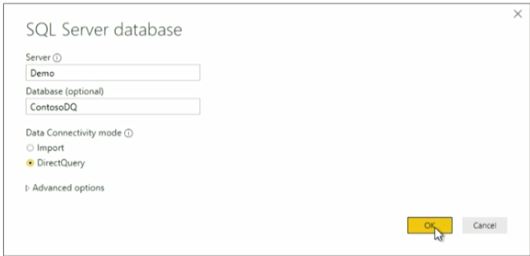
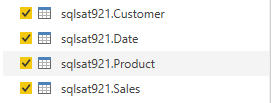
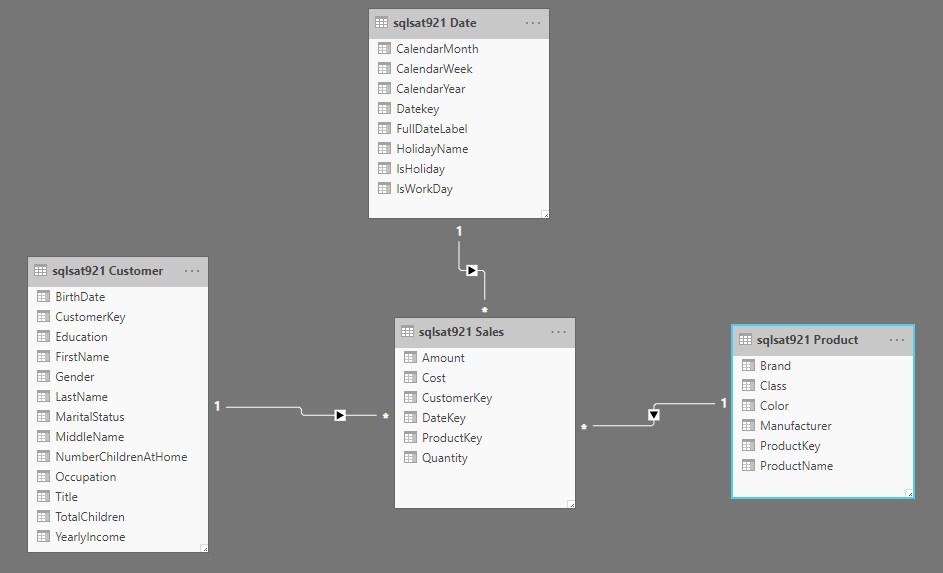
**--------DEMO 1: Creazione Aggregazioni**

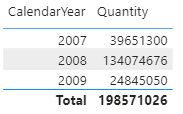
1. Apro **Power BI** vuoto
2. **Date**, **product**, **sales** e **Customer** preso da **SqlSat 921**
3. Carichiamo I dati in **direct query**



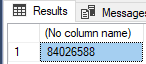
1. Selezionare le tabelle



1. Creiamo ora le relazioni
2. Creare una tabella con **year** e **quantity**



1. Nel renderizzare la tabella **risulta essere veloce perchè ho poche righe ma ricordiamoci che essendo in Direct query per ogni volta che noi trasciniamo o rimuoviamo campi dall’elemento visuale questo esegue una query verso SQL Server se abbiamo un database con tante righe** siamo fregati. Quindi come opzione abbiamo le **aggregazioni.**

 o synapse 

**(Mostrare con il profiler!!!!! che avvengono le interrogazioni)**

1. **AGGREGAZIONE** è una tabella che prepare con I valori pre aggregati. Quindi se devo interrogare il mio database per **Year** e **Quantity** ci prepariamo una tabella ragruppata per **Year** e **Quantity**.
   1. **In questo modo il mio motore invece di agregare I dati va a prendersi I dati già aggregati**
2. Quindi creiamo una tabella con I valori aggregati utilizzando una t-SQL che chiameremo “**SalesByYear**”

--Creazione della sqlsat921.SalesByYear

--DROP TABLE sqlsat921.SalesByYear

SELECT Date.[CalendarYear],

SUM(sales.quantity) AS SumOfQuantity

INTO sqlsat921.SalesByYear

FROM [sqlsat921].[Sales] AS sales

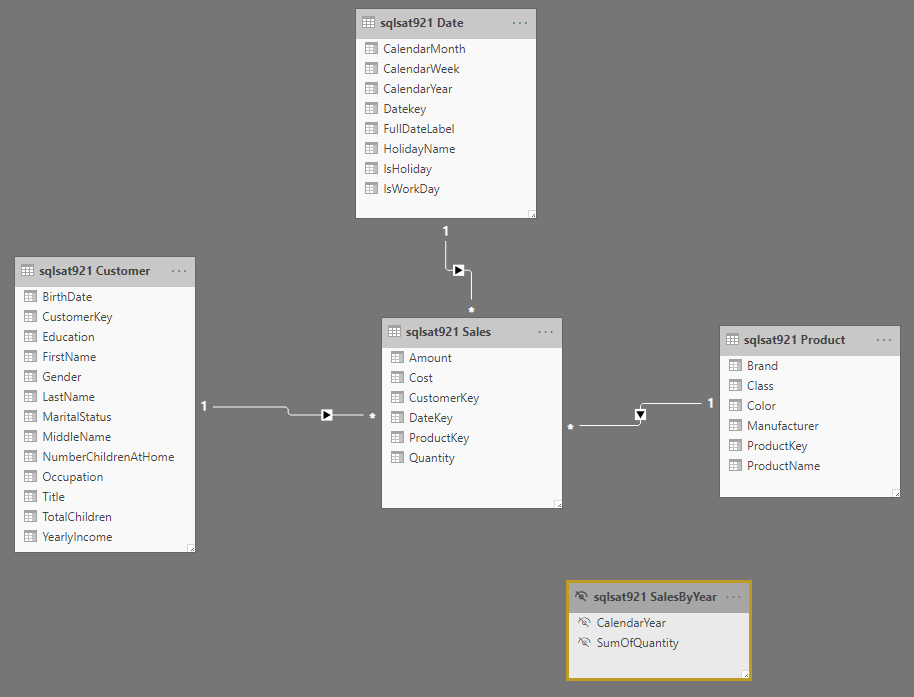
LEFT OUTER JOIN [sqlsat921].Date AS Date

ON sales.[DateKey] = Date.[DateKey]

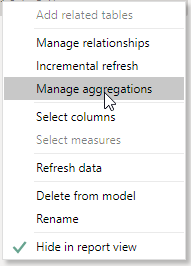
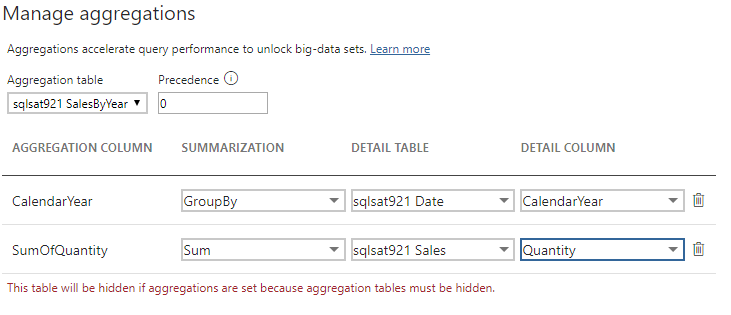
GROUP BY Date.[CalendarYear];

1. Ora dobbiamo **caricare** la nuova tabella sempre in **“Direct Query”**   
   

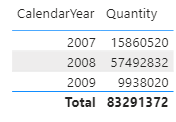
Vediamo che **la tabella è stata caricata nel modello**. Ma per ora questa tabella **rimane inutilizzata** dal modello

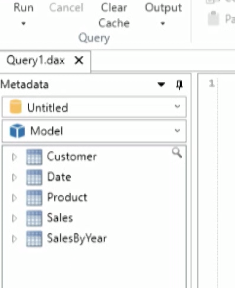


Quindi **dobbiamo specificare** che la tabella appena caricata **è una aggregazione**. Per fare questo ci serviamo del pannello delle tabelle laterali e selezioniamo con il tasto destro del menu che è una “**aggregations**”

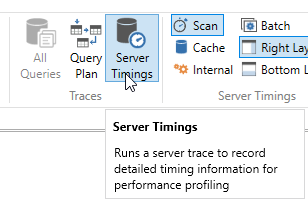
 

Quindi ora dobbiamo creare una tabella con “**Year**” e “**quantity**” in Power BI



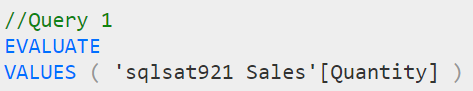
ATTENZIONE ORA Proviamo a capire se la **query colpisce (Hit)** o non colpisce l’aggregazione. Per verificare se questa query utilizza o meno l’aggregazione apriamo **DAX Studio**  


**Attivare il server Timings** e rilanciamo la query

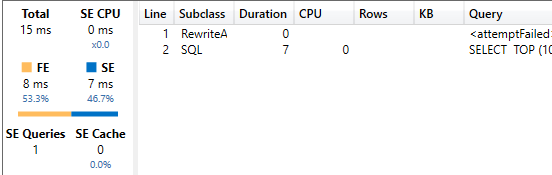
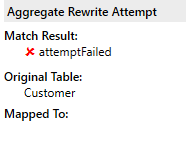


Usiamo **SqlSat921-Script Demo.dax per provare le query**

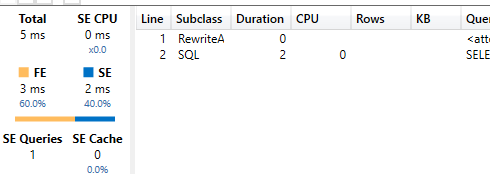
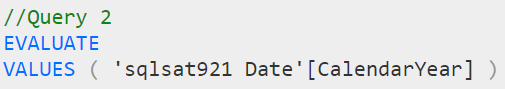
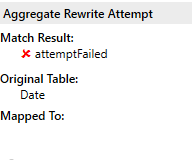
**🡪QUERY 1** Cancellare la cache ed query



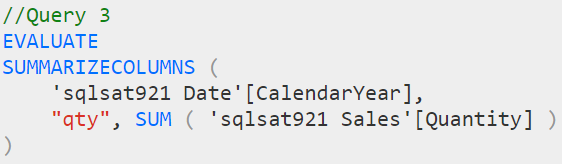
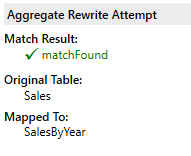
Importante oltre alla query SQL la riga “**RewriteA**” che sarebbe il tentativo che fa il motore di riscrivere la query se vi è una aggregazione che permette di eseguire la query in modo più veloce. **In questo caso non vi è stata una riscrittura della quey**

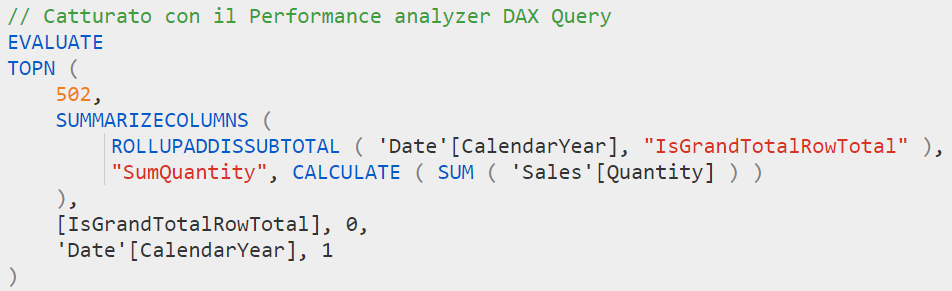
**🡪QUERY 2** Ora poviamo con una query diversa

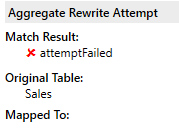
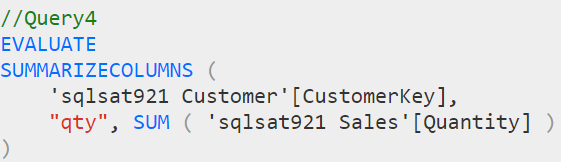
**🡪QUERY 3**: Ora scriviamo un’altra query **che matchia**

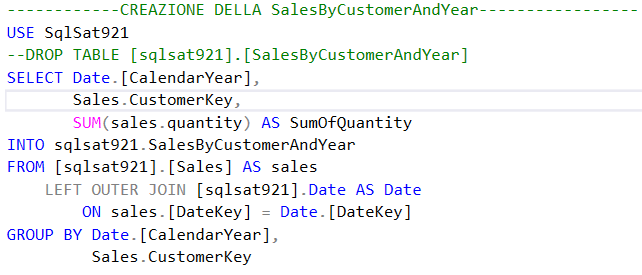
Oppure catturo la query con Dax Analyzer

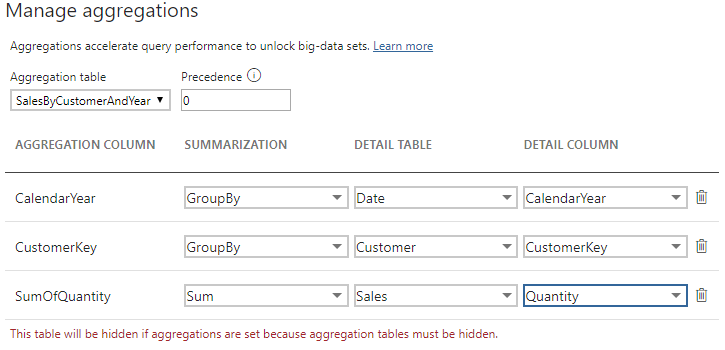


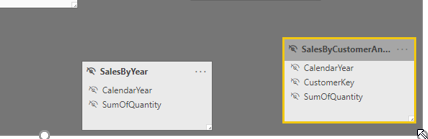
**🡪QUERY 4**: Ora ne eseguiamo un’altra di query **che non matchia**

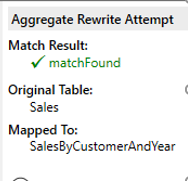
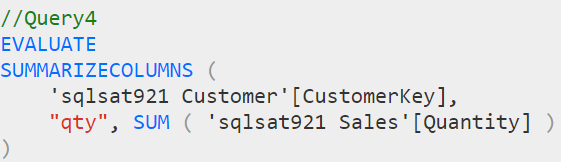


QUINDI CREIAMO UNA AGGREGAZIONE SULLA sul customer

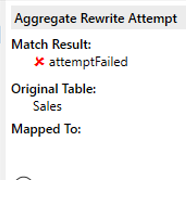
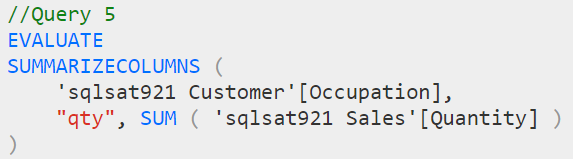


Importiamo ora la tabella appena create  




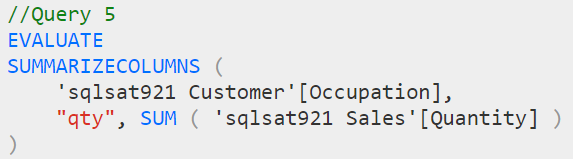
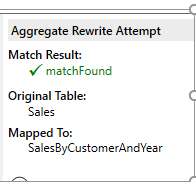


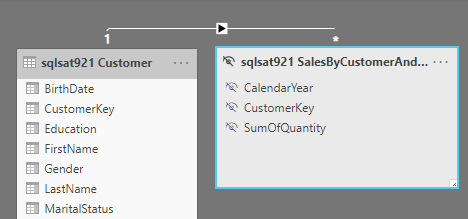
**🡪QUERY 5**: Ora ne eseguiamo un’altra di query  **31 92 96**

[**https://www.menti.com/pi3xjo1qf3**](https://www.menti.com/pi3xjo1qf3) **(la query 5 colpirà l’aggregazione)**

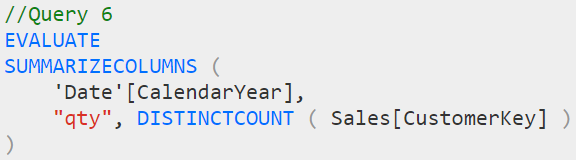
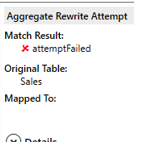
**PERCHE’?** il campo **Occupation** non c’è nell’aggregazione**. R**icordiamoci che la tabella **SalesByCustomerAndYear NON è relazionata** al customer

**SOLUZIONE:** Per fare in modo che tutte le colonne di Customer siano prese in considerazione devo **CREARE una relazione** come **dall’aggregazione alla tabella dei customer**

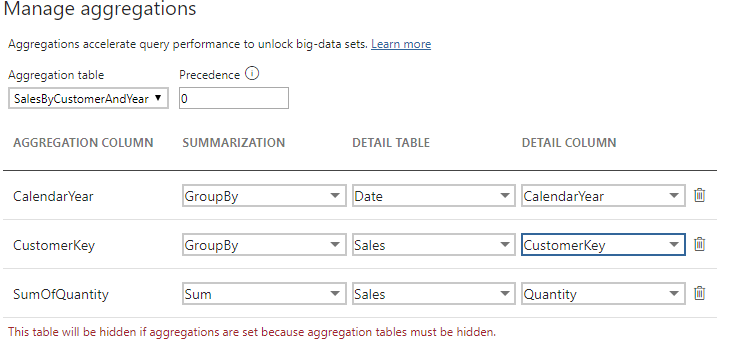
 ****

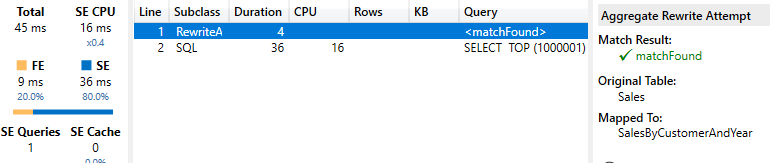
****

**🡪LA DOMANDA è:**  **31 92 96** [**https://www.menti.com/pi3xjo1qf3**](https://www.menti.com/pi3xjo1qf3) **(la query 6 colpirà l’aggregazione)**

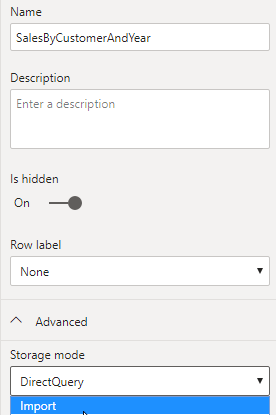
**PERCHE’?** Anche in questo caso abbiamo chiesto di agregare per un campo che non è nella aggregazione infatti è creata su **Customer[CustomerKey] e non su Sales[CustomerKey] .**Quindi dobbiamo cambiare l’aggregazione 😊

****



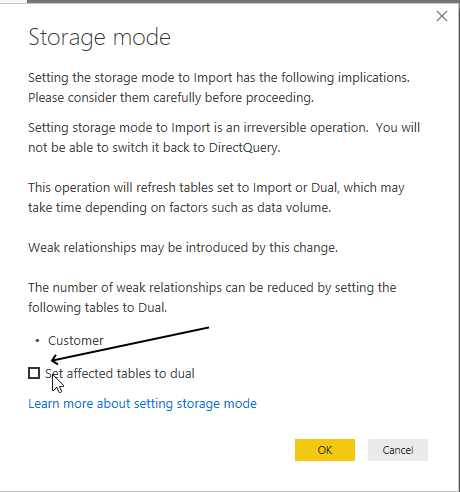
**--- DEMO 2: Weak relationship e concetto di DUAL**

**🡪Ora vogliamo portare la tabella aggregata direttamente in memoria** così è molto veloce. Quindi portiamo la **SalesByCustomerAndYear** in memoria. Cliccando a lato sul combo “**DirectQuery**”



Viene fuori un messaggio che ci dice che:

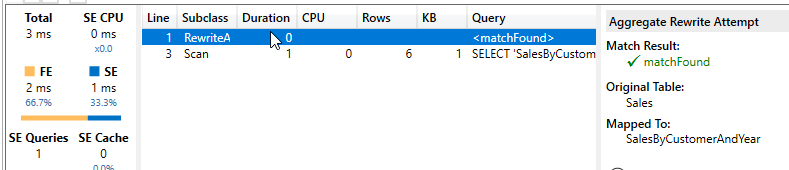
* non saremmo più in grado una volta portata in memoria di ritornare indietro
* verrà introdotta una weak relationship con la modifica

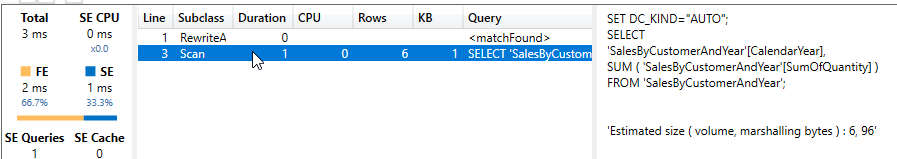
Inoltre ci suggerisce che se vogliamo ridurre le weak relationship dobbiamo settare come **Dual** la tabella **customer** con un flag**. Ma noi non clicchiamo questo flag lo faremo dopo maualemente eventualmente.  
**

**🡪QUERY 7:** Ora eseguiamo questa query

EVALUATE   
SUMMARIZECOLUMNS('Date'[CalendarYear],  
                 "qty",SUM(Sales[Quantity])  
                 )

Vediamo se la query trova una corrispondenza su una aggregazione

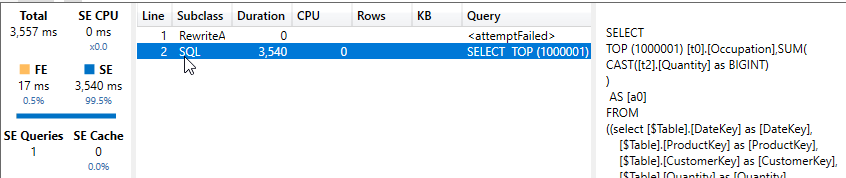


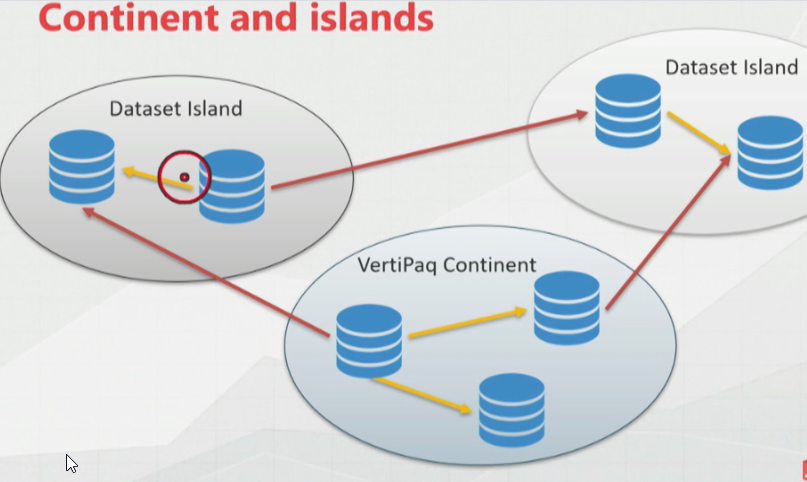
Ha trovato un match su **SalesByCustomerAndYear** ma la query eseguita non è più su SQL ma effettua una scan su Vertipaq perchè ora è in memoria

**🡪QUERY 8:** Ora eseguiamo questa query va sull’aggregazione?

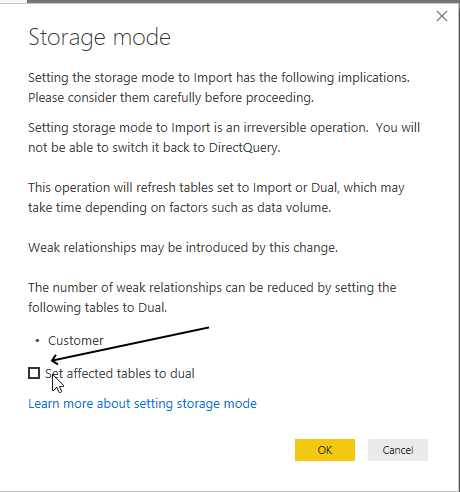
EVALUATE   
SUMMARIZECOLUMNS('customer'[Occupation],  
                 "qty",SUM(Sales[Quantity])  
                 )

**🡪LA DOMANDA è:**  **31 92 96** [**https://www.menti.com/pi3xjo1qf3**](https://www.menti.com/pi3xjo1qf3) **La query va su SQL anche se l’aggregazione è in memory e non fa la hit sulla aggregazione ☹ a differenza della demo 1 (query 5) perchè?**

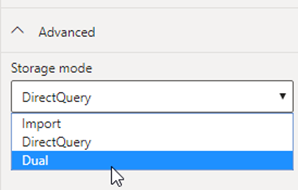


PERCHE’? Perchè l’engine non è in grado di eseguire le query parte in memory e parte in direct Query 😊😊😊 (weak relationship: il motore non può costruire dei join sul tabelle che si trovano sui isole separate)  


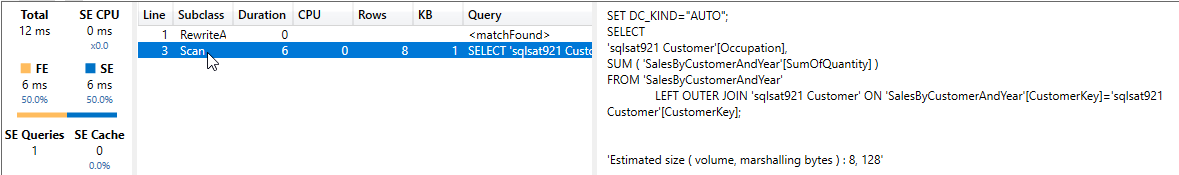
Ecco cosa era il warning che ci è apparso prima e che abbiamo levato il flag

****

Per trasformare una **weak relationship** in una **strong relationship** si dovrebbe portare la customer in memory ma avremmo lo stesso problema poi sulla relazioni della sales che diventerebbe weak. Allora usiamo l’opzione **DUAL**



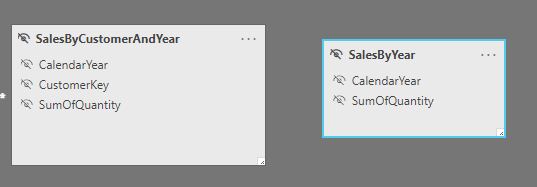
Se ora rieseguiamo la query vediamo che va in Vertipaq e NON in SQL e fa la HIT sull’aggregazione 😊

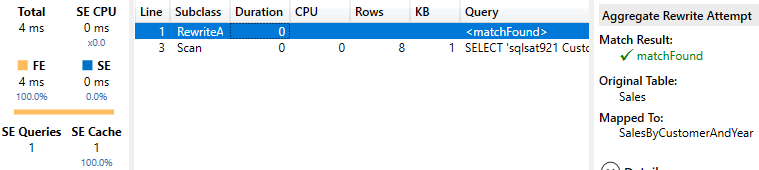


**--- DEMO 3: Precedenze nelle aggregazioni**

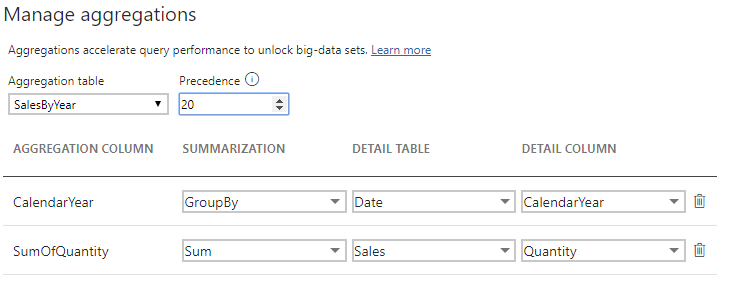
**🡪QUERY 9:** Ora eseguiamo questa query quale aggregazione colpisce (**SalesByCustomerAndYear – in memoria**) o (**SalesByYear in sql** )?

EVALUATE   
SUMMARIZECOLUMNS('Date'[CalendarYear],  
                 "qty",SUM(Sales[Quantity])  
                 )

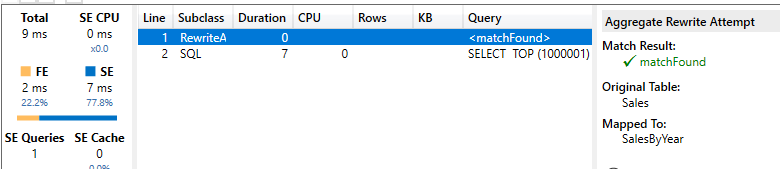
**Per L’engine NON c’è differenza se una aggregazione è in memoria o è in Direct Query mode. L’engine lascia a noi definire quale delle due ha la priorità!!!! Usnado la Precendece ( il più alto vince) se lasciamo fare al motore come in questo caso scegliere l’ultima inserita cioè SalesByCustomerAndYear**

****

Se noi impostiamo la **precedenza a 20 per la SalesByYear**



Vedremo che se eseguiamo la query l’aggregazione che becca sarà quella di SaleByYear



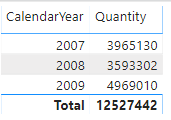
**---DEMO 4: Mantenimento delle aggregazioni**

Cosa succede se nella tabella delle vendite “Sales” qualcosa nella sorgente dati cambia (CRUD)?

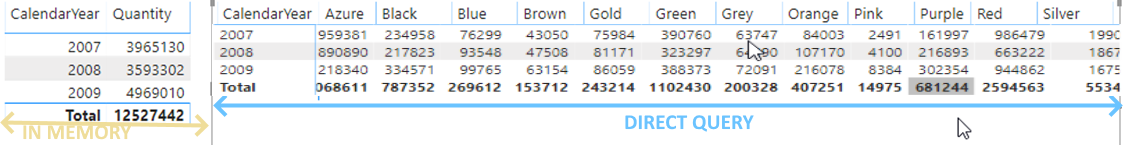
I dati delle Sales sono aggiornate ma I dati delle tabelle importate no ☹ Quindi ci potrebbero essere delle inconsistenze nelle query che eseguiamo. Infatti se la query va sull’agregazione non avremmo tutte le informazioni aggiornate mentre se la query va nella tabella delle vendite allora avremmo quelle più aggiornate

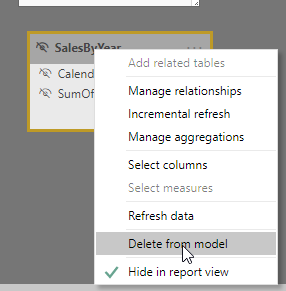
**Sta a noi mantenere aggiornate le informazioni 😊!!!!**

1. Creiamo una griglia in power BI come la seguente:



1. La griglia sopra colpisce una aggregazione quando interroga i dati? **Si**
2. **Ora proviamo a trascinare nelle colonne il color del product che non dovrebbe colpire nessuna aggregazione**. Inoltre **rimuoviamo** anche l’aggregazione **SalesByYear**

****



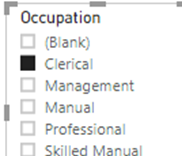
1. Cancelliamo le righe dalla tabella Sales usando la query seguente:

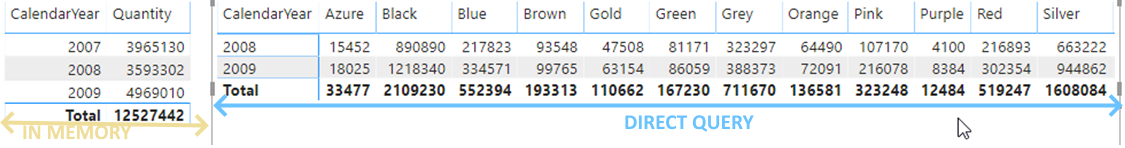
**DELETE FROM SALES Where YEAR(Sales.[Datekey]) = 2007**

**or in synapse**

ALTER TABLE **Sales** SWITCH PARTITION **2** TO **Sales\_20070101** PARTITION **2;**

1. Metto uno slicer nel report e filtro per occupation = “Clerical” **sparisce un anno**



****