Power BI Monitoring 101

2021-06

dev>scope

Rui Romano

+ 15 years "playing" with data

Head of BI Team @ **DevScope** – OPorto, Portugal

#DataArchitect #DataEngineer #PowerBIAddict
#MSFTBI #DataPlatformMVP #PBIPortugal

Rui.Romano@DevScope.net

@RuiRomano

https://www.linkedin.com/in/ruiromano/

https://ruiromanoblog.wordpress.com

https://www.meetup.com/Power-BI-Portugal

Power BI Monitoring 101

- Not a Power BI Intro!
- Why you need monitoring?
- Report Demo
- How to do it? – Fast Paced ☺

Focus on the possibilities not the how…

dev>scope

Can you answer these questions?

- Who are most active users?
- How many **distinct users**? Per Month? Per day? Per hour?
- Which Reports/Workspaces/Datasets/Apps are mostly used?
- Are Personal Workspaces being used? How is content being shared?
- Do your users access from Browser/Mobile/Excel? Which browser?
- Top used DataSource's? FileSystem/OneDrive? SQL? Sharepoint? DataLake?
- DataSets Refreshing trends/average/errors?
- DataSets/Reports not used in more than 1 year?
- How many users with license don't use Power BI in more than 3 months?

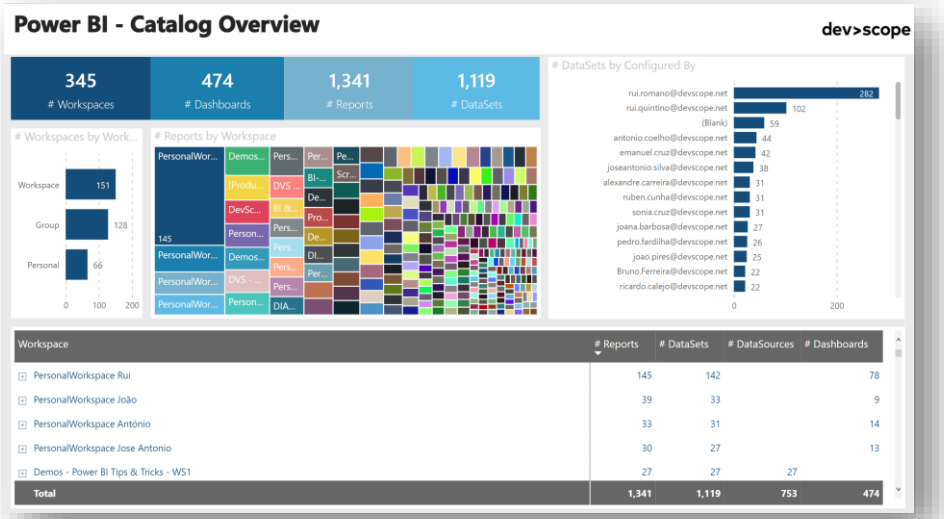
NO?

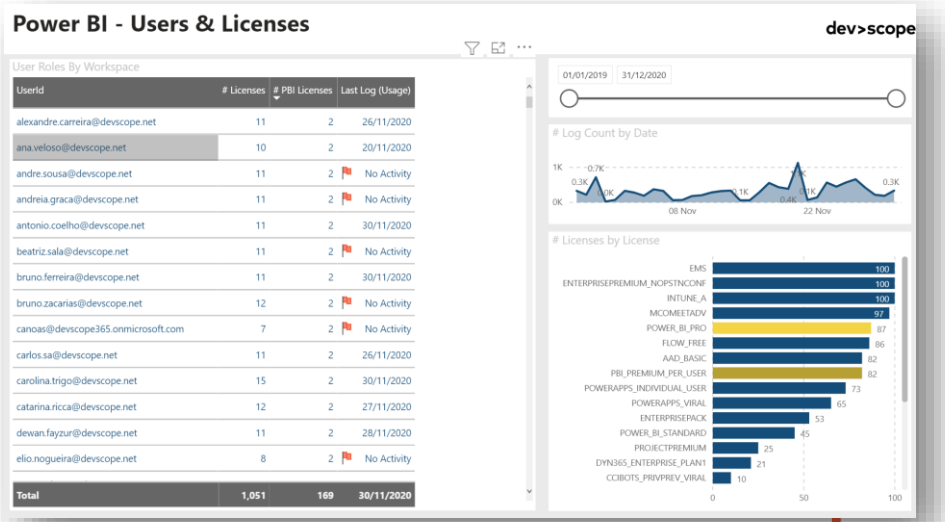
You are “driving blind” and certainly, a **BIG Reality Check** is ahead of you…

<u>Monitoring is one of the main pillars of a good Power BI Governance strategy</u>

This session goal is to motivate you to look at this data, it’s always important to do the “BI of the BI”!

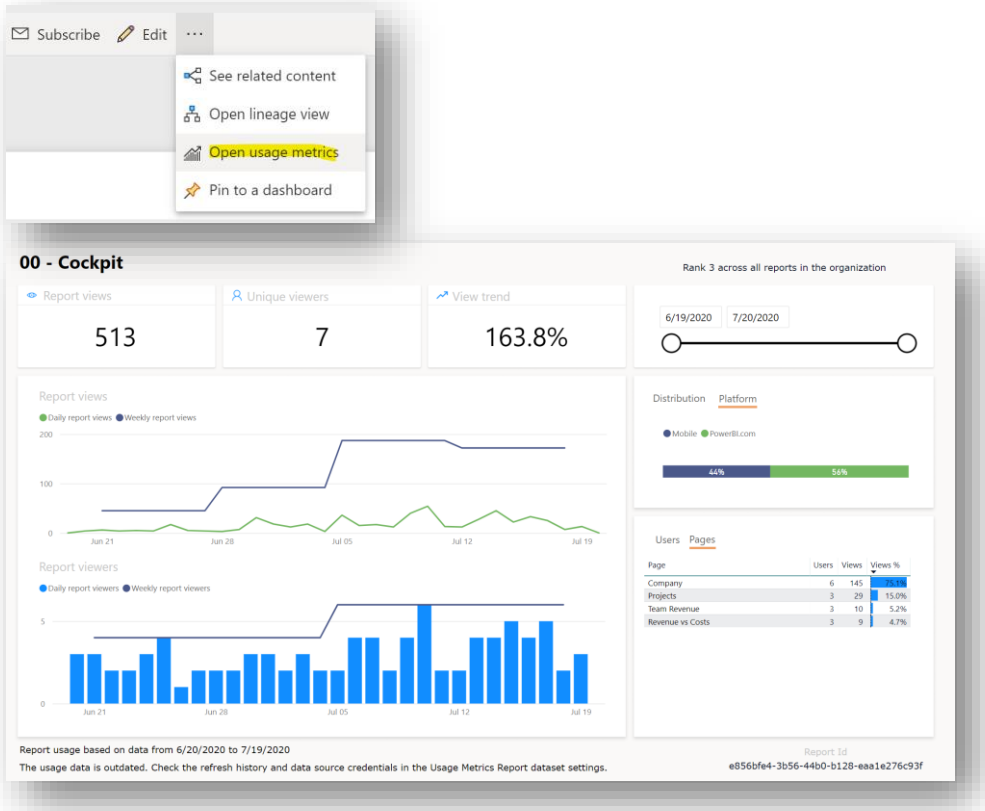
DEMO – Power BI Monitor

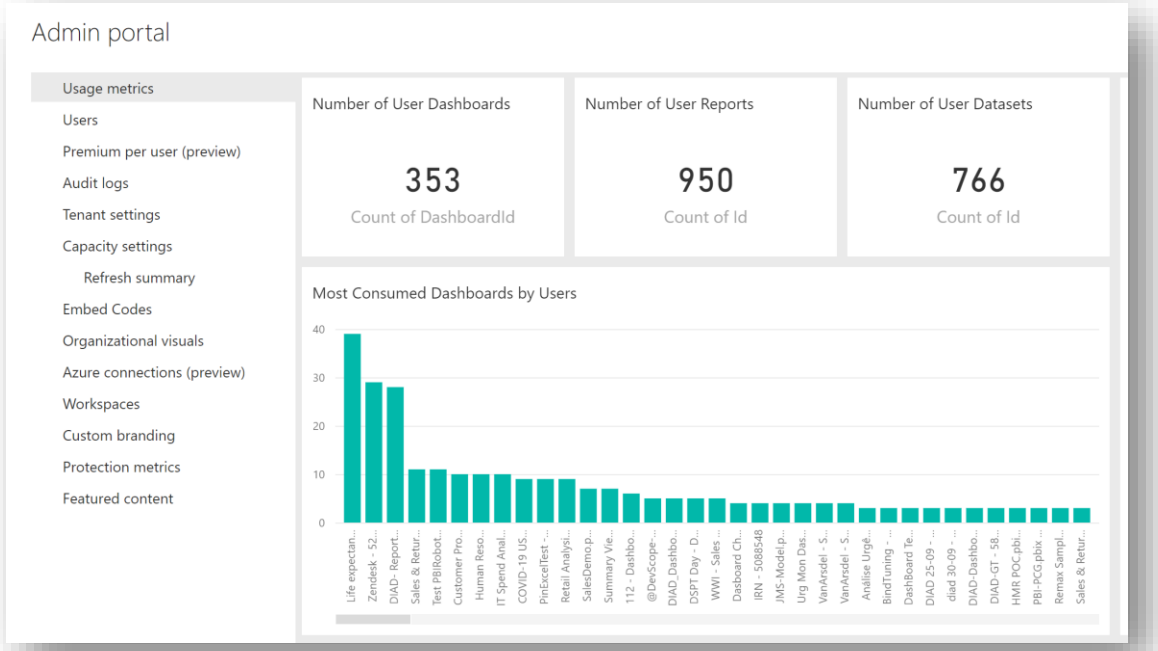




What is available Out of the Box?

Power BI Report Usage	Usage Metrics in Admin Portal
• Workspace level • Page metrics • Client Telemetry data (time to open report) • 7 days	• Tenant Level • Zero Interactivity and Customization • No Refresh Control



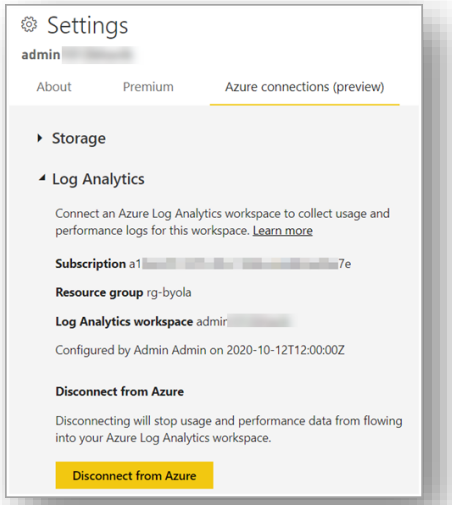


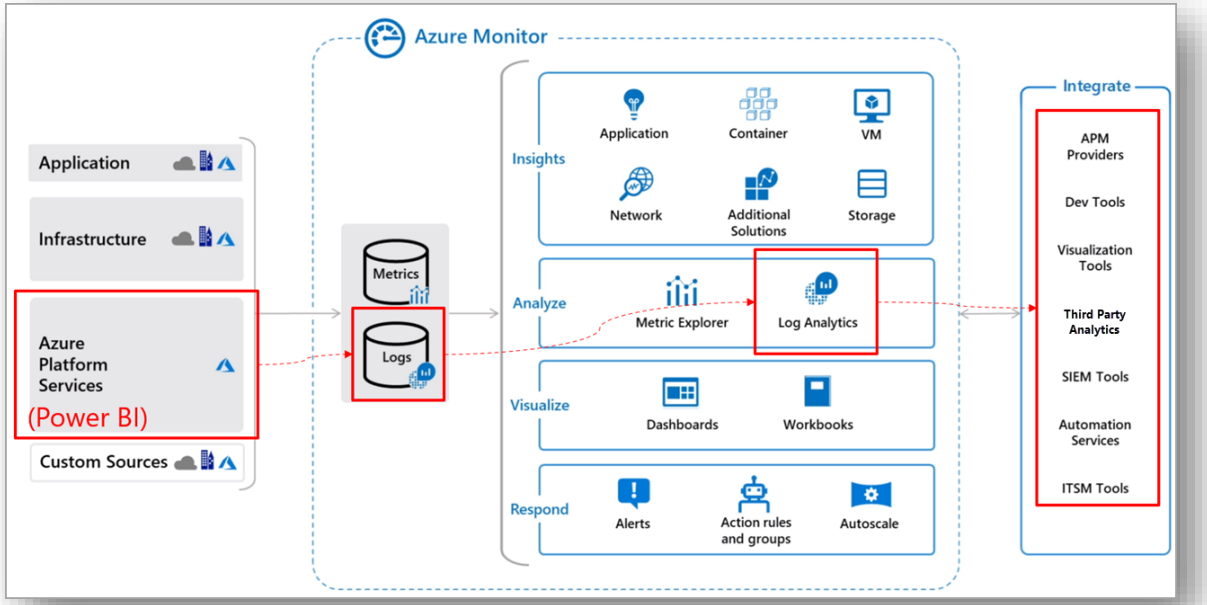
dev>scope

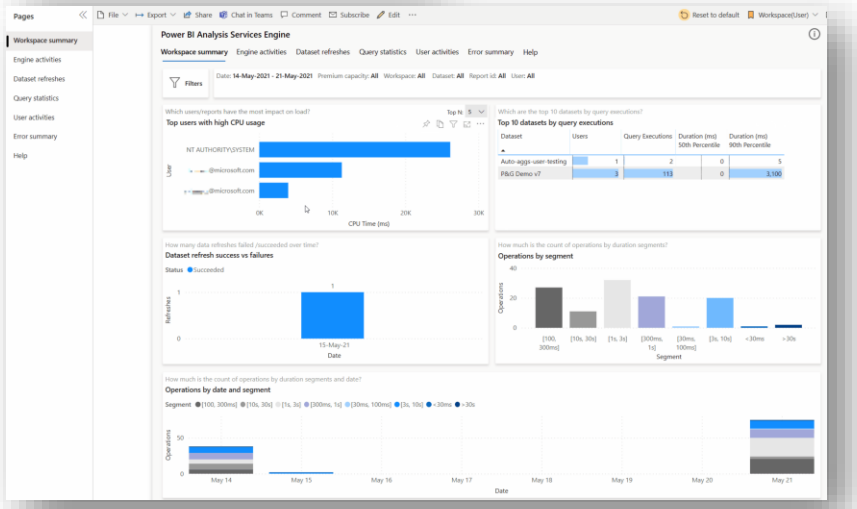
What is available Out of the Box?

Power BI & Log Analytics

- Dataset level – Queries, Refresh Commands,…
- Similar to Azure AS Diagnostics
- Very useful to track performance issues on a large tenant
  - Ex: Correlating a usage spike to the DAX queries
- Pre-Built Power BI Report Template
- **Preview Limitations**
  - Cannot connect multiple workspaces to same Log Analytics







dev>scope

Architecture – Simplest and Easy to Share

Data Sources

Extract

Store

Analyze

PBI REST API
- Metadata / Async
- Activity (**30 days**)

PBI - FetchCatalog.ps1
PBI - FetchDataSetRefresh.ps1

PBI - FetchActivity.ps1

Graph API
- Users
- Licenses

PBI - FetchGraph.ps1

Local Folders
- Metadata
- Activity
- Graph

Power BI DataSet & Report

Config.json

TODO
- Encript file
- Store secrets in a secure location

dev>scope

Architecture – Recommended (one of many possibilities)

Data Sources

PBI REST API
- Metadata / Async
- Activity

Graph API
- Users
- Licenses

Extract

Azure Function
- Fetch Activity
- Fetch Metadata
- Fetch Users & Licenses

OR

Azure Data Factory
- See Just Blindbæk blog

Store

Azure Storage
- \Metadata
- \Activity
- \Graph

Analyze

Power BI Report

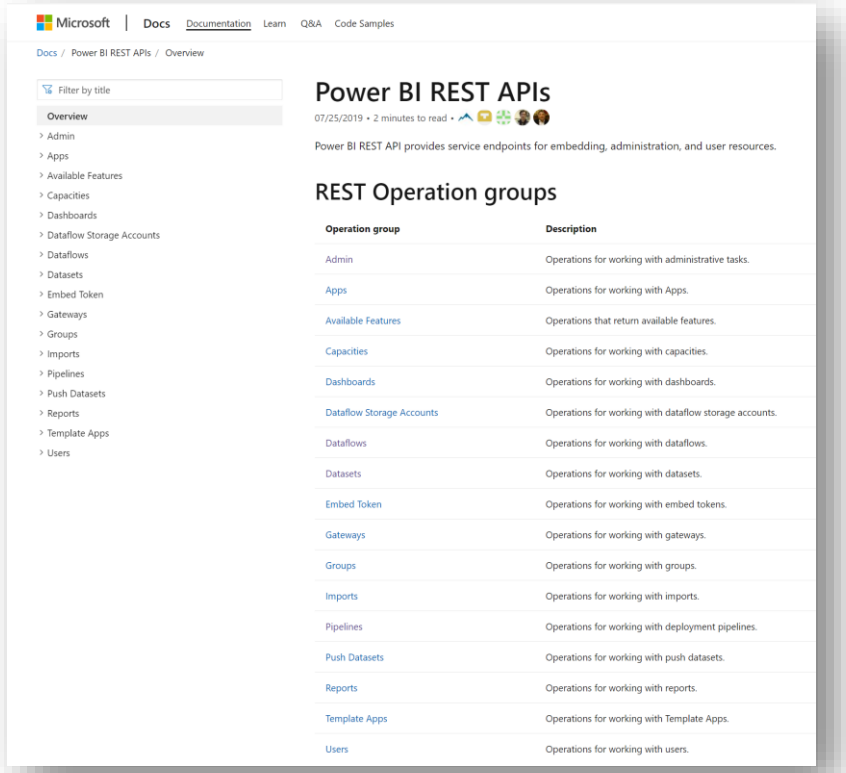
Key Vault

App Insights

dev>scope

API's Overview

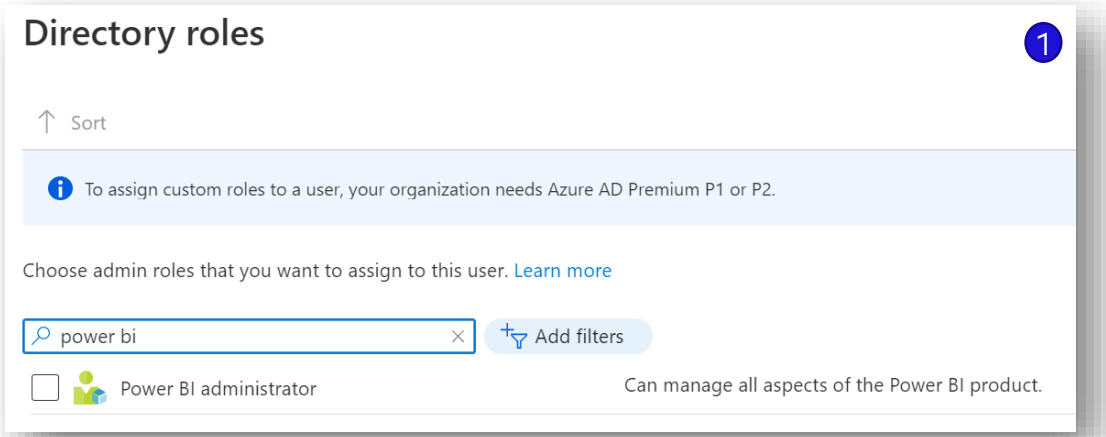
Scope	Resource	API
Power BI Metadata	Workspaces DataSets Reports Dashboards	Power BI Admin Scan APIs
	Users	Power BI Admin API – GetGroupsAsAdmin + Expand Users
	RefreshHistory	Power BI Admin API – GetGroupsAsAdmin + Expand DataSets Power BI API - Get Refresh History In Group
Activity	Power BI Activity Logs	Power BI Admin API - ActivityEvents
Users & Licenses	Users & Licenses	Microsoft Graph API – Users
	Licenses Details	Microsoft Graph API – SubscribedSKUs

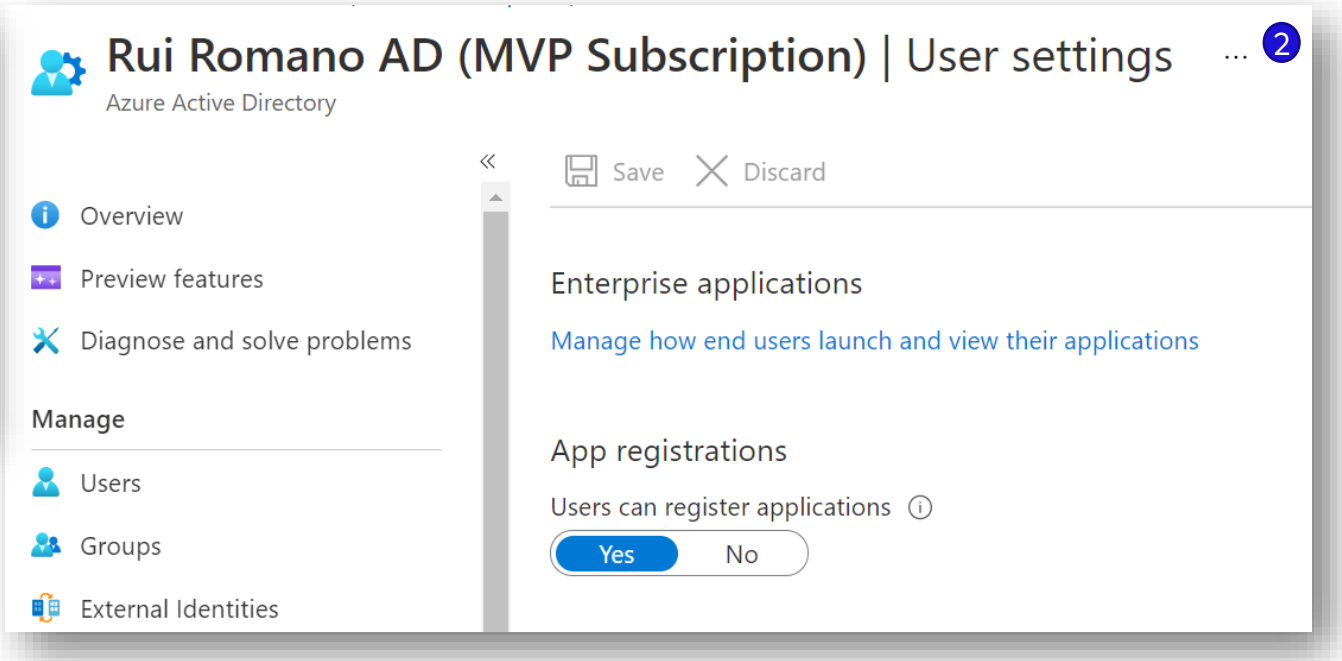


dev>scope

Requirements

- Be a Power BI Administrator (or friend to one ☺)
- Permissions to create an Azure AD Application / Service Principal
- Permission to create an Azure AD Security Group

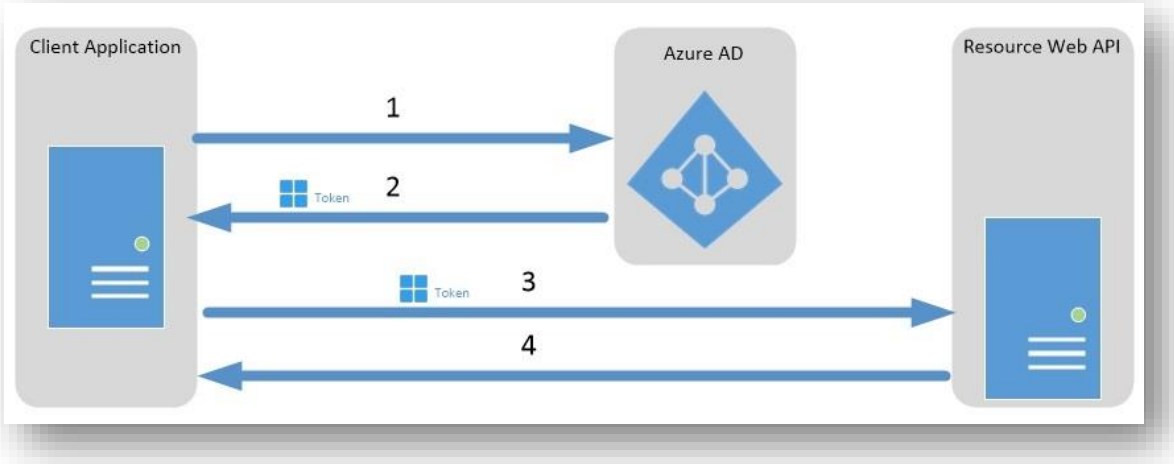




dev>scope

API Authentication

- OAuth 2.0
- Possible Authentication Flows:

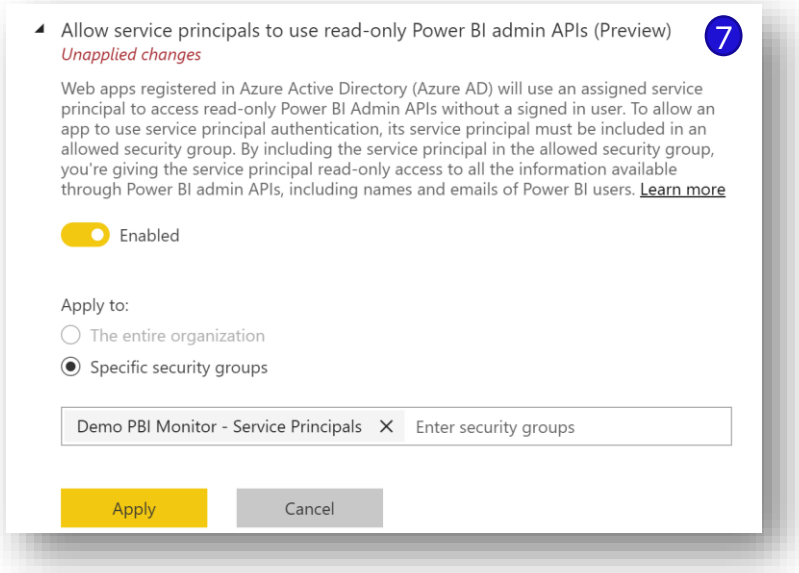


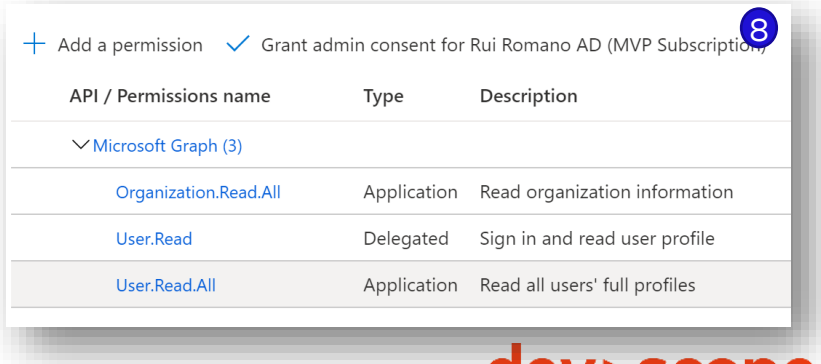
Authentication Flow	Requirements
Client Credentials Flow	Azure AD Service Principal Permissions needed: • Power BI Admin Authorization (read only) • Admin Permissions to read Graph API • User.ReadAll | Organization.ReadAll
Device Code Flow	Power BI Administrator Account
Username & Password Not Recommended	Power BI Administrator Account

Service Principal Step by Step

1. Go to Azure AD Active Directory
2. Go to App Registrations and create a new App (leave defaults)
3. Generate a new App Secret
4. Save the App Id, App Secret & Tenant Id
5. Create an Azure AD Security Group
6. Add the Service Principal to the Security Group as a member
7. Authorize the Security Group in Power BI Admin Portal
8. **Optional** - Authorize the Service Principal to Access Graph API

Note: You don't need to add any Power BI API Permissions





dev>scope

milestones

- ✓ IT/BI Admin Support
- ❑ Extract Data
- ❑ Data Store
- ❑ Power BI DataSet
- ❑ Power BI Report

dev>scope

Script PBI - FetchActivity.ps1

- Incrementally fetch Power BI Activity
  Uses PowerBIPS module
- Uses the Power BI Admin API:
  Get Activity Events
- Can only go back 30 days

```powershell
if ($config.Activity.LastRun)
{
    $pivotDate = [datetime]::Parse($config.Activity.LastRun)
}
else
{
    $config | Add-Member -NotePropertyName "Activity" -NotePropertyValue @{"LastRun" = $null }

    $pivotDate = [datetime]::UtcNow.Date.AddDays(-30)
}

# Gets audit data daily

while($pivotDate -le [datetime]::UtcNow)
{
    Write-Host "Getting audit data for: '$($pivotDate.ToString("yyyyMMdd"))'"

    $odataParams = "startDateTime='$($pivotDate.ToString("s"))'&endDateTime='$($pivotDate.AddHo

    $audits = @(Invoke-PBIRequest -authToken $authToken -resource "activityevents" -admin -odat

    if ($audits.Count -gt 0)
    {
        Write-Host "'$($audits.Count)' audits"

        $outputFilePath = "$outputPath\$($pivotDate.ToString("yyyyMMdd")).json"

        ConvertTo-Json $audits | Out-File $outputFilePath -force
    }
    else
    {
        Write-Warning "No audit logs for date: '$($pivotDate.ToString("yyyyMMdd"))'"
    }

    $config.Activity.LastRun = [datetime]::UtcNow.Date.ToString("o")

    $pivotDate = $pivotDate.AddDays(1)

    # Save config

    ConvertTo-Json $config | Out-File $confi
}
```

```
PS C:\@Repos\Github\pbimonitor> C:\@R
Getting OAuth Token
Getting audit data for: '20210218'
'1322' audits
Getting audit data for: '20210219'
'223' audits

PS C:\@Repos\Github\pbimonitor>
```

```json
{
    "ServicePrincipal": {
                            "AppId": "[redacted]",
                            "AppSecret": "[redacted]",
                            "TenantId": "[redacted]"
                        },
    "Catalog": {
                    "LastRun": "2021-02-18T00:00:00.0000000Z"
                },
    "Activity": {
                    "LastRun": "2021-02-18T00:00:00.0000000Z"
                }
}
```

ub › pbimonitor › Data › Usage

Name

- 20201124.json
- 20201126.json
- 20201127.json
- 20201130.json
- 20201201.json
- 20201207.json
- 20201209.json
- 20201211.json

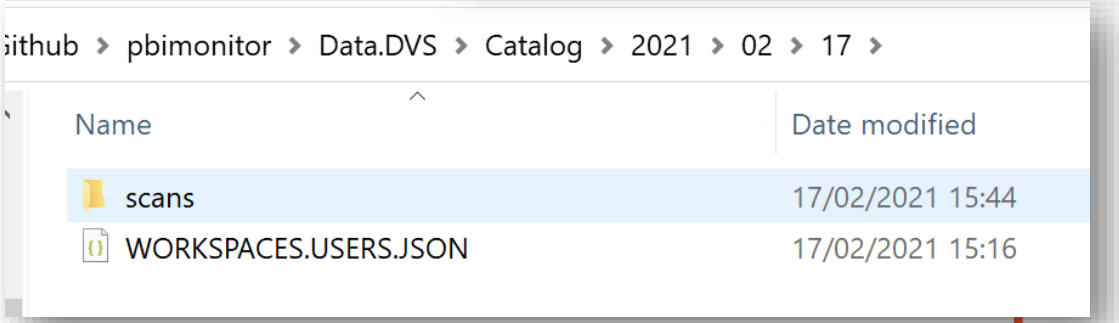
dev>scope

Script – PBI - FetchCatalog.ps1

- Snapshot the entire tenant metatada:
  - Workspaces (personal included)
  - DataSets
  - DataSources
  - Reports
  - Dashboards
  - Users
- Uses the new Async API
  Faster & Incremental
  Lineage + Schema (preview)
- Missing on Async API:
  Workspace Users + Roles
  - Admin Get Groups + $expand=users
  - Admin Apps

```
do
{
    try
    {
        $workspacesBatch = @($workspacesModified | Select -First $batchCount -Skip $skip)

        if ($workspacesBatch)
        {
            Write-Host "Requesting workspace scan: $($skip + $batchCount) / $($workspacesModified.

            $bodyStr = @{"workspaces" = $workspacesBatch.Id } | ConvertTo-Json

            $workspacesScanRequests += Invoke-PBIRequest -authToken $authToken -resource "workspac

            $skip += $batchCount
        }
    }
    catch [System.Net.WebException]
    {
        $ex = $_.Exception

        $statusCode = $ex.Response.StatusCode

        if ($statusCode -eq 429)
        {
            $waitSeconds = [int]::Parse($ex.Response.Headers["Retry-After"])

            Write-Host "429 Throthling Error - Need to wait $waitSeconds seconds..."

            Start-Sleep -Seconds ($waitSeconds + 5)

            $authToken = Get-PBIAuthToken -clientId $config.ServicePrincipal.AppId -clientSecret $
        }
    }
}
while($workspacesBatch.Count -ne 0 -a
```

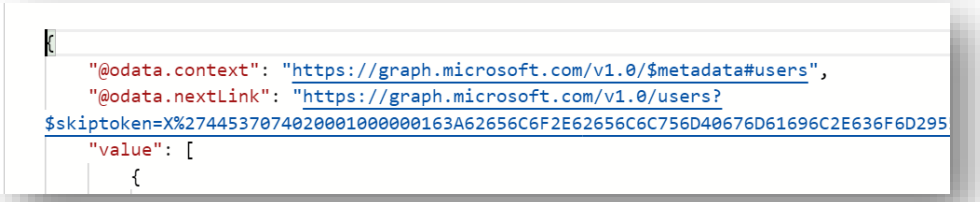
```
PS C:\@Repos\Github\pbimonitor> C:\@Repos\Github\pbimonit
Fetching 5000 /admin/workspaces
Getting workspaces to scan
Since: 2021-02-18T00:00:00.0000000Z
Modified workspaces: 5
Requesting workspace scan: 100 / 5
Waiting for scan results...
Scan 'bbf36328-5210-4006-bd45-830cf22e2793' : 'Succeeded'
Scan Result'bbf36328-5210-4006-bd45-830cf22e2793' : '5'
Ellapsed: 9.0469164s

PS C:\@Repos\Github\pbimonitor>
```

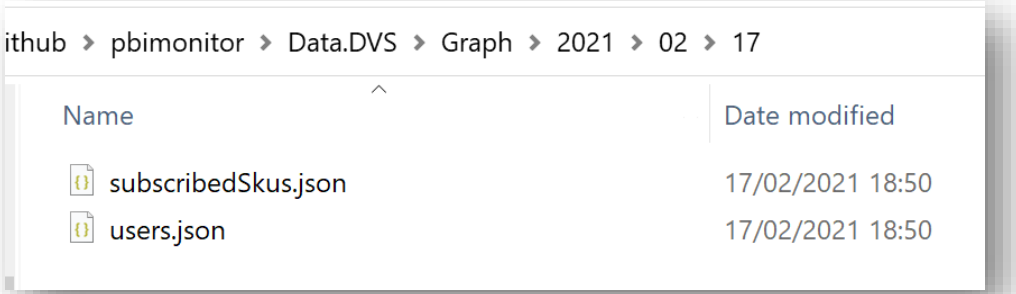


Script – PBI - FetchGraph.ps1

- Snapshot tenant:
  - Users & Assigned Licenses
  - Tenant Subscribed SKUs
- Uses the Microsoft Graph API
  - Users
  - SubscribedSkus
- Paginated API

```powershell
New-Item -ItemType Directory -Path $outputPath -ErrorAction SilentlyContinue | Out-Null
# Get the authentication token
$authToken = Get-AuthToken -resource "https://graph.microsoft.com" -appid $servicePrincipal.AppId -
$graphUrl = "https://graph.microsoft.com/beta"
Write-Host "Getting Users from Graph"
$users = Read-FromGraphAPI -accessToken $authToken -url "$graphUrl/users?`$select=id,mail,companyNa
$filePath = "$outputPath\Graph.Users.json"
$users | ConvertTo-Json -Compress -Depth 5 | Out-File $filePath -Force
Write-Host "Getting SKUs from Graph"
$skus = Read-FromGraphAPI -accessToken $authToken -url "$graphUrl/subscribedSkus?`$select=id,capabi
$filePath = "$outputPath\Graph.SKUs.json"
$skus | ConvertTo-Json -Compress -Depth 5 | Out-File $filePath -Force
```





dev>scope

Script – PBI - FetchDataSetRefresh.ps1

- There is no Admin API to get DataSetRefresh History
- Ensure the service principal is a member of every workspace to monitor, manually or script
- Loop all datasets and call "Refreshes" Api that get the latest refreshes for the dataset

```
foreach($workspace in $workspaces)
{
    $item++

    Write-Host "Processing workspace: '$($workspace.Name)' $item/$total"

    Write-Host "Datasets: $($workspace.datasets.Count)"

    $refreshableDatasets = @($workspace.datasets |? { $_.isRefreshable -eq $true -and $_.addRowsAPIEnabled -eq $false})

    Write-Host "Refreshable Datasets: $($refreshableDatasets.Count)"

    foreach($dataset in $refreshableDatasets)
    {
        try
        {
            Write-Host "Processing dataset: '$($dataset.name)'"

            Write-Host "Getting refresh history"

            $dsRefreshHistory = Invoke-PBIRequest -authToken $authToken -resource "datasets/$($dataset.id)/refreshes" -groupId $workspace.id

            if ($dsRefreshHistory)
            {
                $dsRefreshHistory = $dsRefreshHistory | Select *, @{Name="dataSetId"; Expression={ $dataset.id }}, @{Name="dataSet"; Expressio
                    , @{Name="group"; Expression={ $workspace.name }}, @{Name="configuredBy"; Expression={ $dataset.configuredBy }}

                $dsRefreshHistoryGlobal += $dsRefreshHistory
            }
        }
        catch
        {
            $ex = $_.Exception

            Write-Error -message "Error processing dataset: '$($ex.Message)'" -ErrorAction Continue
```

```
{
  "value": [
    {
      "refreshType": "ViaApi",
      "startTime": "2017-06-13T09:25:43.153Z",
      "endTime": "2017-06-13T09:31:43.153Z",
      "serviceExceptionJson": "{\"errorCode\":\"ModelRefreshFailed_CredentialsNotSpecified\"}",
      "status": "Failed",
      "requestId": "11bf290a-346b-48b7-8973-c5df149337ff"
    }
  ]
}
```

dev>scope

API Throttling

- Power BI & Graph Api's have throttling enabled
- Handle the exception "429 Too Many Requests"

Admin - Get Activity Events

Service: Power BI REST APIs

API Version: v1.0

Returns a list of audit activity events for a tenant.

Note: Activity logging isn't supported for Microsoft Cloud Deutschland. The user must have administrator rights (such as Office 365 Global Administrator or Power BI Service Administrator) to call this API or authenticate via service principal.

This API allows 200 requests per hour at maximum.

```
HTTP/1.1 429 Too Many Requests
Content-Type: text/html
Retry-After: 3600
```

```
}
catch [System.Net.WebException]
{
    $ex = $_.Exception

    $statusCode = $ex.Response.StatusCode

    if ($statusCode -eq 429)
    {
        $waitSeconds = [int]::Parse($ex.Response.Headers["Retry-After"])

        Write-Host "429 Throthling Error - Need to wait $waitSeconds seconds..."

        Start-Sleep -Seconds ($waitSeconds + 5)

        $authToken = Get-PBIAuthToken -clientId $config.ServicePrincipal.AppId -clie
    }
}
```

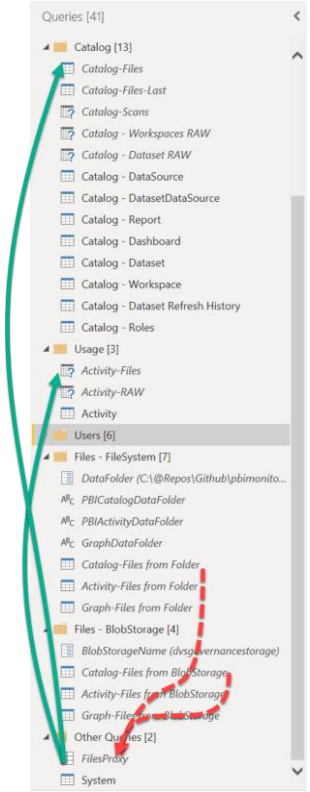
milestones

- ✓ IT Admin Support
- ✓ Extract Data
- ✓ Data Store
- ❑ Power BI DataSet
- ❑ Power BI Report

dev>scope

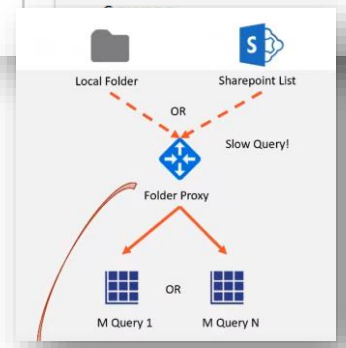
Power BI - PowerQuery

- Timezone offset, all dates are UTC
- Proxy Query to reference all files, easy switch data source (ex: folder or data lake)

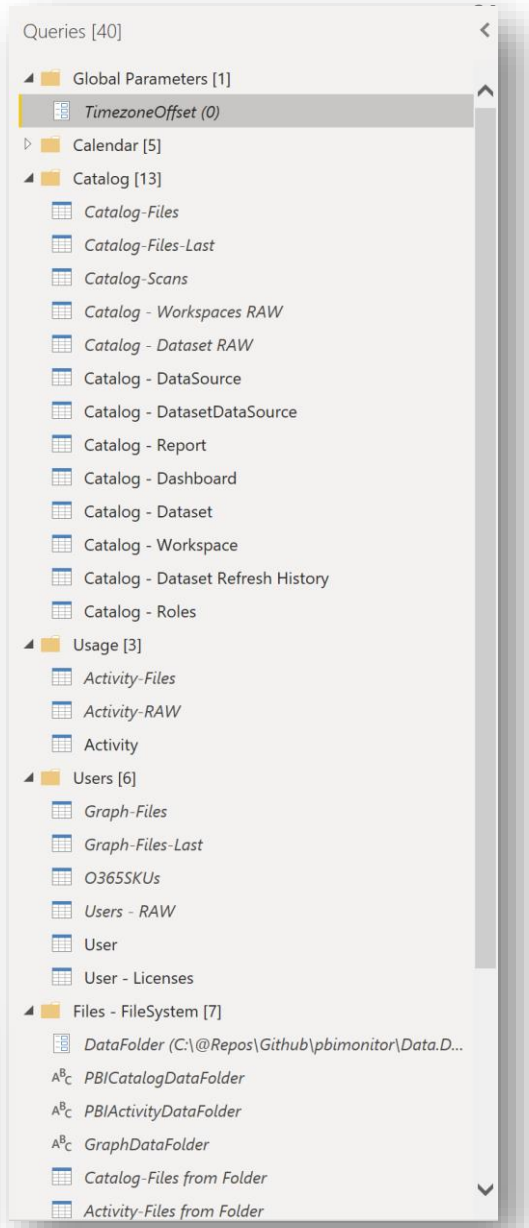


FilesProxy

```
let
    Source = [ActivityFiles = #"Activity-Files from Folder"
    , CatalogFiles = #"Catalog-Files from Folder"
    , GraphFiles = #"Graph-Files from Folder"
    ]
    //Source = [ActivityFiles = #"Activity-Files from BlobStorage"
    //, CatalogFiles = #"Catalog-Files from BlobStorage"
    //, GraphFiles = #"Graph-Files from BlobStorage"
    //]
in
```



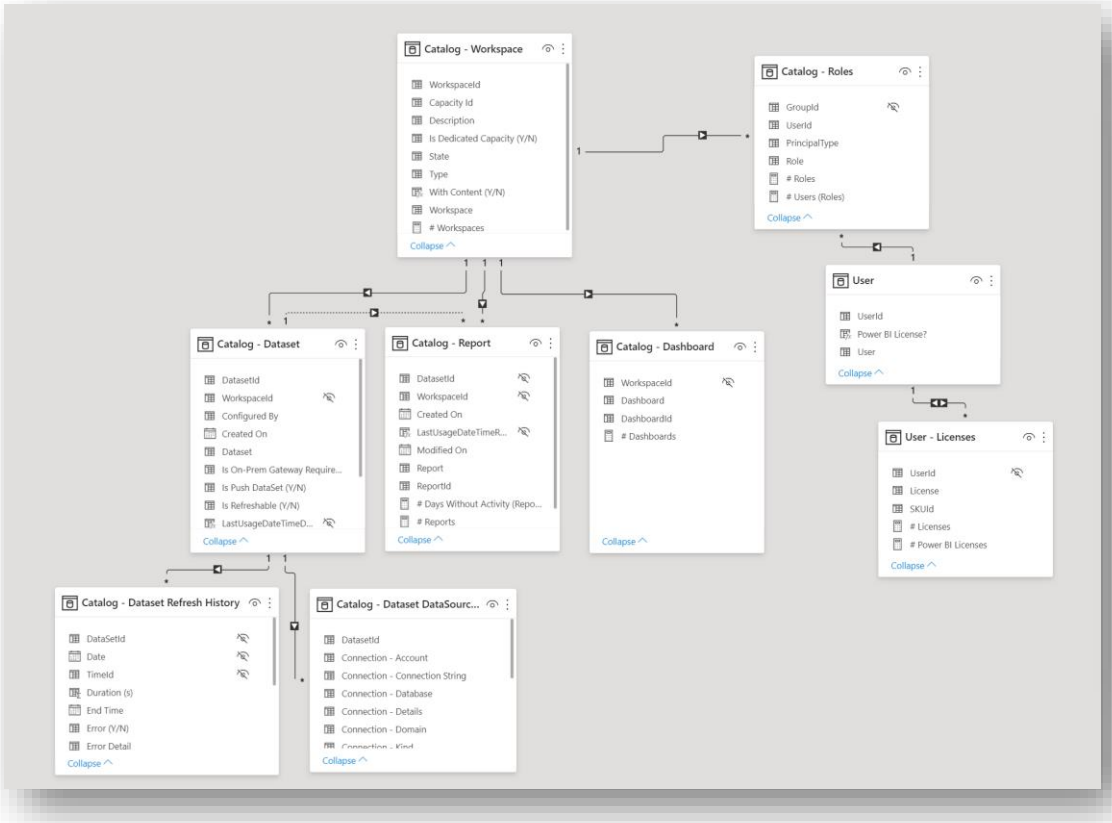
	Content	Filename	Date	FileType
1	Binary	0224742b-abc1-4ddb-8e35-7cda46b31b14.json	10/02/2021	scan
2	Binary	03b400bb-0473-4e16-a5d5-62052443ca05.json	10/02/2021	scan
3	Binary	05f460dd-3944-4fbd-a607-db5c6b8a6a1d.json	10/02/2021	scan
4	Binary	09c9baf6-6df7-4bf5-b7a3-77d277527c49.json	10/02/2021	scan
5	Binary	0bef91e1-9d41-4cd3-9a43-ccc19857b41c.json	10/02/2021	scan
6	Binary	0c0fed54-fb15-4fcd-bca8-8bb062fb5259.json	10/02/2021	scan
7	Binary	0ef464ae-b606-4095-87d6-228553630dda.json	10/02/2021	scan
8	Binary	156a0679-a542-45c7-b1df-bb89181d1d86.json	10/02/2021	scan
9	Binary	18200c8c-7b91-4945-b483-e4190293ea2b.json	10/02/2021	scan
10	Binary	1a75f1d8-57f7-4d87-94db-ad656166b104.json	10/02/2021	scan
11	Binary	1b3eecb5-efda-4b37-965f-1e7fb7f50ac4.json	10/02/2021	scan

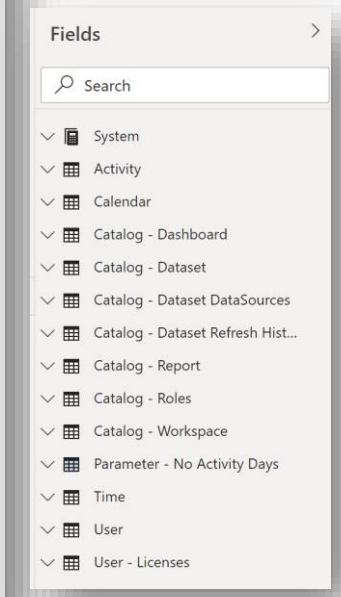


dev>scope

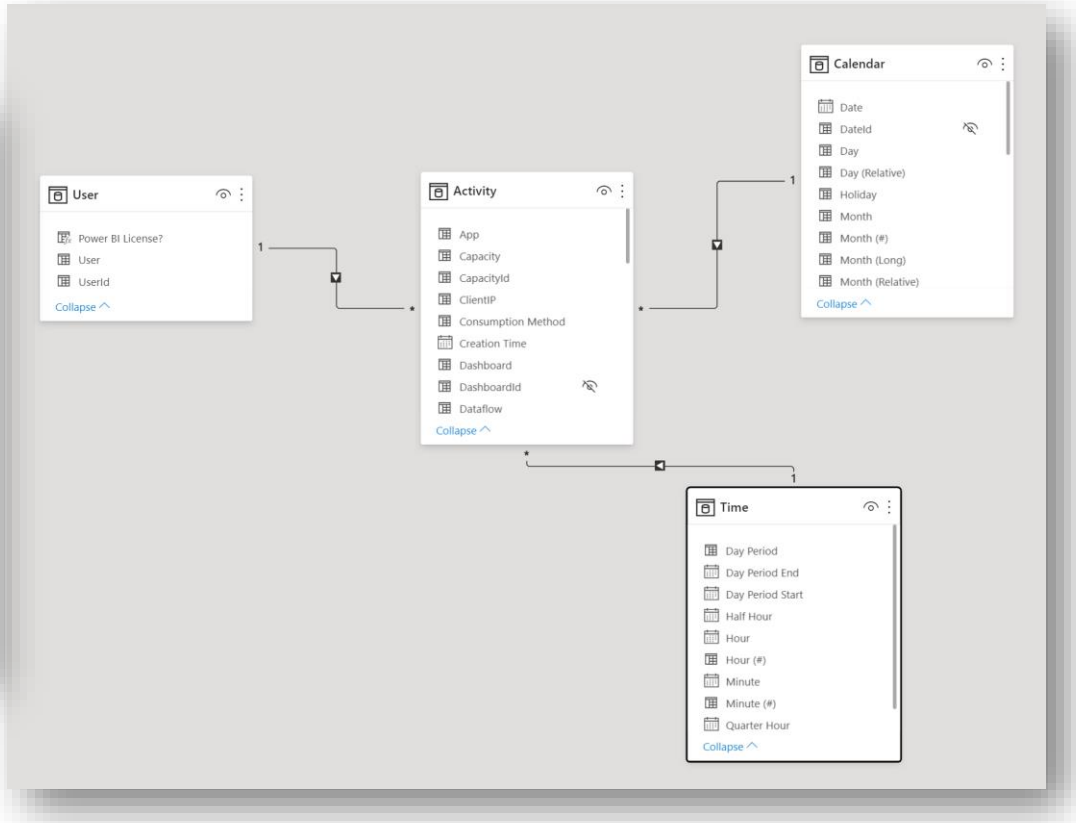
Power BI – DataSet, 2 Models in 1

Catalog





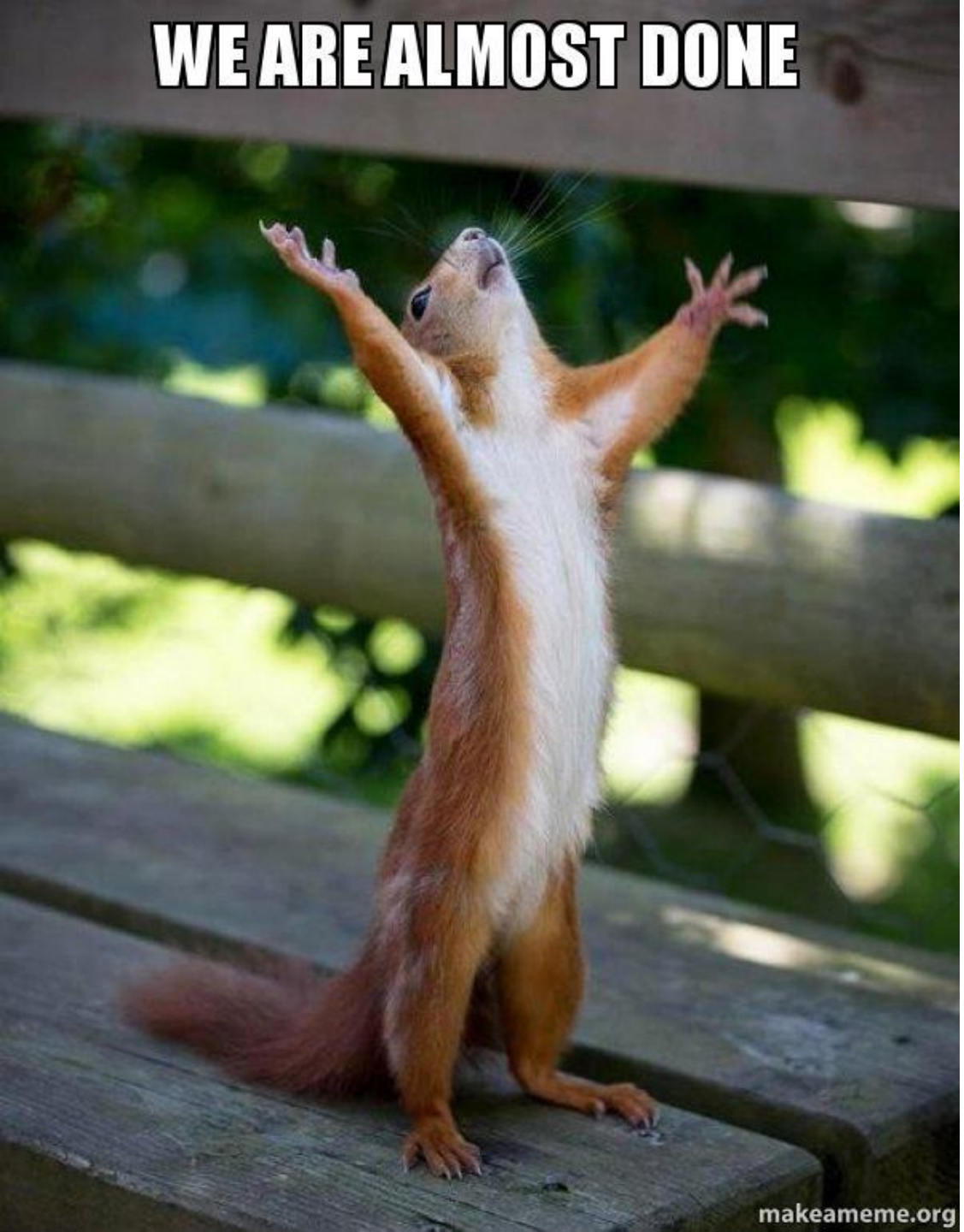
Activity



dev>scope

Power BI – DataSet Main Measures

Catalog	Activity
# Reports # DataSets # Reports … **# Days without Activity** # Licenses # Users	# Logs **# Logs from Excel** # Report Views **# Reports Created** **# Distinct Users by Day** **# Data Views** Activity Grouping => DataViews, Authoring, Admin



milestones

- ✓ IT Admin Support
- ✓ Extract Data
- ✓ Data Store
- ✓ Power BI DataSet
- ❑ Power BI Report

dev>scope

Power BI - Report

- Base theme, easy company branding
- Drillthrough for detail
- Look at No Activity
- Quick search for a report/dataset by id / name
- Make use of **advanced AI features**:
  Explain Increase/Decrease
  Anomaly Detection
  Forecast

dev>scope

milestones

- ✓IT Admin Support
- ✓Extract Data
- ✓Data Store
- ✓Power BI DataSet
- ✓Power BI Report

Thanks!

Rui Romano

- rui.romano@devscope.net
- linkedin.com/in/ruiromano
- @ruiromano
- https://ruiromanoblog.wordpress.com

dev>scope

MVP Microsoft® Most Valuable Professional