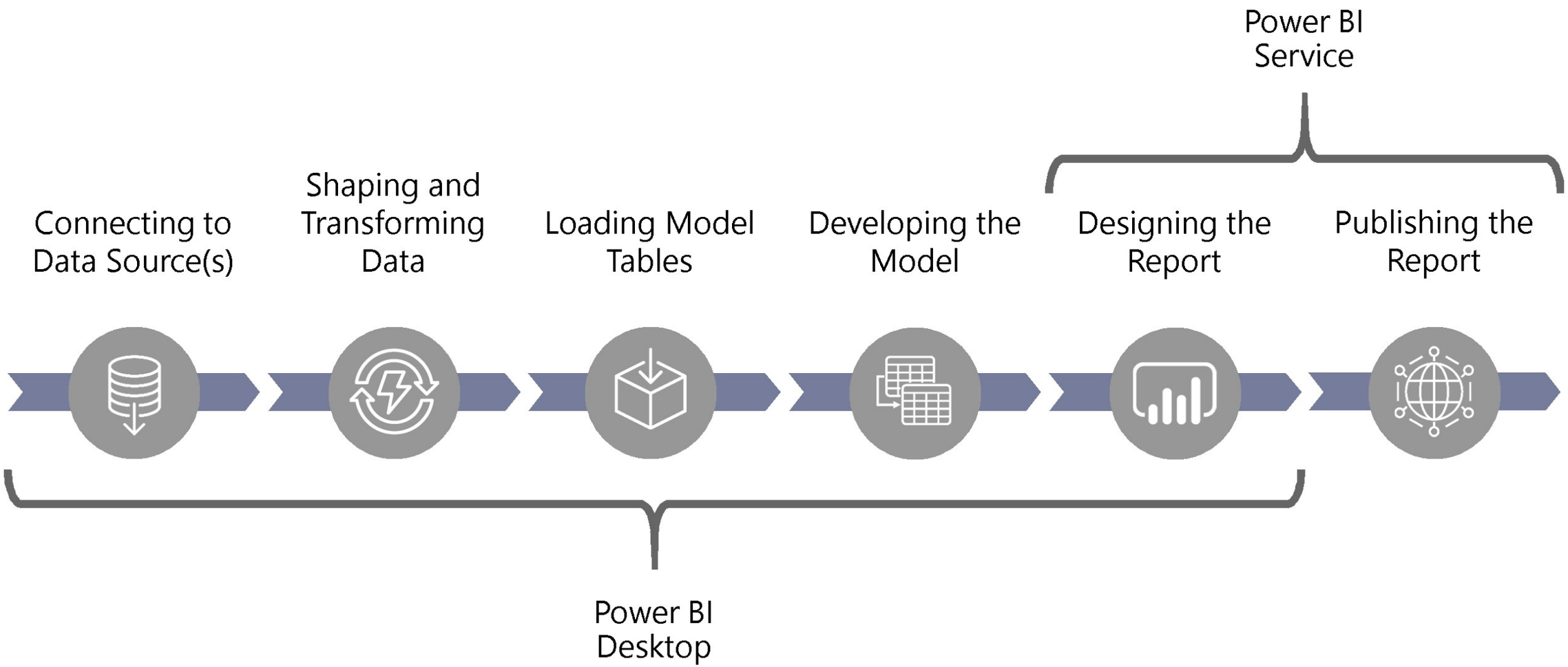
Defining the Development Methodology

Report Development Lifecycle



Introducing Power BI Desktop

- Power BI Desktop produces a single file that has a **PBIX** extension
  - The file is intended to be published to the Power BI Service

PBIX File

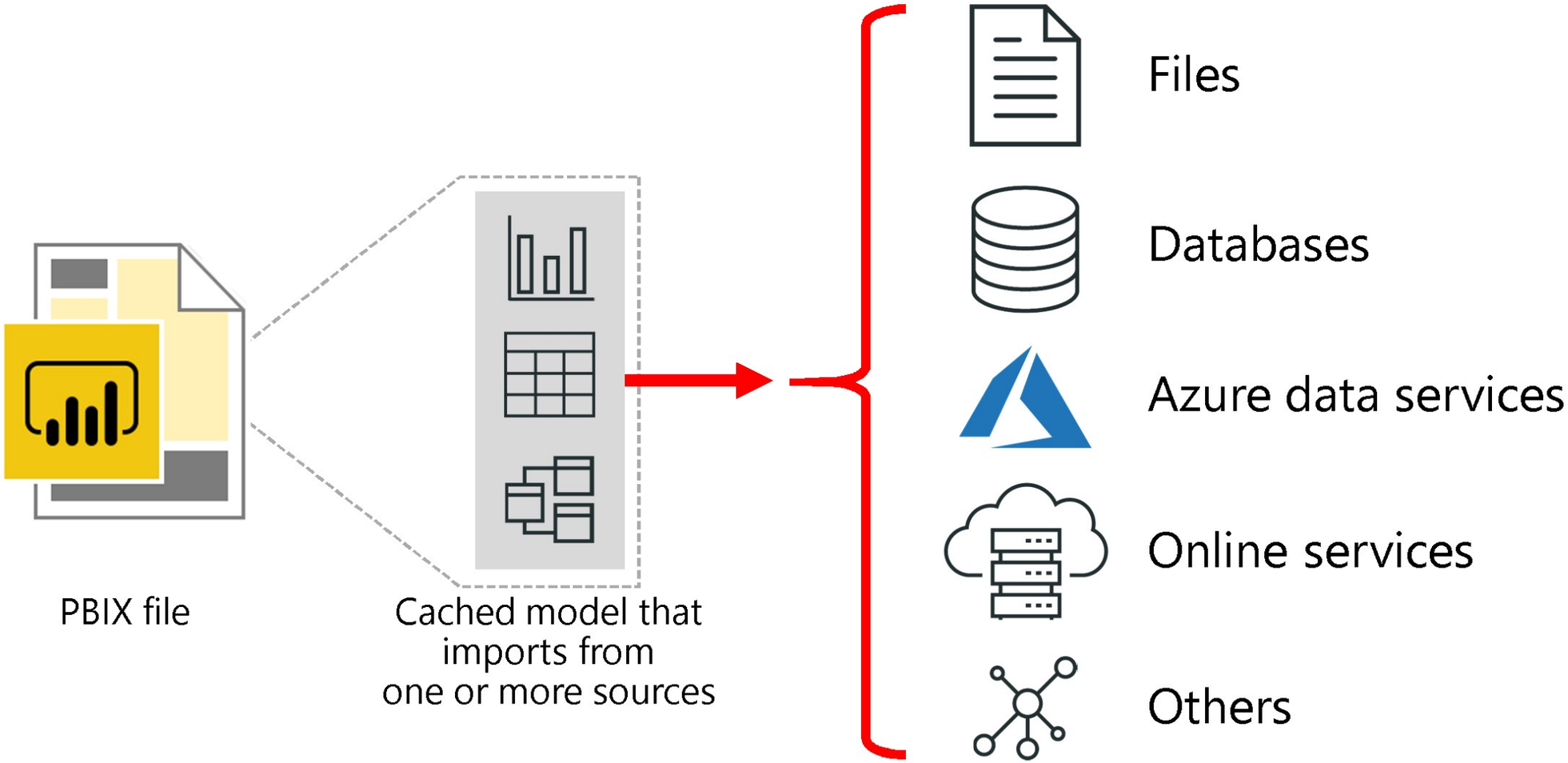
Data Model

Power BI Report

Power BI Service

Defining the Development Modes

Modes of Power BI Desktop ► Import



Defining the Development Methodology

Model Audience

- Understanding the *report* and *model audience* is an important component of the *development lifecycle* – common audiences include:

Yourself

- Developer and consumer are the same person
- **Considerations**:
  - Future support for additional authors or consumers?
  - Build access for ad-hoc reports?[1]

Report Authors

- Users who consume, edit, and/or create reports
- **Considerations**:
  - Build access for ad-hoc reports?[1]
  - PBIX file save permissions?
  - File version history?
  - Change management?
  - Documentation?

Report Consumers

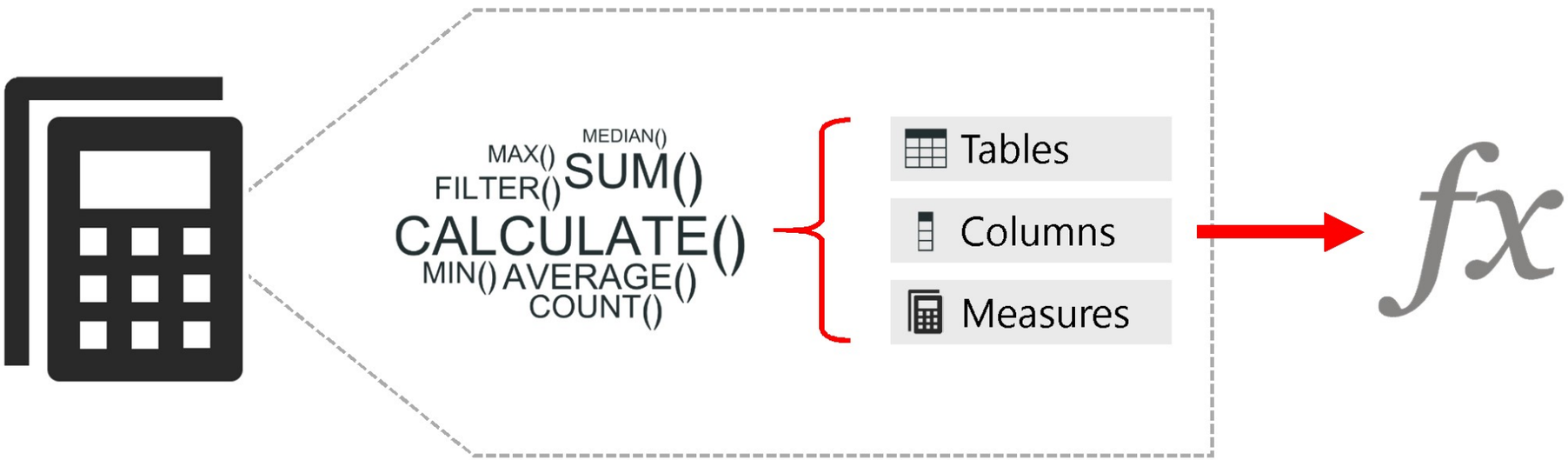
- Users who primary consume solutions
- **Considerations**:
  - Build access for ad-hoc reports?[1]
  - PBIX file save permissions?
  - Documentation?

1: Sources for ad-hoc reporting include: other **PBIX** files, **Excel** files, **XMLA** endpoint connections, etc.

Logic and Syntax

Formula Fundamentals

- Commonly, DAX formulas contain one or more **Functions**:
  - Functions can reference <u>three</u> different model objects to evaluate



DAX Formula | Functions | References | Evaluated Result

Measure Configuration

Measure Properties

General

- **Name**
  - Must be unique within the table
- **Synonyms**
- **Description**
- **Display folder**
- **Is hidden**

Formatting

- **Format**
- **Data type**
- **Currency**
- **Percentage format**
- **Thousands separator**
- **Decimal places**

Advanced

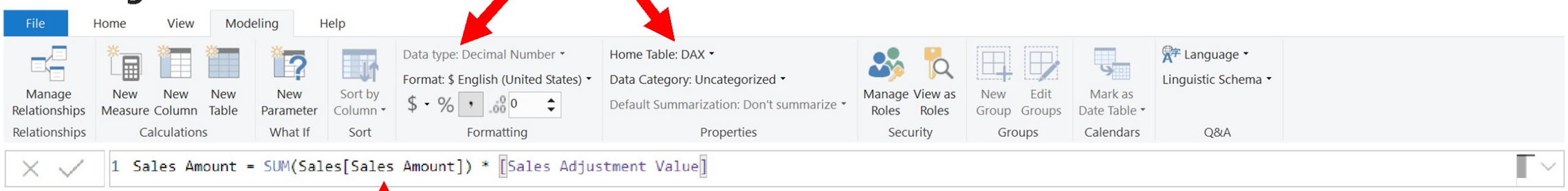
- **Data category**
  - Same behavior as for columns

Configuring Measures

Measure Properties ► Modeling Ribbon

Measure configurations can also be *accessed from here*

Modeling ribbon



Formula bar displays the DAX measure

The **Model** ribbon contains the most common configuration options

Filter Propagation

Relational Data Modeling

- Commonly, DAX measures use relationships to **propagate filters** from *dimension* to *fact tables* when evaluating results

1

Product ID	Product Name	Product Sub Category	Product Category
1	Road Bike 150	Road Bikes	Bikes
2	Mtn Bike 370	Mountain Bikes	Bikes
3	2L Water Bottle	Bike Gear	Accessories

Sum of Road Bike 150 Sales = **$1,094.34**

*

Order ID	Product ID	Amount
1	1	$364.78
1	2	$519.85
1	3	$6.94
2	1	$364.78
2	3	$6.94
3	2	$519.85
3	1	$364.78
4	3	$6.94
4	2	$519.85

Representative of a typical *one-to-many* relationship between a fact and dimension table

Iterator Functions

Examples

- Calculating average sales per day cannot be achieved using **AVERAGE()** – as it will *not evaluate* at the correct row context

```
Average Sales Per Day = AVERAGE( [Sales Amount] )
```

Order ID	Date	Amount
1	1/1/2019	$214.78
2	1/1/2019	$519.85
3	1/2/2019	$6.94
4	1/2/2019	$364.78

Physical table

Each row is evaluated → AVERAGE() of Amount = **$216.59**

Iterator Functions

Examples

- Calculating average sales per day cannot be achieved using **AVERAGE()** – as it will *not evaluate* at the correct row context

```
Average Sales Per Day = AVERAGE( [Sales Amount] )
```

- Using **AVERAGEX()** allows the sales amount to be calculated for *each day first* – before averaging those evaluated results

```
Average Sales Per Day = AVERAGEX( VALUES( 'Calendar'[Date] ), [Sales Amount] ) )
```

Order ID	Date	Amount
1	1/1/2019	$214.78
2	1/1/2019	$519.85
3	1/2/2019	$6.94
4	1/2/2019	$364.78

Physical table

Date	Amount
1/1/2019	$714.63
2/1/2019	$371.72

Virtual table

Each row is evaluated → AVERAGE() of Amount = **$543.17**

Filter Context

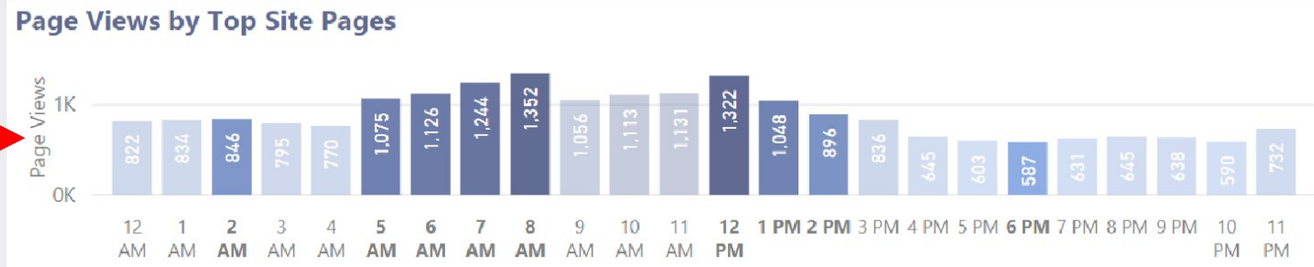
Query Context Example

Cross-filter

Visual (column) filter

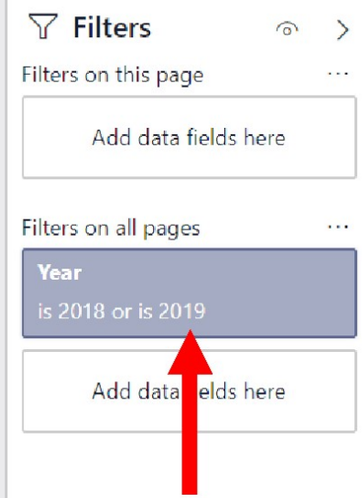
Slicer

Visual (row) filter



- Select all
- Australia
- Belgium
- Brazil
- Canada
- France
- Germany
- Hong Kong
- India
- Italy
- Netherlands
- Norway
- Pakistan
- Poland
- South Africa
- Spain
- United Arab Emirates
- United Kingdom
- United States
- Finland

		Q1 2019	Q2 2019	Q3 2019	Total
Americas	United States	310	465	1,197	1,710	523	**4,205**
	Canada	28	89	116	202	91	**526**
	Total	**338**	**554**	**1,313**	**1,912**	**614**	**4,731**
Europe	United Kingdom	43	86	189	483	95	**896**
	Germany	12	15	52	223	48	**350**
	Spain	15	9	29	90	45	**188**
	Italy	1	3	24	77	18	**123**
	France	4	18	29	58	11	**120**
	Total	**75**	**131**	**323**	**931**	**217**	**1,677**
Asia	India	54	29	63	121	37	**304**
	Total	**54**	**29**	**63**	**121**	**37**	**304**
Oceania	Australia	11	30	69	132	26	**268**
	Total	**11**	**30**	**69**	**132**	**26**	**268**
Africa	South Africa	12	5	30	57	16	**120**
	Total	**12**	**5**	**30**	**57**	**16**	**120**
Total		**490**	**749**	**1,798**	**3,153**	**910**	**7,100**



Report-level filter

Power Query Introduction

Data Preparation Process

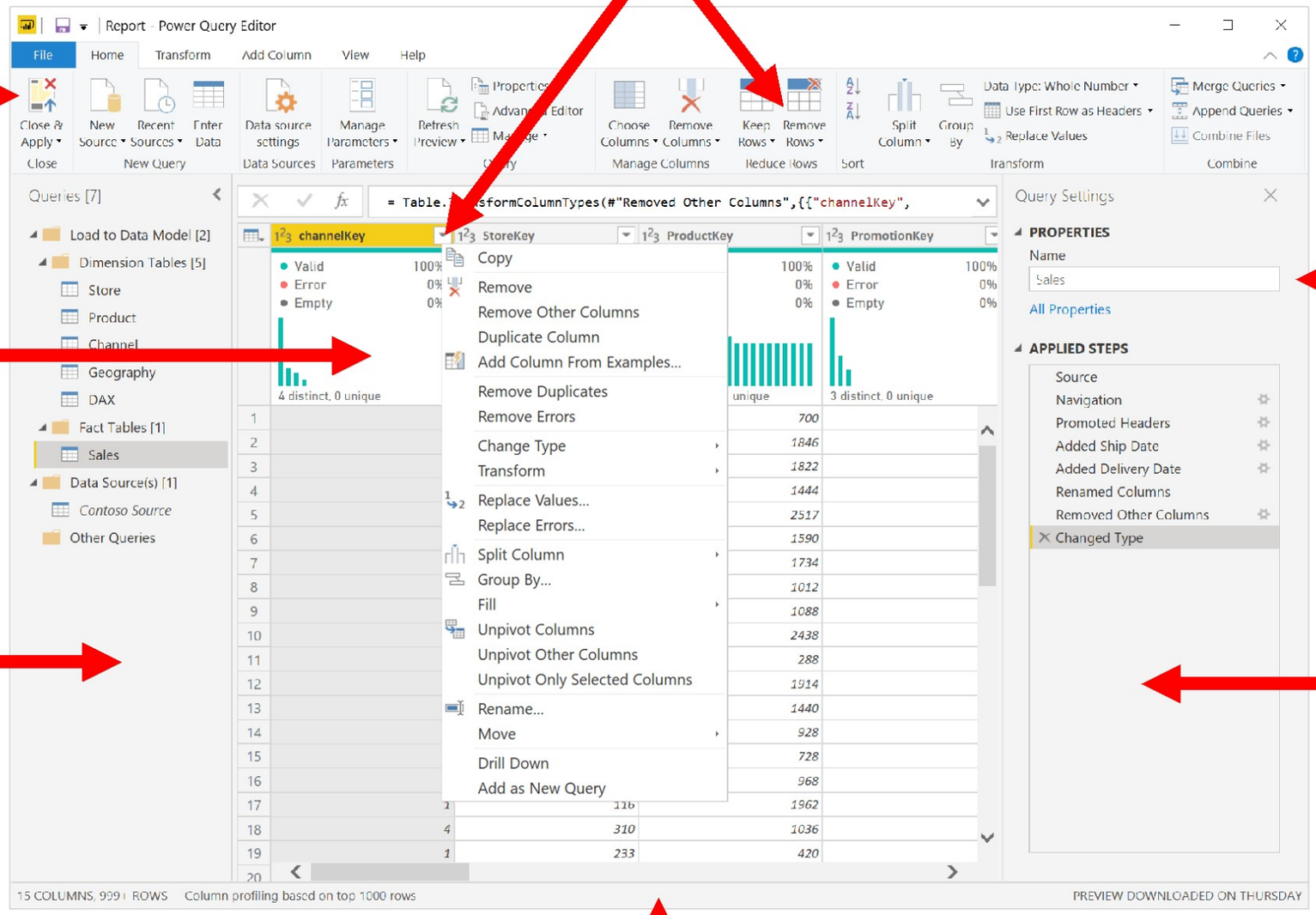
- Three stages of data preparation:
  - **Extracts** data from a variety of sources and locations
  - **Transforms** data through a series of automated transformations
  - **Loads** optimized data into the data model

The *ribbon* and *dropdown* menus provide access to most transformations

Applying processes Queries to load the model

Column Profiling provides info and statistics

The *Queries pane* contains a list of all Queries



Each Query has a *unique name*

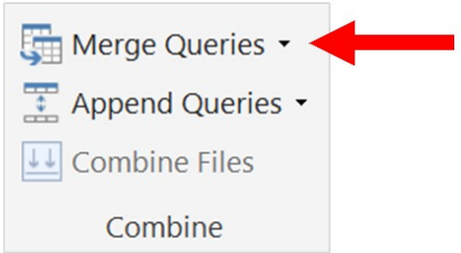
Any Query modifications get added to the list of *applied steps*

The *status bar* displays information about the Query

Transforming Data

Common Transformations ► Combining Queries

- **Merge Queries**:
  - Joins two Queries using common key column(s), and introduces column(s) to the original Query



Product ID	Product Name	Cat ID
1	Road Bike 150	1
2	Dirt Bike 820	2
3	Bike Seat 74	3

+

Cat ID	Category Name
1	City Bikes
2	Off road Bikes
3	Accessories

=

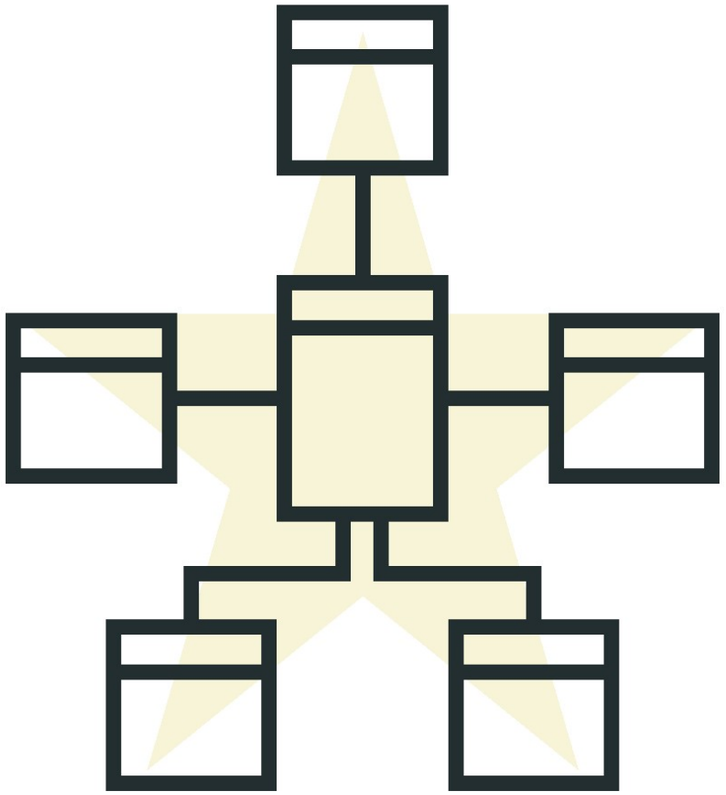
Product ID	Product Name	Cat ID	Category Name
1	Road Bike 150	1	City Bikes
2	Dirt Bike 820	2	Off road Bikes
3	Bike Seat 74	3	Accessories

There are **multiple join types** available when merging Queries

Describing Modeling Design Patterns

Star Vs. Snowflake Schemas

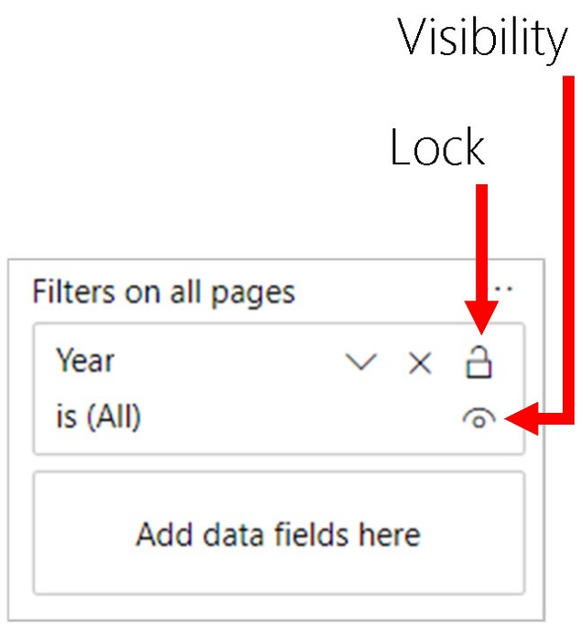
- Data model designs fall into *two primary patterns*:
  - **Star schema**
  - **Snowflake schema**[1]
- These schema originated from *traditional relational databases* – and whose principles *still apply* to Power BI data models
- Commonly, star schemas *perform better* and are considered *best practice* in Power BI



1: Snowflake schemas are outside the scope of this course

Developing Reports

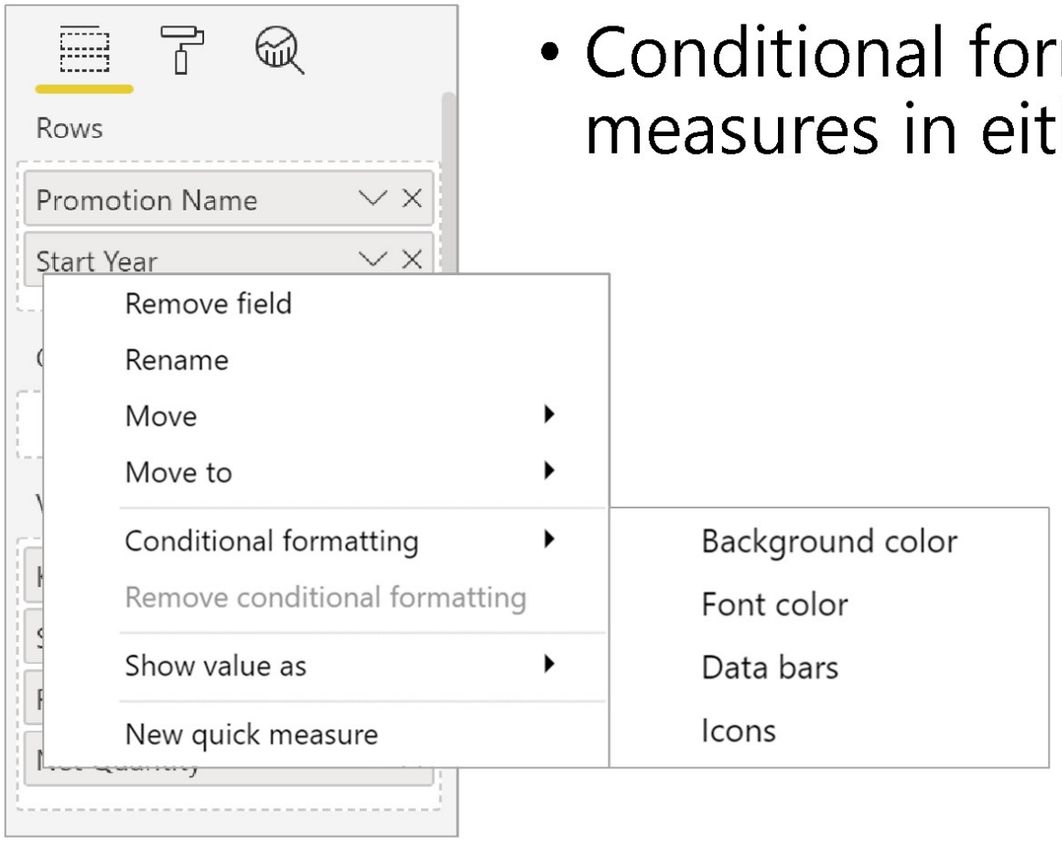
- Filters can also be configured to remove user interactions:
  - **Locked** – where users cannot *edit* or *remove* the filter
  - **Hidden** – where users cannot *see the filter* in the report experience
- **Other filter configurations:**
  - Rename
  - Sort alphabetically[1]
  - Sort manually (drag-and-drop)[1]



1: Sorting applies to slicers placed in the *visuals*, *page*, or *report* wells respectively

Designing Advanced Reports

Conditional Formatting



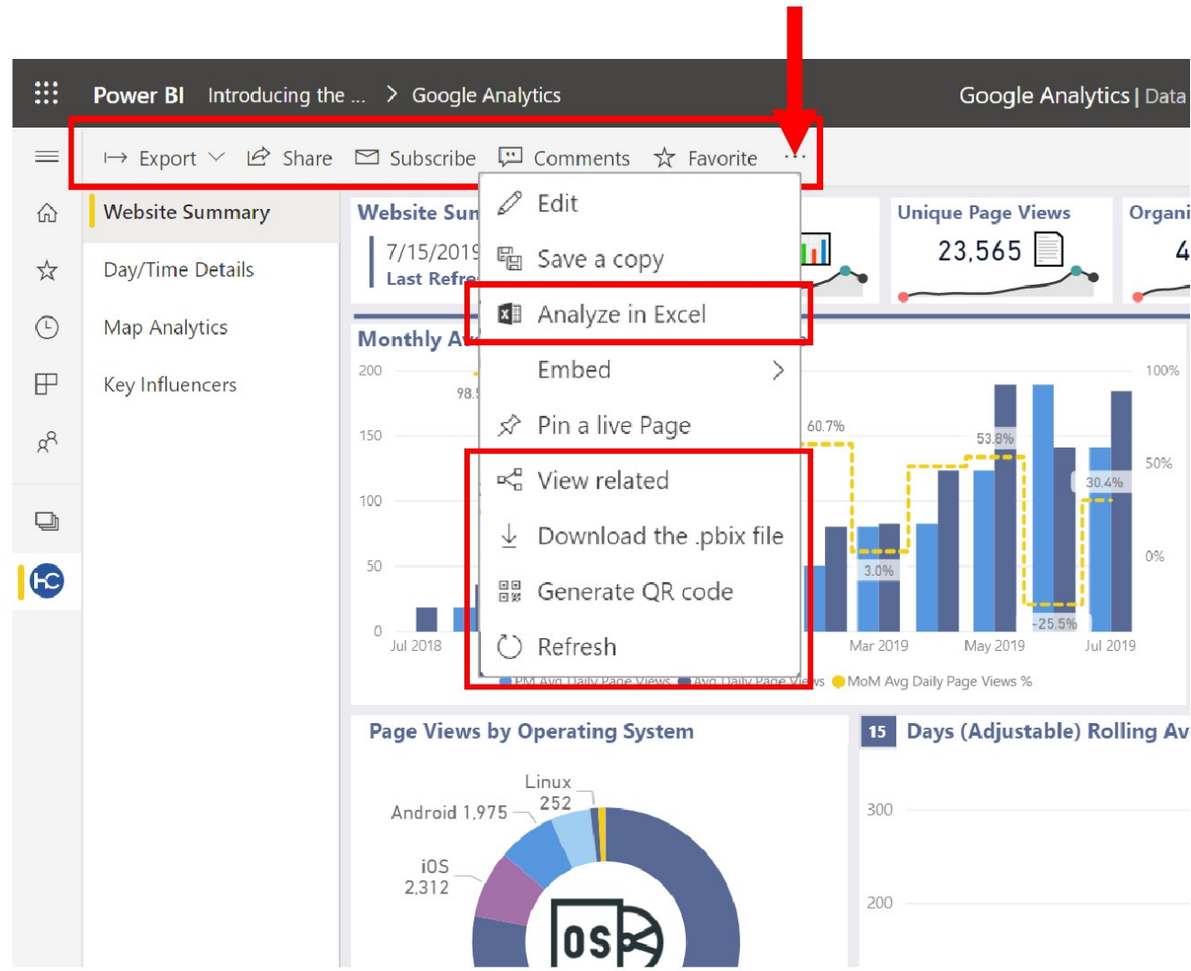
- Conditional formatting can be applied to calculated measures in either the table or matrix visuals

Page Title & Geo	Page Views	Unique Views	Searches	Bounce Rate
Havens Consulting	4,326	3,587	546	47.6%
Files & Templates — Hav...	4,270	3,424	225	72.8%
Blog / Media — Havens ...	3,464	2,921	35	73.2%
Blog Posts — Havens Co...	2,695	2,050	33	63.6%
Power BI Vs. Excel — Hav...	2,367	2,202	1,540	91.2%
Havens Consulting — Se...	1,403	1,034	0	37.9%
About Us — Havens Con...	1,226	1,049	455	47.1%
What Is Power BI? — Hav...	855	723	241	57.5%
Creating a Slicer Pop-Out...	800	679	9	80.1%
Google Analytics Report	786	710	20	80.8%
Total	**29,936**	**24,836**	**4,512**	**70.0%**

Interacting with Reports

User Report Interactions

- Common user report interactions include, but not limited to:
  - Export to:
    - PowerPoint
    - PDF
    - Print
  - Share (via email)
  - Analyze in Excel
  - View related
  - Download the .pbix file
  - Generate QR code
  - Refresh



Sharing Reports and Dashboards

Comparing Sharing Environments

Content Type	My Workspace	Workspaces	Apps
Dashboards	X	X	X
Reports	X	X	X
Workbooks	X	X	X
Datasets	X	X	X
Dataflows		X	X
URL Links[1]			X

1: When using the new apps navigation pane