3Cloud | Microsoft Gold Partner

3CLOUD SOLUTIONS PRESENTS

Moving informal Power BI projects to Enterprise Scale

Paul Turley

Principal Consultant,

Microsoft Data Platform MVP | FastTrack Recognized Solution Architect

3Cloud | Microsoft Gold Partner

Learn how to transition quick, informal Power BI projects to formal, futureproof solutions.
Plan for scale and larger data volumes.
Develop according to best practice designs, support versioning and multi-developer project teams.
Demonstrate techniques, share tips and design patterns from blog series: “Doing Power BI the Right Way”.

Paul Turley

Principal Consultant, Microsoft Data Platform MVP

~25 years in IT, data platform, Business Intelligence & data analytics

SqlServerBI.blog
DataOnTheRoad.blog

TV news

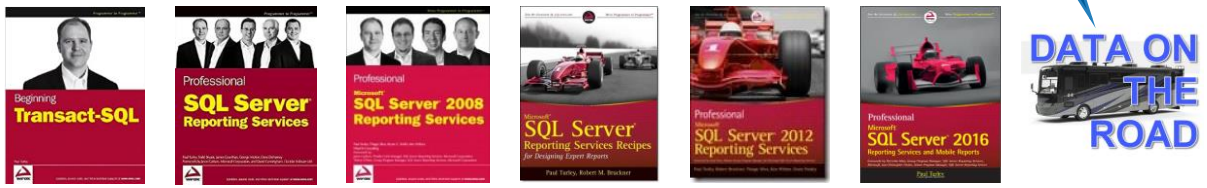
Helpdesk — Healthcare software

IT Training — Broke foot — IT Training & Consulting

BI Analytics — Data Reporting & BI

1986 1989 2003 2009 2021

MVP Microsoft Most Valuable Professional



14 books

DATA ON THE ROAD

Agenda

Let’s talk about what we’re going to talk about

3Cloud

Gold Microsoft Partner
Azure Expert MSP
Microsoft

Microsoft Partner
Microsoft
Gold Cloud Platform
Gold Datacenter
Gold Application Development
Gold Data Platform
Gold Data Analytics
Gold DevOps

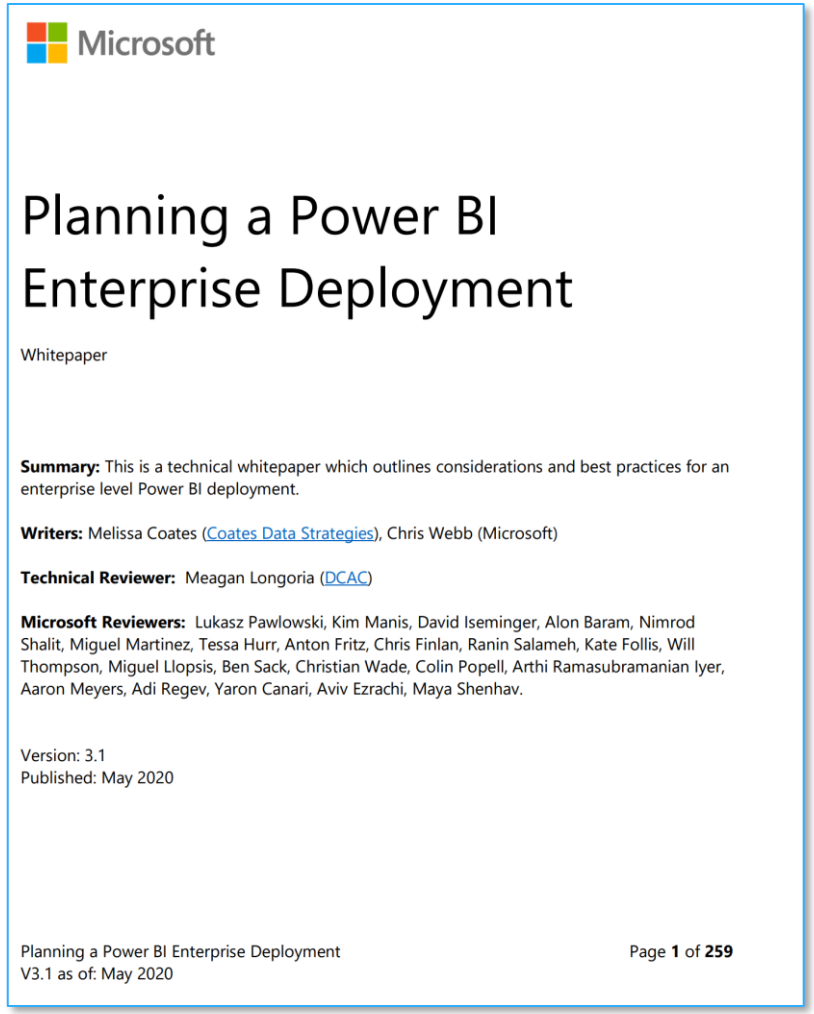
Advanced specializations:

Modernization of Web Applications

Windows and SQL Server Migration

Is Power BI Enterprise Ready?

- ✓ Large-scale Data
- ✓ Broad user support
- ✓ Team Development & Collaboration
- ✓ Data Governance Policies
- ✓ Data Management Policies
- ✓ Lifecycle & Deployment
- ✓ Versioning
- ✓ Embedding & Integration
- ✓ Automation
- ✓ Administration
- ✓ Security:
  - User Access & Data Protection
  - Role-based & User-based filtering



Microsoft

Planning a Power BI Enterprise Deployment

Whitepaper

Summary: This is a technical whitepaper which outlines considerations and best practices for an enterprise level Power BI deployment.

Writers: Melissa Coates (Coates Data Strategies), Chris Webb (Microsoft)

Technical Reviewer: Meagan Longoria (DCAC)

Microsoft Reviewers: Lukasz Pawlowski, Kim Manis, David Iseminger, Alon Baram, Nimrod Shalit, Miguel Martinez, Tessa Hurr, Anton Fritz, Chris Finlan, Ranin Salameh, Kate Follis, Will Thompson, Miguel Llopsis, Ben Sack, Christian Wade, Colin Popell, Arthi Ramasubramanian Iyer, Aaron Meyers, Adi Regev, Yaron Canari, Aviv Ezrachi, Maya Shenhav.

Version: 3.1
Published: May 2020

Planning a Power BI Enterprise Deployment
V3.1 as of: May 2020
Page **1** of **259**

White papers for Power BI - Power BI | Microsoft Docs:
https://docs.microsoft.com/en-us/power-bi/guidance/whitepapers

Typical Power BI Use Cases

- Ad hoc analysis
- Quick reporting
- Transformation & shaping as-you-go
- Data model evolution
- Visualize: try-and-see
- Standards? Who needs standards?
    - Naming conventions
    - Modeling & relationships
    - Measure logic
    - Documentation
    - Version control

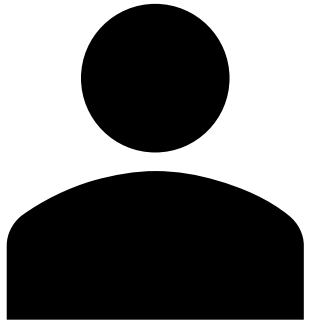
Audience, Size & Scale

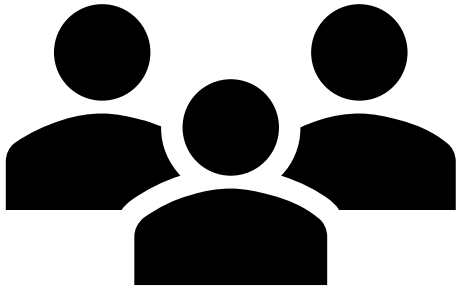
One size doesn’t always fit all

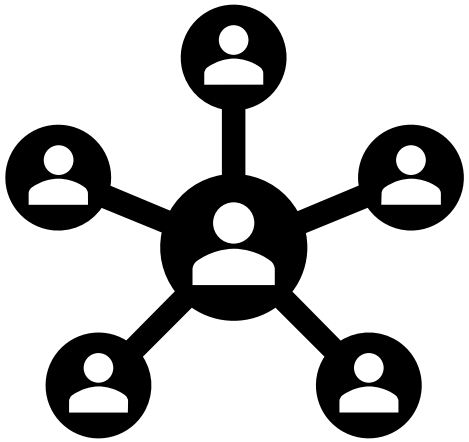
With planning, informal projects can be adapted to meet enterprise needs

Know when to keep and update a design

Know when to toss a design and start over

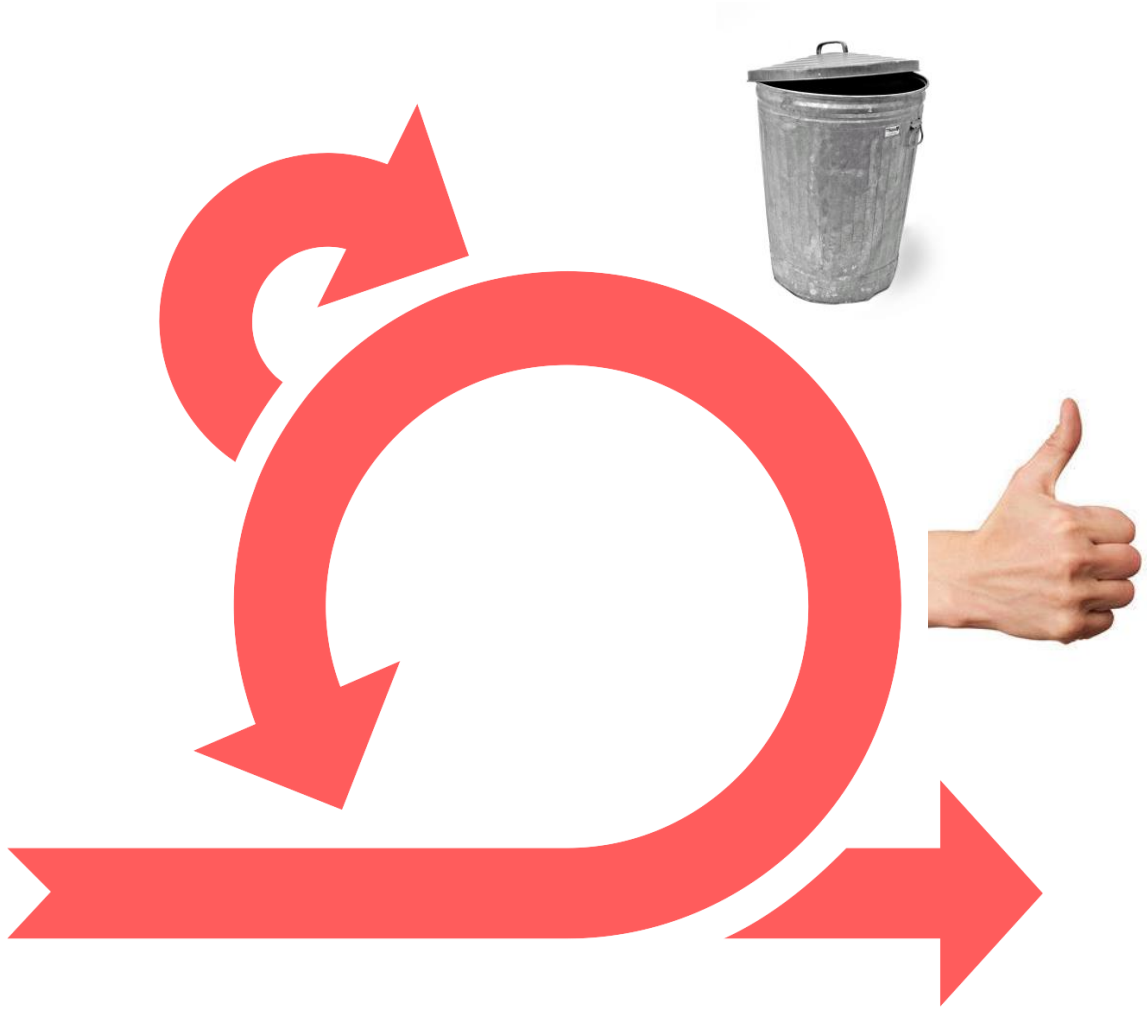






Iterative Design

- Define solution scope
- Work in iterative cycles
- Evaluate design
- Keep what works
- Discard what doesn't
- Learn & improve
- Evaluate to solution scope



Decision Criteria

What components of the solution have “good bones”?

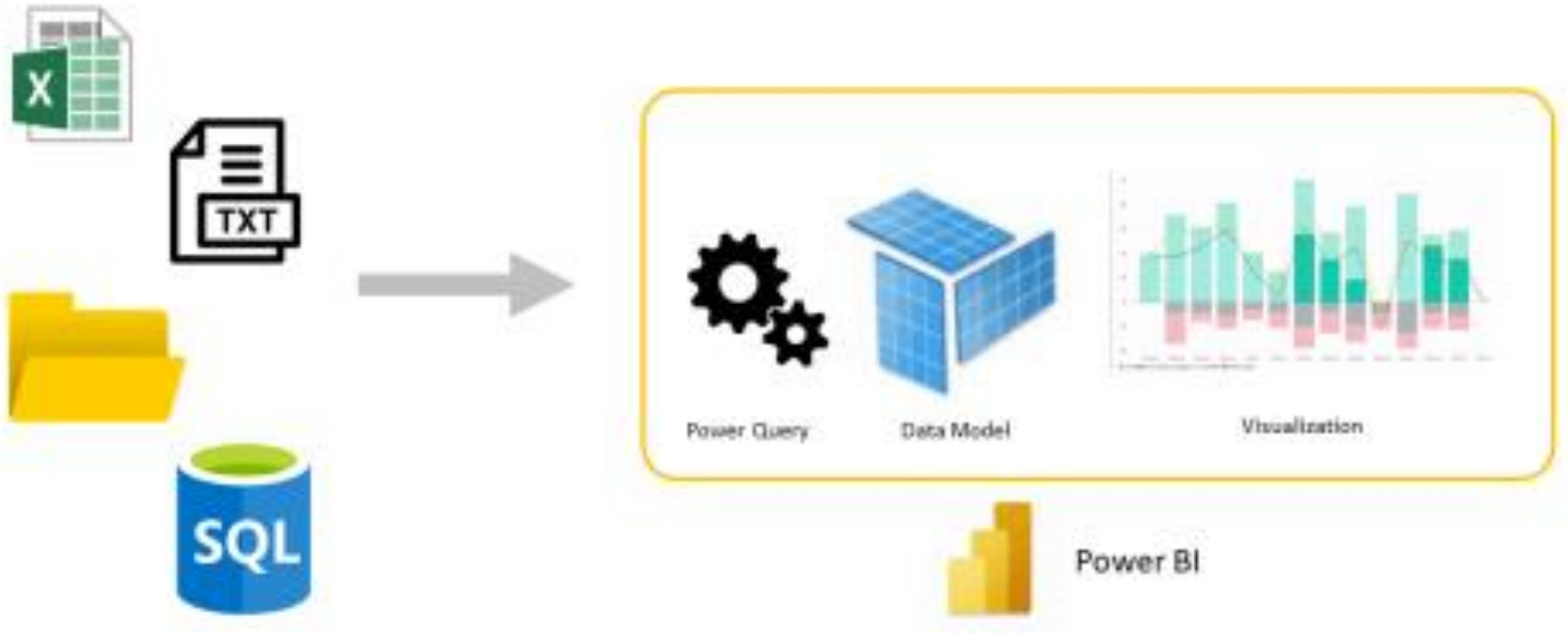
Data Source Objects	Files, queries, tables, views		
Transformation Choice	ELT or ETL, SQL queries, Power Query (does it scale?)		
Data Model Design	Flat tables, star schema, bridges & complex relationships		
Development Standards	Naming conventions, Traceable table & column references		
Deployment Method	Manual, Power BI Desktop, Visual Studio or Tabular Editor, Scripted, Automated		
Report Delivery Method	Web portal, Mobile, Publish to web, Embedded, Teams		

Architecture Patterns

- Is the current solution:
- A good proof of concept to learn from?
- A solid foundation to build upon?

Architecture Patterns

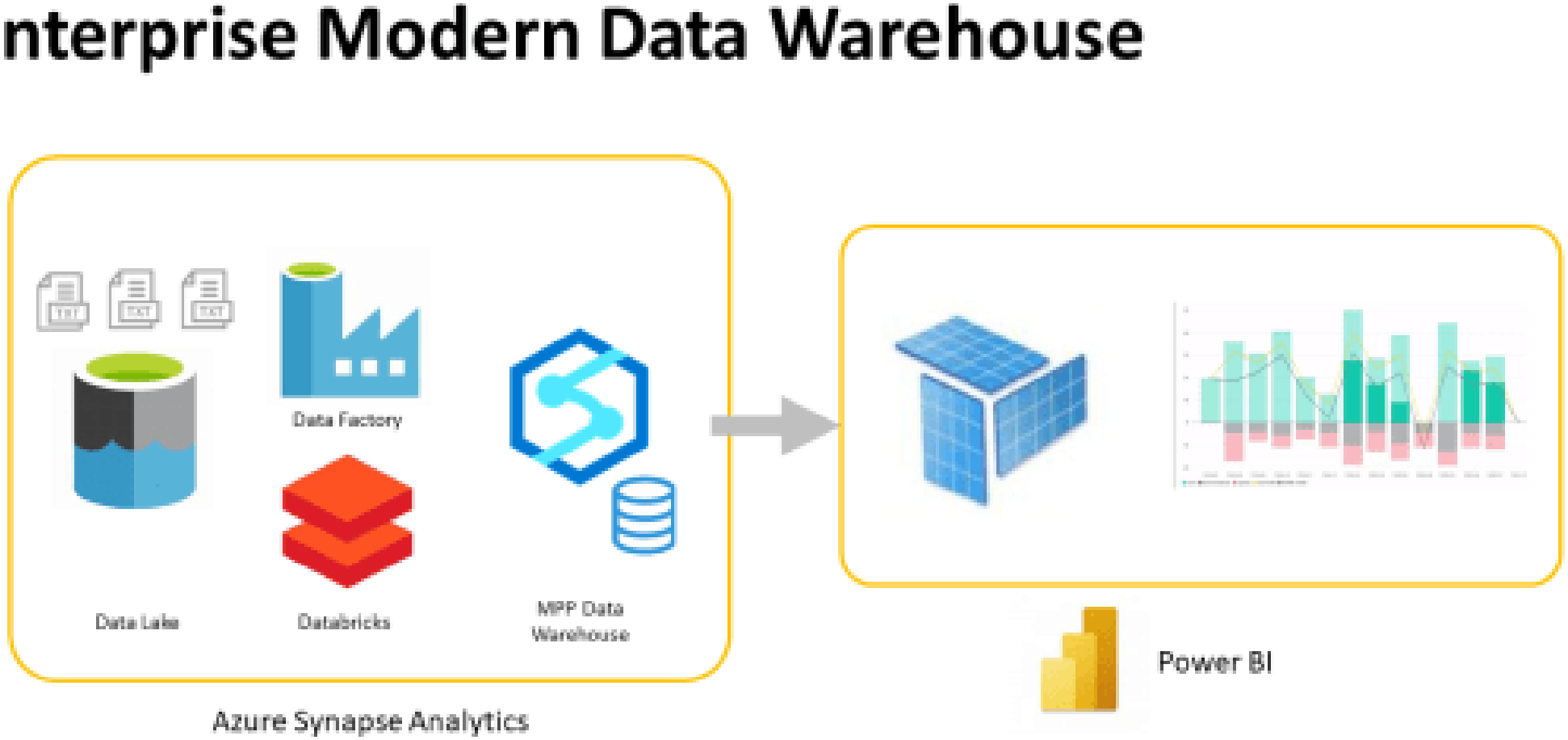
Self-service Data Prep & Analysis



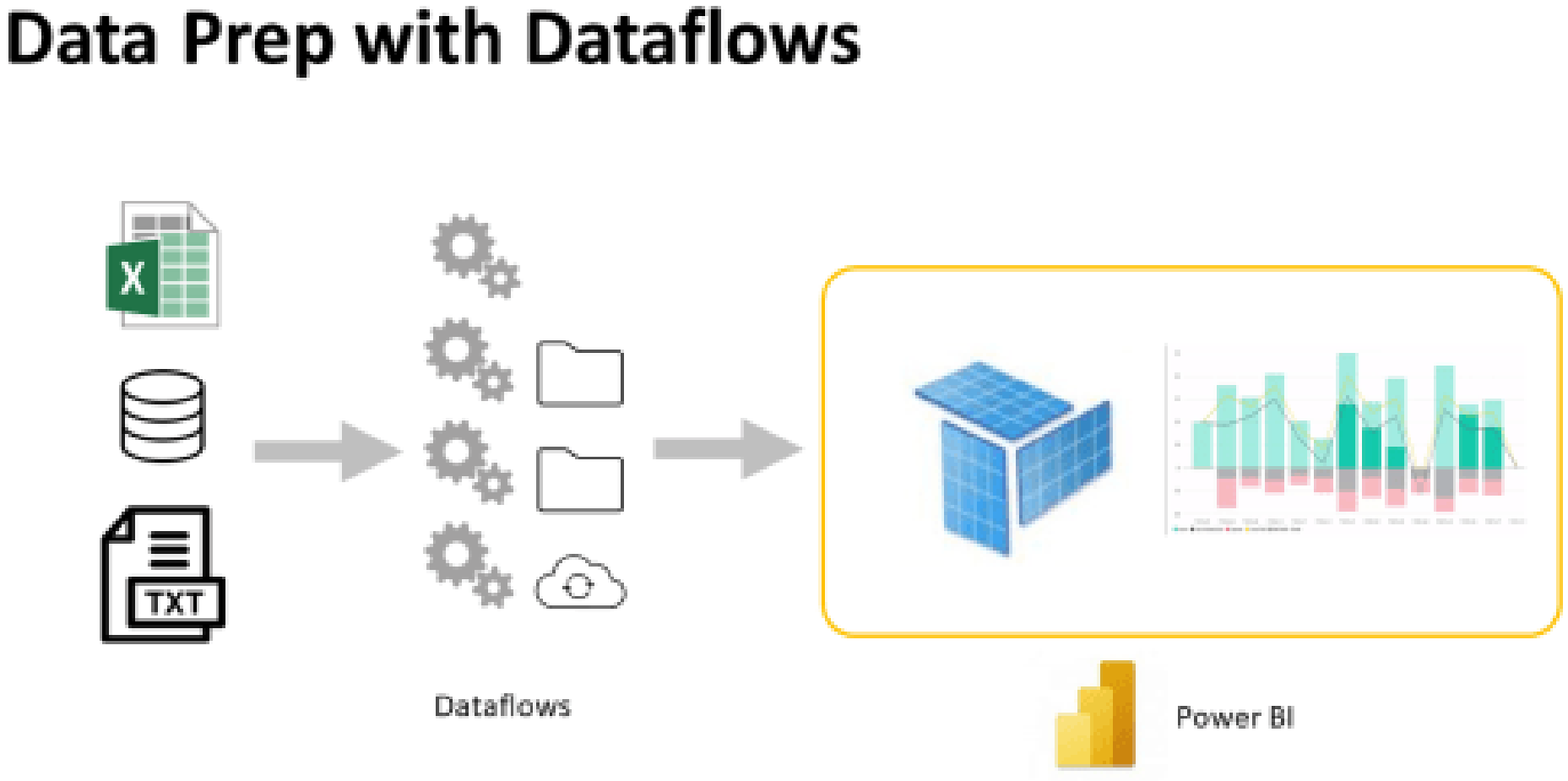
Architecture Pattern

Modern Data Warehouse

Architecture Pattern

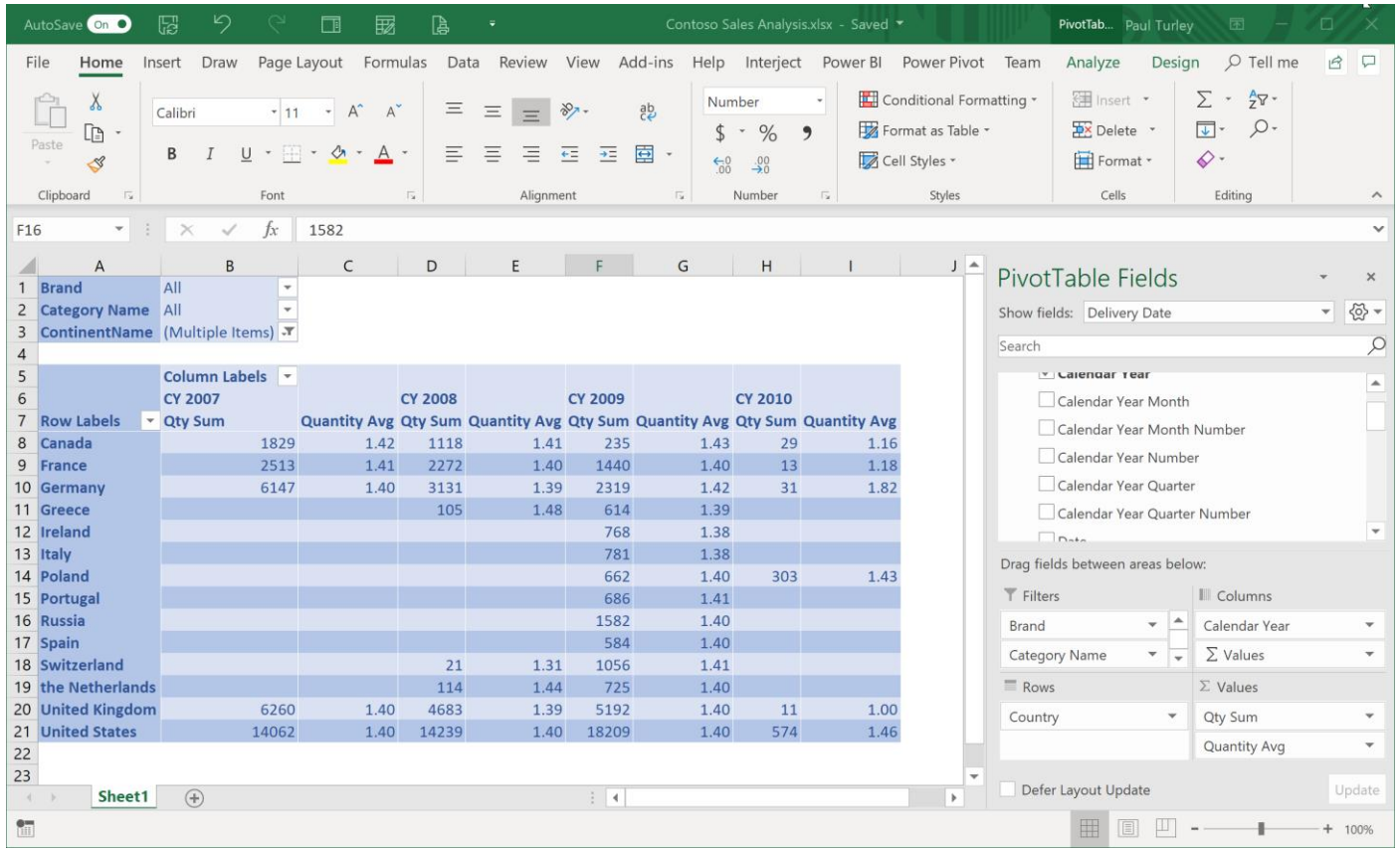


Architecture Pattern



If Users Need Excel, Give them Excel

- Teach analyst users how to use Excel with Power BI
- Don't "export", …"connect"
- "**Analyze In Excel**" allows Excel to connect, live, to a published Power BI dataset
- Now available to Power BI Pro & Free Premium licensed users
- Now available to "free" licensed users in a Premium



Version Control & Lifecycle Management

DATA MODEL

- Separate data model PBIX from Report PBIX
- Migrate large-scale or enterprise data models to Model.BIM file
- Manage enterprise models in Tabular Editor or Visual Studio

REPORT

- **Reality**: Integrating changes made to multiple PBIX files is difficult to manage and challenging to perform technically using current desktop tools. Even with differencing tools and effective version-control, just avoid multi-developer work on Power BI dataset files.
- Leverage Power BI's simplicity and don't over-engineer release management or version control. Until tooling exists to automate deployments, keep this is simple as possible.
- **File versions:**
  Store the dataset and report PBIX files separately using a version number postfix for the file name in the following the format: **Major**.**Minor**.**Revision**. For example:
  Manufacturing Cost Analysis Dataset v1.5.3.PBIX
- **Deployment Pipelines**

Promote Self-service Reporting

3Cloud

Non-governed Data

- Teach & support analyst users to use Power BI to acquire, mashup & model data
- “make mistakes, get messy”
  – Lilly Tomlin, Miss Frizzle
- Deploy to “user” designated workspaces
- User-authored solutions be used to prototype & pattern governed data models

Governed Data

- Publish to a secured & managed workspace
- Promote & Certify datasets
- Use dataflows for standardized common data models
- Enable users to connect to published datasets & create their own reports

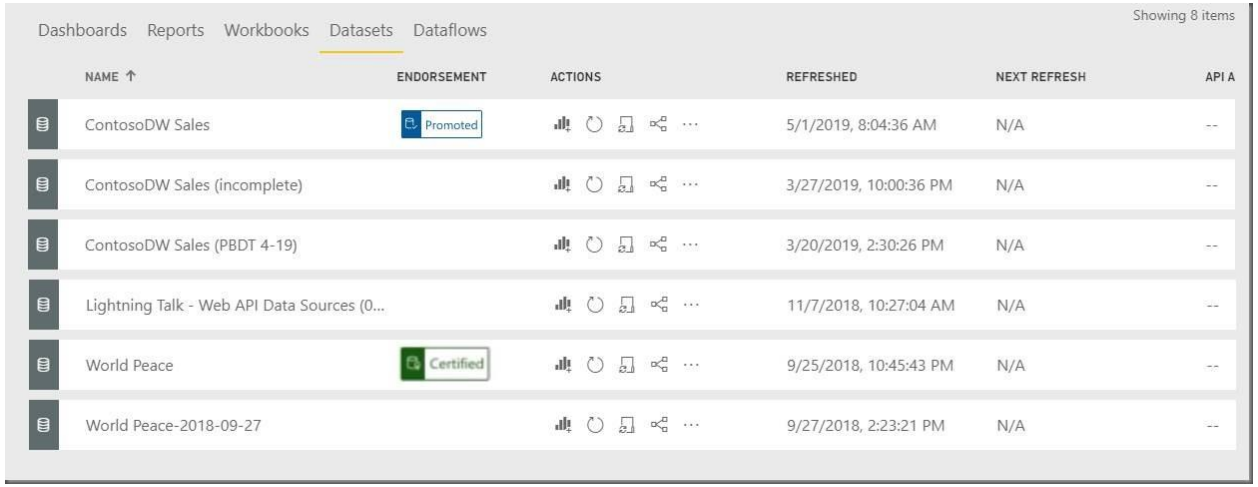
Storage Mode: Import vs DirectQuery

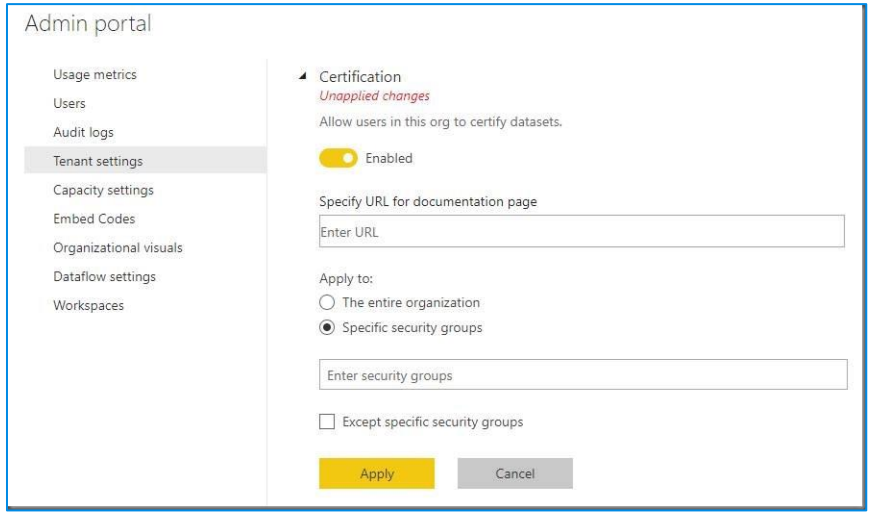
- Power BI is optimized to Import data into memory
- Uses columnar compression
- Highest performance

- DirectQuery doesn't copy data into model
- Usually slower & restrictive
- Hybrid/composite models enable both modes… drill to real-time data

Certified & Shared Datasets

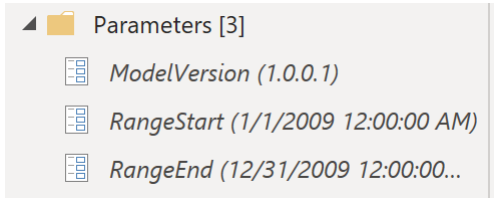
- Use Dataset endorsement & certification in the service
- Certification can be managed by security group
- Access to datasets can be restricted to certified datasets
- Organization defines certification policy & provides documentation

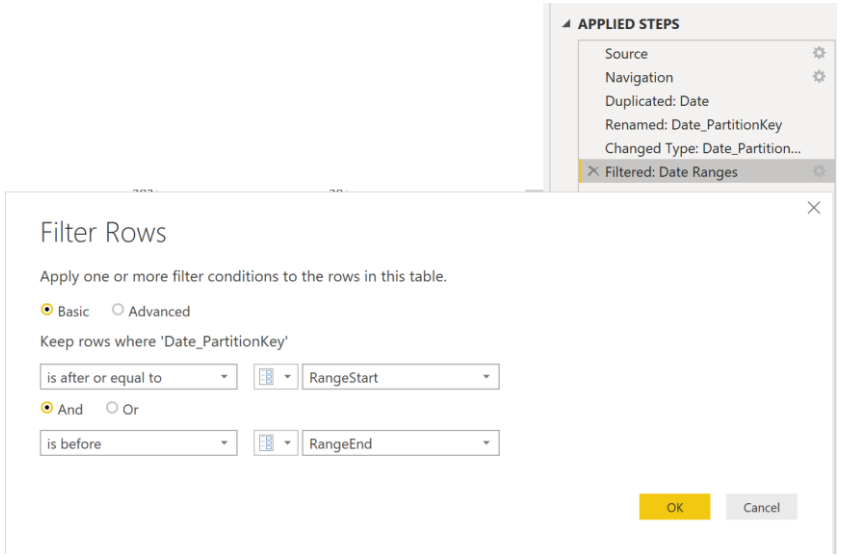


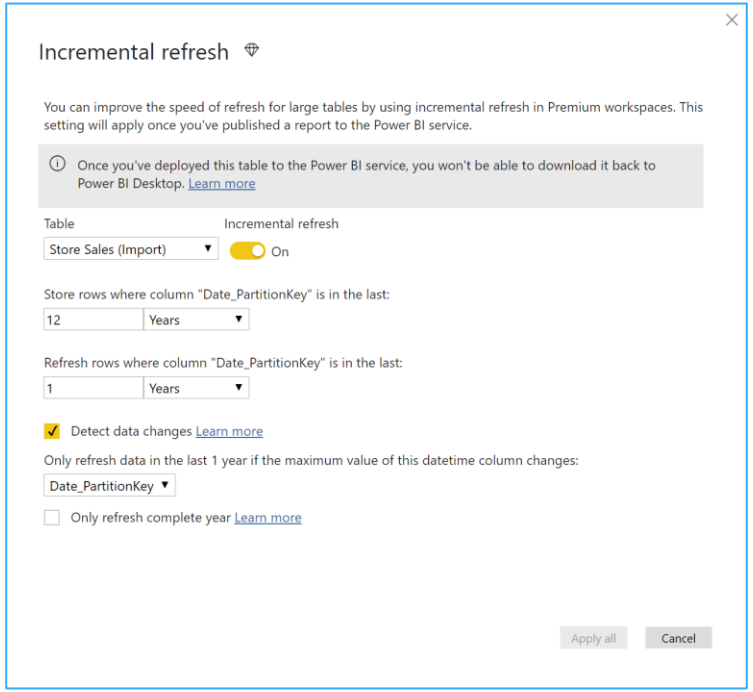


Managing Dataset Size with Parameters

- Use parameters whether implementing incremental refresh or not
- **RangeStart** & **RangeEnd** parameters must be date/time type
- Apply range filter on date/time column in Power Query







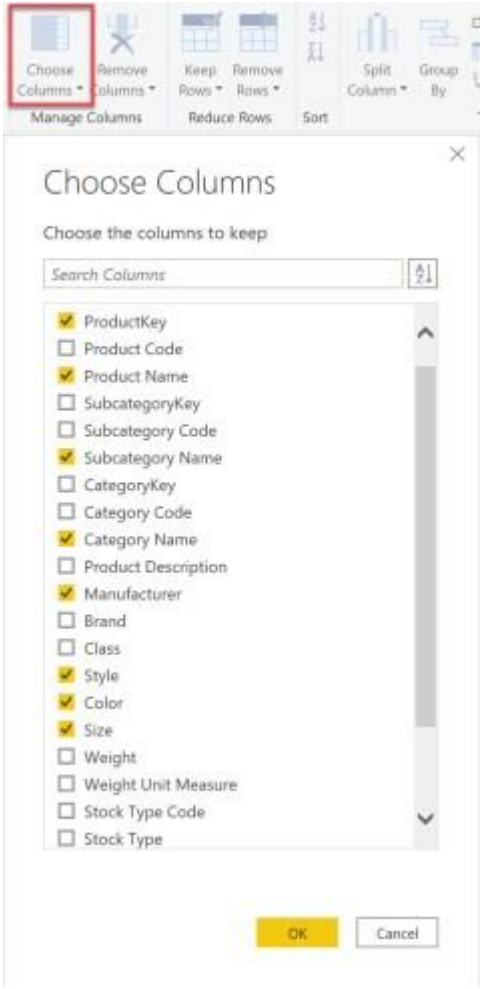
*Incremental Refresh is a Premium [capacity|user] feature but parameters may be used in any data model

Futureproofing Solutions

Difficult to see; always in motion, the future is.

- How is the source data procured and maintained?
- Who is responsible for maintaining source data?
- Who from the business defines the reporting requirements and signs-off on Power BI reports meeting those requirements?
- Who owns the development of the Power BI solution?
- Who is the developer's backup if they become unavailable?
- Are the requirements and project assets sufficiently documented to support such a transition?
- What are the security requirements?
- Who and how will users be given access to reports or dataset(s) for ad hoc analysis?
- Are users assigned to report and dataset permissions directly or through group membership?
- Should users have conditional or restricted access to data within the Power BI dataset?

Reduce Clutter – In Power Query

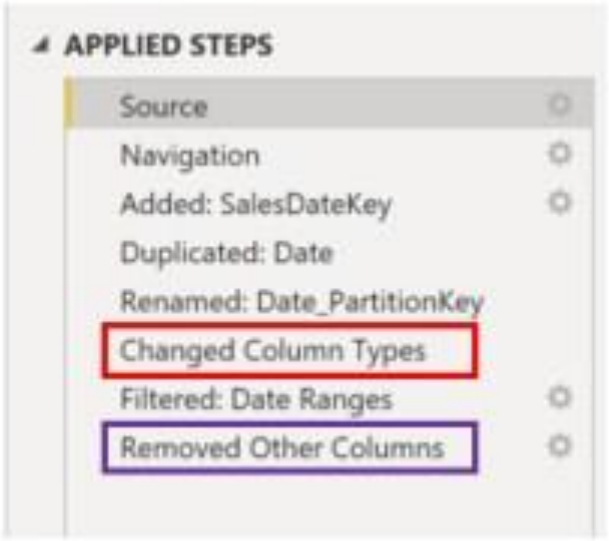


- Filter & parametrize queries
- Partition large tables
- Does It Fold?

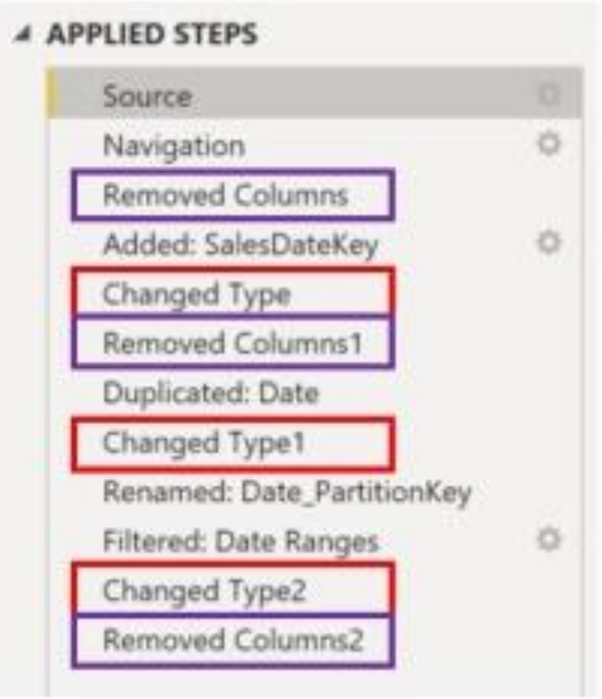
Optimize Power Query

- Test transformations with large data volumes
- Pivot, unpivot & Transpose actions are costly & may not work effectively with large data volume
- Complex and “creative” transformations might work in Desktop or with small data volumes but not in production
- Web service API calls & nested M functions may not work in the service.

Do this:

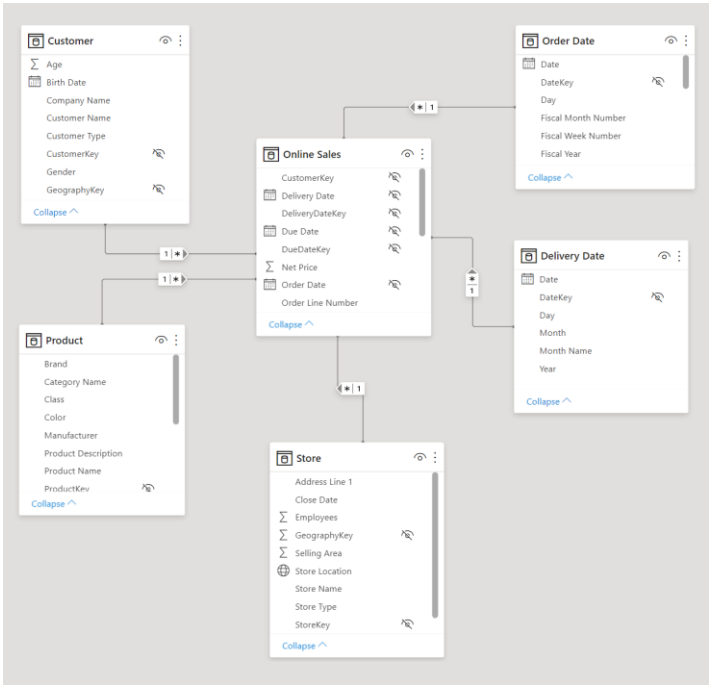


…not this:



Dimensionalize It!

	CUSTOMER	EMPLOYEE [RP]	CARRIER	PRODUCT	PRODUCT TYPE [RU]	SALES LOCATION	DELIVERY ADDRESS	WAREHOUSE	PROMOTION	PROBLEM REASON	ORDER ID [DD]	SHIP MODE	SHIPMENT NUMBER [DD]
	who				what		where			why & how			
SALES TARGETS **salesperson has product type target**		✓				✓	✓						
CUSTOMER ORDERS customer orders product	*	✓			✓	✓	✓	*			✓		*		
PRODUCT SHIPMENTS warehouse worker ships product	✓	✓	✓		✓	✓		✓	✓			✓	✓	*
CARRIER DELIVERIES carrier delivers product	✓		✓		✓	✓		✓				✓	✓	✓
CUSTOMER COMPLAINTS customer complains about product	✓	✓			✓	✓					✓	✓		
PRODUCT RETURNS customer returns product	✓				✓	✓			✓		✓	✓		



- Nearly any reporting data can be shaped into a star schema & optimized for analysis
- Ad hoc models never evolve into a dimensional model

Planning for Scale

How Much Data

Power BI Tabular Engine is optimized for in-memory scans & calculations

Data volume:

- Millions of rows ?
- Billions of rows ?

Data model size:

- 1 GB
- 10 GB
- 100 GB
- 400 GB
- Terabytes & Petabytes

Licensing Power BI

Power BI Pro	$10/user
Premium per User	$20/user
Premium Capacity (P1)	$5k/Month
A1 – A6	$735-$3k
P2 – P5	$10k-$80k

Model Size, Scale & Volume

Power BI Pro License	Premium Per User	Premium Capacity
1 GB	100 GB	Hybrid Data Models
	Premium Capacity	
	400 GB	No Stated Limits

Use:

- Import mode with parameters
- DirectQuery in composite models

[Doing Power BI the Right Way: 10.]
[Designing and Managing Large Datasets | Paul Turley's SQL Server BI Blog]

Planning for Separation

The Thick and Thin of Reports

Separate reports and data models can be:

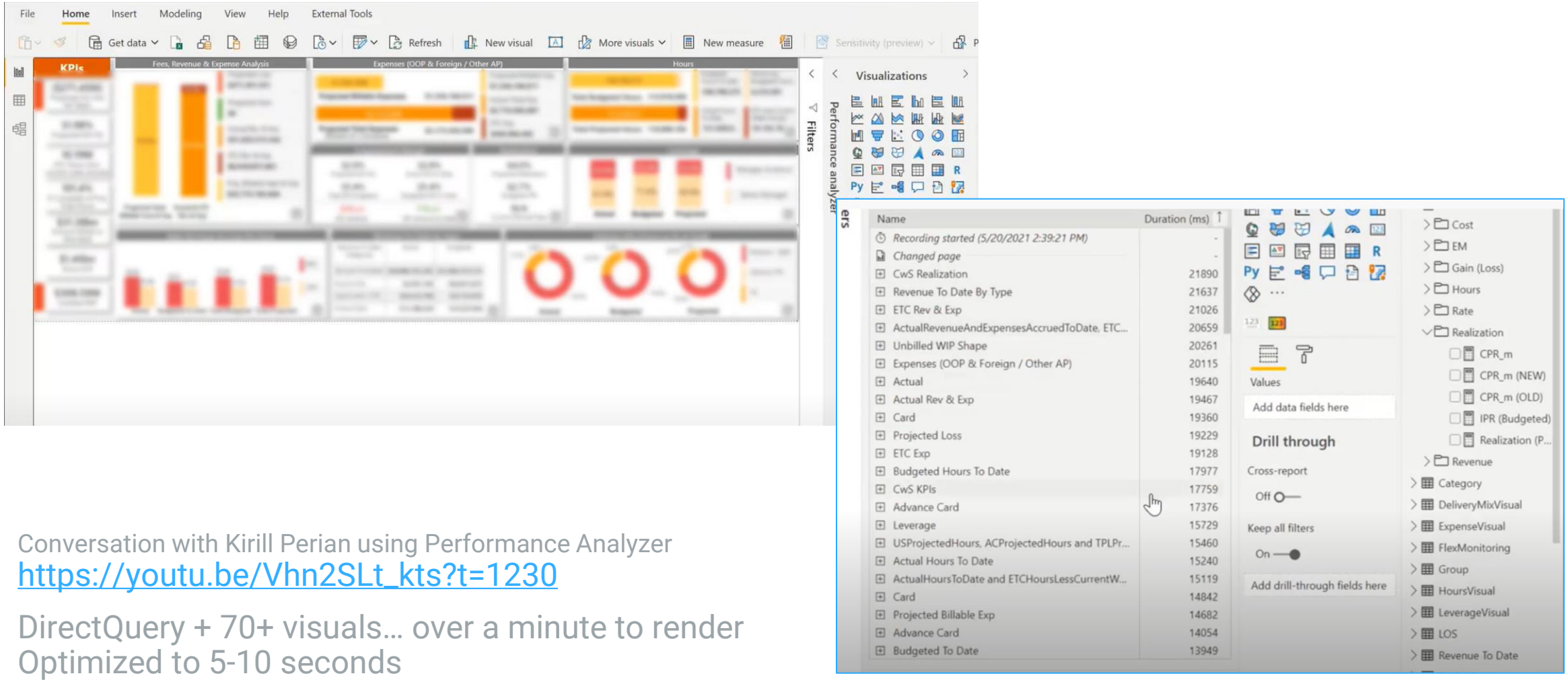
- Versioned
- Deployed & managed separately

[Doing Power BI the Right Way: 7.
Planning for separation – data models and reports | Paul Turley's SQL Server BI Blog]

Hot Swap tool demo: PowerBI.tips
[PowerBI.tips - Hot Swap Connections Webinar – YouTube]
https://www.youtube.com/watch?v=sygIUPMlgi0

Visualize With Care

Reduce Clutter - In Reports



Conversation with Kirill Perian using Performance Analyzer
https://youtu.be/Vhn2SLt_kts?t=1230

DirectQuery + 70+ visuals… over a minute to render
Optimized to 5-10 seconds

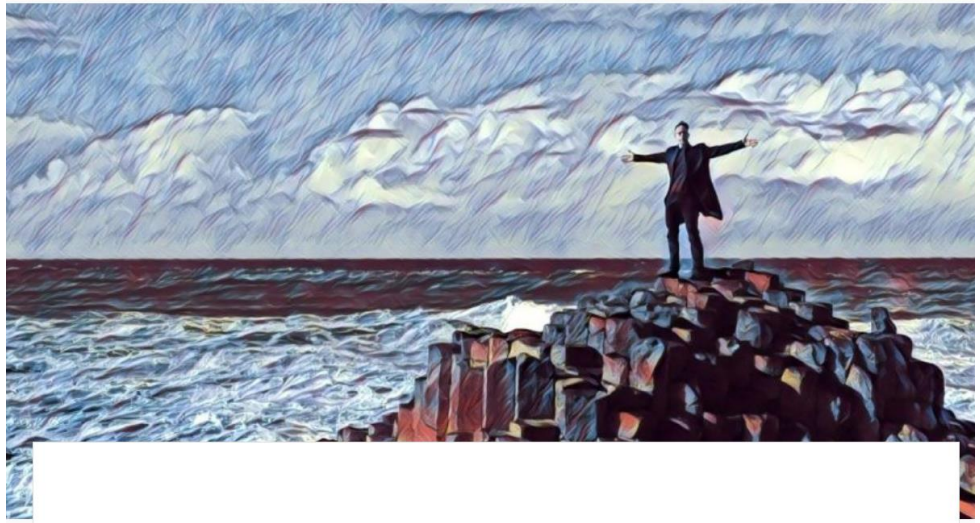
Data Governance & Data Culture

Blog and video series from Matthew Roche

Power BI CAT Team

[Building a data culture – BI Polar (ssbipolar.com)](https://ssbipolar.com/building-a-data-culture/)
https://ssbipolar.com/building-a-data-culture/

DataOnTheRoad.com – Interview with Matthew
https://youtu.be/Qkl-rpn0KSQ



Building a data culture

BI Polar has a series of videos and accompanying blog posts that focus on key aspects building a data culture. This series wrapped up in November 2020, and includes 3 hours 20 minutes of content delivered over 17 videos.

1. Series Intro: Building a Data Culture
2. Data Culture: Executive sponsorship
3. Data Culture: A brief history of business intelligence
4. Data Culture: Roles and responsibilities
5. Data Culture: Picking your battles
6. Data Culture: The importance of community
7. Data Culture: Motivation and encouragement
8. Data Culture: Training for the community
9. Data Culture: Showcasing the art of the possible
10. Data Culture: The importance of a central portal
11. Data Culture: Making stakeholder buy-in explicit
12. Data Culture: Every app is a unique snowflake
13. Data Culture: Community champions with swords
14. Data Culture: The importance of experts
15. Data Culture: Measuring success
16. Data Culture: Wisdom from Sun Tzu
17. Data Culture: Wrapping up and closing words

Development & Release Process

- ✓Separate models (datasets) from reports
- ✓Deploy and manage separately
- ✓Manage data model with Tabular Editor
- ✓Use Deployment Pipelines for Apps

Resources

Enterprise Scale Options

In many ways, Power BI has now surpassed the capabilities of SQL Server Analysis Services. Microsoft are investing in the enterprise capabilities of the Power BI platform by enhancing Power BI Premium Capacity, adding Paginated Report and features to support massive scale specialized use cases. Consider the present and planned capabilities of the Power BI platform; before, choosing another data modeling tool such as SSAS.

Resources:

https://sqlserverbi.blog/2018/07/27/power-bi-for-grownups

https://sqlserverbi.blog/2018/12/13/data-model-options-for-power-bi-solutions

Master Project Preparation Checklist

Solution Audience:

- [ ] Categorize the solution by identifying the author & user roles related to the project:
- [ ] Author role: Business Data Analyst
- [ ] Author role: Skilled Data Modeler, Analyst, Data Scientist
- [ ] Author role: IT BI Developer
- [ ] Users' role: Report/Dashboard Consumer
- [ ] Users' role: Self-service Report Author
- [ ] Users' role: Advanced Data Analyst

Training and Usability Support:

- [ ] Develop & Document Support & training plan for users:
  - [ ] Usability training for read-only report/app users
  - [ ] Self-service reporting for Novice Report Authors & Data Analysts

Solution Type & Architecture:

- [ ] Identify the Solution Type for the project. This will guide other project management designs:
- [ ] Design single PBIX file for small group, departmental project authored by one developer for a limited group of users
- [ ] Design & deploy a separate dataset PBIX file – from report file(s) – when the dataset should be branded as a Certified dataset
- [ ] Design separate dataset and report PBIX files for formal projects with more than one dataset & report developer, to coordinate work
- [ ] Use SSAS/AAS as a data modeling option when those databases exist or where IT operations insist o management development and maintenance through integrated source control (e.g. Visual Studio Team Services & Azure DevOps)
- [ ] Identify the Project Type & related Solution Architecture:
  - [ ] Project type: Formal project
  - [ ] Project type: Informal project
  - [ ] Project type: Hybrid project
  - [ ] Architectural approach: Single PBIX
  - [ ] Architectural approach: Separate dataset and report PBIX
  - [ ] Architectural approach: Report PBIX connected to SSAS or AAS
- [ ] Understand DirectQuery model trade-offs and special use cases. Avoid if possible.
- [ ] Define your Release Management, DevOps & Automation strategy (if any – Might be OK to deploy files manuall, to automate or not to automate)

File & Workspace Management:

- [ ] Create storage locations and folder structure for Development file management:
- [ ] Development file storage
- [ ] Team member collaboration environment & processes
- [ ] Folder synchronization
- [ ] Define File naming standards
- [ ] Decide on dataset and report names
- [ ] Define the Version Control & Lifecycle Management:
  - [ ] Postfix files with 3-part version number
  - [ ] Remove version number from published files in QA and PROD
  - [ ] Create Version History table in Power Query
  - [ ] Increment version numbers in data model
  - [ ] Backup PBIT files for archive
  - [ ] Create measures: Last Refresh Date/Time
  - [ ] Create measure: Current Version
  - [ ] Add data model info page to report
- [ ] Decide on Workspace and App Management, workspace & app name, etc.:
  - [ ] Create PROD workspace (omit PRD from name), assign dedicated capacity if available.
  - [ ] Create QA workspace (post-fix name with QA), assign dedicated capacity
  - [ ] (optionally) Create DEV workspace (postfix name with DEV), dedicated capacity not required (or combine with QA workspace).

Assign licenses and access:

- [ ] Assign Pro licenses to all developers, admins and report author users (QA?)
- [ ] Assign Free licenses to all users if Premium/app deployment will be used
- [ ] Assign membership and access to workspaces

Query Design:

- [ ] Create fact date range filter parameters: RangeStart & RangeEnd to reduce volume in PBIX file under 400 MB.
- [ ] Filter large fact tables with range filters, consider incremental refresh policies if slow and/or over 800 MB compressed.
- [ ] Design source queries (T-SQL?) to reshape source data into conformed dimension & fact tables
- [ ] Create views in database for each dimension and fact
- [ ] Enforce key uniqueness to remove all duplicate keys from all dimension tables
- [ ] Query Date dim/lookup table at source if it exists
- [ ] If not available, generate Date dim/lookup table in Power Query

Data modeling:

- [ ] Build star schemas
- [ ] Enforce dimension key uniqueness
- [ ] Avoid bi-directional filters & unnecessary bridging tables
- [ ] Consider using DAX measures rather than complex & inefficient relationships
- [ ] Create custom columns in Power Query
- [ ] Annotate code
- [ ] Hide all fields not used directly by users
- [ ] Use friendly field names
- [ ] Set to Do Not Summarize

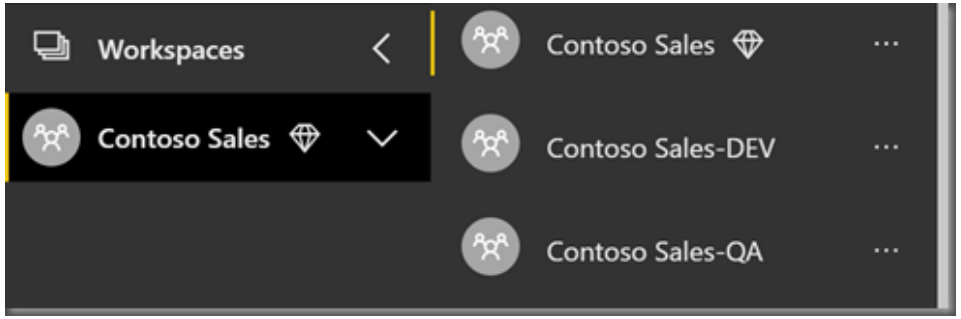
Query Optimization

- ☐ **Decide**: Perform column transformations in ETL, database view, Power Query or DAX?
- ☐ **Decide**: How is process managed & governed? who maintains the query?
- ☐ Avoid using SQL statements in PQ queries. Use database views.
  Views and tables support query folding. SQL statements generally do not.
- ☐ Remove unnecessary columns & filter rows early in the query
- ☐ Consolidate field renaming, removing fields and data type changes
- ☐ Add custom columns in Power Query instead of calculated columns in DAX, where possible
- ☐ Use friendly field names for all fields that won't be hidden in the data model
- ☐ Rename steps and add annotations in M script

Workspace and App Management

For a formal project, create the following workspaces:

- ☐ **DEV Workspace** - Only development team members need Contributor access to this workspace. This workspace does not need to have Premium capacity; unless, developers need to unit test incremental refresh or other Premium features.
- ☐ **QA Workspace** - All testers must have View access for testing and Contributor access for report authoring. Should be in Premium capacity to test incremental refresh.
- ☐ **PROD Workspace** - Omit the “PROD” designation in the name. This workspace will be the name of the published app that users will see in their Apps, Home and Favorite pages so use a name that is simple and sensible. Must have Premium capacity to share the app with non-Pro licensed users.



Power BI Licensing Plan Checklist

Capacity and platform:

Shared capacity service:

- [ ] Assign user licenses

Dedicated capacity:

- [ ] Are Premium features required?
- [ ] Is dedicated capacity needed?
- [ ] Is Premium more cost-effective than licensing all users?

On-premises server:

- [ ] SQL Server Enterprise + SA, or:
- [ ] Premium license

Assign user licenses and access:

- [ ] Assign Pro licenses to all developers, admins and report author users
- [ ] If Premium, use app deployment & assign Free licenses to all users
- [ ] Assign membership and access to workspaces

Managing Power BI Desktop Files

- **Store in a centrally managed network-assessable folder**
  The storage folder should support automatic backup and recovery in the case of storage loss.
- **Report and dataset developers must open files from the Windows file system**
  Files must either reside in or be synchronized with the Windows file system.
- Files containing imported data typically range in size from 100 to 600 MB. Any shared folder synchronization or disaster recovery system should be designed to effectively handle multiple files of this size.

Options:

- OneDrive For Business (shared by team, with folder synchronization).
- SharePoint or SharePoint Online (with folder synchronization).
- GitHub and/or VSTS with local repository & folder synchronization. If used, Git must be configured for large file storage (LFS) if PBIX files are to be stored in the repository.

Data model & Report Architecture

3Cloud

Choose dataset architecture:

- ☐ **Single PBIX file**
  For small group, departmental project authored by one developer for a limited group of users
- ☐ **Separate dataset and report PBIX**
  Design & deploy a separate dataset PBIX file – from report file(s) – when the dataset should be branded as a Certified dataset.
  For formal projects with more than one dataset & report developer, to coordinate work
- ☐ **SSAS/AAS as a data modeling option**
  when those databases exist or where IT operations insist on managing development and maintenance through integrated source control (e.g. Visual Studio Team Services
  & Azure DevOps)
- ☐ **Migrate PBIX** to **Model.BIM**

Model Design Checklist

- [ ] **Model for the user experience, not for developers**
- [ ] **Build star schemas**
  Wherever possible, reshape data into fact a dimension tables with single key, one-to-many relationships from dimensions to fact.
- [ ] **Enforce dimension key uniqueness**
  Just because a key value “should” be unique, there is no guarantee that it will be unless enforced at the data source. Perform grouping and duplicate reduction in the data source views or Power Query queries to guarantee uniqueness. Duplicate record count checks and other mechanisms can be applied to audit source data for integrity but do not allow the data model to violate these rules.
- [ ] **Avoid bi-directional filters & unnecessary bridging tables**
  These data modelling patterns adversely affect performance.
- [ ] **Consider using DAX measures** rather than complex & inefficient relationships
- [ ] **Create custom columns in Power Query**
  Rather than DAX calculated columns wherever possible for row-level derived columns. This maintains a consistent design pattern for maintainability.
- [ ] **Annotate code**
  Use in-line comments and annotations in all code including SQL, M and DAX; to explain calculation logic and provide author and revision information.
- [ ] **Remove all unused fields** – if in doubt, take it out
- [ ] **Hide all fields not used directly by users**
  primary and foreign key columns, numeric columns used to create measures, and columns used to specify the sort order of other fields.
- [ ] **Use friendly field names**
  Rename all visible columns (in Power Query) to short but user-friendly names with mixed case and spaces.
- [ ] **Set to Do Not Summarize**
  Any non-hidden numeric columns that are not intended to roll-up or summarize values. Columns set to summarize are indicated with a Sigma icon.

Model Design Guidelines

- Dimensional design concepts haven't changed in 20 years & are as true as ever
- Dimensional modeling "rules" should be followed but can be relaxed for Power BI in certain cases, such as:
  - Leaving some dimensional attributes in fact tables
  - Use natural keys rather than generating surrogate keys
- The art of dimensional modeling ranges from simple to complex. Start with the basics.
- Flattened "spreadsheet" models are OK for small, informal projects but have significant limitations
- As models grow in size & complexity, data quality challenges will surface that can be solved by implementing proper governance controls

The Kimball Method: https://www.kimballgroup.com/data-warehouse-business-intelligence-resources/kimball-techniques/dimensional-modeling-techniques

Lawrence Corr, Model Storming Agile method: https://modelstorming.com/hierarchy-map

Report Types

Dashboard & Scorecard style reporting

- [ ] Infographics
- [ ] KPIs & scorecards
- [ ] Segmented comparisons
- [ ] Time-series trends

Statistical & Scientific analysis

- [ ] Deviations & percentiles
- [ ] Forecast trends & predictions
- [ ] Scatter plots
- [ ] Population analysis

Financial balances & worksheets

- [ ] Cost accounting & balance sheets
- [ ] General ledger
- [ ] Accounts receivable & payable
- [ ] Invoices
- [ ] Forms & lists


Questions?