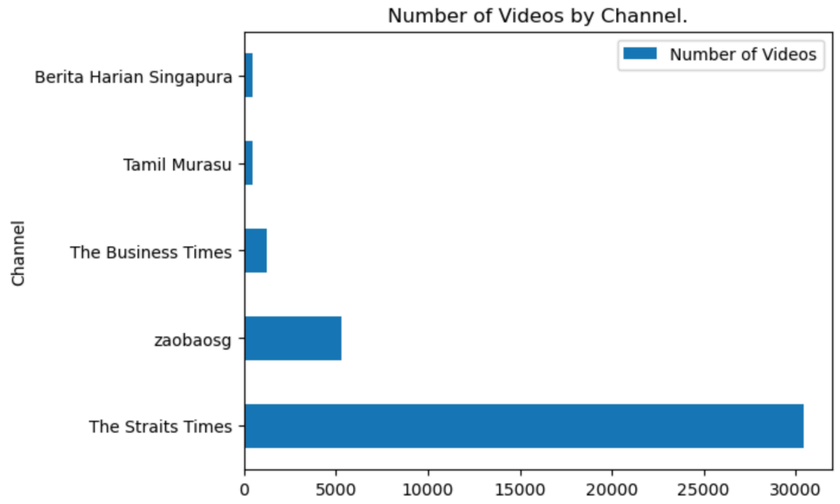
- "The Straits Times" having the most number number of Subscribers and "Tamil Murasu" having the least among the listed channels as of 10th Aug, 2024.

Insights (2/4)

How many video's have been published for each channel?

```
: data = pd.read_sql("""select
md.channel_name,
md.title as "Channel",
stat.video_count as "Number of Videos"
from CORE.tbl_yt_channel_stats stat
inner join CORE.tbl_yt_channel_md md on md.channel_id = stat.channel_id
where stat.rptg_dt='2024-08-10'
order by 3 desc;""",dbcon)
display(data)
data.plot(x='Channel',y=['Number of Videos'],kind='barh',title='Number of Videos by Channel.')
plt.show()
```

	CHANNEL_NAME	Channel	Number of Videos
0	straitstimesonline	The Straits Times	30446
1	zaobaodotsg	zaobaosg	5328
2	TheBusinessTimes	The Business Times	1270
3	Tamil_Murasu	Tamil Murasu	461
4	BeritaHarianSG1957	Berita Harian Singapura	452

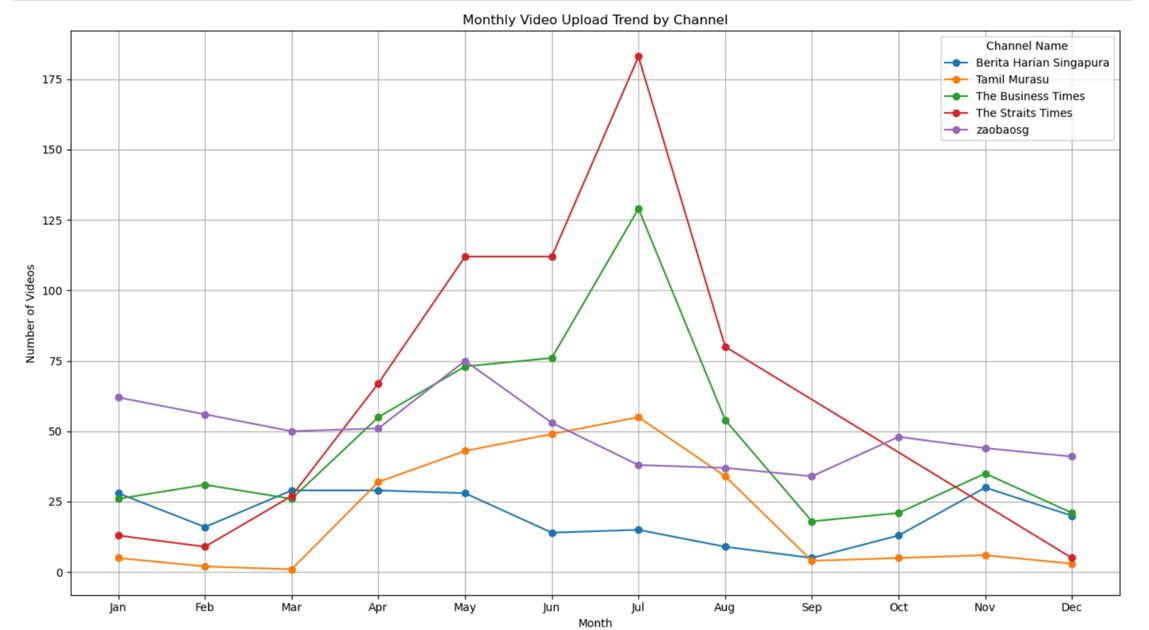


- "The Straits Times" having the most number number of videos and "Berita Harian Singapura" having the least among the listed channels as of 10th Aug, 2024.

Insights (3/4)

What is the trend for videos published by each channel over the last 12 months?

```
df = pd.read_sql("""SELECT
    TO_CHAR(a.published_at, 'Mon') AS "Month",
    TO_CHAR(a.published_at, 'MM') AS "Month_num",
    chnl.title as "Channel Name",
    count(a.id) as "Number of videos"
FROM
    CORE.tbl_yt_video_md a
    inner join CORE.tbl_yt_channel_md chnl on chnl.channel_id = a.channel_id
GROUP BY 1,2,3
order by 2,3;
""",dbcon)
```



- "The Straits Time", "The Business Times" and "Tamil Murasu" published most of their videos in the Month of July and least towards the later half of the year.
- "Zaobaosg" Published most on May and least in July.
- "Berita Harian Singapura" published most during March, April, May and November, least in September.

Insights (4/4)

Which are the most viewed videos?

```
data = pd.read_sql("""select
chnl.title as "Channel Name",
vdo.title as "Video Title",
vdo.url as "URL",
vdo.published_at as "Published Date",
a.view_count as "Views"
from CORE.tbl_yt_video_stats a
inner join CORE.tbl_yt_video_md vdo on vdo.id = a.id
inner join CORE.tbl_yt_channel_md chnl on chnl.channel_id = a.channel_id
where a.rptg_dt = '2024-08-10'
qualify row_number() over(partition by a.channel_id order by a.view_count desc) = 1
order by 5 desc
""",dbcon)
display(data)
```

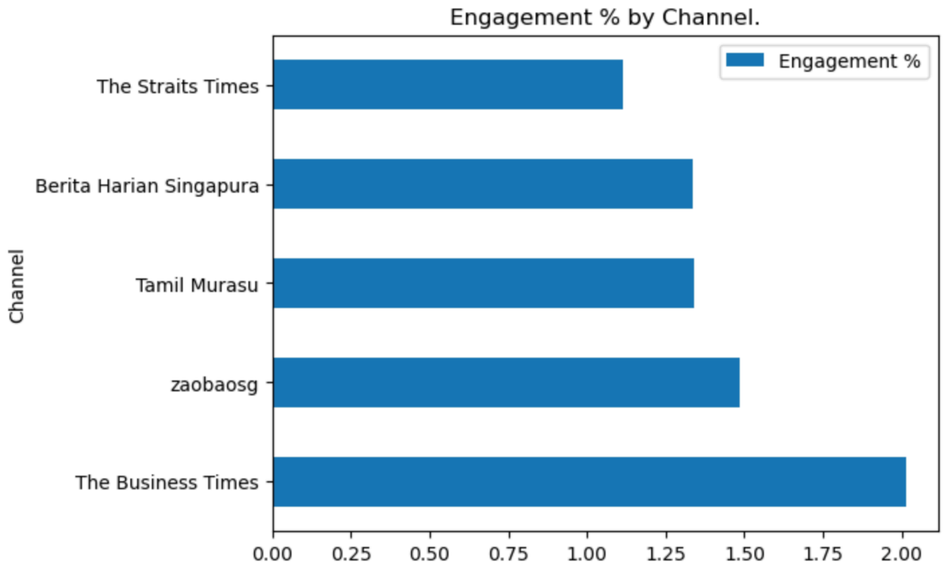
	Channel Name	Video Title	URL	Published Date	Views
0	The Business Times	Chaos at Donald Trump rally, assassination att...	https://www.youtube.com/watch?v=q4Lftwq98DY	2024-07-14 04:40:47	1470189
1	zaobaosg	[ENG SUB] 朝鲜“新星女将军” 金正恩爱女的使命 Kim Jong Un's Da...	https://www.youtube.com/watch?v=g_rn5Behgek	2024-01-19 07:00:15	1176936
2	The Straits Times	WATCH: The moment Trump was shot in right ear ...	https://www.youtube.com/watch?v=NsfMPTiluvY	2024-07-13 23:51:25	632742
3	Berita Harian Singapura	Laporan Khas Berita Harian Singapura: Mengejar...	https://www.youtube.com/watch?v=9QOX5L27mBg	2024-07-27 21:30:22	51752
4	Tamil Murasu	லிட்டில் இந்தியா கலவரம் கற்றுத் தந்த பாடம். 10...	https://www.youtube.com/watch?v=YDrL8POdS0w	2023-12-08 07:31:31	2798

- List of most viewed videos for each channel as of 10th August, 2024

Additional Insights (1/3)

Engagement Rate of each channel since last one year.

```
data = pd.read_sql("""select
chnl.title as "Channel",
sum(a.like_count) as total_likes,
sum(a.dislike_count) as total_dislikes,
sum(a.comment_count) as total_comment,
sum(a.view_count) as total_views,
((total_likes - total_dislikes + total_comment)/total_views)*100 as "Engagement %"
from
CORE.tbl_yt_video_stats a
inner join CORE.tbl_yt_channel_md chnl on chnl.channel_id = a.channel_id
where a.rptg_dt = '2024-08-10'
group by 1
order by 6 desc;""",dbcon)
#display(data)
data.plot(x='Channel',y=['Engagement %'],kind='barh',title='Engagement % by Channel.')
plt.show()
```



- **Engagement Rate** (Channel KPI): ((Total likes - Total dislikes + Comments)/Total views)*100
- Although “The Straits Times” has most number of subscribers and videos, but their Engagement rate is the lowest among others.
- “The Business Times” has the highest engagement rate.

Additional Insights (2/3)

Top 3 most engaging videos of each channel.

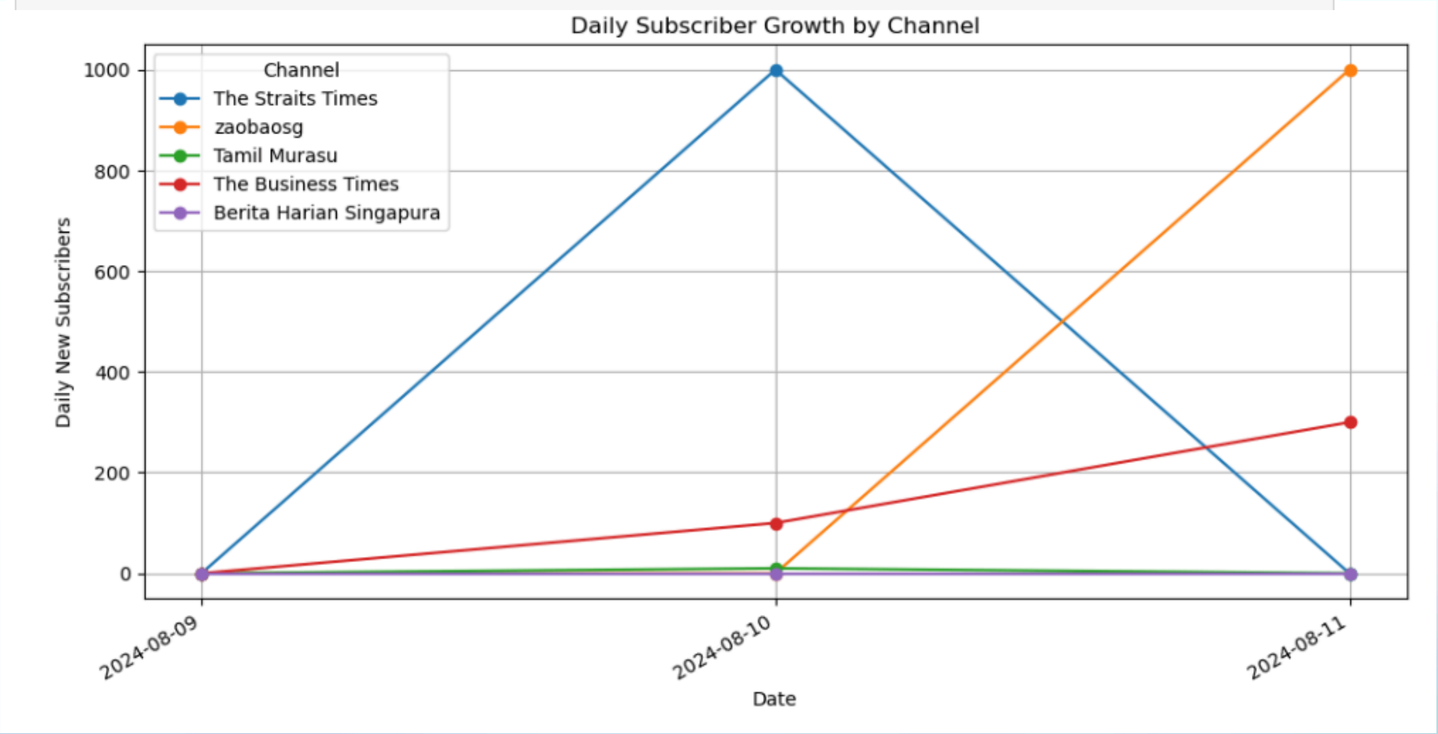
```
: data = pd.read_sql("""select
  chnl.title as "Channel",
  md.title as "Video Title",
  md.url as "Video URL",
  src.engagement_perc as "Engagement %",
  src.engagement_rank as "Rank"
  from
  (select
  id,
  channel_id,
  case when coalesce(view_count,0)>0 then ((like_count - dislike_count + comment_count)/view_count)*100 else 0 end as
  row_number() over(partition by channel_id order by engagement_perc desc) as engagement_rank
  from
  CORE.tbl_yt_video_stats
  where rptg_dt='2024-08-09'
  qualify row_number() over(partition by channel_id order by engagement_perc desc) <= 3) src
  inner join CORE.tbl_yt_channel_md chnl on chnl.channel_id = src.channel_id
  inner join CORE.tbl_yt_video_md md on md.id = src.id
  order by 1,5""",dbcon)
  display(data)
```

	Channel	Video Title	Video URL	Engagement %	Rank
0	Berita Harian Singapura	Kaki Makan: Pemilik Waroeng Anak Indo dan Rumi...	https://www.youtube.com/watch?v=jvPDmuumxGM	10.2041	1
1	Berita Harian Singapura	Kaki Makan: Menu khas Ramadan terap budaya Tur...	https://www.youtube.com/watch?v=EktMlxqvuPA	9.0909	2
2	Berita Harian Singapura	Usaha gigih buah hasil walau galas pelbagai ta...	https://www.youtube.com/watch?v=zsvK6ay04Lo	7.6923	3
3	Tamil Murasu	முரசு காப்பிக் கடை: விலையேற்றத்திலும் குதூகலம்...	https://www.youtube.com/watch?v=LLTLJe5TH48	33.3333	1
4	Tamil Murasu	லாரன்ஸ் வோங் பிரதமராகவும் தர்மன் சண்முகரத்னம் ...	https://www.youtube.com/watch?v=s_Y3C-ILcNc	33.3333	2
5	Tamil Murasu	100,000 எழுத்துருக்களால் ஆன தமிழரசனின் லீ குவா...	https://www.youtube.com/watch?v=heB5HI1gO70	11.3636	3
6	The Business Times	Lens on Daily: Friday, July 26, 2024 (Ep 75)	https://www.youtube.com/watch?v=qeZyrQW9QXY	50.0000	1
7	The Business Times	Lens on Daily: Friday, Jun 7, 2024 (Ep 41)	https://www.youtube.com/watch?v=sfXnhbcAL1Q	12.5000	2
8	The Business Times	BT Future of Finance: How to take advantage of...	https://www.youtube.com/watch?v=BgRp6UBgLHs	11.1111	3
9	The Straits Times	K-pop stars Seventeen become Unesco ambassadors	https://www.youtube.com/watch?v=SiSbTCUp6Xl	10.3064	1
10	The Straits Times	Trump kisses helmet of slain firefighter	https://www.youtube.com/watch?v=CluUBJSKFYo	6.5179	2
11	The Straits Times	Riding the K-pop wave in S'pore: Meet dance co...	https://www.youtube.com/watch?v=EiBO8aUtiLI	5.8628	3
12	zaobaosg	新加坡国庆庆典360度烟火表演绚丽登场！#zaobaosg #sgnews #shorts	https://www.youtube.com/watch?v=LIsLZIzYsNw	10.2894	1
13	zaobaosg	当新加坡成了“坡县”，你能接受吗？	https://www.youtube.com/watch?v=DNeVrTbPalk	10.1573	2
14	zaobaosg	国务资政李显龙入场 现场欢呼声不断！#zaobaosg #sgnews #shorts	https://www.youtube.com/watch?v=qQ1_1WWqe-4	8.2265	3

Additional Insights (3/3)

Day over Day Growth Rate of Subscribers for each channel.

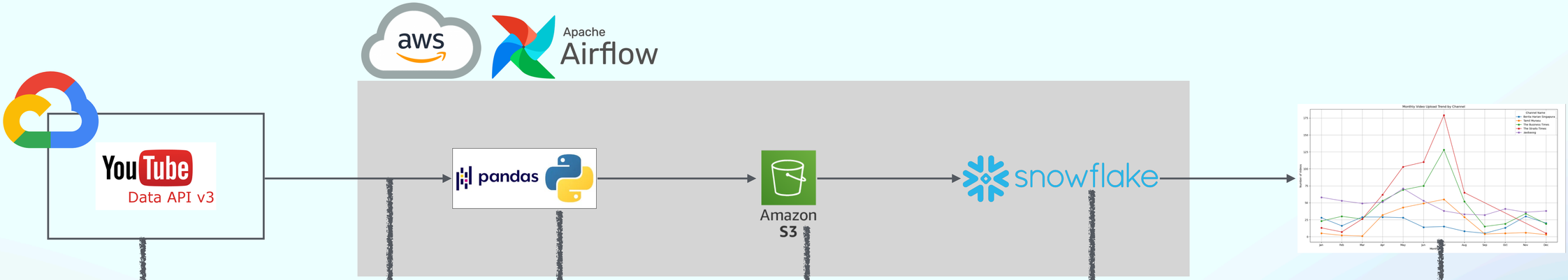
```
df = pd.read_sql("""select
orig.rptg_dt as "Date",
orig.channel_id,
md.title as "Channel",
coalesce((sum(orig.subscriber_count) - sum(prev.subscriber_count)),0) as "Daily new subscribers"
from
core.tbl_yt_channel_stats orig
left join core.tbl_yt_channel_stats prev on orig.channel_id = prev.channel_id and orig.rptg_dt = DATEADD(day, 1, pre
inner join core.tbl_yt_channel_md md on md.channel_id = orig.channel_id
group by 1,2,3
""",dbcon)
#display(data)
```



- This is a calculated KPI based on comparing the Subscribers count of each channel for a date with it's previous day.
- When enough snaps are collected, we can even derive WoW, MoM and YoY growth as well.

Limitations and Improvements scope

There are few limitations in this solution, which are typically due to the shortage of resources and limited timeline. There is always a chance to improve further, let's look at them.



- Youtube Data API can be explored more and **collect many more data points** which can produce important KPIs and Dimensions. I.e. Video Watch duration, Ads related data etc.

1. The YoutubeChannel and SnowflakeLoader class can be **published to PyPI** as a package and can be used across multiple teams in the same department

1. We can use **Spark DataFrame in AWS Glue** for orchestration if the data volume is huge in future.

1. **AWS Aethna** can be introduced to directly query the raw parquet files.
2. Implement **Lifecycle Policies** for Automated Data Management.
3. Use **Intelligent-Tiering** Storage class

1. Implement **Data Governance**.
2. App Specific **semantic** tables loaded by complex PLSQL logic to produce customised Measures and Dimensions.
3. **Define Partitions** in the large tables to minimise data extractions query timings.
4. Implement **purging mechanism** on the snap tables.

1. Data visualisation tools like **Tableau, Power BI** can be introduced for more attractive and complex reports.
2. **Custom UI dashboards** can be built as well which will require API and UI development.

Q & A

Thank you.