How to use PowerBI as a free monitoring tool
(including 5 reports for SQL Server)

About Me

Emanuele Meazzo

Senior Data Management Consultant

Contact Info:

Email: emanuele.meazzo@gmail.com

Blog: tsql.tech

Linkedin: linkedin.com/in/emanuelemeazzo/

GitHub: github.com/EmanueleMeazzo/tsql.tech-Code-snippets

Agenda

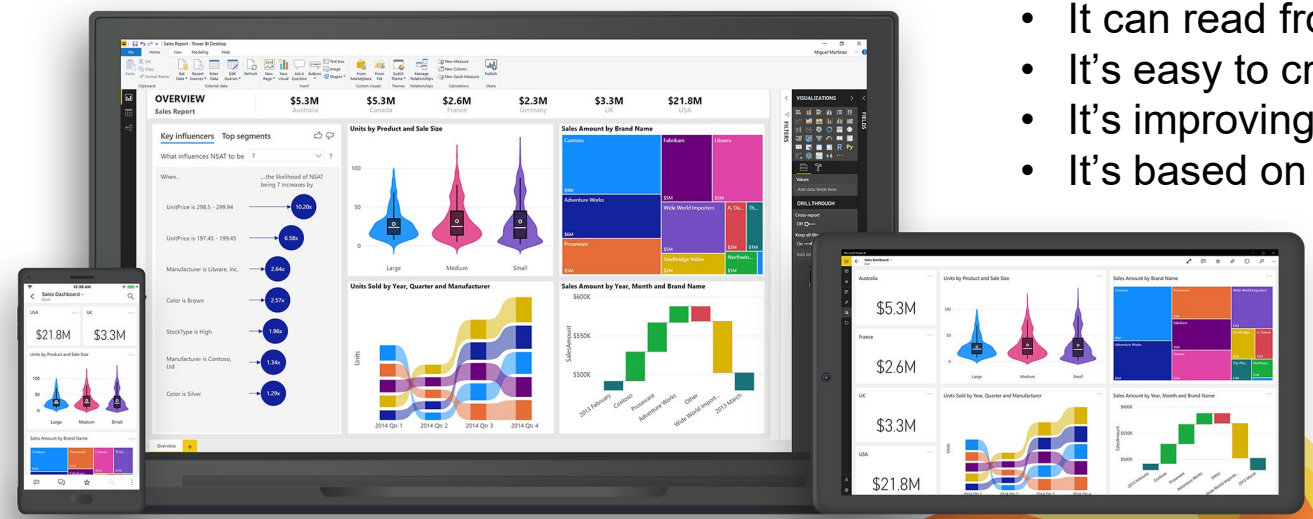
- PowerBI Overview
- PowerBI Components
- How to get data
- Dashboard Examples:
  - FirstRespondersKit Monitoring Dashboard
  - Xevents Dashboard
  - Job Information Dashboard
  - Permission Information Dashboard
  - Streaming DataSet

PowerBI Overview

“Power BI is a business analytics service that delivers insights to enable fast, informed decisions.”

We use it as an (improper) monitoring tool because:

- It’s free*
- It’s on the cloud
- It can read from a lot of data sources
- It’s easy to create graphs and get insights
- It’s improving month by month
- It’s based on SSAS Tabular



PowerBI is **NOT** a monitoring tool

PowerBI is not built to be a ~~**monitoring**~~ tool but a **reporting** tool

A proper monitoring tool is tailored on the specific environment to monitor, it already knows where to find the data.
PowerBI is a generic reporting tool that allows you to analyze the data that you provide, coming from anywhere, in order to monitor and visualize it through a report.

There are some limitations that you will learn to hate:

- Relations between tables don't work the way you expect them to
- Time granularity is way too coarse for certain indicators
- Some trivial operations in SQL are really complex to implement
- Etc..

PowerBI IS useful for monitoring

- It allows to share performance and health information internally or to third parties without granting access to anything but the report
- It allows to give insights to Devs about performance, errors, permissions, etc. without having to teach them how to or having them ask you all the time
- It allows to create a centralized view for all the systems to be monitored, as opposed to checking each environment on its own
- It allows to create reports templates to distribute within the company (or the community!)

PowerBI IS useful for monitoring

Depending on how you're going to use it, you can get away without paying any license:

- PowerBI Desktop (i.e. creating and viewing reports on your PC) is 100% free
- Uploading reports to the PowerBI Cloud in your personal workspace is free
- Embedding & Publishing to web is free
- Manual and Automatic dataset refreshes are free
  - Both from Cloud and On-premise sources via a gateway
  - The gateway can be installed as a service on a dedicated machine for all users or installed on your machine in "personal mode"

PowerBI IS useful for monitoring

So how does Microsoft makes money over PowerBI if everything is basically free?

With the Enterprise features, of course!

Do you need them? It depends. Do you need them to go out and try immediately what we're going to see? Nope.

https://powerbi.microsoft.com/en-us/pricing/

Per-user license type comparison

Here is a list of features supported by per-user license type.

	Free	Pro
Connect to 70+ data sources	✓	✓
Publish to Web	✓	✓
Export to PowerPoint, Excel, CSV	✓	✓
Enterprise distribution		
Apps	✗	✓
Email subscriptions	✗	✓
Embed APIs and controls	✗	✓
Collaboration		
Peer-to-peer sharing	✗	✓
App workspaces	✗	✓
Analyze in Excel, analyze in Power BI Desktop	✗	✓

PowerBI IS useful for monitoring

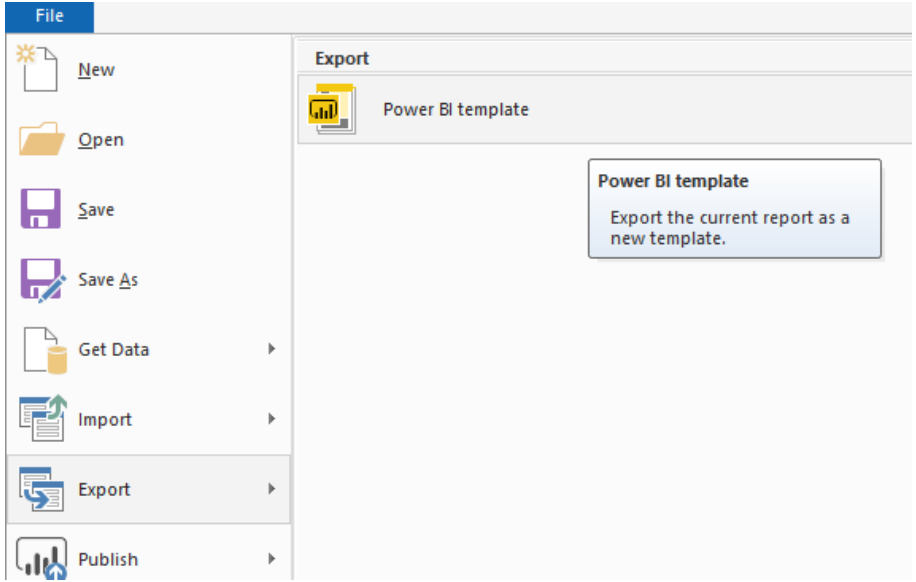
At this point I'm already hearing you saying:
"Ok, but why should I use it as opposed as any other free monitoring tool?"

If you can install any proper monitoring tool for your purpose, great, go for it!
But:

- When you want to quickly visualize some kind of information
- When you cannot modify **anything** in the system
- When you cannot install any third party software
- When for third party software you cannot configure proper security rules
- When you need to monitor something that has no tools available
- When you need to monitor custom processes
- When you need an overview of multiple independent systems

This approach is for you!

PowerBI is useful for monitoring



Report Templates can be created (*.pbit)

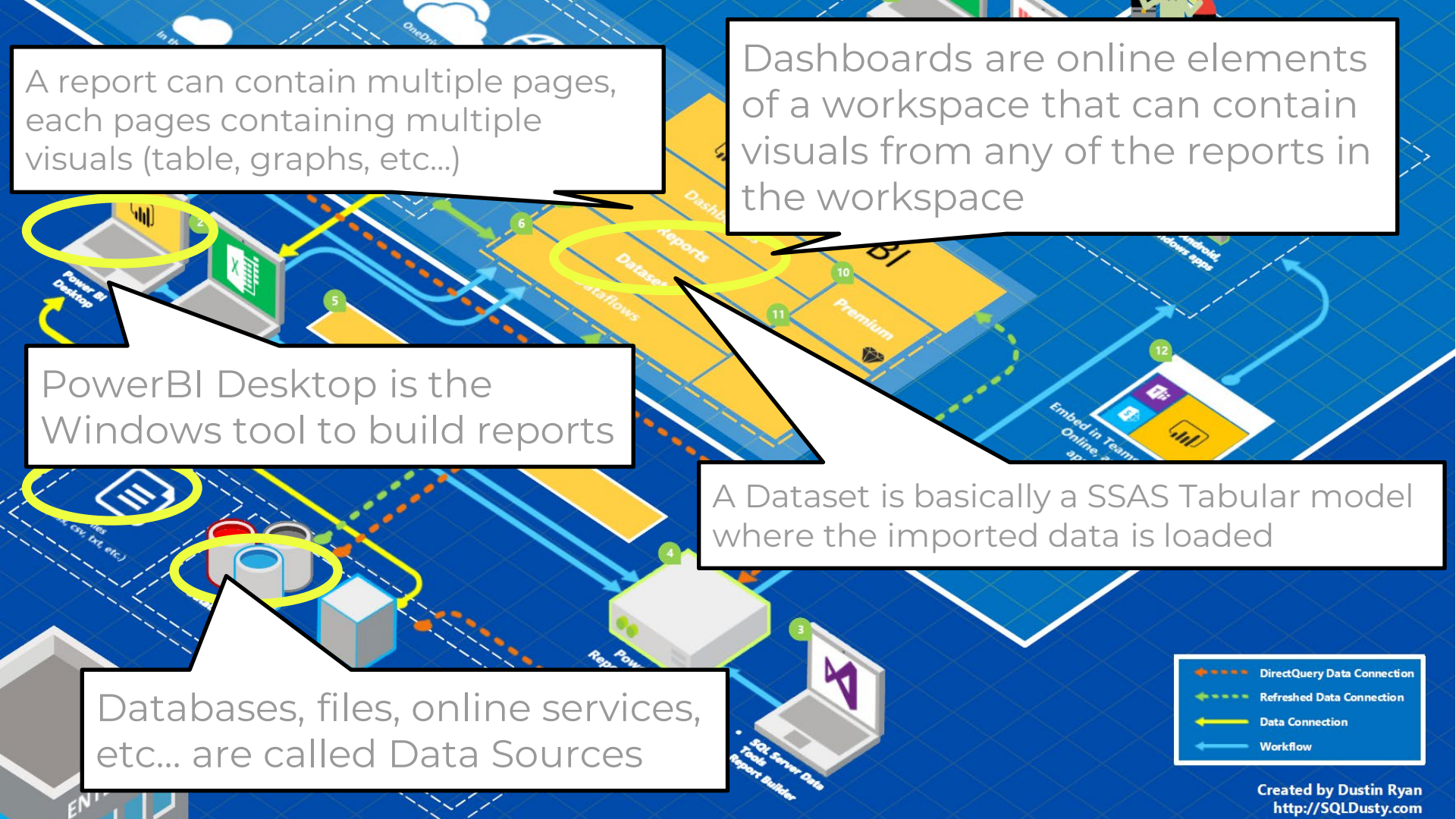
They contain all the modelling info, but none of the data/connections

You can open a template, enter your connection details and you're ready to go

PowerBI Components

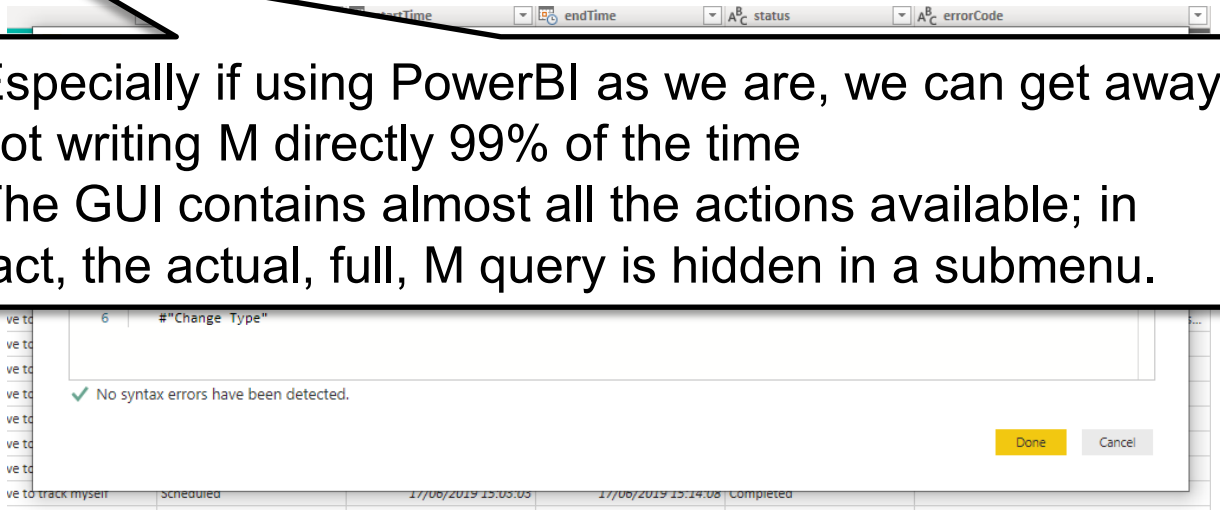
PowerBI Glossary

- **Report**: The pbix file you're working on
- **Report Page**: Each page of the report (can be hidden or visible)
- **Visual**: Each graphical element in a report page
- **Workspace**: A container for dashboards, reports and datasets
- **Dashboard**: Visuals from different reports can be displayed in a single place, called dashboard.
- **App**: The contents of a workspace can be (selectively) bundled in an app that can be distributed


A report can contain multiple pages, each pages containing multiple visuals (table, graphs, etc...)
Dashboards are online elements of a workspace that can contain visuals from any of the reports in the workspace
PowerBI Desktop is the Windows tool to build reports
A Dataset is basically a SSAS Tabular model where the imported data is loaded
Databases, files, online services, etc... are called Data Sources
Power BI Desktop
Reports
Premium
Embed in Teams
DirectQuery Data Connection
Refreshed Data Connection
Data Connection
Workflow
Created by Dustin Ryan
http://SQLDusty.com

The PowerBI Language(s)

M (aka Power Query Formula Language)



Especially if using PowerBI as we are, we can get away not writing M directly 99% of the time
The GUI contains almost all the actions available; in fact, the actual, full, M query is hidden in a submenu.

- Kind of an ETL Language, it gets you from the source data to the model Table
- If you're an Excel Power user, you probably already know (or at least use unknowingly) M

The PowerBI Language(s)

DAX

```
WGrowth =
VAR _MAX =
    CALCULATE (
        MAX ( BlitzFirst_WaitStats_Deltas[Wait Time, Minutes per Minute] );
        VALUES ( BlitzFirst_WaitStats_Deltas[wait_type] )
    )
VAR _AVG =
    CALCULATE (
        AVERAGE ( BlitzFirst_WaitStats_Deltas[Wait Time, Minutes per Minute] );
        ALL ( BlitzFirst_WaitStats_Deltas[wait_type] );
        ALL ( BlitzFirst_WaitStats_Deltas[CheckDate] )
    )
RETURN
    IF (
        RANKX (
            ALL ( BlitzFirst_WaitStats_Deltas[wait_type] );
            DIVIDE ( _MAX; _AVG );
            ;
            DESC;
            DENSE
        ) <= 5;
        DIVIDE ( _MAX; _AVG );
        BLANK ()
    )

```

- The actual data analysis language for PowerBI, also used for business analysis in SSAS Tabular & Excel PowerPivot

Storage Modes

Import:

Impor[illegible]ched. Queries submitted to the Power BI datas[illegible] fulfilled only from [illegible]

DataSet refreshes can be scheduled, but there is a hard limit of 8 scheduled refreshes per day (no limit on manual refreshes)

DirectQuery:

Tables aren't cached. Queries that you submit to the Power BI and that return data from DirectQuery tables can be fulfilled only by executing on-demand queries to the data source. Queries that you submit to the data source use the query language for that data source - for example, SQL.

PowerBI *limitations* as a monitoring tool

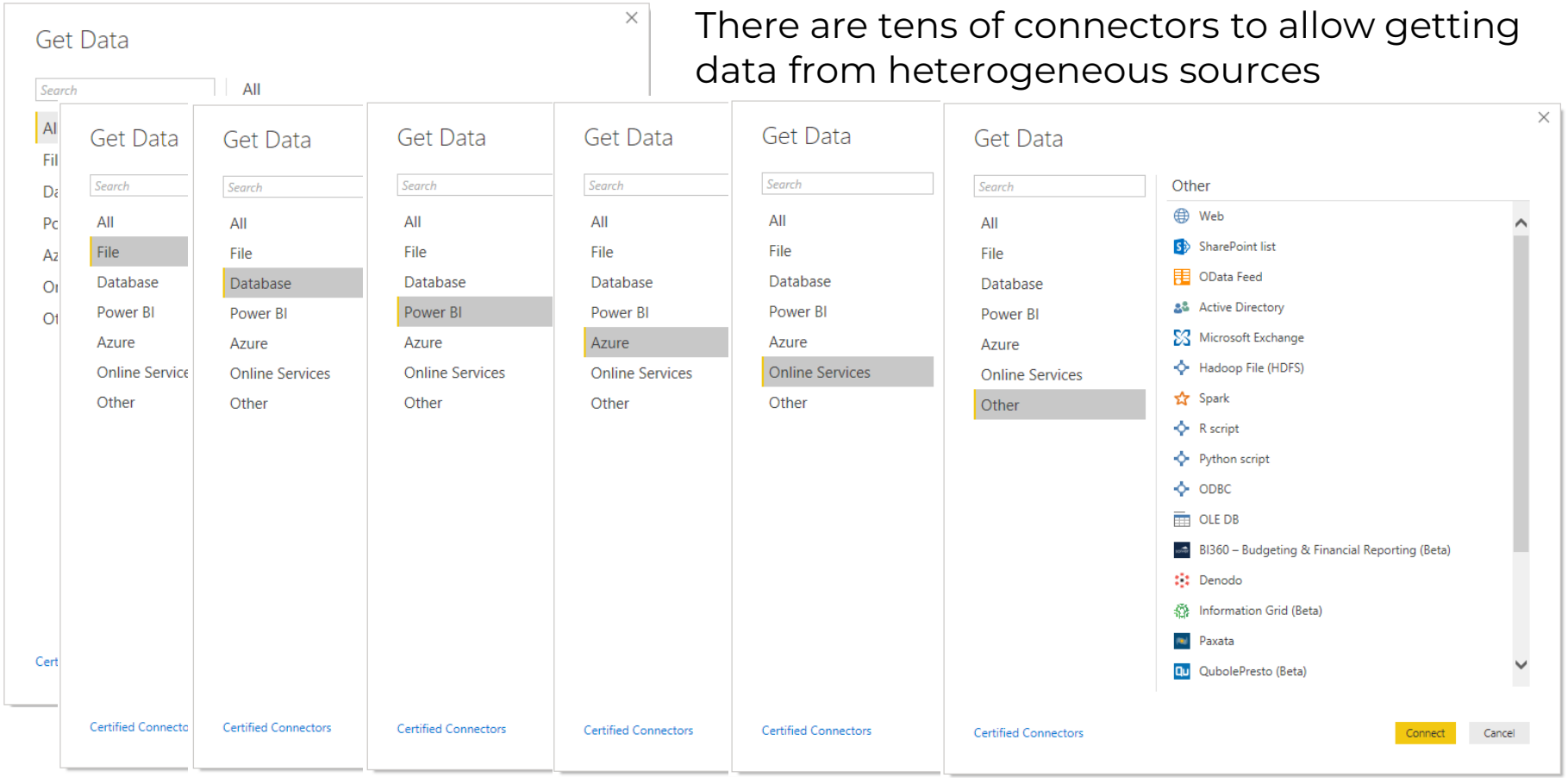
Relationships

- Relationships work best when one-to-many or one-to-one and unidirectional
- Bidirectional relationships can introduce cycles and can be blocked in the model
- There **can be only one active relationship** between one table and another (multiple keys)

Engine

- The dataset is a SSAS Tabular model, based on columnar database engine, good for aggregations, not so good for row by row operations.
- DAX syntax/logic is non straightforward coming from SQL and has a somewhat steep learning curve

Getting Data



You can find the list of PowerBI Connectors here

Finally, hands on!
Let’s create a report from scratch

Premade Examples

Disclaimer

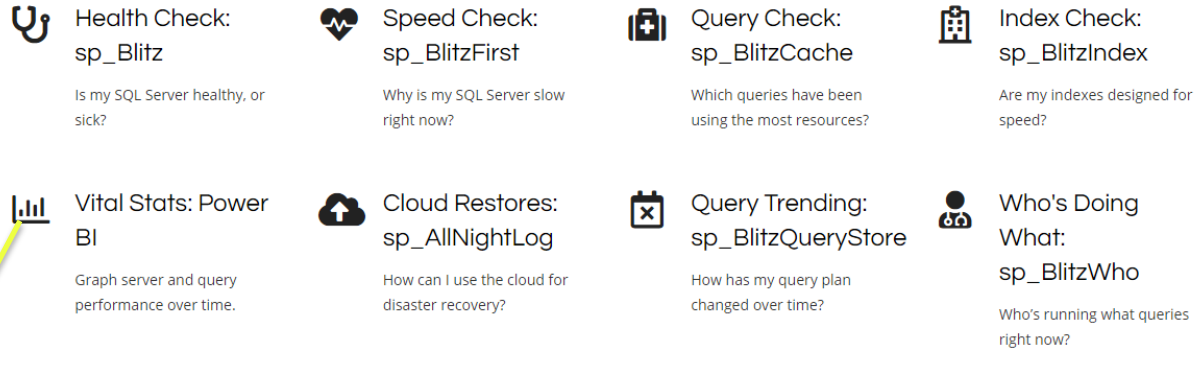
In this section a few examples of PowerBI reports for SQL Server will be shown

The point is to explain the concept via examples, which could then be applied to any platform

Since we're at a SQL Server event, I'll be focusing and providing material for this platform

FirstReponders Kit PowerBI Report

The FirstResponders Kit is an open source collection of procedures that allows SQL Server DBAs to monitor and manage their instances faster, leveraging all the relevant SQL Server DMVs



The PowerBI report is technically deprecated (issues with GitHub collaboration for a binary) However:

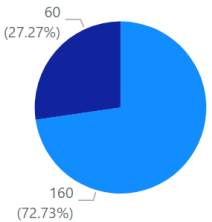
- It will work as-is until the schema of the table/views in the kit is changed (probably never)
- I'll continue to support and evolve it over my repository (as I was it's major contributor, I grew attached to it)

Xevents Dashboard (track query errors)

timestamp UTC	username	database_name	client_app_name	message
9/30/2019 3:08:16 PM	oneuser	tpch	Tk 8.6 for Windows	Conversion failed when converting date and/or time from character string.
9/30/2019 3:07:52 PM	oneuser	tpch	Tk 8.6 for Windows	Incorrect syntax near '-'.
9/30/2019 3:07:52 PM	oneuser	tpch	Tk 8.6 for Windows	Statement(s) could not be prepared.
9/30/2019 3:07:16 PM	oneuser	tpch	Tk 8.6 for Windows	Conversion failed when converting date and/or time from character string.
9/30/2019 3:06:17 PM	oneuser	tpch	Tk 8.6 for Windows	Incorrect syntax near '-'.
9/30/2019 3:06:17 PM	oneuser	tpch	Tk 8.6 for Windows	Statement(s) could not be prepared.
9/30/2019 3:05:59 PM	oneuser	tpch	Tk 8.6 for Windows	Incorrect syntax near '-'.
9/30/2019 3:05:59 PM	oneuser	tpch	Tk 8.6 for Windows	Statement(s) could not be prepared.
9/30/2019 3:05:54 PM	oneuser	tpch	Tk 8.6 for Windows	Incorrect syntax near '-'.
9/30/2019 3:05:54 PM	oneuser	tpch	Tk 8.6 for Windows	Statement(s) could not be prepared.
9/30/2019 3:05:46 PM	oneuser	tpch	Tk 8.6 for Windows	Conversion failed when converting date and/or time from character string.

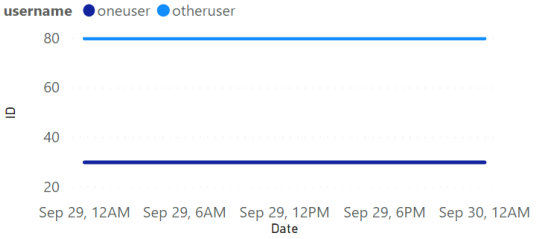
```
select supp_nation, cust_nation, l_year, sum(volume) as revenue
from ( select n1.n_name as supp_nation, n2.n_name as cust_nation,
    datepart(yy,l_shipdate) as l_year, l_extendedprice * (1 -
l_discount) as volume from supplier, lineitem, orders, customer,
 nation n1, nation n2 where s_suppkey = l_suppkey and o_orderkey
    = l_orderkey and c_custkey = o_custkey and s_nationkey =
n1.n_nationkey and c_nationkey = n2.n_nationkey and ( (n1.n_name
    = 'UNITED KINGDOM' and n2.n_name = 'KENYA') or (n1.n_name =
'KENYA' and n2.n_name = 'UNITED KINGDOM')) and l_shipdate between
    '1995-22-01' and '106-12-31') shipping group by supp_nation,
 cust_nation, l_year order by supp_nation, cust_nation, l_year
                option (maxdop 2)
```

Errors by username



username
otheruser
oneuser

Errors by username, by day

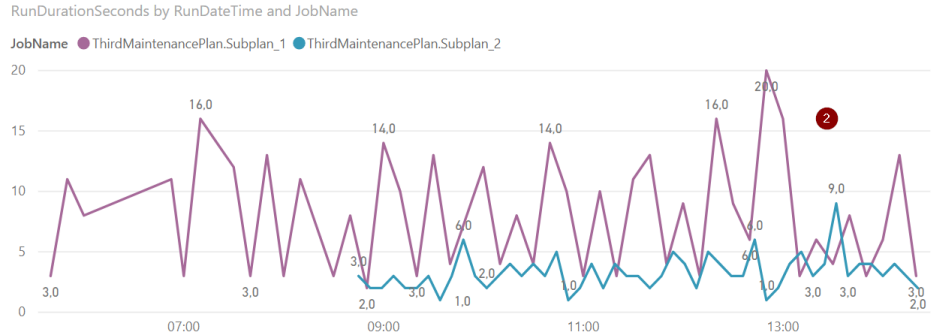


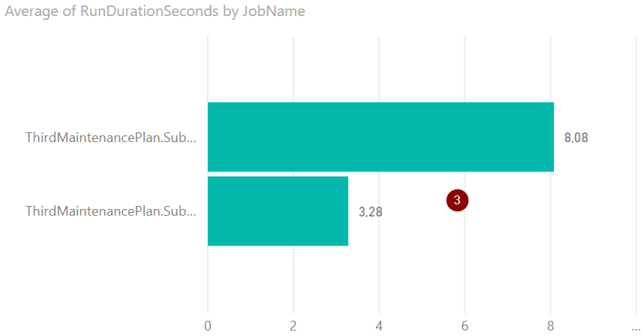
username
oneuser
otheruser

client_app_name
Tk 8.6 for Windows

Date
This
-
Month
9/1/2019 - 9/30/2019

Job Information Dashboard





Job Overview Information

ThirdMaintenancePla...	Enabled	Daily	Sep 23 2018 12:10PM	Sep 23 2018 12:20PM	N/A	Every 1 day(s
name	is_enabled	frequency	last_run_date	next_run_date	last_failure_date	schedule
ThirdMaintenancePla...	Yes	Occurs every 10 Min...	50	0	0	0
schedule_name	schedule_enabled	schedule_detail	job_success	job_cancel	job_fail	job_retry
Never	Not Enabled	When the job fails	No operator	THEBEAST\Menny	00:00:00:02	00:00:00:20
email_when	is_operator_enabled	log_when	operator_name	owner_name	shortest_duration	longest_duration

ThirdMaintenancePla...	Enabled	Daily	Sep 23 2018 12:08PM	Sep 23 2018 12:15PM	N/A	Every 1 day(s
name	is_enabled	frequency	last_run_date	next_run_date	last_failure_date	schedule
ThirdMaintenancePla...	Yes	Occurs every 7 Minut...	50	0	0	0
schedule_name	schedule_enabled	schedule_detail	job_success	job_cancel	job_fail	job_retry
Never	Not Enabled	When the job fails	No operator	THEBEAST\Menny	00:00:00:01	00:00:00:10
email_when	is_operator_enabled	log_when	operator_name	owner_name	shortest_duration	longest_duration

1

RunDateTime

Last 2 Weeks

10/09/2018 - 23/09/2018 4

JobName

- sysutility_get_cache_tables_data_into_aggregate_ta...
- sysutility_get_cache_tables_data_into_aggregate_ta...
- sysutility_get_views_data_into_cache_tables
- ThirdMaintenancePlan.Subplan_1 5
- ThirdMaintenancePlan.Subplan_2

JobName Search 6

Category All 7

RunDurationSeconds

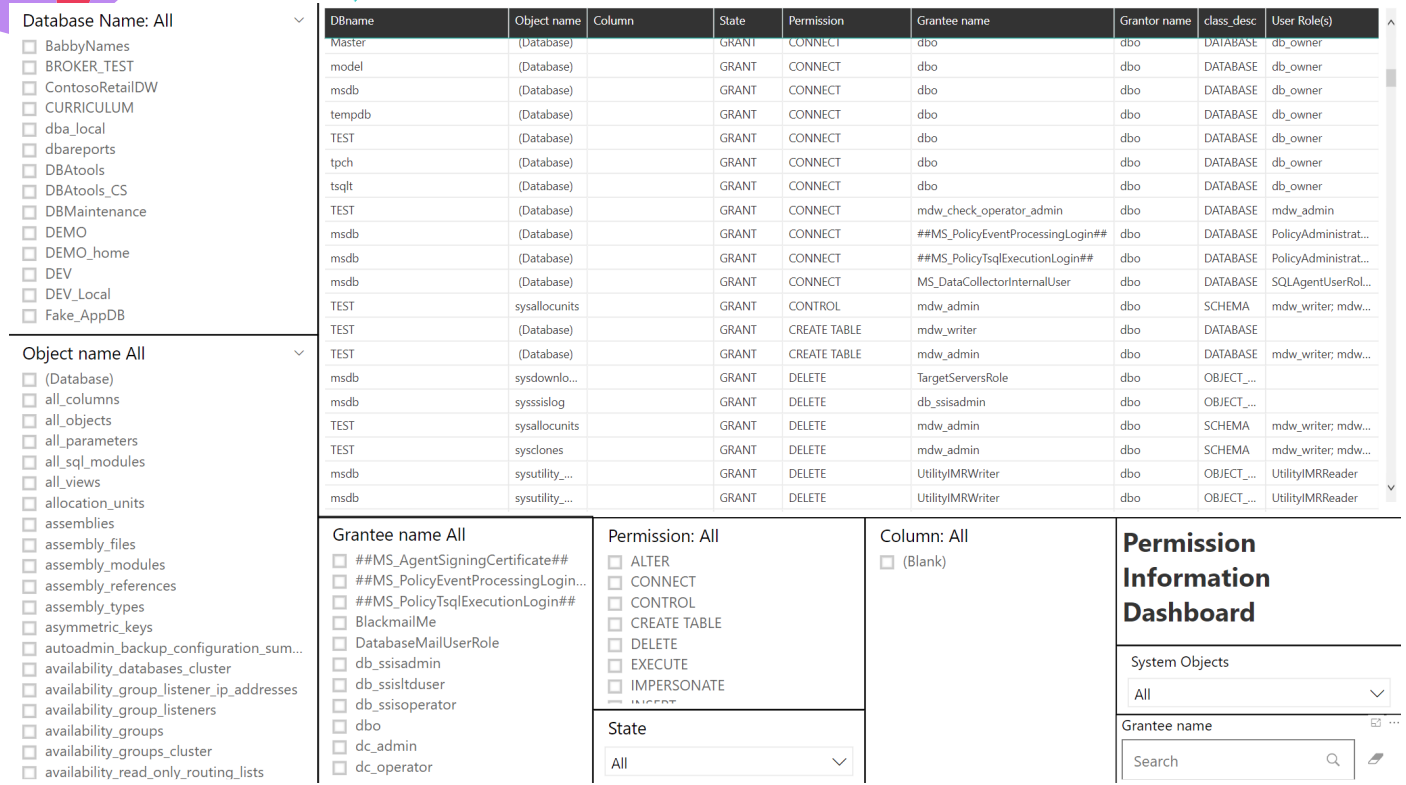
0 435063 8

Job Status 9 All

Schedule_enabled 10 All

https://tsql.tech/a-powerbi-report-for-sql-server-agent-jobs/

Permission Information Dashboard

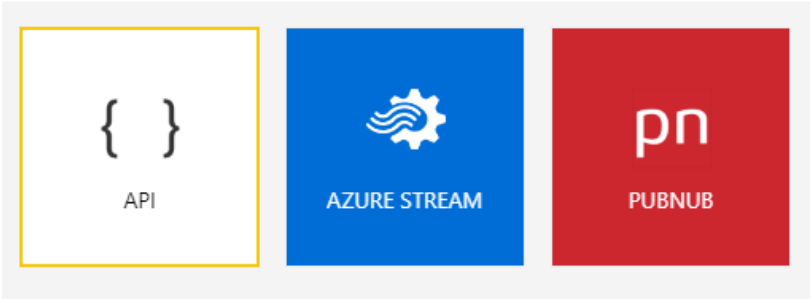


https://tsql.tech/a-sql-server-permission-report-in-powerbi/

Streaming Dataset

New streaming dataset

Choose the source of your data



Data can be **live** streamed to a Dashboard

API info on Demo Stream

Use the API endpoint URL and one of the examples shown below to send data to your streaming dataset. For more information, read our API documentation and integration guide.

Push URL

https://api.powerbi.com/beta/bf6a4def-a7e9-4a4e-a417-9e19ff2599b5/datas

Raw cURL PowerShell

```
[
  {
    "MachineName" : "AAAAA555555",
    "TimeStamp" : "2019-09-21T22:40:01.475Z",
    "FreeMemoryBytes" : 98.6,
    "TotalProcessorTime" : 98.6
  }
]
```

Thanks!

For any question feel free to contact me in any way you feel comfortable with
Contact details are on the 2nd slide!