

Power BI Avanzato

Migliora il tuo modello dati

Versione: Gennaio 2021



Chi sono io

Consulente e formatore in ambito business intelligence, business analytics dal 2002 le attività principali sono legate alla progettazione di data warehouse relazionale e alla progettazione multidimensionale con strumenti Microsoft.



Dal 2017 mi occupo di architetture big Data e in generale del modern Data warehousing in Azure

Docente all'Università di Pordenone nel corso di analisi dei dati e Big Data
Community Lead di 1nn0va (www.innovazionefvg.net)

MCP, MCSA, MCSE dal 2017 MCT e dal 2014 MVP per SQL Server e relatore in diverse conferenze sul tema.

- info@marcopozzan.it
 - @marcopozzan.it
 - www.marcopozzan.it
 - http://www.scoop.it/u/marco-pozzan
 - http://paper.li/marcopozzan/1422524394



SQL Server 2012/2014

Prerequisiti e setup

Connettività internet:

Devi essere connesso a Internet Come minimo, un computer con 2 core e 4 GB di RAM con Windows 8 / Windows Server 2008 R2 o versioni successive

Microsoft Power BI Desktop richiede Internet Explorer 10 o versione successiva

Verifica se hai un sistema operativo a 32 o 64 bit per decidere se devi installare le applicazioni a 32 o 64 bit.

- Cerca il computer sul tuo PC, fai clic con il pulsante destro del mouse sulle proprietà del tuo computer
- Sarai in grado di identificare se il tuo sistema operativo è a 64 o 32 bit in base al "tipo di sistema" come mostrato di seguito
- Display: At least 1440x900 or 1600x900 (16:9) recommended.

Download and install Power BI Desktop: Scarica e installa Microsoft Power BI Desktop from <http://www.microsoft.com/en-us/download/details.aspx?id=45331>. Scegli la versione appropriata a 64 o 32 bit in base alla tua piattaforma. Microsoft Power BI Desktop è disponibile per piattaforme a 32 bit (x86) e 64 bit (x64)

Download File per la classe:

Copia i file dalla tua USB a C:\Power BI_Adv_M\

Aprire un browser alla pagina di riferimento di Power Query:

<https://docs.microsoft.com/en-us/powerquery-m/power-query-m-function-reference>

NOTE: Questo laboratorio utilizza dati anonimi reali ed è fornito da ObviEnce LLC. Visita il loro sito per conoscere i loro servizi: www.obvience.com.

Questi dati sono di proprietà di ObviEnce LLC ed è stato condiviso allo scopo di dimostrare la funzionalità PowerBI con i dati di esempio del settore. Qualsiasi utilizzo di questi dati deve includere questa attribuzione a ObviEnce LLC.

Obiettivi del corso



Alla fine di questo corso, sarai in grado di utilizzare Power BI Desktop per importare e modellare i dati e trasformare i dati per dare una forma corretta al modello. In particolare sarai in grado di:

- Comprendere il modello di dati di Power BI Desktop, i suoi componenti e gli schemi più efficaci
- Otttenere una comprensione del linguaggio Power Query M.
- Creare query utilizzando la barra degli strumenti e l'Editor avanzato
- Comprendere i parametri
- Organizzare le query usando I folder
- Progettare i tuoi modelli con una logica enterprise

COURSE AGENDA

- Module 1: Concetti base di data modelling
- Module 2: Incominciare con M (Power Query Language)
- Module 2 Lab: Trasformazioni di importazione base
- Module 3: Trasformazioni avanzate di M
- Module 3 Lab: Trasformazioni complesse
- Module 4: Variabili , parametri e funzioni
- Module 4 Lab: Creazione dei parametri e dei percorsi
- Module 5: Tecniche di ottimizzazione
- Progetto finale
- Module 6: Dataflows
- Apendici
- UC1: Caso d'uso reale

MODULE 1:

Concetti base di data modelling

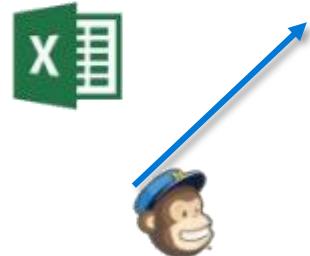
Obiettivi:

- Comprendere l'architettura di base di PBI Desktop
- Perchè dobbiamo comprendere il data modeling

Power BI Desktop Data Flow

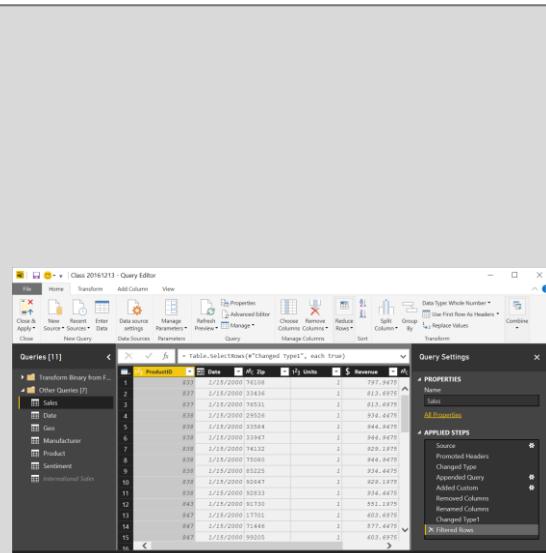


Data Sources



Query Editor

Data Source Connections
Data Transformations
(Prep data for Data Model)



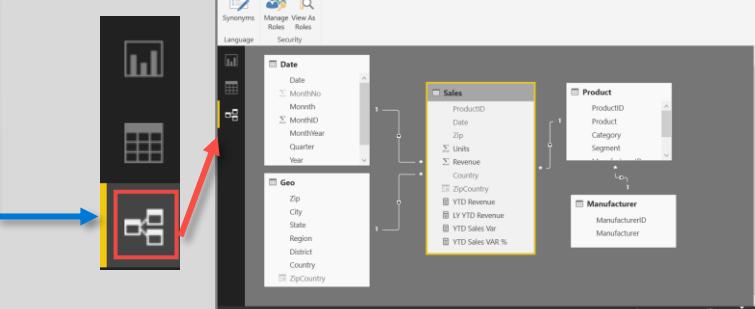
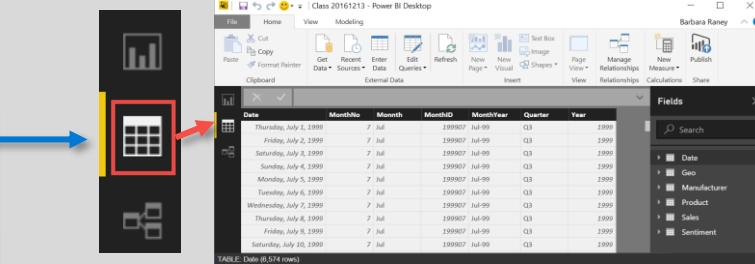
Power BI Desktop file (.PBIX)

Close & Apply



Close:
Closes
Query
Editor

Apply:
Loads
data
from
sources
to Data
Model



Report

Create Visuals

Data

View Tables

Relationships

See how Tables
relate to each
other

Lavorare con 1 singola tabella

In excel siamo abituati a lavorare con una singola tabella

Con excel è semplice ed è già un data model

Ma ci sono delle limitazioni

- Il numero di righe è minore di un milione (Il limite della dimensione limita la dimensione del modello)
- La velocità e la memoria non sono ottimali
- Possiamo fare solo calcoli basici

Lavorare con 1 singola tabella (demo 00): Problema 1



- Problema 1: limite analitico
- Se abbiamo il seguente excel (L'ESPRESSIVITÀ massima è data dalle colonne che si vedono)
- Se non ci sono le colonne nel tabellone non si possono fare le analisi
 - Se dobbiamo aggiungere l'analisi per **Podotto** che non c'è allora dobbiamo portarci il colore dentro la nostra tabella aumentando di fatto le dimensioni. Non ho la possibilità di caricare più tabelle **Aumenteremo la granularità e la dimensione della tabella**

FullDateLabel	Manufacturer	BrandName	ProductCategoryName	SalesQuantity	SalesAmount	TotalCost
2007-03-31	Adventure Works	Adventure Works	Home Appliances	55	14332.268	7651.84
2008-10-22	Contoso, Ltd	Contoso	Cell phones	2040	23504.88	12648.94
2009-01-31	Adventure Works	Adventure Works	TV and Video	194	51593.106	28146.4
2009-01-21	Fabrikam, Inc.	Fabrikam	Cameras and camcorders	282	163007.2	76709.45
2007-12-31	Adventure Works	Adventure Works	Computers	29	14008.43	7944.32
2007-06-22	Contoso, Ltd	Contoso	Cell phones	680	6107.24	3420.44
2007-06-22	Proseware, Inc.	Proseware	Computers	86	71417.6	30786.94
2007-08-23	Adventure Works	Adventure Works	Computers	43	22672.2	9954.6
2009-03-30	The Phone Company	The Phone Company	Cell phones	198	48500.37	24164.56
2008-03-24	Contoso, Ltd	Contoso	Cell phones	306	7353.594	3914.64
2007-09-30	Fabrikam, Inc.	Fabrikam	Home Appliances	44	4805.604	2824.24
2007-11-13	Adventure Works	Adventure Works	Computers	153	47357.97	28256.02
2008-12-06	Contoso, Ltd	Contoso	Computers	32	10790.4	6477.2
2007-11-14	Contoso, Ltd	Contoso	Cameras and camcorders	146	55397.5	25876
2009-12-30	Adventure Works	Adventure Works	Computers	32	15107.75	7952.97
2009-03-13	Wide World Importers	Wide World Importers	Audio	42	7990.92	3607.26
2009-08-11	Wide World Importers	Wide World Importers	Audio	9	1466.1	749.16
2009-09-28	Contoso, Ltd	Contoso	Home Appliances	78	9955.268	5189.27
2008-02-18	A. Datum Corporation	A. Datum	Cameras and camcorders	345	70989.93	32872.58
2007-08-15	Litware, Inc.	Litware	Home Appliances	69	112603.8	56472.35

Lavorare con 1 singola tabella (demo 0): Problema 2

- Problema 2: limite dimensioni di 1 milione di righe
- Granularità = Livello di dettaglio della tabella
- Supponiamo ora di aggiungere la Sottocategoria nel nostro tabellone. Questa non solo aumenta la dimensione della tabella ma mi va ad incidere sulla granularità cioè sul dettaglio minimo di ogni riga della tabella di fatto aumentando le righe e noi sappiamo che abbiamo solo 1 milione di righe
- Più granularità più aumentano le righe oltre alle colonne quindi abbiamo un aumento in profondità e in larghezza

		Category	Subcategory	Sales
Category	Sales			
Bikes	10,000	Bikes	Cross bikes	8,000
		Bikes	Mountain bikes	2,000
Helmets	5,000	Helmets	Colorful helmets	2,000
		Helmets	Lightweight helmets	3,000

Che cosa è un data model?

- Abbiamo visto i limiti di lavorare con una unica tabella in excel
- Usando Power Pivot o Power BI si sfruttano meglio i dati del modello consentono di avere un **Data Model**
- Non ci sono limiti nella dimensione delle tabelle quindi possiamo definire la granularità **a piacimento**, anche con una sola tabella (**Risoluzione del 2 problema visto nelle slide precedenti**)
- Con il data model quindi:
 - **Carichiamo più tabelle in un modello singolo** (**Risoluzione del 1 problema visto nelle slide precedenti**)
 - Costruiamo delle relazioni tra le tabelle
 - Anche se non è argomento di questa parte con un data model possiamo sfruttare la potenza di DAX

Basi di modellazione: errori (demo 01)

- Ora abbiamo scoperto che con Power BI o Power Pivot per Excel i limiti analitici e i limiti di memoria si possono superare
- La prossima fase è pensare a come costruire il data model in maniera corretta in quanto non basta passare ai tool di Power BI e Power Pivot per risolvere i nostri problemi.
- Non possiamo pensare comunque di mettere tutto su una tabella e tra poco vediamo perché

Basi di modellazione: normalizzazione (demo 02)

- Quello che abbiamo fatto nella soluzione della demo precedente si chiama normalizzazione che è il contrario della denormalizzazione
- La normalizzazione è il processo di organizzazione di colonne (attributi) e tabelle di un database per ridurre la ridondanza (risparmiare spazio) dei dati e migliorare l'integrità dei dati (si usa nei database OLTP dove devo avere velocità e risparmio di spazio)

Manufacturer	BrandName	ProductSubcategoryName	Brand Code	Brand Name
Adventure Works	Adventure Works	Coffee Machines	1	Adventure Works
Contoso, Ltd	Contoso	Cell phones Accessories	2	Contoso
Adventure Works	Adventure Works	Televisions	3	Fabrikam
Fabrikam, Inc.	Fabrikam	Camcorders	4	Proseware
Adventure Works	Adventure Works	Laptops	5	The Phone Company
Contoso, Ltd	Contoso	Cellphones Accessories
Proseware, Inc.	Proseware	Projectors & Screens		
Adventure Works	Adventure Works	Laptops		
The Phone Company	The Phone Company	Touch Screen Phones		
Contoso, Ltd	Contoso	Home & Office Phones		
Fabrikam, Inc.	Fabrikam	Microwaves		
Adventure Works	Adventure Works	Desktops		
Contoso, Ltd	Contoso	Projectors & Screens		
Contoso, Ltd	Contoso	Digital SLR Cameras		
Adventure Works	Adventure Works	Desktops		
Wide World Importers	Wide World Importers	Recording Pen		
Wide World Importers	Wide World Importers	Recording Pen		
Contoso, Ltd	Contoso	Microwaves		
A. Datum Corporation	A. Datum	Digital Cameras		
Litware, Inc.	Litware	Washers & Dryers		

Basi di modellazione: denormalizzazione (demo 02)



- La denormalizzazione è l'opposto della normalizzazione, quella aumenta la ridondanza dei dati, con l'obiettivo di migliorare la comprensione del modello (Denormalizzazione nei customer). La denormalizzazione si usa per i fini analitici.
- Il grado di denormalizzazione dipende dalle analisi che dobbiamo fare
 - Abbiamo visto che non è una buona idea perché si sparpagliano le informazioni ma non è proprio vero lo vedremo successivamente

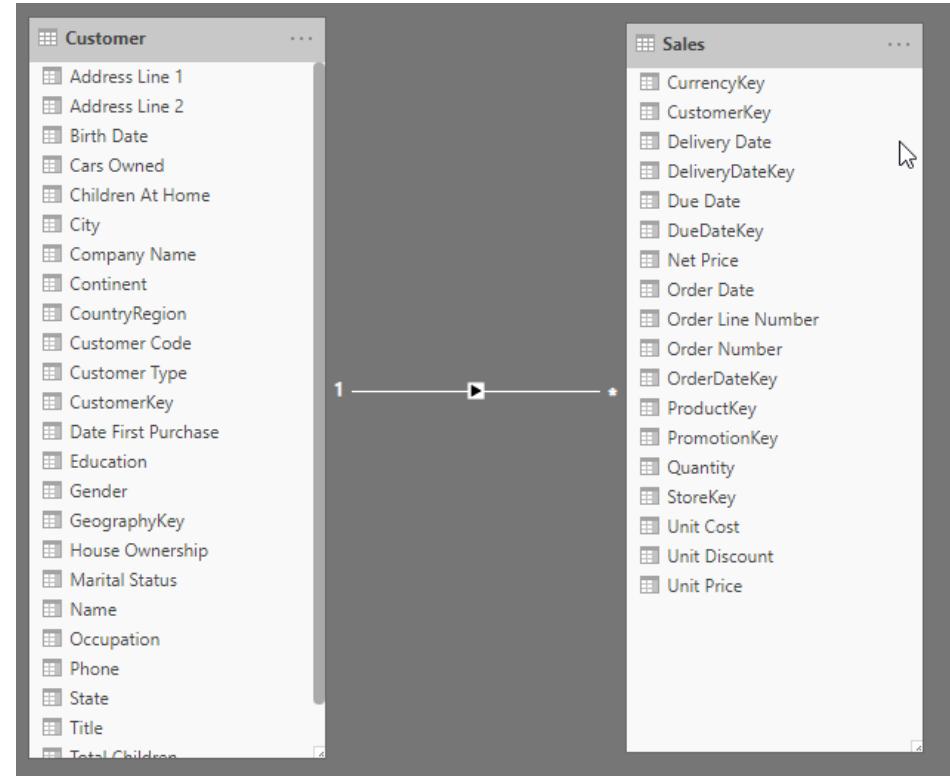
FullDateLabel	Manufacturer	BrandName	ProductCategoryName	SalesQuantity	SalesAmount	TotalCost
2007-03-31	Adventure Works	Adventure Works	Home Appliances	55	14332.268	7651.84
2008-10-22	Contoso, Ltd	Contoso	Cell phones	2040	23504.88	12648.94
2009-01-31	Adventure Works	Adventure Works	TV and Video	194	51593.106	28146.4
2009-01-21	Fabrikam, Inc.	Fabrikam	Cameras and camcorders	282	163007.2	76709.45
2007-12-31	Adventure Works	Adventure Works	Computers	29	14008.43	7944.32
2007-06-22	Contoso, Ltd	Contoso	Cell phones	680	6107.24	3420.44
2007-06-22	Proseware, Inc.	Proseware	Computers	86	71417.6	30786.94
2007-08-23	Adventure Works	Adventure Works	Computers	43	22672.2	9954.6
2009-03-30	The Phone Company	The Phone Company	Cell phones	198	48500.37	24164.56
2008-03-24	Contoso, Ltd	Contoso	Cell phones	306	7353.594	3914.64
2007-09-30	Fabrikam, Inc.	Fabrikam	Home Appliances	44	4805.604	2824.24
2007-11-13	Adventure Works	Adventure Works	Computers	153	47357.97	28256.02
2008-12-06	Contoso, Ltd	Contoso	Computers	32	10790.4	6477.2
2007-11-14	Contoso, Ltd	Contoso	Cameras and camcorders	146	55397.5	25876
2009-12-30	Adventure Works	Adventure Works	Computers	32	15107.75	7952.97
2009-03-13	Wide World Importers	Wide World Importers	Audio	42	7990.92	3607.26
2009-08-11	Wide World Importers	Wide World Importers	Audio	9	1466.1	749.16
2009-09-28	Contoso, Ltd	Contoso	Home Appliances	78	9955.268	5189.27
2008-02-18	A. Datum Corporation	A. Datum	Cameras and camcorders	345	70989.93	32872.58
2007-08-15	Litware, Inc.	Litware	Home Appliances	69	112603.8	56472.35

Abbiamo imparato che se il data model è fatto bene le nostre formule DAX saranno molto semplici 😊

E il primo step per diventare modellatori di dati

Basi di modellazione: Business Entity

- Nella demo che abbiamo fatto la tabella dei customer è vista come una business entity
- Tutti i campi della customer sono riferiti al concetto di customer a differenza della tabella delle vendite
- Ogni business entity DEVE avere una tabella per se
- Ogni business ha differenti entità
 - Products, Customer, Resellers
 - Patients, Medications, doctor
 - Claim, Customer
- Ogni business entity ha una caratteristica unica



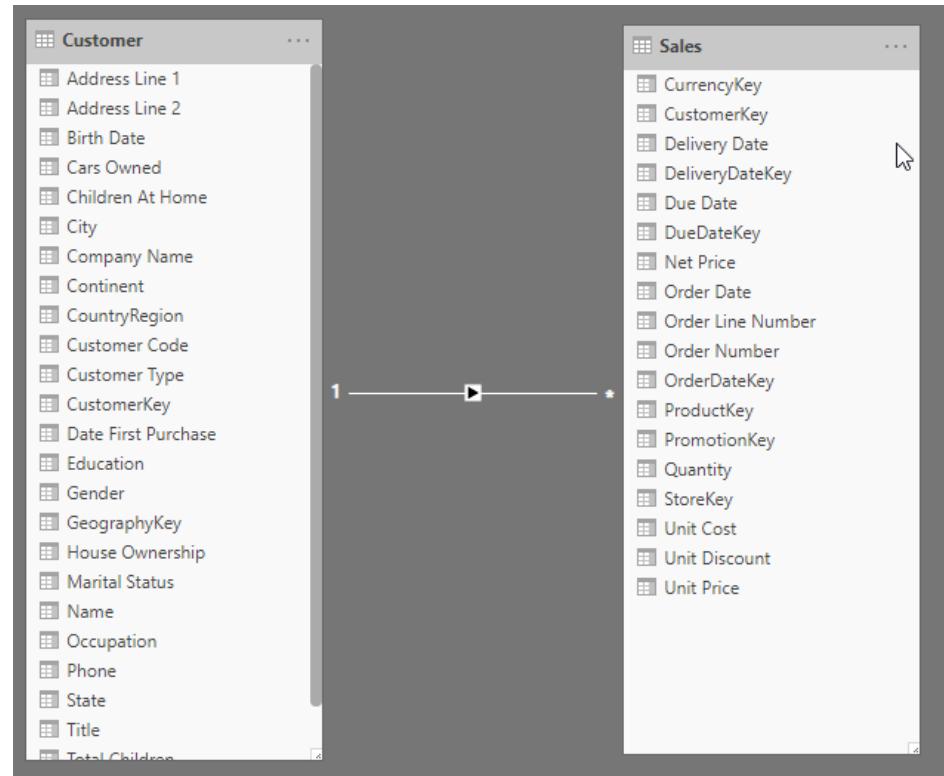
Basi di modellazione: Business Entity

- Perché il campo **Education** di **Customer** non è una tabella?
- Education non è una business entity ma è un **attributo del customer**. Customer è la tabella rilevante
- Definire che cosa è un attributo e cosa è una entity non è subito facile ma viene dall'esperienza e dalla pratica dipende dagli scenari ad esempio:
 - Education è possibile vederla come attributo di un customer in certi scenari mentre in altri potrebbe essere vista come tabella a se per rappresentare una business entity

1 ENTITY = 1 TABLE

Basi di modellazione: Business Entity

- Quando si hanno più tabelle in un modello per forza di cose bisogna unirle con una relazione
- Tabelle senza relazioni in un modello non servono a nulla
- Una relazione può avvenire tra due tabelle
- Una relazione ha una direzione:
 - **Customer lato uno**
 - **Sales: lato molti**
- Best Practice: Dare lo stesso nome di colonna tra le tabelle che devo mettere in relazione

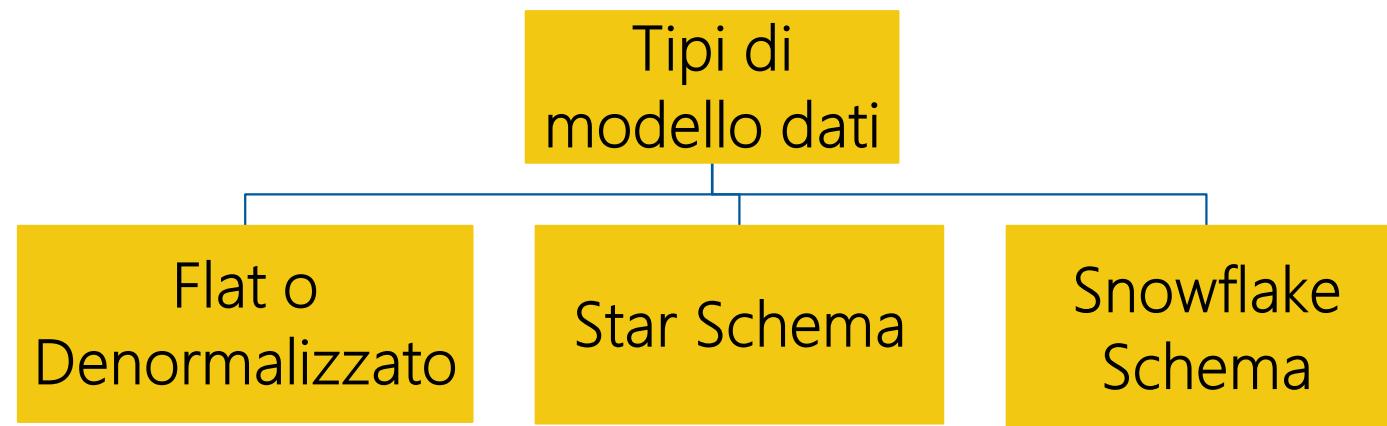


Basi di modellazione: Granularity

- Quando si hanno più tavelle in un modello la granularità è differente
- Ogni tabella ha la sua granularità non come nel tabellone unico
 - Customer ha una granularità a livello di customer
 - Data ha una granularità a livello di data
 - Product ha una granularità a livello di prodotto
- Le vendite (Sales) hanno una granularità definita dalle relazioni entranti
 - Customer, Date e product level
 - Se abbiamo entrambe le tre tavelle

Tipi di modelli di dato

Il modello dati unisce fatti e dimensioni



Nota: Questo non è un elenco esaustivo, ma sono i tipi di modello più comuni utilizzati da Power BI.

Tipi di modelli di dato: Flat Schema

Flat or Denormalized schema (“Kitchen sink” schema) **evitaimolo!!!**

	1 ² 3 ProductID	A ^B C Product	Date	1 ² 3 CustomerID	A ^B C Email	A ^B C Last Name	A ^B C First Name	A ^B C Full Name	1 ² 3 CampaignID	1 ² 3 Units	1.2 CatSegID
1	676	Maximus UC-41	9/25/2011	70283	Farrah.Kent@xyz...com	Kent	Farrah	Farrah Kent	22	1	10
2	585	Maximus UC-50	3/24/2014	70283	Farrah.Kent@xyz...com	Kent	Farrah	Farrah Kent	15	1	10
3	585	Maximus UC-50	11/30/2014	138334	Martha.Mcclain@xyz...com	Mcclain	Martha	Martha Mcclain	8	1	10
4	585	Maximus UC-50	6/21/2015	27193	Hedda.Mcintosh@xyz...com	Mcintosh	Hedda	Hedda McIntosh	22	1	10
5	585	Maximus UC-50	1/6/2013	238970	Lunea.Walker@xyz...com	Walker	Lunea	Lunea Walker	21	1	10
6	585	Maximus UC-50	3/22/2013	182241	Upton.Page@xyz...com	Page	Upton	Upton Page	17	1	10
7	449	Maximus UM-54	9/25/2011	195385	Drake.Wells@xyz...com	Wells	Drake	Drake Wells	22	1	4
8	449	Maximus UM-54	9/30/2014	168009	Wallace.Bender@xyz...com	Bender	Wallace	Wallace Bender	17	1	4
9	449	Maximus UM-54	8/12/2014	110391	Astra.Erickson@xyz...com	Erickson	Astra	Astra Erickson	20	1	4
10	449	Maximus UM-54	4/16/2014	49327	Echo.Bradley@xyz...com	Bradley	Echo	Echo Bradley	7	1	4
11	449	Maximus UM-54	2/28/2013	65952	Yoko.Gross@xyz...com	Gross	Yoko	Yoko Gross	17	1	4
12	449	Maximus UM-54	6/6/2013	97	Yoshi.Grant@xyz...com	Grant	Yoshi	Yoshi Grant	10	1	4
13	449	Maximus UM-54	5/14/2013	56757	Brian.Carrillo@xyz...com	Carrillo	Brian	Brian Carrillo	10	1	4
14	449	Maximus UM-54	4/9/2015	248715	Mark.Hewitt@xyz...com	Hewitt	Mark	Mark Hewitt	19	1	4
15	449	Maximus UM-54	4/28/2013	248715	Mark.Hewitt@xyz...com	Hewitt	Mark	Mark Hewitt	8	1	4
16	449	Maximus UM-54	3/28/2014	240831	Oscar.Avila@xyz...com	Avila	Oscar	Oscar Avila	18	1	4
17	449	Maximus UM-54	2/26/2014	201004	Duncan.Mcintosh@xyz...com	Mcintosh	Duncan	Duncan McIntosh	19	1	4
18	615	Maximus UC-80	5/14/2012	212645	Jacob.Santiago@xyz...com	Santiago	Jacob	Jacob Santiago	22	1	10
19	615	Maximus UC-80	5/14/2012	70666	Hilary.Coller@xyz...com	Collier	Hilary	Hilary Collier	22	1	10
20	615	Maximus UC-80	5/14/2012	114459	Chester.Mitchell@xyz...com	Mitchell	Chester	Chester Mitchell	22	1	10
21	615	Maximus UC-80	5/14/2012	221670	Sage.Yang@xyz...com	Yang	Sage	Sage Yang	22	1	10
22	615	Maximus UC-80	6/3/2012	168009	Wallace.Bender@xyz...com	Bender	Wallace	Wallace Bender	22	1	10
23	615	Maximus UC-80	6/3/2012	154439	Iliana.Dunlap@xyz...com	Dunlap	Iliana	Iliana Dunlap	22	1	10
24	615	Maximus UC-80	6/4/2012	191391	Joelle.Lee@xyz...com	Lee	Joelle	Joelle Lee	22	1	10

- Tutti gli attributi per il modello esistono in una singola tabella
- Altamente inefficiente
- Il modello ha copie extra di dati e prestazioni lente
- Le dimensioni di una tabella piatta possono esplodere molto rapidamente quando il modello di dati diventa complesso

Tipi di modelli di dato: Star Schema

Componenti di un modello di dati – Fact Table (Tabella dei fatti)

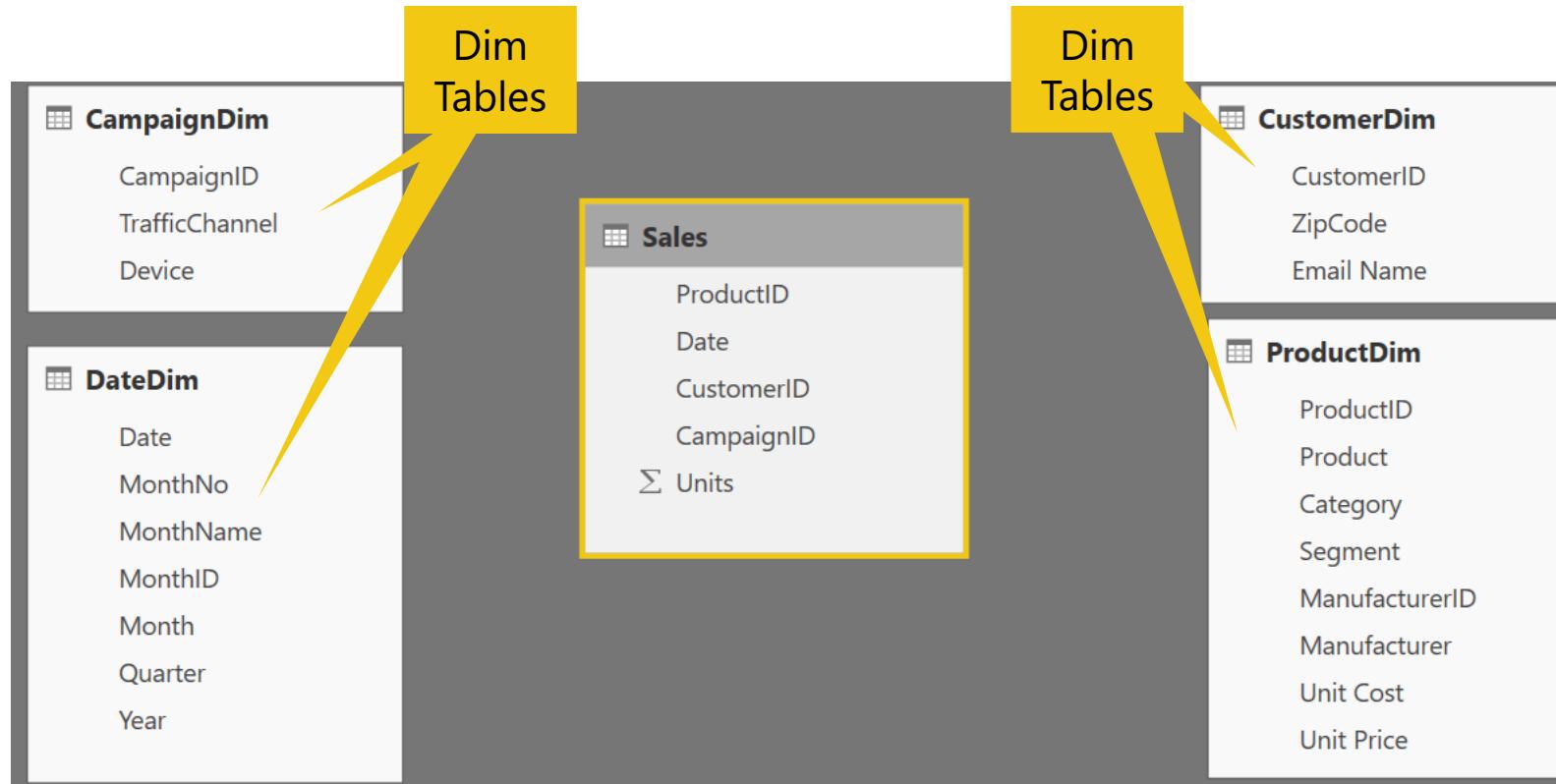


Fact Table

- Contiene le misure (oggetti che posso aggregare) di un processo di business
- **Esempi:**
 - Transactions
 - Sales Revenue
 - Units
 - Cost
- Le misure sono normalmente filtrabili **Esempio:** By Month, By Customer

Tipi di modelli di dato: Star Schema

Componenti di un modello di dati – Dim Table



Dim Table

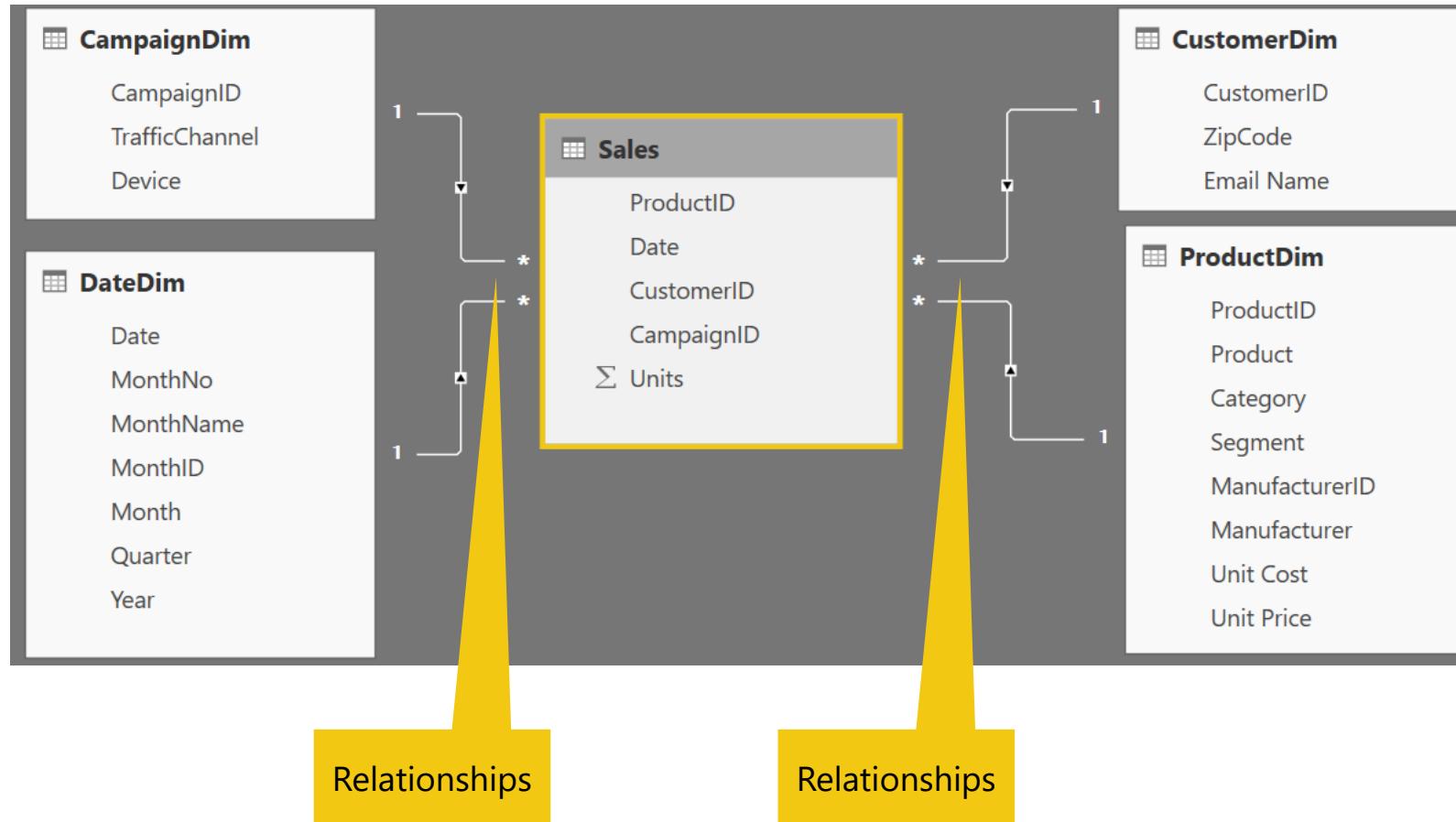
- Una Dim (o Dimensione)
La tabella contiene attributi descrittivi che definiscono come un fatto dovrebbe essere aggregato

Esempio:

By month,
By Customer, By Geo

Tipi di modelli di dato: Star Schema

Componenti di un modello di dati - Relazioni

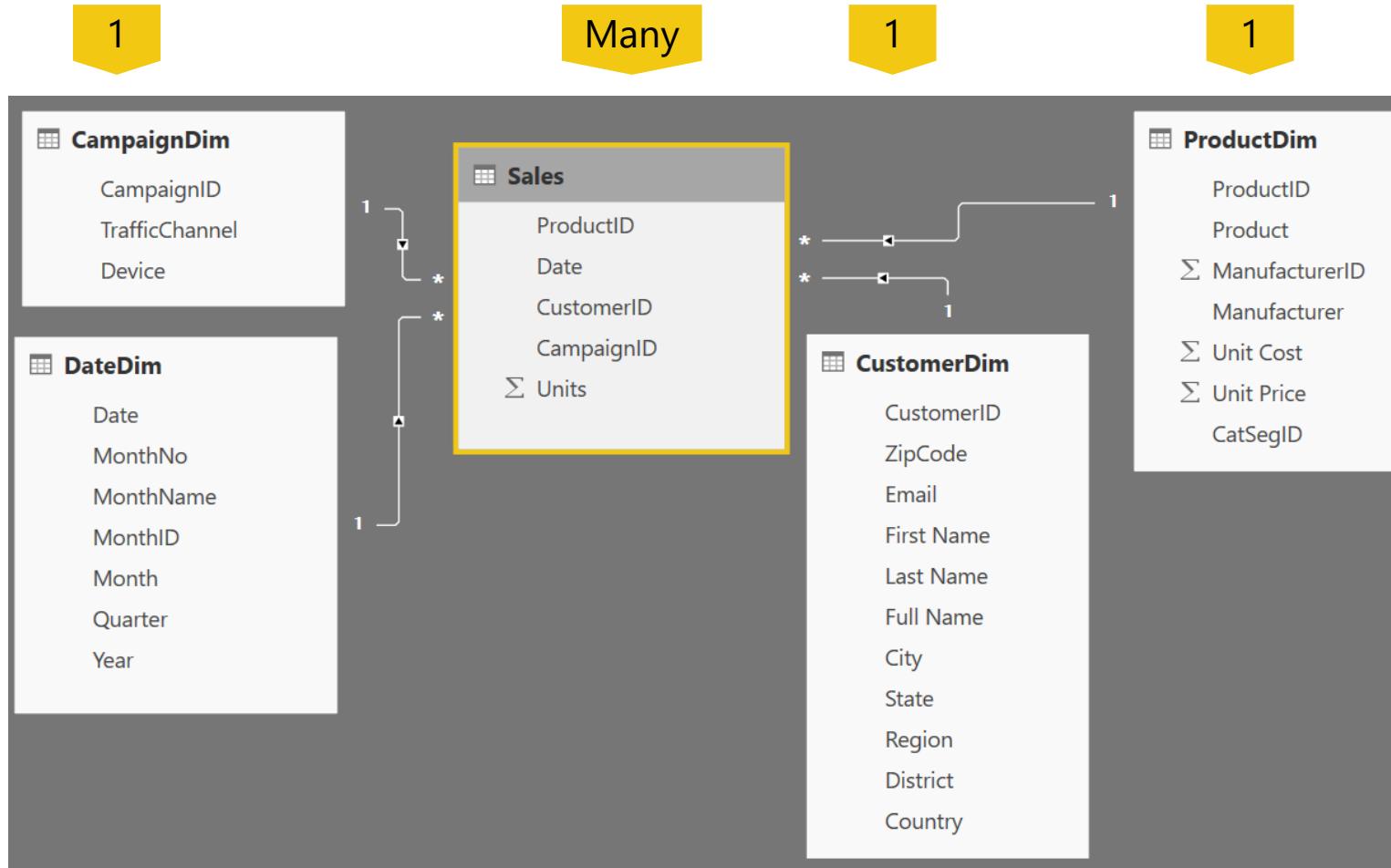


Relazioni

- La connessione tra due 2 tables (normalmente i fatti e le tabelle delle dimensioni) usando una colonna fra di loro
- 3 tipi di relazioni
 - 1 to Many
 - 1 to 1
 - Many to Many
(Attenzione!!! Utilizzare una bridge table)

Tipi di modelli di dato: Star Schema

Star Schema



- Tabella dei fatti al centro
- Circondato da dim
- Sembra una "stella"
- La tabella dei fatti è il lato "Molti" della relazione (da una a molte)

Tipi di modelli di dato: Recap Fatti

Note importanti sulle nostre tabelle dei fatti

- Le tabelle dei fatti sono tipicamente molto grandi
 - Possiamo avere bilioni di vendite
 - Ma è diverso da avere bilioni di customers
- I fatti sono associati a numeri
 - La quantità di vendite
- Non tutti i numeri appartengono alla tabella dei fatti
 - Prezzo di un prodotto sta nella tabella del prodotto (dimensione)
 - Quantità di prodotti venduti appartiene alla tabella dei Fatti
- Le misure dipicamente sono aggregazioni sulle tabelle SUM(Sales[Quantity])

Tipi di modelli di dato: Recap Dimensioni

Note importanti sulle nostre tabelle delle dimensioni

- Le dimensioni tipicamente sono piccole tabelle
 - 1 milione di righe per una dimensione è già un numero grande
- Le dimensioni contengono molte stringhe
 - Nome del customer, indirizzo ecc...
 - Colore del prodotto e dimensione e ccc
- Sono tipicamente quelle su cui si effettua lo slice
- Quando dobbiamo calcolare in genere le usiamo per conteggiare su di loro
 - Numero di clienti
 - Numero di prodotti
- Le dimensioni non sono relazionate ad altro.

Tipi di modelli di dato: Helper column

Molte volte dobbiamo creare delle colonne o delle tabelle o relazioni che non sono utili per l'utente finale ma per il modello

Questi oggetti si chiamano «helper objects»

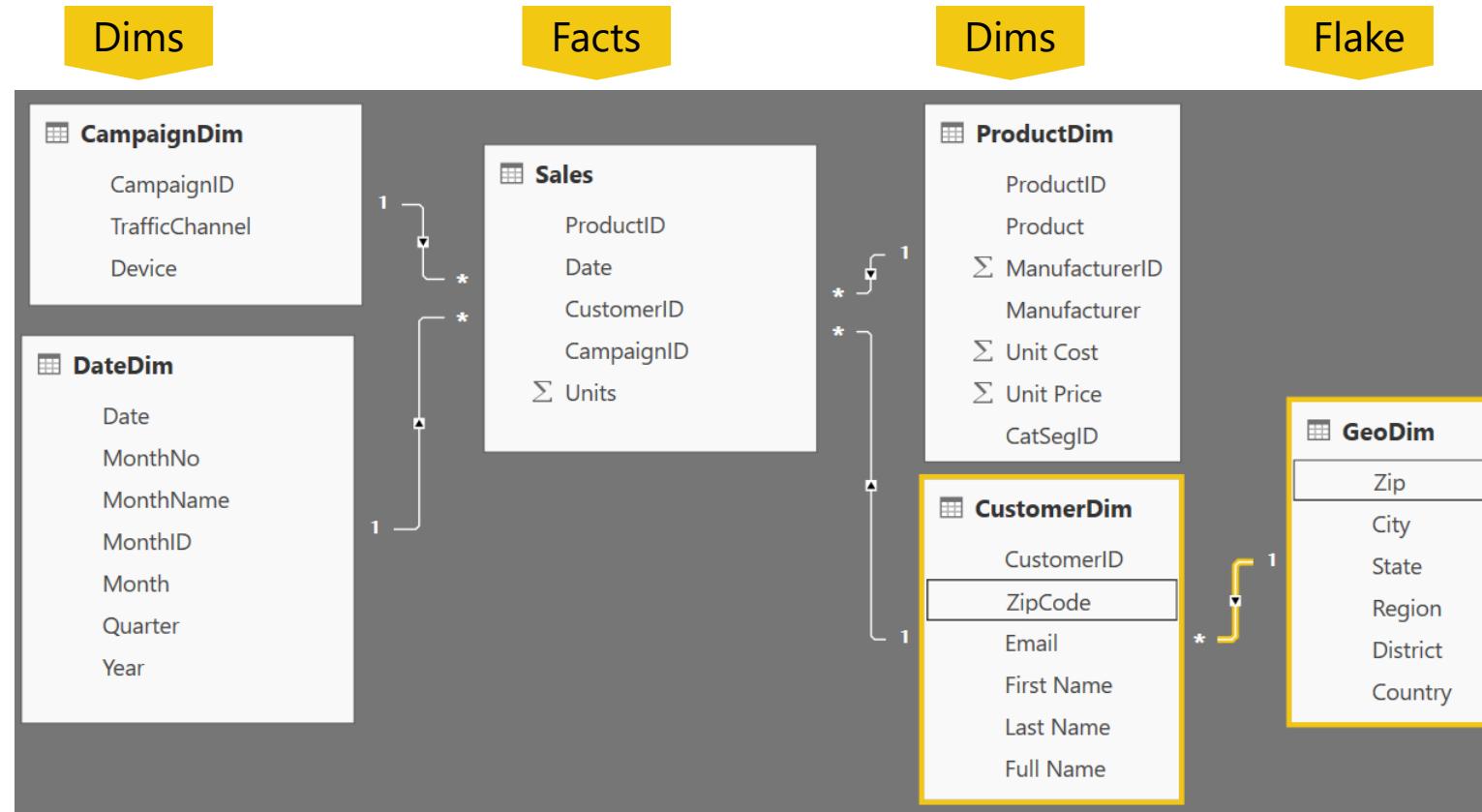
- Helper Tables (configurazioni, Calcoli complessi intermedi)
- Helper Columns (branding, segmentazione ecc..)
- Helper Relationship

Un esempio di Helper Columns è la colonna che creiamo per fare il sort di un'altra colonna «**sort by column**»

La differenza tra le tavole dei fatti e le dimensioni sono il cuore di uno star schema in un modello di BI

Tipi di modelli di dato: Snowflake

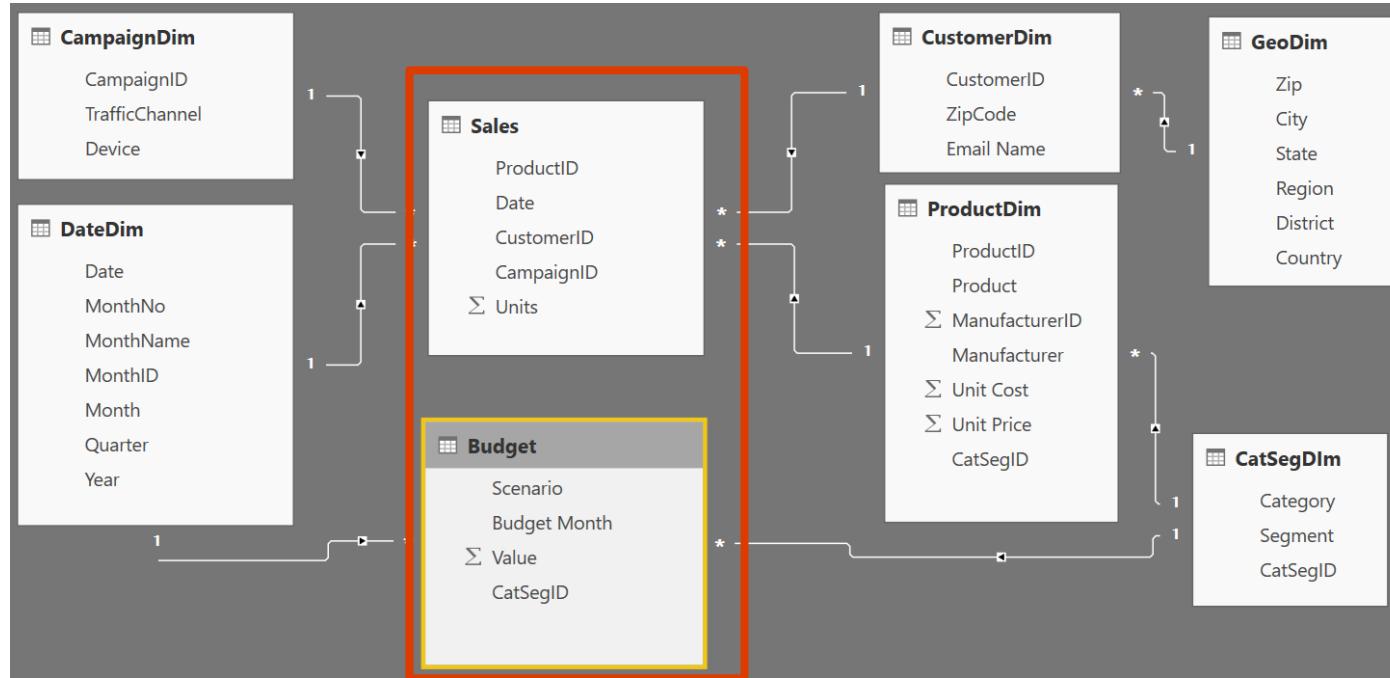
**Snowflake Schema (evitiamolo perchè richiede più attenzioni 😊
!!! Denormalizziamo tutto in una dimensione)**



- Il centro è uno schema a stella
- Tabella dei fatti al centro
- Circondato da dim
- Dim "fiocco di neve" collegate ad altre dim
- Se ne hai molti, sembra un "fiocco di neve"
- Le tabelle Dim o Fact possono essere il lato "Molti" della relazione

Basi di Data Modeling: Diverse Granularità

Granularità & Fatti multipli



SalesFact (Giornaliera per prodotto)

BudgetFact (Mensile per categoria di prodotto e segmento di prodotto)

- Grain (**granularità**) misura il livello di dettaglio della tabella
- Esempio:
una riga per order o per oggetti per giorno o mese è il grain
- Se i tuoi fatti hanno differenti granularità, dovete dividerle in differenti tabelle dei fatti e **connetterle tra loro da dimensioni condivise** (Shared) al livello di granularità più basso di ogni singola tabella dei fatti

Granularità & Fatti multipli

- Tieni presente che le tue misure (calcoli) dovrebbero essere posizionate su una tabella dei fatti, ma in Power BI in realtà non importa quale.
- Quando si ha un dato di fatto con il grain del mese, come il budget, è possibile creare una colonna calcolata che contiene il primo (o l'ultimo) giorno del mese da utilizzare nella creazione della relazione.
- Uno dei punti chiave è che ciò che fa il slice (taglio di una analisi) per tutte le misure proviene dalle (Dimensioni) che contengono gli attributi e quindi sono gli attributi che fanno lo slice in una analisi e non le colonne della tabella dei fatti.

Basi di Data Modeling

Quante dimensioni?

Un'entità = una dimensione

- Più dimensioni sono spesso inutili e confuse

Regole empiriche:

- Numero di dimensioni
 - 10 dimensioni sono buone
 - 20 dimensioni sono probabilmente errate
 - 30 dimensioni sono totalmente errate
- Rapporto dimensione / attributo:
 - 1 dimensione con 20 attributi è buona
 - 20 dimensioni con 1 attributo non sono valide

Basi di Data Modeling

Quante tavole dei fatti?

Multipli eventi uniti assieme

Order No	Customer	Date	Type	Amount
1	Contoso	01/01/2017	Order	100.00
1	Contoso	01/15/2017	Ship	60.00
1	Contoso	01/15/2017	Ship	40.00
2	Imageware	01/16/2017	Order	90.00
3	Imageware	01/20/2017	Order	45.00
4	Contoso	01/22/2017	Order	25.00

Modello corretto

Orders			
Order No	Customer	Order Date	Amount
1	Contoso	01/01/2017	100.00
2	Imageware	01/16/2017	90.00
3	Imageware	01/20/2017	45.00
4	Contoso	01/22/2017	25.00

Shipments		
Order No	Date	Amount
1	01/15/2017	60.00
1	01/15/2017	40.00

Data Types

Types numerici

- Numeri interi
- Numeri decimali
- Numeri decimali (Floating point stored as integer)
- Boolean

Tipi Date/Time

- Date – Internamente memorizzati come interi
- Time – internamente memorizzati come una frazione da 0 a 1
- Date Time

Altri tipi

- Testo
- **Any: non dovresti mai vederlo in un modello di dati. Possono succedere cose brutte !!**

Pro Tip: Data type è diverso da data format vediamo la differenza nella slide sucessiva

*Impostate il
Data Types
in
Query Editor*

*Impostate
Data Formats
(\$ %, etc)
nel data model*

Data Types

I data type sono il modo in cui i dati vengono archiviati, al contrario del data format, ovvero il modo in cui i dati vengono visualizzati

Decimale fisso (Valuta)

Il decimale fisso è 19,4 - 19 posto a sinistra del decimale e 4 a destra, il che consente una corretta elaborazione degli euro. Ideale per i numeri di valuta.

Decimale (che ha decimale mobile) può essere meno efficiente quando i dati hanno una precisione eccessiva

Suddividere DateTime in colonne Date e Time in modo che si comprimano meglio

Numeri decimali fissi: memorizzati come numeri interi utilizzano meno memoria dei decimali

Se il modello non è quello giusto

- Il codice DAX tende ad essere molto complesso
- Le formule sono difficili da pensare
- La complessità si trasforma in problemi di prestazioni

Con il modello corretto

- Il codice DAX è semplice, come dovrebbe essere
- Le prestazioni sono eccezionali

Costruire il modello giusto richiede esperienza → altrimenti a cosa servo ☺

Cosa significa data modeling

- Conoscere diversi schemi
- Essere in grado di abbinare il modello a un pattern
- Applicare il pattern
- Adattare le piccole differenze che appaiono in un modello custom
- I pattern si imparano con l'esperienza e questo corso non copre questi argomenti

Il tuo (business/modello) è diverso?



All'inizio, senti sempre il tuo modello diverso da quelli standard

ma ri cordati che il 99,9% delle volte, non è così

Non deviare dalla modellazione standard, a meno che tu
non sai davvero cosa stai facendo

«La Business Intelligence è nata nel 1958
e in 60 anni abbiamo analizzato quasi tutti i modelli esistenti
Abbiamo scoperto che gli schemi a stella sono l'opzione
migliore»

Quindi perché è importante avere il data model?



- Migliora la comprensibilità dei dati
- Aumenta le prestazioni dei processi
- Evita di lavorare tanto con i linguaggio analitico per eseguire i calcoli
- Aumenta la resilienza al cambiamento

(Demo 1) Disegnare un buon data model

Star Schema – E' la struttura ideale per la maggior parte dei modelli

RAM è preziosa!!!! (Quando il file viene chiuso, viene salvato sul disco rigido
Quando viene aperto, il DB viene archiviato nella RAM (circa il 60% della RAM disponibile
della macchina ricordate!!!) e nel servizio, i set di dati sono limitati a 1 GB)

Alcuni consigli per risparmiare RAM e aumentare la velocità del modello

- Se una tabella dei fatti contiene un campo ID univoco per ciascun record, **rimuoverlo**
 - Esempio. Transaction ID
- **Ordinare le colonne** prima di portarli in un modello di dati Power BI
- Modificare tutti i tipi di dati ANY in un tipo di dati appropriato

1. Cosa consente alle tabelle nel modello di dati di "dialogare" tra loro?
2. Quale tipo di schema ha tutti gli attributi per il modello in una singola tabella?
3. Quando utilizzeresti più tabelle dei fatti

Modulo 1 Lab 0: Data Modeling

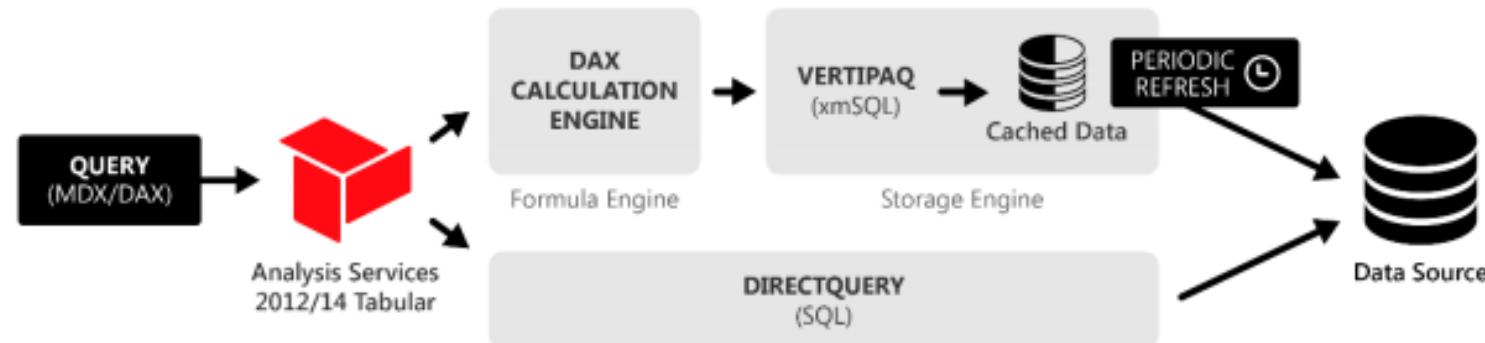
VERTIPAQ

Gestione delle query

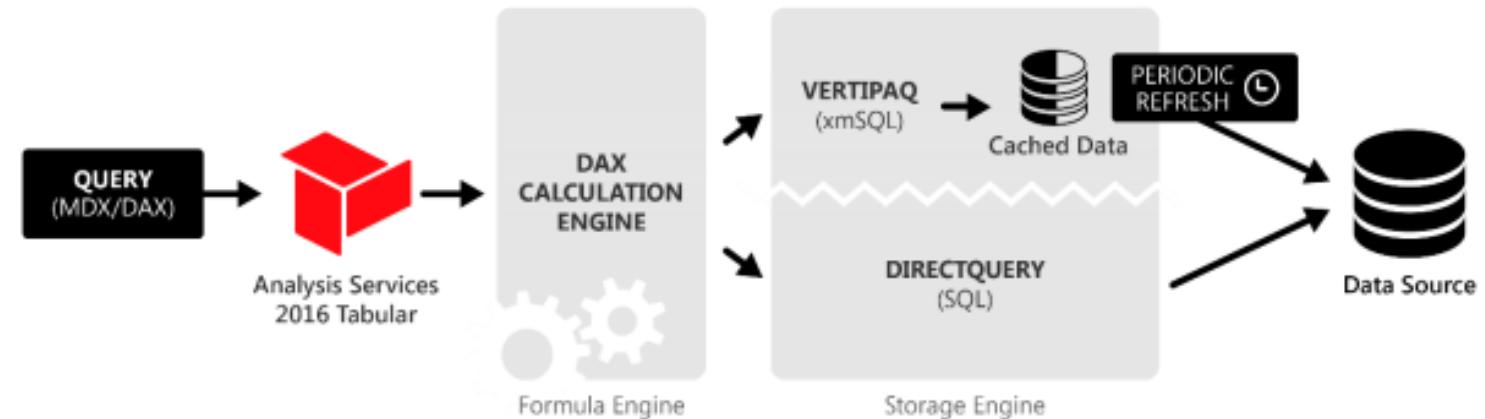
- **Direct Query mode**
 - Trasformazione da DAX a SQL (solo per database relazionali)
 - Query sul motore SQL Server aumentando il workload
 - Molte limitazioni (solo connessione SQL, solo DAX no MDX, limiti DAX, no colonne calcolate, no security)
 - No tempo di processing
 - Real-time queries
- **In Memory mode (vertipaq) (default)**
 - I dati vengono copiati in memoria
 - Singolo accesso all sorgente dati
 - Le query girano su uno «snapshot dei dati»

Gestione delle query: SSAS 2012/14 vs 2016

- **PROCESS** = Lettura dalla sorgente dati in memoria
- Vertipaq contiene il risultato del processing del database



SSAS 2012/2014



SSAS 2016

Che cosa è Vertipaq (xVelocity)?

- E' un database in memoria (dati sono in memoria)
- E' basato su una metodologia relazionale
- Database colonna

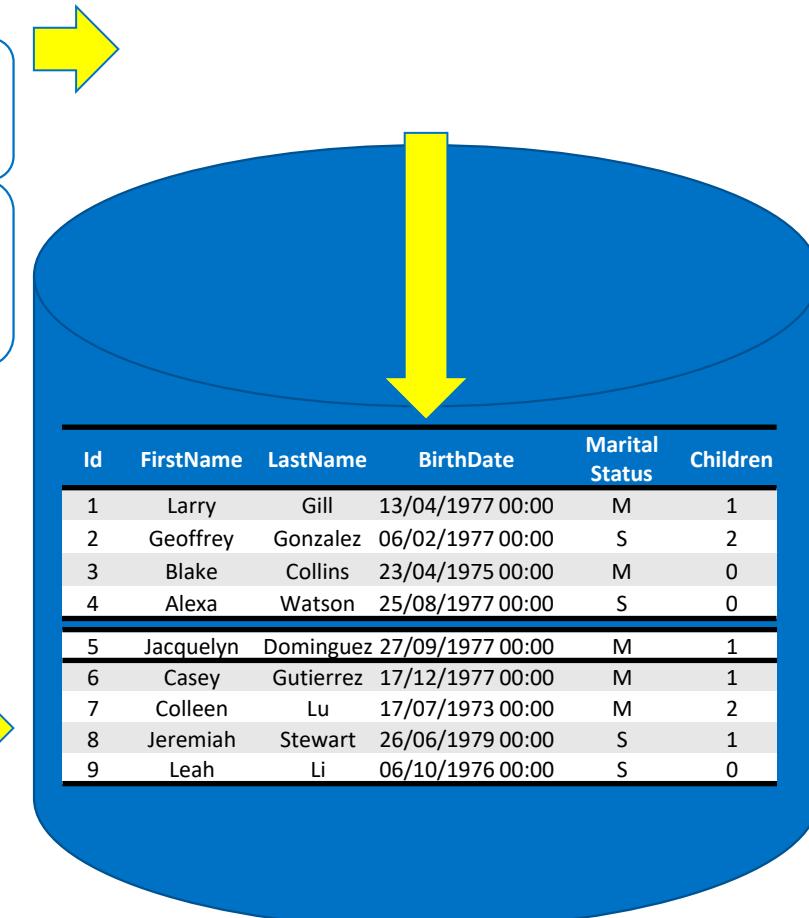
Pro e contro di Vertipaq

- Pro
 - I dati compressi in memoria permettono delle query veloci
 - Posso caricare dati nel modello da sorgenti dato diverso
 - Un singola scansione(query) delle tabelle sorgenti
- Contro
 - Il processing richiede tempo (latenza negli aggiornamenti)
 - E' richiesto un tempo di CPU per processare i dati che costa in termini di risorse
 - Tutti i dati devono stare in memoria!!!
 - I dati devono essere compiati dalle sorgenti dato in SSAS e la rete potrebbe essere un collo di bottiglia.

Come lavora un row storage

Allocò spazio su disco (pagine)

Id	FirstName	LastName	BirthDate	Marital Status	Children
1	Larry	Gill	13/04/1977 00:00	M	1
2	Geoffrey	Gonzalez	06/02/1977 00:00	S	2
3	Blake	Collins	23/04/1975 00:00	M	0
4	Alexa	Watson	25/08/1977 00:00	S	0
5	Jacquelyn	Dominguez	27/09/1977 00:00	M	1
6	Casey	Gutierrez	17/12/1977 00:00	M	1
7	Colleen	Lu	17/07/1973 00:00	M	2
8	Jeremiah	Stewart	26/06/1979 00:00	S	1
9	Leah	Li	06/10/1976 00:00	S	0



Allocò ancora spazio
su disco perchè finito

Caratteristiche del row storage

- SELECT * di una tabella
 - Legge tutta la tabella dalla prima all'ultima riga
- SELECT **SUM(children)** della tabella
 - Legge tutta la tabella dalla prima all'ultima riga e poi si legge solo la colonna children per fare la somma
 - **Prestazioni pessime** per un dato piccolo devo fare tanto I/O su disco che poi non serve

Caratteristiche del row storage

- Il risultato della SELECT SUM(children) **non cambia nemmeno se faccio I/O in memoria e non su disco**
 - Trasferisco i dati dalla memoria alla cache della CPU dove si svolge il calcolo
 - **La cache interna è molto limitata**
 - Pessimo uso perché carico una grande parte di memoria per poi usarne un pezzettino
 - Ottengo comunque prestazioni peggiori rispetto al fatto di lavorare su dati più compatti (**Problema**)

Soluzione con gli indici

- Se devo fare spesso la **SUM(children)**
 - Creo un indice su children
 - La query richiede solo il campo children (l'indice copre la query), **leggo solo l'indice e non tutta la tabella**
 - L'indice contiene dati più compatti e mi aiuta per I/O
 - Gli indici in generale riducono il numero di colonne di una tabella e ottimizzano l'I/O



Concetto di Column Storage

Caratteristiche column storage

- Portiamo il concetto di indice in memoria
- Estremizziamo il concetto di indice

Id	FirstName	LastName	BirthDate	Marital Status	Children
1	Larry	Gill	1 13/04/1977 00:00	M	1
2	Geoffrey	Gonzalez	0 06/02/1977 00:00	S	2
3	Blake	Collins	2 23/04/1975 00:00	M	0
4	Alexa	Watson	2 25/08/1977 00:00	S	0
5	Jacquelyn	Dominguez	2 27/09/1977 00:00	M	1
6	Casey	Gutierrez	1 17/12/1977 00:00	M	1
7	Colleen	Lu	1 17/07/1973 00:00	M	2
8	Jeremiah	Stewart	2 26/06/1979 00:00	S	1
9	Leah	Li	0 06/10/1976 00:00	S	0



Caratteristiche column storage

- Faccio una **SELECT SUM(children)** della tabella
 - Non devo fare un indice
 - La colonna è già l'indice perché contiene solo children
 - Molto veloce
- Faccio **SELECT SUM(children) GROUP BY FirstName**
 - Non è più così bello
 - Devo leggere due colonne: Children e FirstName
 - Devo poi unire il risultato **pagando tempo di CPU**

Caratteristiche column storage

- Rispetto alla row storage
 - più velocità se devo leggere una colonna
 - più velocità se leggo poche colonne
 - più lento se faccio SELECT *

Column vs Row

- Memorizzazione su colonna
 - Accesso veloce ad una singola colonna
 - Ho la necessità di **materializzare le righe**
 - Spendiamo CPU per ridurre I/O (**le CPU le possono fare più veloci o metterne tante ☺**)
- Memorizzazione per riga
 - Accesso veloce alla singola riga
 - Non necessita di materializzazione
 - Spendiamo l'I/O per ridurre la CPU (**i dischi o la memoria non si possono fare più veloci ☹**)

pensare (colonnare)

- Siamo in un mondo in cui lo storage è fatto da una sola colonna 😊.
- Ci sono cose che si possono fare memorizzando i **dati in colonna** che sono più efficienti rispetto ai dati memorizzati su riga

Sapete come funziona la compressione di SQL Server?

- **per riga** (non fa grandi cose... pulizie spazi bianchi, ridurre caratteri unicode, ridurre i decimali)
 - **per pagina** (identifica all'interno della pagina parti uguali e le indicizza per poi comprimere la pagina creando un indice all'inizio)
-

Compressione in Vertipaq

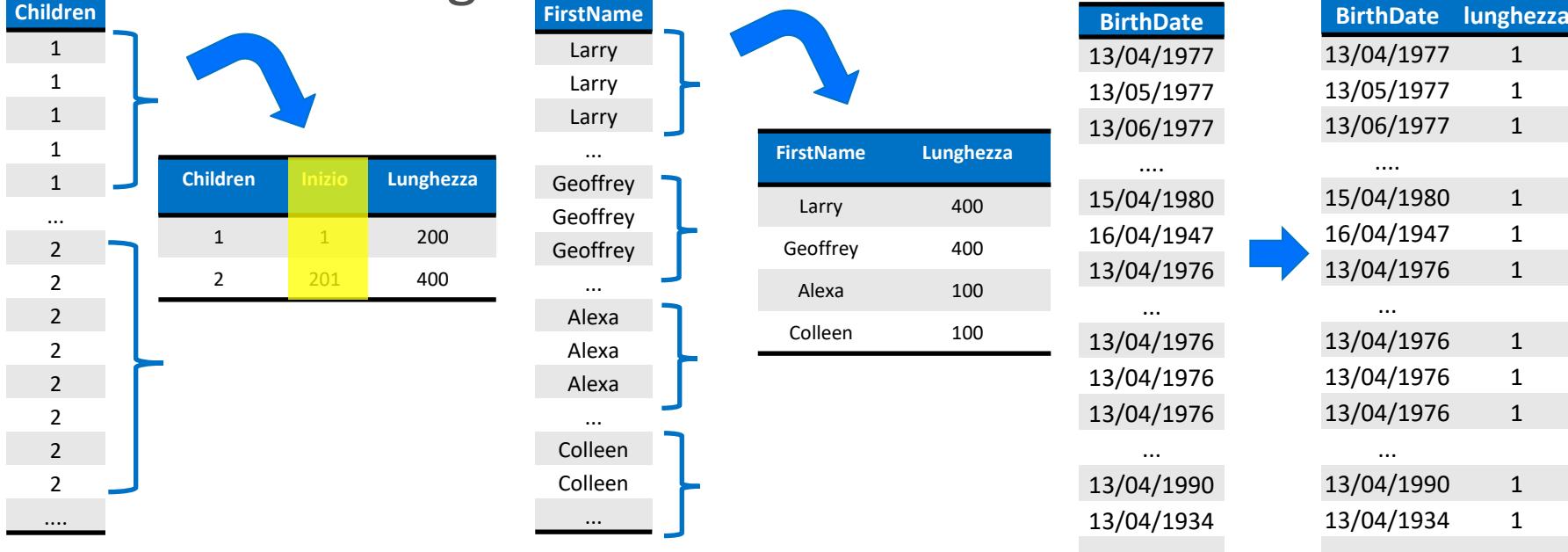
- **Vertipaq** (simile compressione di **pagina in SQL Server**)
 - Identifica parti uguali nell'aria di memoria
 - Crea una struttura per rappresentare le parti uguali e ottiene la **struttura compressa della colonna**
 - Più efficiente di SQL perché si ragiona solo su una **colonna con pochi valori distinti** rispetto alla pagina di SQL in cui ho righe con più colonne e con meno valori distinti ☺.



— **Vediamo come Vertipaq esegue la compressione**

Run Length Encoding 1 livello

Potrei anche decidere di togliere la colonna inizio e tenere solo la fine



Le date cambiano così di frequente che se provassi a comprimerla avrei su lunghezza tutti 1 e otterrei una tabella più grande dell'originale
Vertipaq lascia l'originale.

Run Length Encoding 1 livello

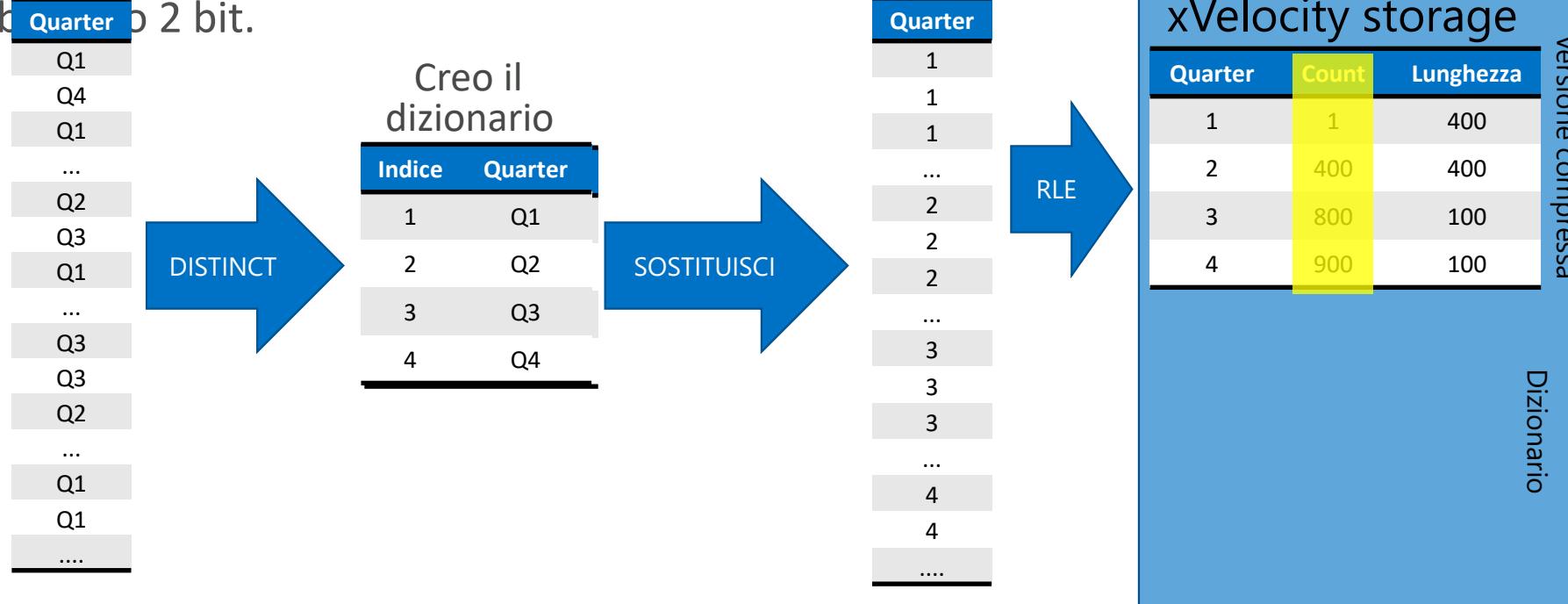
- Vertipaq non usa mai più memoria rispetto alla colonna sorgente...se non riesce a comprimerla la lascia come è
- Vertipaq durante il **processing di un tabella**
 - Divide la tabella in colonne
 - Comprime ogni colonna con RLE
 - **Attenzione!!! L'ordinamento delle colonne deve essere lo stesso per ogni colonna perchè devo materializzare i dati delle varie colonne (se ne occupa vertipaq ☺) buon ordinamento = buona compressione**

Dictionary encoding - 2 livello

- Più importante di RLE
- Vediamo i passi per creare il Dictionary
 1. Vertipaq legge una colonna di tipo stringa
 2. Effettua il distinct della colonna
 3. Ogni valore stringa è associato ad un numero in un Dictionary
 4. Sostituisco i valori stringa nella colonna con i numeri del Dictionary

Dictionary encoding - 2 livello

Conoscendo i possibili valori della stringa utilizzo il numero minimo di bit per rappresentarla. In questo caso 4 possibili valori binari su 2 bit.



Con il dictionary encoding Vertipaq è **datotyping independent**. Non ha nessuna importanza il tipo dei campi che si utilizzano nelle viste per popolare il modello

Conclusioni

- Una stringa nella tabella (osceno) dei fatti non ha più nessun prezzo grazie al **dictionary encoding**
- **DOVETE vivere ☺ pensando che Vertipaq memorizza i dati in questo modo.** E' fondamentale quando andrete a costruire un modello con Vertipaq
- Importa solo il numero di valori distinti delle colonne
 - Tanti valori distinti occupano più spazio (+ RAM) ed più lungo fare analisi
 - Pochi valori distinti occupano poco spazio (- RAM) e tutte operazioni ridotte

Conclusioni

- **Dictionary Encoding**
 - Avviene quando è necessario: per una colonna con valori **interi** e con valori **distinti** molto alti conviene memorizzare il numero perché il dizionario sarebbe troppo grande
 - Rende le tabelle datatype independent
- **RLE Encoding**
 - Solo se i dati compressi sono più piccoli dell'originale
 - Dipende fortemente dall'ordine dei dati
 - SSAS sceglie il **sorting** migliore durante il process (10 s/milione di righe). Trovare stesso ordinamento per le colonne è difficile. **Thomas Kejser**: + 25% compressione con ordinamento sorgente

Conclusioni sulla compressione

- La compressione deriva dal fatto che abbiamo:
 - Column Store
 - Dictionary Encoding
 - RLE Encoding
- **Compressione:** uso meno RAM e quindi **più velocità** e il modello riesce a stare nel server ☺. Scansione colonne sono più veloci
- Il valore di compressione che ci possiamo aspettare è....
 - **Non lo sa nessuno ma la risposta commerciale è 10x**
anche si può arrivare a 50x o a 2x

Conclusioni sulla compressione

- Ridurre l'utilizzo della memoria è il modo migliore per ottimizzare Tabular

CPU Register	1 ns
L1 Cache	4 ns
L2 Cache	10 ns
Main Memory	100 ns
Flash Drive	100,000 ns
Hard Drive	10,000,000 ns

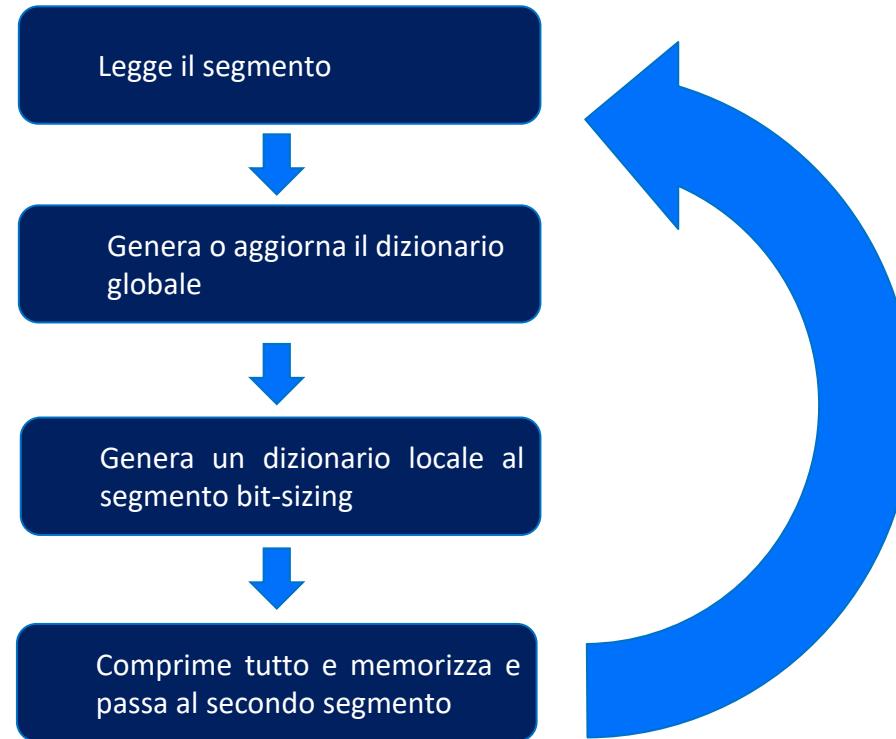
Segmentation

- Fino ad ora abbiamo visto come Vertipaq processa e comprime una colonna
- **Cosa succede con la tabella intera?**
 - In realtà Vertipaq non processa tutta la tabella prima di fare la compressione perché non avrebbe abbastanza memoria
 - **Si usa la tecnica della segmentation**

Segmentation

- Ogni tabella è divisa in segmenti (dimensione variabile)
 - 8 milioni di righe per ogni segmento in SSAS
 - 1 milione di righe in PowerPivot
- C'è un **dizionario globale** per la tabella
- **Bit-sizing** (forma compatta del dizionario) è locale ad ogni segmento
- Ci sono delle DMV per avere informazione sui segmenti

Segmentation cycle

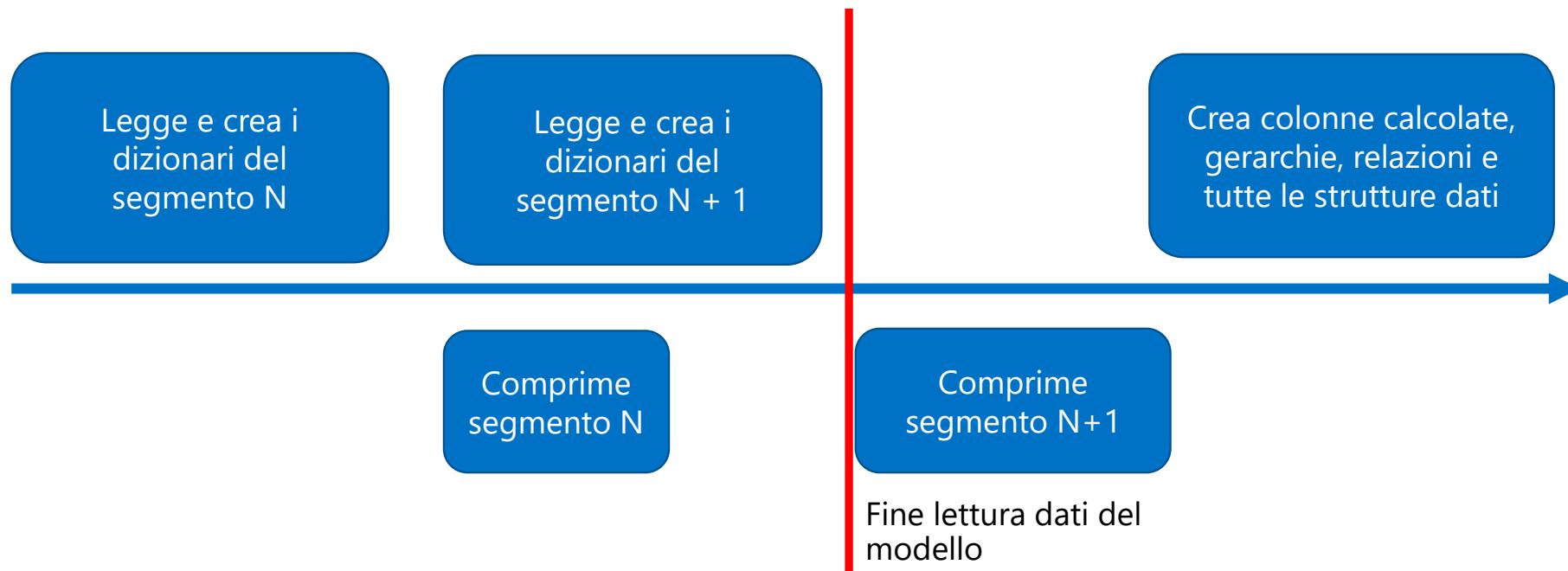


Importanza della Segmentation

- Viene usata per lavorare su un **insieme ridotto di dati per la compressione** (1 o 8 milioni di righe)
- Viene usata come base per il parallelismo **all'interno delle query**
 - Quando Vertipaq risolve una query usa un thread per ogni segmento della tabella (**per fare la scansione**)
 - Se ho meno di 8 milioni userà un solo thread perché è antipoduttivo usarne di più
 - Se ho 80 milioni di righe userà 10 thread su 10 core separati (**ideale ma impensabile per conflitto sul bus**)

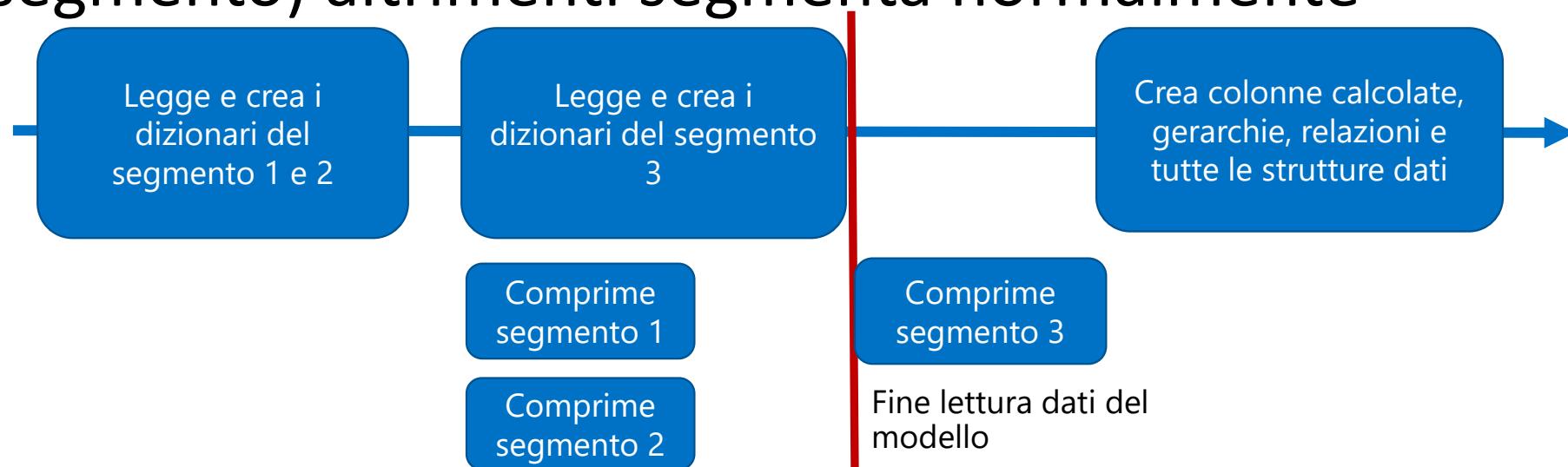
Segmentation

- Fasi della segmentazione durante il processing



Segmentation: 3 segmento

- Vertipaq cerca di ridurre il numero di segmenti da caricare
- fa un tentativo di leggere i primi due segmenti assieme (**come fosse unico**). Se ci sono 12 milioni di righe è inutile leggerli in due passi e legge direttamente 16 milioni di righe (primo segmento) altrimenti segmenta normalmente



Configurazione di tabular

- La configurazione e a livello di istanza
- DefaultSegmentRowCount (0 = default)
- ProcessingTimeboxSecPerMRow per decidere il tempo entro al quale deve ordinare

Category	Value	Type	Unit
ThreadPool \ Query \ Max Threads	0	0	0
ThreadPool \ Query \ Min Threads	0	0	0
VertiPaq \ Default Segment RowCount	0	0	0
VertiPaq \ Processing Timebox SecPerMRow	-1	-1	-1
VertiPaq \ ThreadPool \ Group Affinity			yes string
Write Transaction Idle Timeout	0	0	0

< >

Show Advanced (All) Properties

Save Only Modified Properties

Reset default

Dimensione dei segmenti

DefaultSegmentRowCount

- Default: 8M
- Necessita essere una Potenza di 2, al di sopra di 1M
- Larger: Migliore compressione e le query sono veloci
- Smaller: lavora poco e viene impostata durante la fase di processing

ProcessingTimeboxSecPerMRow

- Default: -1 = 10 sec
- Smaller: poca compressione
- Larger: migliore compressione
- Aumentare per colonne molto larghe ($\sim > 200$)

Dimensione dei segmenti e le partizioni

Le partizioni devono essere larghi più dei segmenti

Regola generale : una partizione necessita di essere un multiplo della dimensione di un segmento, più largo è il multipl

Segment Size	Partition Size	Notes
8M	8M	Perfect
8M	1M	Bad, partition too small
8M	10M	Bad, 2° segment too small
8M	24M	Good, 3 segments per partition
8M	60M	Good, many segments per partition
64M	10M	Bad, partition too small
64M	128M	Great if the database is very large

Uso memoria: processing

- **Process data:** carica, compressione, memorizzazione
- **Process recalc:** colonna calcolate, indici, relazioni, gerarchie
- **Process Full: data + recalc**
- In totale un oggetto necessita di **tre** volte del suo spazio
 - Process transazionale
 - Il vecchio cubo è tenuto in memoria e continua a rispondere
 - Nuovi dati sono processati e alla fine si scambiano i cubi
 - **Memoria per dati vecchi 1x e memoria per il processing 2x**
 - **Evitare di fare il process full o il process su singola tabella**
 - **Eseguire il ProcessClear => attenzione a fare il backup**

Uso memoria: memorizzazione

- L'uso **della memoria nella memorizzazione dipende da:**
 - Numero di colonne
 - Cardinalità di ogni colonna (valori distinct)
 - Tipo di dato (varia il dizionario)
 - Numero di righe
- Non ci sono formule per calcolare lo spazio occupato da una tabella. **L'unico modo è creare un prototipo!!!**
 - **Attenzione ad avere un prototipo con dati veri ☺ i dati nascosti sfalsano la distribuzione dei dati**

Uso memoria: le query

- La cache richiede memoria
- Le query semplici richiedono un po' di memoria
- Le query complesse richiedono molta memoria
 - Fare spooling per valori temporanei
 - Materializzare un **dataset** (se faccio una query su più colonne alla fine devo unire i risultati ☺)
 - **Problema:** in quanto molte volte può capitare che la versione materializzata sia più grande della tabella originale

Materialization

- Se vogliamo eseguire su un database colonna la seguente query:

```
SELECT SUM(num730) AS N730,[COD_Ufficio]
FROM [dbo].[Dichiarazioni730]
WHERE [COD_Utente] = 345 AND [Tipo730] = 1
GROUP BY [COD_Ufficio]
```

COD_Utente	Tipo730	Cod_Ufficio	num730
345	1	4555	234
1678	2	2345	100
345	1	6545	400
100	1	444	3

- Ci sono diverse tecniche ma agli estremi ci sono:
 - Early Materialization
 - Late Materialization

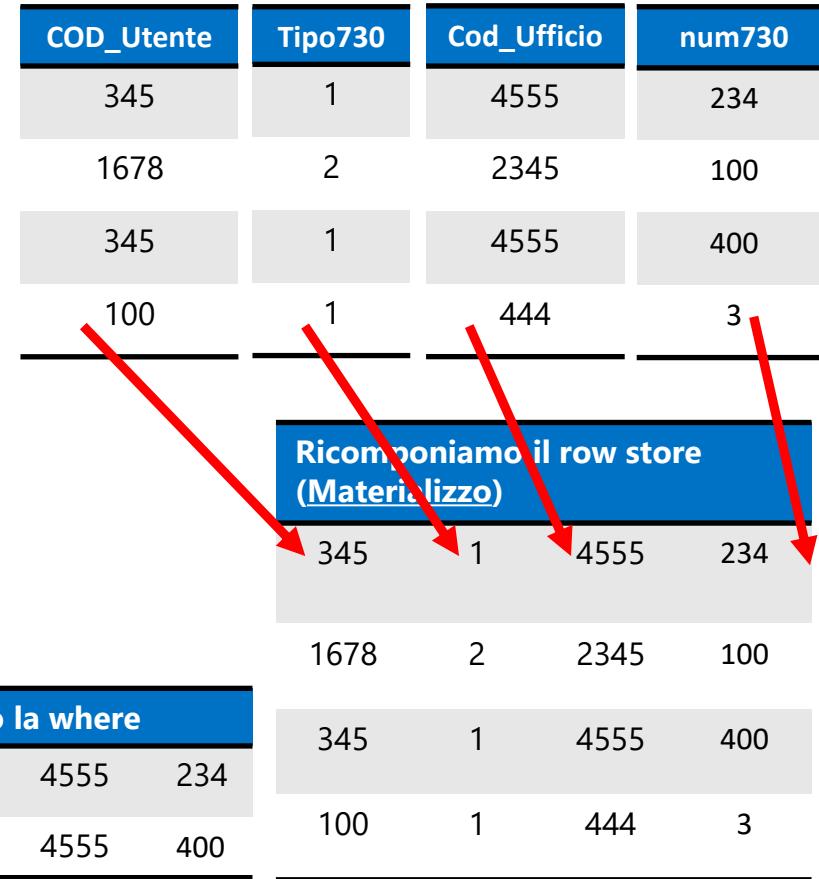
Early materialization

```
SELECT SUM(num730) AS N730,[COD_Ufficio]
FROM [dbo].[Dichiarazioni730]
WHERE [COD_Utente] = 345 AND [Tipo730] = 1
GROUP BY [COD_Ufficio]
```

La fregatura è che faccio tanto lavoro per comprimere in colonne separate e poi devo riunire tutto. **Uso tanta memoria se faccio select ***

		Proiezione per num730 e cod_ufficio	
Sommo		4555	234
4555	634	4555	400

Applico la where			
345	1	4555	234
345	1	4555	400

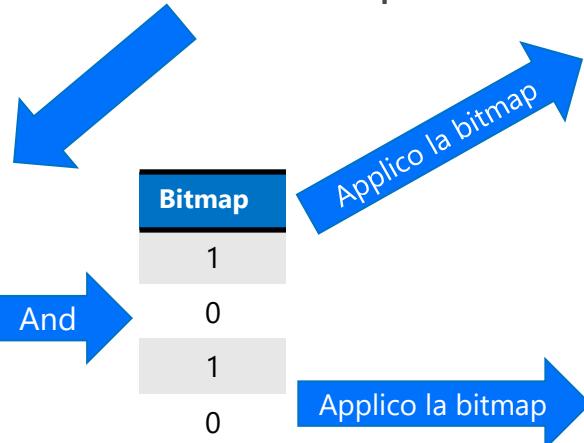


Late materialization

```
SELECT SUM(num730) AS N730,[COD_Ufficio]
FROM [dbo].[Dichiarazioni730]
WHERE [COD_Utente] = 345 AND [Tipo730] = 1
GROUP BY [COD_Ufficio]
```

Applico la clausola
where sulle due
colonne separate

Bitmap	Bitmap
1	1
0	0
1	1
0	1



COD_Utente	Tipo730	Cod_Ufficio	num730
345	1	4555	234
1678	2	2345	100
345	1	4555	400
100	1	444	3
Cod_Ufficio	Cod_Ufficio		
4555	4555		
2345	4555		
4555			
444			
num730	num730		
234	234		
100	400		
400			
3			
Materializzo			
4555	234		
4555	400		
Sommo			
4555	634		

Quando avviene?

- La materializzazione avviene per Join complessi
- La materializzazione avviene per iterazioni complesse
- Durante il salvataggio di dati temporanei

Praticamente devo sempre materializzare 😊

Demo 1

Riduzione modello

Modulo 2: Iniziamo con M (Power Query Language)

Obiettivi del modulo

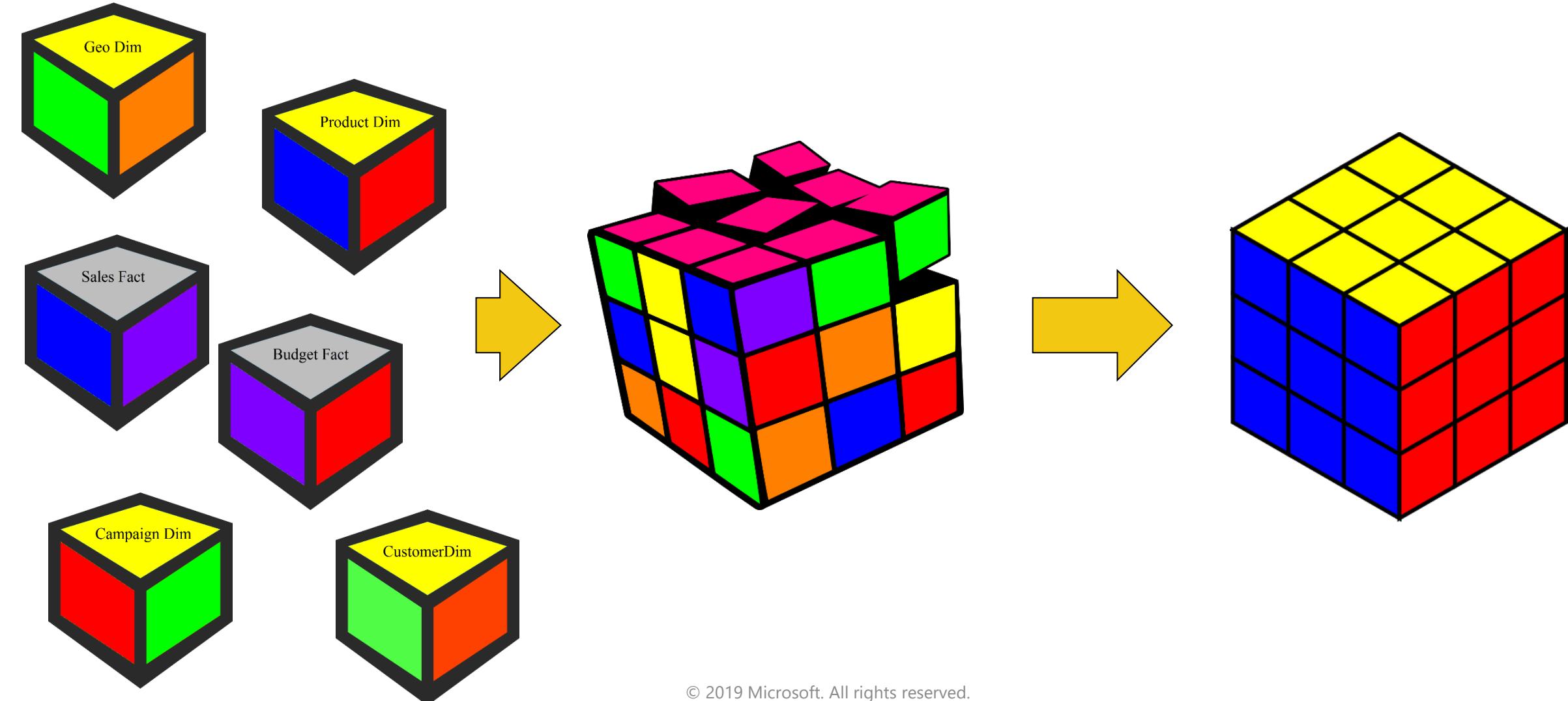
Obiettivi

- Comprendere la sintassi di base della lingua M.
- Comprendere le funzioni e la struttura delle parole chiave
- Comprendere come connettersi ai dati
- Comprendere la struttura di base dell'Editor avanzato

Agenda

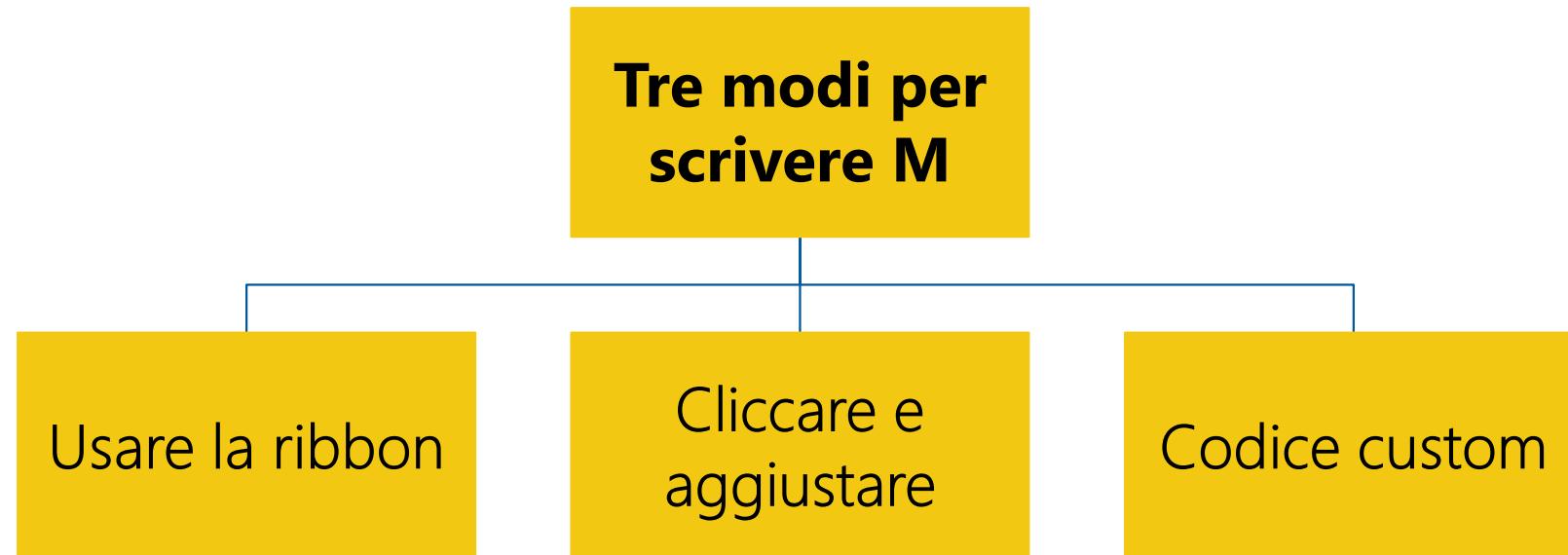
- Panoramica del linguaggio M di Power BI.
- Importazione di dati base
- Introduzione alle funzioni di testo
- Creare una dimensione degenere

M ha incredibili capacità di trasformare i dati per ottimizzarli per il modello dati



Come uso M?

- Non c'è niente di sbagliato nell'usare l'interfaccia utente per creare le tue query, ma usare M è molto più potente.
- Imparando a leggere il codice generato dall'interfaccia utente e quindi aggiornandolo in base alle proprie esigenze, è possibile realizzare trasformazioni incredibili.



Semplice

Avanzato

Concetti chiave della sintassi M



Mè la **chiave** per la Trasformazione dei dati in Power BI

Molte Trasformazioni sono a portata di clic sui **Ribbons**

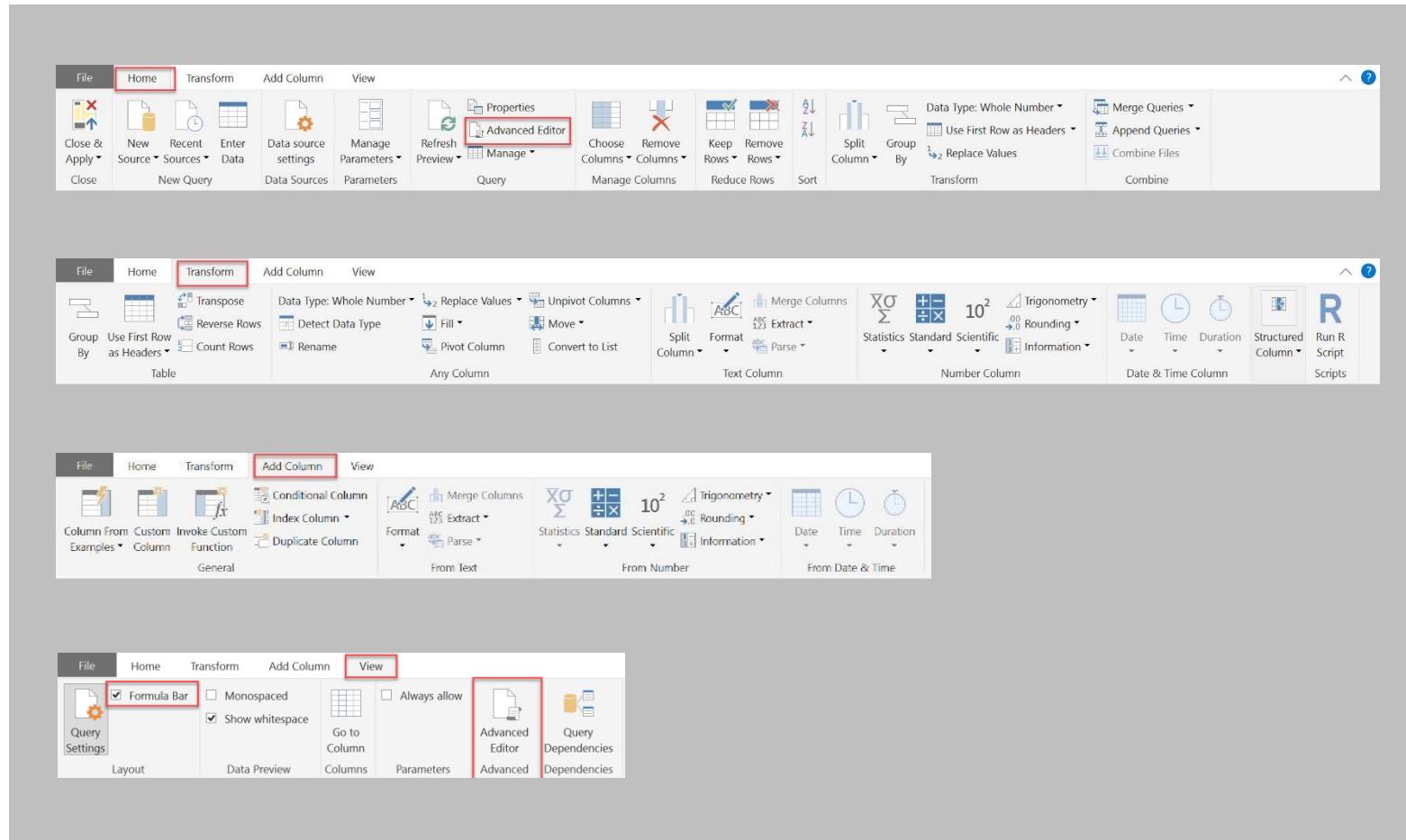
Accendere **Formula Bar** per vedere la sintassi generata da M con i click

Vedere la sintassi completa su **Advanced Editor**

M è basato su colonna

- Molte Trasformazioni accadono aggiungendo colonne

M è **CaSe SeNsItIvE!**



Concetti chiave della sintassi M



M Formulas

Trasformazioni fondamentali

- **Sintassi:**
Object.Action()
- **Oggetti:**
Table, List, Record, Text, Number, Date, etc.
- **Azioni (Camel Case):**
ToListAsync, DayOfWeek, ToNumber, FromText, etc.

M helper Words

non fare il capitalize per queste!!!

- let in each
- if then else
- true false error
- and or not
- try otherwise
- as is meta section shared type

Key Punctuation

- [Column] – Per riferirsi ad una colonna
- {List} – usare {} per indicare una lista

M # Key Words

Usato per creare nuovi artfatti e literals

- #date(2016,01,01)
#date([Year],[Month],[Day])
- #datetime(2016,02,26, 09,15,00)
- #time(09,15,00)
- #datetimezone(2013,02,26, 09,15,00, 09,00)
- #table({"X","Y"},{0,1},{1,0})
- #duration(0,1,30,0)
- #binary #infinity #nan #sections
#shared

M è CaSe SeNsItIvE!

Power Query Formula Language

Funzioni di testo

MyColumn	Text.Start	Text.End	Text.Range (Partial)	Text.Range (Full)	Text.StartsWith (abc)	Text.StartsWith (ABC)	Text.Length	Text.Contains (123)	Text.PositionOf (5)
ABC12345xyz	ABC	5xyz	234	2345xyz	FALSE	TRUE	11	TRUE	7



M											Excel											
Posizione	0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10	11
Mia Colonna	A	B	C	1	2	3	4	5	x	y	z	A	B	C	1	2	3	4	5	x	y	z

Ritorna "ABC"	Text.Start([MyColumn], 3)	LEFT([MyColumn], 3)
Ritorna "234"	Text.Range([MyColumn], 4, 3)	MID([MyColumn, 5, 3)
Trova la posizione di "5"	PositionOf([MyColumn], "5") = 7	SEARCH("5", [MyColumn]) = 8

- M è basato su indice 0, quindi Text.Range & Text.PositionOf inizia a contare da 0
- Excel MID & SEARCH sono basati su indice 1 cioè inizia a contare da 1
- Excel non fa distinzione tra maiuscole e minuscole.

Funzioni di testo

MyColumn	Text.Start	Text.End	Text.Range (Partial)	Text.Range (Full)	Text.StartsWith (abc)	Text.StartsWith (ABC)	Text.Length	Text.Contains (123)	Text.PositionOf (5)
ABC12345xyz	ABC	5xyz	234	2345xyz	FALSE	TRUE	11	TRUE	7

Sintassi funzioni M	Sintassi Excel	Output M
Text.Start([MyColumn], 3)	LEFT([MyColumn], 3)	ABC
Text.End([MyColumn], 4)	RIGHT([MyColumn], 4)	5xyz
Text.Range([MyColumn], 4, 3) *	MID([MyColumn], 5, 3) **	234
Text.Range([MyColumn], 4) *	MID([MyColumn], 5) **	2345xyz
Text.StartsWith ([MyColumn], "abc")	IF(LEFT([MyColumn], 3)="abc", TRUE(), FALSE())	FALSE (Excel -> TRUE)
Text.StartsWith ([MyColumn], "ABC")	IF(LEFT([MyColumn], 3)="ABC", TRUE(), FALSE())	TRUE
Text.Length([MyColumn])	LEN([MyColumn])	11
Text.Contains ([MyColumn], "123")	IF(ISNUMBER(SEARCH("123", [MyColumn])), TRUE(), FALSE()) **	TRUE
Text.PositionOf ([MyColumn], "5") *	SEARCH("5", [MyColumn]) **	7 (Excel -> 8)

M è CaSe SeNsItIvE!

M è basato su indice 0, quindi Text.Range & Text.PositionOf inizia a contare da 0
Excel MID & SEARCH sono basati su indice 1 cioè inizia a contare da 1

if ... then ... else

- Sintassi base:
if <test if true> **then** <result if true> **else**
<result if false>
- if può essere incatenato o nidificato
- Non sono richieste parentesi () eccetto:
 - Se il test o i risultati presentano più condizioni:
 (condition 1 **and** condition 2)
 (condition 1 **or** condition 2)
 (condition 1 **or** (condition 2 and condition 3))
- Se vengono utilizzati più if assicurarsi di avere lo stesso numero di "if", "then", "else"

Simple example (1 each of if, then & else)

```
=if [Column2] = "Key" then [Column2] else "Other"
```

Chained example (3 each of if, then & else)

```
=if Text.EndsWith([Column2],"Key") then [Column2] else  
if [Column2] = "TotalCost" then "Cost $" else  
if [Column2] = "SalesQuantity" then "Sales #" else null
```

Nested example (3 each of if, then & else)

```
=if Text.EndsWith([Column2],"Key") then  
  if [Column2] = "TotalCost" then "Cost $" else  
    if [Column2] = "SalesQuantity" then "Sales #" else null  
  else "Item #"
```

Multiple conditions example (1 each of if, then & else)

```
=if (Text.Length([Column2]) = 3 or Text.Length([Column2]) = 2 )  
then true else false
```

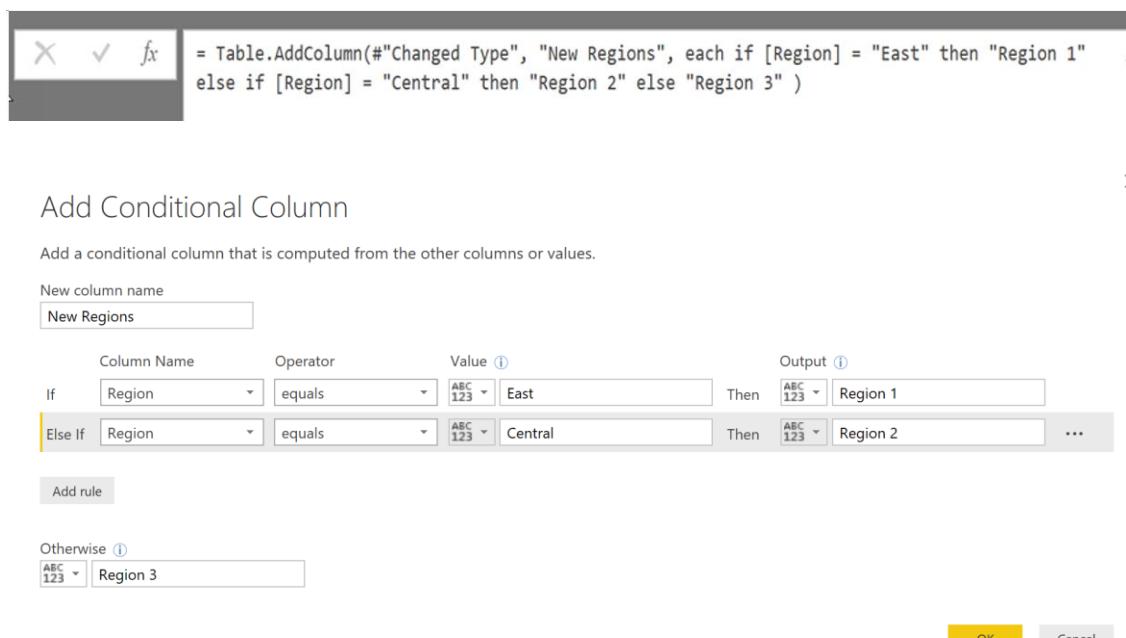
M è **CaSe SeNsItIvE!**

if ... then ... else usando Conditional e Custom

Se si utilizza la colonna condizionale o la colonna personalizzata, i risultati sono gli stessi

66	12204	Albany, NY, USA	NY	East	District #01	USA	Region 1
67	45710	Albany, OH, USA	OH	East	District #15	USA	Region 1
68	74721	Albany, OK, USA	OK	Central	District #22	USA	Region 2
69	97322	Albany, OR, USA	OR	West	District #34	USA	Region 3
70	97321	Albany, OR, USA	OR	West	District #34	USA	Region 3
71	76430	Albany, TX, USA	TX	Central	District #22	USA	Region 2

1. Come viene letto nella formula bar quando si utilizza il formato Colonna condizionale



The screenshot shows the 'Add Conditional Column' dialog. At the top, the formula bar displays:

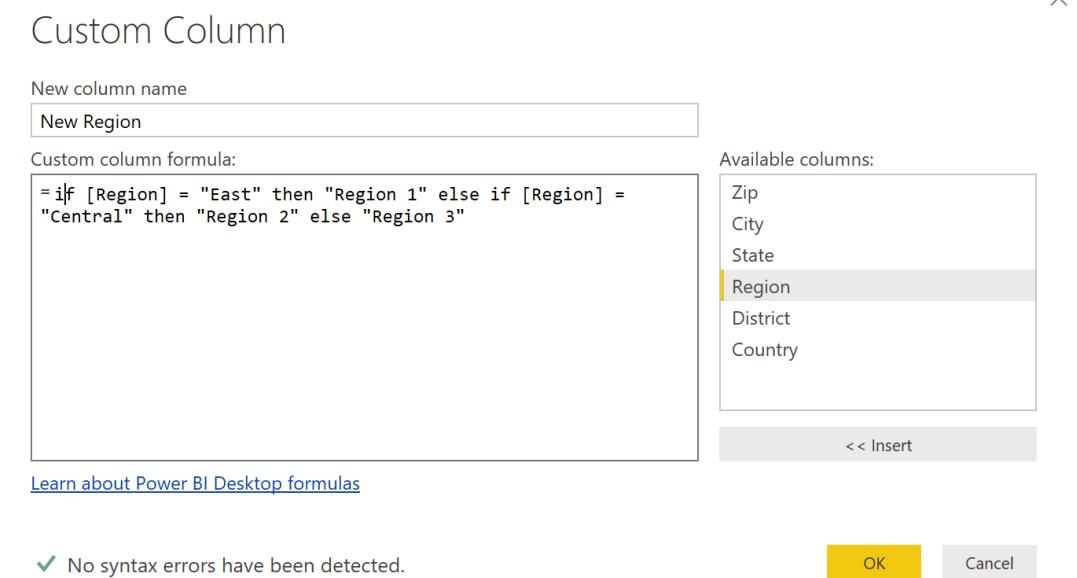
```
= Table.AddColumn(#"Changed Type", "New Regions", each if [Region] = "East" then "Region 1" else if [Region] = "Central" then "Region 2" else "Region 3" )
```

The main area of the dialog is titled 'Add Conditional Column' and contains the following fields:

- New column name: New Regions
- Column Name: Region
- Operator: equals
- Value: East
- Output: Region 1
- If Else If: Region equals Central then Region 2
- Otherwise: Region 3

At the bottom right are 'OK' and 'Cancel' buttons.

2. Quando si utilizza il formato Colonna condizionale



The screenshot shows the 'Custom Column' dialog. It includes the following sections:

- Custom Column section:
 - New column name: New Region
 - Custom column formula:

```
= if [Region] = "East" then "Region 1" else if [Region] = "Central" then "Region 2" else "Region 3"
```
- Available columns list:
 - Zip
 - City
 - State
 - Region
 - District
 - Country
- Buttons at the bottom: '<< Insert', 'OK', and 'Cancel'.

A note at the bottom left says: 'Learn about Power BI Desktop formulas'. A status message at the bottom right says: '✓ No syntax errors have been detected.'

if ... then ... else usando Columns From Examples

Utilizzare la funzionalità «columns from examples» per creare la logica if else se non si desidera scrivere la logica condizionale

 Add Column From Examples

Enter sample values to create a new column (Ctrl+Enter to apply).

Transform: if [Region] = "East" then "Region 1" else if [Region] = "Central" then "Region 2" else null

OK Cancel

	A ^B _C Zip	A ^B _C City	A ^B _C State	A ^B _C Region	A ^B _C District	A ^B _C Country	
55	22842	Mount Jackson, VA, USA	VA	East	District #07	USA	
66	22843	Mount Solon, VA, USA	VA	East	District #07	USA	
67	22844	New Market, VA, USA	VA	East	District #07	USA	
68	22845	Orkney Springs, VA, USA	VA	East	District #07	USA	
69	22846	Penn Laird, VA, USA	VA	East	District #07	USA	
70	22847	Quicksburg, VA, USA	VA	East	District #07	USA	
71	22848	Pleasant Valley, VA, USA	VA	East	District #07	USA	
72	22849	Shenandoah, VA, USA	VA	East	District #07	USA	
73	22850	Singers Glen, VA, USA	VA	East	District #07	USA	
74	22851	Stanley, VA, USA	VA	East	District #07	USA	
75	22853	Timberville, VA, USA	VA	East	District #07	USA	
76	22901	Charlottesville, VA, USA	VA	East	District #07	USA	
77	22902	Charlottesville, VA, USA	VA	East	District #07	USA	
78	22903	Charlottesville, VA, USA	VA	East	District #07	USA	
79	22904	Charlottesville, VA, USA	VA	East	District #07	USA	
80	22905	Charlottesville, VA, USA	VA	East	District #07	USA	
81	22906	Charlottesville, VA, USA	VA	East	District #07	USA	
82	67155	Wilmore, KS, USA	KS	Central	District #21	USA	
83	67156	Winfield, KS, USA	KS	Central	District #21	USA	
84	67159	Zenda, KS, USA	KS	Central	District #21	USA	
85	67201	Wichita, KS, USA	KS	Central	District #21	USA	

Custom
Region 1
Region 2
Region 2
Region 2
Region 2

Modulo 2 Lab 1: Importazione e Trasformazioni base

Cosa vogliamo?

Report su Actual & Budget Revenue; Units, Cost & Gross Profit

- VtB – Actual Variance to Budget
- YoY – Year over Year
- MoM – Current Month vs Prior Month
- One Division percent of Total

 VanArsdel

Come descriviamo i dati

- Per Product Category
- Per Product Segment
- Per Campaign
- Per Customer

Denormalizzato in un modello dati

Come si trasforma un report Excel comune in un modello di dati

- One Excel sheet with multiple columns
- What am I counting or aggregating (FactTable)

Come descriviamo i dati

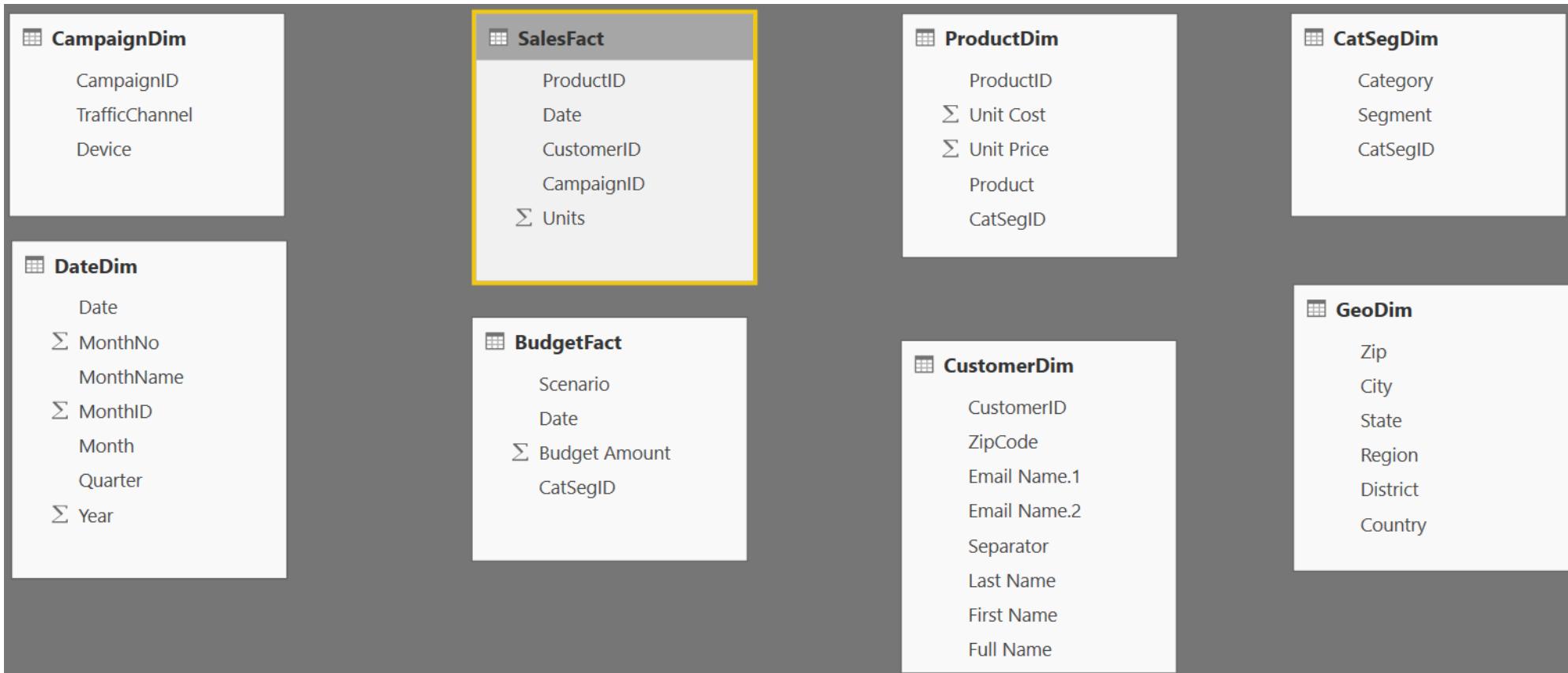
- Per Product Category
- Per Product Segment
- Per Campaign
- Per Customer

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
Product ID	Product	Category	Segment	Manufacturer ID	Manufacturer	Unit Cost	Unit Price	Date	Customer ID	Zip Code	Email	City	State	Region	District	Country	Campaign ID	Traffic Channel	Device	Units
1	676	Maximus UC-41	Urban	Convenience	VanArdel	55.5674175	76.11375	2011-03-25	070263	37013	(Farrah.Kent@xyzs.com); Kent, Farrah	Antioch, TN, USA	TN	East	District #13	USA	22	Mail	Paper	1
2	443	Maximus UM-54	Urban	Moderation	VanArdel	74.7239175	102.36375	2011-03-25	185385	02873	(Drake.Wells@xyzs.com); Wells, Drake	Wakefield, RI, USA	RI	East	District #02	USA	22	Mail	Paper	1
3	615	Maximus UC-80	Urban	Convenience	VanArdel	49.8186675	68.24475	2012-05-14	212645	21111	(Jacob.Santiago@xyzs.com); Santiago, Jacob	Monkton, MD, USA	MD	East	District #05	USA	22	Mail	Paper	1
4	615	Maximus UC-80	Urban	Convenience	VanArdel	49.8186675	68.24475	2012-05-14	070666	23625	(Hilary.Coller@xyzs.com); Collier, Hilary	Anderson, SC, USA	SC	East	District #11	USA	22	Mail	Paper	1
5	615	Maximus UC-80	Urban	Convenience	VanArdel	49.8186675	68.24475	2012-05-14	114453	60585	(Chester.Mitchell@xyzs.com); Mitchell, Chester	Plainfield, IL, USA	IL	Central	District #31	USA	22	Mail	Paper	1
6	615	Maximus UC-80	Urban	Convenience	VanArdel	49.8186675	68.24475	2012-05-14	221670	68503	(Sage.Yang@xyzs.com); Yang, Sage	Lincoln, NE, USA	NE	Central	District #30	USA	22	Mail	Paper	1
7	633	Maximus UC-98	Urban	Convenience	VanArdel	41.3871675	56.63475	2012-05-14	026974	78247	(Hammett.Gill@xyzs.com); Gill, Hammett	San Antonio, TX, USA	TX	Central	District #23	USA	22	Mail	Paper	1
8	443	Maximus UM-48	Urban	Moderation	VanArdel	65.1486675	89.24475	2012-06-03	266392	65058	(Rebecca.Cleveland@xyzs.com); Cleveland, Rebecca	Meta, MO, USA	MO	Central	District #28	USA	22	Mail	Paper	1
9	487	Maximus UM-92	Urban	Moderation	VanArdel	80.4786675	110.24475	2012-06-03	224757	30523	(Quintessa.Ochoa@xyzs.com); Ochoa, Quintessa	Clarksville, GA, USA	GA	East	District #09	USA	22	Mail	Paper	1
10	615	Maximus UC-80	Urban	Convenience	VanArdel	49.8166675	68.24475	2012-06-03	168009	17313	(Wallace.Bender@xyzs.com); Bender, Wallace	Dallastown, PA, USA	PA	East	District #05	USA	22	Mail	Paper	1
11	615	Maximus UC-80	Urban	Convenience	VanArdel	49.8186675	68.24475	2012-06-03	154433	38003	(Iliana.Dunlap@xyzs.com); Dunlap, Iliana	Federal Way, WA, USA	WA	West	District #34	USA	22	Mail	Paper	1
12	443	Maximus UM-48	Urban	Moderation	VanArdel	65.1486675	89.24475	2012-06-04	034374	44266	(Earo.Crosby@xyzs.com); Crosby, Earo	Ravenna, OH, USA	OH	East	District #14	USA	22	Mail	Paper	1
13	549	Maximus UC-14	Mix	Productivity	VanArdel	40.2374175	55.11375	2012-06-04	173640	02304	(Sophia.Leonard@xyzs.com); Leonard, Sophia	Providence, RI, USA	RI	East	District #02	USA	22	Mail	Paper	1
14	615	Maximus UC-80	Urban	Convenience	VanArdel	49.8186675	68.24475	2012-06-04	181331	07032	(Joelle.Lee@xyzs.com); Lee, Joelle	Kearny, NJ, USA	NJ	East	District #03	USA	22	Mail	Paper	1
15	615	Maximus UC-80	Urban	Convenience	VanArdel	49.8186675	68.24475	2012-06-04	064968	28025	(Upton.Barrett@xyzs.com); Barrett, Upton	Concord, NC, USA	NC	East	District #11	USA	22	Mail	Paper	1

Importazione dettagliata dei dati

Hai accesso ai dati per il tuo modello, ma non è nella giusta "forma"

- Inizia con le query di base Get Data per portare i dati nell'Editor di query
- Utilizzare una combinazione di UI e M per trasformare i dati in un modello di dati semplificato



Modulo 2 Lab: Importa più tabelle da un'unica fonte

Se l'origine rispecchia la tabella richiesta dal modello, importare più tabelle contemporaneamente

CampaignDim

	1 ² 3 CampaignID	A ^B C TrafficChannel	A ^B C Device
1		1 Organic Search	Desktop
2		2 Organic Search	Mobile
3		3 Organic Search	Tablet
4		4 SEO	Desktop

GeoDim

	1 ² 3 Zip	A ^B C City	A ^B C State	A ^B C Region	A ^B C District	A ^B C Country
1	22654	Star Tannery, VA, USA	VA	East	District #07	USA
2	22655	Stephens City, VA, USA	VA	East	District #07	USA
3	22656	Stephenson, VA, USA	VA	East	District #07	USA
4	22657	Strasburg, VA, USA	VA	East	District #07	USA

ProductDim

	1 ² 3 ProductID	A ^B C Product	A ^B C Category	A ^B C Segment	1 ² 3 ManufacturerID	A ^B C Manufacturer
1	392	Maximus RP-01	Rural	Productivity	7	VanArsdel
2	393	Maximus RP-02	Rural	Productivity	7	VanArsdel
3	394	Maximus RS-01	Rural	Select	7	VanArsdel
4	396	Maximus UM-01	Accessory	Accessory	7	VanArsdel

DateDim

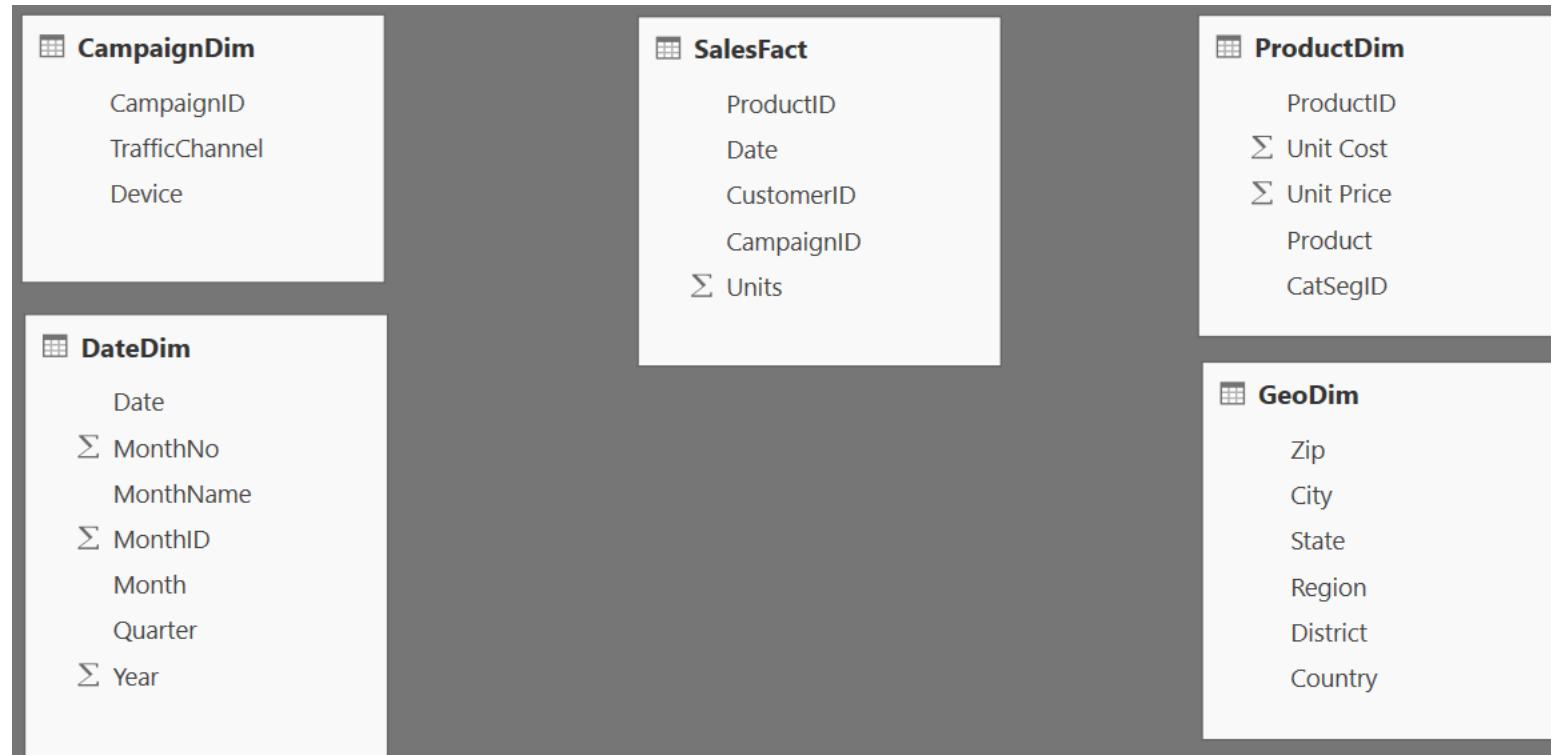
	1 ² 3 Date	1 ² 3 MonthNo	A ^B C MonthName	1 ² 3 MonthID	Month	A ^B C Quarter	1 ² 3 Year
1	1/1/2011	1	Jan	201101	1/11/2016	Q1	2011
2	1/2/2011	1	Jan	201101	1/11/2016	Q1	2011
3	1/3/2011	1	Jan	201101	1/11/2016	Q1	2011
4	1/4/2011	1	Jan	201101	1/11/2016	Q1	2011

Sales

	1 ² 3 ProductID	Date	1 ² 3 CustomerID	1 ² 3 CampaignID	1 ² 3 Units
1	676	9/25/2011	70283		22
2	449	9/25/2011	195385		22
3	615	5/14/2012	212645		22
4	615	5/14/2012	70666		22
5	615	5/14/2012	114459		22

Questo è lo stato attuale del nostro modello di dati

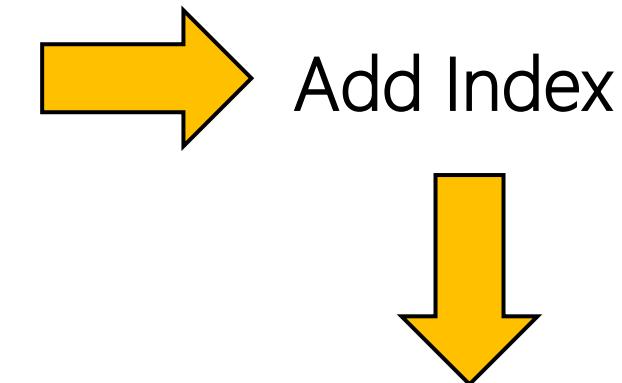
- Le relazioni verranno "disegnate" quando il modello viene caricato



Modulo 2 Lab: Creare CatSegDim (Category Segment)

Estrai delle combinazioni uniche di categorie e segmenti dalla dimensione del prodotto

	1 ² ProductID	A ^B Product	A ^B Category	A ^B Segment	1 ² ManufacturerID	A ^B Manufacturer
1	392	Maximus RP-01	Rural	Productivity	7	VanArsdel
2	393	Maximus RP-02	Rural	Productivity	7	VanArsdel
3	394	Maximus RS-01	Rural	Select	7	VanArsdel
4	396	Maximus UM-01	Accessory	Accessory	7	VanArsdel
5	397	Maximus UM-02	Accessory	Accessory	7	VanArsdel



CatSegDim è una dimensione degenere viene estratto dai dati nella tabella dei fatti.

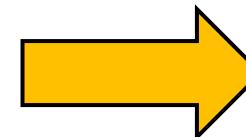
Invece di creare due dimensioni, una per categoria di prodotto e una per segmento di prodotto, stiamo usando le combinazioni distinte dei due campi per creare una nuova dimensione combinata.

	1 ² CatSegID	A ^B Category	A ^B Segment
1	1	Rural	Productivity
2	2	Rural	Select
3	3	Accessory	Accessory
4	4	Urban	Moderation
5	5	Urban	Regular
6	6	Urban	Extreme
7	7	Mix	All Season
8	8	Mix	Productivity
9	9	Youth	Youth
10	10	Urban	Convenience

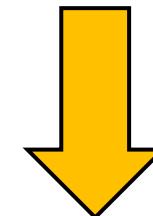
Modulo 2 Lab: Aggiorna la Dim Product

Applicare il nuovo campo CatSegID alla dimensione del prodotto

	ProductID	Product	Category	Segment	ManufacturerID	Manufacturer
1	392	Maximus RP-01	Rural	Productivity	7	VanArtsdel
2	393	Maximus RP-02	Rural	Productivity	7	VanArtsdel
3	394	Maximus RS-01	Rural	Select	7	VanArtsdel
4	396	Maximus UM-01	Accessory	Accessory	7	VanArtsdel
5	397	Maximus UM-02	Accessory	Accessory	7	VanArtsdel



Replace with
Index



	ProductID	CatSegID	Product	Unit Price	Unit Cost
1	392	1	Maximus RP-01	51.05625	37.2710625
2	393	1	Maximus RP-02	51.05625	37.2710625
3	394	2	Maximus RS-01	164.05725	119.7617925
4	396	3	Maximus UM-01	90.79875	66.2830875
5	397	3	Maximus UM-02	149.61975	109.2224175
6	398	3	Maximus UM-03	157.49475	114.9711675
7	400	3	Maximus UM-05	84.49875	61.6840875
8	402	3	Maximus UM-07	102.36975	74.7299175
9	403	3	Maximus UM-08	113.85675	83.1154275
10	404	3	Maximus UM-09	186.36975	136.0499175
11	405	3	Maximus UM-10	191.61975	139.8824175
12	406	3	Maximus UM-11	191.61975	139.8824175

Modulo 2 Lab: Creare la dimensione Customer

Utilizzo di una colonna di garbage (immondizia) per compilare CustomerDim

	CustomerID	ZipCode	Email Name
1	1	90250	(Meghan.Alexander@xyz...): Alexander, Meghan
2	2	90250	(Leah.Kemp@xyz...): Kemp, Leah
3	3	90250	(Tamekah.Stevens@xyz...): Stevens, Tamekah
4	4	90250	(Dexter.Haney@xyz...): Haney, Dexter
5	5	90250	(Jonah.Moon@xyz...): Moon, Jonah
6	6	90250	(Brock.Burnett@xyz...): Burnett, Brock
7	7	90250	(Lamar.Daugherty@xyz...): Daugherty, Lamar
8	8	90250	(Dorian.Turner@xyz...): Turner, Dorian
9	9	90250	(Olympia.Rodriguez@xyz...): Rodriguez, Olympia
10	10	90250	(Colby.Snow@xyz...): Snow, Colby
11	11	90250	(Elizabeth.Acosta@xyz...): Acosta, Elizabeth
12	12	90250	(Chase.Garrett@xyz...): Garrett, Chase



In una situazione del mondo reale, dovresti semplicemente utilizzare Dividi per delimitatore anziché utilizzare Text.PositionOf(), ma questo esercizio ci consente di esplorare alcune funzioni di M:

Text.PositionOf()

Text.Start()

Text.Range()

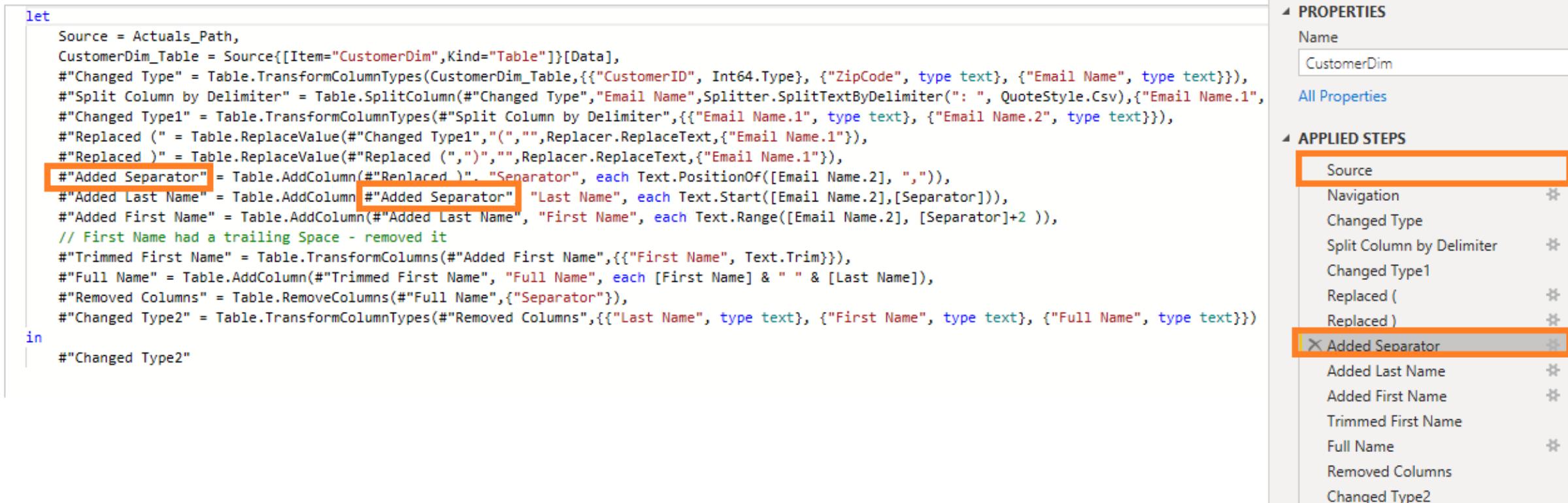
	CustomerID	Email	ZipCode	First Name	Last Name	Full Name
1		1 Meghan.Alexander@xyz...	90250	Alexander	Meghan	Alexander Meghan
2		2 Leah.Kemp@xyz...	90250	Kemp	Leah	Kemp Leah
3		3 Tamekah.Stevens@xyz...	90250	Stevens	Tamekah	Stevens Tamekah
4		4 Dexter.Haney@xyz...	90250	Haney	Dexter	Haney Dexter
5		5 Jonah.Moon@xyz...	90250	Moon	Jonah	Moon Jonah
6		6 Brock.Burnett@xyz...c...	90250	Burnett	Brock	Burnett Brock
7		7 Lamar.Daugherty@xyz...	90250	Daugherty	Lamar	Daugherty Lamar
8		8 Dorian.Turner@xyz...c...	90250	Turner	Dorian	Turner Dorian
9		9 Olympia.Rodriguez@xyz...	90250	Rodriguez	Olympia	Rodriguez Olympia
10		10 Colby.Snow@xyz...	90250	Snow	Colby	Snow Colby
11		11 Elizabeth.Acosta@xyz...	90250	Acosta	Elizabeth	Acosta Elizabeth
12		12 Chase.Garrett@xyz...c...	90250	Garrett	Chase	Garrett Chase

Modulo 2 Lab: Editor avanzato

L'editor avanzato contiene tutto il codice generato dall'interfaccia utente (interfaccia utente) e il codice aggiornato nella formula bar.

- Ogni riga di codice nell'editor avanzato ha la sintassi: **Variabile = riga di codice**
- Ogni **riga successiva fa riferimento** alla variabile in una riga **di codice precedente**

CustomerDim



The screenshot shows the Power BI Advanced Editor interface. On the left, there is a code editor window containing M code for a query named "CustomerDim". The code is as follows:

```
let
    Source = Actuals_Path,
    CustomerDim_Table = Source{[Item="CustomerDim",Kind="Table"]}[Data],
    #"Changed Type" = Table.TransformColumnTypes(CustomerDim_Table,{{"CustomerID", Int64.Type}, {"ZipCode", type text}, {"Email Name", type text}}),
    #"Split Column by Delimiter" = Table.SplitColumn(#"Changed Type","Email Name",Splitter.SplitTextByDelimiter(":", QuoteStyle.Csv),{"Email Name.1", "Email Name.2"}),
    #"Changed Type1" = Table.TransformColumnTypes(#"Split Column by Delimiter",{{"Email Name.1", type text}, {"Email Name.2", type text}}),
    #"Replaced (" = Table.ReplaceValue(#"Changed Type1",",\"",Replacer.ReplaceText,{"Email Name.1"}),
    #"Replaced )" = Table.ReplaceValue(#"Replaced (",")","",Replacer.ReplaceText,{"Email Name.1"}),
    #"Added Separator" = Table.AddColumn(#"Replaced )", "Separator", each Text.PositionOf([Email Name.2], ",")),
    #"Added Last Name" = Table.AddColumn(#"Added Separator", "Last Name", each Text.Start([Email Name.2],[Separator])),
    #"Added First Name" = Table.AddColumn(#"Added Last Name", "First Name", each Text.Range([Email Name.2], [Separator]+2 )),
    // First Name had a trailing Space - removed it
    #"Trimmed First Name" = Table.TransformColumns(#"Added First Name",{{"First Name", Text.Trim}}),
    #"Full Name" = Table.AddColumn(#"Trimmed First Name", "Full Name", each [First Name] & " " & [Last Name]),
    #"Removed Columns" = Table.RemoveColumns(#"Full Name",{"Separator"}),
    #"Changed Type2" = Table.TransformColumnTypes(#"Removed Columns",{{"Last Name", type text}, {"First Name", type text}, {"Full Name", type text}})
in
    #"Changed Type2"
```

On the right side of the editor, there is a sidebar titled "Query Settings" which includes sections for "PROPERTIES" and "APPLIED STEPS". The "APPLIED STEPS" section lists the steps taken to transform the data, with some steps highlighted by orange boxes. The steps listed are:

- Source
- Navigation
- Changed Type
- Split Column by Delimiter
- Changed Type1
- Replaced (
- Replaced)
- Added Separator
- Added Last Name
- Added First Name
- Trimmed First Name
- Full Name
- Removed Columns
- Changed Type2

- **NON** è obbligatorio che una riga di codice faccia riferimento alla variabile nella riga di codice IMMEDIATAMENTE precedente.
 - è possibile creare variabili (hard coded o dinamiche) e usarle successivamente.
- // commenti a riga singola prefissando la riga con "://" o / * commento multiplo / *
- Se il nome di una variabile **contiene uno spazio ()** o **un carattere speciale (_ - &)**, devono essere circondati da # "nome variabile"
- E' più facile rinominare un passaggio in "Impostazioni query" facendo clic con il pulsante destro del mouse e selezionando Rinomina, piuttosto che provare a rinominare direttamente nell'editor avanzato (poiché è necessario assicurarsi di correggere entrambi gli aspetti del nome)
- Puoi anche modificare l'ultima riga "Changed Type2" con uno degli altri nomi a sinistra del = e quindi mostrare in che modo ciò influisce su di esso.

1. Non è possibile importare più tabelle da un'unica fonte (T / F)?
2. Perché dovremmo aver bisogno di una dimensione del fiocco di neve?
3. Quale sintassi useresti per scoprire se una colonna ha "Abc" come i suoi primi tre caratteri?
4. O gli ultimi tre?

Modulo 3: Trasformazioni M avanzate

Obiettivi

- Comprendere le trasformazioni M.
- Comprendere Unisci e altre query combinate
- Scopri come utilizzare più query in una trasformazione avanzata
- Utilizzo della profilazione dei dati

Agenda

- Creazione delle trasfromazioni M chiave di Power BI
- Tipi di merge in M di Power BI
- Crea BudgetFact usando un approccio multi-query

Trasformazioni principali

Transpose



Converti righe in colonne e colonne in righe

Sintassi:

Table.Transpose([Previous](#))

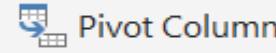
- Vengono perse le intestazioni di colonna

	ABC Color	ABC Letter	123 Value
1	Red	A	1
2	Red	B	2
3	Blue	C	3
4	Blue	D	4
5	Green	E	5

	ABC Column1	ABC Column2	ABC Column3	ABC Column4	ABC Column5
1	Red	Red	Blue	Blue	Green
2	A	B	C	D	E
3	1	2	3	4	5



Pivot Column



Converti la colonna selezionata a valori di intestazioni di colonna

Sintassi:

Table.Pivot([Previous](#),List.Distinct([Previous](#)[Color]), "Color", "Value", List.Sum)

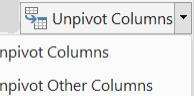
- Il valore nella colonna [Color] è convertita in intestazioni di colonna
- Il valore nella colonna [Value] è riempita dove applicabile

	ABC Color	ABC Letter	123 Value
1	Red	A	1
2	Red	B	2
3	Blue	C	3
4	Blue	D	4
5	Green	E	5



	ABC Letter	123 Red	123 Blue	123 Green
1	A	1	null	null
2	B	2	null	null
3	C	null	3	null
4	D	null	4	null
5	E	null	null	5

Unpivot Columns



Converti le intestazioni di colonna selezionate in una colonna di valori

Sintassi:

- Table.UnpivotOtherColumns([Previous](#), {"Letter", "Attribute", "Value"})
- Le intestazioni di colonna diventano valori di colonna [Attribute]
- I valori unpivoted diventano la colonna [Value]

	ABC Letter	123 Red	123 Blue	123 Green
1	A	1	null	null
2	B	2	null	null
3	C	null	3	null
4	D	null	4	null
5	E	null	null	5



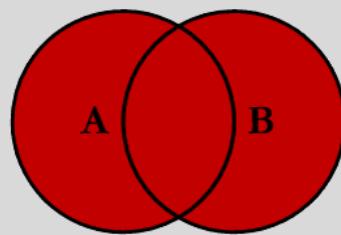
	ABC Letter	ABC Attribute	1.2 Value
1	A	Red	1
2	B	Red	2
3	C	Blue	3
4	D	Blue	4
5	E	Green	5

Nota: [Previous](#) => Nome del passaggio precedente nella query

Tipi di join – Tipi di merge

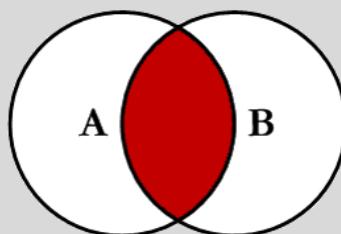
Full Outer (JoinKind.FullOuter)

Mostra **tutte le righe da entrambe le queries**. Come SQL "Union All"



Inner (JoinKind.Inner)

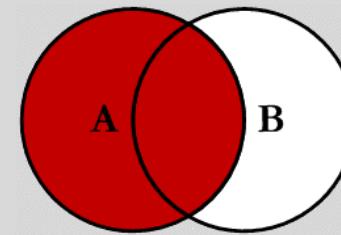
Limita le righe SOLO a quelle che corrispondono nelle colonne unidi match



Left Outer (JoinKind.LeftOuter)

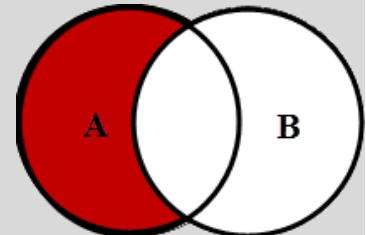
Default per il Merge

Mostra le righe dalla prima query (**A**) e gli attribute della seconda query (**B**) basandosi sul colonne di match.



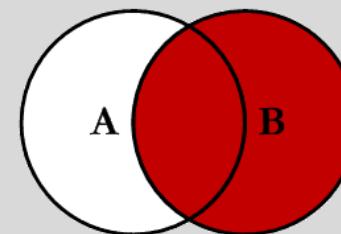
Left Anti (JoinKind.LeftAnti)

Mostra solo le righe della prima query(**A**) dove non ci sono match nella seconda query (**B**) basandosi sulle colonne di match



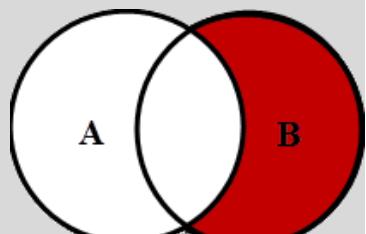
Right Outer (JoinKind.RightOuter)

Mostra tutte le righe della seconda query (**B**) e gli attribute della prima query (**A**) basandosi sulle colonne di match



Right Anti (JoinKind.RightAnti)

Mostra solo le righe della seconda query (**B**) dove non ci sono match nella prima query (**A**) basandosi sulle colonne di match



Nota: È possibile utilizzare una o più colonne per creare il join

Tipi di join – Tipi di merge

L'immagine seguente mostra le tabelle prima della merge e dopo la merge:

Table 01: Left Table

Employees who attended

DAX Basics San Jose Conference

EmployeeNameSanJose

Sol Marroquin

Kiera Mcfall

Raven Beatty

Elinore Dees

Wei Lockwood

Donald Eldridge

Claudio Beam

Angelita Packer

Reyna Luke

Beaulah Wenger

Malvina Hamer

Vivan Keeney

Table 02: Right Table

Employees who attended

DAX Basics Bellevue Conference

EmployeeName

Raven Beatty

Roxanna Mercier

Fanny Denning

Lesha Nobles

Wei Lockwood

Gertrudis Fitzpatrick

Angelita Packer

Beaulah Wenger

Malvina Hamer

Bernita Crutcher

Shiela Anaya

Yolonda Armstead

Who Attended Both Conferences?

AND Logical Test

Inner Join

EmployeeNameSanJose

Raven Beatty

Wei Lockwood

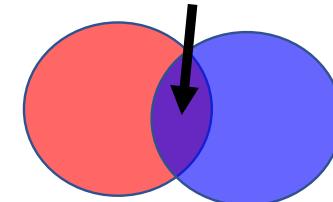
Angelita Packer

Beaulah Wenger

Malvina Hamer

Logic:

AND Logical Test



PQ Name:

Inner Join

Tipi di join – Tipi di merge

L'immagine seguente mostra le tabelle prima della merge e dopo la merge:

Table 01:
Product Table = Left Table = dProduct
with Supplier ID Column (Foreign Key)

Product	SupplierID	Price	Cost
Aspen	CO	23	11
Carlota	GB	26	12.75
Majestic Beaut	GB	29	15.85
Quad	GB	43	22.5
Sunshine	CO	19	1.25
Kangaroo	CC	14	6.95

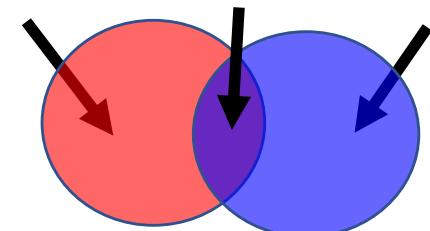
Product	SupplierID	Price	Cost	SupplierID.1	Name	City	State
Aspen	CO	23	11	CO	Colorado Boomerangs	Gunnison	CO
Carlota	GB	26	12.75	GB	Gel Boomerangs	Oakland	CA
Majestic Beaut	GB	29	15.85	GB	Gel Boomerangs	Oakland	CA
Quad	GB	43	22.5	GB	Gel Boomerangs	Oakland	CA
Sunshine	CO	19	1.25	CO	Colorado Boomerangs	Gunnison	CO
Kangaroo	CC	14	6.95				
				DB	Darnell Booms	Manchester	MA

Table 02:
Supplier Table = Right Table = dSupplier
with Supplier ID Column (Primary Key)

SupplierID	Name	Column	City	State
CO	Colorado Boomerangs	Gunnison	CO	
GB	Gel Boomerangs	Oakland	CA	
DB	Darnell Booms	Manchester	MA	

Logic:

OR Logical Test

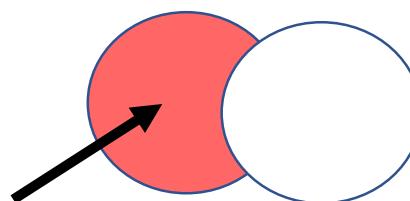


PQ Name:

Full Outer

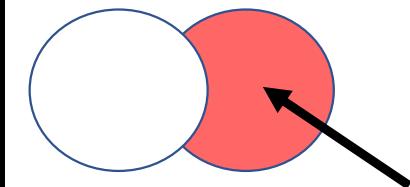
Tipi di join – Tipi di merge

L'immagine seguente mostra le tabelle prima della merge e dopo la merge

Employees who attended DAX Basics San Jose Conference	Employees who attended DAX Basics Bellevue Conference	In Table 01, NOT in Table 02 Left-Anti Join / Merge																																	
EmployeeNameSanJose <div style="background-color: #e6f2ff; padding: 2px;"> <input type="button" value="▼"/> </div> <table border="1"> <tr><td>Sol Marroquin</td></tr> <tr><td>Kiera Mcfall</td></tr> <tr><td>Raven Beatty</td></tr> <tr><td>Elinore Dees</td></tr> <tr><td>Wei Lockwood</td></tr> <tr><td>Donald Eldridge</td></tr> <tr><td>Claudio Beam</td></tr> <tr><td>Angelita Packer</td></tr> <tr><td>Reyna Luke</td></tr> <tr><td>Beaulah Wenger</td></tr> <tr><td>Malvina Hamer</td></tr> <tr><td>Bernita Crutcher</td></tr> <tr><td>Shiela Anaya</td></tr> <tr><td>Yolonda Armstead</td></tr> </table>	Sol Marroquin	Kiera Mcfall	Raven Beatty	Elinore Dees	Wei Lockwood	Donald Eldridge	Claudio Beam	Angelita Packer	Reyna Luke	Beaulah Wenger	Malvina Hamer	Bernita Crutcher	Shiela Anaya	Yolonda Armstead	EmployeeNameBellevue <div style="background-color: #e6f2ff; padding: 2px;"> <input type="button" value="▼"/> </div> <table border="1"> <tr><td>Raven Beatty</td></tr> <tr><td>Roxanna Mercier</td></tr> <tr><td>Fanny Denning</td></tr> <tr><td>Lesha Nobles</td></tr> <tr><td>Wei Lockwood</td></tr> <tr><td>Gertrudis Fitzpatrick</td></tr> <tr><td>Angelita Packer</td></tr> <tr><td>Beaulah Wenger</td></tr> <tr><td>Malvina Hamer</td></tr> <tr><td>Bernita Crutcher</td></tr> <tr><td>Shiela Anaya</td></tr> <tr><td>Yolonda Armstead</td></tr> </table>	Raven Beatty	Roxanna Mercier	Fanny Denning	Lesha Nobles	Wei Lockwood	Gertrudis Fitzpatrick	Angelita Packer	Beaulah Wenger	Malvina Hamer	Bernita Crutcher	Shiela Anaya	Yolonda Armstead	EmployeeNameSanJose <div style="background-color: #e6f2ff; padding: 2px;"> <input type="button" value="▼"/> </div> <table border="1"> <tr><td>Sol Marroquin</td></tr> <tr><td>Kiera Mcfall</td></tr> <tr><td>Elinore Dees</td></tr> <tr><td>Donald Eldridge</td></tr> <tr><td>Claudio Beam</td></tr> <tr><td>Reyna Luke</td></tr> <tr><td>Vivan Keeney</td></tr> </table> <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <p><u>Logic:</u> Only in Left Table</p>  <p><u>PQ Name:</u> Left Anti</p> </div>	Sol Marroquin	Kiera Mcfall	Elinore Dees	Donald Eldridge	Claudio Beam	Reyna Luke	Vivan Keeney
Sol Marroquin																																			
Kiera Mcfall																																			
Raven Beatty																																			
Elinore Dees																																			
Wei Lockwood																																			
Donald Eldridge																																			
Claudio Beam																																			
Angelita Packer																																			
Reyna Luke																																			
Beaulah Wenger																																			
Malvina Hamer																																			
Bernita Crutcher																																			
Shiela Anaya																																			
Yolonda Armstead																																			
Raven Beatty																																			
Roxanna Mercier																																			
Fanny Denning																																			
Lesha Nobles																																			
Wei Lockwood																																			
Gertrudis Fitzpatrick																																			
Angelita Packer																																			
Beaulah Wenger																																			
Malvina Hamer																																			
Bernita Crutcher																																			
Shiela Anaya																																			
Yolonda Armstead																																			
Sol Marroquin																																			
Kiera Mcfall																																			
Elinore Dees																																			
Donald Eldridge																																			
Claudio Beam																																			
Reyna Luke																																			
Vivan Keeney																																			

Tipi di join – Tipi di merge

L'immagine seguente mostra le tabelle prima della merge e dopo la merge

Table 01: Employees who attended DAX Basics San Jose Conference	Table 02: Employees who attended DAX Basics Bellevue Conference	Who Attended Only Bellevue Conference? In Table 02, NOT in Table 01 Right-Anti-Join Merge																															
EmployeeNameSanJose <table border="1"> <tr><td>Sol Marroquin</td></tr> <tr><td>Kiera Mcfall</td></tr> <tr><td>Raven Beatty</td></tr> <tr><td>Elinore Dees</td></tr> <tr><td>Wei Lockwood</td></tr> <tr><td>Donald Eldridge</td></tr> <tr><td>Claudio Beam</td></tr> <tr><td>Angelita Packer</td></tr> <tr><td>Reyna Luke</td></tr> <tr><td>Beaulah Wenger</td></tr> <tr><td>Malvina Hamer</td></tr> <tr><td>Vivan Keeney</td></tr> </table>	Sol Marroquin	Kiera Mcfall	Raven Beatty	Elinore Dees	Wei Lockwood	Donald Eldridge	Claudio Beam	Angelita Packer	Reyna Luke	Beaulah Wenger	Malvina Hamer	Vivan Keeney	EmployeeNameBellevue <table border="1"> <tr><td>Raven Beatty</td></tr> <tr><td>Roxanna Mercier</td></tr> <tr><td>Fanny Denning</td></tr> <tr><td>Lesha Nobles</td></tr> <tr><td>Wei Lockwood</td></tr> <tr><td>Gertrudis Fitzpatrick</td></tr> <tr><td>Angelita Packer</td></tr> <tr><td>Beaulah Wenger</td></tr> <tr><td>Malvina Hamer</td></tr> <tr><td>Bernita Crutcher</td></tr> <tr><td>Shiela Anaya</td></tr> <tr><td>Yolonda Armstead</td></tr> </table>	Raven Beatty	Roxanna Mercier	Fanny Denning	Lesha Nobles	Wei Lockwood	Gertrudis Fitzpatrick	Angelita Packer	Beaulah Wenger	Malvina Hamer	Bernita Crutcher	Shiela Anaya	Yolonda Armstead	EmployeeNameBellevue <table border="1"> <tr><td>Roxanna Mercier</td></tr> <tr><td>Fanny Denning</td></tr> <tr><td>Lesha Nobles</td></tr> <tr><td>Gertrudis Fitzpatrick</td></tr> <tr><td>Bernita Crutcher</td></tr> <tr><td>Shiela Anaya</td></tr> <tr><td>Yolonda Armstead</td></tr> </table> <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <p><u>Logic:</u> Only in Right Table</p>  <p><u>PQ Name:</u> Right Anti</p> </div>	Roxanna Mercier	Fanny Denning	Lesha Nobles	Gertrudis Fitzpatrick	Bernita Crutcher	Shiela Anaya	Yolonda Armstead
Sol Marroquin																																	
Kiera Mcfall																																	
Raven Beatty																																	
Elinore Dees																																	
Wei Lockwood																																	
Donald Eldridge																																	
Claudio Beam																																	
Angelita Packer																																	
Reyna Luke																																	
Beaulah Wenger																																	
Malvina Hamer																																	
Vivan Keeney																																	
Raven Beatty																																	
Roxanna Mercier																																	
Fanny Denning																																	
Lesha Nobles																																	
Wei Lockwood																																	
Gertrudis Fitzpatrick																																	
Angelita Packer																																	
Beaulah Wenger																																	
Malvina Hamer																																	
Bernita Crutcher																																	
Shiela Anaya																																	
Yolonda Armstead																																	
Roxanna Mercier																																	
Fanny Denning																																	
Lesha Nobles																																	
Gertrudis Fitzpatrick																																	
Bernita Crutcher																																	
Shiela Anaya																																	
Yolonda Armstead																																	

Tipi di join – Tipi di merge

L'immagine seguente mostra le tabelle prima della merge e dopo la merge

Table 01 = Left Table = fSales
Product = Foreign Key

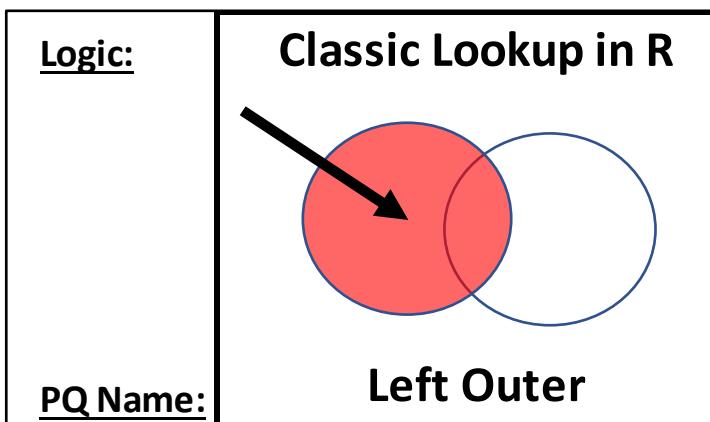
Product	Units
Quad	48
Kangaroo	168
Carlota	132
Carlota	72
Sunshine	108
Quad	156
Carlota	96
Sunshine	60
Sunshine	24
Carlota	120
Quad	24

Table 02 = Right Table = dProductPrice
Product = Primary Key

Product	Price
Carlota	\$26
Quad	\$43
Sunshine	\$19
Majestic Beaut	\$27

Goal: Return Sales Table with new Price Column
Replaces VLOOKUP or Relationships (**Classic Lookup**)

Product	Units	Price
Quad	48	43
Carlota	132	26
Carlota	72	26
Kangaroo	168	
Sunshine	108	19
Quad	156	43
Carlota	96	26
Sunshine	60	19
Sunshine	24	19
Carlota	120	26
Quad	24	43



Tipi di join – Tipi di merge

L'immagine seguente mostra le tabelle prima della merge e dopo la merge

Table 01: Left Table
Product Table = Left Table = dProductRightOuter
with Supplier ID Column (Foreign Key)

Product	SupplierID	Price	Cost
Aspen	CO	23	11
Carlota	GB	26	12.75
Majestic Beaut	GB	29	15.85
Quad	GB	43	22.5
Sunshine	CO	19	1.25
Kangaroo	CC	14	6.95

Table 02: Right Table
Supplier Table = Right Table = dSupplierRightOuter
with Supplier ID Column (Primary Key)

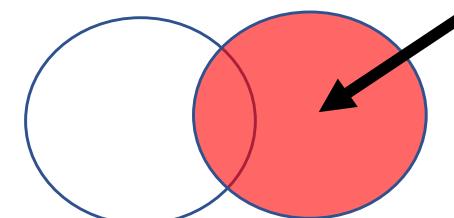
SupplierID	Name	City	State
CO	Colorado Boomerangs	Gunnison	CO
GB	Gel Boomerangs	Oakland	CA
DB	Darnell Booms	Manchester	MA

Goal: is to show all Suppliers from the Right Table
and all of the Related Products, but do NOT show
Products without a Supplier

Product	Price	Cost	SupplierID	Name	City	State
Aspen	23	11	CO	Colorado Boomerangs	Gunnison	CO
Carlota	26	12.75	GB	Gel Boomerangs	Oakland	CA
Majestic Beaut	29	15.85	GB	Gel Boomerangs	Oakland	CA
Quad	43	22.5	GB	Gel Boomerangs	Oakland	CA
Sunshine	19	1.25	CO	Colorado Boomerangs	Gunnison	CO
			DB	Darnell Booms	Manchester	MA

Logic:

Classic Lookup in L



PQ Name:

Right Outer

Fuzzy Merge

Fuzzy Merge ti consente di applicare gli algoritmi Fuzzy Matching quando confronti le colonne e prova a trovare corrispondenze tra le tabelle che vengono unite.

Merge

Select a table and matching columns to create a merged table.

ProductDim

ProductID	Product	Category 1	Segment 2	ManufacturerID	Manufacturer	Unit Cost	Unit Price
392	Maximus RP-01	Rural	Productivity	7	VanArsdel	37.2710625	51.0562
393	Maximus RP-02	Rural	Productivity	7	VanArsdel	37.2710625	51.0562
394	Maximus RS-01	Rural	Select	7	VanArsdel	119.7617925	164.0572
396	Maximus UM-01	Accessory	Accessory	7	VanArsdel	66.2830875	90.7987
397	Maximus UM-02	Accessory	Accessory	7	VanArsdel	109.2224175	149.6197

CatSegDim

CatSegID	Category 1	Segment 2
1	Rural	Productivity
2	Rural	Select
3	Accessory	Accessory
4	Urban	Moderation
5	Urban	Regular

Join Kind

Left Outer (all from first, matching from second)

Use fuzzy matching to perform the merge

► Fuzzy merge options

Similarity threshold (optional)

Ignore case

Match by combining text parts (optional)

Maximum number of matches (optional)

Transformation table (optional)

Tipi di join – Altri modi di combinare le query



Merge

Combina gli attributi (colonne) di due query in base alle colonne unite

- Le colonne utilizzate per il join devono trovarsi in entrambe le query (possono essere una o più)
- Entrambe le query vengono rieseguite ad ogni esecuzione
- In termini di database, questo è un JOIN

Append

"Stack" record di due (o più) query

- Entrambe le query (tutte) si rieseguono con ogni esecuzione di append
- Le intestazioni di colonna devono essere chiamate uguali
- Le colonne non corrispondenti verranno aggiunte a destra come colonne extra

Duplicate

Copia una query in un determinato momento (Salva con nome)

- La query duplicata può essere modificata in modo indipendente senza alterare la query originale
- L'esecuzione della seconda query (duplicata) NON esegue la prima query (originale)

Reference

Utilizza l'output di una query come INPUT in un'altra query. (Link in una catena)

- Tutte le modifiche successive apportate alla prima query influiscono sulla seconda durante l'aggiornamento
- Quando viene eseguita la seconda query, esegue nuovamente tutti i passaggi della prima query

Reference vs Duplicate

Quando si utilizzano tabelle e query in Power Query in Power BI, è possibile copiarle tramite queste azioni:

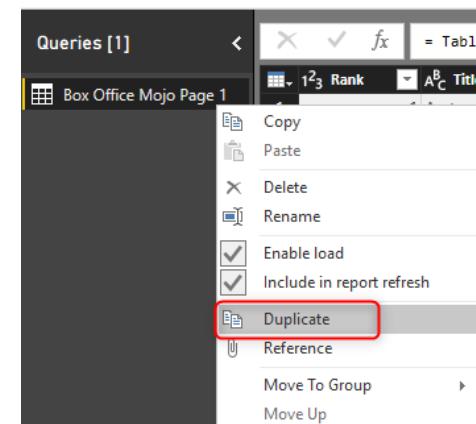
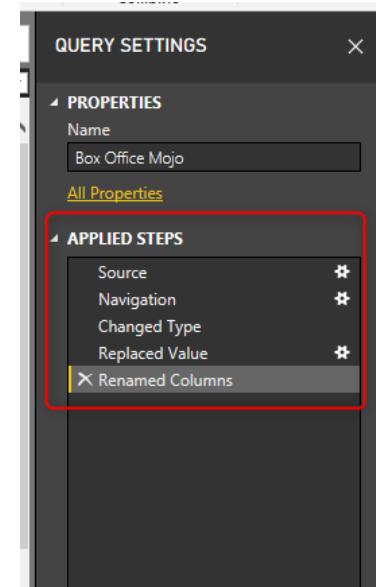
- Duplica (Duplicate)
- Riferimento (Reference)

Mi è sempre stata fatta la domanda su quale sia la vera differenza tra queste due azioni.

La spiegazione è semplice ma molto importante da capire. Perché quando conosci la differenza, la userai correttamente.

Reference vs Duplicate

- Se stai cercando di copiare un'intera query con tutti i suoi passaggi, Duplicate è il tuo amico.
- Ad esempio, supponiamo che abbiamo ottenuto i dati da una pagina web che ci mostra le informazioni sui film del best seller. Il link di riferimento è <https://www.boxofficemojo.com/alltime/world/?pagenum=1&p=.htm>
- Dopo aver effettuato delle trasformazioni ti rendi conto che i dati del link indicato sono solo per i primi cento film perché quella pagina web non contiene i film rimanenti. Per ottenere gli altri, è necessario passare alla pagina 2, che ha un URL diverso (pagenum=2), ma la stessa struttura di dati.
- Bene, cosa devi fare? Dobbiamo rifare tutti i passaggi di trasformazione anche per la pagina 2? Non serve possiamo sfruttare Duplicate.



Reference vs Duplicate

- Il riferimento è un altro modo di copiare una query, tuttavia, la grande differenza è che quando si fa riferimento a una query, la nuova query avrà solo un passaggio: un puntamento alla query originale.
- Una query referenziata non avrà i passaggi applicati della query originale da utilizzare quando **NON vogliamo cambiare nessuna delle query esistenti**, perché vogliamo usarle come fonte per altre operazioni.
- L'unico passaggio è il link alla sorgente dati dalla query originale. Cosa significa? Significa che se si apportano modifiche alla query originale, questa nuova query avrà un impatto su tutte le query collegate.

The screenshot illustrates the creation of a new query from an existing one. On the left, the 'Queries [2]' pane shows two existing queries: 'Box Office Mojo Page 1' and 'Box Office Mojo Page 2'. A context menu is open over 'Box Office Mojo Page 1', with the 'Reference' option highlighted by a red box. On the right, the main workspace shows a new query named 'Box Office Mojo Page 1 (2)' in the 'Queries [3]' pane. This new query has a single step labeled 'Source' in the 'QUERY SETTINGS' pane, indicating it refers back to the original query. Red arrows point from the 'Reference' menu item to both the new query name and the 'Source' step.

Step	Description
1	Original Query: Box Office Mojo Page 1
2	New Query: Box Office Mojo Page 1 (2)
3	Step Type: Source

Isolamento dall'originale o dipendenza dall'originale

Duplica crea una nuova copia con tutti i passaggi esistenti. La nuova copia verrà isolata dalla query originale. È possibile apportare modifiche alla query originale o nuova e NON si influenzano a vicenda. Il riferimento, d'altra parte, è una nuova copia con un solo passaggio: ottenere dati dalla query originale. Se si apporta una modifica alla query originale, la nuova query avrà un impatto. Per esempio; Se si rimuove una colonna dalla query originale, la nuova query non la avrà se ha utilizzato il metodo di riferimento per la copia.

Limitazione del riferimento

Non è possibile utilizzare le query referenziate in tutte le situazioni. Immaginiamo che si abbia una query 1 e che venga creato un riferimento a quella in una query 2. Non è possibile utilizzare il risultato della query 2 nella query 1! **Creerà un riferimento circolare. Stai combinando una query con un riferimento alla query stessa, è impossibile!**

Utilizzo impostazioni internazionali

Le Impostazioni internazionali / Impostazioni locali in Excel e Power BI Desktop Power Query definiscono come

Date, orari e numeri sono interpretati nei file di testo.

- Nel sistema operativo Windows le impostazioni internazionali e internazionali nel Pannello di controllo determinano i numeri, Valuta, orari, date vengono interpretati sul tuo computer
- Power Query utilizza le impostazioni internazionali e locali del sistema operativo Windows per interpretare i dati.
- Il problema sorge se si ottengono file di testo da un paese diverso che utilizza un formato diverso per Date,
- Tempi e numeri

Utilizzo impostazioni internazionali

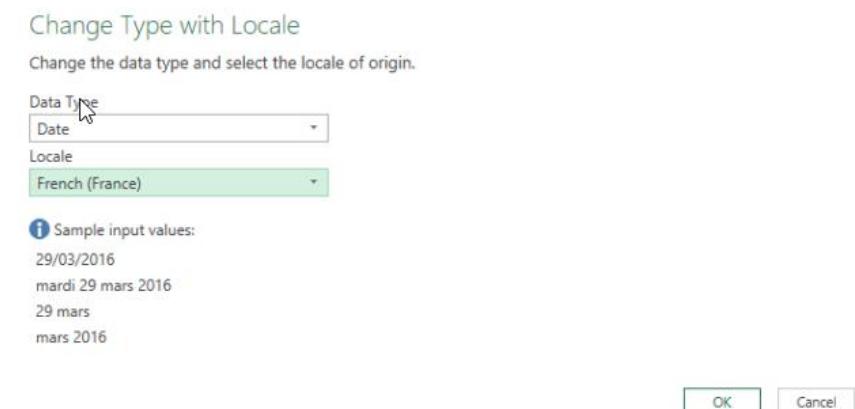
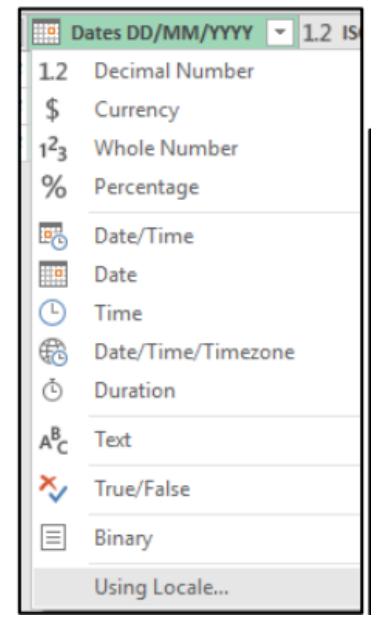
Due modi per risolvere il problema:

1. Impostazioni Reginal (valido per tutti i file di testo importati)
 - Power Query utilizza le impostazioni internazionali e locali del sistema operativo Windows per interpretare i dati
 - È possibile modificare le Impostazioni principali in Power Query Editor in modo che tutti i file di testo importati vengano tutti interpretati per l'impostazione reginale specificata.
 - Opzioni query, quindi nella finestra di dialogo Opzioni query fai clic su Impostazioni principali, come mostrato qui a lato. (versione Excel e Power BI)

The screenshot shows two overlapping dialog boxes from the Power BI Options interface. The top dialog is titled 'Query Options' and has a sidebar with 'GLOBAL' settings like Data Load, Power Query Editor, Security, Privacy, Account, and Diagnostics. Under 'CURRENT WORKBOOK', 'Regional Settings' is highlighted with a green background. The bottom dialog is titled 'Options' and has a sidebar with 'GLOBAL' settings like Data Load, Power Query Editor, DirectQuery, R scripting, Security, Privacy, Updates, Usage Data, Diagnostics, Preview features, and Auto recovery. Under 'CURRENT FILE', 'Regional Settings' is also highlighted with a yellow background. Both dialogs have a 'Locale' dropdown set to 'English (United States)'.

Utilizzo impostazioni internazionali

2. Impostazione "Data Type using locale ..." per singole colonne (stesso processo in Excel e Power BI Desktop)
 - È possibile modificare le impostazioni internazionali per una colonna specifica utilizzando l'icona Tipo di dati nella parte superiore di colonna e quindi facendo clic sull'opzione "Using Locale..." e quindi selezionando le impostazioni internazionali dalla finestra di dialogo, come mostrato qui a lato

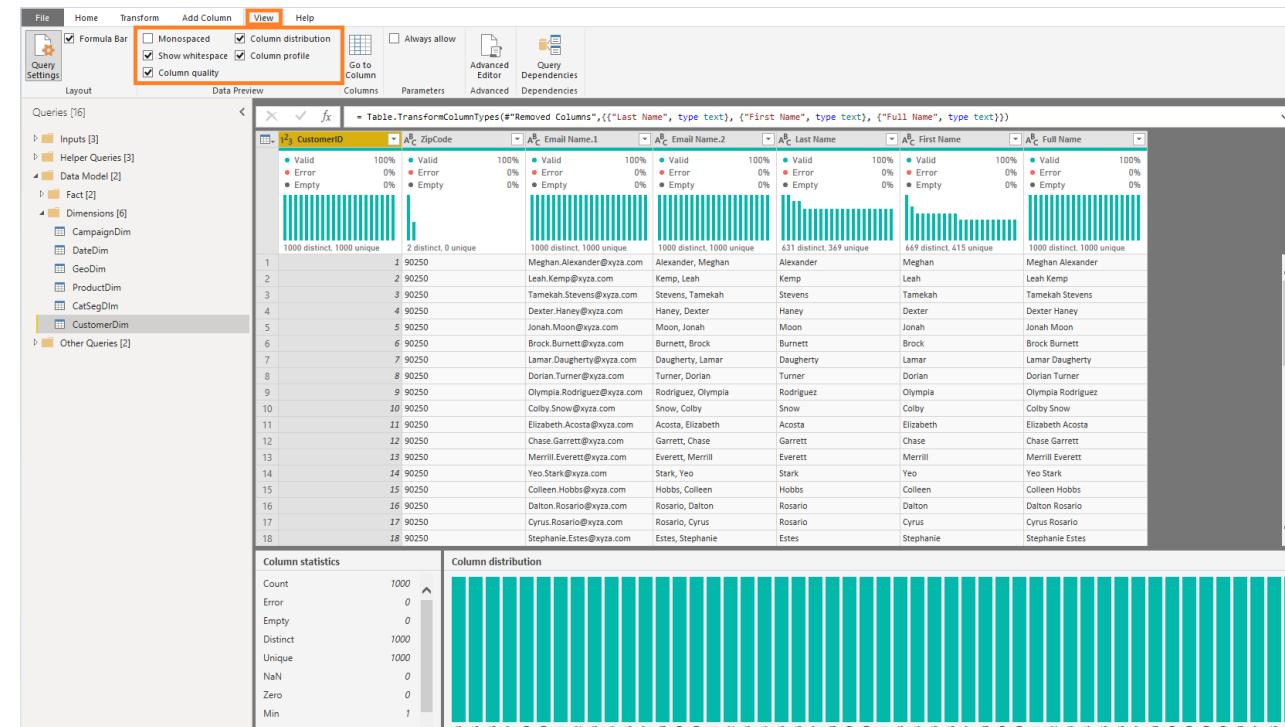


Data Profiling



La profilazione dei dati consente di comprendere facilmente e rapidamente la distribuzione dei dati

- **Statistiche sulle colonne**— # di errori, valori vuoti, validi, duplicati e univoci. Misure di distribuzione del valore come Min / Max / Media / Mediana, ecc.
- **Distribuzione delle colonne**—istogrammi di distribuzione del valore in linea, inclusa anche la possibilità di Mantenere o Rimuovere valori di riga , che genererà il passaggio

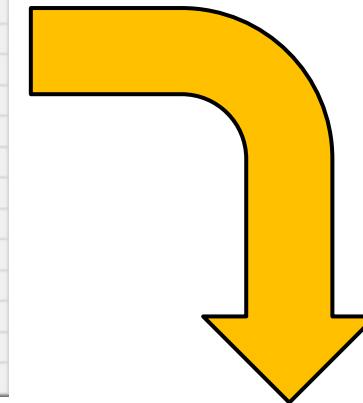


Modulo 3 Lab 1: Trasformazioni complesse

Modulo 3 Lab: Creare la tabella dei fatti Budget

Obiettivo: trasformare un file di budget "ampio" con tre righe extra in alto e un'intestazione di tre righe , in una tabella BudgetFact utilizzabile

	A ^B _C Column1	A ^B _C Column2	A ^B _C Column3	A ^B _C Column4	A ^B _C Column5	A ^B _C Column6	A ^B _C Column7	A ^B _C Column8	A ^B _C Column9
1	Budget Spreadsheet for VanArsdel								
2									
3									
4		Forecast							
5		2016	2016	2016	2016	2016	2016	2016	
6	Category	Segment	Dec	Nov	Oct	Sep	Aug	Jul	Jun
7	Accessory	Accessory	44190.57888	50598.81566	54740.5709	64442.9079	98285.91328	61545.41064	91299.65402
8	Mix	All Season	11442.14474	14120.78693	18109.64804	26737.36704	37439.93736	45563.53694	29301.60844
9	Mix	Productivity	19538.89812	17597.55926	22835.18396	25557.336	34963.40484	17727.72593	73468.33494
10	Rural	Select	311.708775	172.2601125	662.79129	314.98992	311.708775	334.67679	167.338395
11	Urban	Convenience	120710.4406	129923.2814	169468.7696	219703.7363	333460.8569	345230.9632	262211.9656
12	Urban	Extreme	20868.84072	46971.33037	70793.02886	11957.87523	10668.924	14286.02201	5340.80274
13	Urban	Moderation	251155.7122	322984.2215	362385.6466	529188.1249	675144.698	642300.7198	651766.3129
14	Urban	Regular	689.7969225	427.4372025	2989.28376	701.48442	731.3103	1381.096238	151.17039
15	Youth	Youth	3931.03074	2891.005425	5397.748965	6437.887575	10355.73462	11738.08624	12144.36878



	1.2 CategoryID	A ^B _C Scenario	Date	\$ Budget Amount
1	3	Forecast	12/1/2016	44190.5789
2	3	Forecast	11/1/2016	50598.8157
3	3	Forecast	10/1/2016	54740.5709
4	3	Forecast	9/1/2016	64442.9079
5	3	Forecast	8/1/2016	98285.9133

Modulo 3 Lab: Creare la tabella dei fatti Budget

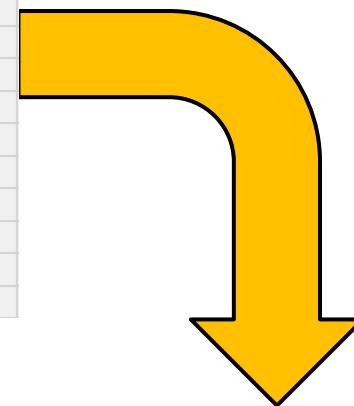
Importa CSV, rimuovi righe vuote e rinomina la query in BudgetFact_Data

	ABC Column1	ABC Column2	ABC Column3	ABC Column4	ABC Column5	ABC Column6	ABC Column7	ABC Column8	ABC Column9
1			Forecast						
2			2016	2016	2016	2016	2016	2016	2016
3	Category	Segment	Dec	Nov	Oct	Sep	Aug	Jul	Jun
4	Accessory	Accessory	44190.57888	50598.81566	54740.5709	64442.9079	98285.91328	61545.41064	91299.65402
5	Mix	All Season	11442.14474	14120.78693	18109.64804	26737.36704	37439.93736	45563.53694	29301.60844
6	Mix	Productivity	19538.89812	17597.55926	22835.18396	25557.336	34963.40484	17727.72593	73468.33494
7	Rural	Select	311.708775	172.2601125	662.79129	314.98992	311.708775	334.67679	167.338395
8	Urban	Convenience	120710.4406	129923.2814	169468.7696	219703.7363	333460.8569	345230.9632	262211.9656
9	Urban	Extreme	20868.84072	46971.33037	70793.02886	11957.87523	10668.924	14286.02201	5340.80274
10	Urban	Moderation	251155.7122	322984.2215	362385.6466	529188.1249	675144.698	642300.7198	651766.3129
11	Urban	Regular	689.7969225	427.4372025	2989.28376	701.48442	731.3103	1381.096238	151.17039
12	Youth	Youth	3931.03074	2891.005425	5397.748965	6437.887575	10355.73462	11738.08624	12144.36878

Modulo 3 Lab: Creare la tabella dei fatti Budget

Crea un duplicato ed estrai solo le righe di intestazione in una query separata

	ABC Column1	ABC Column2	ABC Column3	ABC Column4	ABC Column5	ABC Column6	ABC Column7	ABC Column8	ABC Column9
1		Forecast	Forecast	Forecast	Forecast	Forecast	Forecast	Forecast	Forecast
2		2016	2016	2016	2016	2016	2016	2016	2016
3	Category	Segment	Dec	Nov	Oct	Sep	Aug	Jul	Jun
4	Accessory	Accessory	44190.57888	50598.81566	54740.5709	64442.9079	98285.91328	61545.41064	91299.65402
5	Mix	All Season	11442.14474	14120.78693	18109.64804	26737.36704	37439.93736	45563.53694	29301.60844
6	Mix	Productivity	19538.89812	17597.55926	22835.18396	25557.336	34963.40484	17727.72593	73468.33494
7	Rural	Select	311.708775	172.2601125	662.79129	314.98992	311.708775	334.67679	167.338395
8	Urban	Convenience	120710.4406	129923.2814	169468.7696	219703.7363	333460.8569	345230.9632	262211.9656
9	Urban	Extreme	20868.84072	46971.33037	70793.02886	11957.87523	10668.924	14286.02201	5340.80274
10	Urban	Moderation	251155.7122	322984.2215	362385.6466	529188.1249	675144.698	642300.7198	651766.3129
11	Urban	Regular	689.7969225	427.4372025	2989.28376	701.48442	731.3103	1381.096238	151.17039
12	Youth	Youth	3931.03074	2891.005425	5397.748965	6437.887575	10355.73462	11738.08624	12144.36878



	ABC Column1	ABC Column2	ABC Column3	ABC Column4	ABC Column5	ABC Column6	ABC Column7	ABC Column8
1			Forecast	Forecast	Forecast	Forecast	Forecast	Forecast
2			2016	2016	2016	2016	2016	2016
3	Category	Segment	Dec	Nov	Oct	Sep	Aug	Jul

Modulo 3 Lab: Creare la tabella dei fatti Budget

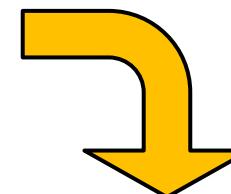
Trasponi le intestazioni, combina in un'unica colonna e trasponi indietro

	ABC Column1	ABC Column2	ABC Column3	ABC Column4	ABC Column5	ABC Column6	ABC Column7	ABC Column8
1				Forecast	Forecast	Forecast	Forecast	Forecast
2				2016	2016	2016	2016	2016
3	Category	Segment		Dec	Nov	Oct	Sep	Aug

	ABC 123 Column1	ABC 123 Column2	ABC 123 Column3
1			Category
2			Segment
3	Forecast	2016	Dec
4	Forecast	2016	Nov
5	Forecast	2016	Oct
6	Forecast	2016	Sep
7	Forecast	2016	Aug
8	Forecast	2016	Jul
9	Forecast	2016	Jun
10	Forecast	2016	May
11	Forecast	2016	Apr
12	Forecast	2016	Mar
13	Forecast	2016	Feb
14	Forecast	2016	Jan
15	Budget	2016	Dec
16	Budget	2016	Nov



	ABC 123 FullyCombinedHeader
1	Category
2	Segment
3	Forecast~12/01/2016
4	Forecast~11/01/2016
5	Forecast~10/01/2016
6	Forecast~09/01/2016
7	Forecast~08/01/2016
8	Forecast~07/01/2016
9	Forecast~06/01/2016
10	Forecast~05/01/2016
11	Forecast~04/01/2016
12	Forecast~03/01/2016
13	Forecast~02/01/2016
14	Forecast~01/01/2016
15	Budget~12/01/2016
16	Budget~11/01/2016



	ABC 123 Column1	ABC 123 Column2	ABC 123 Column3	ABC 123 Column4	ABC 123 Column5
1	Category	Segment	Forecast~12/01/2016	Forecast~11/01/2016	Forecast~10/01/2016

Modulo 3 Lab: Creare la tabella dei fatti Budget

Aggiungi **BudgetFact_Data** alla fine delle intestazioni, quindi rimuovi le righe di intestazione estranee, infine imposta la prima riga come intestazione

	ABC 123 Column1	ABC 123 Column2	ABC 123 Column3	ABC 123 Column4	ABC 123 Column5	ABC 123 Column6	ABC 123 Column7
1	Category	Segment	Forecast~12/01/2016	Forecast~11/01/2016	Forecast~10/01/2016	Forecast~09/01/2016	Forecast~08/01/2016



	ABC 123 Column1	ABC 123 Column2	ABC 123 Column3	ABC 123 Column4	ABC 123 Column5	ABC 123 Column6	ABC 123 Column7
1			Forecast	Forecast	Forecast	Forecast	Forecast
2			2016	2016	2016	2016	2016
3	Category	Segment	Dec	Nov	Oct	Sep	Aug
4	Accessory	Accessory	44190.57888	50598.81566	54740.5709	64442.9079	98285.91328
5	Mix	All Season	11442.14474	14120.78693	18109.64804	26737.36704	37439.93736
6	Mix	Productivity	19538.89812	17597.55926	22835.18396	25557.336	34963.40484

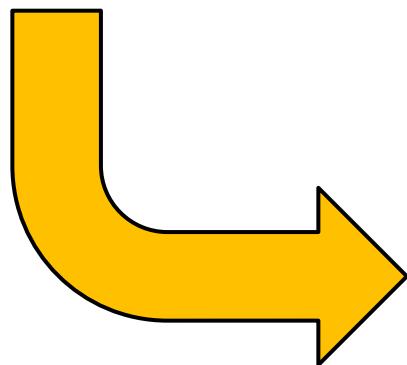


	ABC 123 Category	ABC 123 Segment	ABC 123 Forecast~12/01/2016	ABC 123 Forecast~11/01/2016	ABC 123 Forecast~10/01/2016	ABC 123 Forecast~09/01/2016	ABC 123 Forecast~08/01/2016
1	Accessory	Accessory	44190.57888	50598.81566	54740.5709	64442.9079	98285.91328
2	Mix	All Season	11442.14474	14120.78693	18109.64804	26737.36704	37439.93736
3	Mix	Productivity	19538.89812	17597.55926	22835.18396	25557.336	34963.40484
4	Rural	Select	311.708775	172.2601125	662.79129	314.98992	311.708775
5	Urban	Convenience	120710.4406	129923.2814	169468.7696	219703.7363	333460.8569

Modulo 3 Lab: Creare la tabella dei fatti Budget

Unpivot delle colonne di Forecast, dividerle e rinominarle, quindi impostare i tipi di dati

	ABC 123 Category	ABC 123 Segment	ABC 123 Forecast^12/01/2016	ABC 123 Forecast^11/01/2016	ABC 123 Forecast^10/01/2016	ABC 123 Forecast^09/01/2016	ABC 123 Forecast^08/01/2016
1	Accessory	Accessory	44190.57888	50598.81566	54740.5709	64442.9079	98285.91328
2	Mix	All Season	11442.14474	14120.78693	18109.64804	26737.36704	37439.93736
3	Mix	Productivity	19538.89812	17597.55926	22835.18396	25557.336	34963.40484
4	Rural	Select	311.708775	172.2601125	662.79129	314.98992	311.708775
5	Urban	Convenience	120710.4406	129923.2814	169468.7696	219703.7363	333460.8569

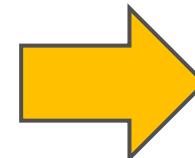


	ABC Category	ABC Segment	ABC Scenario	Date	\$ Value
1	Accessory	Accessory	Forecast	12/1/2016	44190.5789
2	Accessory	Accessory	Forecast	11/1/2016	50598.8157
3	Accessory	Accessory	Forecast	10/1/2016	54740.5709
4	Accessory	Accessory	Forecast	9/1/2016	64442.9079
5	Accessory	Accessory	Forecast	8/1/2016	98285.9133
6	Accessory	Accessory	Forecast	7/1/2016	61545.4106
7	Accessory	Accessory	Forecast	6/1/2016	91299.654
8	Accessory	Accessory	Forecast	5/1/2016	120579.7261
9	Accessory	Accessory	Forecast	4/1/2016	103436.9095

Modulo 3 Lab: Creare la tabella dei fatti Budget

Unisci in CatSegID e rimuovi colonne estranee

	Category	Segment	Scenario	Date	\$	Value
1	Accessory	Accessory	Forecast	12/1/2016	44190.5789	
2	Accessory	Accessory	Forecast	11/1/2016	50598.8157	
3	Accessory	Accessory	Forecast	10/1/2016	54740.5709	
4	Accessory	Accessory	Forecast	9/1/2016	64442.9079	
5	Accessory	Accessory	Forecast	8/1/2016	98285.9133	
6	Accessory	Accessory	Forecast	7/1/2016	61545.4106	
7	Accessory	Accessory	Forecast	6/1/2016	91299.654	
8	Accessory	Accessory	Forecast	5/1/2016	120579.7261	
9	Accessory	Accessory	Forecast	4/1/2016	103436.9095	

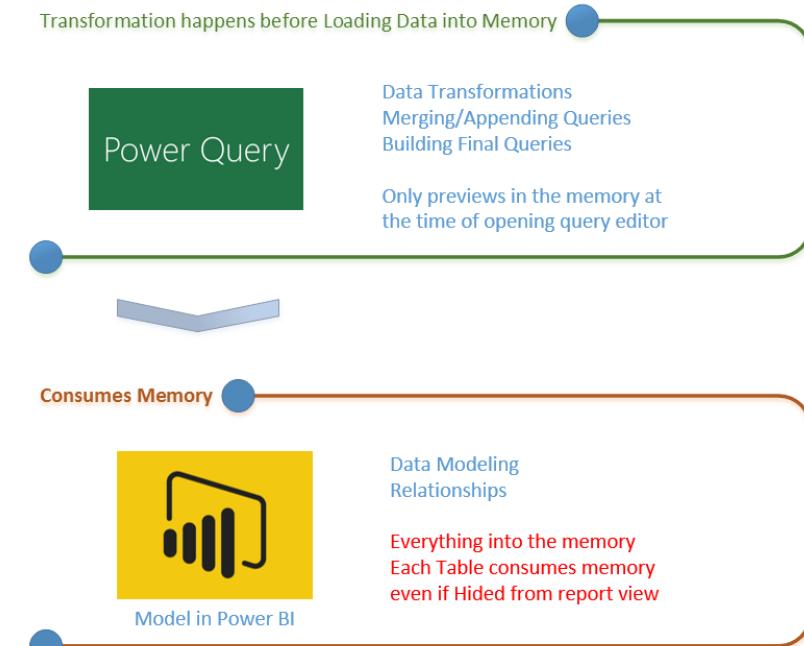


	1	2	3	CatSegID	4	5	6	7	8	9	Scenario	Date	\$	Value
1				3	Forecast						12/1/2016		44190.5789	
2				3	Forecast						11/1/2016		50598.8157	
3				3	Forecast						10/1/2016		54740.5709	
4				3	Forecast						9/1/2016		64442.9079	
5				3	Forecast						8/1/2016		98285.9133	
6				3	Forecast						7/1/2016		61545.4106	
7				3	Forecast						6/1/2016		91299.654	
8				3	Forecast						5/1/2016		120579.7261	
9				3	Forecast						4/1/2016		103436.9095	



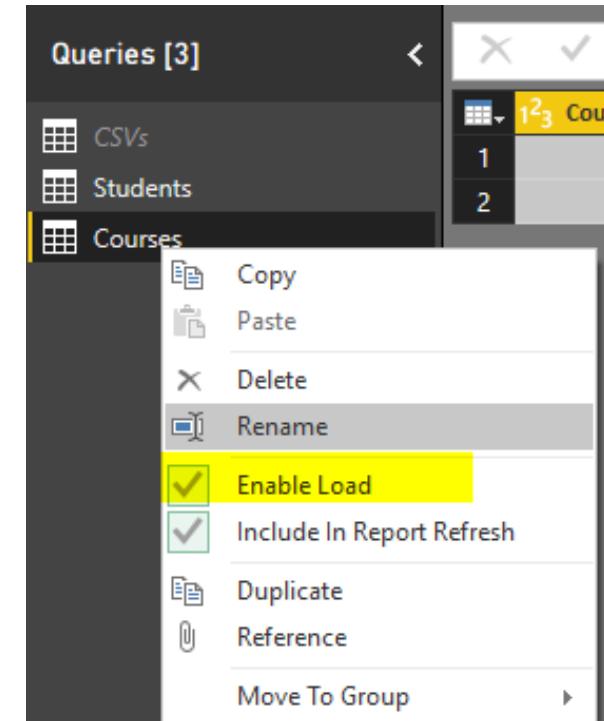
Load

- Per impostazione predefinita, quando si sceglie di chiudere e applicare le trasformazioni dall'editor di query, tutte le query verranno caricate nel modello, indipendentemente dal fatto che si desideri utilizzarle nel modello finale o meno.
- È possibile disabilitare il caricamento di alcune query, in particolare le query utilizzate come trasformazione intermedia per produrre la query finale per il modello. Questo è un'operazione da fare quando il tuo modello diventa grande.



Load

- L'approccio migliore è disabilitare il caricamento prima di chiudere l'editor delle query cliccando con il tasto destro sulla query e scegliendo «Enable Load» come si vede nell'immagine laterale
- Disabilitare il caricamento non significa che la query non verrà aggiornata, significa solo che la query non verrà caricata in memoria.
- Quando si fa clic su Aggiorna modello in Power BI o quando si verifica un aggiornamento pianificato, anche le query contrassegnate come Disabilitate verranno aggiornate, ma i loro dati verranno utilizzati come origine intermedia per altre query invece di caricarsi direttamente nel modello.



1. Quale tipo di trasformazione trasforma tutte le righe in colonne e le colonne in righe?
2. Quale tipo di unione useresti per ottenere tutte le righe da due query, anche se le colonne unite non corrispondessero?
3. Cosa succede in un'append se i nomi delle colonne non corrispondono?

Modulo 4 Lab 1: Variable, Parameters e Functions

Obiettivi

- Comprendere cosa sono le variabili e i parametri
- Comprendere le funzioni personalizzate
- Comprendere i vantaggi della parametrizzazione delle tue query
- Comprendere che cos'è "Inserisci dati" e come possono essere utilizzati
- Scopri come organizzare le query utilizzando i gruppi di cartelle

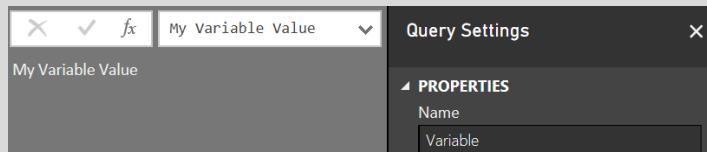
Agenda

- Variables & Parameters in M in Power BI
- Inserimento Dati
- Sfruttare le Variables e Parameters
- Creare il registro delle modifiche

Variabili in M

Le query con valore costante ,tabelle o liste di risultati

- Si creano partendo dalle blank query
- Il nome della query è il nome della variabile da usare nelle altre query
- Il valore della variabile si vede nella formula bar come sotto



- Nell'Advanced Editor potrebbe avere il seguente formato



Parametri in M

I parametri sono il livello successivo delle variabili.

- I parametri possono avere qualsiasi tipo di dato
- Il più comune sono valori singola



- L'elenco dei valori per un parametro può anche provenire da una query formattata come **Lista**



- I parametri sono usati come input di sorgente

```
= Excel.Workbook(File.Contents( Path & ActualsFile ), null, true)
```

Utilizzo dei Parametri

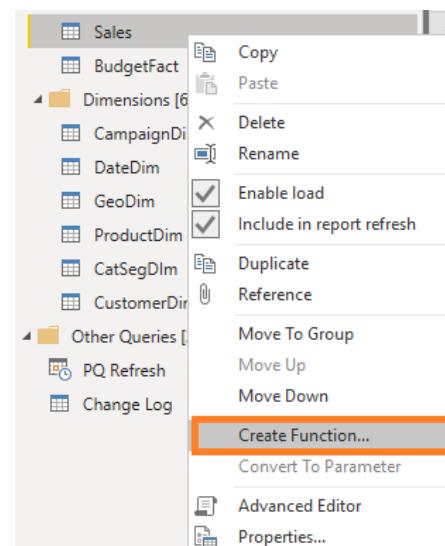
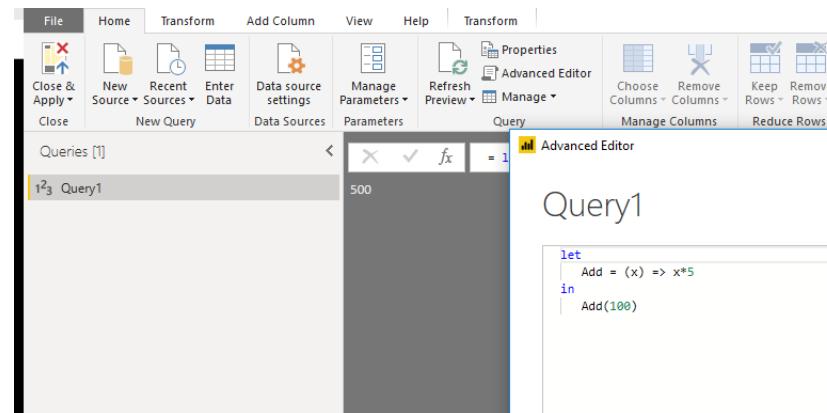
Essere creativi!

- SQL Server
 - Per il nome dei database
 - Per il server (Switch Dev, Test, Prod)
- "Per le clausole where" in SQL Queries
 - Per estrarre tutte le org oppure una
- Nome del sito del tema di SharePoint
 - Se si basano su un template è possibile condividere facilmente le query
- File Paths e File Names
 - Avere un unico posto in cui è necessario apportare una modifica

Funzioni Custom

Le funzioni personalizzate ti aiutano a riutilizzare il codice

1. Creare una funzione usando la query vuota
2. Creare una funzione usando la query di base



Richiamare una funzione su colonna (Demo 2)

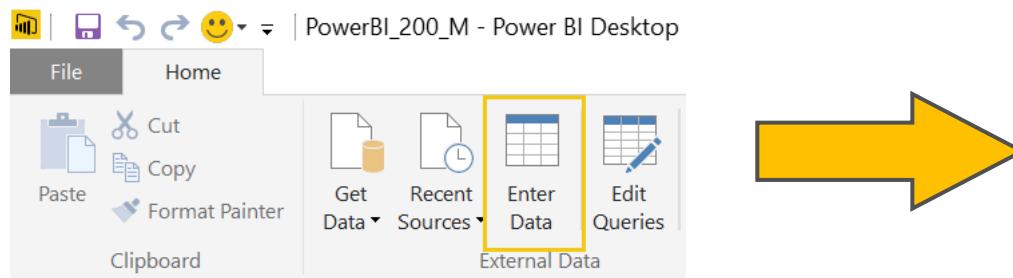


Nell'aggiornamento di settembre 2016 per Power BI nasce un nuovo pulsante che permette di richiamare una funzione personalizzata nell'editor di query in Power BI. Questa è una funzionalità davvero utile.

Con questo tasto Power BI ha introdotto la possibilità di generare automaticamente funzioni da query che utilizzano parametri, senza la necessità di scrivere codice, e possiamo anche richiamare facilmente le funzioni facendo clic su un pulsante. Ciò significa che si possono fare modelli di caricamento dei dati molto più avanzati per gli utenti che non conoscono M e c'è ancora meno bisogno che qualcuno apra la finestra Advanced Editor e inizi a scrivere codice.

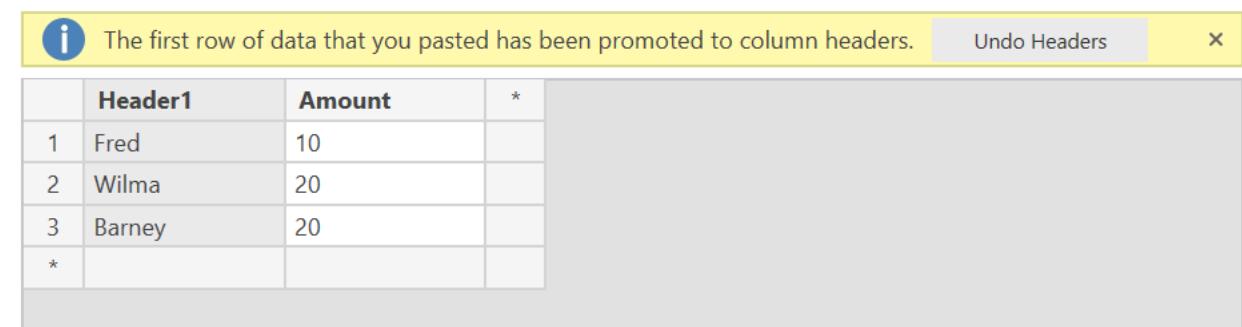
Enter Data

L'utente immette i dati per aggiungere una tabella al volo. Servono per tabelle di mappatura

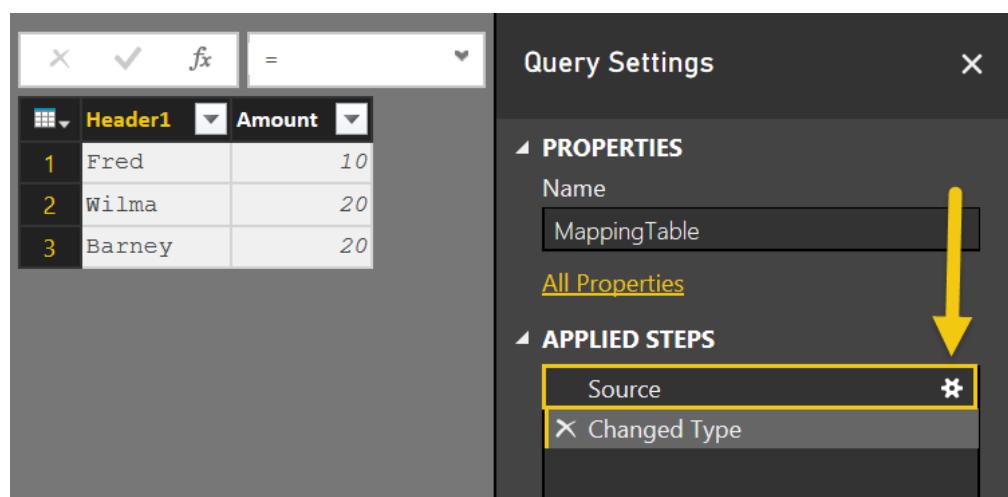


Create Table

Create a table by typing or pasting content.



	Header1	Amount	*
1	Fred	10	
2	Wilma	20	
3	Barney	20	
*			



	Header1	Amount
1	Fred	10
2	Wilma	20
3	Barney	20

Query Settings

PROPERTIES

Name: MappingTable

All Properties

APPLIED STEPS

Source *

Changed Type

<- Per aggiungere ulteriori righe o colonne in un secondo momento, fai clic sull'icona a forma di ingranaggio.

Module 4 Lab 1:

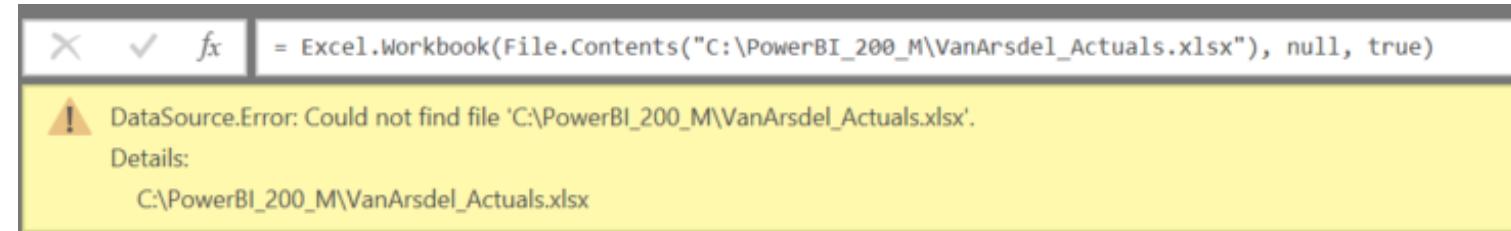
Creazione di Parametri e Percorsi

Modulo 4 Lab: Considerazioni su File source



Tutte le query nel nostro file dipendono dal fatto che il file si trova in un percorso specificato

- Se si sposta il file di origine e si tenta di aggiornare le query, si riceverà un errore
 - C:\ to File Share
 - Different folder location
 - To SharePoint
 - To OneDrive
- Ogni query dovrà essere cambiata OGNI volta che il file viene spostato

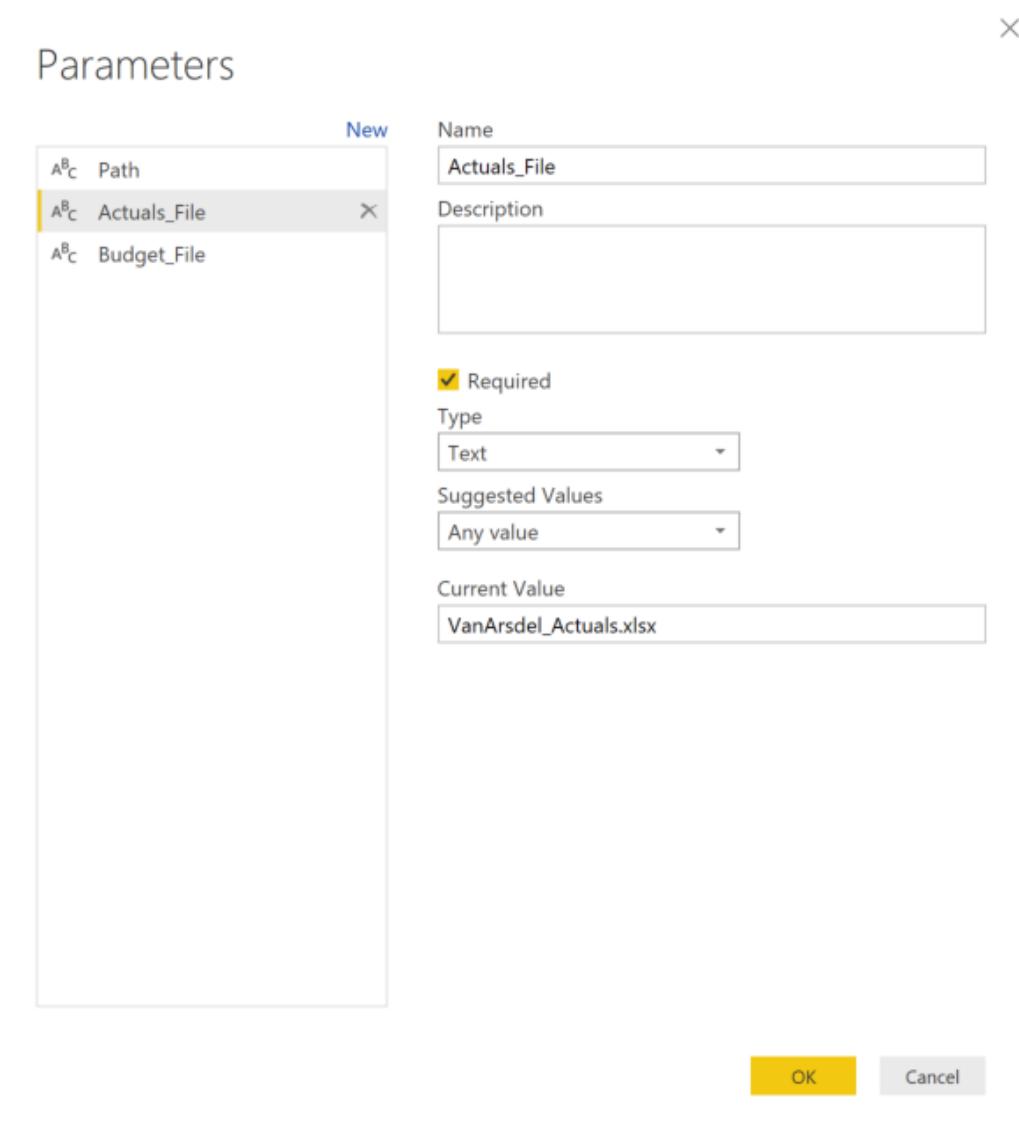


La creazione di parametri con percorsi dinamici risolve questo problema

Modulo 4 Lab: Creazione di Parametri

Crea parametri

- Actual File
- Budget File
- Path



Modulo 4 Lab: Path dinamico per Excel Source File

Utilizzare una query vuota per creare il percorso dinamico

- Usare una blank query
- Popola un advanced Editor di query con il testo che abbiamo fornito
- Per ogni query che usa l'excel source aggiorniamo la sorgente (applied step) con il nome della nuova variabile

=**Actuals_Path**

Advanced Editor

Actuals_Path

```
let  
    FilePath = Path, /*External reference to text query = FilePath  
    FileName = Actuals_File, /* Wrapping */  
  
    PathSlash = if Text.StartsWith(FilePath,"http") then "/" else "\",  
    FullPath = FilePath & (if Text.EndsWith(FilePath, PathSlash) then "" else PathSlash) & FileName,  
  
    Source = if Text.StartsWith(FilePath,"http")  
        then Excel.Workbook(Web.ContentsFullPath), null, true)  
        else Excel.Workbook(File.ContentsFullPath), null, true)  
in  
    Source
```

✓ No syntax errors have been detected.

Done Cancel

Query Settings

PROPERTIES

Name: DateDim

All Properties

APPLIED STEPS

Source

Navigation

Changed Type

	Name	Data	Item	Kind	Hidden
1	Date	Table	Date	Sheet	FALSE
2	Campaign	Table	Campaign	Sheet	FALSE
3	Customer	Table	Customer	Sheet	FALSE
4	Product	Table	Product	Sheet	FALSE
5	Geo	Table	Geo	Sheet	FALSE
6	Sales	Table	Sales	Sheet	FALSE
7	DateDim	Table	DateDim	Table	FALSE
8	CampaignDim	Table	CampaignDim	Table	FALSE
9	CustomerDim	Table	CustomerDim	Table	FALSE
10	ProductDim	Table	ProductDim	Table	FALSE

Modulo 4 Lab: Path dinamico per Excel Source File



Utilizzare la query Actuals_Path come variabile in altre query

- In Advanced Editor, aggiorna l'origine Excel alla nuova origine Actuals_Path.

CampaignDim

Original

```
let
    Source = Excel.Workbook(File.Contents("C:\PowerBI_200_M\PowerBI_200_M_Data.xlsx"), null, true),
    CampaignDim_Table = Source{[Item="CampaignDim",Kind="Table"]}[Data],
    #"Changed Type" = Table.TransformColumnTypes(CampaignDim_Table,{{"CampaignId", type text}, {"Traffic Chann
in
    #"Changed Type"
```

CampaignDim

Updated

```
let
    Source = Actuals Path,
    Campaign_Sheet = Source{[Item="CampaignDim",Kind="Table"]}[Data],
    #"Changed Type" = Table.TransformColumnTypes(Campaign_Sheet,{{"CampaignID", Int64.Type}, {"Traffic Chann
in
    #"Changed Type"
```

Modulo 4 Lab: Path dinamico per CSV Source File

Segui uno schema simile per rendere dinamico il file CSV Budget

- Usare una blank query
- Popola un advanced Editor di query con il testo che abbiamo fornito
- Per ogni query che usa l'excel source aggiorniamo la sorgente (applied step) con il nome della nuova variabile
= **Budget_Path**

Advanced Editor

Budget_Path

```
let
    FilePath = Path, //External reference to text query = FilePath
    BudgetFilename= Budget_File, /* Wrapping comment line */

    PathSlash = if Text.StartsWith(FilePath,"http") then "/" else "\","
    FullPath = FilePath & (if Text.EndsWith(FilePath, PathSlash) then "" else PathSlash) & BudgetFilename,
    Source = if Text.StartsWith(FilePath,"http")
        then Csv.Document(Web.ContentsFullPath ),[Delimiter=",", Encoding=1252, QuoteStyle=QuoteStyle.None])
        else Csv.Document(File.ContentsFullPath ),[Delimiter=",", Encoding=1252, QuoteStyle=QuoteStyle.None])
in
    Source
```

No syntax errors have been detected.

Done Cancel

Query Settings

Column1	Column2	Column3	Column4	Column5
Budget Spread...				
	Forecast	Forecast	Forecast	
	2016	2016	2016	
Category	Segment	Dec	Nov	Oct
Accessory	Accessory	44190.57888	50598.81566	54740.57
Mix	All Season	11442.14474	14120.78693	18109.64
Mix	Productivity	19538.89812	17597.55926	22835.18
Rural	Select	311.708775	172.2601125	662.7912

PROPERTIES

Name: BudgetFact_Data

[All Properties](#)

APPLIED STEPS

Source

Removed Top Rows

Added Index

Modulo 4 Lab: Funzioni custom

Creare una funzione per ottenere il numero di giorni dall'inizio dell'anno

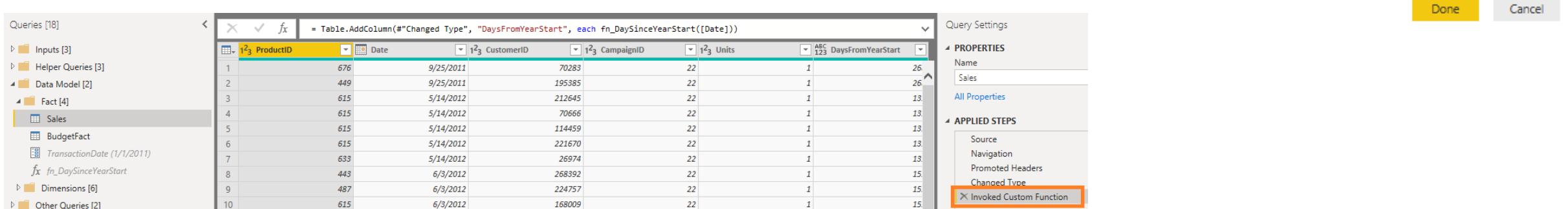
- Usare una blank query
- Popola un advanced Editor di query con il testo che abbiamo fornito
- Invocare una funzione custom per la tabella delle Sales. Ciò fornirà il numero di giorni dall'inizio dell'anno per ogni transazione

Advanced Editor

```
fn_DaySinceYearStart
```

let
Source = (TransactionDate as date) => let
 YearStart = #date(Date.Year(TransactionDate),1,1),
 #"DateDiff" = Duration.From(TransactionDate-YearStart),
 #"NumberDays" = Duration.Days(#"DateDiff") + 1
in
 #"NumberDays"
in
Source

✓ No syntax errors have been detected.



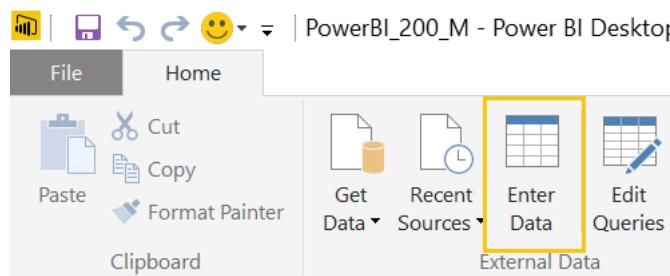
The screenshot shows the Power BI Query Editor interface. On the left, the 'Queries [18]' pane lists various data sources and queries, with 'Sales' selected. In the center, a table view shows columns: ProductID, Date, CustomerID, CampaignID, Units, and DaysFromYearStart. The formula bar at the top indicates a step: `= Table.AddColumn(#"Changed Type", "DaysFromYearStart", each fn_DaySinceYearStart([Date]))`. On the right, the 'Query Settings' pane displays 'Properties' (Name: Sales, All Properties) and 'Applied Steps' (Source, Navigation, Promoted Headers, Changed Type, **Invoked Custom Function**). The 'Done' and 'Cancel' buttons are at the bottom right.

Module 4 Lab: Create un registro delle modifiche



Utilizzare Immetti dati per creare un registro modifica

- Creare una tabella all'interno della query per ospitare il registro delle modifiche
- Denominare la query Registro modifica
- Aggiorna la tabella ogni volta che viene apportata una modifica al codice
- Se decidi di caricare il file nel modello di dati, hai la possibilità di generare report sulla versione corrente e sulla data dell'ultimo aggiornamento



Create Table

Create a table by typing or pasting content.

	Version	Date	Made By	Change	*
1	2.0	12/15/2016	V-Barran (Barbara)	Version 2 Updates	
2	2.01	1/3/2017	V-Barran (Barbara)	Lab Updates	
*					

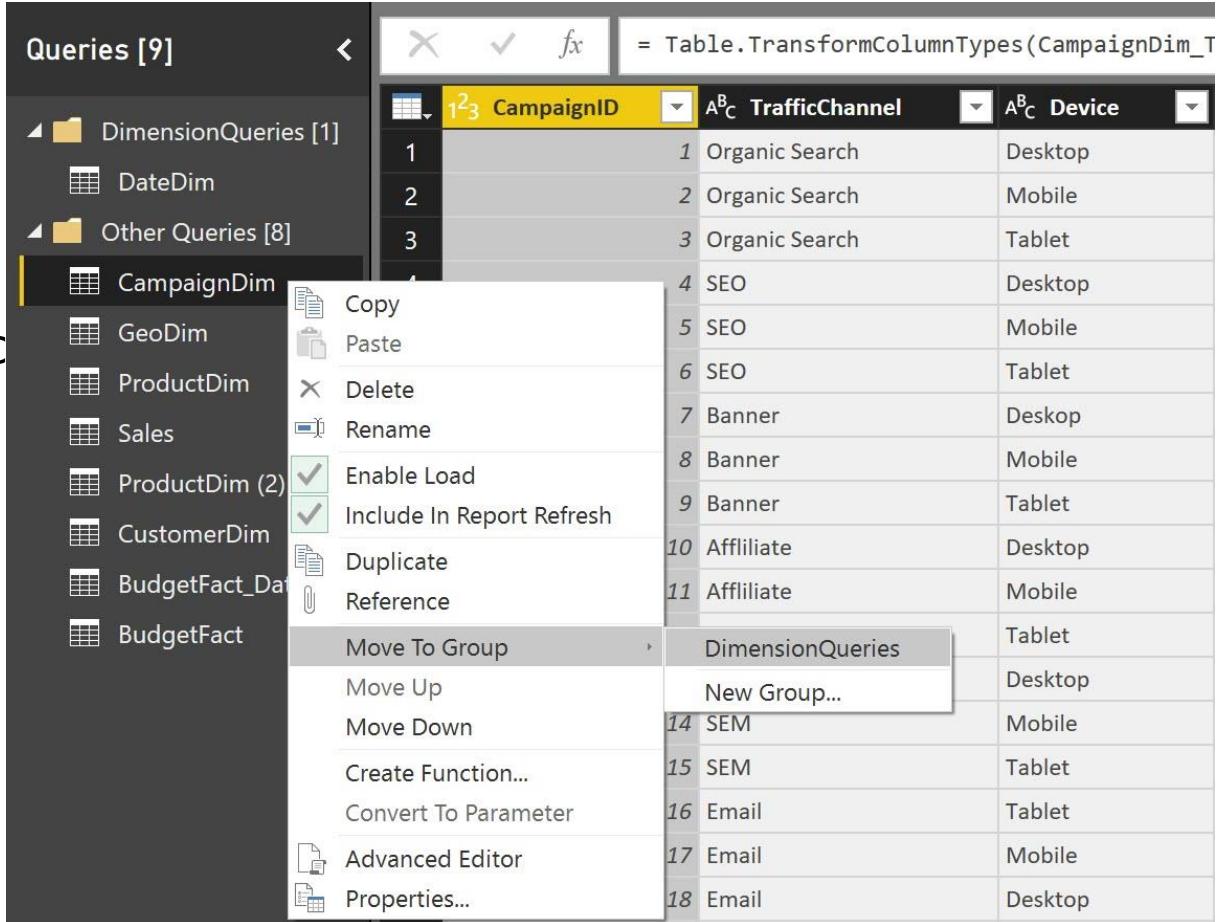
Report

Version	Change	Date
2.00	Version 2 Updates	12/15/16
2.01	Lab Updates	1/3/17

Module 4 Lab: Organizza le tue query

Utilizzare le cartelle per raggruppare le query

- Crea gruppi logici per gestire le tue query
 - Può creare sottogruppi
- Usa Drag and Drop per spostare le query nelle cartelle



The screenshot shows the Power BI Query Editor interface. On the left, the 'Queries [9]' pane lists various queries grouped into categories: 'DimensionQueries [1]' (containing 'DateDim') and 'Other Queries [8]' (containing 'CampaignDim', 'GeoDim', 'ProductDim', 'Sales', 'ProductDim (2)', 'CustomerDim', 'BudgetFact_Dat', and 'BudgetFact'). A context menu is open over the 'CampaignDim' query, showing options like 'Copy', 'Paste', 'Delete', 'Rename', 'Enable Load' (checked), 'Include In Report Refresh' (checked), 'Duplicate', 'Reference', 'Move To Group' (highlighted), 'Move Up', 'Move Down', 'Create Function...', 'Convert To Parameter', 'Advanced Editor', and 'Properties...'. The 'Move To Group' option has a submenu with 'DimensionQueries' selected. On the right, a preview grid displays data from the 'CampaignDim' query with columns: CampaignID, TrafficChannel, and Device. The data includes rows for Organic Search, SEO, Banner, SEM, Email, and Desktop, Mobile, Tablet devices.

CampaignID	TrafficChannel	Device
1	Organic Search	Desktop
2	Organic Search	Mobile
3	Organic Search	Tablet
4	SEO	Desktop
5	SEO	Mobile
6	SEO	Tablet
7	Banner	Desktop
8	Banner	Mobile
9	Banner	Tablet
10	Affiliate	Desktop
11	Affiliate	Mobile
14	SEM	Tablet
15	SEM	Desktop
16	Email	Mobile
17	Email	Tablet
18	Email	Desktop

Connettori custom

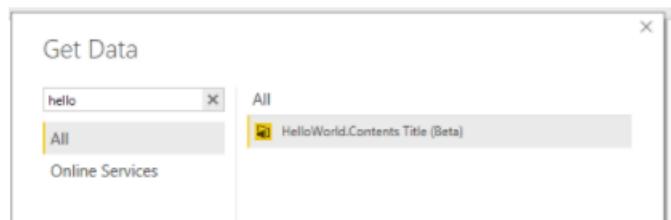
Usa il linguaggio M per creare il tuo connettore personalizzato!

- Samples and SDK on github

<https://github.com/Microsoft/DataConnectors>

Getting Started with Data Connectors

Data Connectors for Power BI enables users to connect to and access data from your application, service, or data source, providing them with rich business intelligence and robust analytics over multiple data sources. By integrating seamlessly into the Power Query connectivity experience in Power BI Desktop, Data Connectors make it easy for power users to query, shape and mashup data from your app to build reports and dashboards that meet the needs of their organization.



Data Connectors are created using the [M language](#). This is the same language used by the Power Query user experience found in Power BI Desktop and Excel 2016. Extensions allow you to define new functions for the M language, and can be used to enable connectivity to new data sources. While this document will focus on defining new connectors, much of the same process applies to defining general purpose M functions. Extensions can vary in complexity, from simple wrappers that essentially just provide "branding" over existing data source functions, to rich connectors that support Direct Query.

Please see the [Data Connector technical reference](#) for more details.

Get Data

Search

All

File

Database

Power BI

Azure

Online Services

Other

?

It's not
here ?

All

Excel

Text/CSV

XML

JSON

Folder

PDF

SharePoint folder

SQL Server database

Access database

SQL Server Analysis Services database

Oracle database

IBM Db2 database

IBM Informix database (Beta)

IBM Netezza

MySQL database

PostgreSQL database

Certified Connectors

Connect Cancel

1. È possibile utilizzare un parametro per passare facilmente le query a un altro server (T / F)?
2. Fornire un esempio per l'utilizzo di Immettere dati?
3. Vantaggi nel mettere query nei gruppi di query (cartelle)?

Esercitazione sugli argomenti appresi

Modulo 5: Tecniche di ottimizzazione

Obiettivi

- Comprendere le migliori pratiche quando si lavora con grandi volumi di dati
- Comprendere le migliori pratiche per utilizzare il potere delle origini dati sottostanti
- Comprendi le migliori pratiche in modalità Query diretta
- **esploreremo il linguaggio di query di Power BI, M, per approfondire le nostre conoscenze e illustreremo come importare dati da una varietà di fonti**

Data Loading/Refresh: High Level

- Power Query / M ha molte funzioni punta e clicca
- L'ordinamento e la configurazione dei passaggi non ottimali possono rallentare lo sviluppo in Power BI Desktop e aggiornarsi una volta pubblicati

Goals

- Spingere il lavoro verso la sorgente dato
- Evitare operazioni di memoria/cpu intensive o sparpagliare le attività fuori di power query
- Ridurre il numero di rallentamenti o di crash di Power BI Desktop

Tip: Disable Background Data in Desktop



Scenario

- Le origini lente possono richiedere del tempo per restituire i dati
 - Molte tabelle/query sono lente o falliscono il refresh
 - Frequenti attese in Power Query per eccessivo utilizzo di cpu/mem

Soluzione proposta

- Disabilitare in Power Query Editor l'opzione

Background Data

Allow data preview to download in the background

- Utilizzare i parametri flag per restituire sottosinsiemi di dati più piccoli durante la modifica delle query

Filosofia dietro al caricamento delle tabelle



Il sistema che ospita il motore mashup deve avere una capacità sufficiente per archiviare tutto ciò che è stato restituito dall'origine (che potrebbe essere miliardi di righe). Tentare di eseguire questa operazione potrebbe comportare l'esaurimento delle risorse del sistema (ad es. Memoria o spazio su disco). **Il tutto anche se poi di fatto mi servono solo tre righe**

Per fortuna, Power Query non gestisce le espressioni di tabella in questo modo semplicistico. Invece, M utilizza:

- Streaming,
- Query folding

- Streaming: Diciamo che M esegue l'espressione usando lo streaming

```
1 let
2   Source = SomeDataSourceReturningATable,
3   Filtered = Table.SelectRows(Source, each [Office] = "Chicago"),
4   Result = Table.FirstN(Filtered, 3)
5 in
6   Result
```

1. Quando vengono richiesti i contenuti di **Result**, **Table.FirstN** nella fase **Result** inizia chiedendo alla fase precedente (**Filtered**) una riga di dati.
2. Quando **Filtered's Table.SelectRows** riceve questa richiesta, si gira e chiede al passaggio **Source** una riga di dati, che **Source** fornisce.
3. Quando **SelectRows** di **Filtered** riceve questa riga, verifica se supera il test **[Office] = "Chicago"** di **Filtered**.
4. In tal caso, **SelectRows** riporta la riga in **Result** (passo 5); in caso contrario, scarta la riga quindi ne richiede un'altra, ripetendo da 1
5. Una volta che **FirstN** di **Result** ha ricevuto una riga, genera quella riga, quindi si gira e chiede a **Filtered** una seconda riga e riparte da 1

- **IMPORTANTE:** Ogni passaggio produce righe una alla volta, richiedendo solo tutte le righe necessarie dal passaggio precedente per produrre la riga richiesta. Lavorando con dati sufficienti (rispetto all'ipotetico approccio semplicistico con cui abbiamo iniziato a memorizzare l'intero output di ogni passaggio in memoria)

- Nell'esempio precedente quando una funzione ha elaborato una riga, l'ha passata o scartata quindi non contiene tutti i dati. Tuttavia, questo non è vero per ogni operazione (**Internal In-Memory**). Supponiamo di aggiungere un passaggio di ordinamento al nostro esempio:

```
1 let
2   Source = SomeDataSourceReturningATable,
3   Filtered = Table.SelectRows(Source, each [Office] = "Chicago"),
4   Sorted = Table.Sort(Filtered,{{"TotalSales", Order.Descending}}),
5   Result = Table.FirstN(Sorted , 3)
6 in
7   Result
```

- L'ordinamento (in genere) richiede il recupero di tutte le righe dal passaggio precedente in modo che possano essere inserite nell'ordine corretto.
- Sorted** ottiene tutte le righe da **Filtered**, le ordina, salva in memoria le righe, quindi restituisce la prima riga dall'insieme ordinato.

- **IMPORTANTE!!!:** Dopo che il Sort restituisce una riga, non ne avrà mai più bisogno, quindi può rimuovere quella riga dalla sua memoria 😊 (miglioria)
- **IMPORTANTE!!!:** Questa memorizzazione interna delle righe in memoria non è una cache persistente; piuttosto, è limitato nell'ambito di una singola chiamata durante una **singola esecuzione della query**.
- **IMPORTANTE!!!:** Non c'è una condivisione di questi set di righe in memoria quando una funzione viene invocata più volte, come **Table.Sort**
- **IMPORTANTE!!!:** Sopra, abbiamo parlato di "in memoria". La memoria può essere paginata su disco. Lavorare con la memoria paginata su disco è molto, molto, molto più lento di lavorare con la memoria memorizzata nella RAM. In alcuni ambienti, il paging inizia quando l'utilizzo della memoria totale di una query supera i 256 MB.

Tip: Attenzione all'ordine delle operazioni



- L'ordine delle operazioni può avere un impatto significativo sulla quantità di dati da conservare in memoria. Per vedere questo, contrapponiamo due varianti di un'espressione. Entrambi producono lo stesso output ma possono differire in modo significativo nelle risorse locali utilizzate.

```
1 let
2   Source = SomeDataSourceReturningATable,
3   Sorted = Table.Sort(Source, {"TotalSales", Order.Descending}),
4   Filtered = Table.SelectRows(Sorted, each [Office] = "Chicago"),
5   Result = Table.FirstN(Filtered, 3)
6 in
7   Result

1 let
2   Source = SomeDataSourceReturningATable,
3   Filtered = Table.SelectRows(Source, each [Office] = "Chicago"),
4   Sorted = Table.Sort(Filtered, {"TotalSales", Order.Descending}),
5   Result = Table.FirstN(Sorted, 3)
6 in
7   Result
```

- Andiamo a vedere perché? 😊

Tip: Attenzione all'ordine delle operazioni

quando è in esecuzione lo streaming, se la query contiene passaggi che contengono righe in memoria, provare a posizionare tutti i passaggi di filtro applicabili prima dei passaggi di mantenimento delle righe. In questo modo, i passaggi del filtro ridurranno la quantità di ciò che deve essere in memoria.

Query folding

Query Folding: Lo streaming può comportare l'estrazione di molte righe che vengono successivamente scartate. Nell'esempio che abbiamo utilizzato, è potenzialmente necessario eseguire lo streaming di miliardi di righe dall'origine per produrre le tre righe di output richieste. Se invece interagissi direttamente con la fonte, probabilmente potresti dirgli esattamente quello che voglio e produrrebbe proprio quello.

Query folding

- Se l'origine fosse un database SQL, dovresti scrivere qualcosa del tipo:

```
1 | SELECT TOP 3 *
2 | FROM Customers
3 | WHERE Office = 'Chicago'
4 | ORDER BY TotalSales DESC;
```

- Potenzialmente non ti vengono inviati miliardi di righe perché tu possa ordinare per trovare le tre che desideri; invece, quell'elaborazione avviene sul sistema esterno e vengono restituiti solo i risultati finali (**molto meno dati che attraversano il cavo**). Dovrebbe essere intuitivamente ovvio quale approccio sia più efficiente. 😊
- Per fortuna, il query folding di M offre la possibilità di sfruttare le prestazioni delle query native senza la necessità di scriverle da soli. (traduce in una query nativa dalla query in M)

Query folding

Con la query query, se il nostro sistema di origine è un database SQL, è abbiamo una query in M come sotto:

```
1 let
2   Source = SomeDataSourceReturningATable,
3   Filtered = Table.SelectRows(Source, each [Office] = "Chicago"),
4   Sorted = Table.Sort(Filtered, {"TotalSales", Order.Descending}),
5   Result = Table.FirstN(Sorted, 3)
6 in
7   Result
```

Questa viene internamente tradotta nel seguente modo

```
let
  Result = Value.NativeQuery(SomeDataSourceReturningATable, "SELECT TOP 3 * FROM Customers WHERE Office = 'Chicago' ORDER BY
    TotalSales DESC;")
in
  Result
```

IMPORTANTE!!!: prova a mettere tutte le operazioni query folding prima di qualsiasi operazione non query folding in modo che la massima quantità di elaborazione possa essere scaricata sulle origini dati.

Tip: Usare il Query Folding

Scenario

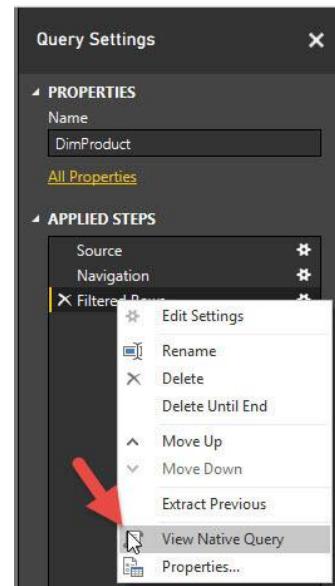
- Alcune trasformazioni possono essere convertite nella lingua di origine nativa

Perchè è una funzionalità da considerare?

- Le operazioni spinte verso il basso sono spesso
- più veloci

Soluzioni proposte

- Tasto destro del mouse, Visualizza query nativa (Query Folding)
- Assicurati che tutti i passaggi del query folding avvengano prima, insieme



Mutabilità delle tabelle

- Una variabile che sembra contenere una tabella (o un elenco) in realtà contiene solo un handle per l'espressione che produce la tabella (o l'elenco).
- Quando vi si accede, quell'handle esegue la logica che produce i dati richiesti. Mentre handle è immutabile per tutta la durata dell'esecuzione della query, i dati restituiti quando vengono invocati non lo sono.
- Questo perché i dati vengono prodotti su richiesta ogni volta che viene coinvolto l'handle. Il fatto che i dati restituiti non siano immutabili può comportare la modifica dei valori durante l'esecuzione di una query M.

Mutabilità delle tabelle

- L'espressione seguente restituisce una coppia di tabelle. Uno detiene tutti i clienti associati all'ufficio di Chicago; l'altro contiene i tre clienti con i maggiori importi delle vendite totali. Entrambe partono dal Source.

```
let
    Source = SomeDataSourceReturningATable,
    ChicagoOffice = Table.SelectRows(Source, each [Office] = "Chicago"),
    LargeCustomers = Table.FirstN(Table.Sort(Source, {{ "TotalSales", Order.Descending }}), 3),
    Result = { ChicagoOffice, LargeCustomers }
in
    Result
```

- nella prima tabella (**ChicagoOffice**). In essi trovi il **cliente ABC**:
 - CustomerID = 123, Customer = "ABC", Office = "Chicago", **TotalSales = 50255**
- nella seconda tabella (**LargeCustomer**), trovi anche il **cliente ABC** (apparentemente, è assegnato all'ufficio di Chicago ed è uno dei tuoi migliori clienti):
 - CustomerID = 123, Customer = "ABC", Office = "Chicago", **TotalSales = 62199**

Mutabilità delle tabelle

Come è potuto accadere?

- Alla luce del fatto che le variabili di tabella ed elenco mantengono gli handler dell'espressione che produce la tabella o l'elenco, questo comportamento ha senso.
- Sia **ChicagoOffice** che **LargeCustomers** sono stati invocati per produrre risultati.
- Le vendite totali per ABC devono essere cambiate tra queste due chiamate all'origine dati (forse un'altra vendita è stata elaborata nel momento intermedio).
- Se questa possibilità non è accettabile, ci sono due opzioni: **rielaborare l'espressione per eliminare le invocazioni multiple** (che potrebbero essere o non essere possibili) o **memorizzare manualmente nella cache** (buffer) l'output.

Usiamo il Buffering

Tip: Buffering

- Quando è necessario stabilizzare una tabella (o un elenco), come si fa?

```
1 | BufferedTable = Table.Buffer(SomeExpressionProducingATable)  
1 | BufferedList = List.Buffer(SomeExpressionProducingAList)
```

- La prima volta che viene richiamato `Table.Buffer`, viene caricato un buffer leggendo tutti i valori dall'origine e salvandoli in memoria.
- I dati in memoria vengono usati dai passi della query durante l'esecuzione della query. I buffer non sono condivisi tra esecuzioni separate della stessa query.

```
let  
    Source = SomeDataSourceReturningATable,  
    ChicagoOffice = Table.SelectRows(Source, each [Office] = "Chicago"),  
    LargeCustomers = Table.FirstN(Table.Sort(Source, {{ "TotalSales", Order.Descending }}), 3),  
    Result = { ChicagoOffice, LargeCustomers }  
in  
    Result
```



```
let  
    Source = SomeDataSourceReturningATable,  
    BufferedSource = Table.Buffer(Source), ChicagoOffice = Table.SelectRows(BufferedSource , each [Office] = "Chicago"),  
    LargeCustomers = Table.FirstN(Table.Sort(BufferedSource , {{ "TotalSales", Order.Descending }}), 3),  
    Result = { ChicagoOffice, LargeCustomers }  
in  
    Result
```

Tip: Buffering

- Il buffering di tutta l'origine potrebbe comportare un problema di risorse, a seconda della quantità di dati contenuti.
- Decidete voi quando utilizzare il buffering, quando i vantaggi che ne conseguono valgono il costo associato.
- Quando esegui il buffer, sii consapevole di quanti dati stai memorizzando. Ridurre al minimo questa quantità, ove possibile.
- Ad esempio, se si sa che tutti i grandi clienti hanno almeno \$ 50.000 di vendite totali, è possibile limitare il buffering ai clienti con vendite superiori a questo importo o per ufficio

```
let
    Source = SomeDataSourceReturningATable,
    BufferedSource = Table.Buffer(Source),
    ChicagoOffice = Table.SelectRows(BufferedSource, each [Office] = "Chicago"),
    LargeCustomers = Table.FirstN(Table.Sort(BufferedSource, {{ "TotalSales", Order.Descending }}), 3),
    Result = { ChicagoOffice, LargeCustomers }
in
    Result
```



```
BufferedSource = Table.Buffer(Table.SelectRows(Source, each [Office] = "Chicago" or [TotalSales] >= 50000))
```

Tip: Buffering (Demo 3)

- Il buffering serve anche per ridurre il query folding quando non serve vediamo con una demo

```
let
    //Connect to SQL Server
    Source = Sql.Database(".\mssqlserver19", "AdventureWorksDW2016"),
    //Get first 2000 rows from FactInternetSales
    dbo_FactInternetSales = Table.FirstN(Source{[Schema="dbo",Item="FactInternetSales"]}[Data], 2000),
    //Remove unwanted columns
    RemoveColumns = Table.SelectColumns(
        dbo_FactInternetSales,
        {"SalesOrderLineNumber", "CustomerKey", "SalesAmount"}),
    //Get sorted list of values from SalesAmount column
    RankValues = List.Sort(RemoveColumns[SalesAmount], Order.Descending),
    //Calculate ranks
    AddRankColumn = Table.AddColumn(RemoveColumns , "Rank", each List.PositionOf(RankValues,[SalesAmount])+1)
in
    AddRankColumn
```



```
to SQL Server
Sql.Database(".\mssqlserver19", "AdventureWorksDW2016"),
st 2000 rows from FactInternetSales
nternetSales = Table.FirstN(Source{[Schema="dbo",Item="FactInternetSales"]}[Data], 2000),
//Remove unwanted columns
RemoveColumns = Table.SelectColumns(
    dbo_FactInternetSales,
    {"SalesOrderLineNumber", "CustomerKey", "SalesAmount"}),
//Get sorted list of values from SalesAmount column
RankValues = List.Buffer(List.Sort(RemoveColumns[SalesAmount], Order.Descending)),
//Calculate ranks
AddRankColumn = Table.AddColumn(RemoveColumns , "Rank", each List.PositionOf(RankValues,[SalesAmount])+1)
in
    AddRankColumn
```

- List.Buffer () memorizza l'elenco ordinato dei valori utilizzati per calcolare il grado in memoria, il che significa che verrà valutato solo una volta; nella query originale sembra che questo passaggio e quelli precedenti vengano valutati più volte.

Tip: Disabilitare il Privacy Settings

Scenario

- Utilizzo dei dati da una fonte per filtrarne un'altra
- Le impostazioni di privacy predefinite impediscono la fuga di dati tra le fonti

Perchè è indesiderato come scenario?

- L'intera tabella deve essere caricata in memoria prima di applicare il filtro

Soluzioni proposte

- Se sicuro, disabilita la privacy o imposta entrambe le fonti su Organizzazione
- Le impostazioni sulla privacy possono essere fonte di confusione, guarda Chris Webb

Scenario

- Grouping/aggregations/merges elaborate in M

Perchè è indesiderato come scenario?

- Se non viene effettuato il query folding, è necessario caricare in memoria l'intera tabella prima di passare al passaggio successivo

Soluzioni proposte

- Considera invece la misura DAX
- Personalizza la query di origine utilizzando Advanced Editor

Scenario

- Merge va usato nell'Editor di query su 2 tabelle di grandi dimensioni (forse milioni di righe)

Perchè è indesiderato come scenario?

- L'intera tabella deve essere caricata in memoria per eseguire il join

Soluzione proposta

- Intervieni alla fonte utilizzando la query personalizzata, quindi manipolala ulteriormente in query Editor

Tip: Non caricare query intermedie in memoria

Scenario

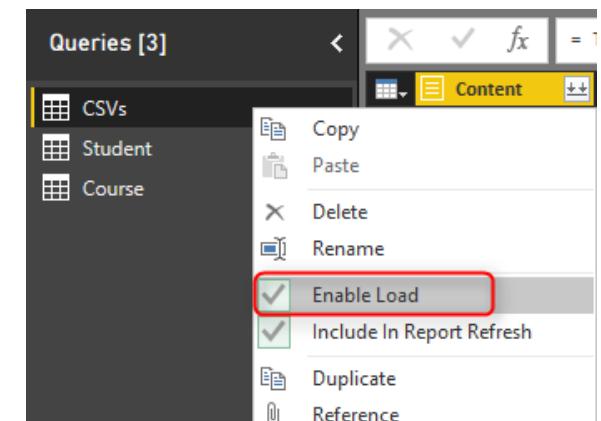
- Il modello contiene query di M che vengono utilizzate solo come tabelle intermedie in altre query. L'autore nasconde queste tabelle per impedire l'accesso dell'utente.

Perchè è indesiderato come scenario?

- Consuma memoria quando viene utilizzato il modello, ma la tabella non viene mai toccata in modo interattivo
- Nascondere solamente non è sicuro

Soluzione proposta

- Disabilitare il caricamento nel Query Editor



Tip: Spegnere Auto Date/Time

Scenario

- I dati di origine hanno molte colonne di tipo date con ampi intervalli
- L'impostazione predefinita è Attiva
- Il sistema utilizza 1/1/1900 o simile per la data vuota

Perchè è indesiderato come scenario?

- Crea molte tabelle di date interne che possono avere dimensioni significative e possono gonfiare modelli più piccoli

Soluzione proposta

- Disabilitare il funzionamento nei settaggi

Ottimizzazione delle query dirette

Scenario

- In base alla progettazione, DQ esegue query parallele per ogni interazione che influisce sui risultati del report
- Le fonti possono essere sopraffatte da molte / costose query
- In alcuni casi, gateway o PBI possono essere il fattore limitante

Goals

- Sfruttare i punti di forza delle origini dati
- Ridurre la concorrenza e l'accodamento
- In generale, tutti i suggerimenti per ridurre il numero / dimensione delle query trarranno grande vantaggio da DQ

Tip: Trasforma le trasformazioni in DQ

Scenario

- Esecuzione di filtri, calc derivati e altre trasformazioni in M

Perchè è indesiderato come scenario?

- In molti casi, i predicati non verranno mandati alla fonte se non in un ordine specifico.
- Le fonti DQ possono essere altamente ottimizzate per join e filtri, quindi non demandarle alla sorgente dati spesso può rallentare Power BI

Soluzione proposta

- Evita le trasformazioni in M
- Assicurarsi innanzitutto che tutte le operazioni con query folding vengano eseguite insieme
- Scrivi codice nativo sull'origine (ad es. TSQL, PL / SQL)

Contact Support

Report Errors, Issues – Support.PowerBI.com

Resources use presentation mode to click the hyperlinks

- Community.PowerBI.com – Community Forum
- [Data Stories Gallery](#) – Get inspired with Data Stories by other Power BI users
- [R-Visuals Gallery](#) – Get inspired by others use of R for analyzing their data
- Visuals.PowerBI.com – Custom PBI visuals and R visuals you can download and use in your story
- [Power BI Blog](#) - weekly updates
- [User Voice for Power BI](#) – Vote on (or submit) your favorite new ideas for Power BI
- Issues.PowerBI.Com – log issues with the community
- [Guided Learning Self Service Power BI training](#)
- [DAX Formula Language](#) – syntax for DAX
- [DAX Patterns](#) – Great website to learn new patterns for the DAX Language
- [Power Query Formula Language](#) – syntax for the “Query” language

Instructors: Marco Pozzan

Laboratorio finale: Lab 2

Dataflow

Obiettivi:

- Comprensione dell'architettura del dataflow
- Creazione e consumo di dataflow
- Creazione di uno schema a stella
- Importazione ed esportazione di dataflow
- Utilizzo delle funzionalità del dataflow premium

Perchè concentrarsi tanto sulla preparazione dei dati

"Analysts spend up to 80% of their time on data preparation delaying the time to analysis and decision making."

- Gartner

Evoluzione del caricamento dati

Stato prima dei dataflow: Self-Service BI senza dataflow

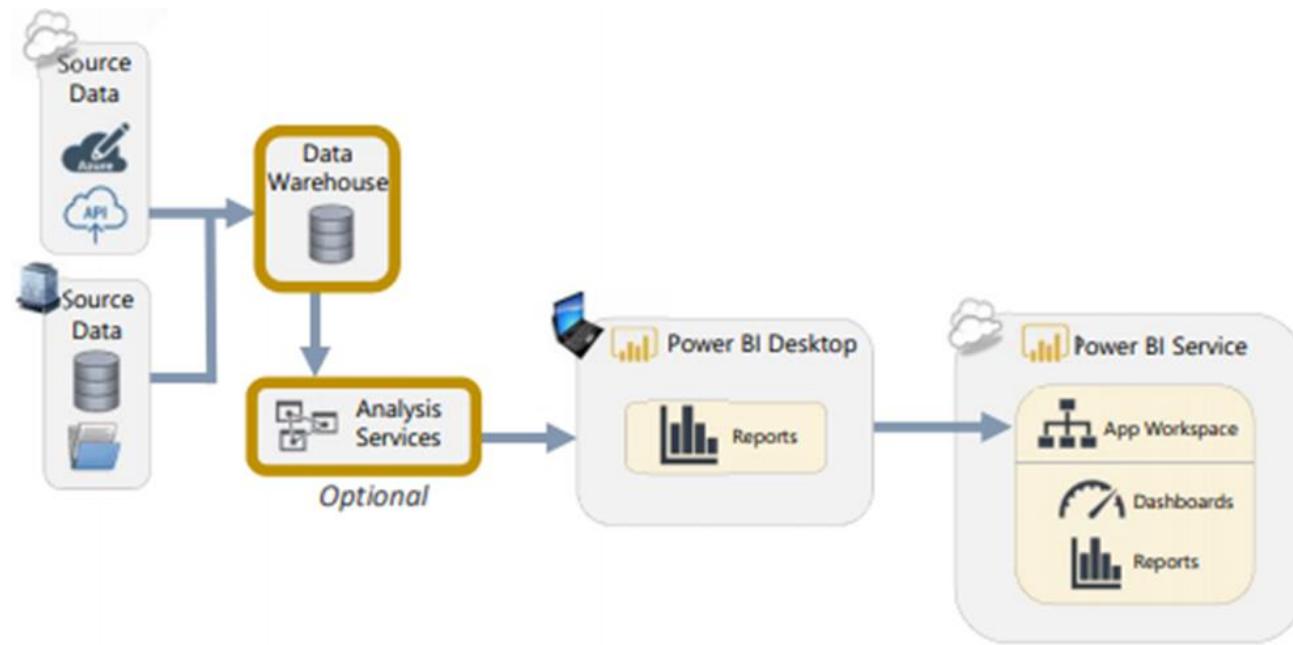
- **Pro:**
 - Si ha una completa libertà nella self-service BI
 - SME (Small and Medium Size Enterprises) si gestiscono la pulizia dei dati
- **Limitazioni:**
 - Query Editor è confinato ad un singolo PBIX File
 - Altri tool non possono accedere all'output del query editor



Evoluzione del caricamento dati

Stato prima dei dataflow: Corporate BI senza dataflow

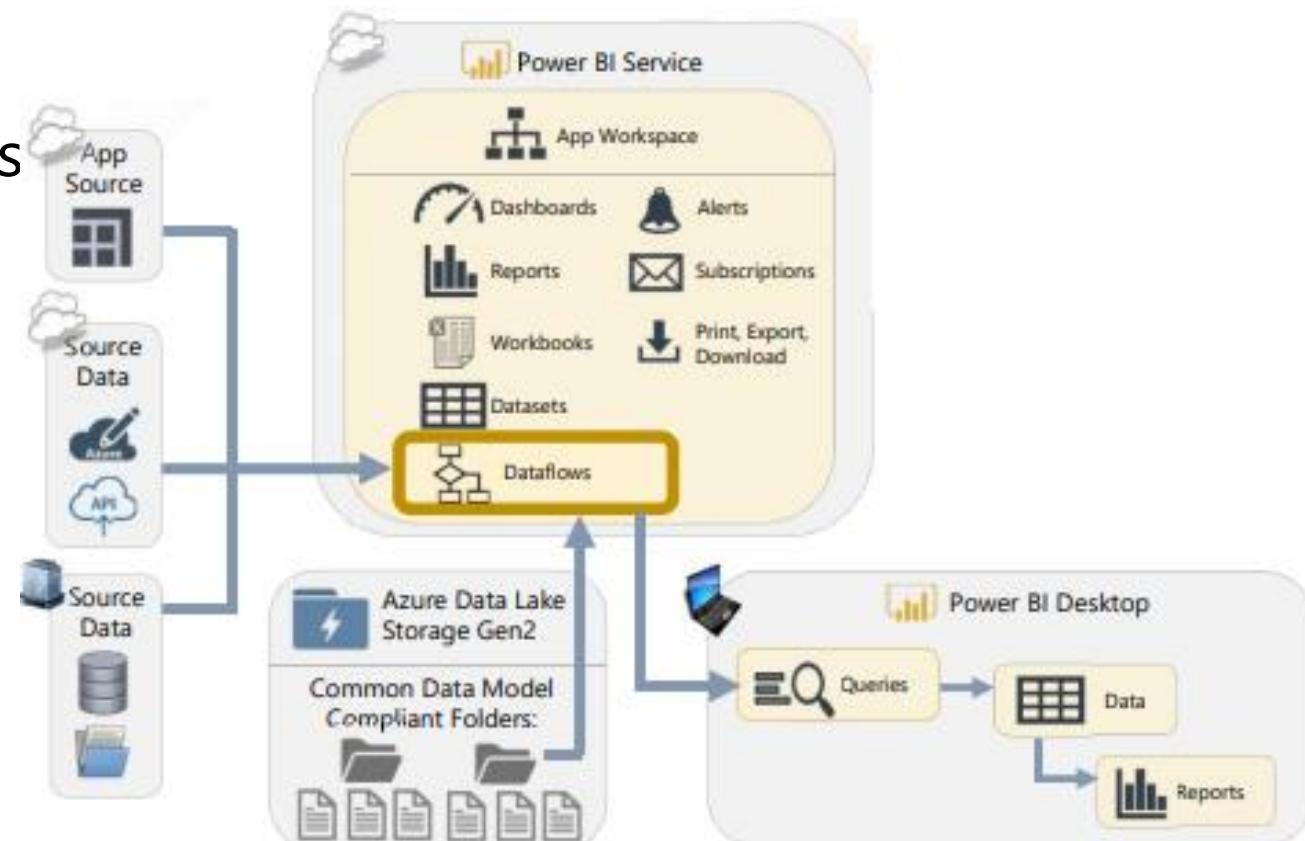
- **Pro:**
 - Si ha la centralizzazione della distribuzione del dato per la corporate BI
- **Limitazioni:**
 - DWH e livello semantico non incontrano tutti i bisogni
 - I grandi DWH non possono reagire velocemente alle esigenze
 - Richiede dei data engineering



Evoluzione del caricamento dati

Stato dopo i dataflow: Corporate BI con dataflow

- **Pro:**
 - La preparazione del dato è familiare attraverso un tool facile per il business
 - La preparazione dei dati è messa a disposizione per molti dataset



Caratteristiche dei dataflow

Compatibilità:

- Preparazione dei dati self-service che può essere riutilizzata
- Riutilizzo dei dataflow da diversi dataset

Fatto per:

- Analisti responsabili per l'acquisizione dei dati e della loro pulizia che vogliono mantenere uniche queste attività
- Analisti che vogliono usare dati preparati da altri colleghi

Scopi:

- Promuovono la consistenza del dato
- Riducono i costi, il tempo e gli expertise richieste

Scenari dei dataflow



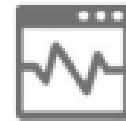
Standardizzazione e riusabilità dei dati: I dati sono abbastanza preziosi da avere molti casi d'uso per molti dataset, diversi tipi di analisi, molti tipi di app



Pre-Processamento: Elaborare set di dati più grandi che superano le risorse disponibili laptop locale o Power BI Desktop



Stage dei dati: Fornire dati per i modellisti di dati di Power BI per completare la preparazione dei modelli

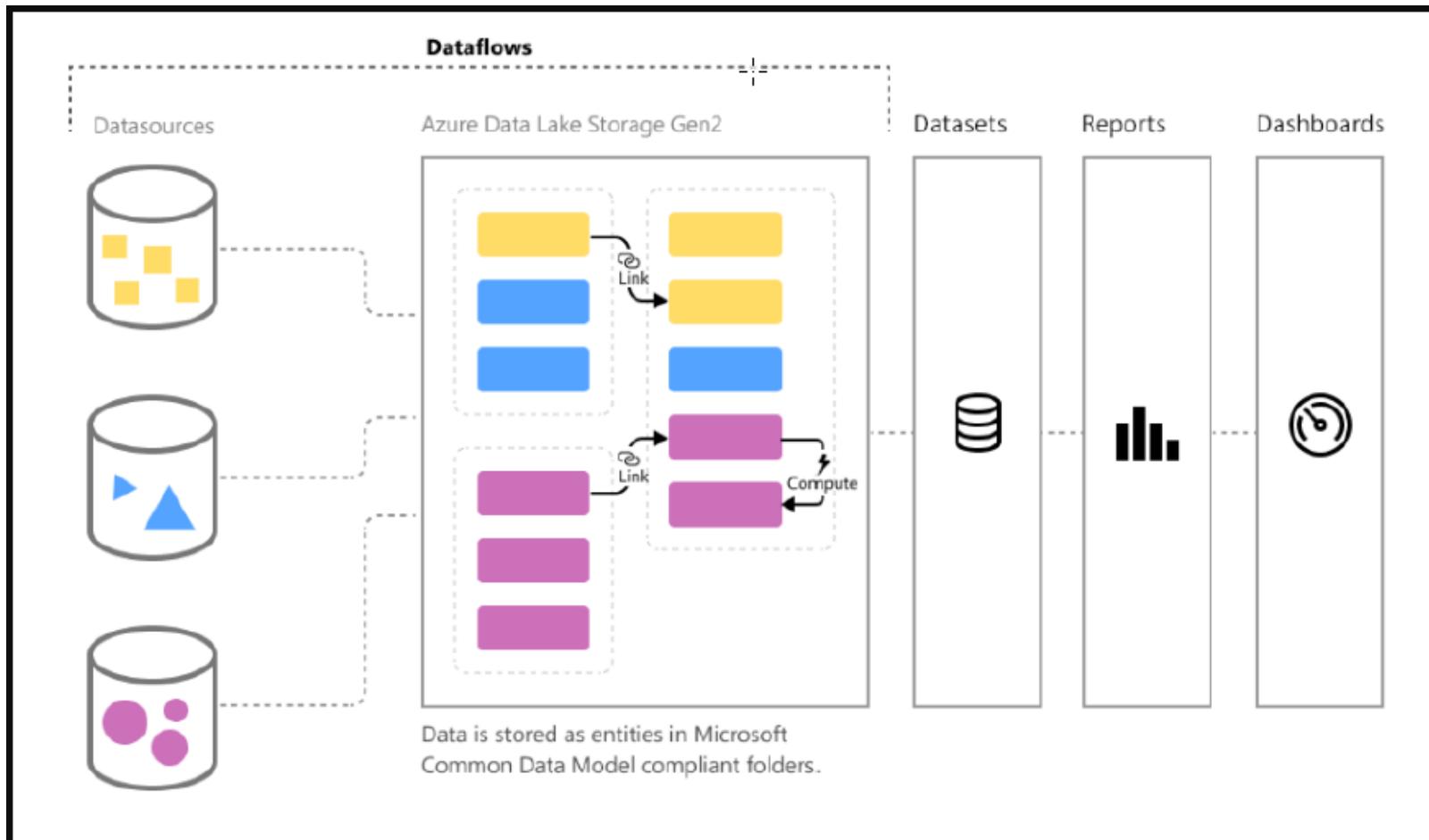


Ridurre il caricamento sui sistemi sorgenti: Ridurre al minimo il numero di query inviate al sistema di origine

Architettura dataflow

I flussi di dati usano l'archiviazione di Azure Data Lake Gen2

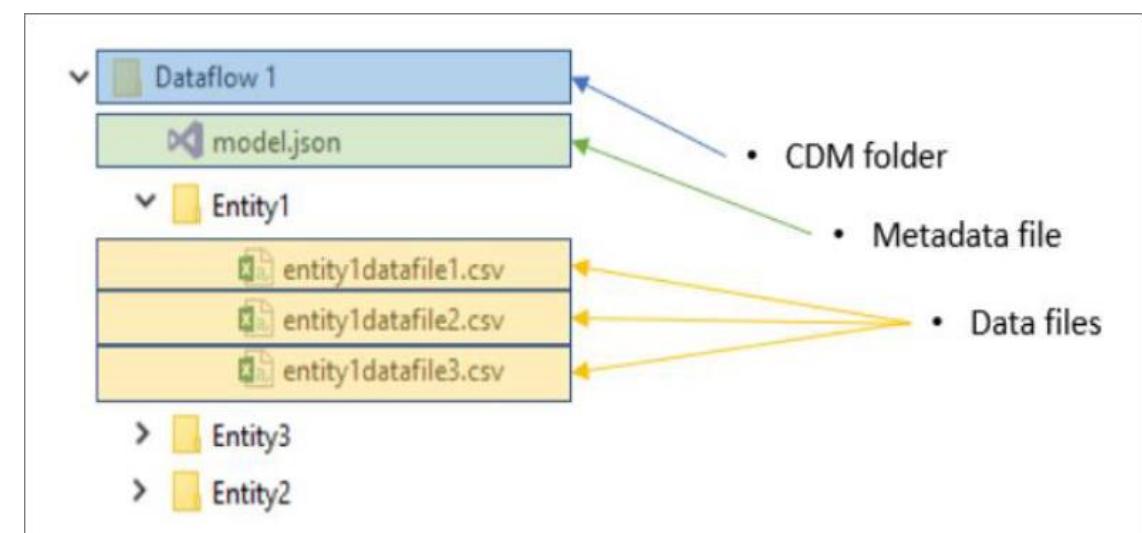
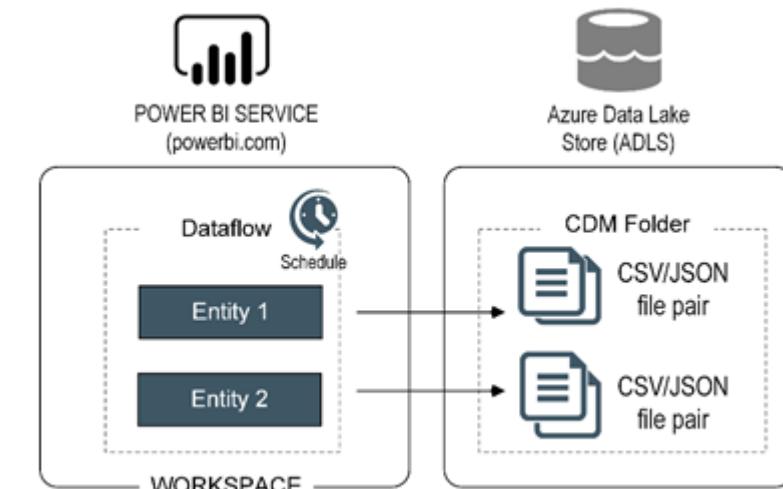
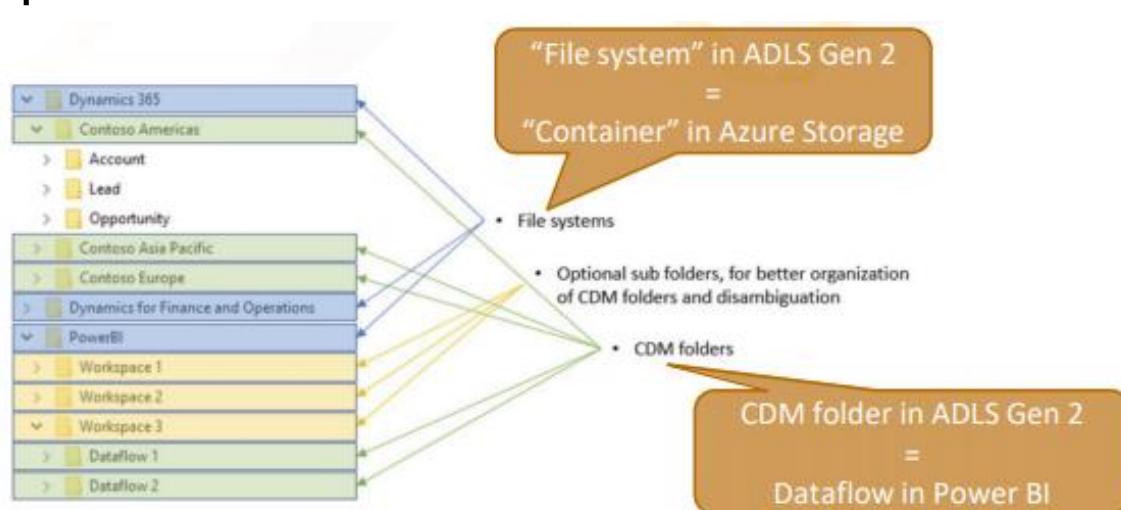
- Archiviazione progettata per soddisfare le esigenze dei big data
- Dataflow serializzati nel formato definito dal Common Data Model (CDM)



Dettagli sullo storage dataflow

Il formato di serializzazione è definito dalle specifiche del Common Data Model

- Metadati del dataflow archiviati nel file **model.json**
- Righe di dati dei dataflow sono archiviate in file CSV
- Per impostazione predefinita, Power BI gestisce l'archiviazione dei dataflow dietro le quinte



Common data model metadata

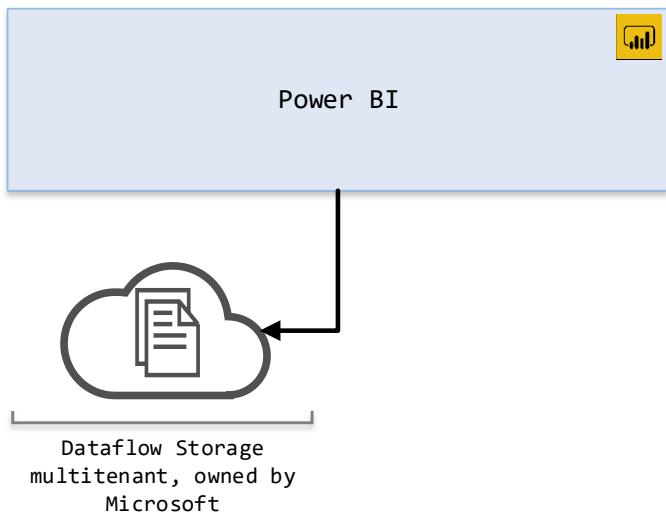
L'output del flusso di dati è memorizzato nel formato CDM.

- Il file model.json contiene metadati relativi alle entità
 - Il file model.json contiene il codice M per le query

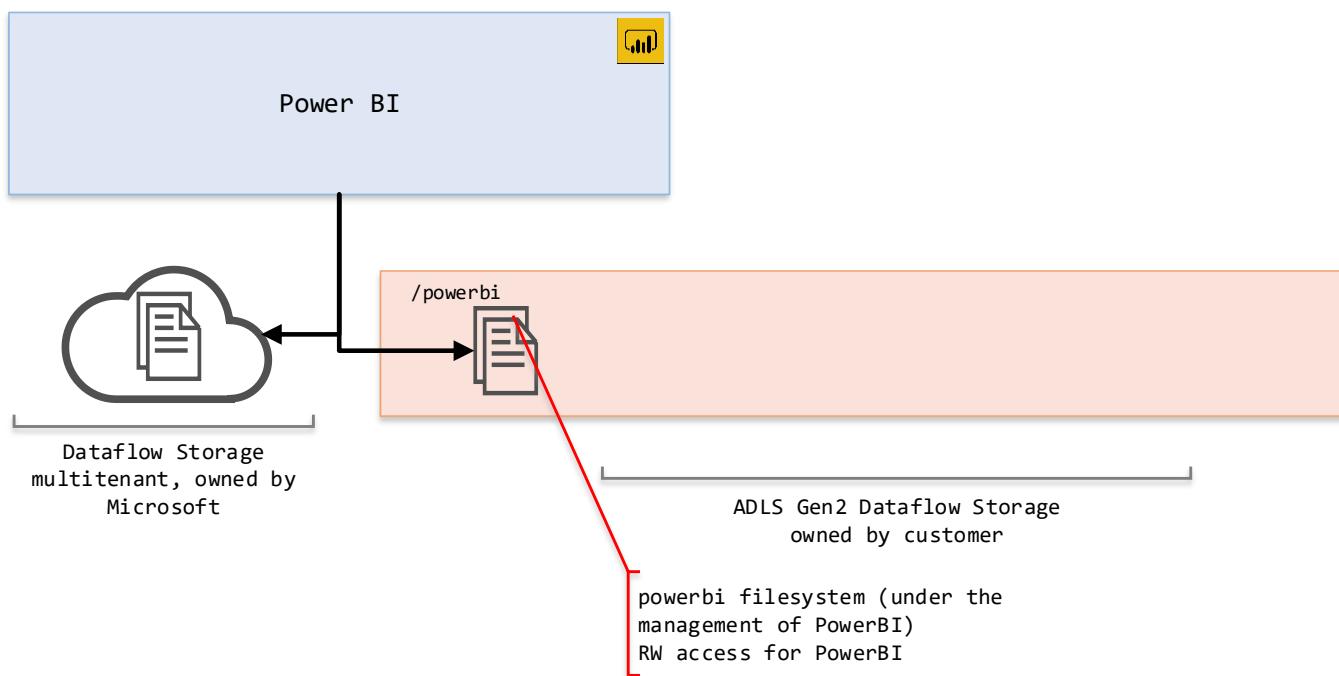
```
[{"name": "Wingtip Sales Dataflow",
"description": "A sample dataflow",
"version": "1.0",
"culture": "en-US",
"modifiedTime": "2019-10-21T17:54:50.1618626+00:00",
"pbi:mashup": {
    "fastCombine": false,
    "allowNativeQueries": false,
    "queriesMetadata": {
        "Customers": { "queryId": "58d2a7e0-0298-4d94-8285-7af1f3d54b15", "quer
        "Products": { "queryId": "10577951-df4b-407c-b6fb-c923880ba1ed", "quer
        "Orders": { "queryId": "ad08816d-be0d-4f6f-b19e-755f23c8fb0f", "quer
        "Sales": { "queryId": "4613190e-da33-4a3a-af5d-a567cbde4dd2", "quer
    },
    "document": "section Section1;\r\n\nshared Customers = let\r\n    Source = $[...]
},
"entities": [
    {
        "$type": "LocalEntity",
        "name": "Customers",
        "entityType": "Table"
    }
]
```

```
{ "$type": "LocalEntity", "name": "Products", "description": "",  
  "ppi:refreshPolicy": { "$type": "FullRefreshPolicy", "location": "Products.csv" },  
  "attributes": [  
    { "name": "ProductId", "dataType": "int64" },  
    { "name": "Product", "dataType": "string" },  
    { "name": "Description", "dataType": "string" },  
    { "name": "Category", "dataType": "string" },  
    { "name": "Subcategory", "dataType": "string" },  
    { "name": "UnitCost", "dataType": "decimal" },  
    { "name": "ListPrice", "dataType": "decimal" },  
    { "name": "Product Image", "dataType": "string" }  
  ],  
  "partitions": [  
    {  
      "name": "Part001",  
      "refreshTime": "2019-10-21T17:59:55.5031318+00:00",  
      "location": "https://wabieus2cdsap1.blob.core.windows.net:443/913b7aae-5  
    }  
  ]  
},  
{
```

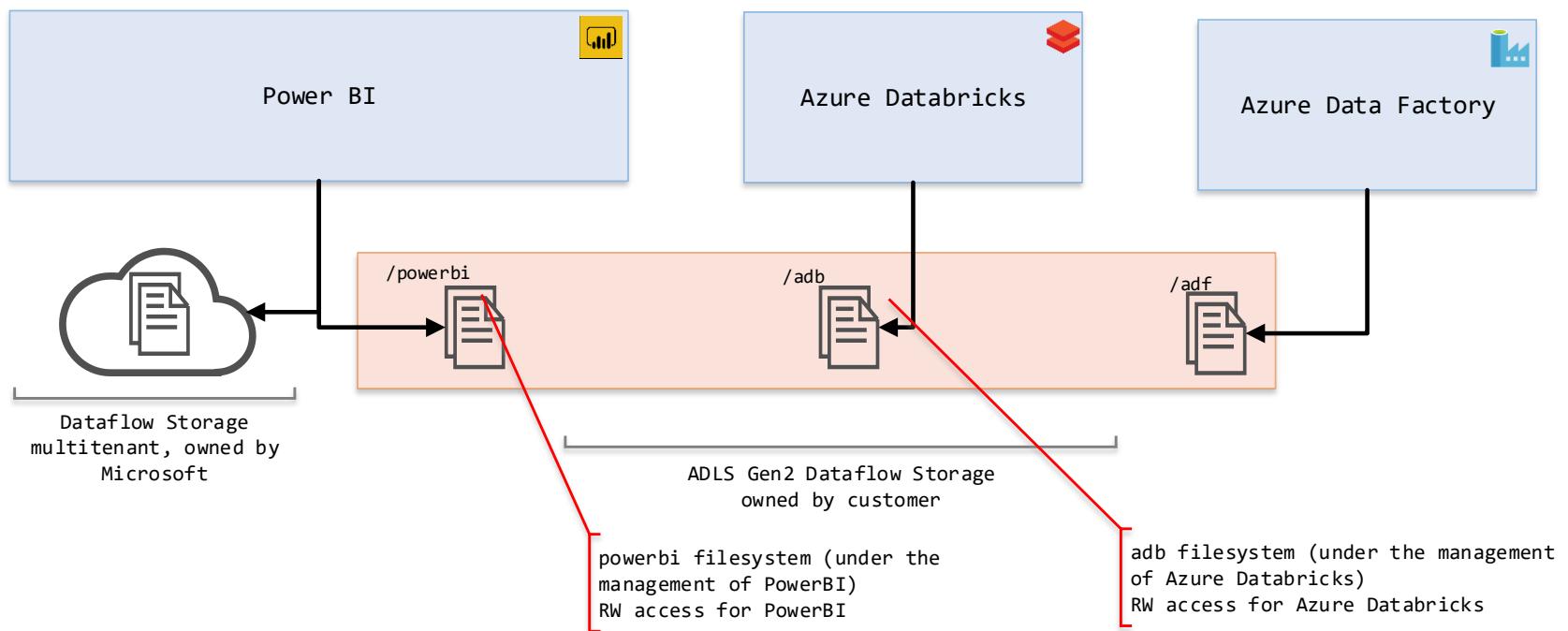
Architettura del common data model



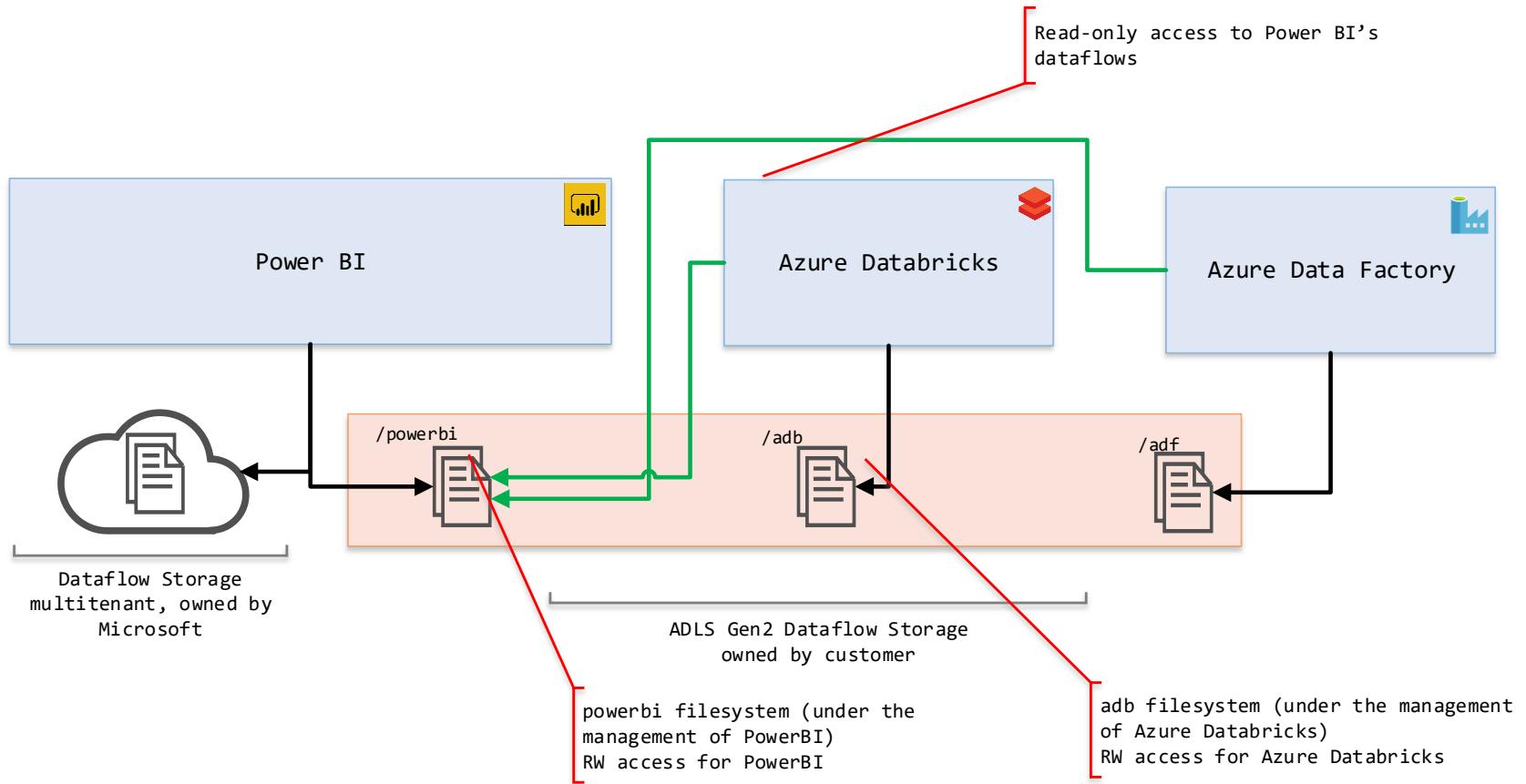
Architettura del common data model



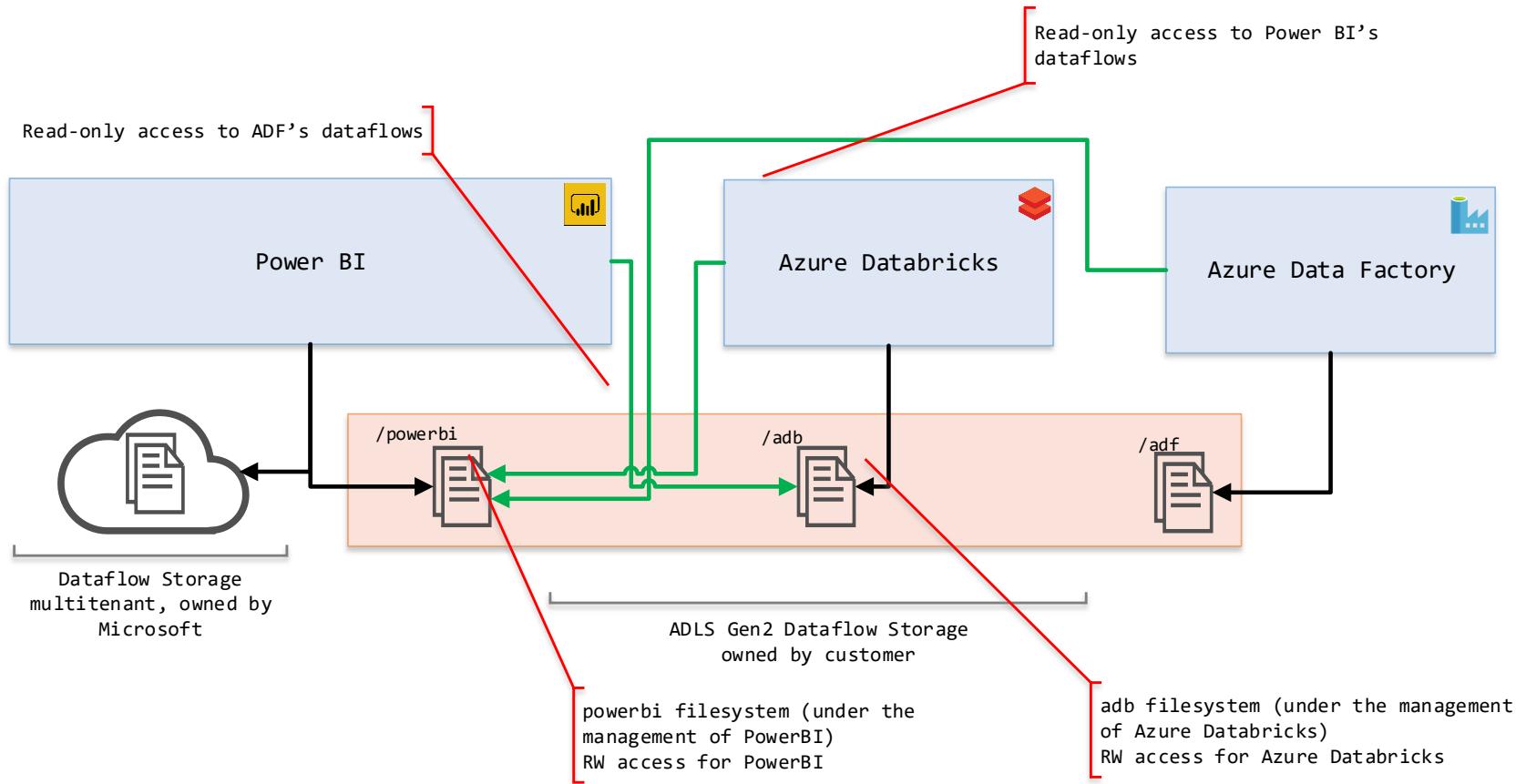
Architettura del common data model



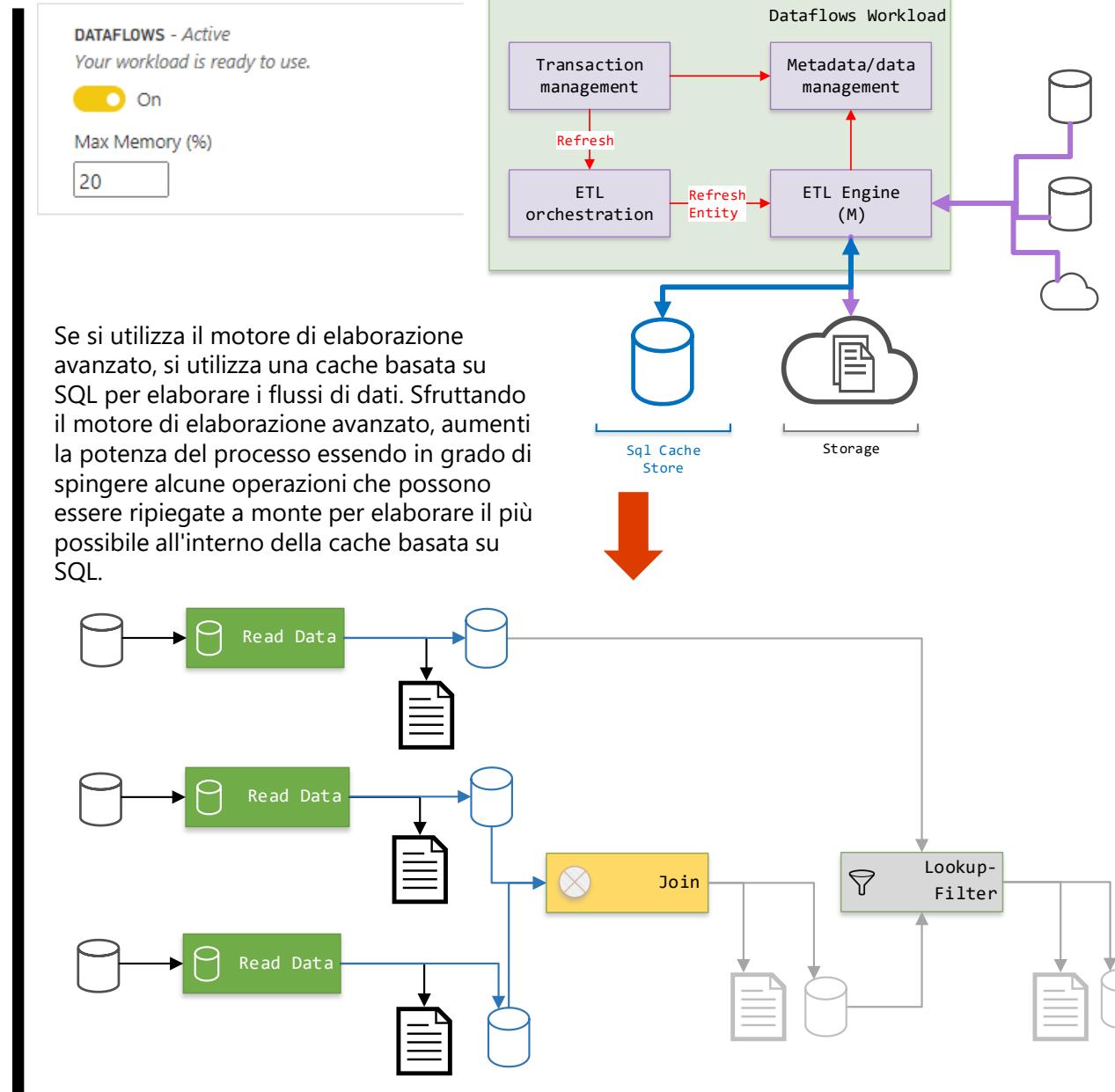
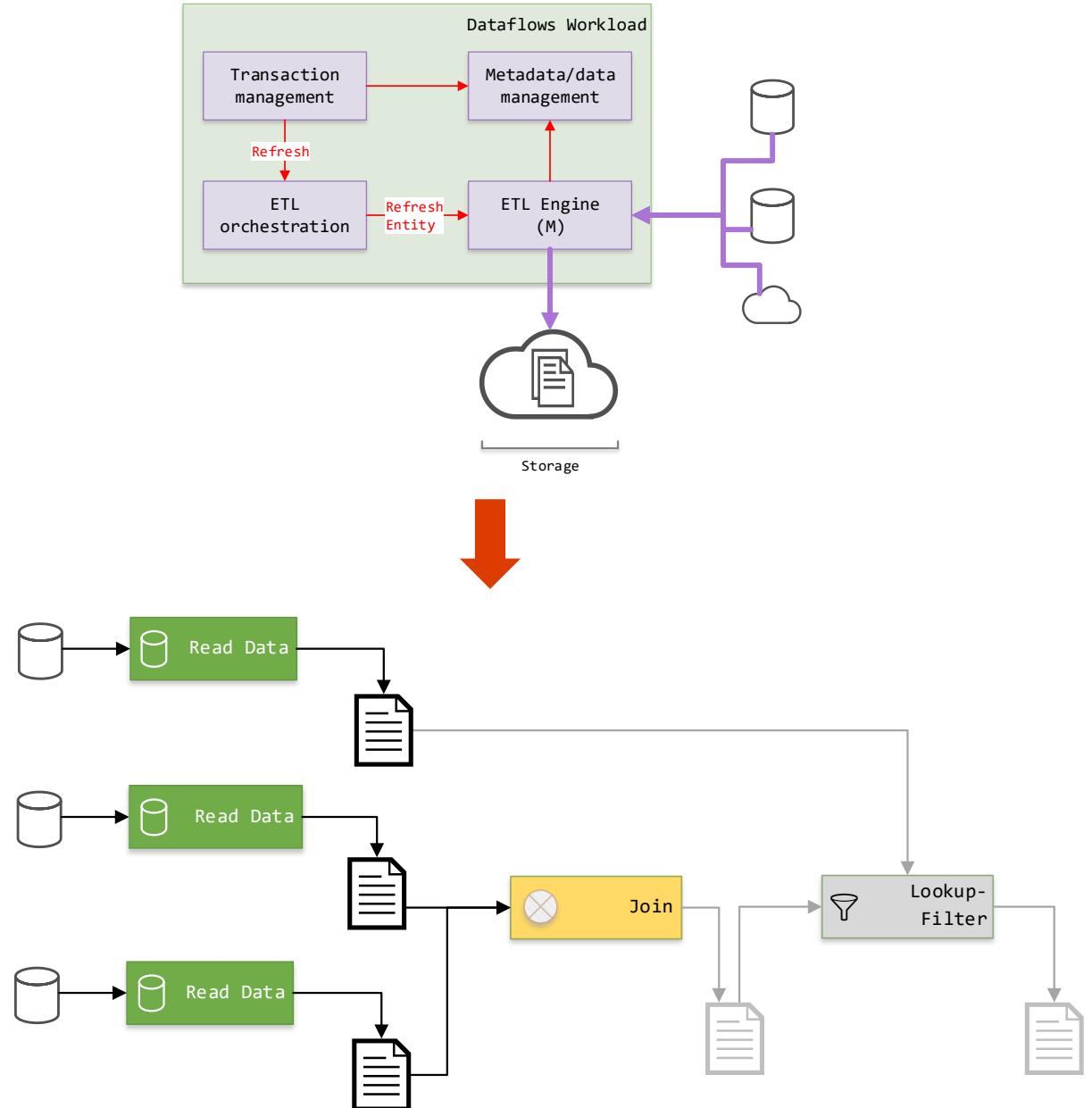
Architettura del common data model



Architettura del common data model



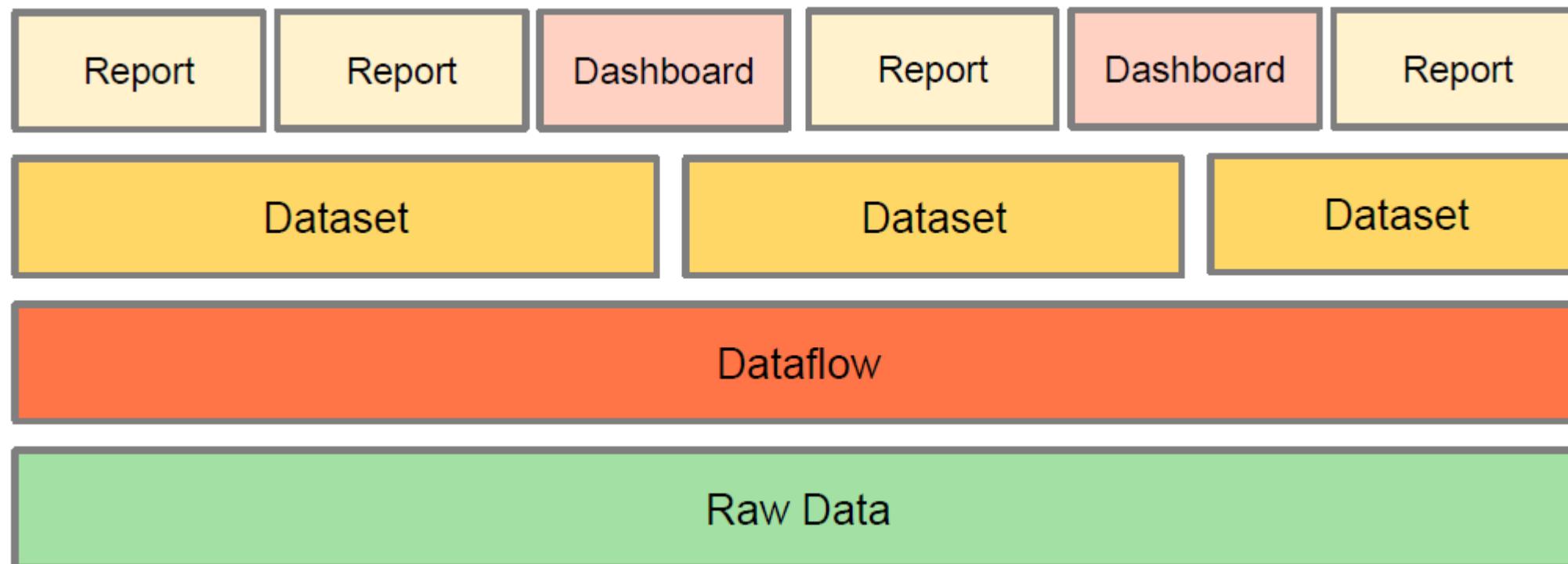
Dettagli sullo storage dataflow con premium capacity



Progetta una soluzione Power BI con dataflow

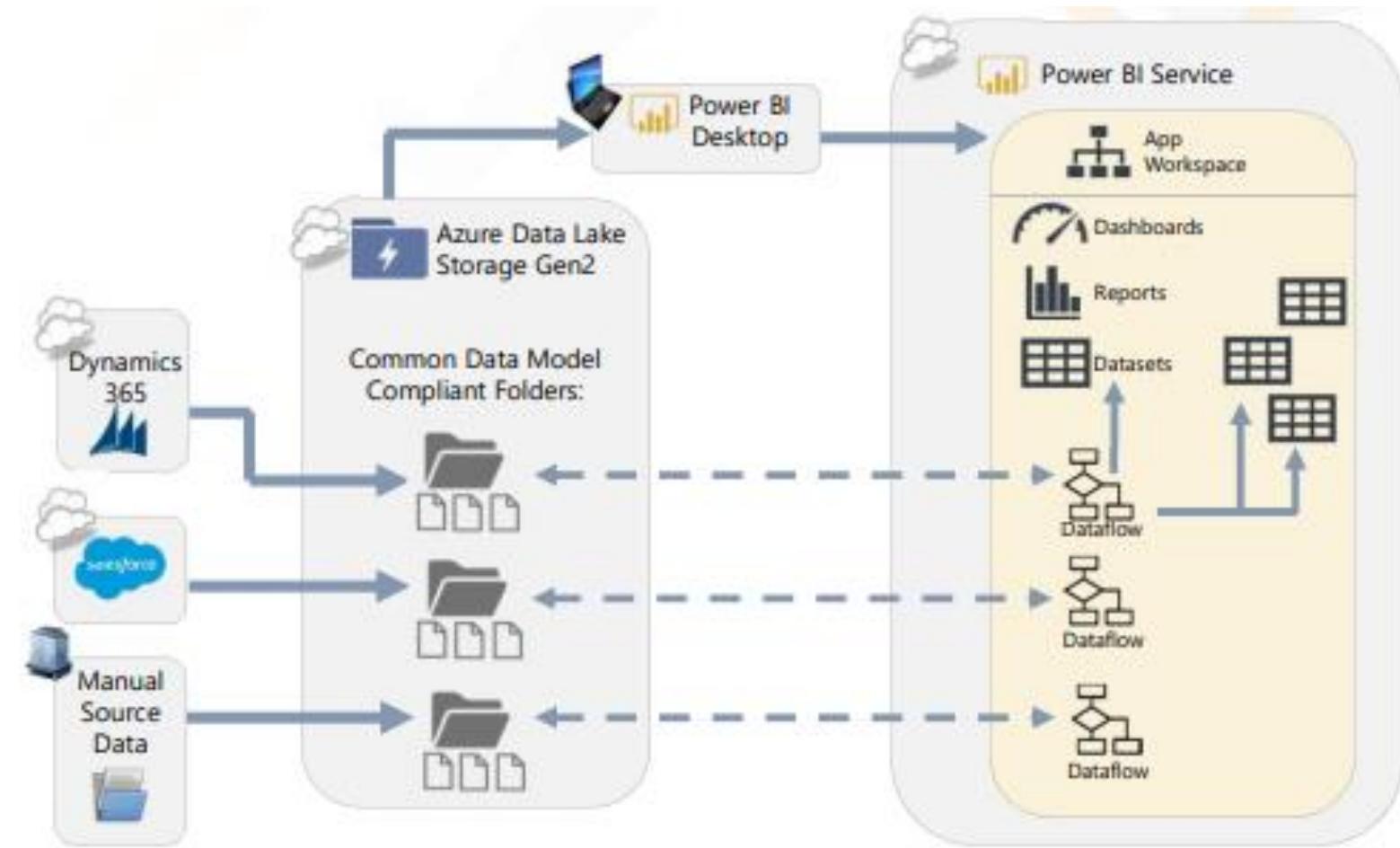
I Dataflow sono utilizzati per raccogliere tutti i dati

- I progetti di Power BI Desktop importeranno dati dai dataflow
- Il lavoro ETL non è più necessario nei progetti di Power BI Desktop



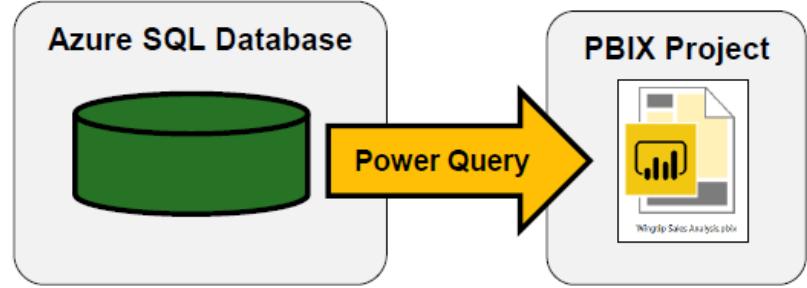
Progetta una soluzione Power BI con dataflow

- Un singolo flusso di dati può alimentare i dati di molti dataset
- Un dataset rimane definito e contiene Calcoli, relazioni, RLS e altro
- Ci sono due aggiornamenti separati schedulati uno per il dataset e uno per il dataflow

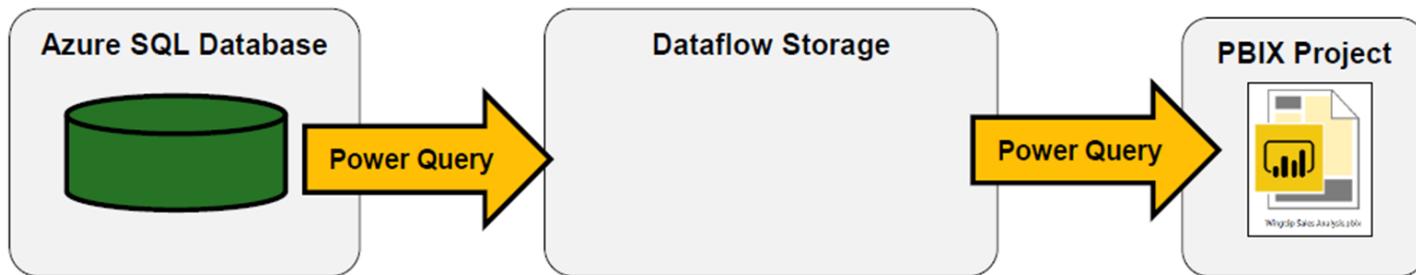


Progetta una soluzione Power BI con dataflow

- Progetto Power BI Desktop senza Dataflow



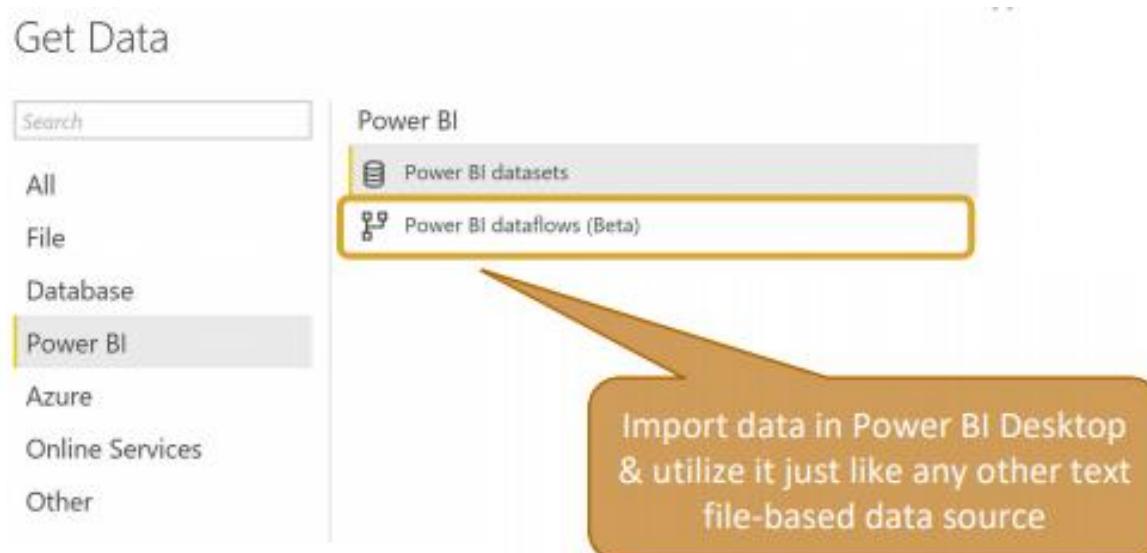
- Progetto Power BI Desktop con Dataflow



Progetta una soluzione Power BI con dataflow

Entità del dataflow utilizzate da Power BI Desktop

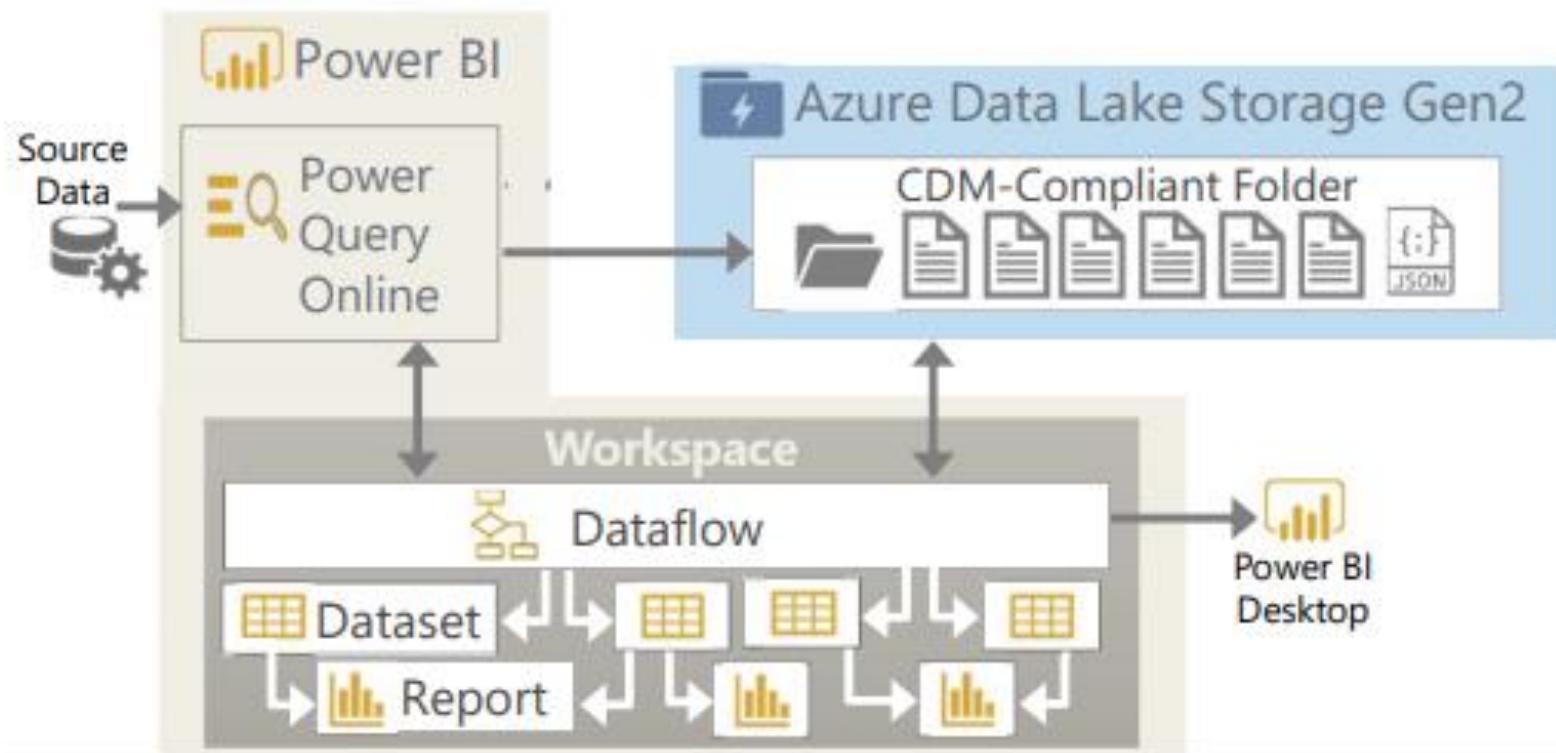
- Utilizzare Power BI Dataflow come sorgente
- Il dataset con il dataflow può essere pubblicato in qualsiasi area di lavoro dell'app



Tre modi di usare i Dataflow: Tipo 1

Struttura dati: Gestita da Power BI

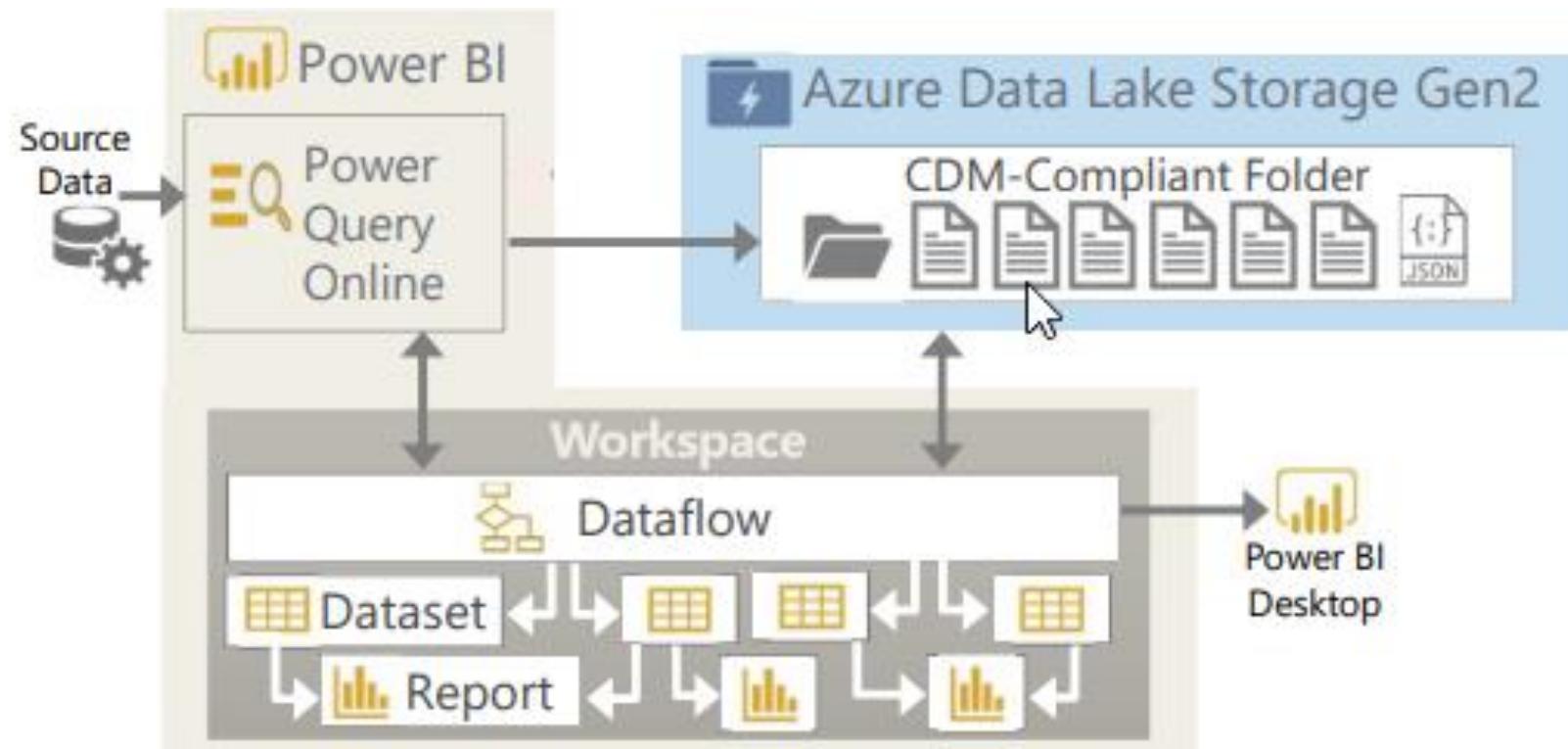
Data lake: I file non sono accessibili da altri tool



Tre modi di usare i Dataflow: Tipo 2

Struttura dati: Gestita da Power BI

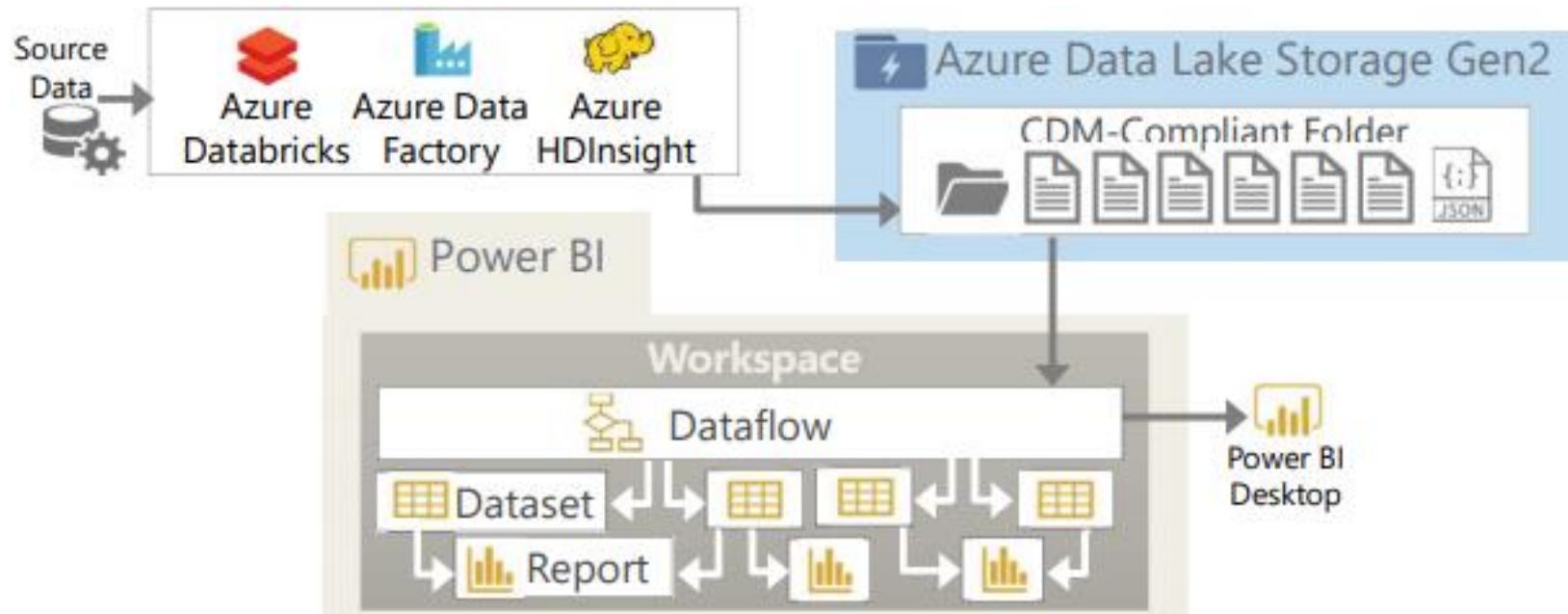
Data lake: lo storage di data lake è associato al tenant di Power BI



Tre modi di usare i Dataflow: Tipo 3

Struttura dati: Gestito da altri tool

Data lake: lo storage di data lake è associato al tenant di Power BI



Entità Dataflow

Esiste uno o più dataflow all'interno di un'area di lavoro (workspace)

- Il dataflow contiene una o più entità
- L'Entity è una tabella con schema ben definito
- L'Entity è popolata eseguendo una query (codice M)

The screenshot shows the 'Entities' blade in the Microsoft Power BI service. At the top, there are tabs for 'Entities' (which is selected) and 'Machine learning models'. A message 'Changes saved' is displayed. Below the tabs are buttons for 'Edit entities', 'Add entities', and 'Close'. The main area displays a table with four rows, each representing an entity:

ENTITY NAME	ENTITY TYPE	ACTIONS
▶ Customers	Custom	
▶ Sales	Custom	
▶ Orders	Custom	
▶ Products	Custom	

Dataflows usano Power Query nel browser

- Esperienza di modifica familiare agli utenti con Power Query
- Power Query è disponibile con un esperienza utente di preparazione del dato di tipo **web-based self service**
- Supporta lo stesso numero di **300+** **transformations** di PBI desktop Power Query (M Engine)
- Correntemente ci sono ~**45 connectors**, incluse connessioni a sorgenti cloud & on-prem via **On-premises data gateway**
- **Usa la potenza del cloud per elaborare grandi volumi di dati in Power BI**
- **Sfrutta il calcolo di Power BI per trasformare i dati in modo semplice e rapido**

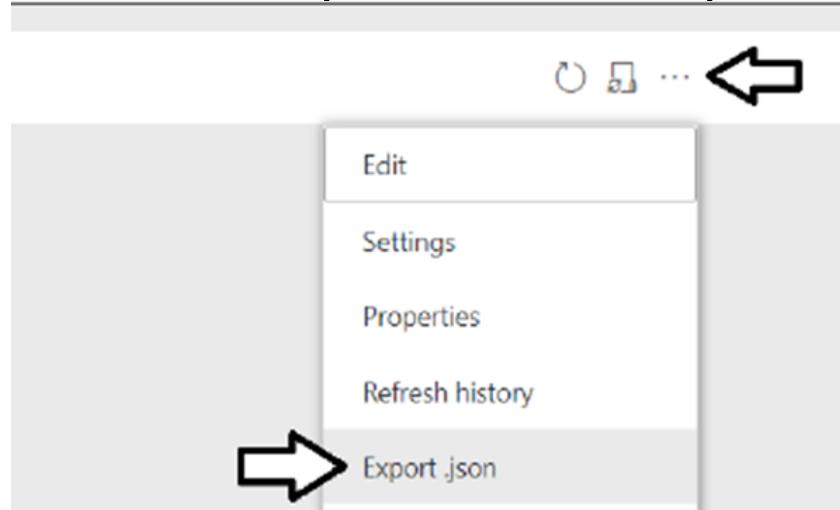
The screenshot displays the Power BI Data Integration interface. On the left, the navigation pane shows 'Data Integration' selected. The main area is split into two panes: 'Edit queries' on the top right and 'Choose data source' on the bottom right.

Edit queries pane: This pane shows a table of data with columns: CustomerID, OrderID, OrderDate, and Region. Transformation steps are listed on the left, including 'Unpivot columns', 'Replace values', and various column operations like 'Split column', 'Format', 'Extract', 'Parse', 'Statistics', and 'Clean'. A preview of the transformed data is shown on the right.

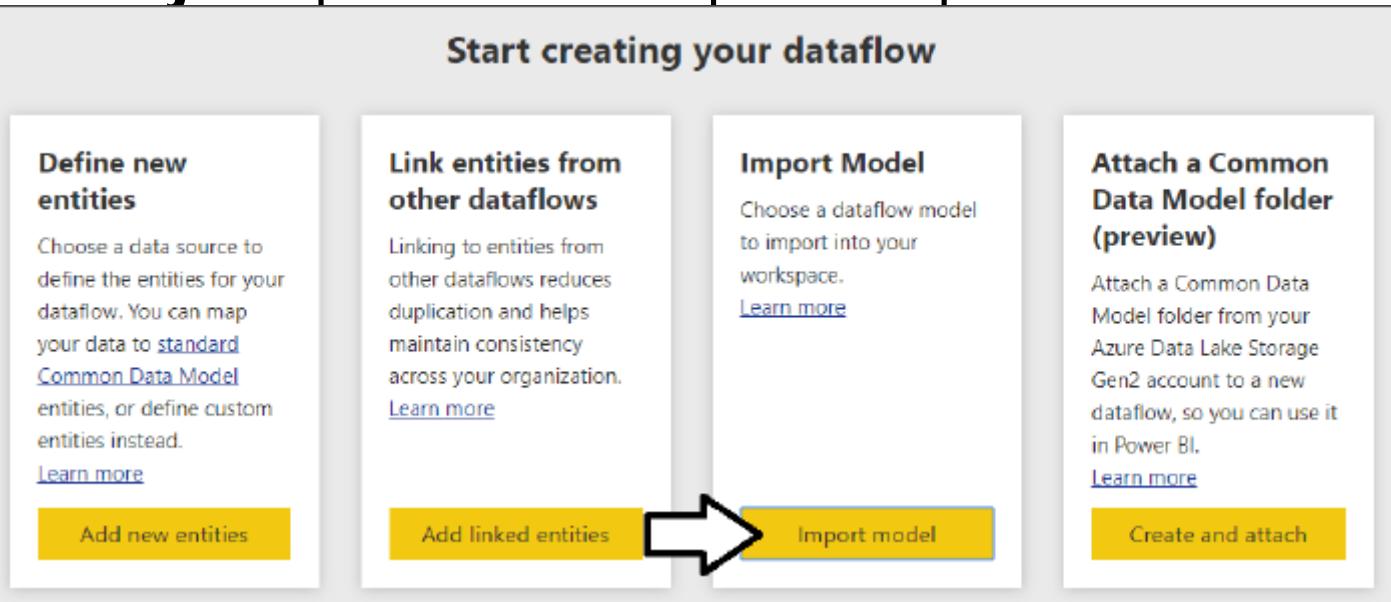
Choose data source pane: This pane lists various data sources and templates. Under 'Data sources', categories include File, Database, Azure, Online services, and Other. Specific connectors shown include Access File, Excel File, JSON File, Text/CSV File, XML File, MySQL database, Oracle database, PostgreSQL database, Google BigQuery, IBM Db2 database, Sybase database, Teradata database, Azure Blob storage, Azure Data Explorer, Azure HDInsight Spark, Azure SQL Data Warehouse, Azure SQL database, SharePoint Online list, Common Data Service for Apps, Microsoft Exchange Online, Salesforce objects, Dynamics 365 for Sales, and more.

Importare ed esportare i dataflow

- Il dataflow può essere esportato come **model.json**



- model.json** può essere importato per creare un nuovo dataflow



Licenze e Funzionalità DataFlow

La creazione del dataflow richiede **Power BI Pro**

- **PRO** I dataflow non possono essere creati in aree di lavoro personali
- **PREMIUM** I dataflow possono essere creati in aree di lavoro personali
- **PREMIUM** Entità collegate (**linked entities**)
- **PREMIUM** Entità calcolate (**computed entities**)
- **PREMIUM** Funzionalità di intelligenza artificiale (AI)
- **PREMIUM** Aggiornamento incrementale
- **PREMIUM** Esecuzione parallela delle trasformazioni
- **PREMIUM** **Direct Query su dataflow**

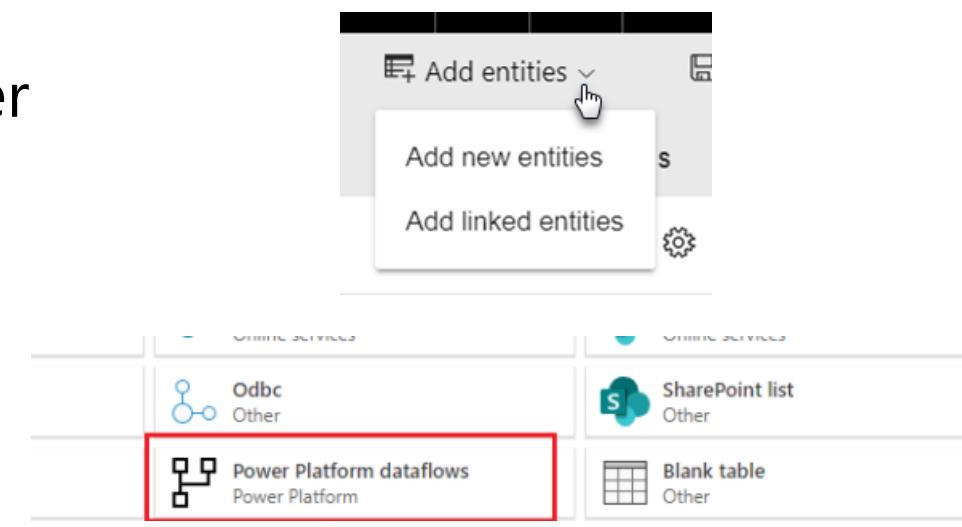
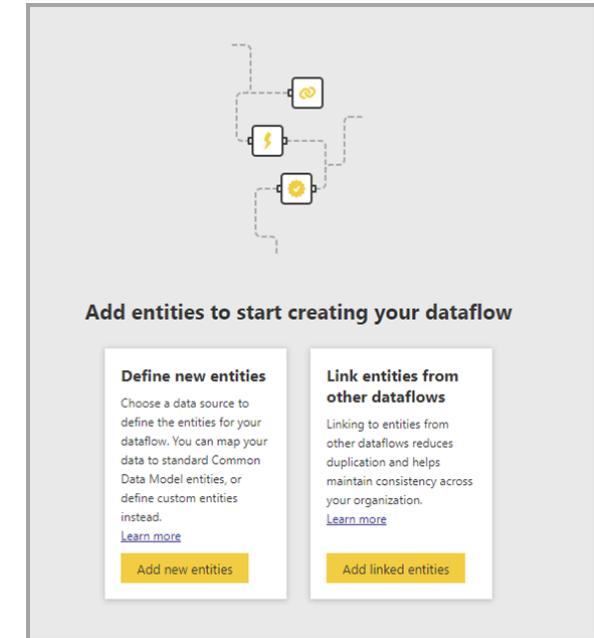
Feature	Pro	Premium
Storage allocation	10GB per user	100TB per Premium node
Data ingestion	Serial ingestion	Parallel ingestion
Refresh frequency	Up to 8x/day	Up to 48x/day
Incremental updates	--	Yes
Linked entities	--	Yes
Computed entities	--	Yes
Cognitive Services AI	--	Yes

Funzionalità premium del Dataflow

- Entità collegate (**linked entities**)
- Entità calcolate (**computed entities**)
- Funzionalità di intelligenza artificiale (AI)
- Aggiornamento incrementale
- Motore di calcolo avanzato
- **Direct Query su dataflow**

Entità collegate (Linked Entity)

- Le entità collegate(**linked entity**) consentono di condividere i dati tra:
 - Dataflow diversi nello stesso workspace
 - Dataflow diversi in diversi workspace
- La creazione di una **linked entity** non duplica i dati di origine
 - È possibile utilizzare un'entità esistente in un altro workspace come origine
 - Utilizza lo stesso codice M utilizzato dal dataset per ottenere dati da un'entità
 - Le entità collegate sono di sola lettura
 - Se vuoi ulteriori trasformazioni crei un'entità calcolata
 - La vista a diagramma semplifica la visualizzazione dell'utilizzo di entità collegate



Entità calcolate (Computed Entity)

- Entità calcolate (**Computed Entity**) basate su di altre entità
 - Consente alle entità di utilizzare altre entità come sorgenti all'interno di un dataflow
 - Se viene usata un'entità calcolata, la parte comune (condivisa) della trasformazione verrà elaborata una volta e archiviata in Azure Data Lake Storage. Le rimanenti trasformazioni verranno quindi elaborate dall'output della trasformazione comune. Nel complesso, questa elaborazione è molto più veloce.
- Scenari utili
 - Stai creando più entità all'interno di un dataflow dallo stesso dato non elaborato (source) e non desideri che i dati vengono presi dall'origine più di una volta
 - Stai avendo problemi a causa del motore di Power Query che ha l'abitudine di richiedere dati più volte durante una singola query



Funzionalità di intelligenza artificiale (AI)

E' possibile usare l'intelligenza artificiale (IA) con i dataflow. Ci sono tre modi diversi di utilizzare l'AI:

- **Servizi cognitivi**
- **Machine Learning automatizzato**
- **Integrazione con Azure Machine Learning**

Cognitive Services

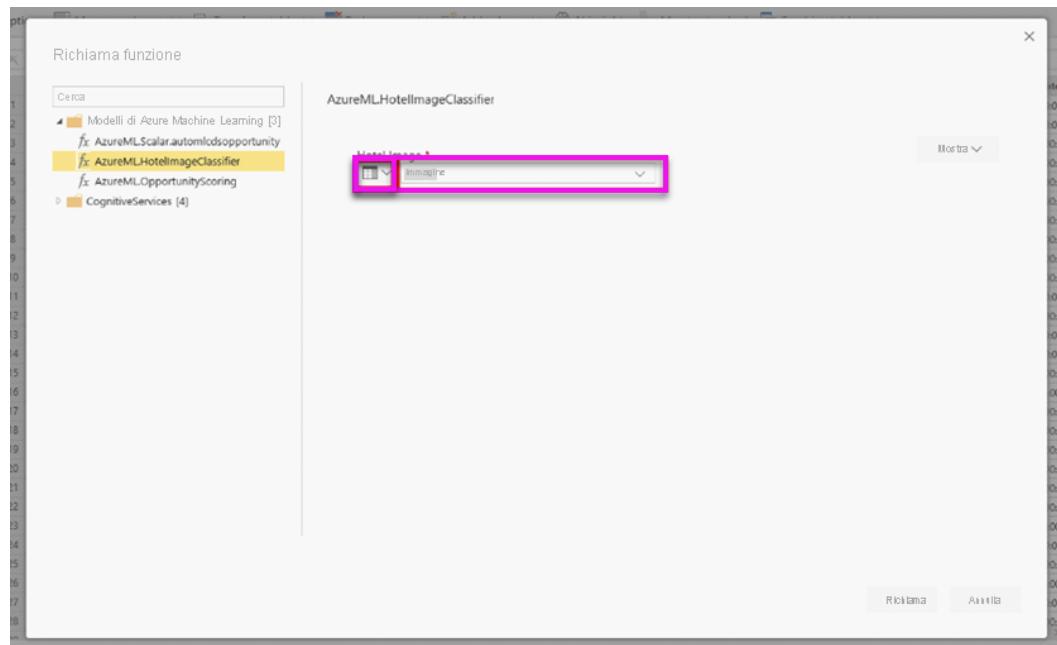
- **Servizi cognitivi** in Power BI è possibile applicare diversi algoritmi per arricchire i dati nella preparazione dei dati con i **dataflow**.
- I servizi attualmente supportati sono
 - Analisi dei sentiment,
 - Estrazione chiavi dalle frasi,
 - Rilevamento lingua
 - Assegnazione di tag alle immagini.
- Le trasformazioni vengono eseguite nel servizio Power BI e **non richiedono una sottoscrizione di Servizi cognitivi di Azure**.

Auto ML

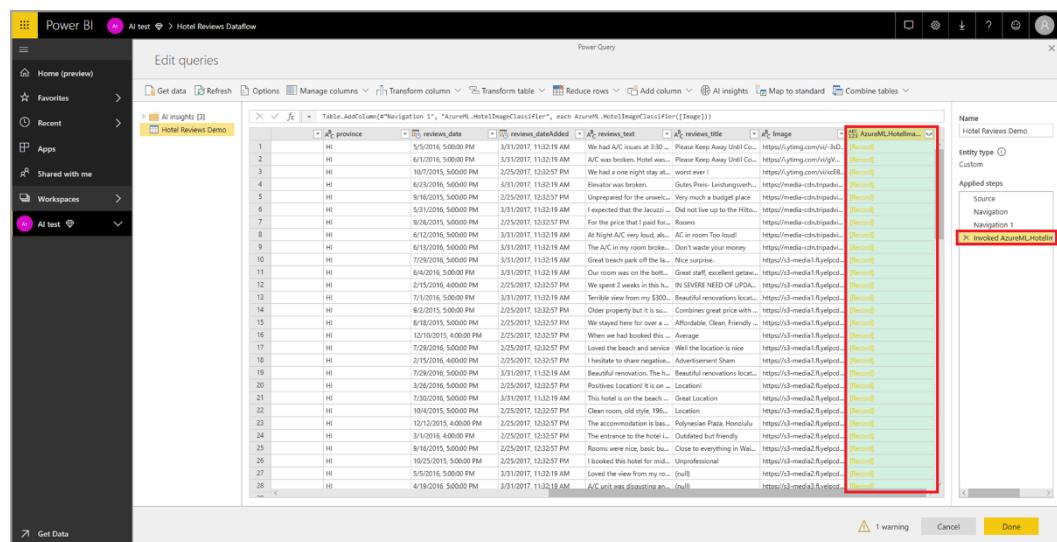
- Machine Learning automatizzato (**AutoML**) per i dataflow consente agli analisti aziendali di eseguire il training, convalidare e richiamare i modelli di Machine Learning (ML) direttamente in Power BI.
- **Include una UI semplice per la creazione di un nuovo modello di ML** in cui gli analisti possono usare i propri dataflow per specificare i dati di input per il training del modello.
- **Il servizio estrae automaticamente le caratteristiche più rilevanti**, seleziona un algoritmo appropriato e ottimizza e **convalida il modello di ML**.
- **Dopo aver eseguito il training di un modello**, Power BI genera automaticamente un report prestazioni che include i risultati della convalida. **Il modello può quindi essere richiamato per tutti i dati nuovi o aggiornati all'interno di un dataflow**.

Integrazione con Machine Learning

- **Ora con Power BI è facile incorporare le informazioni dettagliate dei modelli ospitati in Azure Machine Learning, usando semplici passi e con pochi click.**
- Per usare questa funzionalità, un data scientist può semplicemente concedere all'analista di Power BI l'accesso al modello di Azure Machine Learning usando il portale di Azure.



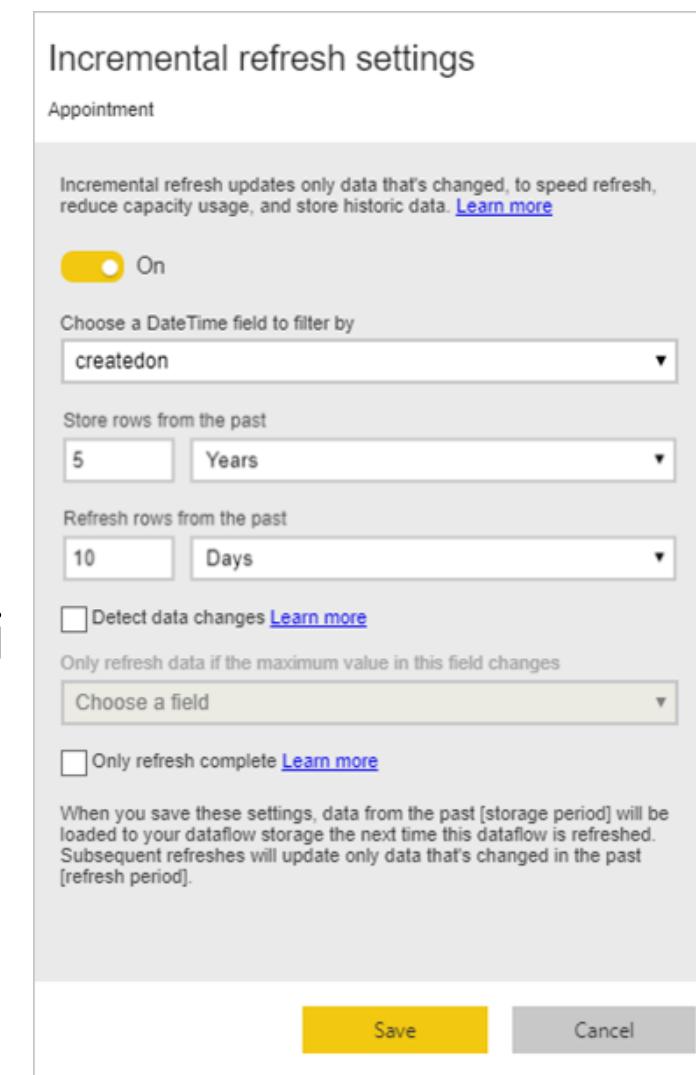
- **All'inizio di ogni sessione, Power Query individua tutti i modelli di Azure Machine Learning a cui l'utente ha accesso e li espone come funzioni dinamiche di Power Query.**



- L'utente può quindi richiamare queste funzioni da Power Query o richiamando direttamente la funzione in M.

Aggiornamento Incrementale

- In alcuni casi trasferire una copia completa dei dati in Power BI ad ogni aggiornamento non è pratico. L'**aggiornamento incrementale**, offre i seguenti vantaggi:
 - L'aggiornamento avviene più velocemente: è necessario aggiornare solo i dati modificati. Ad esempio, aggiorna solo gli ultimi cinque giorni di un dataflow di 10 anni.
 - **L'aggiornamento è più affidabile:** ad esempio, non è necessario mantenere connessioni di lunga durata a sistemi di origine volatili.
 - **Il consumo di risorse è ridotto:** meno dati da aggiornare riduce il consumo complessivo di memoria e altre risorse.
 - L'aggiornamento incrementale è disponibile nei dataflow creati in Power BI e nei dataflow creati in Power Apps.



Motore di calcolo avanzato

- Il motore di calcolo avanzato in Power BI consente ai sottoscrittori di Power BI Premium di usare la propria capacità per ottimizzare l'uso dei dataflow. L'uso del motore di calcolo avanzato offre i vantaggi seguenti:
 - Riduzione drastica del tempo di aggiornamento necessario per i passaggi ETL** con esecuzione prolungata sulle entità calcolate, ad esempio l'esecuzione di operazioni **join**, **distinct**, **filter** e **group by**
 - Eseguire query DirectQuery sulle entità
- Motore di calcolo avanzato solo con Power BI Premium A3



Direct Query su Dataflow (Motore di calcolo avanzato)

- È possibile usare DirectQuery per connettersi direttamente ai dataflow e quindi connettersi direttamente al proprio flusso di dati senza dover importare i dati.
- L'uso di DirectQuery con i dataflow consente i miglioramenti seguenti:
 - **Evita pianificazioni di aggiornamenti separate:** DirectQuery si connette direttamente a un flusso di dati, eliminando la necessità di creare un set di dati importato. Di conseguenza, non sono più necessarie pianificazioni di aggiornamenti separate per il dataflow e il dataset per la sincronizzazione dei dati.
 - **Filtro dei dati:** DirectQuery è utile per lavorare su una visualizzazione filtrata dei dati all'interno di un dataflow.

4 Impostazioni del motore di calcolo avanzato (anteprima)

Configura le impostazioni del motore di calcolo avanzato per questo flusso di dati.

Disabilitato

Disattiva il motore di calcolo avanzato per questo flusso di dati.

Ottimizzato

Il motore di calcolo avanzato verrà attivato solo quando questo flusso di dati sarà collegato a un altro flusso di dati al fine di ottimizzare le prestazioni.

Attivato

Attiva il motore di calcolo avanzato per questo flusso di dati.

Importare da un data lake esterno

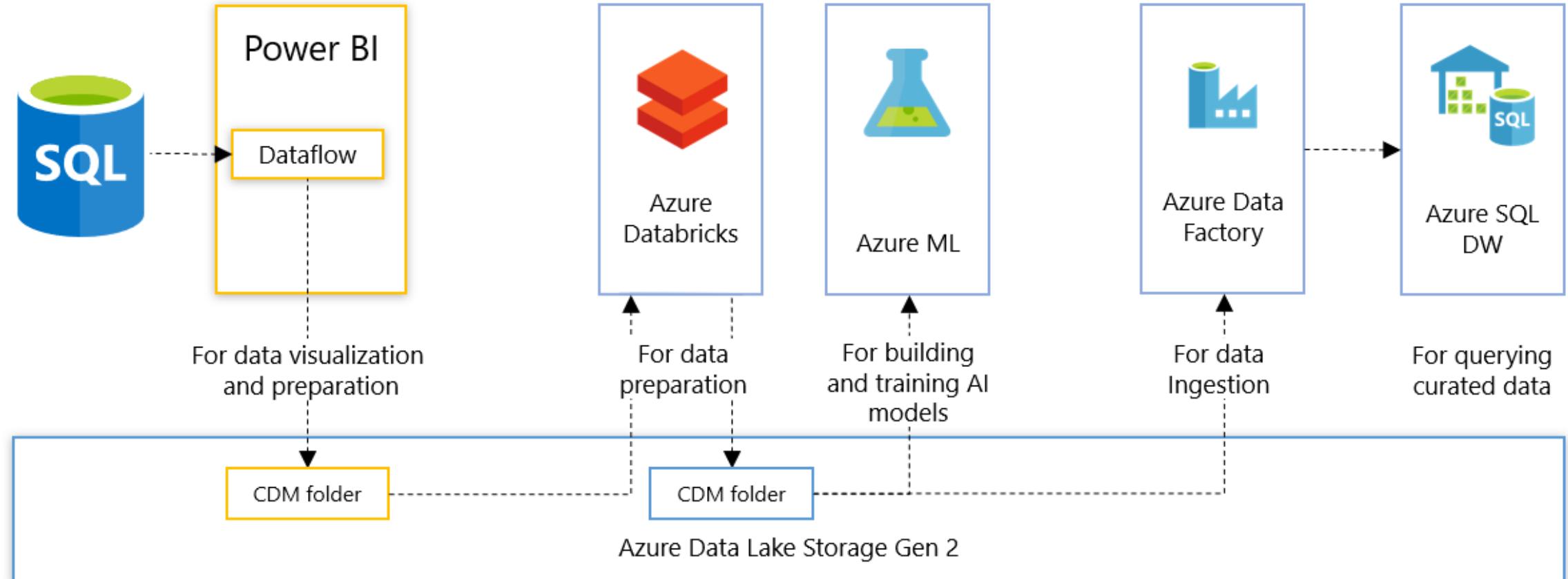
- Importare i dati da un modello che si trova su data lake

The screenshot shows a user interface for importing data from an external data lake. It consists of four main sections:

- Definisci nuove entità**: Describes how to choose a data source to define entities for the data flow. It mentions mapping data to standard Common Data Model entities or defining custom entities. A link to "Altre informazioni" is provided.
- Collega entità da altri flussi di dati**: Explains connecting entities from other data flows. It states that this connects entities from other flows to reduce duplication and ensure consistency across the organization. A link to "Altre informazioni" is provided.
- Importa il modello**: Allows selecting a model from a data flow to import into the workspace. A link to "Altre informazioni" is provided.
- Collega una cartella Common Data Model (anteprima)**: Describes connecting a Common Data Model folder from an Azure Data Lake Storage Gen2 account to a new data flow. A link to "Altre informazioni" is provided.

At the bottom of each section is a yellow button: "Aggiungi nuove entità", "Aggiungi entità collegate", "Importa il modello", and "Crea e collega". The "Crea e collega" button is highlighted with a mouse cursor icon.

Dataflow come base di partenza per altri scenari



Business analysts

Low/no code

Data scientists
Data engineers

Medium to high code

Benefici dei dataflow

- Sostituisce altri strumenti ETL (ad es. Azure Data Factory, Power Automate)
- Disaccoppia il lavoro degli ETL dai set di dati nei progetti PBIX
- Abilita la condivisione di tabelle provenienti dalla sorgente tra set di dati
- Riduce il numero di query sulle origini dati live
- Elimina la necessità di connettere i computer degli utenti direttamente all'origine dati
- Centralizza gli sforzi per pulire e preparare i dati
- Condividi le tabelle che non hanno origine (esempio tabelle del calendario)

Svantaggi dei dataflow

- Aggiunge ulteriore complessità
- I dati devono essere **aggiornati in 2 fasi separate**
- **Non** supporta le funzionalità di **modellazione dei dati di DAX**
- Alcune funzionalità dei dataflow richiedono capacità Premium

Module 6 Lab 3:

Impostazione di un Dataflow

Apendice A: Risposte per il controllo su quanto appreso nei moduli

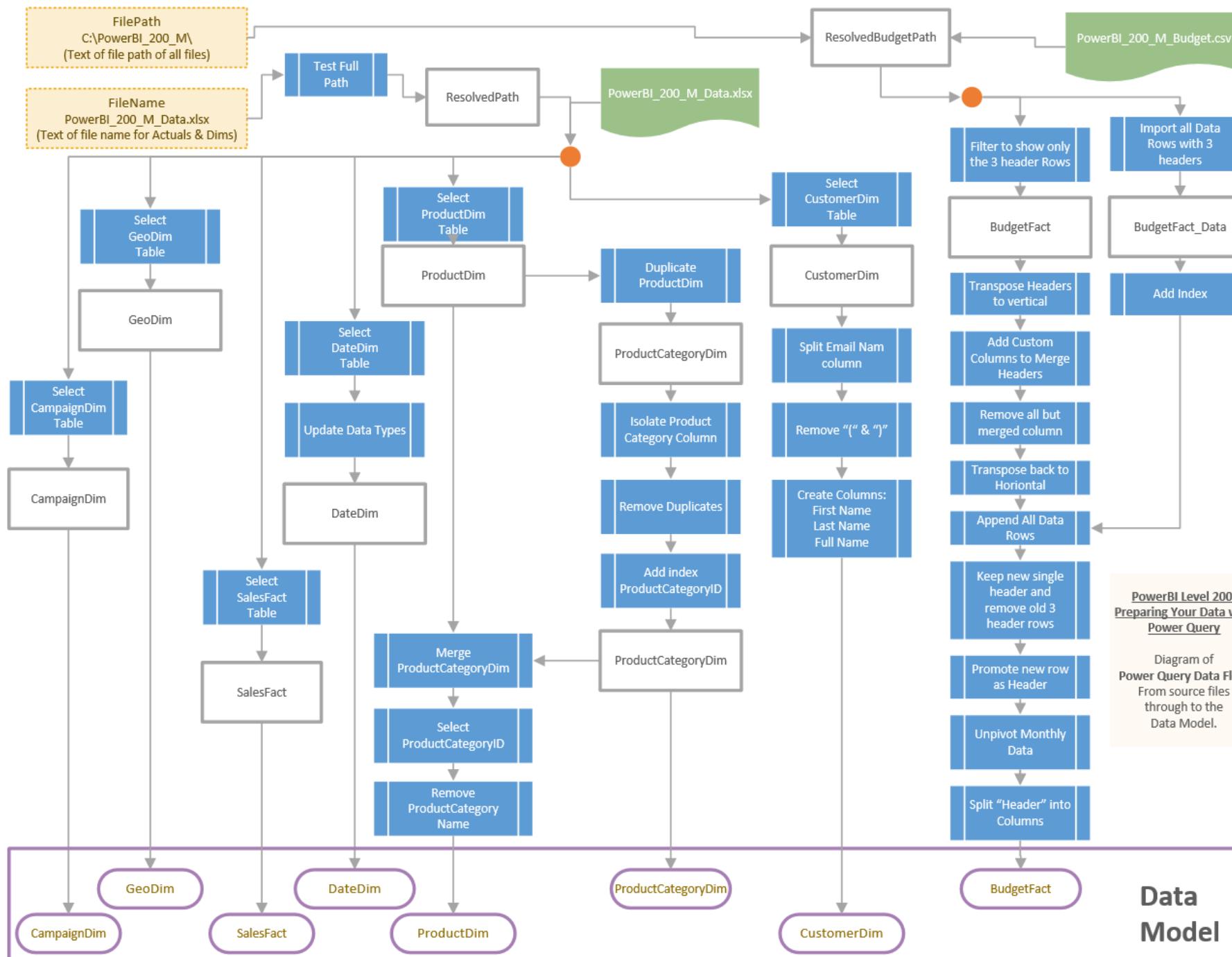
1. Cosa consentono le tabelle nel modello di dati di "dialogare" tra loro?
 1. Le relazioni
2. Quale tipo di schema ha tutti gli attributi per il modello in una singola tabella?
 1. Denormalizzato o "piatto": lo schema del lavello della cucina
3. Quando useresti più tabelle dei fatti
 1. Molteplici motivi
 1. Grani diversi
 2. Diversi attributi dimensionali
 3. Una dimensione con grano diverso: Prodotto v. Categoria prodotto
 4. I fatti provengono da sistemi diversi con cadenze di aggiornamento diverse.

1. Non è possibile importare più tabelle da un'unica fonte (T / F)?
 1. FALSO: PUOI importare più tabelle da un'unica fonte
2. Perché dovremmo aver bisogno di una dimensione a fiocco di neve?
 1. Una dimensione del fiocco di neve può essere utile per rimuovere i campi duplicati da una dimensione grande. O usare il fiocco di neve per una dimensione primaria fuori da un secondo fatto con un grano più alto.
3. Quale sintassi useresti per scoprire se una colonna ha "Abc" come i primi tre caratteri?
 1. Text.StartsWith ([Column], "Abc")
4. O gli ultimi tre?
 1. Text.EndsWith ([Column], "Abc")

1. Quale tipo di trasformazione trasforma tutte le righe in colonne e le colonne in righe?
 1. Trasporre
2. Quale tipo di unione useresti per ottenere tutte le righe da due query, anche se le colonne in merge non corrispondessero?
 1. Full outer join
3. Cosa succede in un'appendice se i nomi delle colonne non corrispondono?
 1. Le colonne extra vengono aggiunte a destra delle colonne di unione. Se il campo non esisteva nell'origine per la riga, il campo è nullo

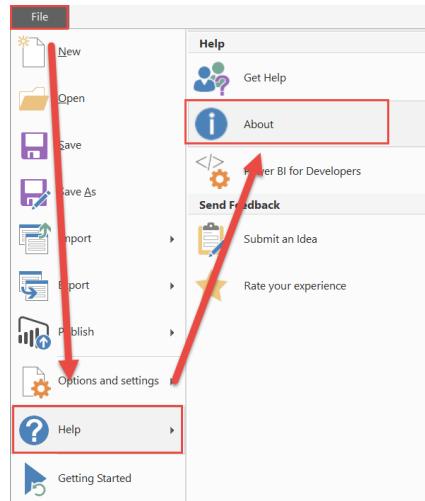
1. È possibile utilizzare un parametro per passare facilmente le query a un altro server (T / F)?
 1. Vero: il parametro rende tutto più semplice, poiché tutti i possibili valori possono essere popolati, quindi l'utente deve solo scegliere.
2. Fornire un esempio per l'utilizzo di immissione dati?
 1. Registro delle modifiche o tabella di mappatura
3. Vantaggi nel mettere query nei gruppi di query (cartelle)?
 1. Più facile da mantenere organizzato il codice.

Apendice B: Esempio di diagramma Data Flow diagram usando Visio



Apendice C: Estrarre tutte le query M in un PBIX

Estrazione di query M



Microsoft Power BI Desktop

Microsoft Power BI Desktop is a companion product to app.powerbi.com.
Cloud: PowerBIDesktopMsitCloud

Version: 2.41.4581.361 64-bit (November 2016)

User ID: 92d66055-9ff8-4437-944f-66beb954ad2e

Session ID: 217807f0-5f12-424d-b858-76c8a27287b1 [Copy session diagnostics to clipboard](#)

[Privacy Statement](#)

Paste in Notepad

Close

```
section Section1;

shared ToolsVSO = let
    Source = Excel.Workbook(Web.Contents(
        "https://microsoft.sharepoint.com/teams/o365-sfb/Shared%20Documents/Reports/Skype%20at%20Mic
        VSO_Sheet = Source{[Item="VVC Tools VSO",Kind="Sheet"]}[Data],
        #"Removed Top Rows" = Table.Skip(VSO_Sheet,1),
        #"Promoted Headers" = Table.PromoteHeaders(#"Removed Top Rows"),
        #"Removed Columns0" = Table.RemoveColumns(#"Promoted Headers", {"Title 2", "Title 4"}),
        #"Added Task Name" = Table.AddColumn(#"Removed Columns0", "Task", each if [Title 9] <> null
        if [Title 8] <> null then [Title 8] else
        if [Title 7] <> null then [Title 7] else
        if [Title 6] <> null then [Title 6] else
        "Other"),
        #"Added Work Item Type" = Table.AddColumn(#"Added Task Name", "Work_Item_Type", each if [Ta
        if [Title 5] <> null then "Story" else
        (if [Title 3] <> null then "Feature" else
        "Epic")),
        #"Reordered Columns" = Table.ReorderColumns(#"Added Work Item Type", {"ID", "Work Item Type"
        "Title 6", "Title 7", "Title 8", "Title 9", "Assigned To", "State", "Tags", "Area Path", "C
        "Story Points", "Target Date", "Completed Work", "Remaining Work", "Stack Rank", "Task"}),
        #"Split Column by Delimiter" = Table.SplitColumn(#"Reordered Columns", "Iteration Path",Spli
        true), {"Iteration Path.1", "Iteration Path.2"}),
        #"Changed Type1" = Table.TransformColumnTypes(#"Split Column by Delimiter", {{"Iteration Pat
        #"Added Custom1" = Table.AddColumn(#"Changed Type1", "Iteration Path", each if [Iteration P
        Path.2]<>" then ([Iteration Path.2] & "-2") else null)),
        #"Renamed Columns" = Table.RenameColumns(#"Added Custom1", {"Iteration Path", "Sprint Plann
        #"Removed Columns" = Table.RemoveColumns(#"Renamed Columns", {"Iteration Path.1", "Iteration
        #"Added Epic ID" = Table.AddColumn(#"Removed Columns", "Epic ID", each if [Work_Item_Type]
        #"Added Feature ID" = Table.AddColumn(#"Added Epic ID", "Feature ID", each if [Work Item Ty
```

Apendice D: Altri argomenti di M.

Date e Time Literals

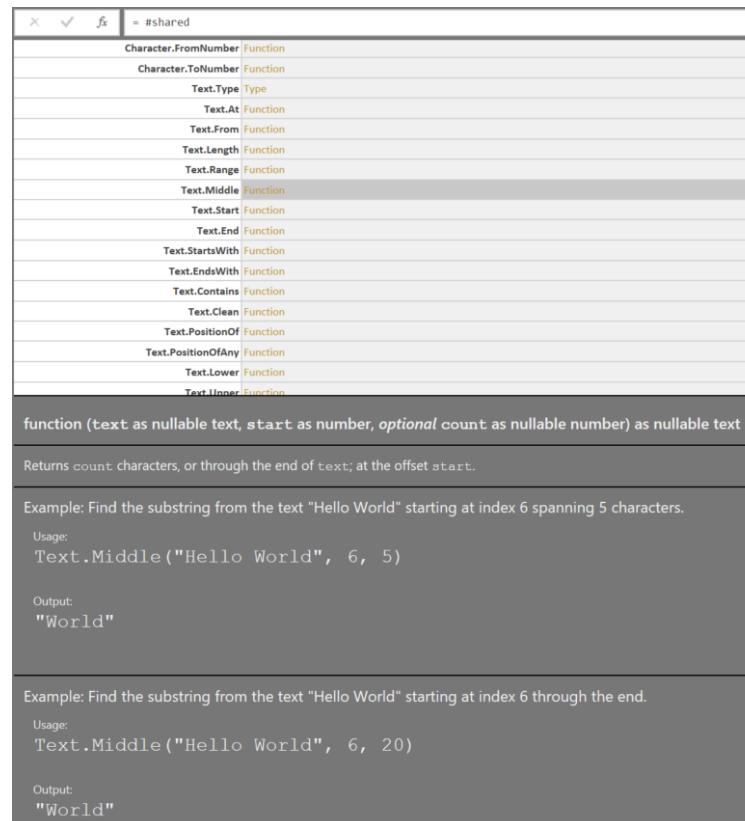
Codifica un modo statico una Date o time literal nelle tue query

X ✓ fx = Table.AddColumn(Source, "Result", each Expression.Evaluate([#"Example (as text)"]))

Function	Example (as text)	Components	Result
1 #date	#date(2015, 4, 22)	year, month, day	4/22/2015
2 #time	#time(09, 15, 00)	hour, minute, second	9:15:00 AM
3 #datetime	#datetime(2015, 4, 22, 09, 15, 00)	year, month, day, hour, minute, second	4/22/2015 9:15:00 AM
4 #datetimezone	#datetimezone(2015, 4, 22, 09,15,00, 09, 00)	year, month, day, hour, minute, second, timezone	4/22/2015 9:15:00 AM +09:00
5 #duration	#duration(2, 1, 30, 45)	day, hour, minute, second	2.01:30:45
6 #infinity	#infinity	no extra components	Infinity

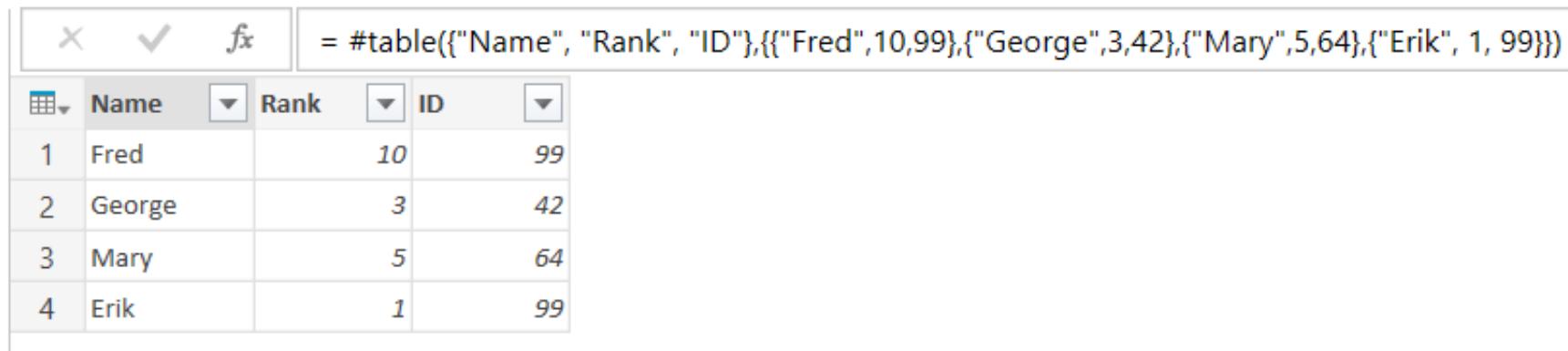
Accedi alla documentazione delle funzioni da Modifica query

- Digitare **=#shared** in Power Query formula bar
- Converti l'elenco in tabella
- Elenco ordinato per nome della funzione
- Clicca nello spazio bianco accanto alla funzioni per vedere il testo di help e gli esercizi
- Se si fa clic sulla parola "Funzione", si aggiunge un passaggio alla Power Query concentrandosi su quella funzione



Accedi alla documentazione delle funzioni da Modifica query

- `#table({ "title1", "title2"} , { { 0,1 }, { 1,0 }, { 0, 0 } })`



	Name	Rank	ID
1	Fred	10	99
2	George	3	42
3	Mary	5	64
4	Erik	1	99

Utilizzare per aggiungere un campo ID null al pull esistente che al momento non ha il valore



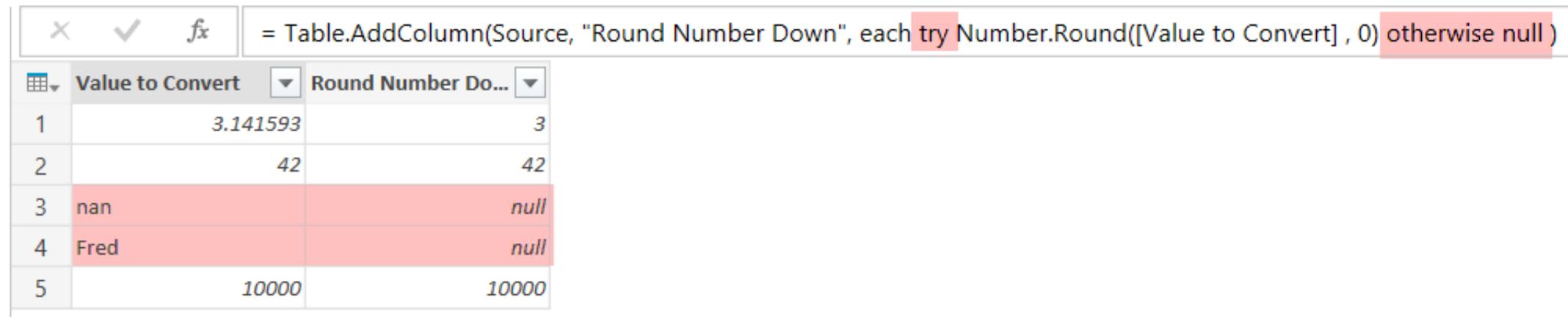
	ID	Name	Description
1	-1	na	null

Riepilogo dei tipi di oggetto

- **Formula Expression** – l'Espressione che viene calcolata per produrre un risultato. Il risultato può essere un **List**, **Record**, **Table** o singolo(scalare) **Value**
- **List** – Una sequenza ordinata di valori esempio {1, 2, 3}
- **Field** - - Un campo è una coppia nome / valore
- **Record** - Una serie di campi. Un campo è una coppia nome / valore in cui il nome è un valore di testo unico nel record del campo Per esempio: [CustomerID = 1, Name = "Bob"]
- **Operators** M supporta operatori comuni >, >=, <, <=, =, <>, or, and, not, +, -, *, /, &

try...otherwise

Esempio di gestione degli errori in una formula



The screenshot shows the Power BI M code editor with the following code:

```
= Table.AddColumn(Source, "Round Number Down", each try Number.Round([Value to Convert] , 0) otherwise null )
```

Below the code is a preview table with two columns: "Value to Convert" and "Round Number Do...". The data rows are:

	Value to Convert	Round Number Do...
1	3.141593	3
2	42	42
3	nan	null
4	Fred	null
5	10000	10000

Funzioni

Funzionalità avanzata! Parametrizzare una query, quindi chiamarla per ogni riga in un'altra query

getAccountID

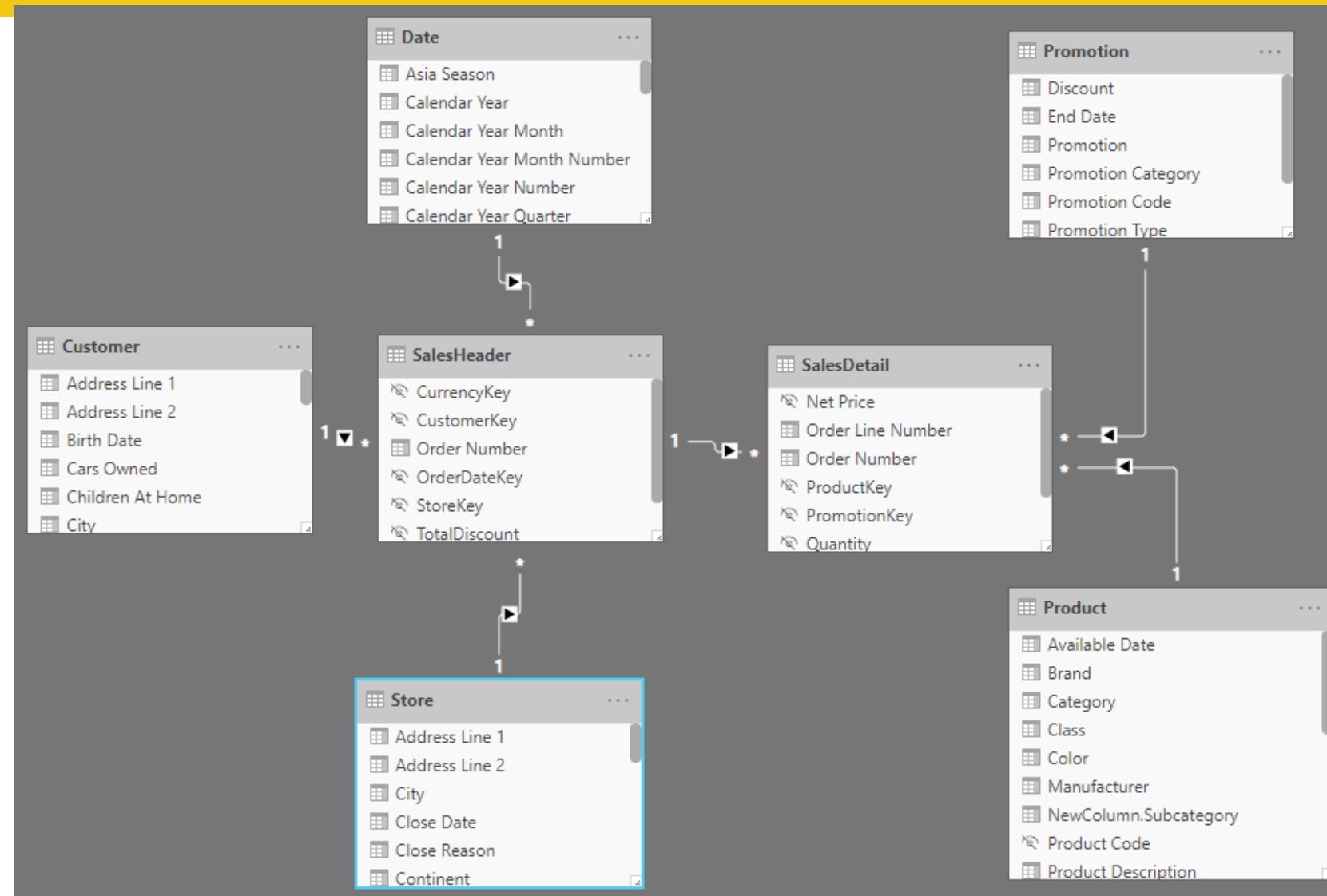
```
(Gamertag, NetworkID) => let
    Source = Json.Document(Web.Contents("http://www.bungie.net/platform/destiny/SearchDestinyPlayer/" & NetworkID & "/" & Gamertag),
    Response = Source[Response],
    Response1 = Response[{0}],
    #"Converted to Table" = Record.ToTable(Response1),
    #"Filtered Rows" = Table.SelectRows(#"Converted to Table", each ([Name] = "displayName" or [Name] = "membershipId")),
    #"Pivoted Column" = Table.Pivot(#"Filtered Rows", List.Distinct(#"Filtered Rows"[Name]), "Name", "Value"),
    #"Renamed Columns" = Table.RenameColumns(#"Pivoted Column".{{"displayName": "gamertag"}, {"membershipId": "accountID"}})
```

The screenshot shows the Power BI Data Editor interface. On the left, the 'Queries [13]' pane is open, displaying a tree structure with 'Gamertag List [1]' and 'functions [5]'. Under 'functions [5]', the 'getAccountID' function is selected and highlighted with a yellow background. In the center, the 'Enter Parameters' dialog is displayed, prompting for 'Gamertag (optional)' and 'NetworkID (optional)'. At the bottom right of the dialog is an 'Invoke' button. Below the dialog, the main workspace shows the 'Accounts' query editor. The query code is partially visible, starting with 'let' and defining variables for 'Source' and 'Response1'.

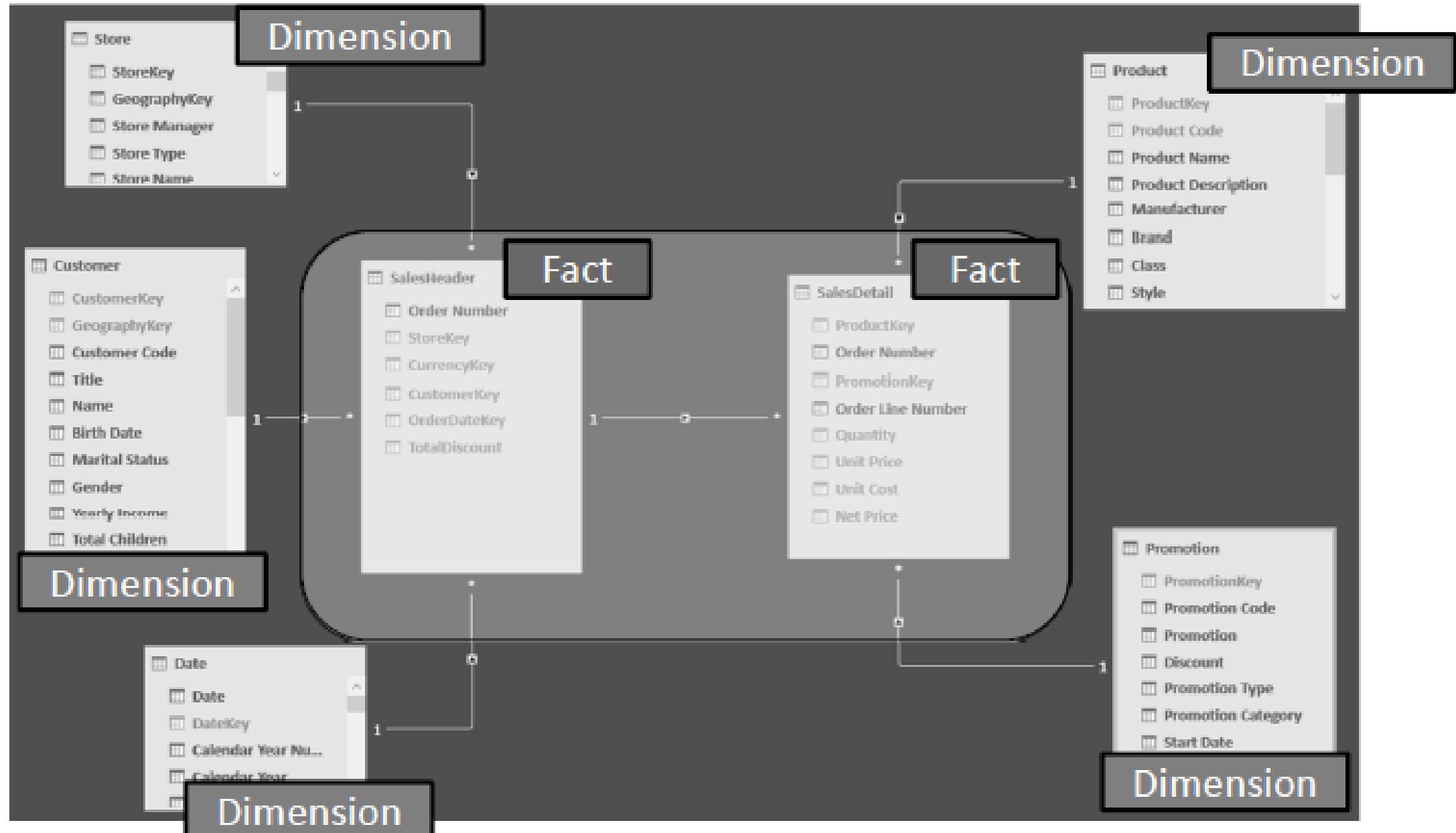
UC1: Caso reale analizzato

Caso Reale

- Analizziamo un caso reale di ordini e dettagli
 - Fatture / linee delle fatture
 - Ordini / Linee degli ordini
 - Team / individualità
- Il modello appare come due tabelle dei fatti unite tra di loro
- Collegare le due tabelle dei fatti aumenta la complessità e normalmente non è una buona idea



Caso Reale



Caso Reale: Problemi

- Essendo due tavole dei fatti, entrambe le tavole memorizzano le informazioni
- L'aggregazione dell'intestazione produce risultati errati, se interrogato da qualsiasi dimensione non collegata ad essa

Continent	CY 2007	CY 2008	CY 2009	Total	Color	CY 2007	CY 2008	CY 2009	Total
Asia	\$56,282.74	\$52,457.75	\$42,562.23	\$151,302.72	Azure	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29
Europe	\$52,153.03	\$25,881.80	\$32,700.45	\$110,735.27	Black	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29
North America	\$50,070.90	\$42,118.73	\$41,404.66	\$133,594.29	Blue	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29
Total	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29	Brown	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29
					Gold	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29
					Green	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29
					Grey	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29
					Orange	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29
					Pink	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29
					Total	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29

- Quale soluzione?

Caso Reale: Soluzione 1

- Applichiamo il filtro bidirezionale nel modello



Brand	CY 2007	CY 2008	CY 2009	Total
A. Datum	\$23,396.50	\$6,625.26	\$6,818.09	\$36,839.84
Adventure Works	\$31,014.00	\$11,358.74	\$10,899.31	\$53,272.05
Contoso	\$51,975.46	\$24,946.38	\$28,580.97	\$105,502.81
Fabrikam	\$14,426.09	\$22,265.30	\$17,478.13	\$54,169.52
Litware	\$13,876.12	\$17,539.56	\$14,796.98	\$46,212.66
Northwind Traders	\$21,013.75	\$3,754.67	\$5,361.97	\$30,130.39
Proseware	\$12,812.25	\$9,740.69	\$15,318.93	\$37,871.87
Southridge Video	\$23,171.83	\$4,380.71	\$4,788.64	\$32,341.19
Tailspin Toys	\$1,029.51	\$707.81	\$1,775.78	\$3,513.10
The Phone Company	\$4,027.90	\$7,067.17	\$6,330.28	\$17,425.35
Wide World Importers	\$7,125.39	\$19,366.80	\$14,494.74	\$40,986.93
Total	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29

SUM is 458,265.70

Caso Reale: Recap visto fino a qui

- **Valori nei dettagli**
 - Può essere tagliato da qualsiasi dimensione
 - Condurre a comportamenti standard e additivi
- **Valori nell'intestazione**
 - Può essere suddiviso per dimensioni collegate all'intestazione
 - Ma producono risultati errati con dimensioni di dettaglio
 - Sempre lo stesso numero
 - Valori non additivi (errati)

Caso Reale: Soluzione 2

- Calcolare lo sconto come una percentuale sulla testata dell'ordine

Order Number	StoreKey	CurrencyKey	CustomerKey	OrderDateKey	TotalDiscount	DiscountPct
20080604724008	307	1	13009	20080604	€ 0.54	10.00%
200805105CS561	307	1	19098	20080510	€ 13.99	10.00%
20070605820450	307	1	9431	20070605	€ 181.80	10.00%
20070510215734	307	1	4735	20070510	€ 32.90	10.00%
200801156CSS31	307	1	19092	20080115	€ 311.99	15.00%
200704013CS473	307	1	19082	20070401	€ 698.60	20.00%
20071115726159	307	1	15160	20071115	€ 1.80	15.00%
200905028CS712	307	1	19122	20090502	€ 97.80	10.00%
20070422714011	307	1	3012	20070422	€ 5.59	20.00%
200902218C9699	307	1	19116	20090221	€ 335.98	20.00%
20070213824152	307	1	13163	20070213	€ 307.35	20.00%
200902076CS697	307	1	19115	20090207	€ 10.30	20.00%
20071227722905	307	1	11906			
20080414822856	307	1	11857			

SalesHeader[DiscountPct] =

```

DIVIDE (
    SalesHeader[TotalDiscount],
    SUMX (
        RELATEDTABLE ( SalesDetail ),
        SalesDetail[Unit Price] * SalesDetail[Quantity]
    )
)
```

Caso Reale: Soluzione 2

- Per calcolare correttamente lo sconto, si aggregano le righe di dettaglio e - per ogni riga - si calcola lo sconto usando la percentuale memorizzata nell'intestazione

```
[DiscountValueCorrect] =  
SUMX (  
    SalesDetail,  
    RELATED ( SalesHeader[DiscountPct] ) * SalesDetail[Unit Price]  
    * SalesDetail[Quantity]  
)
```

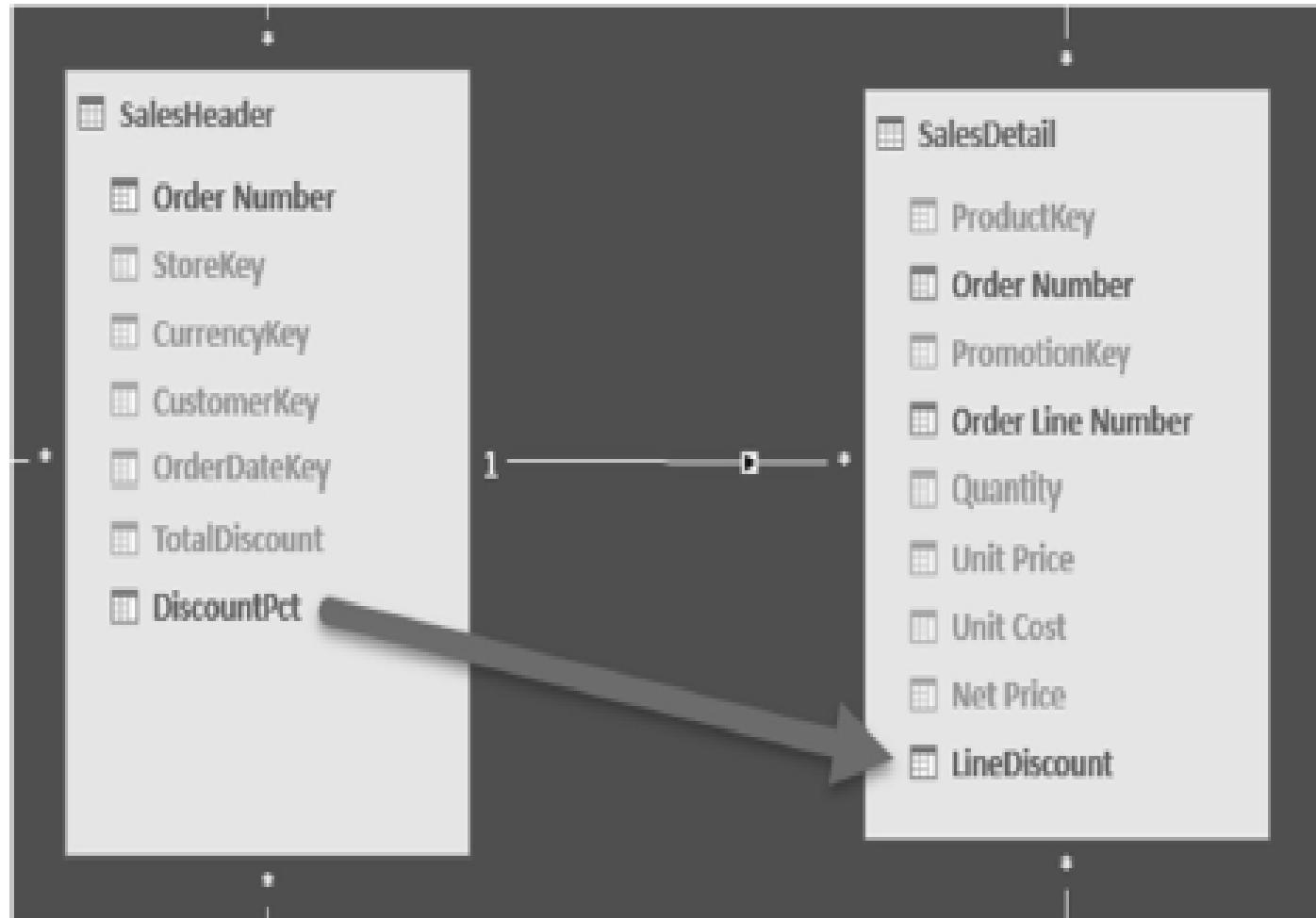
Brand	▲ DiscountValue	DiscountValueCorrect
A. Datum	\$36,839.84	\$26,489.32
Adventure Works	\$53,272.05	\$47,608.89
Contoso	\$105,502.81	\$89,994.41
Fabrikam	\$54,169.52	\$49,618.90
Litware	\$46,212.66	\$43,991.90
Northwind Traders	\$30,130.39	\$29,794.45
Proseware	\$37,871.87	\$34,436.21
Southridge Video	\$32,341.19	\$17,100.27
Tailspin Toys	\$3,513.10	\$3,513.10
The Phone Company	\$17,425.35	\$17,287.23
Wide World Importers	\$40,986.93	\$35,797.62
Total	\$395.632.29	\$395.632.29

La soluzione 2 è una soluzione
possibile ma bisogna fare un po' di
ginnastica con DAX

Il problema vero è che il modello non
è uno star schema!!!!

Caso Reale: Soluzione 3

- Lo sconto può essere memorizzato nella tabella dei dettagli
- Questo, trasforma la tabella SalesHeader in una dimensione



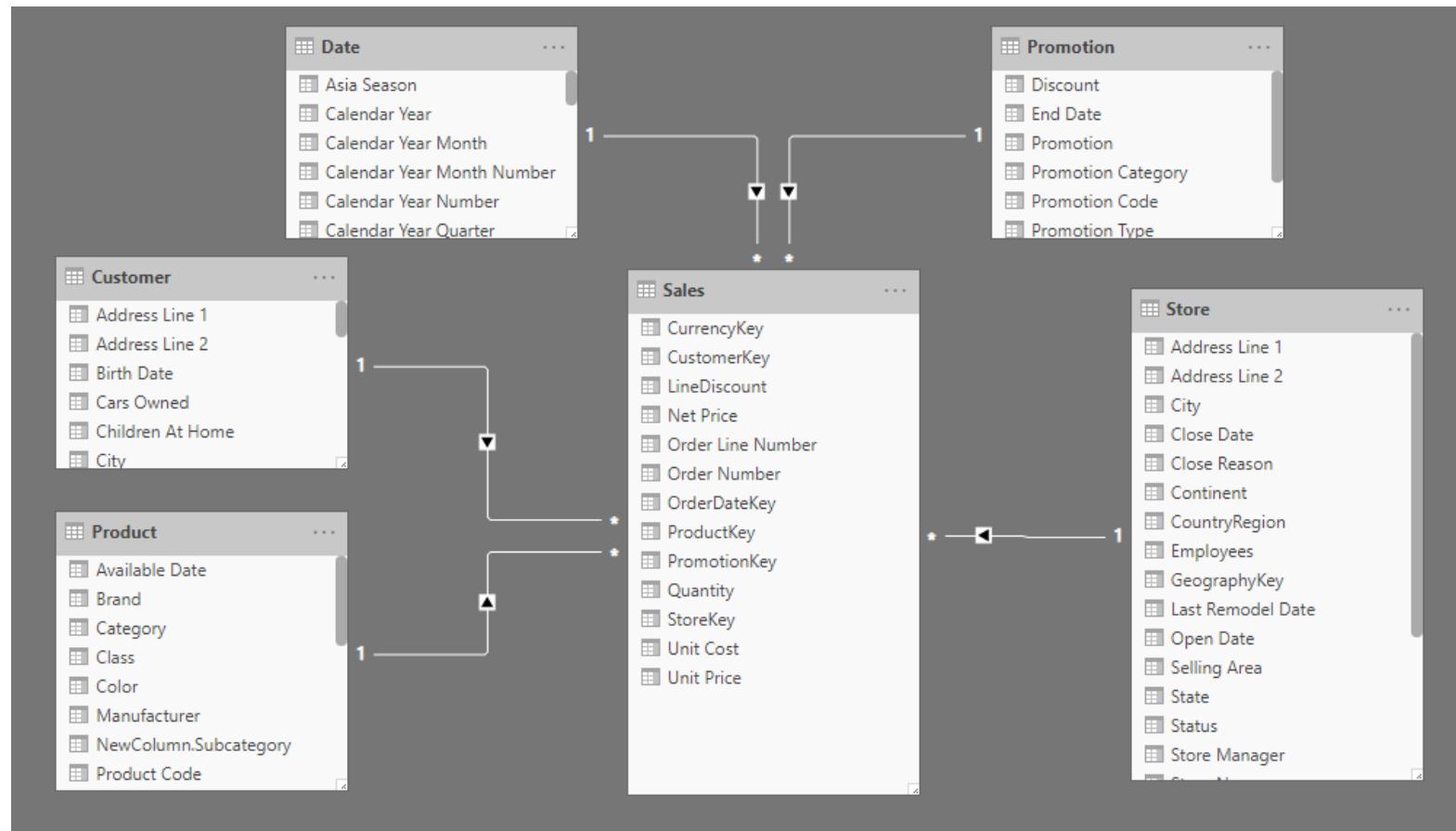
La soluzione 3 è migliore della soluzione 2 ma bisogna fare ancora del lavoro in DAX

Abbiamo ottenuto uno snowflake schema!!!

Vogliamo ottenere uno star schema dallo snowflake schema!!!

Una volta identificato la serie di fatti e dimensioni, tu non hai bisogno di una tabella di testata e una di dettaglio

Gli schemi a stella sono sempre la scelta migliore



Caso Reale: Recap

- Nell'esempio, abbiamo assegnato lo sconto come a una percentuale
- Pertanto, abbiamo utilizzato il prezzo come fattore di allocazione
- Altri scenari potrebbero essere diversi
 - Ad esempio: spedizione assegnata in base al peso
- Trovate il fattore di allocazione corretto
- Denormalizzare i valori nella tabella dei fatti
- La maggior parte dei modelli di testate / dettaglio può essere appiattita facilmente, cioè denormalizzare le colonne in una singola tabella dei fatti

Domande?