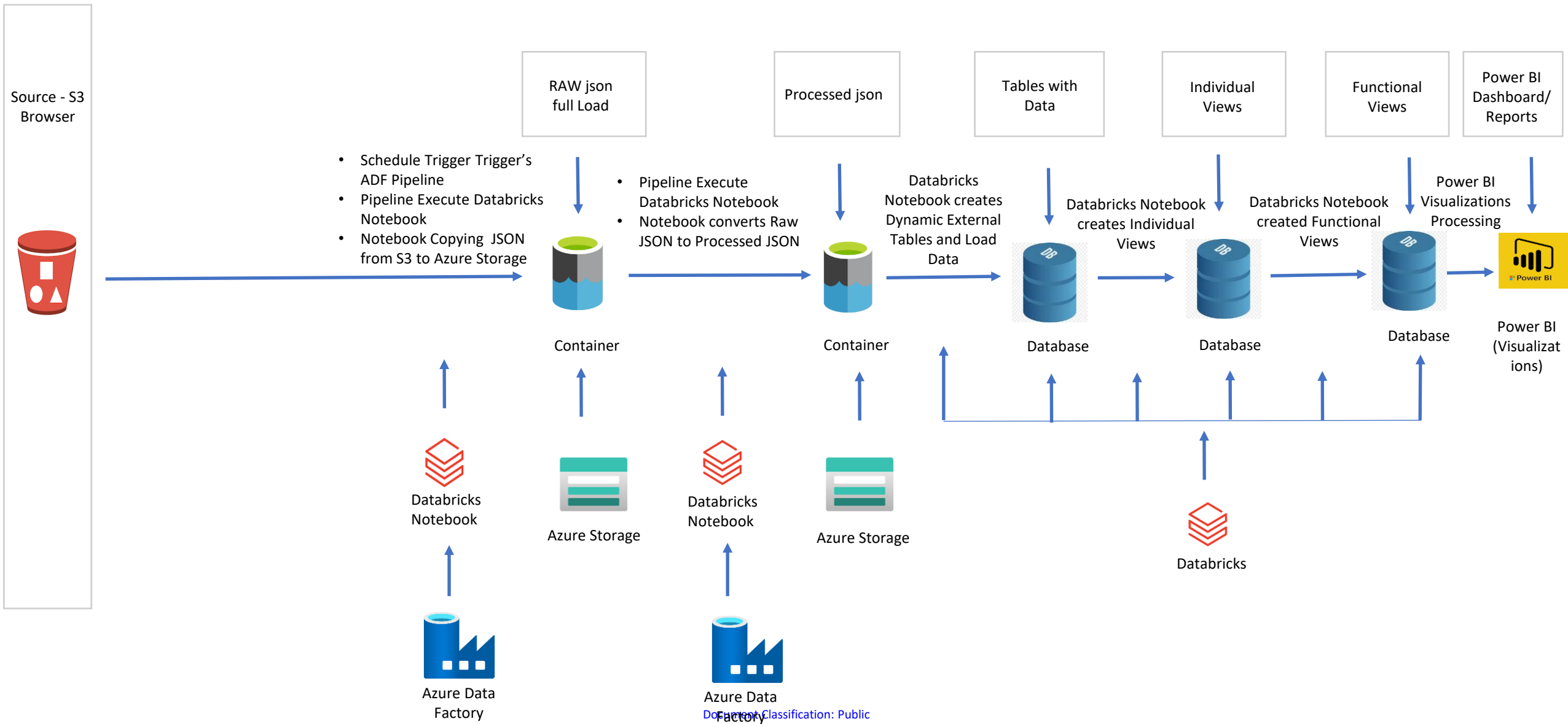


Staffing Origin Dashboards

Full Load Architecture





Source
- S3
Browser



Azure
Data
Factory



Output

Microsoft Azure | Data Factory | io-da-dev-ci-so-df

Search factory and documentation

chennakesava.avvaru@infoorigin.com

master branch | Validate all | Save all | Publish

Preview experience

PL_SO_FULL_LOAD_... | PL_SO_FULL_LOAD_...

Saved | Save as template | Validate | Debug | Add trigger

```
graph LR; GetMetadata[Get Metadata  
Get Table Names] --> ForEach[ForEach  
Ingest Each File  
Activities: 6  
Check if full load ... --> Set Table Name --> Copy AMS3 to ... --> Copy To Raw Folder --> +]; ForEach --> Notebook[Notebook  
SO-FULL-LOAD-I...]; Notebook --> ExecutePipeline[Execute Pipeline  
PL-BACKUP-SO-A...  
PL_SO_BACKUP_S3_A...]; Notebook --> Lookup[Lookup  
MAIL_ID]; ExecutePipeline --> Web[Web  
MAIL_NOTIFICATI...];
```

The diagram illustrates an Azure Data Factory pipeline. It begins with a 'Get Metadata' activity (Get Table Names). This is followed by a 'ForEach' loop (Ingest Each File) containing six activities: 'Check if full load ...', 'Set Table Name', 'Copy AMS3 to ...', 'Copy To Raw Folder', and an addition symbol. The pipeline then proceeds to a 'Notebook' activity (SO-FULL-LOAD-I...). From the 'Notebook', the flow splits into two parallel paths: one leading to an 'Execute Pipeline' activity (PL-BACKUP-SO-A..., PL_SO_BACKUP_S3_A...) and another leading to a 'Lookup' activity (MAIL_ID). Both paths converge and lead to a final 'Web' activity (MAIL_NOTIFICATI...).



NB_SO_PY_FULL_LOAD_DEV_CURRENT_RELEASE_OLD02052023

Python

Run all

Terminated

Schedule

Share

File Edit View Run Help [Last edit was now](#) [Provide feedback](#)

Python

```
1 # Association Table
2 associationFlag = True
3 # association table
4 try:
5     associationDF = spark.read.parquet("/mnt/st3/staffing-origin-landing-zone/association/")
6
7 except AnalysisException as ae:
8     print("association folder is empty")
9     associationFlag = False
10 if (associationFlag):
11     associationDF.count()
12     associationDF.na.fill("")
13
14 # Convert to INT
15 associationDF = associationDF.withColumn("ASSOCIATION_ID", col("ASSOCIATION_ID").cast(IntegerType()))
16 # Convert columns to DateType
17 associationDF = associationDF.withColumn("AE_INSERT_TS", col("AE_INSERT_TS").cast(TimestampType()))
18 associationDF = associationDF.withColumn("AE_UPDATE_TS", col("AE_UPDATE_TS").cast(TimestampType()))
19 associationDF = associationDF.withColumn("CANDIDATE_START_TS", col("CANDIDATE_START_TS").cast(DateType()))
20 associationDF = associationDF.withColumn("CANDIDATE_END_TS", col("CANDIDATE_END_TS").cast(DateType()))
21 # Add date/year columns for partition
22 associationDF = associationDF.withColumn("DATE_COL",
23     current_timestamp()).withColumn("YEAR", year(col("DATE_COL")).cast(StringType()))
24
25 #Adding HASH key
26 associationDF= associationDF.withColumn("ROW_HASH", sha2(concat_ws("||", *associationDF.columns), 512))
27 # Write to Curated Folder
28 associationDF.coalesce(1).write.partitionBy("year").mode("append").format("delta").save("/mnt/st3/staffing-origin-processed/association/")
29 #display(association_incremental_df)
30
```

SO-DB-VIEWS - DEV

Dataset

Refreshed: 7/17/23, 6:07:19 PM

StaffingOrigin DataSets

Candidate Search Report

...

Association Remark Report

...

Association Status Report

...

Requirement Report

...

Email Reports Report

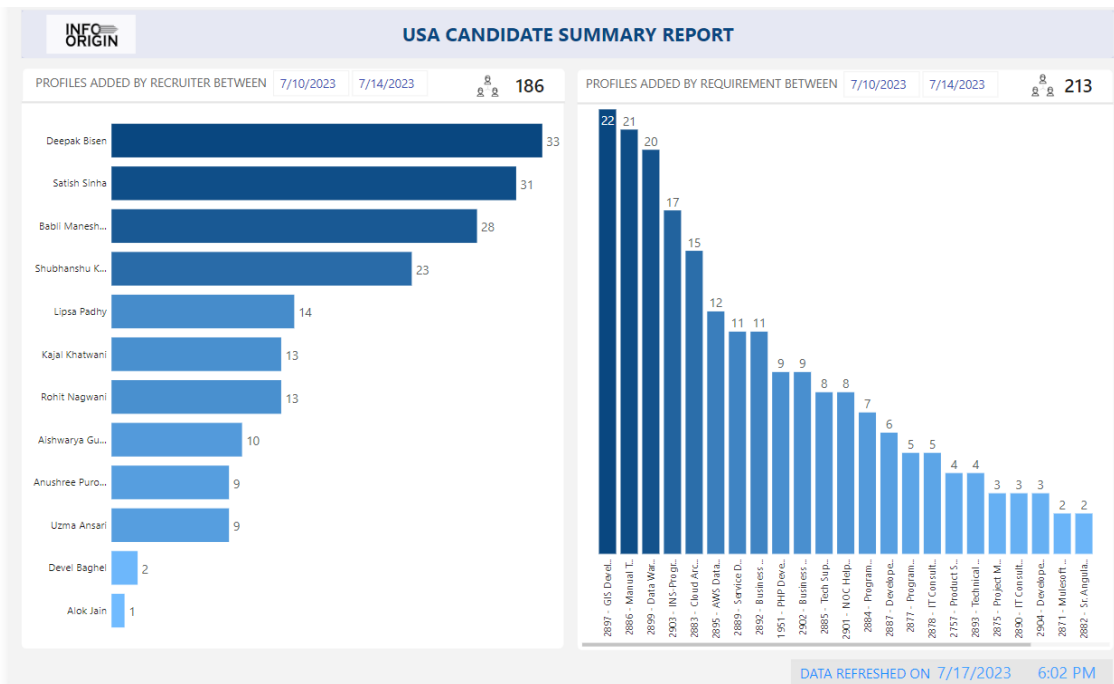
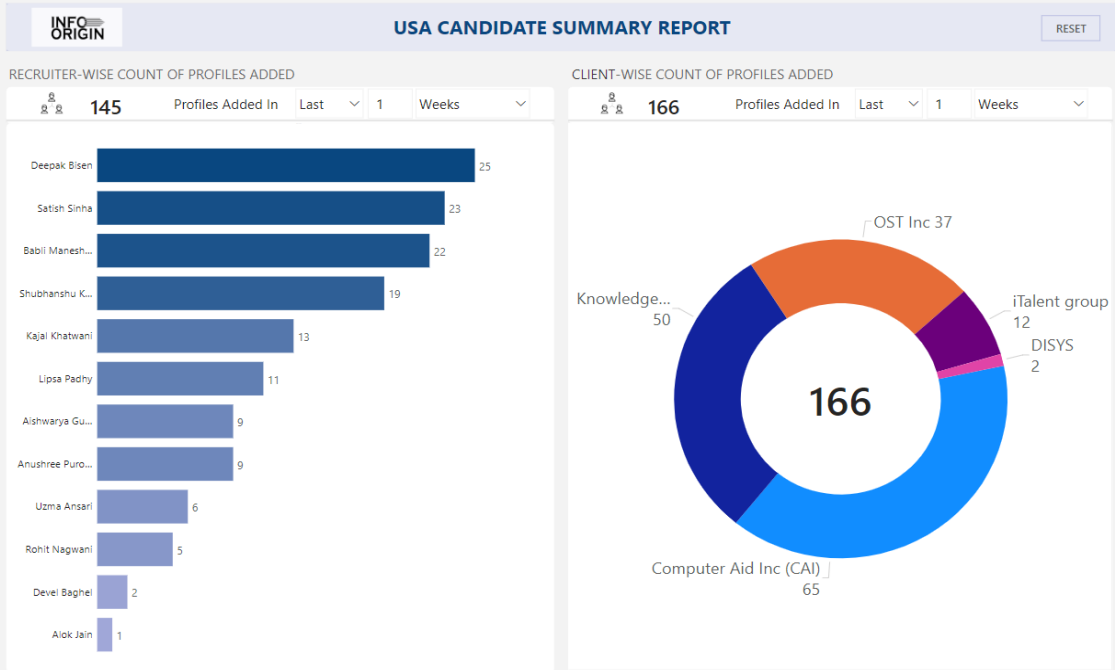
...

Candidate Summary Report

...

USA Report

...



INFO ORIGIN

VONAGE CALLS REPORT

12-01-2023 - 16-07-2023

9065

INBOUND

17893

OUTBOUND

27031

TOTAL

20673

ANSWERED

3961

MISSED

516

ATTEMPTED

1851

VOICEMAIL

30

BLOCKED

USER-WISE COUNT OF ANSWERED CALLS AND THEIR TOTAL DURATION (IB = INBOUND, OB = OUTBOUND)

USER	1	2	3	4	5	6
Aishwarya Gupta	IB-03 00:07, OB-69 00:29	IB-05 00:03, OB-41 00:53	IB-04 00:09, OB-109 02:20	IB-02 00:02, OB-16 00:09	IB-03 00:07, OB-60 00:40	IB-05 00:24, OB-48 00:29
Alok Jain	OB-03 00:04	IB-04 00:03, OB-09 00:29	IB-09 00:15, OB-07 00:06	OB-04 00:00	IB-01 00:01, OB-12 00:10	IB-03 00:04, OB-07 00:10
Anushree Purohit	IB-02 00:03, OB-86 00:50	IB-05 00:14, OB-114 01:28	IB-05 00:15, OB-80 00:55	IB-01 00:03, OB-68 00:45	IB-01 00:01, OB-80 00:39	IB-02 00:03, OB-117 01:14
Babli Maneshwar	IB-01 00:03, OB-13 00:27	IB-04 00:02, OB-16 00:30	IB-05 00:13, OB-48 00:50	IB-05 00:21, OB-28 00:20	IB-02 00:08, OB-37 00:37	IB-05 00:12, OB-49 00:58
Deepak Bisen	IB-07 00:26, OB-73 00:53	IB-05 00:06, OB-71 00:56	OB-66 01:04	IB-03 00:07, OB-32 00:35	IB-02 00:03, OB-79 01:10	IB-05 00:12, OB-71 01:26
Devel Baghel	IB-01 00:02, OB-19 00:12	IB-01 00:06, OB-34 00:23	IB-03 00:02, OB-48 00:29	IB-08 00:11, OB-37 00:25	IB-01 00:01, OB-21 00:15	IB-01 00:05, OB-11 00:19
Ghana Praskah	IB-03 00:02		IB-01 00:00		IB-02 00:02	IB-02 00:00
Info Origin	IB-06 00:10, OB-38 00:26	IB-14 00:19, OB-72 01:00	IB-19 00:20, OB-73 01:15	IB-05 00:11, OB-24 00:20	IB-16 00:13, OB-106 00:45	IB-18 00:29, OB-122 01:58
Main Auto Attendant	IB-07 00:01	IB-12 00:05	IB-18 00:17	IB-02 00:00	IB-02 00:00	IB-12 00:09
Naveen Kumar		IB-01 00:06, OB-01 00:00	IB-01 00:01, OB-07 00:16		IB-01 00:01, OB-05 00:04	IB-04 00:06, OB-07 00:05
Nitin Kumar	IB-06 00:11, OB-14 00:14	IB-10 00:09, OB-44 00:34	IB-16 00:20, OB-30 00:31	IB-08 00:12, OB-06 00:03	IB-05 00:02, OB-19 00:20	IB-08 00:07, OB-35 00:28
Pankaj Bhojwani	OB-03 00:10	IB-01 00:01	IB-02 00:12, OB-01 00:01	IB-02 00:06	IB-01 00:01, OB-02 00:03	IB-03 00:04, OB-01 00:00
Prashant Kedambady	IB-03 00:01, OB-06 00:05	IB-02 00:26, OB-02 00:03	IB-02 00:03			OB-02 00:05
Preeti Chauhan			IB-01 00:00			
Satish Sinha	IB-07 00:16, OB-42 00:43	IB-11 00:42, OB-53 00:44	IB-19 00:33, OB-78 01:30	OB-35 00:33	IB-19 00:31, OB-72 01:13	IB-20 00:38, OB-92 01:22
Uzma Ansari	IB-07 00:21, OB-15 00:23	IB-04 00:12, OB-24 00:23	IB-11 00:30, OB-103 01:32	IB-03 00:04, OB-16 00:12	IB-14 00:40, OB-21 00:34	IB-15 00:26, OB-66 00:55
Total	IB-53 01:49, OB-381 05:01	IB-79 02:40, OB-481 07:27	IB-116 03:15, OB-650 10:54	IB-39 01:22, OB-266 03:25	IB-70 01:57, OB-514 06:34	IB-103 03:05, OB-628 09:34