Brian Bønk Rueløkke

Principal & Enterprise arkitekt, Data & Analytics

Fellowmind

https://linkedin.com/in/brianbonk

https://brianbonk.dk

Microsoft MVP

FastTrack Recognized Solution Architect Power BI
2022 >>

Microsoft Certified Trainer
Data Platform

2018 >>

Agenda

Quick warm up demo

What is Kusto / ADX + free stuff

The marriage with Power BI

Demo

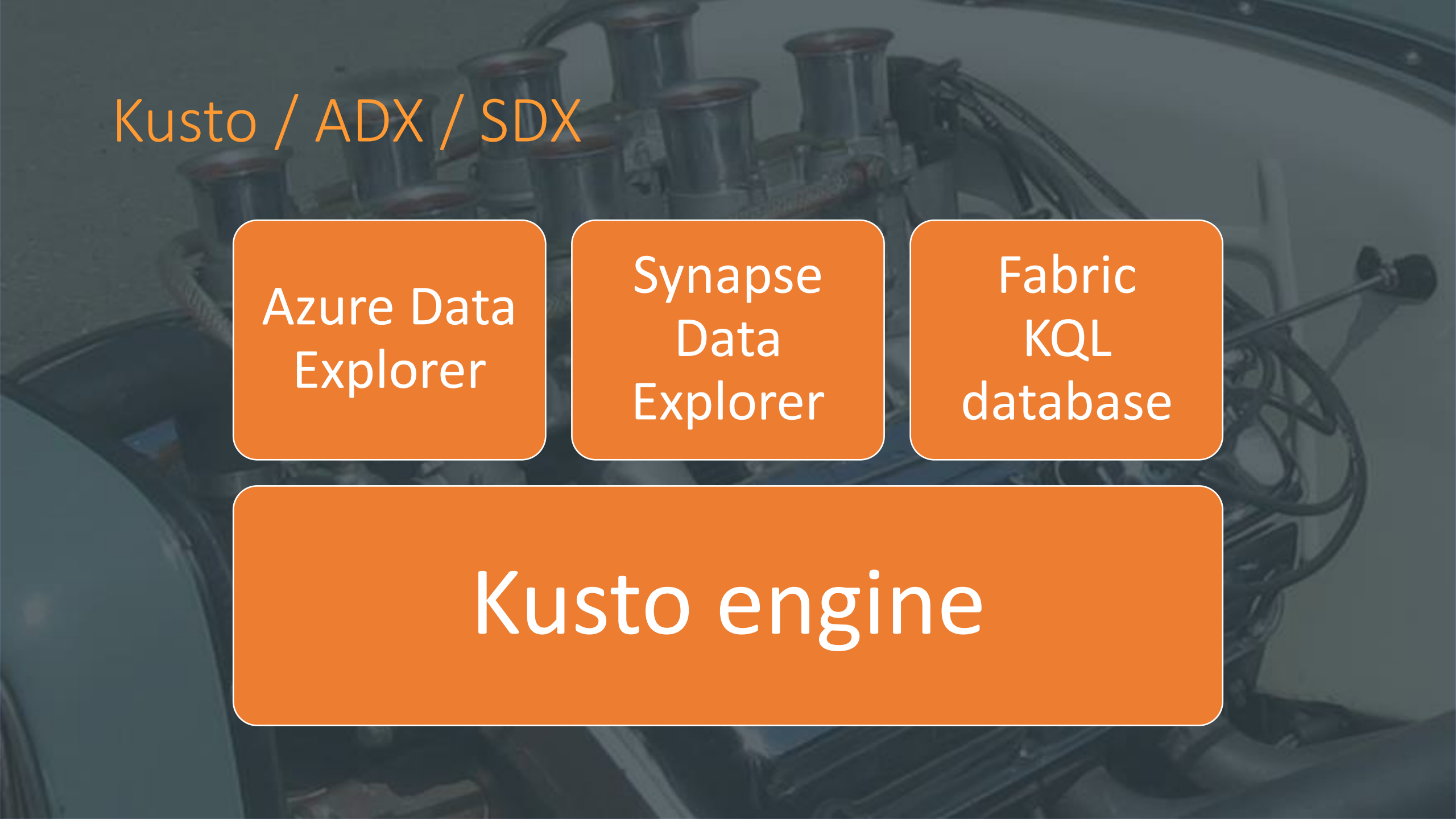
QUICK WARM UP

DEMO

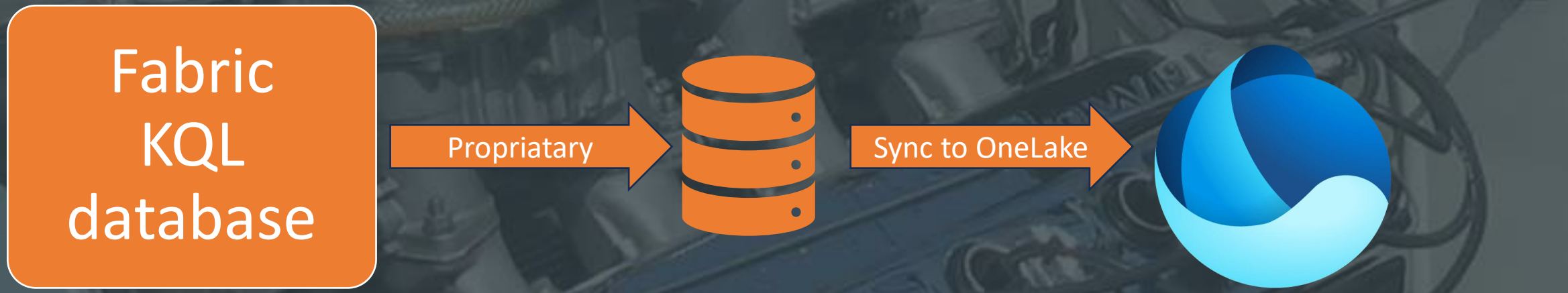
Live coding

(hopefully no demo-ghost 👻)

Kusto / ADX / SDX



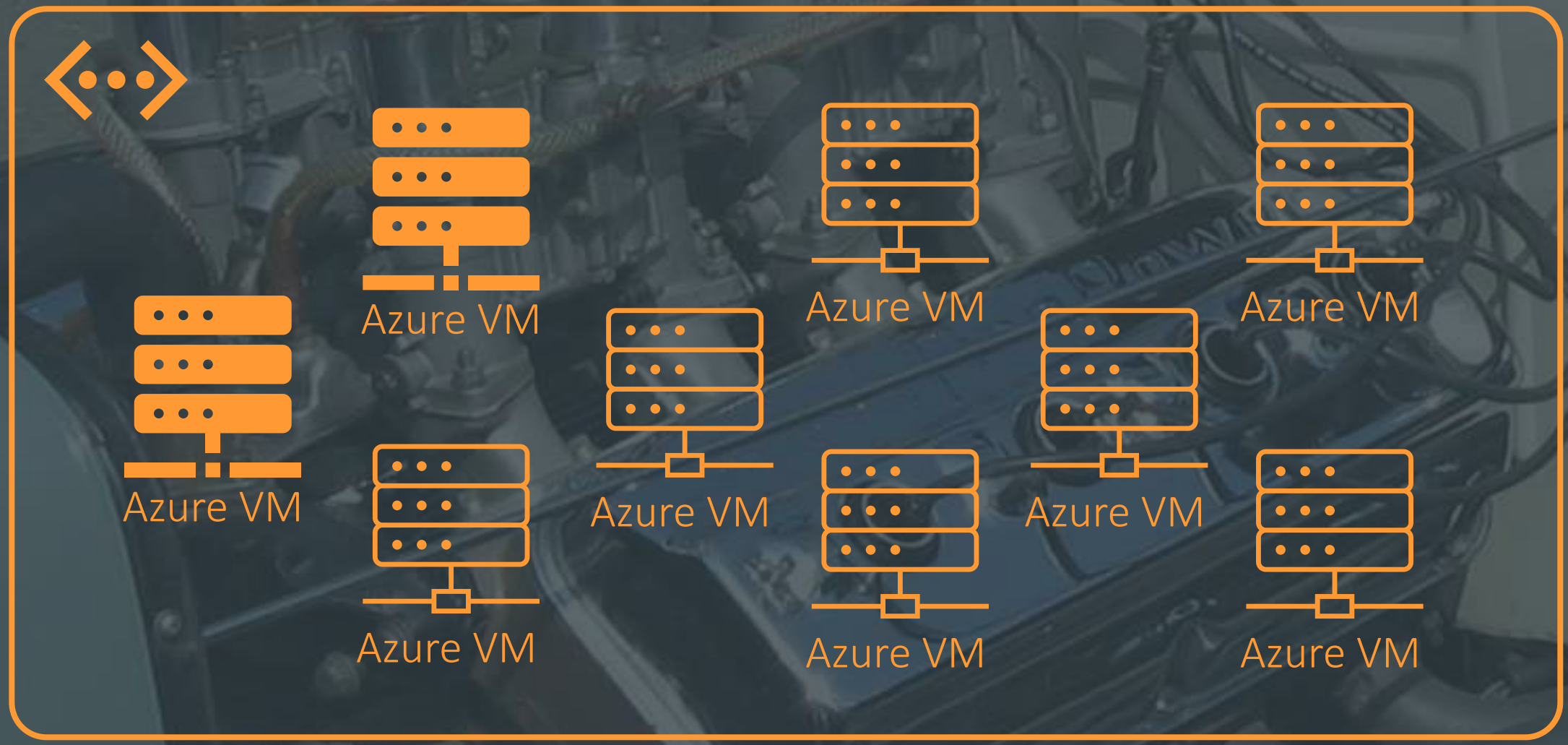
The Fabric experience for Kusto



Kusto / ADX / SDX

- Distributed engine on several disks (SSD or spinning)
- Columnar storage → read only needed columns and not all data
- Clusters are Azure VMs → highly parallel
  - For Fabric we don’t know yet
- Read-only, delete rarely and no updates
- Fast ingestion → no consistency checks

Kusto / ADX / SDX



Kusto / ADX / SDX

Admin node:
responsible for maintaining the overall cluster metadata

Query Head:
Responsible for accepting and processing Kusto query, when you see, Kusto engine or Kusto query planer, usually refer to query head node.

Data Node:
The most common role, like its name indicates, this node is responsible for: first. storing data; second. contribute the CPU and memory when executing the Kusto query.

Gateway Node:
Responsible for processing external API calls, authentication, and request dispatches.

Fabric architecture for Kusto

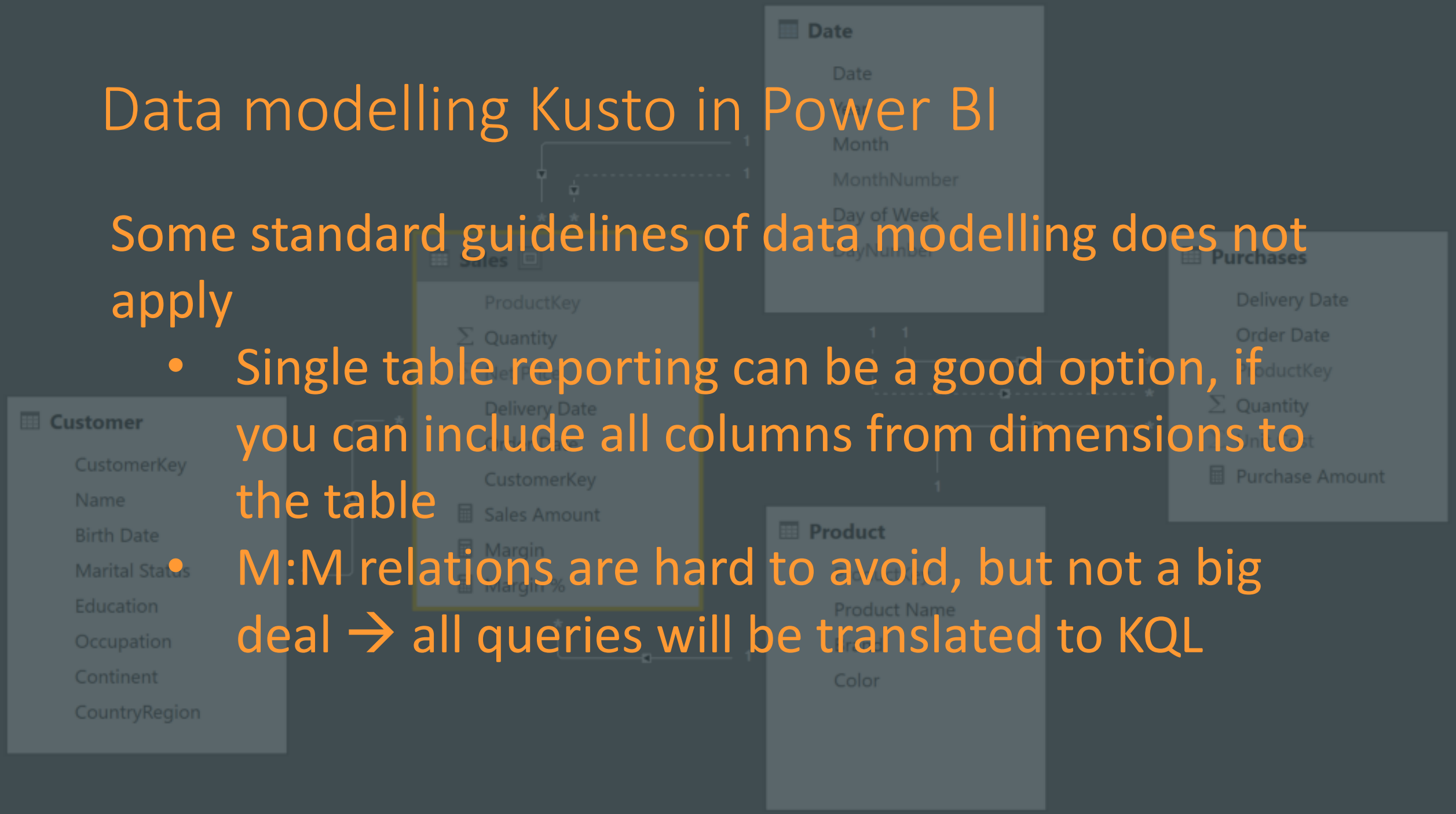
That’s all we know right now


Get started for free
https://dataexplorer.azure.com/freecluster

Data modelling Kusto in Power BI

Some standard guidelines of data modelling does not apply

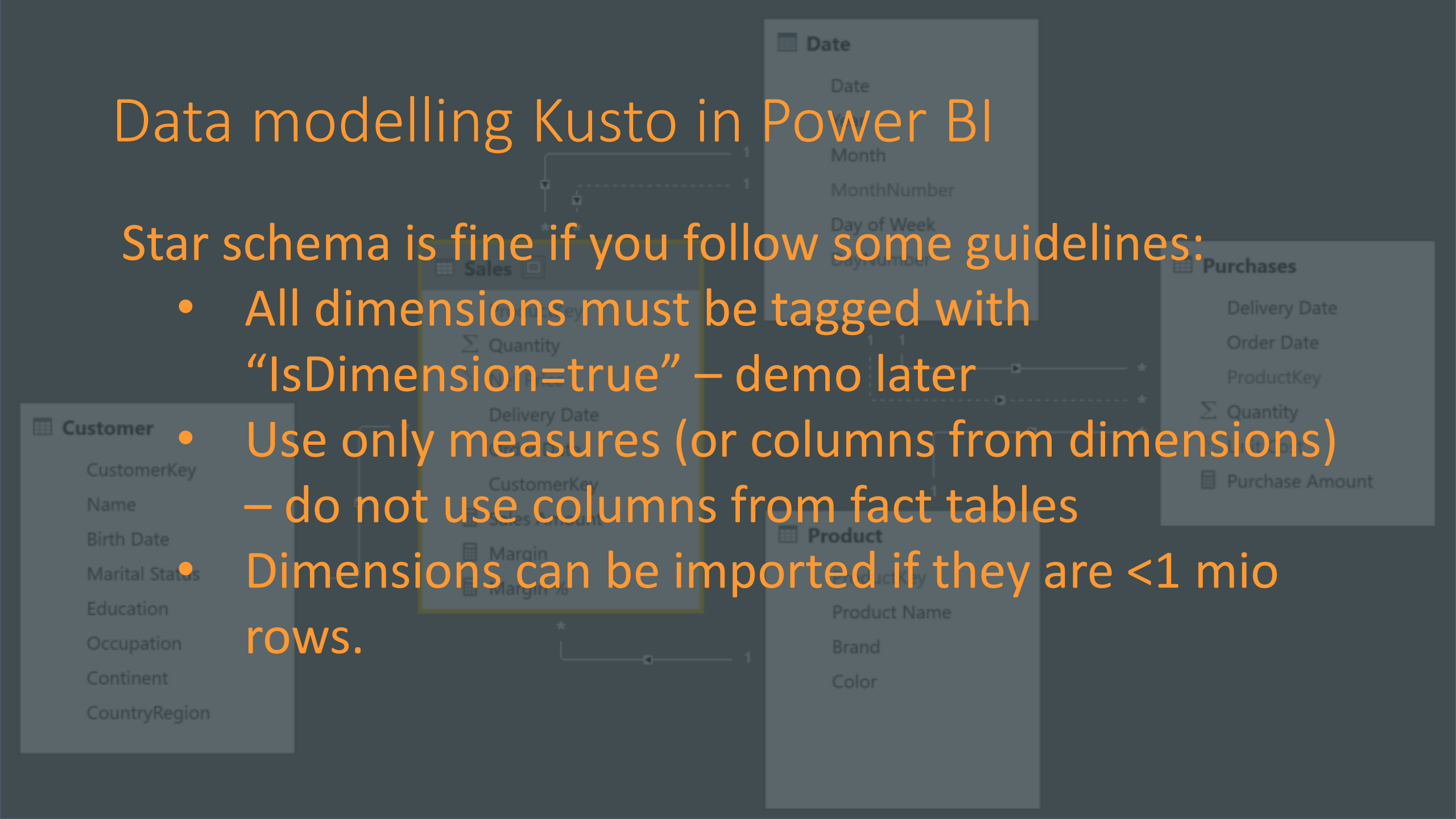
- Single table reporting can be a good option, if you can include all columns from dimensions to the table
- M:M relations are hard to avoid, but not a big deal → all queries will be translated to KQL



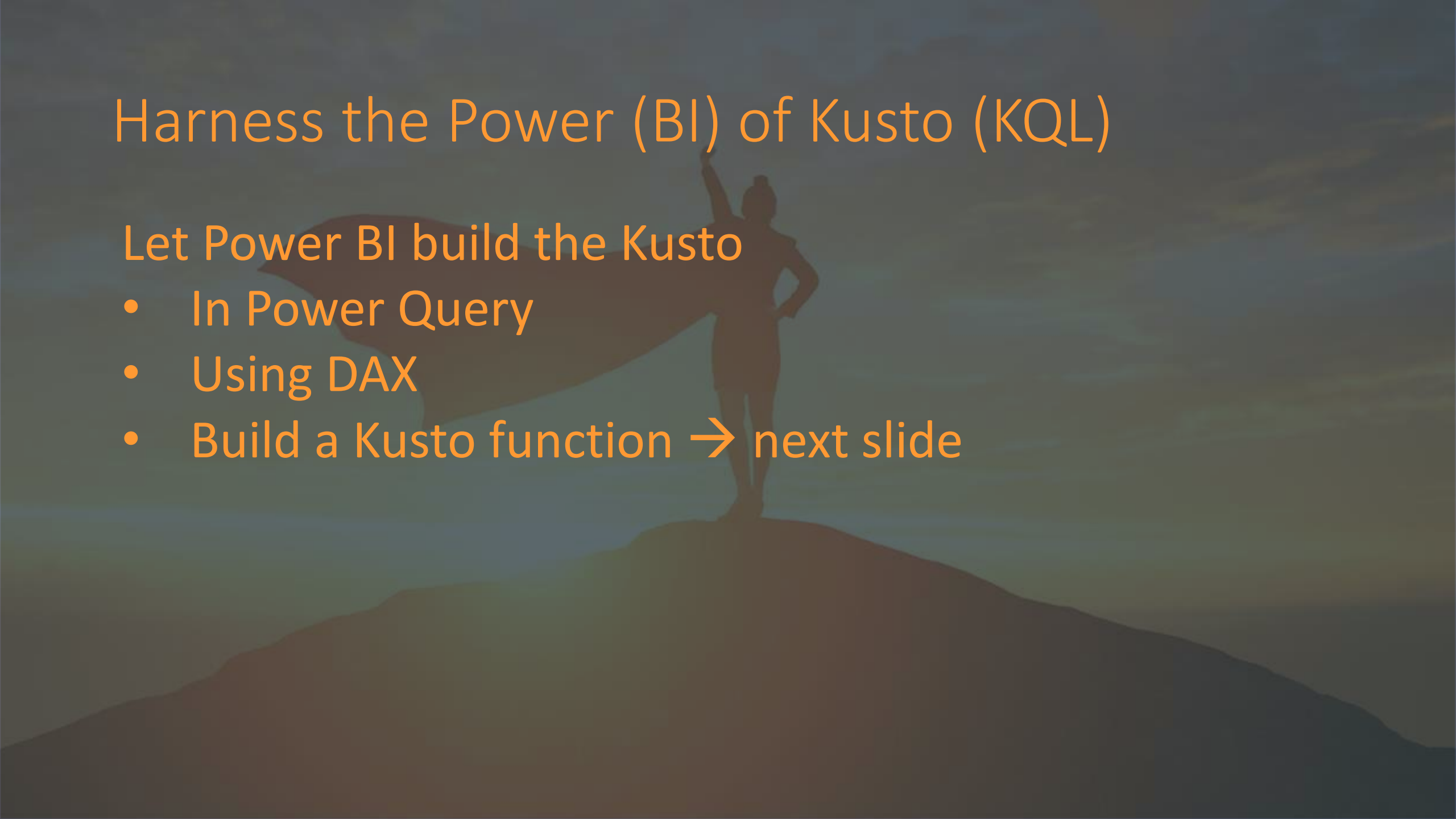
Data modelling Kusto in Power BI

Star schema is fine if you follow some guidelines:

- All dimensions must be tagged with “IsDimension=true” – demo later
- Use only measures (or columns from dimensions) – do not use columns from fact tables
- Dimensions can be imported if they are <1 mio rows.



Harness the Power (BI) of Kusto (KQL)



Let Power BI build the Kusto

- In Power Query
- Using DAX
- Build a Kusto function → next slide

Harness the Power (BI) of Kusto (KQL)

```
.create-or-alter function GetSysLogs(TimeWindow:string , Bucket:string )
{
cluster('help').database('SampleLogs').RawSysLogs
| where timestamp > ago(totimespan(TimeWindow))
| summarize LogCount=count() by name, bin(timestamp, totimespan(Bucket))
| order by timestamp asc
}

GetSysLogs('5d','1h')
```

Power BI Builds the query

From Power Query From Filters From Relations From Visuals

```
["Customers"]
| where ["RegionCountryName"] == "United States"
| project-rename ["semijoin1.c5"] = ["CustomerKey"], ["semijoin1.c27"] =
["StateProvinceName"]
| join hint.strategy=broadcast kind=inner (["SalesFact"]
| project ["SalesAmount"],["TotalCost"],["CustomerKey"]
| extend ["Margin"]=(["SalesAmount"]) -
(["TotalCost"]),["t0_0"]=tolong(["SalesAmount"])
| project-rename ["basetable0.c5"] = ["CustomerKey"], ["basetable0.a0"] = ["t0_0"],
["basetable0.a1"] = ["t2_0"]) on $left.["semijoin1.c5"] == $right.["basetable0.c5"]
| summarize ["a0"]=sum(["basetable0.a0"]), ["a1"]=sum(["basetable0.a1"]) by
["semijoin1.c27"]
```

The marriage with Power BI

DEMO

Live coding
(hopefully no demo-ghost 👻)

Monitor your Kusto queries

- Use .show queries → demo later
- Separate queries by user
- Separate queries by report
- Pay attention to CPU consumption and not just duration

Monitor your queries

DEMO

Live coding
(hopefully no demo-ghost 👻)

.show queries

```
.show commands-and-queries
// | where ClientActivityId startswith "KPBI"
| where StartedOn > datetime(2022-12-04T09:04:51.198128Z)
| where User ==current_principal_details().UserPrincipalName
| order by StartedOn asc
| extend delay=datetime_diff("Millisecond",next(StartedOn),LastUpdatedOn)
| extend delay=iff(delay<0 or delay> 5000,0,delay)
| extend MB=format_bytes(MemoryPeak)
| extend Isgetschema=Text has "getschema"
| extend IsPreview=Text has "limit 1000 "
| extend Len=strlen(Text)
| extend TextLength=strlen(Text)
| extend ScannedData=format_bytes(tolong(CacheStatistics.Shards.Hot.HitBytes))
| fork
  Queries=(where CommandType =="Query" and  Isgetschema==false and IsPreview ==false |  project StartedOn,  LastUpdatedOn,  Duration, TotalCpu, ScannedData,  MB,Text | order by StartedOn asc)
  Commands=(where CommandType =="AdminThenQuery" |  project StartedOn,LastUpdatedOn, Duration, TotalCpu, MB, ScannedData ,Text | order by StartedOn asc)
  Detail=(project StartedOn,State,FailureReason, Duration, delay, TotalCpu,Isgetschema, MB, ScannedData,ClientActivityId, Text | order by StartedOn asc)
  Slow=(where CommandType =="Query" | project Duration, TotalCpu, MB, ScannedData,Text | order by Duration)
  Getschema=(where CommandType =="Query" and  Isgetschema==true |  project StartedOn,Text)
  Summary=(summarize Commands=countif(CommandType =="AdminThenQuery"),Queries=countif(CommandType=="Query"),DelayCommand=sumif(delay, CommandType =="AdminThenQuery"),
    DelayQuery=sumif(delay, CommandType =="Query"),mn=min(StartedOn),mx=max( LastUpdatedOn),TotCPU=sum(TotalCpu),TotDuration=sum(Duration)
     | extend OveralDuration=(mx-mn))
```

Thank you!

Brian Bønk Rueløkke

Principal & Enterprise arkitekt, Data & Analytics

Fellowmind

https://linkedin.com/in/brianbonk

https://brianbonk.dk

Microsoft MVP

FastTrack Recognized Solution Architect Power BI 2022 >>

Microsoft Certified Trainer Data Platform 2018 >>