Insight Quest

Brett Powell's Data & Analytics Blog

# POWER BI WORKSPACE SCRIPTS

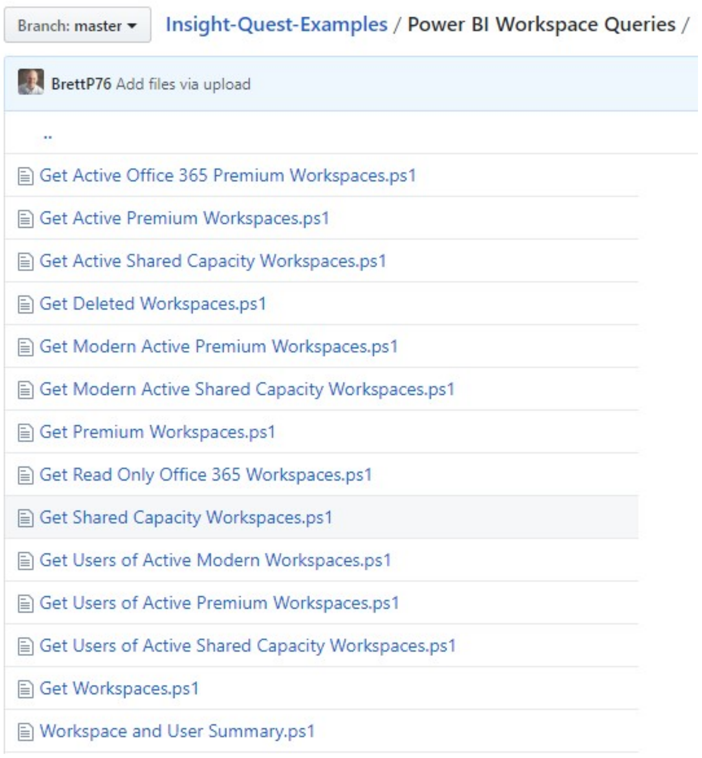
December 24, 2018 Power BI, PowerShell, Visual Studio 4 comments

If you're a Power BI administrator you're probably already familiar with the Power BI Management module of PowerShell cmdlets. Additionally, as an alternative to PowerShell, you may have seen the recent announcement for the new Workspaces page in the Power BI Admin Portal.

While more features and improvements to the Workspaces page will be implemented over time, it may still be useful to have several pre-built PowerShell scripts to access and analyze workspaces in your Power BI tenant. As just one simple example, you may want to retrieve all the users of workspaces assigned to premium capacity.

## Workspace Scripts

I've uploaded the following 14 PowerShell scripts to GitHub:


Branch: master
Insight-Quest-Examples / Power BI Workspace Queries /
BrettP76 Add files via upload
..
Get Active Office 365 Premium Workspaces.ps1
Get Active Premium Workspaces.ps1
Get Active Shared Capacity Workspaces.ps1
Get Deleted Workspaces.ps1
Get Modern Active Premium Workspaces.ps1
Get Modern Active Shared Capacity Workspaces.ps1
Get Premium Workspaces.ps1
Get Read Only Office 365 Workspaces.ps1
Get Shared Capacity Workspaces.ps1
Get Users of Active Modern Workspaces.ps1
Get Users of Active Premium Workspaces.ps1
Get Users of Active Shared Capacity Workspaces.ps1
Get Workspaces.ps1
Workspace and User Summary.ps1

**Power BI Workspace Scripts**

With the exception of the Workspace and User Summary script, each PowerShell file contains the following three parts:

1. Retrieve a subset of workspaces or users of those workspaces
2. Compute the count of the subset of workspaces or users of those workspaces
3. Export a subset of workspaces or users of those workspaces to a CSV file

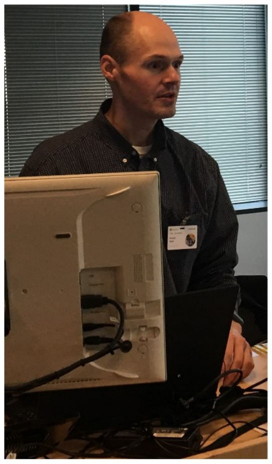
Depending on your scenario, you may only need one or two of the three parts. To export out the details of the PowerShell scripts to a CSV file for further analysis in a tool like Excel or Power BI Desktop you can just revise the export file variable (part 3) to a directory on your PC or network. Of course you can tweak or extend the scripts to your preferences such as choosing to include personal workspaces (excluded

this blog and receive notifications of new posts by email.

Enter your ema

Follow

## INSIGHT QUEST



## BLOG POST HISTORY

December 2018

| M | T | W | T | F | S | S |
|---|---|---|---|---|---|---|
| | | | | | 1 | 2 |
| 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 17 | 18 | 19 | 20 | 21 | 22 | 23 |

in the scripts).

## Workspace Dimensions

Essentially the scripts just filter down the workspaces of the organization based on the following four dimensions:

- IsOnDedicatedCapacity (true or false)
- State (Active, Deleted, Removing)
- Type (Workspace, Group, PersonalGroup)
- IsReadOnly (true or false)

The blog post on the new Workspaces page describes the available values for these columns. Probably the most important columns are IsOnDedicatedCapacity (Is the workspace in Premium or Shared Capacity?) and Type (Is the workspace a modern workspace or the legacy Office 365 group-based workspace). There are of course serious performance and functionality implications for the content and consumption of that content based on these values.

### Script Example: Active Premium Workspaces

```
Connect-PowerBIServiceAccount

#1. Active Premium workspaces

Get-PowerBIWorkspace -Scope Organization | Where {($_.

#2. Count of active premium workspaces

$ActivePremiumWorkspaces = Get-PowerBIWorkspace -Sc
```

| M | T | W | T | F | S | S |
|---|---|---|---|---|---|---|
| 24 | 25 | 26 | 27 | 28 | 29 | 30 |
| 31 | | | | | | |

« Nov Feb »

### FIVE POPULAR POSTS

Conditional Formatting for Card Visuals

Refresh Power BI Datasets with PowerShell

Retrieve Power BI Pro Licensed Users

Power BI Project Example

Data Profiling with Power Query

```
$ActivePremiumWorkspaces.Count

#3. Export of active premium workspaces to CSV file

$ExportFile = "C:\Users\Brett Powell\Desktop\ActivePrem

$ActivePremiumWorkspaces | Export-Csv $ExportFile
```

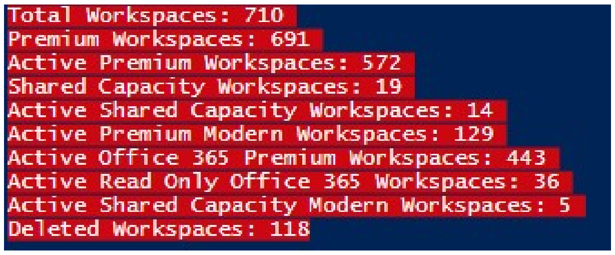
The scripts use the Where-Object to filter the workspaces down based on one or multiple columns such as excluding personal workspaces. The User scripts build on top of the workspace object to access the users for each workspace and then remove duplicates.

Note: You need to be a Power BI service admin to execute the scripts given the organization-wide scope.

## Workspace and User Summary

In some cases you may not need the details of workspaces or users but rather just a high level count of workspaces and users by the various dimensions. For example, you want to know how many of your workspaces are in premium capacity versus shared capacity, how many are active, and the split between modern workspaces and the legacy Office 365 group-based workspaces.

The Workspace and User Summary script targets this use case by passing the counts of variables to a formatted Write-Host cmdlet:



```
Total Workspaces: 710
Premium Workspaces: 691
Active Premium Workspaces: 572
Shared Capacity Workspaces: 19
Active Shared Capacity Workspaces: 14
Active Premium Modern Workspaces: 129
Active Office 365 Premium Workspaces: 443
Active Read Only Office 365 Workspaces: 36
Active Shared Capacity Modern Workspaces: 5
Deleted Workspaces: 118
```

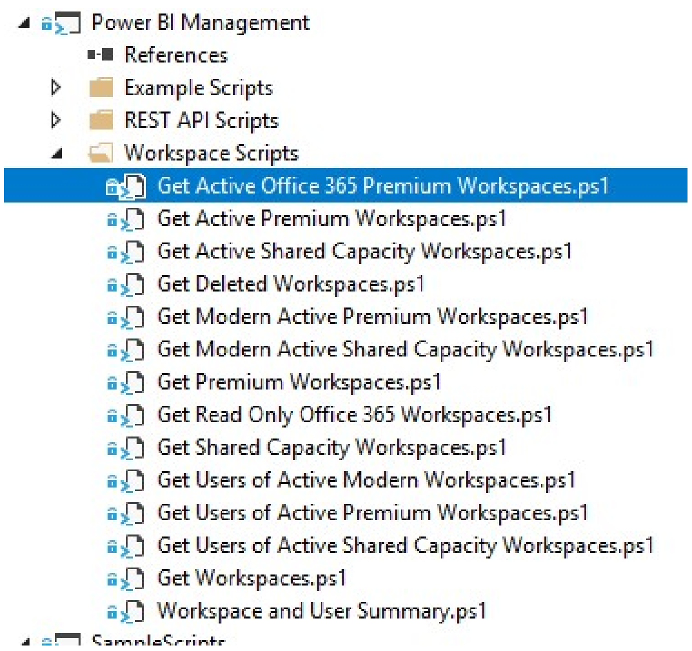
**Workspace and User Summary**

Just like the example above, the counts of users by workspace segment are also output to a formatted Write-Host cmdlet. However, I've only retrieved three different lists of users for now.

*Obviously these scripts are focused exclusively on reading the data, not writing to the Power BI service such as adding users or changing permissions. A future blog post may feature this scenario.

## PowerShell in Visual Studio

As mentioned in a previous blog post, you may consider using PowerShell Script projects in Visual Studio as a means of organizing and managing your BI administration related PowerShell scripts such as processing Tabular models, scaling up/down resources, or these workspace scripts:


Power BI Management
References
Example Scripts
REST API Scripts
Workspace Scripts
Get Active Office 365 Premium Workspaces.ps1
Get Active Premium Workspaces.ps1
Get Active Shared Capacity Workspaces.ps1
Get Deleted Workspaces.ps1
Get Modern Active Premium Workspaces.ps1
Get Modern Active Shared Capacity Workspaces.ps1
Get Premium Workspaces.ps1
Get Read Only Office 365 Workspaces.ps1
Get Shared Capacity Workspaces.ps1
Get Users of Active Modern Workspaces.ps1
Get Users of Active Premium Workspaces.ps1
Get Users of Active Shared Capacity Workspaces.ps1
Get Workspaces.ps1
Workspace and User Summary.ps1
SampleScripts

**PowerShell Script Project in Visual Studio**

I still use the Windows PowerShell ISE significantly but I'm increasingly comfortable with the support available inside Visual Studio (e.g. PowerShell Interactive Window). These tools can be downloaded here.

## Wrapping Up

As in so many cases, the graphical interface tool (Workspaces page in Power BI Admin portal) may be sufficient for many or most scenarios thus rendering custom scripts pointless. However, if (when) it's necessary to do anything not supported out-of-the-box by the GUI, having a solid set of pre-built scripts to work with or extend could be very useful.

If you find this blog post and others helpful feel welcome to subscribe (click Follow) to get notified of new posts. Additionally, as always feel free to share thoughts or ideas for improvements. Happy Holidays.

**Share this:**

Tweet Share Share

**Related**

Parallel Power BI Scripts
PowerShell 7.0 was made generally available last month and one of it's new features is
In "Power BI"

Power BI Artifacts Per Workspace
In response to last week's post regarding the retrieval of the artifacts (reports,
In "Power BI"

Power BI Artifact to Workspace Relationships
In "Power BI"

« Analysis Services Monitoring Report v1.0

Group Email Notification for Dataset Refresh Failure »

## 4 COMMENTS

Pingback: Power BI Workspace Scripts - Learn Power BI

**Roger Halabi** says:

April 25, 2019 at 3:36 pm

Great article. Is there any script for counting reports and dashboards in each workspace for the entire capacity?

★ Like

Reply

**Brett Powell** says:

April 26, 2019 at 3:03 am

Hi Roger,
Please see the new blog post Premium Capacity Contents (http://bit.ly/2UW151i) for a script and details to retrieve this data (count of reports, dashboards, etc for a capacity). You might also read through the Power BI Admin Datasets post for details on a data model that could support this kind of analysis.
Regards,
Brett

★ Like

Reply

**Roger Halabi** says:

April 26, 2019 at 2:10 pm

Thx Brett. I will review the link you provided.

★ Like

## LEAVE A REPLY

Enter your comment here...

Blog at WordPress.com.