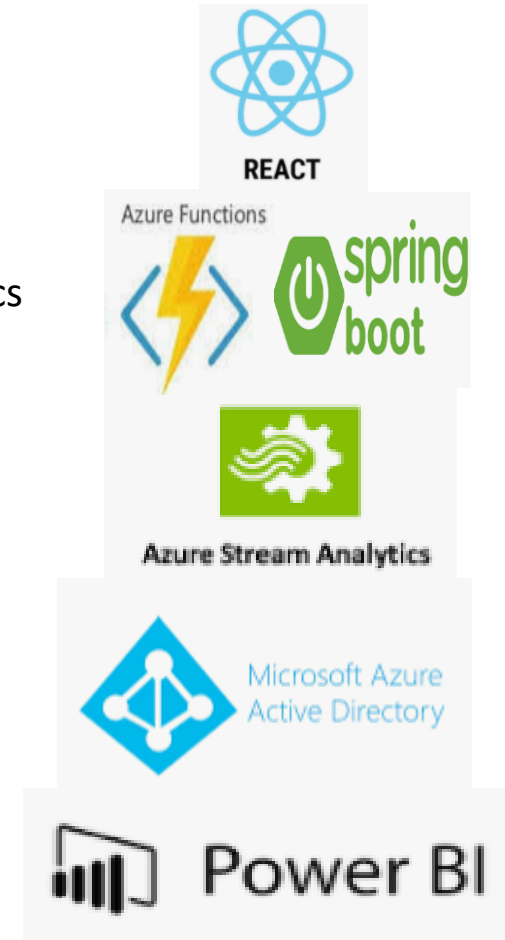
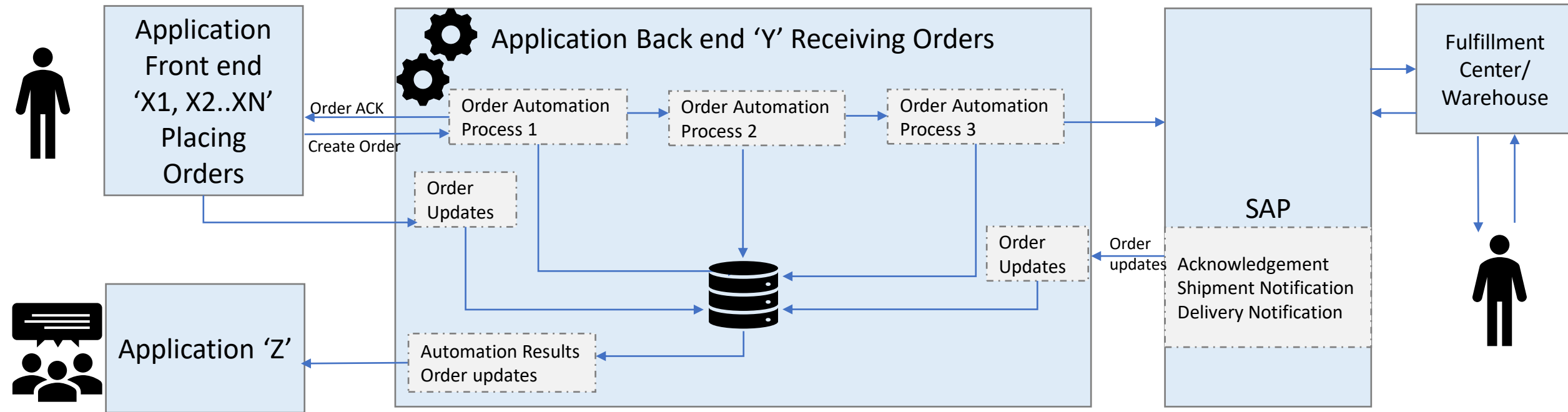


## Reporting

1. Business use case Example
2. Design : Scheduled refresh reports in MS Power BI using MS Azure Active Directory
3. Design : Live Streaming reports in MS Power BI using MS Azure Active Directory & MS Azure Stream Analytics
4. Detailing all Components
5. Workflow, One-time setup - MS Azure Active Directory & Azure Stream Analytics
6. ReactJS, MS Power BI, MS Azure Active Directory, micro service – Detailed Design for Reporting
7. Scheduled reports, Live Streaming reports demo & Azure App Functions demo
8. Pre-requisites for Reporting



# 1. Business use case

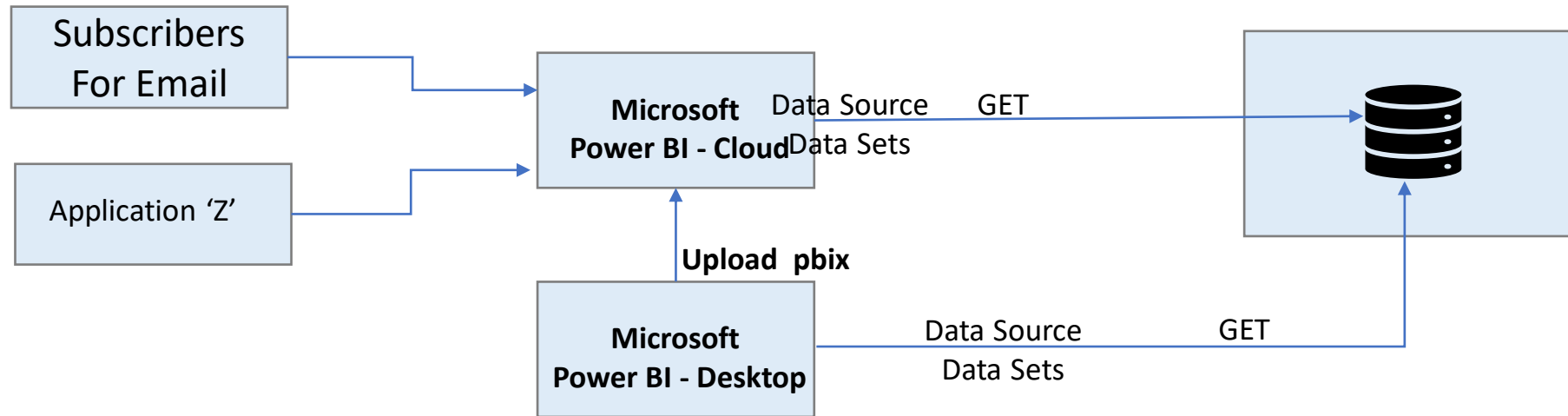


## Application Z or Business needs following

1. Number of orders placed from Application X, sent to automation for Application Y
2. How many processed in Application Y – Success, Failed, In Process...any other status
3. Reports in near real time and batched updates
4. Batched updates report with slicing and dicing available to see the order volume for any duration

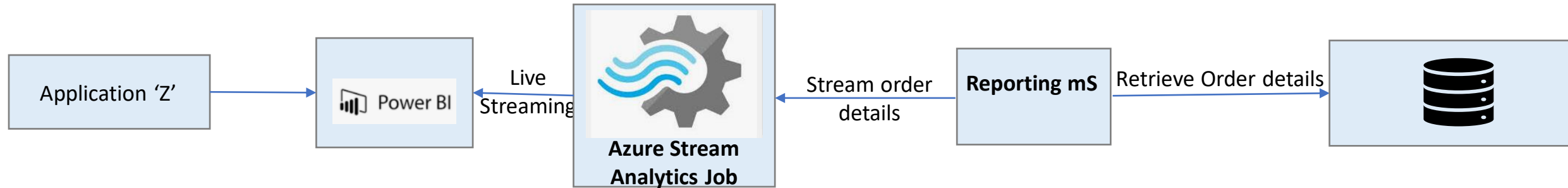
Real-time	When you need information processed immediately (such as at a bank ATM)
Near real-time	When speed is important, but you don't need it immediately
Batch	When you can wait > 5 mins

## 2. Scheduled refresh reports using MS Power BI, MS Azure Active Directory



- MS Power BI can be scheduled for 48 refreshes ( @ 30 mins interval per hour)
- Application 'Z' to fetch refresh reports from MS Power BI @ 5<sup>th</sup> min and @ 35<sup>th</sup> min in an hr

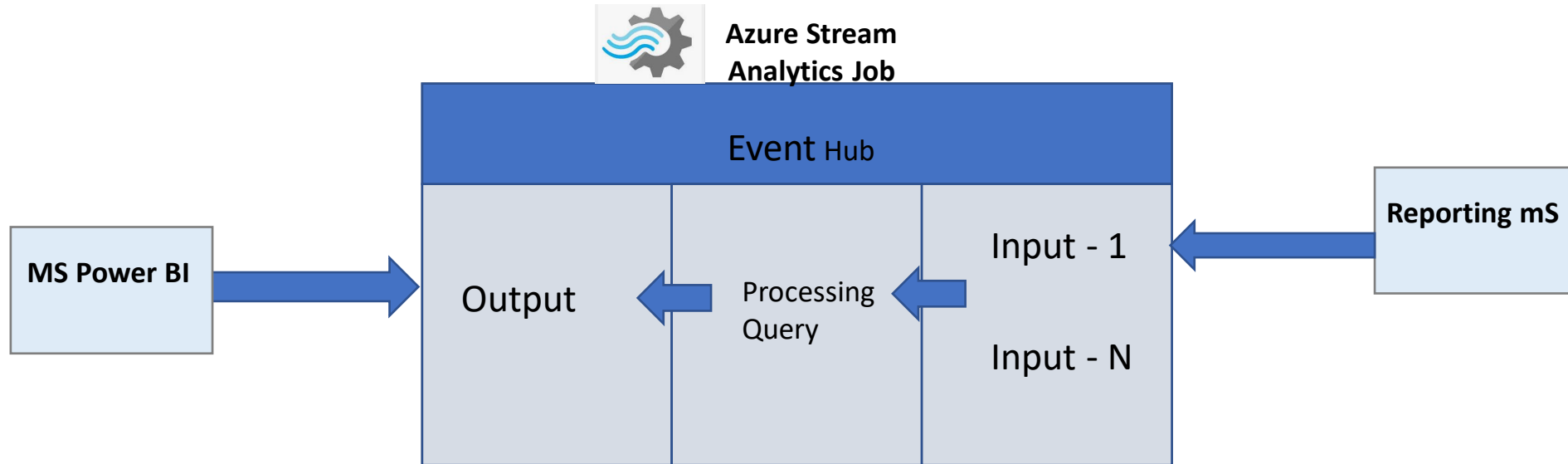
### 3.1 Live Streaming reports using MS Power BI, MS Azure Active Directory, MS Azure Stream Analytics



#### Live Streaming

- LIVE Streaming jobs configured in “Azure Stream Analytics” and shows status of orders in near real time

## 3.2 Reporting – Azure Stream Analysis



# 4.1 Component details

## Components involved for Batch reporting

1. DB for data
2. MS Power BI Desktop version
  1. To create PBIX ( report)
  2. Connection parameters to DB
  3. Charts as needed for display of reports
3. Application 'Z' for retrieving reports
4. MS Azure Active Directory
  1. Create tenant and Subscription( optional)
  2. Create users and have MS Power BI Cloud account ( as Step 3)
  3. App registration
  4. Create Security group
5. MS Power BI Cloud account
  1. To create workspace
  2. To publish PBIX report to Cloud account
  3. Schedule refresh intervals
  4. Email Subscription for publishing reports via email
  5. Admin consent to 'Allow service principals to use Power BI APIs'
  6. Provide workspace access to Security group( Service principal)

## 4.2 Work-flow in MS Azure Active Directory for Reporting

### Components involved for Live Stream reporting

1. MS Azure Stream Analytics Job
  1. To create Input event for accepting data
  2. To create Output for MS Power BI data set
  3. Query editor to format output
2. Reporting mS
  1. To pull data from DB
  2. To publish data to MS Azure Stream Analytic Job
3. MS Power BI Cloud account
  1. To fetch data from Live Streaming data set
4. Application 'Z' for retrieving reports

## 5.1 Workflow, One-time setup - MS Azure Active Directory & MS Power BI

<https://portal.azure.com>

Login - emailID@org.com

Navigate to  
Azure Active Directory

Create 1 user  
(pbembed)

Login using pbembed

App Registrations  
Register your application  
"myapplication"

Above steps generates –  
Client ID for the  
applicaiton

Generate  
Client secret code

Grant Power BI Service API  
Permissions - Scope

Grant Admin Consent

Navigate to  
Groups

Security group creation  
"myapplication-security-group"  
Security object ID

Assign client ID as member to  
security group created

Final outcome

1. Client ID
2. Client Secret
3. Tenant ID  
(org.onmicrosoft.com)

1. Security object ID

<http://app.powerbi.com>

Login - pbembed@org.com

Navigate to  
Admin Portal  
Possible on PRO license only

Navigate to  
Tenant Settings

Enable check mark - "Allow  
service principals to use  
Power BI APIs"

Provide security group  
"Security object ID"

Navigate to  
MyWorkspace

Navigate to  
Workspace Access

Navigate to  
Workspace Access

Provide client ID  
"Client ID"

Navigate to  
MyWorkspace

Go to any report and from  
URL, we can get  
"report ID"



## 5.2 Workflow, One-time setup - MS Azure Active Directory & MS Power BI

<https://portal.azure.com>

Login - emailID@org.com

Navigate to  
Azure Active Directory

Create new Tenant

Create 2 users  
(admin, pbembed)

Login using pbembed

App Registrations  
Register your application  
"myapplication"

Above steps generates –  
Client ID for the  
applicaiton

Generate  
Client secret code

Grant Power BI Service API  
Permissions - Scope

Grant Admin Consent

Navigate to  
Groups

Security group creation  
"myapplication-security-group"  
Security object ID

Assign client ID as member to  
security group created

Final outcome  
1. Client ID  
2. Client Secret  
3. Tenant ID  
(tommjerry.onmicrosoft.com)  
1. Security object ID

<http://app.powerbi.com>

Login - pbembed@org.com

Navigate to  
Admin Portal  
Possible on PRO license only

Navigate to  
Tenant Settings

Enable check mark - "Allow  
service principals to use  
Power BI APIs"

Provide security group  
"Security object ID"

Navigate to  
MyWorkspace

Navigate to  
Workspace Access

Navigate to  
Workspace Access

Provide client ID  
"Client ID"

Navigate to  
MyWorkspace

Go to any report and from  
URL, we can get  
"report ID"

## 5.3 Workflow, One-time setup - MS Azure Stream Analytics

<https://portal.azure.com>

Login

Navigate to Event Hubs,  
create new policy

Navigate to Stream  
Analytics Job

Navigate to Managed  
Identity, enable and save  
it

Add Inputs by giving  
details of Event Hub  
created

Add Output by selecting  
PBI.

Login to PBI and to your  
workspace

Under workspace access, provide  
Stream Analytics Job name and  
admin to add it as Service Principal

Edit Query based on Input and  
Output

Start the job and copy the end  
points from policy to application  
which send live streaming

## 5.4 Workflow, One-time setup - MS Azure Stream Analytics

[Home](#) >

# ReportingASAJob

Stream Analytics job

Search (Ctrl+ /) << ▶ Start □ Stop 🗑 Delete

**Overview**

- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems

**Settings**

- Properties
- Locks

**Job topology**

- Inputs
- Functions
- Query
- Outputs

**Configure**

**Monitoring**

Your job is running. Learn how to setup alerts and monitor your job using metrics. →

ochreportingpbi Power BI

Metric	Value
Input Events (Sum) ochreportingasajob	143
Output Events (Sum) ochreportingasajob	143
Runtime Errors (Sum) ochreportingasajob	0

**Resource utilization**

Metric	Value
SU % Utilization (Max) ochreportingasajob	14 %

## 6.1 POC : Report tag in ReactJS

Input from user   reportNeeded = **"Prod1Rpt"**

## <Report

tokenType="Embed"

accessToken= {accessToken}

```
embedUrl= {embedUrl}
```

embedId= {embedId}

&lt;/Report&gt;

```
1 {
2   "datasets": [
3     {
4       "id": "5ffa16ac-a591-46b0-8040-c21a17067c1a"
5     }
6   ],
7   "reports": [
8     {
9       "allowEdit": true,
10      "id": "f5be1bf9-e7de-46e6-8e2e-0818d107027d"
11    }
12  ]
13 }
```

{token}

Post API <https://login.microsoftonline.com/{tenantId}/oauth2/v2.0/token>

With client id, client secret, scope, grant type(client credentials)

## Response – auth token from MS Identity Platform

```
{embedUrl}
```

## Using “reportNeeded”,

search in couchbase document to get embed URL and data set ID

```

1 4
2  "odata.context": "http://wabi-us-east-2-b-primary-redirect.analysis.windows.net/v1.0/myorg/
3  $metadata?Microsoft.PowerBI.ServiceContext.Api.V1.GenerateTokenResponse",
4  "token":
5  "H4sIAAAAAAAB2Ux66KvgBE_-VtsU5mdIsyNCEB17Qw16cc8byv3vG-1qdU1X_Fjx3Y1Y9P3zeJpG5FB9jg
6  CFwz-X-Ivxs1MrZ4H8z2aGcPyi0ppsg7w1JsQnd4fEtyaZHTbE41wOTGFHNgZgQxp3jRZD08Ad11d7mwp0Up8k6
7  ak95jy7Vb7PPhAx1cpQ0qTn1m288fomuI1fmydMQDR7QwXpJq3a5VjA6EteyJk7f3w-EUzq875L80SDtPTtQ1Gyan
8  Vbdp0H41M6hgchIIPAAcYmG4vLKGWv1jPtZMXzQeAkNklbFz2ARAG3fUX2YekFLcKJas10c0i8uNslvGdzR3s
9  UjP1nHm7YaeQdY1jZed1YRhm0K1Y-3e192Rm0K1b-uoqF0K1b3JOVARUE_-Joq9K1h-uoqF0K1b3JOVARUE_-
10 RoZ2mYnrfL2cL8n5pmc081ktXk0K09pj3mefFlYjdzw14JMLSc_3pP3yv3rKjxk05J23610sldK0L17vG0dM3Y
11 CaLnsGdW43S_MushbTas4Wd5_kwJ6K06Inh11K9uNkcnf6PBMQ0K1_oefFJ335s_fjJuoak-S14KxxX313uaRs
12 gby9z104M2SUyberZcTqG5s1EuhK0Mfpj2Gw1dFmNt9F736G6Ue2U0puneEYs0Q4dPpH3KdekksC90XfG0H
13 0jVp_-66z4qkZcs2FP49V1xVJ1Y2H818szjCRRK1L1Cqloq_U_DfsmzoK-dkd-3msJ9v80Q7L8K0EUBQ9bX8G
14 LM080R1F8RH52N1L3V1nhcF8R5j4H0QDhsfAp0jJRM14C4jQmQ4zG2UD00z4m3v6h8a0L0G0E0M0R143
15 UCeG2YS9rL6qTlaow8w1ulbF8wrHQ1KRSYt7y0EOXEQ5X35M2CLjKz8GfTr1_E6j824e6ITme1qC1yKq0BvF
16 OKZda_1808tmU805YocFRUG1t714B1AqU0v1L3LzUhwX5aqaDEXW_fv3BfE9v9q4j-b8qV7dQ4hphFhX1w_F0E
17 h3vY1ZdaKt807Hs4OL14v-w3h0AbV5hg_bkVJvTTL_XEn0yYj45aQ0Z7R80ZU0qC13Gcs37C7SLj8b-GLT0H
18 ZS5CsP50KH0fIskwhrr0W813JUF5jAB9Fe4cz87r2Y9FR7VUzU6FIwepU5Qy_Q0kTPkSt3a07K124m7Q1XA37tCb
19 cte3D31LcYvFvM9-h711NvdnKAdHUu6e34H4Cte-qVC5NyJ7a0zRsDX6ry_-ybt0b61e0ZyQXpsxtV2nqbt1C
20 ade130Yab8BQ95yS54535A0ZDQF37rB85j323PHOEZxscSMK1Ljg8Y_v0PcrgLMDIFAA=-
21 ey3jBhVzdGvYVX3sJioiAhR0cHM6LY9XQUJ13VLTUVU8U1QyLUTUfJ3TUF5W51ZwRpcmVjdC5hbmFseXk0IjY3
22 1dC1s1mVtYmVrNmVhdH9YXZm1Ons1b0gXZK3XUw11ZiQ1OmZbHn1FX0=-",
23
24  "tokenId": "91cc38a3-8ffa-4e4b-a401-a3d8-45401b15d22",

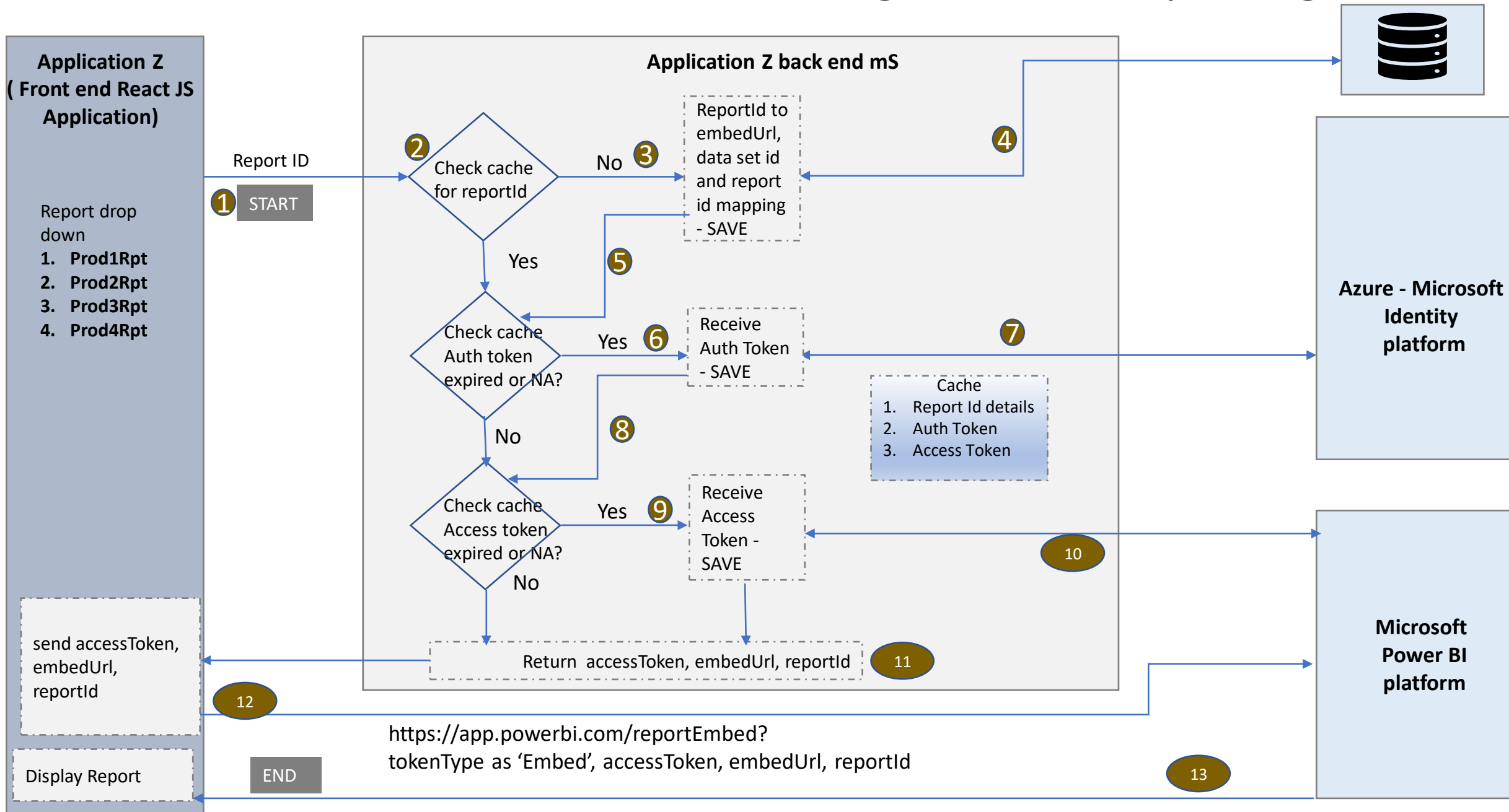
```

```
{embedId}
```

Post API <https://api.powerbi.com/v1.0/myorg/GenerateToken>, with datasets, report ids and auth token

## Response – accessToken from MS Power BI

## 6.2 ReactJS and MS Power BI Integration for Reporting



## 6.3 POC : One time-Setup for each report – embed url and data set id

1. Navigate to below URL and Sign in using the new user created  
<https://docs.microsoft.com/en-us/rest/api/power-bi/reports/getreport>

2. Click “Try it” and

```
HTTP

GET https://api.powerbi.com/v1.0/myorg/reports/{reportId}
```

3. Sign in and in Parameters tab, provide “reportid” from Power BI url of the report. Click run to execute the api

4. The response will have embed url and data set id.

Reports - Get Report

Service: Power BI REST APIs  
API Version: v1.0

Returns the specified report from "My Workspace".

Required scope: Report.ReadWrite.All or Report.Read.All  
To set the permissions scope, see [Register an app](#).

HTTP

GET https://api.powerbi.com/v1.0/myorg/reports/{reportId}

URI Parameters

Name	In	Required	Type	Description
reportId	path	True	string uuid	The report id

Body

JSON

```
{
  "@odata.context": "http://wabi-paas-1-scus-redirect.analysis.windows.net/v1.0/myorg/reports/{reportId}",
  "id": "f5be1bf9-e7de-46e6-8e2e-0818d107027d",
  "reportType": "PowerBIReport",
  "name": "Retail Analysis Sample",
  "webUrl": "https://app.powerbi.com/groups/me/reports/f5be1bf9-e7de-46e6-8e2e-0818d107027d",
  "embedUrl": "https://app.powerbi.com/reportEmbed?reportId=f5be1bf9-e7de-46e6-8e2e-0818d107027d",
  "isOwnedByMe": true,
  "datasetId": "5ffa16ac-a591-46b0-8040-c21a17067c1a"
}
```

MANAGE ACCESS TOKENS	
All Tokens    Delete ▼	Token Name      Reporting-Token 🗑️
Reporting-Token	Access Token eyJ0eXAI0ijKV1QiLCjHbGciOijSUz1NiIsIng1dCI6ImtnMkxZczJUMENUaklmajRydDZKSXluZW4zOCIsImtpZC16ImtnMkxZczJUMENUaklmajRydDZKSXluZW4zOCJ9.eyJhdWQIOijodHRwczoV2FuYWw5c2lzLndpbmRvd3MubmV0L3Bvd2VyYmkvYXBpliawXNzljoiaHR0cHM6Ly9zdHMu d2luZG93cy5uZXQvZjY1ZTEyODctMWVlYS00NmVILWI3ZWYtMWQ3 MmRkNmUyZDBILylslmlhdCI6MTYwMTY1Nzg5MCwibmJmloxNjAxN jU3ODkwLCJleHAiOiE2MDE2NjE3OTAsImFpbyI6IkUyUmdZSkfAN2Y wMXJWN3oySjNGYjVKVno5cnRCQUE9liwiYXBwaWQiOi0NDZjM2U3Y S1iYjBmLTQ3YTktYjA5Yy00OGVmZGM5N2RmYjkiLCJhcHBpZGFjcil6lj EilCjpZHAiOiJodHRwczoV2FuYWw5c2lzLm5ldC9mNjVIMTI4Ny 0xZWVhLTQ2ZWUyYjdIZi0xZDcyZGQ2ZTJkMGUvliwib2lkloiN2lyMTY 2MWYtMTI0OC00NDE3LWE5Y2EtYjQ0ZWJhN2Y0YzJmliwicmgiOilwLk FBQUFoeEpIXVvZTdrcTYTM3eDF5M1c0dERuby1iRVFQdTzsSHNKekEk 3OXIYMzdsMUFBQS4iLCJyb2xlcyI6WyJUZW5hbncUUmVhZFdyaxRILk FsbClslRIlbmFudC5SZWFkLkFsbcjdlCjZzdWliOi0i3YjlxNjYxZi0xMjQ4LT Q0MTctYjY1iNDRIYmE3ZjRjMmYiLCJ0aWQiOiJmNjVIMTI4Ny0xZW VhLTQ2ZWUyYjdIZi0xZDcyZGQ2ZTJkMGUiLCJ1dGkiOiJacFJfODIQZEIF ZUJhcW9RQkdKUEFRliwidmVyljoiMS4wLn0.WXanWvMflifHgigdrRrR1 D3LNK...

## 6.6 POC : One time-Setup for each report – embed url and data set id

Postman :

Generate datasetid for report in MS Power BI( can be done for any report)

GET API <https://api.powerbi.com/v1.0/myorg/reports/{reportid}> -> reportid shall be from Power BI url

<https://api.powerbi.com/v1.0/myorg/reports/71e06a2b-1310-4339-9abf-2db0d8f99346>

## Auth Token – Received from MS Identity Platform thru login api

## Response

## Embed URL,

Data set ID

## Configuration in Couchbase

```
[
  {
    "report-1": {
      "embedUrl": "<embedUrl>",
      "datasetId": "<datasetId>",
      "reportId": "<reportId>"
    }
  }
]
```

```
1 {
2   "@odata.context": "http://wabi-us-east2-b-primary-redirect.analysis.windows.net/v1.0/myorg/  

3     $metadata#reports/$entity",
4   "id": "b9169c2f-58ec-42e4-b21e-72e8202dd3dd",
5   "reportType": "PowerBIReport",
6   "name": "Customer Profitability Sample",
7   "webUrl": "https://app.powerbi.com/groups/me/reports/b9169c2f-58ec-42e4-b21e-72e8202dd3dd",
8   "embedUrl": "https://app.powerbi.com/reportEmbed?reportId=b9169c2f-58ec-42e4-b21e-72e8202dd3dd&  

9     config=eyJjbHVzdGVyVXJsIjoiaHR0cHM6Ly9XQUJ1LVVTLUVBU1QyLUItUFJ1TUFWS1yZWRpcmVjdC5hbmFseXNpcy53aW5k  

10    b3dzLm5ldCIiImVtYmVkrMvHdHvyZXMiOmsibW9kZXJ1Rw1iZWQiOnRydWV9fQ%3d%3d",
11   "isOwnedByMe": true,
12   "datasetId": "d8d09301-81f9-4068-ad9d-bc71bf3b360d"
13 }
```

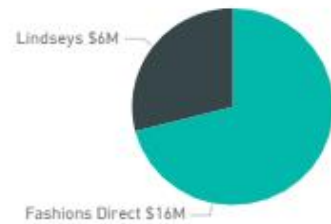


# 7.1 ReactJS and MS Power BI Integration for Scheduled Reporting

## Report

### Store Sales Overview

This Year Sales by Chain



10  
New Stores

104  
Total Stores

This Year Sales by PostalCode and Store Type

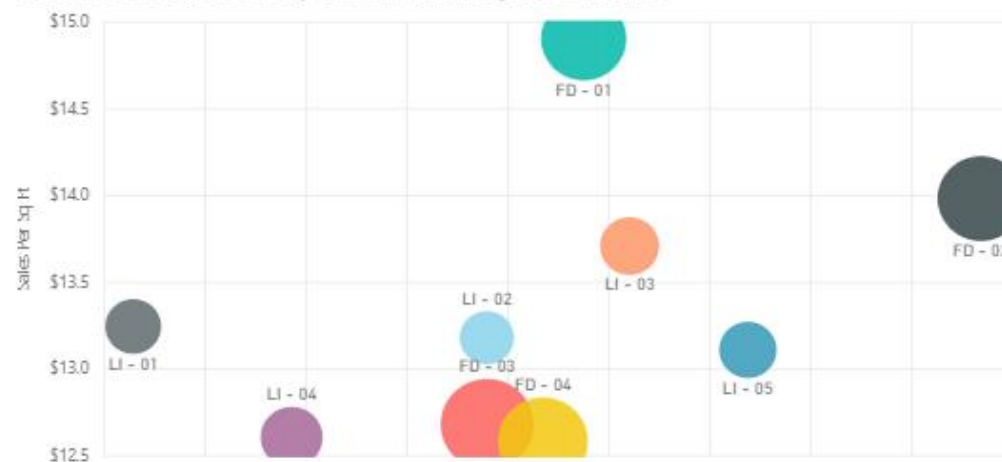
Store Type ● New Store ● Same Store



Total Sales Variance by FiscalMonth and District Manager



Total Sales Variance %, Sales Per Sq Ft and This Year Sales by District and District



Filters

Search

Filters on this page

Chain is (All)

City is (All)

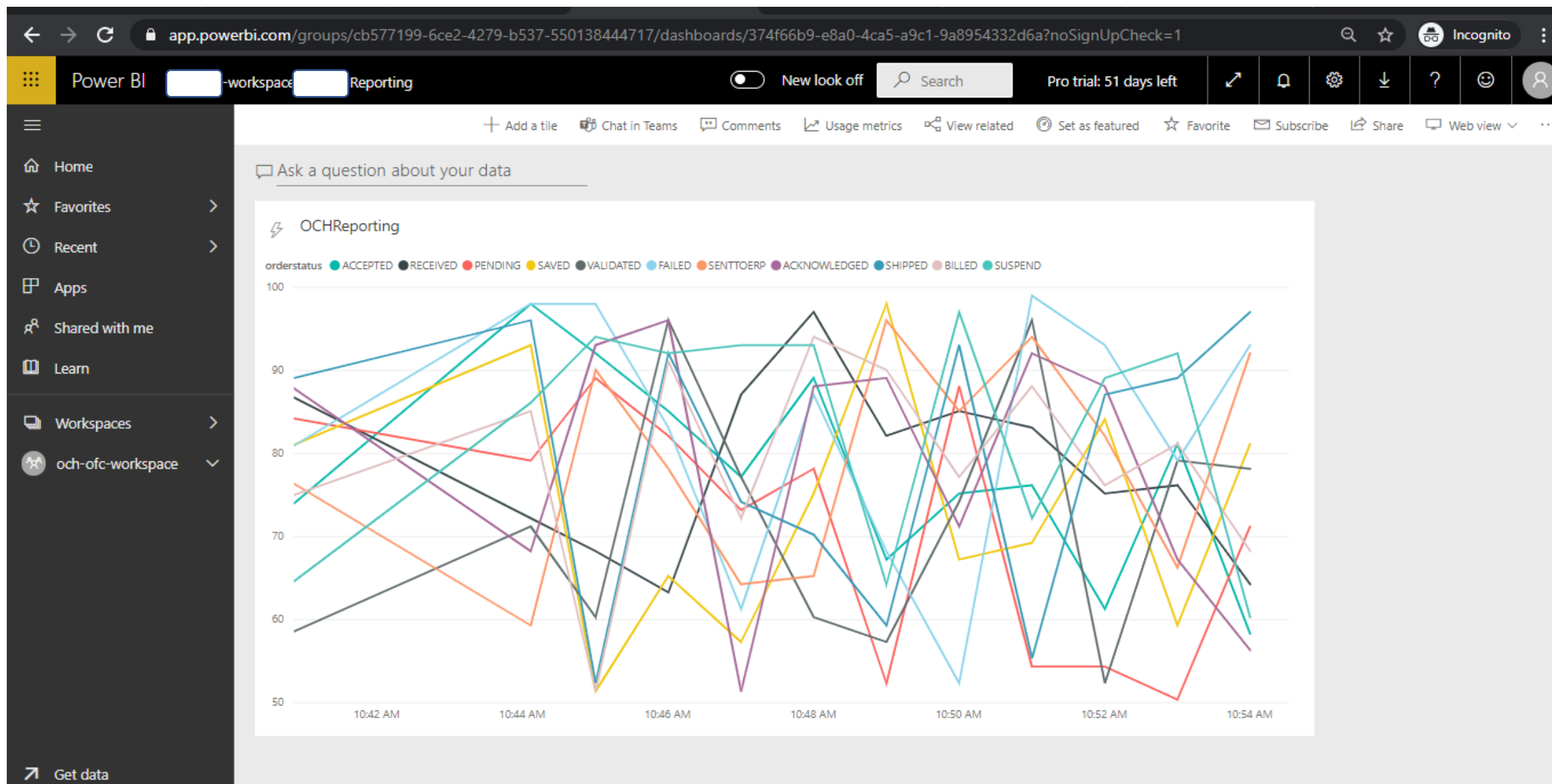
District is (All)

Name is (All)

Open Month is (All)

Store Type is (All)

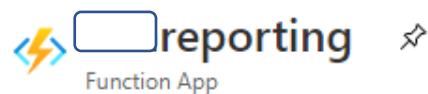
# 7.2 ReactJS and MS Power BI Integration for Near Real Time stream Reporting



# 7.3 Azure App Functions – Demo( can replace reporting mS)

REPORTING DEMO

Home > ochreporting



Search (Ctrl+/)



[Browse](#) [Refresh](#) [Stop](#) [Restart](#) [Swap](#) [Get publish profile](#) [Reset publish profile](#) [Download app content](#) [Delete](#) ...

Overview

- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Security
- Events (preview)

Functions

Functions

- App keys
- App files
- Proxies

Deployment

Deployment slots

Essentials

Resource group (change)	: java-functions-group	URL	: <a href="https://ochreporting.azurewebsites.net">https://ochreporting.azurewebsites.net</a>
Status	: Running	Operating System	: Windows
Location	: West US	App Service Plan	: <a href="#">java-functions-app-service-plan (Y1: 0)</a>
Subscription (change)	: <a href="#">Azure subscription 1</a>	Properties	: <a href="#">See More</a>
Subscription ID	: a67c6bef-e70d-40ab-a9d2-3b1c60d3c1b9	Runtime version	: 3.0.14492.0
Tags (change)	: <a href="#">Click here to add tags</a>		

Metrics [Features \(9\)](#) [Notifications \(0\)](#) [Quickstart](#)

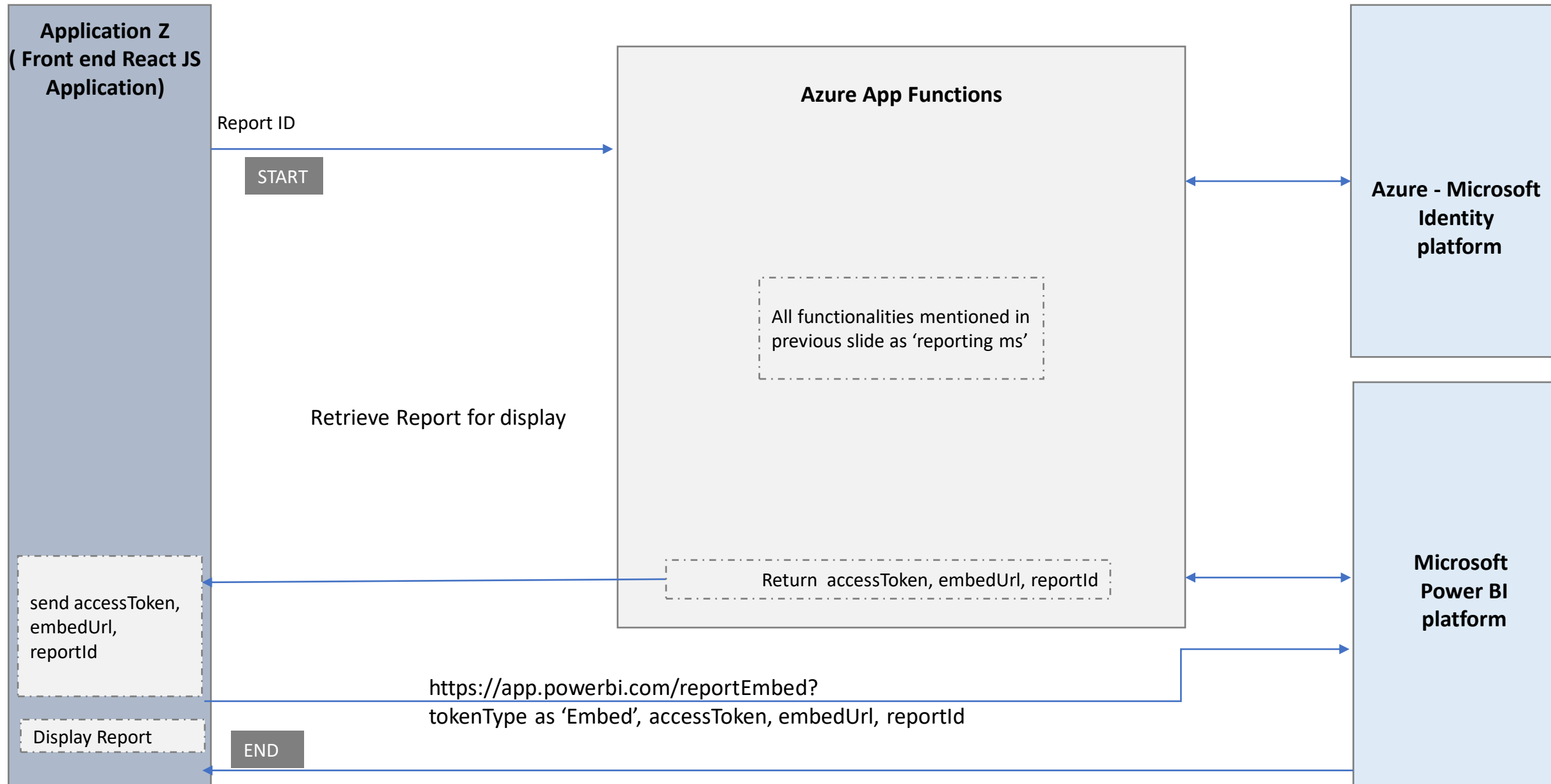
Memory working set



Function Execution Count



## 7.4 Azure App Functions – Demo( can replace reporting mS)



## 8. Pre-requisites for Reporting

1. Power BI Pro license for subscription of reports
2. 'reportinguser' required in existing tenant for getting token from MS Identity Platform
3. Existing tenant admin to grant Power BI api permissions and grant admin consent to 'reportinguser'
4. MS Azure subscription to create Azure Stream Analytics job & Azure App Functions