**Current Data calls made from UI side**

We make below 3 data calls in sequence, which are dependent on each other

**1. Notify that we are exporting to powerBI**

Api call - `https://api.powerbi.com/v1.0/myorg/groups/${this.props.groupId}/reports/${reportId}/ExportTo`,

Method – POST

Headers - {

'Content-type': 'application/json',

Authorization: Access\_token

},

payload : {

format,//(pdf/ppt)

powerBIReportConfiguration: {

settings: {

includeHiddenPages: true

}

},

retry //this is false always

}

Response received :

{

@odata.context: "http://wabi-us-north-central-f-primary- redirect.analysis.windows.net/v1.0/myorg/groups/18dae83f-a44c-4166-bd62-18262c08797c/$metadata#exports/$entity"

createdDateTime: "2021-05-10T04:44:00.9394621Z"

expirationTime: "0001-01-01T00:00:00Z"

id: "MS9CbG9iSWRWMi03ZjVhMzlhZC1lYjY2LTRiM2MtOGJjMS1kMWFhMGQwOWEyMjktTDBhRWN1WXRNMlVVdi5EMmNYMkpsbkNmOW1lenJpZ2R5U3RoT0N5Lg=="

lastActionDateTime: "2021-05-10T04:44:00.9394621Z"

percentComplete: 0

reportId: "8001691e-d400-43f0-86ac-1e5c9f6dc9d7"

status: "NotStarted"

}

**Use the "id" value in above response as "exportId" in below data call**

**2. To get download percentage or download tracker**

Api call - `https://api.powerbi.com/v1.0/myorg/groups/${groupId}/reports/${reportId}/exports/${exportId}`,

Headers {

'Content-type': 'application/json',

Authorization: Access\_token

},

Method - GET

Response received : {

@odata.context: "http://wabi-us-north-central-f-primary-redirect.analysis.windows.net/v1.0/myorg/groups/18dae83f-a44c-4166-bd62-18262c08797c/$metadata#exports/$entity"

createdDateTime: "2021-05-10T04:44:00.9394621Z"

expirationTime: "0001-01-01T00:00:00Z"

id: "MS9CbG9iSWRWMi03ZjVhMzlhZC1lYjY2LTRiM2MtOGJjMS1kMWFhMGQwOWEyMjktTDBhRWN1WXRNMlVVdi5EMmNYMkpsbkNmOW1lenJpZ2R5U3RoT0N5Lg=="

lastActionDateTime: "2021-05-10T04:44:01.1968235Z"

percentComplete: 0

reportId: "8001691e-d400-43f0-86ac-1e5c9f6dc9d7"

reportName: "ROI & Free Cash Flow V2.0"

status: "Running"

}

* **The above call is made recursively for every 30 seconds until ,the status in above response changes to "Succeeded" and percentageComplete is 100.**
* **Use the reportName from above data call for 3rd data call mentioned below**
* **Also Use the "id" value from the response of export data call(1st data call) as "exportId" in below data call**

**3. fetch the PowerBi report**

Api call - `https://api.powerbi.com/v1.0/myorg/groups/${groupId}/reports/${reportId}/exports/${exportId}/file`,

Headers {

'Content-type': 'application/json',

Authorization: Access\_token

},

Method - GET

responseType: 'blob',

**Response received - information will be in blob type, which will be converted to file format (either PDF or PPT) at client side.**

**Design approaches**

Case 1. Report Found in cache

Cache

API

PowerBI

UI



1.Using response from API, we convert response into req file format and download

2. Show 0% at start of data call and show 100% when API responds.

Drawback - No intermediate percentage.

Case 2.Report not Found in cache (Make data calls from UI if not found in cache)



1. Show 0% at start of data call and when cache returns response stating that report is not found in cache, then use the current existing approach of directly communicating the powerBI for response and to get progress percentage.

2. Then send one more data call to update the cache from UI.

(or)

Case 2) Report not Found in cache (Make data calls from API layer)



1. Show 0% at start of data call and we should continuously send the response of percentage call to UI .
2. once percentage call finishes with 100% then API part fires “Get file “ call and

* Case 1) gets valid response .
* Update the cache
* Only After updating cache, return the response to UI .

Sending multiple percentage response initially and final PowerBi response needs to be validated from backend

1. Using response from API, we convert response into req file format and download

If any one of intermediate call fails, then from API, send exception . Using this exception, at UI in catch block we will default to existing behaviour of communicating directly to powerBI.

Case 3. Use Scheduler Service to update cache periodically with all reports and Expose API endpoint to get the reports from cache



1.Run a scheduler service periodically which will do below set of actions

* Get all reports from DB (from view irpc.vw\_report\_details)
* From each report, get the reportId , groupId . Using this 2 data information, fire the 3 powerBI sequential requests .
* Copy the PowerBI report response from above PowerBI data call, and update the cache for the corresponding report.
* Now Reports response can be directly downloaded from the prefetched cache instead of firing 3 PowerBI data calls everytime from UI

2. Expose an Rest Endpoint , when fired from UI or PostMan, will directly fetch the powerBI response for the corresponding report and copy the response into requested format file (PDF/PPT) and download the same.

Cache Structure .

Cache will be having 3 columns

ReportId - Unique ReportId of the each report acting as primary key

PDF\_Response – PDF response for the report

PPT\_Response – PPT response for the report

Open Questions

1. Are there any security constraints if we store reportId in cache?

Constraints/Limitations on PowerBI

1. After downloading 3 or 4 reports , then I am getting error from PowerBI like

{"error":{"code":"ActivityLimitReached","message":"You have too many pending requests to make another right now. Please wait."}}

We are also getting other errors from PowerBi sometimes like, mentioned below

1. InvalidRequest, Report requested for export is not on dedicated capacity
2. {"Message":"An error has occurred."} // Unknown
3. {"error":{"code":"InvalidRequest","message":"Export report requires effective identity to be provided for the report's dataset"}} -> Resolvable

Prod

GOWR -> {"error":{"code":"InvalidRequest","message":"Export report requires effective identity to be provided for the report's dataset"}}

Initiative Tracker - Downloaded Successfully

CTU - Downloaded Successfully

Sam's P&L -> {"Message":"An error has occurred."}

Sam's DST -> {"error":{"code":"ActivityLimitReached","message":"You have too many pending requests to make another right now. Please wait."}}

QA -

WMT Seg CFR Packet - Downloaded Successfully - 5.0 MB of file

Plan Ahead - Downloaded Successfully - 4.8 MB

ROI & Free Cash flow - Downloaded Successfully - 5.9 MB

Sam's P&L -> {"error":{"code":"ActivityLimitReached","message":"You have too many pending requests to make another right now. Please wait."}}