

QUICK START MANUAL



Copyright 2020 - PARIS Technologies Inc. All Rights Reserved.





QUICK START MANUAL



Information in this document and in documents associated with other PARIS Technologies Training materials is subject to change without notice, and does not represent a commitment on the part of PARIS Technologies. No part of this manual may be reproduced or transmitted in any form, or by any means whatsoever, without the written permission of PARIS Technologies International, Inc.

© Copyright 2020 PARIS Technologies International, Inc.

Windows is a trademark of the Microsoft Corporation.

Microsoft Excel is a trademark of the Microsoft Development Corporation.



This page has been left blank intentionally.

Table of Contents

1.	Intro	duction to OLATION [®]	1
2.	Dem	o Overview	1
3.	Crea	te a New OLATION [®] Database	2
4.	Crea	te the SALES Cube	4
		Create Dimensions from Relational Tables	5 .14 .15 .17 .19 .21 .22 .23 .24
5.	View	ring the OLATION Data	26
	5.1. 5.2.	Create the Dynamic Excel Front End—PowerExcel Slice	.34 .35 .37 .40 .41
6.	OLA	TION [®] and Real-Time Application Results	46
		Pull Data Updates from the Relational Source down to the OLATION Database Write-back (for Planning) to the Source Relational Database	
7.	Defir	ne Formula in OLATION [®]	53
		Define the Cube Formula Statement Checking the Cube Formula Results	
8.	Crea	te the PRODUCT_SALES Cube	60
		Pre-Work: Prepare the Product Factdata SQL Table	.60 .62 .63 .65 .66 .67 .68

	8.2.6.Save the Dimension	69
8.3.	Create the PRODUCT_SALES Cube	71
	8.3.1.Assign the Measure Member	73
	8.3.2.Arrange the Dimension Order	74
	8.3.3.Define Relationships	74
	8.3.4.Define Cube Setting: Disable READ-ONLY Setting	75
	8.3.5.Save the Cube	76
8.4.	View the PRODUCT_SALES Cube	77
8.5.	Create a Cross-Cube Formula for the PRODUCT SALES Cube	79
	8.5.1.Viewing the Results of the Cross-Cube Formula	82

OLATION[®] Quick Start Manual

1. Introduction to OLATION[®]

The Olation[®] Quick Start Manual is designed to introduce users to the fundamentals of the Olation[®] application by using step-by-step procedures to create an Olation[®] model. PARIS Technologies recommends that new Olation[®] users familiarize themselves with Introduction to Olation section of the Olation User Manual as a prerequisite to this manual. This Quick Start manual also assumes that that product is installed, with prerequisites in place to use the product.

By working through this Quick Start Manual, you will learn about the elemental functions of the Olation application: how to create an Olation database; how to create Dimensions from relational tables as well as create a multidimensional type of Dimension; how to define Dimension attributes; how to create or edit the Dimension hierarchy; how to create a Cube and define Cube attributes, and; how to create simple internal Cube and cross-cube formulas. Note that—especially if you are an Administrator, you will not only build cubes (business models) by the steps described here but will undoubtedly need to understand the concepts and follow the steps covered in these pages because they convey how to use, customize and advance your models. These skills will enable you to vastly increase the potential business uses and benefits of the product at your organization.

2. Overview

The exercises in this Module serve as a quick tutorial on how to build simple Olation Models which will cover topics such as:

- Basic Modeling in Olation*
- Defining Dimension and Cube Attributes
- Creating simple Cube and Cross-Cube Formulas
- Dynamic Writeback Capability of an Olation Model

*Note: the underlying SQL database *Using_Olation* is utilized in this manual, so it will be necessary to install/access it in order to follow the step-by-step exercises. Furthermore: this database has been organized in such a way as to quickly enable you to build a cube from relational tables. In a production system, you will likely need to reorder/remake tables with a view in mind of the cube(s) you intend to create in Olation. As well, this database does not feature indexed tables, which you will more likely encounter in a real-world scenario.

3. Create a New OLATION[®] Database

This section is concerned with creating an Olation database from a source relational database. The Olation database and its components will be built from logic in the source database and will demonstrate the dynamic connection between Olation and the source database.

IMPORTANT: For this exercise, we will create an Olation database hosted on an Olation Server working with a local instance of SQL Server. We will be using the **USING_OLATION** SQL database to build our database and its corresponding Dimensions, Cubes and attributes.

To create a new OLATION® Database:

- 1. Launch OLATION[®] Studio.
- 2. In the Database Explorer, expand **Servers**, then right-click on **localhost** (this will be our preferred Olation Server for this demo) and select **Connect**, to establish a connection to the Server.

Note: If you wish to connect to a different Olation Server, you will be required to register that Olation Server.

To register a server: go to Database Explorer \rightarrow right-click on Servers \rightarrow select Register Server.

3. Next, right-click on the preferred Olation Server (**localhost**) and select **New Database** from the options.



Detekson Trees	01 / VO	
Database Type:	Olation for MS	SQL Server®
Server:	(local)	
Jeivei.		
	Connect to	SQL Server using NT Authenticatio
SQL Admin User:		
- ·		
Password:		
Database Name:		
Source Database	э	
Display Name		
Secure Database	-	False
	- nal Source	False False False

This displays the New Olation Database dialog box.

4. Select a **Database Type** from the drop-down list, i.e., **Olation for MS SQL Server**[®]. (as shown above).

Cancel

5. Specify a SQL Server—in this case, (local) for localhost is being used.

0K

- Enable the Connect to SQL Server using NT Authentication checkbox option. Alternatively, you can disable the option and enter instead a <SQL Admin User> and <Password>.
- 7. Click on the **Database Name** drop-down and select the preferred source SQL database for the Olation model you are about to create. Following this example, select **USING_OLATION**.
- In the Display Name field, type the <name of new Olation Database>.
 Here is where you specify the preferred name for your Olation database. By default, this field will take the name of the selected Database. For our demo keep the default name, i.e., USING_OLATION.
- There are 3 additional settings (Secure Database; Live from Relational Source; Save database to local file) that you can configure for your Olation database. For now, leave the default settings. [NOTE: additional Olation features and capabilities not covered in this Quick Start manual can be found in the Olation User Manual or the product Help file.]
- 10. Click **OK**.

This will create and initialize an Olation database. The new Olation database will appear in the Olation's Database Explorer, as in the following image.

OLATI**O**N



Note that the Tables shown are the same as those that can be seen in SQL Server (*dim_Accounts*, *dim_Actual_vs_Budget*, *dim_Months*, *dim_Regions* and *Factdata*). It is worth noting that Views and Queries from SQL Server, and External Data, can be reached as well.

4. Create the SALES Cube

Now, with the database is created, we can begin building Dimensions, Dimension components, and the Cube itself, all built from these relational tables.

Important: Please note that specific images and figures may be different from the data set you are working on. This exercise is meant to serve as a guide on the steps to be followed to help you understand how to use OLATION but still allow you to work with your own data set.

For this section, the objective is to create a SALES Cube that will be composed of the Dimensions *Account*, *Version, Month* and *Region*. We will also write a simple cube formula to handle basic calculations across specific intersections within the Cube. Following that, we will create a view out of this Olation Cube and

demonstrate the writeback capability between a front-end client (e.g., PowerExcel or PowerOLAP) and the source relational database (in this example the MS SQL Database).

4.1. Create Dimensions from Relational Tables

Dimensions are lists of related terms used to organize data and are used to construct Cubes, the modeling structures used for planning, analytics and reporting on data.

The first step is to build Dimensions. One of the significant features of OLATION[®] is that it permits a user to create a Dimension using common relational database components: tables, queries and views. From relational tables, we define the component Members and configure other attributes for Dimensions. Again, as mentioned earlier, we will be creating four Dimensions from the existing SQL tables: **Account**, **Version**, **Month** and **Region**.

4.1.1. Create the Account Dimension

- 1. Go to the Database Explorer pane (left section of the Olation window), expand on **Tables** and then expand on **dbo**. Select the table '**dim_Accounts**'; right-click and select **New Dimension**.
- In the Create a New Dimension dialog box, type a <name for your Dimension>. You can give the Dimension any name you want. For this exercise, keep the name Account for the Dimension.

Create New	Dimension	
New Dimer	nsion	
Name:	Account	
	Create	Cancel

3. Click Create.

This opens the Table window on the right. This also opens the **Table Tab** along the OLATION ribbon.

OLATI**O**N

		Olation			- 🗆 X
File Home Relational Multi-Dimensional	Table				Style 👻 🥅 MAX 🔇
82 20					
Save Refresh New Delete					
Table Dimension					
atabase Explorer 🛛 📮	K / 🔯 dim_Accounts				• :
Servers	Design Data				
Iocalhost USING OLATION	Column Name	Data Type	Allow Nulls	2↓	
L Dimensions	► Account	nvarchar		✓ Database Column	
© Cubes	Parent	nvarchar		Catalog Name	USING_OLATION
Tables	Weighting	int		Schema Name	dbo
🖃 🎹 dbo	Ordering	int		Table Name Column Name	dim_Accounts Account
dim Accounts		Inc		Allow Nulls	True
dim_Actual_vs_Budget	•			Identitifier Column	False
iii dim_Months				Catalog Name	
III dim_Regions					
III Factdata					
1 III Views					
D Queries	Account				
🗉 🗹 External Data	2↓ 🖻				
Bookmarks	✓ Common				
	Dimension Name			Account	
Local Databases	Display Name				
	Dimension Type All Hierarchy			Standard True	
	Data Source Type			Relational Table	
	Hierarchy Type			Columnar	
	> Database Source			USING_OLATION.dbo.dim_Acc	ounts
	Total Aggregate			False	
	Dimension Name Display name for this dimen	ision			
	biopidy fidmo for this dimor				

The Table Tab (per above image) includes definitions of the SLQ table being accessed—top left (Design/Data tabs), top right (Database Column), and properties of the Account dimension (bottom half, Account tab). Using these in concert will enable you to build the Dimension according to desired specifications.

About Dimension Members

Dimensions are composed of Detail and Aggregate Member types. Detail Members "add up" to Aggregate Members. Aggregate Members are "parent" Members when they aggregate a group of "child" Members. For example, you may create a Dimension named *Month*, whose *First Quarter* member can be an Aggregate Member of *January, February*, and *March*, each of them a Detail Member. Note that child Members can also themselves be an Aggregate.

We will next add Dimension Members to the Account dimension, and create a logical Aggregate.

- 1. In the **Design Tab**, select the top row, where 'Account' appears under Column Name.
- 2. Right-click and then select **Set as Dimension Member**, as shown below.

	lim_Accounts						• ×
Design	Data						
	Column Name	Data Type	Allow Nulls		i 2↓ 🖾		
•	Account	Set As Dime	ension Unique ID		/ Database Column		
	Parent	Set As Dimension Unique			Catalog Name	USING_OLATION	
		Set As Dime	ension Member		Schema Name dbo		
	Weighting	Create As D	imension Alias		Table Name	dim_Accounts	
	Ordering	Create As D	Dimension Property		Column Name	Account	
					Allow Nulls	True	
•		Create As D	imension Subset		Identitifier Column	False	
		Set As Hiera	archy Weight		atalog Name		
		Define As H	lierarchy Level	→			

Define Hierarchy

Parent-child relationships exist in the underlying *dim_Accounts* table, ensuring that an accurate calculation occurs when *Sales* and *Cost of Sales* "hierarchize" as *Margin*. To create this hierarchy:

- 1. In the **Design Tab**, select the row corresponding with the Column Name 'Parent'.
- 2. Right-click and then select **Define As Hierarchy Level**, then select **At Root**. (At Root means that this will be the highest-level Aggregate Member.)

/ 👿 d	lim_Accounts					• >
Design	Data					
	Column Name	Data Type	Allow Nulls		2↓ 🖾	
	Account	nvarchar	\checkmark		✓ Database Coli	umn
•	Parent				Catalog Name	USING_OLATION
	Set As Dimension Unique ID				Schema Name	dbo
	Weighting Set As Dimension Member			Table Name	dim_Accounts	
	Ordering	Create As Dime	Dimension Alias		Column Name	Parent
			ension Allas		Allow Nulls	True
÷		Create As Dime	ension Property		Identitifier Colum	n False
		Create As Dime	ension Subset		Catalog Name	
		Set As Hierarch	iy Weight			
		Define As Hiera	archy Level	•	At Root	

Define Hierarchy Weight

1. First, go to the **Data Tab** (circled in the following image): note that in the Cost of Sales has a "-1" Weighting (see arrow).

Desig	Data				
	Account	Parent	Weighting	Ordering	
•	Margin Pont				
	Sales	Margin		1	
	Cost of Sales	Margin		-1	
	Margin				

To ensure that *Cost of Sales* (as a negative value) and *Sales* (as a positive value) calculate correctly:

- 2. Back in the Design Tab, select the row with Column Name 'Weighting'.
- 3. Right-click and then select Set as Hierarchy Weight (see next image).

/ 👿 d	lim_Accounts							• ×
Design	Data							
	Column Name	Data Type	Allow Nulls			≵ ↓ 💼		
	Account	nvarchar	\checkmark		~	Database Column		
	Parent	nvarchar				Catalog Name	USING_OLATION	
	And a labor and a second					Schema Name	dbo	
•	Weighting	Set As Dimensi	Set As Dimension Unique ID			Table Name	dim_Accounts	
	Ordering	Set As Dimensi	ian Marahar			Column Name	Weighting	
		Set As Dimensi	ion Member			Allow Nulls	True	
-		Create As Dime	ension Alias			Identitifier Column	False	
		Create As Dime	ension Property		Cal	talog Name		
		Create As Dime	ension Subset					
		Set As Hierarch	ny Weight					
		Define As Hiera	archy Level	•				

Note the bottom half of the "dim_Accounts" pane: here are selections for various attributes or properties that you can choose to further define the Dimension.

The first consideration we will give to this Dimension is defining its hierarchy (or hierarchies).

The next 2 selections will explain further about Hierarchies and Aggregates—specifically as they relate to the Dimension property selections that can be made in the bottom half of the right-hand pane (in this case, the *Account* dimension):

Enable/Disable the All Hierarchy option

- 1. At the bottom half of the window of the Account Tab, scroll down to see the **All Hierarchy** option.
- Click the drop-down button that appears on the far right and ensure that False is selected.
 Note: Alternatively, you can double-click on the All Hierarchy field to change selections from *True* to *False* and vice versa.

The 'All Hierarchy' option: A selection of **True** creates a Member at the root called "All", adding every Detail Member under it. (For this example, there is no logical need for an *All* Dimension member.)

Design	lim_Account				• 2
*	Column Naie Account Parent Weighting Ordering	Data Type nvarchar int int	Allow Nulls	Image: Second system USING_OLATION Catalog Name USING_OLATION Schema Name dbo Table Name dim_Accounts Column Name Account Catalog Name Image: Second system Catalog Name Image: Second system	
Co Din Dis Din All Da				Account Standard True True False	v

Enable/Disable the Total Aggregate option

- 1. At the bottom half of the window scroll down to see the **Total Aggregate** option.
- Click the drop-down button that appears on the far right and select False.
 Note: Alternatively, you can double-click on the Total Aggregate field to the change selection from *True* to *False*.

Note: The 'Total Aggregate' option: A selection of **True** creates a Total Member(s) that sits at the top of each defined Hierarchy. (For this example, there is no logical need for such a calculation.)

/ 🗟 a	lim_Accounts						• ×		
Design	Data								
	Column Name	Data Type	Allow Nulls		ê 2↓ 🖻				
•	Account	nvarchar			👻 Database Column		^		
	Parent	nvarchar			Catalog Name	USING_OLATION			
	Weighting	int	\checkmark		Schema Name Table Name	dbo dim_Accounts			
	Ordering	int			<u></u>		×		
					Catalog Name				
				_					
Accoun	t								
A Z									
	mmon						^		
_	nension Name			1	Account				
	play Name nension Type				Standard				
	Hierarchy				False				
Da	ta Source Type				Relational Table				
	rarchy Type				Columnar				
	tabase Source al Aggregate				ISING_OLATION.dbo.dim_Accounts False		\sim		
Filt				_	True				
	t Or <mark>son</mark>				alse				
	ate Missing Member Aggregate				ате				
	rue to automatically o	reate a total aggreg	ate member						
Schol	no lo automatical ju								

Defining a Sort Order

You can define the order of the Members as they appear listed within a Dimension—for example, you may want them arranged in an ascending or descending order; or you may also arrange the list based on a sequence defined within a reference column.

To define the sort order:

1. At the bottom half of the Table window, select the **Sort Order** option. An ellipsis button will appear on the far right (see next image).

2. Click on that **ellipsis button** to bring up the **Edit Sort Order** dialog box (next page, second image).

	✓ Text dim_Accounts → X								
Desigi	Data								
	Column Name	Data Type	Allow Nulls		2↓ 📼				
•	Account	nvarchar	\checkmark		Catalog Name	USING_OLATION	^		
	Parent	nvarchar			Schema Name	dbo	_		
	Weighting	int	\checkmark		Table Name Column Name	dim_Accounts Account			
	Ordering	int			10 61 0	-	×		
					Catalog Name				
Accou	int			_					
A	ll Hierarchy				False		^		
	ata Source Type				Relational Table				
	ierarchy Type atabase Source				Columnar USING_OLATION.dbo.dim_Accounts				
	otal Aggregate				False				
	lter								
	ort Order reate Missing Membe	r			False				
	Order								
	It ordering from relatio	nal source							

3. In the Edit Sort Order dialog, click the Measure Columns icon and select Ordering.

	Edit Sort Order				?	\times
	<u>∎</u> ∙ ± * ÷÷*					
	Account					~
	Parent					
	Weighting					
	Ordering	2				
						× .
ŀ	<					>
			OK	Cancel		

4. Click **OK** to close the dialog (which appears below).

?	Х
	-
	-
	?

Enable/Disable Create Missing Member

The Create Missing Member option allows you to create a default Member in which values are placed for "missing" Dimension members (i.e., transaction records that have no Member indicated in transactional tables). For this Dimension, turn off the Create Missing Member option:

- 1. Go to the bottom half of the screen and select the Create Missing Member option.
- Click the drop-down button that appears on the far right and select False.
 Note: Alternatively, you can double-click on the Create Missing Member field to change the selection from *True* to *False*.

Note: If you enable this option (set to 'True')—see the additional row (**Missing Member**) displayed directly below –you can enter in the textbox the preferred name for those missing Member[s].

	Column Name	Data Type	Allow Nulls	_			
	Account	nvarchar		- 10	2↓		
•				_	 Database Column Catalag Nama 		
	Parent	nvarchar		_	Catalog Name Schema Name	USING_OLATION dbo	
	Weighting	int	\checkmark		Table Name	dim_Accounts	
	Ordering	int			<u></u>		
					Catalog Name		
							-
<mark>⊉↓</mark> Filte	, C				Ordering		
<mark>≹</mark> ↓ Filte Sort	er t Order				Ordering False		
Filte Sori Cre	, C	1			-		~
Filte Sort Crea Hier	, C er t Order ate Missing Membe				False		~

Save the Dimension

To save the Dimension:

1. Go to the Olation ribbon: in the **Table tab**, click on the **Save** icon.



NOTE, on the left, in the Database Explorer window, when you expand under Dimensions, the newly created **Account** dimension now exists.

The "dim Accounts" pane, when completed, will look as follows.

/ 👿 d	im_Accounts				•	
Design	Data					
	Column Name	Data Type	Allow Nulls	2↓ 🖻		
•	Account	nvarchar		✓ Database Column		
	Parent	nvarchar		Catalog Name	USING_OLATION	
	Weighting	int		Schema Name	dbo	
				Table Name	dim_Accounts	
	Ordering	int		Column Name	Account	
				Allow Nulls Identitifier Column	True False	
				Catalog Name		
Account 2↓ ✓ Cor	mmon					
	timension Name tisplay Name			Account		
	splay Name mension Type			Standard		
	Hierarchy			False		
	a Source Type			Relational Table		
	rarchy Type			Columnar		
	abase Source			USING_OLATION.dbo.dim_Acco	ounts	
	al Aggregate			False		
Filte	er t Order			Ordering		
	t order ate Missing Membe			False		
	ate missing membe rarchies	1		None		
	pping					
	que Identifier					
> Mer	mber			Account		
> Alia				None		
	perties			None		
	osets			None		
	ights			Weighting		
> Hie	rarchy			Parent		
	sion Name name for this dimer	ision				

The next steps will concern creating the Dimensions Version, Month and Region.

4.1.2. Create the Version Dimension

- 1. In the Database Explorer pane, select **dim_Actual_vs_Budget**; right-click on it and select **New Dimension**.
- 2. In the Create New Dimension dialog box, enter Version in the textbox and click Create.
- 3. In the **Design Tab**, select the row corresponding to the **Actual vs Budget** column name; right-click on it and select **Set As Dimension Member**.
- 4. Define a hierarchy for the Version dimension:

For this example, we want to create a hierarchy whereby *Variance* has *Sales* and *Cost of Sales* as child members. To do this, go to the **Design Tab** and right-click on the row beginning **Parent**; select **Define As Hierarchy Level**, then select **At Root**.

/ 🗖 (dim_Accounts 🛛 🔯 dim_Actual_ys_Budget					
Design	Data					
	Actual vs Budget	Parent	Weight			
•	Actual	Variance				
	Budget	Variance		·1		
*						

Note: The screenshot above shows, in the Data Tab, the *Actual vs Budget* column. This column contains *Actual* and *Budget* as Members. The 'Parent' column contains *Variance*. By setting this column as the root (which you did above), *Variance* will calculate as *Actual* minus *Budget*.

5. Set the hierarchy weight for the *Version* dimension: back in the **Design Tab**, select the row corresponding to the **Weight** column name; right-click on it and select **Set As Hierarchy Weight** (see next image).

esign	Data							
	Column Name	Data Type	Allow Nulls			ĝ↓ 🖻		
	Actual vs Budget	nvarchar			~	Database Column		
	Parent	nvarchar				Catalog Name	USING_OLATION	
	Weight	in t				Schema Name	dbo	
	weight	Set As Dimens	sion Unique ID			Table Name	dim_Actual_vs_Budget	
			•			Column Name	Weight	
		Set As Dimens	sion Member			Allow Nulls	True	
		Create As Dim	ension Alias			Identitifier Column	False	
		Create As Dim	ension Property					
		Create As Dim	iension Subset					
		Set As Hierarc	hy Weight		Ca	atalog Name		
		Define As Hier	rarchy Level	•				

- 6. Disable the 'All Hierarchy' option: locate the All Hierarchy setting and select False.
- 7. Disable the 'Total Aggregate': locate the **Total Aggregate** setting and select **False**.

8. Disable the 'Create Missing Member' option: locate the **Create Missing Member** setting and select **False**.

	tine Assessments /		Ded4			
Desigr	dim_Accounts	a aim_Actual_vs	_Budget			
	Column Name	Data Type	Allow Nulls		2↓ 📼	
	Actual vs Budget	nvarchar	\checkmark		✓ Database Column	
	Parent	nvarchar			Catalog Name	USING_OLATION
					Schema Name	dbo
	Weight	int			Table Name	dim_Actual_vs_Budget
					Column Name	Weight
					Allow Nulls	True
					Identitifier Column	False
					Catalog Name	
	-					
• A						
	ommon					
	imension Name			۱	/ersion	
	isplay Name					
	imension Type				Standard	
	l Hierarchy				alse	
	ata Source Type				Relational Table	
	ierarchy Type atabase Source				Columnar	turk in Rodent
-	atabase Source otal Aggregate				JSING_OLATION.dbo.dim_Ao F alse	ctual_vs_budget
	lter				a190	
	ort Order					
	eate Missing Member			I	alse	
	ierarchies			-	None	
	apping					
	nique Identifier					
	ember			1	Actual vs Budget	
> Al	iases				Vone	
> Pr	operties				Vone	

9. Save the **Version** dimension: again, click the **Save Table** icon. The **Version** dimension now appears under **Dimensions** in the Database Explorer.

4.1.3. Create the Month Dimension

Set to true to create a default member into which all missing/null values will be inserted

1. In the Database Explorer pane, select **dim_Months**; right-click on it and select **New Dimension**.

None

Weight

Parent

2. In the Create New Dimension dialog box, enter Month in the textbox and click Create.

Subsets

> Weights

> Hierarchy

Create Missing Member

- 3. In the **Design Tab**, select the row corresponding to the **Month** column name; right-click on it and select **Set As Dimension Member**.
- 4. Begin defining the hierarchy for the Month dimension: right-click on the row corresponding to the **Quarter** column name; select **Define As Hierarchy Level**, then select **At Root**.
- Define an Alias for the Month dimension: for this example, we want to set the Short_Name column as the alias of the Month dimension, so that it can display the month names in their shortened or abbreviated form.
 To do this, right-click on the row corresponding to the Short_Name column name; select Create As Dimension Alias.
- Define a Subset for the Month dimension: for this example, we want to define a Subset, which is a custom list of Members.
 To do this, right-click on the row corresponding to the Subset_Months column name; select Create As Dimension Subset.
- Disable the 'All Hierarchy' option: on the Property Grid locate the All Hierarchy option and select False. (There will be no need to add all months, as they will be aggregated in the following selection.)
- We want to enable the 'Total Aggregate' option by creating a Total of all 'Quarters'. To do this, go to the **Property Grid** on the right and locate the **Total Aggregate** setting, then select **True**.
- Disable the 'Create Missing Member' option: To do this, go to the Property Grid on the right and locate the Create Missing Member setting then select False.
- 10. Here we will do something different: we want to preserve the "Ordering" that has been set up in the Table, so that *January* is followed by *February*, *March*, etc. when the Dimension is built.
 - Go to the **Property Grid** at the bottom then locate and select the **Sort Order** option.
 - Click on the **ellipsis button** corresponding to it on the far right to bring up the **Edit Sort Order** dialog box.
 - Click the Measure Columns icon and select Ordering from the options.
 - Click OK.
- 11. Save the **Month** dimension—it will also appear under **Dimensions** in the Database Explorer pane. Under the dim_Months tab, the Dimension and its attributes will appear as in the following image.

	Column Name	Data Type	Allow Nulls		21		
	Year	nchar			 Database Column 		
•	Quarter	nvarchar			Catalog Name	USING_OLATION	
	Month	nvarchar			Schema Name	dbo	
					Table Name	dim_Months	
	Ordering	int			Column Name	Quarter	
	Subset_Months	nvarchar	\checkmark		Allow Nulls Identitifier Column	True False	
	Short_Name	nvarchar				raise	
					Catalog Name		
					catalog Hallic		
	L com						
ź	1						
	mmon						
	Dimension Name			M	onth		
	Display Name Dimension Type			SI	andard		
	Hierarchy				alse		
	ta Source Type				elational Table		
	rarchy Type			C	Columnar		
	tabase Source				USING_OLATION.dbo.dim_Months		
	tal Aggregate			T	rue		
Filte							
	t Order				Ordering		
	ate Missing Member trarchies				False None		
	apping			N	UNC		
	ique Identifier						
	mber			м	onth		
Alia	ises			S	Short_Name		
	perties				None		
	bsets			S	Subset_Months		
> Weights							
Hie	rarchy			ų	uarter		
	rarchy			Q	uarter		

4.1.4. Create the Region Dimension

- 1. In the Database Explorer pane, select **dim_Regions**; right-click on it and select **New Dimension**.
- 2. In the **Create New Dimension** dialog box, enter **Region** in the textbox and click **Create**.

OLATION

- 3. In the **Design** tab, select the row corresponding to the **Region** column name; right-click on it and select **Set As Dimension Member**.
- 4. Define the hierarchy for the Region dimension, with *World* as the highest-level parent; with the continents as direct children, and; with the regions as the leaf level Members:
 - Right-click on the row corresponding to the **World** column name, select **Define as Hierarchy Level**, then **At Root**.
 - Right-click on the row corresponding to the **Continent** column name; select **Define As Hierarchy Level**, and choose **Sub-Level of World**.
- 5. Disable the 'All Hierarchy' option, i.e., set to False.
- 6. Disable the 'Total Aggregate' option, i.e., set to **False**.
- 7. Disable the 'Create Missing Member' option, i.e., set to **False**.
- 8. Save the **Region** dimension—it will also appear under **Dimensions** in the Database Explorer pane. Under the dim_Regions tab, the Dimension and its attributes will appear as in the following image.

/ 🗖	dim_Accounts	🛿 dim_Actual_vs_Bu	udget 🛛 🔯 dim_Mont	ths dim_Regions 🗸 🗸		
Design	Data					
	Column Name	Data Type	Allow Nulls			
•	Region	nvarchar		✓ Database Column		
, ·	Continent	nyarchar		Catalog Name USING_OLATION		
				Schema Name dbo		
	World	nvarchar		Table Name dim_Regions		
				Column Name Region		
				Allow Nulls True		
				Identitifier Column False		
				Catalog Name		
	ommon mension Name			Region		
	splay Name			region		
	mension Type			Standard		
Al	Hierarchy			False		
	ata Source Type			Relational Table		
	erarchy Type			Columnar		
	atabase Source			USING_OLATION.dbo.dim_Regions		
	otal Aggregate			False		
	ter ort Order					
	eate Missing Member			False		
	erarchies			None		
	apping					
	nique Identifier					
	ember			Region		
	iases			None		
	operties			None		
	ubsets 'eights			None		
	eights erarchy			World/Continent		
	orarony			TOTOLOGINICII		
	e Missing Member true to create a defaul	lt member into which a	all missing/null values will	pe inserted		



This completes the creation of the four Dimensions: *Account*, *Version*, *Month* and *Region*. At this point, you see all four Dimensions listed in the Database Explorer.

4.2. Create a Cube from a Relational Table - SALES Cube

We proceed to the all-important step of creating a Cube. For this exercise, we will create a Cube called SALES, composed of the Dimensions we created previously: *Account, Version, Month* and *Region*.

For this exercise, we will be creating the SALES Cube from the SQL table called *dbo.Factdata*, which should contain all the transactional data.

To create the SALES Cube:

- 1. Among the **Tables** in Database Explorer, select **Factdata**; right-click on it and select **New Cube**. The **Create New Cube** dialog box appears.
- 2. In the dialog box, type a <**name for your cube**>. For this exercise, type **SALES** as the cube name. Note that you can give the Cube any name you want.

Create New Cube	
New Cube	
Name: SALES	
Create	Cancel

3. Click Create. The Add Dimensions to Cube dialog box appears.

Add Dimensions to Cube	
Simensions	
☐ Account ☐ Version ☐ Month ☐ Region	
OK Cancel	

- 4. Select the Dimensions to be included into the Cube by checking the corresponding checkboxes. In this example select all four Dimensions: **Account**, **Version**, **Month**, and **Region**.
- 5. Click **OK**. This opens the **Cube Definition Window** on the right, under the SALES Tab. Note also that a **Cube Tab** also appears along the Olation ribbon.

↓	Olation	— 🗆
File Home Relational Multi-Dimensional	Cube	X
🖻 🖶 🖌 🖾 🔒 🕅	fx# 🛟 🕅 fx fx	D S 🛊 🖄
ave Print Check Dimension Measure Formul	Persistent Cube (F1) Member (F2) Function Generate PC	Dependency Region Cube (F3) Dimension (F4)
• •	Calculation	· · · · · · · · · · · · · · · · · · ·
Cube	Formulas/Persistent Calculations	Dependencies General
tabase Explorer 🛛 🗛 🕽	🛛 🖊 🔯 dim Accounts 🖓 🔯 dim Actual vs_Budget 🕅 🔯 dim	Months 🛛 🔯 dim_Regions 🖉 😚 SALES 🚽
Servers		
Iccalhost		Formulas Dependencies Persistent Calculations
USING_OLATION		
□ ¹ ² Dimensions	✓ Configuration	
∠ Account	Name	SALES
	Display Name	
Version	Туре	Relational Source
🖾 Month	Numeric Values Only	False
🖾 Region	Administrative Cube	False
🗘 Cubes	Relational Synchronization	Rebuild
🗆 🎫 Tables	Synchronization Frequency	On Demand
🗆 🎫 dbo	Read-Only	False
dim Accounts	Transaction Log Batch Relational Updates	False False
dim_Actual_vs_Budget	Formulas	Faise
dim_Actual_vs_budget	Unlimited Calculation Depth	False
_	Maximum Calculation Depth	30
dim_Regions	Use Dependencies	False
III Factdata	Protection	False
🗉 🔯 Views	Expire	False
Queries	 Measure Dimension 	
🗉 🗹 External Data	Dimension Name	SALES Measure
Bookmarks	Database Source	Factdata
	Data Source Type	Relational Table
Local Databases	Query Condition	
	 Microsoft® SSAS Integration 	SALES Hardson
	Synchronize Olation Cube	False
	SSAS Server Name/Address Use Windows Authentication to RDBMS	True
	Calculate values through Olation	True
	Olation Web API Server URL	http://localhost
	SSAS Log File	ntp.//ibouinost
	Name The name of the cube	
	The name of the cube	

4.2.1. Assign the Measure Member

The Measure Member will provide the numbers at the intersections of the Dimensions within the Cube—the "factdata" that the model will provide for reporting, analytics and planning purposes.

- 1. In the Sales Cube Definition Window, go to the Measures Tab.
- 2. Select from the list which item(s) you want to designate as Measures enabling the checkbox(es). For this exercise check the **Amount** checkbox.

🖉 dim_Accounts 🖓 🔯 dim_Actual_vs_Budget 🖓 🔯 dim_Months 🖓 🔯 dim_Regions 🖉 🌍 SALES									
Settings	Measures	Dimensions	Relation	nships Formulas	Depende	ncies Pei	rsistent Calcu	lations	
Column Nam	e	Data Type	Unique Id	Display Name	Format	Increme	Version	Drill	Virtual
🗌 dim_Acti	ua_VS_Budget	nvarchar			*0.0				
dim_Acc	ounts	nvarchar			*0.0				
🗌 dim_Reg	jions	nvarchar			*0.0				
dim_Mor	nths	nvarchar			*0.0				
🗹 Amount		decimal			*0.0				

4.2.2. Arrange the Dimension Order

You can arrange the order of the Dimensions in Olation so that when you report in the preferred client/viewer, it will by default appear in the order you specified. In the Dimensions Tab of the Cube Definition Window, the bottom-most Dimension goes to the Rows, the second to the last Dimension goes to the Columns, while all other Dimensions at the top go to the Filter section.

To define the order of the Dimensions:

- 1. In the Sales Cube Definition Window, go to the Dimensions Tab.
- 2. Drag and drop the Dimensions in the following order as shown in the image below:

🖉 dim_Accounts 🛛 🔯	dim_Actual_vs_Budget 🛛 🔯 dim_Months 🖉 🔯 dim_Regions 🖉 😚 SALES	→ ×
Settings Measures	Dimensions Relationships Formulas Dependencies Persistent Calculations	
Dimension Name	Туре	
SALES Measure	🖄 Measure	
Version	1∠, Standard	
Region	l∠, Standard	
Account	🗠 Standard	
Month	I∠, Standard	

4.2.3. **Define Relationships**

We will next create relationships between the relational tables and the Measure table. The objective here is to create a link between the data or columns of the Measure table with the corresponding columns of other tables, views or Dimensions.

1. Go to the **Relationships Tab**. This will display the Dimension tables and the Measure table.

The four Dimension tables—*Version, Account, Region* and *Month*—appear. Following this example, you will need to link all four Dimension tables to the *Measure* table. To create a link, click on a column name from the Measure table on the left then drag and drop to the appropriate column name on the relational or Dimension table on the right. You will see an arrow that links one part of the Measure table to the corresponding place on the other relational table.

- 2. Click on **dim_Actual_Vs_Budget** from the *Measure Dimension* table on the left; drag and drop to **Actual vs Budget** in the *Version Dimension* table on the right.
- 3. Click on **dim_Accounts** from the *Measure Dimension* table on the left; drag and drop to **Account** in the *Account Dimension* table on the right.
- 4. Click on **dim_Regions** from the *Measure Dimension* table on the left; drag and drop to **Region** in the *Region Dimension* table on the right.
- 5. Click on **dim_Months** from the *Measure Dimension* table on the left; drag and drop to **Month** in the *Month Dimension* table on the right.



The completed Relationships Tab will look as in the below image.

Notice that tables are color-coded in such a way that the Measure table is always Blue, other Dimension tables are Orange, and—though one was not created here—Date dimensions appear Green. Inserting another intermediate table will bring it up in this tab as another color.

4.2.4. Define Cube Setting: Disable READ-ONLY Setting

For this exercise, we want to disable the Read-Only setting to allow the SALES cube to have the write-back capability between a front-end and the Olation cube:

- 1. In the Sales Cube Definition Window, go to the Settings (first) Tab.
- 2. In the Configuration section, locate the **Read-Only** setting, click on the corresponding drop-down button and select **False**.

4.2.5. Save the Cube

To save the Cube:

1. Go to the **Cube Tab** on the Olation ribbon and click the **Save** icon.



Once saved, the Sales cube is listed under Cubes in the Database Explorer.

◯ 〒 File Home Relational Multi-Dimension	al Cu	Olation	- 🗆 Style + 🥅 MA
			1.12
👘 🖶 🗸 🖾 🖬 🖞	x 1	🗙 📬 🖾 fx fx	🕩 🌑 😚 🖄
			Dependency Region Cube (F3) Dimension (F4)
	- Cal	culation • • •	
Cube		Formulas/Persistent Calculations	Dependencies General
Database Explorer 🛛 🖓	×	🔯 dim_Accounts 🛛 🔯 dim_Actual_vs_Budget 🖓 🔯 dim_	Months 🛛 🔯 dim Regions 🏈 SALES
Servers			Formulas Dependencies Persistent Calculations
Iocalhost			Formulas Dependencies Feisisteni Calculations
USING_OLATION		2↓ 🖻	
L Dimensions	~	Configuration	
🖾 Account		Name Display Name	SALES
Version		Type	Relational Source
🖾 Month		Numeric Values Only	False
🖾 Region		Administrative Cube	False
📃 🗘 Cubes		Relational Synchronization	Rebuild
SALES		Synchronization Frequency	On Demand False
Tables		Read-Only Transaction Log	False
🗉 🎫 dbo		Batch Relational Updates	False
🗉 式 Views	~		
Queries		Unlimited Calculation Depth	False
External Data		Maximum Calculation Depth	30
Bookmarks		Use Dependencies Protection	False False
		Expire	False
O Local Databases			1 4100
		Dimension Name	SALES Measure
		Database Source	Factdata
		Data Source Type	Relational Table
		Query Condition Microsoft® SSAS Integration	
		Synchronize Olation Cube	False
		SSAS Server Name/Address	
		Use Windows Authentication to RDBMS	True
		Calculate values through Olation	True
		Olation Web API Server URL SSAS Log File	http://localhost
		OLEDB for OLAP	
		Qualify Names	False
	~	Persistent Calculations	
		Recurse only on Change	True
		Maximum Persistent Calculation Depth Generate Values on Load	3 False
		Power0LAP® Integration	1 0150
		Link Olation Cube to PowerOLAP	False
		PowerOLAP Server Name	localhost
		PowerOLAP Database Name	False
		Olation Server Name Calculation engine	localhost True
		Read-only	False
	~	Saving	
		Save Cube to File	False
	~	Value Caching	
		Cache Intermediate Aggregates Cache Level	True High
		ame	
	Т	ne name of the cube	

5. Viewing the OLATION Data

Olation[®] supports many clients as front ends or browsers of data, providing users a wide selection when working with an OLATION model.

5.1. Create the Dynamic Excel Front End -- PowerExcel Slice

For this exercise, we will use PowerExcel as the front-end client to view and access the Olation Sales cube.

To create a PowerExcel Slice:

- 1. Launch the Excel application and go to the PowerExcel Tab.
- 2. First, create a '**PowerExcel connection**'. To do this:
 - In the PowerExcel Connections control group, click the **Connections** icon.

AutoSave 💽 🖪 🏷 ୯୦ 🗢	Book1 - Excel		Ē	– 🗆 🗙
File Home Insert Page Layout For	mulas Data Review View De	eveloper Help <mark>PowerExcel</mark>	♀ Search	5
New Open Save Refresh Show Find Options Sidebar OLA Fx	Connections Options Clear Cache Dimension	About Change PowerExcel Password		
PowerExcel Slice	owerExcel Connection Dimension	Help		^

- In the Connections dialog that appears, click New.
- In the Name field, enter the <name of the PowerExcel connection>.
 For example, name this as USING_OLATION PXL.
- In the URL field, enter the <correct URL>.
 Note: This URL will be the URL of the Olation Server where the source Olation database is currently running/opened. (Note that localhost and the correct port are used in this example.)
- Click on the Database drop-down and select the source Olation database (i.e., USING_OLATION)—see next image.
 Important: The source Olation database must be running/opened in an Olation server so it can be accessible via PowerExcel.
- Click OK.

The connection is established; assuming all remains the same, these steps will not need to be repeated to create Slices from the selected database from this point forward.

Connection	s		Х
Connections:	USING_OLATION PowerExcel Olation		New Delete
Name:	USING_OLATION		
URL: Database:	http://localhost:4387 USING_OLATION	~	
	Use Windows Authentication	Cancel	

3. Create a **PowerExcel Slice**.

To do this:

• In the **PowerExcel Tab** of the Excel ribbon, go to the PowerExcel Slice control group and click the **New** or **New PowerExcel Slice icon**.

AutoSave 💽 🖪 🏷 🤆 🗢	Book1 - Excel	b – – ×
File Home Insert Page Layout Form	nulas Data Review View Developer Help	PowerExcel $ ho$ Search 🖻 🖵
New Open Save Refresh Show Find Options	Connections Options Clear Edit About Change	
Sidebar OLA Fx	Cache Dimension PowerExcel Connection Dimension Help	

- In the **PowerExcel sidebar** that appears on the right-hand side of Excel select the <**correct PowerExcel connection**>. Following this example, select **USING_OLATION PXL**.
- Select the <correct Cube>. For this example, select SALES.
- Re-arrange the Dimensions in your PowerExcel Slice. Drag and drop the Dimensions along the Filter, Column and Row sections of the PowerExcel sidebar.

Arrange the Dimensions as follows:

	FILTER:
	SALES Measure
	Version
	Region
	COLUMNS:
	Account
	ROWS:
	Month
L	

- 5. Set the display Members for the Dimensions along the Filter section. To do this:
 - In the Filter section of the PowerExcel sidebar, double-click on a Dimension.
 - Clear/Delete the Members appearing on the right-hand pane of the Select Members dialog. Then drag and drop the preferred display Member from the left-hand side to the right-hand side of the dialog.

Note: The Member appearing on the right-hand side of the Select Members dialog will be set as the display Member. In this example (see next image), *Actual* is the Member for the *Version* dimension, and *United States* is the Member for the *Region* dimension. **Note:** You can make use of the toolbar buttons located at the upper portion of the Select Members dialog to configure your display Member.

- Click the green checkmark (OK button) to commit the changes.
- 6. Set the display Members for the Column/Row section. To do this:
 - In the Column or Row section of the PowerExcel sidebar, double-click on a Dimension.
 - Clear/Delete the Members appearing on the right-hand pane of the Select Members dialog. Then drag and drop the preferred display Member/s from the left-hand side to the right-hand side of the dialog.

Note: The Members appearing on the right-hand side of the Select Members dialog will be displayed along the Columns or Rows of the resulting PowerExcel Slice.

Note: You can make use of the toolbar buttons located at the upper portion of the Select Members dialog to configure the display Members for the columns and rows.

• Define the order that you want the Members to appear by dragging and dropping them on top or below each other.

In this example (see next image) Sales, Cost of Sales, Margin, and Margin Pcnt are shown in Columns. And January, February, March, 1st Quarter.... etc., to 4th Quarter and Total Quarter are in Rows

- Click the green checkmark (OK button) to commit the changes.
- 7. Select a PowerExcel Slice type, e.g., **PivotTable**.
- 8. Select where you want to insert the PowerExcel Slice, i.e., whether in a New Workbook, New Worksheet or Current Worksheet. Following our example, select **Current Worksheet**.
- Define the starting cell where the PowerExcel will be generated into. Following this example, select the cell A1 as the starting point.
- 10. Click the Insert button. The resulting PowerExcel Slice will appear. Note: If you configure your Olation Cube to have a write-back capability and you have the access rights (assuming you are accessing a secured Olation database), then you will be able to enter values in your PowerExcel Slice and push those updates back to the Cube. Bear in mind, however, that you can write only on Detail (non-Aggregate) intersections.

	AutoSave 💽 🕅	8 9 ° °			Boo	ok1 - Exe			Sign in		×
F	ile Home	Insert Page	Layout Formu	las Data	Review Vi	iew I	Developer	Help	PowerExcel 🔎 Search	Ŕ	P
< N		Refresh Show Sidebar C		Connections Option	Cache D	Edit Edit Imension		Change Change cel Password Help			
В1	L			• : × ·		OLADat:	abase("USIN	IG_OLATIO	N")		,
	А	В	с	D	E		F	G 🔺			
_	Database:	USING_OLATION	1						PowerExcel	.	×
2	Cube:	SALES							Database		
	Dimensions:	Filter	SALES Measure	Members	Amount				USING_OLATION		~ 7
		Filter	Version	Members	Actual				Cube		
_		Filter	Region	Members	United State				SALES	•	~ 14
		Column	Account	Range	\$B\$10:\$E\$10				T Filters		
,		Row	Month	Range	\$A\$11:\$A\$2	6			¹ 公 SALES Measure: Amount	t	
3	OLAPivotTable								12 Version: Actual		
, 0	OLAPIVOTIADIE	Sales	Cost of Sales	Margin	Margin Pcnt				^L Region: United	States	
	January	5ares 5000		4160	-	0					
	February	5285		4325		0					
	March	5586		5006		0			🔁 Columns		
	1st Quarter	15871	2380	13491		0			ピ Account: Sales,Cost of S	ales, Margin, Margin Pcnt	
5	April	5905	700	5205		0					
б	May	6241	660	5581		0					
7	June	6597	700	5897		0					
8	2nd Quarter	18743	2060	16683		0			Rows	a March 1st Occurrence and March	
	July	6973		5953		0			⊷ Month: January,Februar	y, March, 1st Quarter, April, May,	June,2
-	August	7370		6570		0					
-	September	7791		6931		0			<		>
-	3rd Quarter	22134		19454		0			Options	PowerExcel Slice	
3	October November	3591 4453		3591 4453		0			Constrain Empty Rows	PivotTable	
_		5027		5027		0			Delete Removed Rows	Read/Write Formulas	
5 6		13071		13071		0			Expandable Members	Power Query	
7		15571		15071		-			LApanuable Members	O Forei Query	
8									Insert Into		
9									🔵 New Workbook 🛛 🖲) Current Worksheet	
0									○ New Worksheet	Location: \$A\$1	Y
1											
2									🔲 Update		
0	sh Sh	eet1 (+)			4			▼			
	Sn Sn	teerr +			•			•			

You can also 'Stack' or 'Nest' Dimensions. For this example:

- In the PowerExcel sidebar, drag the Version dimension on the Columns section just above the Account dimension.
- Next, double-click on the Version dimension to bring up the Select Members dialog and • select Actual and Budget as display Members.
- Click the green checkmark button to go back to the PowerExcel Slice then click the Update button. The Slice will look as follows:

OLATION

AB C D E F G H I 1 Database: USING_OLATION E F G H I 2 Cube: SALES Distasse: USING_OLATION Image: September Sept	Δ	utoSave 💽	5 °C	^y + ≠				Book1 - Exc	:el		■ – □ ×	
New Open Sove Refreit States Field Deverfact	Fil	e Home	Insert Pa	ge Layout	Formulas E	Data Review	v View	Developer	Help Pow	erExcel 🔎 Search	n 🖻 Share 🖵 Comments	
PowerExcel Since PowerExcel Connection Dimension Help A3	C Ne) 🗁 🖳	Refresh Show		-		ar Edit		Change			
A B C D E F G H I 1 Database: USING_QLATION Database: USING_QLATION > > Database: USING_QLATION > > Database: USING_QLATION > Database: USING_QLATION > Database: USING_QLATION > Database: USING_QLATION > > > Database: USING_QLATION > > > Database: USING_QLATION > > > > Database: USING_QLATION > > > Database: USING_QLATION > > > Database: USING_QLATION > > Database: USING_QLATION > > Database: Database: D		P		01 012 H X	PowerE						~	
A B C D E F G H I 1 Database: USING_QLATION Database: USING_QLATION > > Database: USING_QLATION > > Database: USING_QLATION > Database: USING_QLATION > Database: USING_QLATION > Database: USING_QLATION > > > Database: USING_QLATION > > > Database: USING_QLATION > > > > Database: USING_QLATION > > > Database: USING_QLATION > > > Database: USING_QLATION > > Database: USING_QLATION > > Database: Database: D		19 🔹 😧 🖉 👘 ΞΟΙ ΔΡίνατΤαϸίρ(ἐΒέ) έβές έβές: ἐβές:										
Database: USING_OLATION PowerExcel × × 2 Cube: SALES SALES Amount <td>A9</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>>D>Z,>D>3;>E></td> <td>3, 9093; 9093, 9094; 9094</td> <td>, 5856:5256,5857:5257,58512,FALSE,FALSE,FALSE) *</td>	A9								>D>Z,>D>3;>E>	3, 9093; 9093, 9094; 9094	, 5856:5256,5857:5257,58512,FALSE,FALSE,FALSE) *	
Datasse: Oshie Outlon Image	1		-	-	D	E	F	G	Н	▲	BowerEvcol 🔹 🗙	
2 Dimensions: Filter SALES Measur Members Amount 4 Filter Region Members United States 5 Column1 Account Range \$\$511:\$511 Column2 Column2 Account Range \$\$511:\$511 Column2 Account Range \$\$511:\$511 Filter Filters SALES Version Range \$\$511:\$511 Column2 Account Actual Actual Actual Actual Actual Actual Actual Actual Actual Margin Port Sales Cost of Sales Margin Margin Port Columns Listics Listics Listics Listics Columns 2 January 5000 8440 0 4444 0 4444 0 4444 0 4444 0 Listics	_		_	ON							POWerexcer	
Image: Pilter Region Members United States Column1 Version Range \$\$\$\$10\$\$\$10 States	_											
Column1 Version Range \$B\$10:\$i\$10 Column2 Column2 Account Range \$B\$11:\$i\$11 Column2 Column3 Columa3 Column3 Column3	-										USING_OLATION V	
S Column2 Account Range \$911:\$i\$11 Filters 0 Actual Margin Port Sales Cost of Sales Margin Margin Port Sales Cost of Sales Margin Port Margin Port Sales Cost of Sales Margin Port Sales Cost of Sales	4 5											
Row Month Range \$A\$12:\$A\$27 0 Actual Margin Port Sales Cost of Sales Margin Port Margin Port Margin Port Sales Cost of Sales Margin Port Margin Port Margin Port Sales Cost of Sales Margin Port Margin Port Margin Port Sales Cost of Sales Margin Port Margin Port Margin Port Sales Cost of Sales Margin Port Sales Costof Sales Margin Port Sales	5										SALES 🗸 🧏	
Image: Solution of the second secon	ъ 7											
OLAP!votTable Actual Actual Actual Budget	/ 8		RUW	NUTIT	Range	\$A\$12;\$A\$27						
Actual Actual Actual Budget Budget<											🖾 Region: United States	
1 Sales Cost of Sales Margin Margin Port Margin Port<	10		Actual	Actual	Actual	Actual	Budget	Budget	Budget	Budget		
2 January 500 840 4160 0 4444 0 4444 0 3 February 5285 960 4325 0 4444 0 4444 0 4 March 5586 580 5006 0 4444 0 4444 0 5 1st Quarter 15871 2380 13431 0 13332 0 0 13332 0 13332 0 13332 0 13332 0 0 13332 0 13332 0 13332 0 13332 0 0 0 0 0 0 0 13332 0	11						-		0			
a February 5285 960 4325 0 4444 0 4444 0 4 March 5586 580 5006 0 4444 0 4444 0 5 1st Quarter 15871 2380 13491 0 13332 0 13332 0 13332 0 4444 0 6 April 5905 700 5087 0 4444 0 4444 0 8 June 6597 700 5897 0 4444 0 4444 0 0 July 6973 1020 5953 0 4444 0 4444 0 1 August 7370 860 6931 0 4444 0 4444 0 2 September 7791 860 6931 0 4444 0 4444 0 3 3rd Quarter 2134 2680 19454 0 13332 0 13332 0 13332 0 13332 0	-					0			0	~		
4 March 5586 580 5006 0 4444 0 4444 0 5 1st Quarter 15871 2380 13491 0 13332 0 13332 0 6 April 5995 700 5205 0 4444 0 4444 0 8 June 6597 700 5897 0 4444 0 4444 0 9 2nd Quarter 18749 2060 16683 0 13332 0 13332 0 1 August 7370 800 6570 0 4444 0 4444 0 2 September 7791 860 6931 0 4444 0 4444 0 3 3rd Quarter 2214 2680 19454 0 13332 0 13332 0 13332 0 13332 0 Constrain Empty Rows PowerExcel Slice PowerExcel Slice PowerExcel Slice PowerExcel Slice PowerQuery PowerQuery PowerDuery PowerDuery <t< td=""><td>_</td><td>'</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>🌄 Columns</td></t<>	_	'									🌄 Columns	
5 1st Quarter 15871 2380 13491 0 13332 0 13332 0 6 April 5905 700 5205 0 4444 0 4444 0 7 May 6241 660 5581 0 4444 0 4444 0 8 June 6597 0 16897 0 4444 0 4444 0 9 2nd Quarter 18743 2060 16668 0 13332 0 13332 0 1 August 7370 800 6570 0 4444 0 4444 0 1 August 7370 800 65931 0 4444 0 4444 0 3 ard Quarter 22134 2680 19454 0 13332 0 13332 0 13332 0 5 November 4453 0 4444 0 4444 0 0 0 0 0 0 0 0 0 0	_									-		
6 April 5905 700 5205 0 4444 0 7 May 6241 660 5581 0 4444 0 4444 0 8 June 6597 700 5997 0 4444 0 4444 0 9 2nd Quarter 18743 2060 16683 0 13332 0 13332 0 1 August 7370 800 6570 0 4444 0 4444 0 2 September 7791 860 6931 0 4444 0 4444 0 3 ard Quarter 21332 0 13332 0 13332 0 13332 0 4 October 3591 0 4444 0 4444 0 0 0 0 0 13332 0 13332 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 <t< td=""><td>_</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>-</td><td>ピ、Account: Sales,Cost of Sales,Margin,Margin Pcnt</td></t<>	_									-	ピ、Account: Sales,Cost of Sales,Margin,Margin Pcnt	
7 May 6241 660 5581 0 4444 0 4444 0 8 June 6597 700 5997 0 4444 0 4444 0 9 2nd Quarter 18743 2060 16683 0 13332 0 13332 0 13332 0 13332 0 13332 0 1332 0 1 1 August 7370 800 6570 0 4444 0 4444 0 4444 0 4444 0 4444 0 4444 0 4444 0 4444 0 4444 0 4444 0 4444 0 4444 0 4444 0 4444 0 4444 0 6 0 cober 3591 0 4444 0 4444 0 6 December 5027 0 4444 0 4444 0 6 December 5027 0 5027 0 4444 0 4444 0 6 Delete Removed Rows Read/Write Formulas <td>_</td> <td></td> <td></td> <td></td> <td></td> <td>0</td> <td></td> <td></td> <td></td> <td></td> <td></td>	_					0						
8 June 6597 700 5897 0 4444 0 4444 0 9 2nd Quarter 18743 2060 16683 0 13332 0 13332 0 13332 0 0 July 6973 1020 5953 0 4444 0 4444 0 1 August 7370 800 6570 0 4444 0 4444 0 2 September 7791 860 6931 0 4444 0 4444 0 3 3d Quarter 22134 2680 19454 0 13332 0 13332 0 6 December 3591 0 4444 0 4444 0 Coststrain EmptyRows © PivotTable © PowerExcel Silce © Coststrain EmptyRows © PowerQuery © Read/Write Formulas 6 December 5027 0 4444 0 4444 0 4444 0 Expandable Members © Power Query © Read/Write Formulas © Read/Write Formulas © Read/Write	_					0	4444	0	4444	1 0		
9 2nd Quarter 18743 2060 16683 0 13332 0 13332 0 0 July 6973 1020 5953 0 4444 0 4444 0 1 Jugust 7370 800 6570 0 4444 0 4444 0 2 September 7791 860 6931 0 4444 0 4444 0 3 3rd Quarter 22134 2680 19454 0 13332 0 13332 0 3 October 3591 0 4444 0 4444 0 6 Constrain Empty Rows © PowerStxel Slice Options PowerStxel Slice Options PowerQuery PowerQuery New Workbook Read/Write Formulas Power Query Power Query Power Query Inset. Into New Workbook Current Worksheet New Worksheet Location: \$451 Yet 9	_					0						
1 August 7370 800 6570 0 4444 0 4444 0 2 September 7791 860 65931 0 4444 0 4444 0 3 3rd Quarter 22134 2680 19454 0 13332 0 13332 0 4 October 3591 0 3932 0 4444 0 4444 0 6 December 5027 0 4444 0 4444 0 <t< td=""><td>19</td><td>2nd Quarter</td><td></td><td></td><td>16683</td><td>0</td><td>13332</td><td>0</td><td>13332</td><td>2 0</td><td>ピ、Month: January, February, March, 1st Quarter, April, May, June, 2</td></t<>	19	2nd Quarter			16683	0	13332	0	13332	2 0	ピ、Month: January, February, March, 1st Quarter, April, May, June, 2	
2 September 7791 860 6931 0 4444 0 4444 0 3 3/d Quarter 22134 2680 19454 0 13332 0 New Workbook © Current Worksheet New Workbook © Current Worksheet New Worksheet New Worksheet New Worksheet New Worksheet I coation: \$A\$11 Yet 1	20.	July	6973	1020	5953	0	4444	0	4444	1 0		
2 September 7791 860 6931 0 4444 0 4444 0 3 3 3/d Quarter 22134 2660 19454 0 13332 0 13332 0 13332 0 0 0 0 0 0 0 14332 0 13332 0	21	August	7370	800	6570	0	4444	0	4444	1 0		
a for dealer 2414 2000 17454 0 17454 0 17454 0 4 October 3591 0 4444 0 4444 0 0 10000			7791	860	6931	0	4444	0	4444	1 0		
Stocker 3051 0 00000 0	23	3rd Quarter	22134	2680	19454	0	13332	0	13332	2 0		
6 December 5027 0 5027 0 4444 0 4444 0 Figure 1 Expandable Members Power Query 7 4th Quarter 13071 0 13332 0 13332 0 Inset Into Inset Into New Workbook © Current Worksheet 0 New Worksheet Location: \$A\$1 Ye 1	24	October	3591	0	3591	0	4444	0	4444	1 0	Constrain Empty Rows PivotTable	
7 4th Quarter 13071 0 13332 0 13332 0 Insert Into 8	25	November	4453	0	4453	0	4444	0	4444	1 0	Delete Removed Rows O Read/Write Formulas	
8 Insert Into 9 Insert Into 0 Insert Into 1 Insert Into	26	December	5027	0	5027	0	4444	0	4444	1 0	Expandable Members O Power Query	
8 Image: Constraint of the second s	_	4th Quarter	13071	0	13071	0	13332	0	13332	2 0		
9 O <td>28</td> <td></td>	28											
	29											
	30									_	O New Worksheet Location: \$4\$1	
Sheet1 (+) : (31											
		Sh	eet1 (+)			÷ •			•	Update	
eedy 🛅 🔲 – + 100%	Read	ty 🛅									III II + 100%	

11. Revert to the previous PowerExcel Slice orientation by moving the **Version** dimension back to the Filter area and selecting **Actual** as the display Member.

By inspecting each Dimension (to inspect, double-click on these Dimensions along the PowerExcel sidebar to bring up the Select Members dialog), we see the following Dimensions and their corresponding Members: (Note that the images show both the Members and Hierarchy Tabs.)

IMPORTANT: If the Dimensions are placed along the Filter section of the PowerExcel Slice, when you bring up the Select Members dialog you will only see the Members and Hierarchy Tabs.

If the Dimensions are placed along the Rows or Columns sections of the PowerExcel Slice, when you bring up the Select Members dialog, you will see the Members, Hierarchy and Subsets Tabs displayed.


Dimension: SALES Measure









Dimension: Account



5.2. Viewing OLATION Data via PowerOLAP's OLATION Linked Cube Feature

PowerOLAP—along with PowerExcel—is a PARIS Technologies product that also provides front-end capabilities to work with an Olation model. This section presupposes some familiarity with the use of PowerOLAP—including the first few steps that follow.

5.2.1. Create a New PowerOLAP Database

For this demo, we will create a New PowerOLAP database and load it on the PowerOLAP MDB Server.

To create a new PowerOLAP Database (directly loaded on the PowerOLAP Server):

1. First, launch the **PowerOLAP Server** and select a **server** where you want to create your new PowerOLAP database.

For this example, let us just choose the **localhost**.

Image: Save Save and Backup Database Databas		rOLAP Server Control Program	- 0	
Stat Stop ProductPlanning User Number of Current Users: 0 Block Users User Database Task Status User Database Task Status User Database Task Status Cancel All Calculations	Copen Save Save and Backup Close tabase Database Database	Options Refresh About		
	alhost HR Demo Mona Lisa PandA_Base_Clean_v2 ProductPlanning 2 Cancel All Calcu Settings Synchronization S C:\Program Files TCP/IP Port: 45 Enable Secu	Stop Server will stop in Minutes. Inform Users Inform Users Minutes. Users: 0 Block Users Database Task Status ulations Cancel User Calculation Disconnect User Server Database Location: Browse [x86]\Power0LAP Server\SS Databases 30 HTTP Port: 80 Enable Diagnostic Logging inty Browse Intervention		
	Password:			

2. Click on the **New Database** command to create a new database—for this example we will call it **OLATION_Report**.

5.2.2. Create a New OLATION Linked Cube

Now that the OLATION_Report database is opened in the PowerOLAP Modeler client, you can proceed to create the Olation Linked Cube:

1. In the PowerOLAP ribbon, go to the **Tools Tab**; then in the OLATION Cube control group select the **New Linked Cube** command icon.

This will bring up the OLATION Data Source dialog where the user specifies a valid provider to be used to allow PowerOLAP to communicate to a specified database.

	PowerOLAP		_	
Home Model Slice	Data Tools			🗟+ Style 🔻 🕜
Configure Synchronize Remove Expired Locks and Logins Synchronization Server	Image: Segin My Commit My Cancel My Analysis Nev Analysis		v Linked Cube LATION Cube	
For Help, press F1	localhost	OLATION_Report		

Fill out the required fields in the Olation Data Source dialog box:

- Select a Source Type.
 Since in the example, we are trying to access a local Olation database, choose OLATION Server.
- 3. In the **Source** field, type in the appropriate source. Following this exercise, enter **local** or **localhost**.

Source Type:	OLATION Server	
Source:	localhost 🗨	
	Connect using Windows Authentication	
User:		
Password:		

4. Click Next.

Fill out the required fields in the succeeding Database Name dialog box:

- 5. Click on the **Database drop-down** and select the name of the Olation database you want to connect to. For this example, select **USING_OLATION**.
- 6. Choose how to connect to a Secured Olation Database. Connect either via *Windows Authentication* or by providing a valid *User* and *Password*. For this exercise, since the source Olation database has no Security, we can skip this step.

Dat	abase Name		×
	– Database – – –		_
	Database:		
	Dalabase.		
	Name:	_	
		Connect using Windows Authentication	
	User:		
	Password:		
		< Back Next > Cancel Help	

7. Click Next. The OLATION Cube dialog appears.

OLATION Cube				×
Cube Source Cube:	SALES			
	, 🔽 Linked	🔲 Read-Only	🗌 Use Olatio	n Security
	< Back	Finish	Cancel	Help

- 8. In the **OLATION Cube** dialog box, select **SALES** as the source Cube.
- 9. Take notice of the three checkboxes at the bottom: *Linked*, *Read-Only* and *Use Olation Security*. For this exercise, since we want to allow write-back and we are accessing an unsecured Olation database, enable the **Linked** checkbox option.
- 10. Click **Finish**—a message appears indicating that the Olation cube was successfully built.
- 11. Click **OK**.

5.2.3. Viewing the OLATION Metadata Structure built in PowerOLAP

We will next view the Cube and Dimension (the metadata structure) built in PowerOLAP by creating the Linked Cube from Olation.

'Account' Hierarchy	— 🗆 X				
Subset: All 💽 Filter:	• 🕼 🗇 🗇 🖉 🐣 祭 祭 🎝 🕈 🖓 🗸				
X All X Margin # Sales # Cost of Sales # Margin Pont	B[∠ Account b Σ All m				
Account Dimension					





Member	s Short_Name	L	ОК
1 Total Quar	ter		
2 1st Quarte	r		Add Group.
3 2nd Quarte	er		Rename Grou
4 3rd Quarte	r		
5 4th Quarte	r		Delete Grou
5 January	Jan		
7 February	Feb		Help
3 March	Mar		
9 April	Apr		
0 May	May		
1 June	Jun		
2 July	Jul		
3 August	Aug		
4 Septembe	r Sep		
5 October	Oct		
6 November	Nov		
7 December	Dec		

You can also view the **Alias** and **Subset** Groups that you have configured for the *Month* dimension.

The alias created from the Short_Name column in the dbo.dim_Months table



e subset created from the Subset_Months column in t	ne <i>abo.aim_iviontns</i> ta
Cubes	×
l I	Add
SALES	Delete
	Rename

The subset created from the Subset Months column in the dbo.dim Months table

The SALES Olation Cube built in PowerOLAP

5.2.4. Viewing the Fact Data - Create a PowerOLAP Slice View

Next we will create a PowerOLAP Slice to see the transactional data coming from the Olation cube.

1. In the PowerOLAP ribbon, go to the Slice Tab, and in the Slice control group select the New Slice command icon.



The New Slice dialog appears, displaying existing Cubes in the database-in this case, the SALES cube.

New Slice		×
Cube:		
SALES		
ОК	Cancel	Help

2. With the **SALES** cube selected, click **OK**. Press the **F9** key or click the **Recalculate** button to show data in the Slice.

		Power	DLAP - [SALES : Untitled	11			– 🗆 X
		- over	in the second se	.1		B+ :	Style - 🙆 💷 🕱
Home Model Slice	Data Tools					<u>a</u> + :	style 💘 🖉 💷 🗶
+ C Close Save Save Slice Slice Slic		Paste Cut Copy	Clear Fill Fill Down Right	Graph Constrain	Drill Worksheet	Recalculate	Tile Arrange
Slice		Clipboard	Edit	Options	Work	sheet	Window
Filter:		+ All	– Margin	Sales	Cost of Sales	Margin Pont	
SALES Measure: Amount	– Total Quarter	129026.00	84218.00	106622.00	22404.00	0.00	
Version: Variance	+ 1st Quarter	112364.00	99344.00	105854.00	6510.00	0.00	
C Region: World	+ 2nd Quarter	16907.00	6007.00	11457.00	5450.00	0.00	
	+ 3rd Quarter	18722.00	3834.00	11278.00	7444.00	0.00	
	+ 4th Quarter	-18967.00	-24967.00	-21967.00	3000.00	0.00	
	January	104926.00	99746.00	102336.00	2590.00	0.00	
	February	2620.00	-1700.00	460.00	2160.00	0.00	
	March	4818.00	1298.00	3058.00	1760.00	0.00	
Column Labels:	April	5348.00	1948.00	3648.00	1700.00	0.00	
	May	5717.00	1977.00	3847.00	1870.00	0.00	
	June	5842.00	2082.00	3962.00	1880.00	0.00	
	July	6227.00	1707.00	3967.00	2260.00	0.00	
	August	6857.00	789.00	3823.00	3034.00	0.00	
Row Labels:	September	5638.00	1338.00	3488.00	2150.00	0.00	
C Month: All	October	-4038.00	-6038.00	-5038.00	1000.00	0.00	
	November	-7944.00	-9944.00	-8944.00	1000.00	0.00	
	December	-6985.00	-8985.00	-7985.00	1000.00	0.00	
For Help, press F1			lo	talhost	OLATION_Report		

Note: You can also re-arrange the Slice or change the Filter Members as per your preference.

5.2.5. Arranging the Slice View

1. You can rearrange the Slice by dragging Dimensions along the Filter, Column and Row panes. You can also change the page Members to be displayed, and specify Members to be displayed along the columns and rows.

Next, to re-arrange the Slice view and then create a dynamic Excel front end:

• Go to the **Filter section** and double-click on the Dimension whose display Member you want to change. For example, double-click on the **Version** Dimension; in the **Edit Slice** dialog, go to the right-hand pane and double-click on the preferred display Member. Alternatively, you can also select the new display Member on the right-hand pane and click

the **Set Page Member** icon that can be found along the toolbar at the top.

Note: If the preferred Member is not in the right-hand pane, just drag and drop the appropriate Member from the Member list on the left-hand pane of the Edit Slice dialog box to the right-hand pane. Bear in mind that the Member marked in yellow is the currently selected display Member.

Make your Filter selections as follows:

FILTER:	
SALES Measure: Amount	
Version: Actual	
Region: United States	

Next, change the display Members for the 'Account' dimension along the Columns so that only Sales, Cost of Sales, Margin, and Margin Pcnt appears.
 Go to Column Labels section and double-click on Account. In the Edit Slice dialog that appears, include in the right-hand pane only those Members you want to see. In this case, make sure that only Sales, Cost of Sales, Margin, and Margin Pcnt are included, arranging them in that order. Click the green checkmark (OK button) when done.

Note: You can use the toolbar located at the top of the right-hand pane of this dialog, or you can also use drag and drop or the delete key to add or remove Members to the display list. You can also drag and drop Members into any order you wish.

• Go to the **Row Labels section** and double-click on **Month**. In the **Edit Slice** dialog, order the Members along the Rows so that months appear above their respective quarters and the *Total Quarter* aggregate appears at the bottom:

🔯 Edit 'Month' for 'Untitled 1'	— 🗆 X
Members Hierarchy	⊂⊃⊃◊∛≧♀҂?✓
\$\$ } \$\$\$\$\$\$\$\$\$\$\$\$\$	 # January # February # March ∑ 1st Quarter # April # May # June ∑ 2nd Quarter # July # August # September ∑ 3rd Quarter # October # November # December ∑ 4th Quarter ∑ Total Quarter

- Click the green checkmark (OK button) when done to go back to the Slice.
 Note: If you want to move around Dimensions, you can drag and drop them accordingly along the Filter, Column and Row areas.
- Click the Recalculate button along the Slice Tab of the PowerOLAP ribbon or press the F9 key to refresh the Slice view.
 The reconfigured Slice looks as follows:



Filter:	4	Sales	Cost of Sales	+ Margin	Margin Pont
C SALES Measure: Amount	January	5000.00	840.00	4160.00	0.00
Version: Actual	February	5285.00	960.00	4325.00	0.00
🖄 Region: United States	March	5586.00	580.00	5006.00	0.00
	+ 1st Quarter	15871.00	2380.00	13491.00	0.00
	April	5905.00	700.00	5205.00	0.00
	May	6241.00	660.00	5581.00	0.00
	June	6597.00	700.00	5897.00	0.00
olumn Labels:	+ 2nd Quarter	18743.00	2060.00	16683.00	0.00
🛆 Account: Subset	July	6973.00	1020.00	5953.00	0.00
	August	7370.00	800.00	6570.00	0.00
	September	7791.00	860.00	6931.00	0.00
	+ 3rd Quarter	22134.00	2680.00	19454.00	0.00
ow Labels:	October	3591.00	0.00	3591.00	0.00
C. Month: Subset	November	4453.00	0.00	4453.00	0.00
	December	5027.00	0.00	5027.00	0.00
	+ 4th Quarter	13071.00	0.00	13071.00	0.00
	+ Total Quarter	69819.00	7120.00	62699.00	0.00

Note: You can also 'Nest' or 'Stack' Dimensions in a Slice view. An example of a 'Nested Dimension' is shown below, with *Version* and *Account* along the Columns.

ter:			Actual			Budget	
SALES Measure: Amount		Sales	Cost of Sales	+ Margin	Sales	Cost of Sales	+ Margin
A Region: United States	January	5000.00	840.00	4160.00	4444.00	0.00	4444.00
	February	5285.00	960.00	4325.00	4444.00	0.00	4444.00
	March	5586.00	580.00	5006.00	4444.00	0.00	4444.00
	+ 1st Quarter	15871.00	2380.00	13491.00	13332.00	0.00	13332.00
	April	5905.00	700.00	5205.00	4444.00	0.00	4444.00
	May	6241.00	660.00	5581.00	4444.00	0.00	4444.00
	June	6597.00	700.00	5897.00	4444.00	0.00	4444.00
olumn Labels:	+ 2nd Quarter	18743.00	2060.00	16683.00	13332.00	0.00	13332.00
Version: Subset	July	6973.00	1020.00	5953.00	4444.00	0.00	4444.00
🛆 Account: Subset	August	7370.00	800.00	6570.00	4444.00	0.00	4444.00
	September	7791.00	860.00	6931.00	4444.00	0.00	4444.00
	+ 3rd Quarter	22134.00	2680.00	19454.00	13332.00	0.00	13332.00
	October	3591.00	0.00	3591.00	4444.00	0.00	4444.00
ow Labels:	November	4453.00	0.00	4453.00	4444.00	0.00	4444.00
🛆 Month: Subset	December	5027.00	0.00	5027.00	4444.00	0.00	4444.00
	+ 4th Quarter	13071.00	0.00	13071.00	13332.00	0.00	13332.00
	+ Total Quarter	69819.00	7120.00	62699.00	53328.00	0.00	53328.00

- 2. Going back to our previously reconfigured Slice (before a demonstration of 'nesting' Dimensions), save the Slice:
 - With the Slice opened and active (meaning that this should be the currently selected Slice, in case several Slices are opened), go to the **Slice Tab** and click the **Save Slice As** command.



- In the Save Slice As dialog, enter the <name of the Slice>., e.g., Comparative Margin per Quarter.
- Click OK.

5.2.6. Create the Dynamic Excel Front End – PowerOLAP Slice to Excel

We will next create a dynamic, bi-directional Excel front end.

 With the Slice opened and active, go to the PowerOLAP ribbon, Slice Tab, click on the Worksheet drop-down and select OLAPTable as the Excel connection type. Or you can simply press the F8 key.

The following spreadsheet will be generated.

	AutoSave 💽 ศ	8 9 , 6,	÷	Sheet	1 - Excel		Sign	in 🖪	3 —		×
F	ile Home In	sert Page Layou	Formulas Data	Review View	Developer Help	PowerExc	el PowerC	DLAP D	Search	ß	P
	■ E → B	$\begin{array}{c c} \text{ibri} & \bullet & 11 & \bullet \\ \hline I & \underline{U} & \bullet & A^{\bullet} & A \\ \bullet & & & & A^{\bullet} & \bullet \\ \hline & & & & & & \\ \text{Font} \end{array}$	_ `	∃ - \$ - %	😿 Cell Style	as Table =		Insert 👻 Delete 👻 Format * Cells	P Editing	Jdeas Ideas	~
A	1			• : ×	√ <i>f</i> _x Dat	abase:					~
	A	В	с	D	E	F	G	н	I I	L I	
1	Database:	OLATION_Repo	rt								
2	Cube:	SALES									
3	Page:	SALES Measure	Amount								
4		Version	Actual								
5		Region	United States								
6		Account	Along Columns								
7		Month	Along Rows								
8	OLAPTable										
9		Sales	Cost of Sales	Margin	Margin Pcnt						
10	January	5000	840	4160	0						
11	February	5285	960	4325	0						
12	March	5586	580	5006	0						
13	1st Quarter	15871	2380	13491	0						
14	April	5905	700	5205	0						
15	May	6241	660	5581	0						
16	June	6597	700	5897	0						
17	2nd Quarter	18743	2060	16683	0						
18	July	6973	1020	5953	0						
19	August	7370	800	6570	0						
20	September	7791	860	6931	0						
21	3rd Quarter	22134	2680	19454	0						
22	October	3591	0	3591	0						
23	November	4453	0	4453	0						
24	December	5027	0	5027	0						
25	4th Quarter	13071	0	13071	0						
26	Total Quarter	69819	7120	62699	0						
27											
28											

 Now, to show data for a different region, e.g., *Canada*, double-click on cell C5 (this contains the OLAPMember connection reference for the *Region* dimension).

C5	5			• : ×	× ✓ f∗ =OLAPMember(\$B\$1, "Region",8)	
	А	В	С	D	Select A Member	x
1	Database:	OLATION_Re	port		Sector Member	
2	Cube:	SALES			- Region	
3	Page:	SALES Measu	re Amount		Hierarchy Members	
4		Version	Actual			
5		Region	United States		B····Σ World	·
б		Account	Along olumns		B South America	
7		Mo			i → Σ North America	
8	OLAPTable	Double	-click on Cell C	5 which	# Mexico	
9		Sale contain	s the OLAPMen	nber	# United States	
10	January	functio	n for the Region	dimension.	n. In the second se	
11	February					
12	March		Select a Member	0		
13	1st Quarter		pears, select a d			
14	April	Membe	er, e.g., Canada			
15	May					
16	June	65	97 700	58	58	- I
17	2nd Quarter	187	43 2060	166	66	
18	July	69	73 1020	59	59 OK Cancel Help	
19	August	73	70 800	65	65	

3. In the Select a Member dialog that appears, select **Canada** then click **OK**.

4. Back in the Excel worksheet, press **F9** to refresh values.

Notice how the values have changed to reflect data for Canada Actuals.

C5	5			• : ×	$\checkmark f_x$	=OL	APMembe	r(\$B\$1, "R	egion",6)
	А	В	С	D	E		F	G	н
1	Database:	OLATION_Report	rt						
2	Cube:	SALES							
3	Page:	SALES Measure	Amount						
4		Version	Actual						
5		Region	Canada						
6		Account	Along Columns						
7		Month	Along Rows						
8	OLAPTable								
9		Sales	Cost of Sales	Margin	Margin Pc	nt			
10	January	500	500)	0			
11	February	750	500	250)	0			
12	March	2793	500	2293	3	0			
13	1st Quarter	4043	1500	2543	3	0			
14	April	2952	500	2452	2	0			
15	May	3121	500	262:	L	0			
16	June	3298	500	279	3	0			
17	2nd Quarter	9371	1500	787:	L	0			
18	July	3487	500	298	7	0			
19	August	3685	500	3185	5	0			
20	September	3895	500	3395	5	0			
21	3rd Quarter	11067	1500	956	7	0			
22	October	1795	500	1295	5	0			
23	November	1041	500	543	L	0			
24	December	2513	500	2013	3	0			
25	4th Quarter	5349	1500	384	9	0			
26	Total Quarter	29830	6000	23830)	0			
27									

6. OLATION[®] and Real-Time Application Results

We are now prepared for our next step—to show the true dynamism of OLATION[®], working with Excel and PowerExcel, and, importantly, the "real, real time" applications that can be created—for example, for Sales planning.

6.1. Pull Data Updates from the Relational Source down to the OLATION Database

This section concerns data updates from a source relational database, which will be reflected into an Olation model. The screenshot below shows a PowerExcel report previously created out of the Olation database called USING_OLATION. Note that there is no Budget data (all *zeros*) for the region *Mexico*.

Ą	AutoSave 💽 Off	8 9 . C	÷		Book1 -	Excel		⊡ – ⊡ ×
Fi	le Home I	nsert Page La	ayout Formula	s Data Revi	iew View	Developer Help	p PowerExcel 🔎 Searc	h 🖻 🖓
N	·	Refresh Show Sidebar	Find Options OLA Fx	Connections Op PowerExcel C	Cache		About Change owerExcel Password Help	~
A9				• : ×	√ f×		e(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4	
A5							۳(۶۵۶۱,۶۵۶۷,۶۵۶۵,۶۵۶۵,۶۵۶	·· >E > 4, > D > J . > E > J, > D > 0 . > E > 0,
	A	В	С	D	E	F A	PowerExcel	- x
	Database:	USING_OLATIO	N				PowerExcel	
	Cube:	SALES					Database	
	Dimensions:	Filter	SALES Measure		Amount		USING_OLATION	✓ ¥.
4		Filter	Version	Members	Budget		Cube	
5		Filter	Region	Members	Mexico		SALES	~ 🎽
6		Column	Account	Range	\$B\$10:\$E\$10		T Filters	
7		Row	Month	Range	\$A\$11:\$A\$26	5	🖾 SALES Measure: Amoun	t
8							L: Version: Budget	
	OLAPivotTable						C Region: Mexico	
10		Sales	Cost of Sales	Margin	Margin Pcnt			
	January	0				0		
	February	0		_		0	Sa Columns	
	March	0	-			0	4 Account: Sales,Cost of S	Sales Margin Margin Pont
	1st Quarter	0				0		
	April	0	-			0		
	May	0	-	-		0		
	June	0	-			0	Rows	
	2nd Quarter	0		-		0	4 Month: January,Februa	ry, March, 1st Quarter, April, May, June, 2
	July	0		_		0		
	August	0				0		
	September 3rd Quarter	0				0	<	>
	october	0				0	Options	PowerExcel Slice
	November	0	-	_		0	Constrain Empty Rows	PivotTable
	December	0	-			0	Delete Removed Rows	Read/Write Formulas
	4th Quarter	0	-	-		0	Expandable Members	
20	Han Quarter	0	0	0			Expandable Members	O Power Query
27							Insert Into	
29							🔘 New Workbook 🛛 🖲) Current Worksheet
30							O New Worksheet	Location: 👫
31								
	< → Sh	eet1 +		: •			Update	
Rea	dy Calculate 🖁					🔓 Disp	olay Settings 🔠 🗉	······································

Assume that there are new transactional records added into the SQL database relational source—for example, Budget figures for *Mexico* in *January* and *February*, as in the succeeding screenshot (see highlighted records):

		- 🤇 - 🕅 🛛			👻 🧔 Generic Deb	ugger 👻 🍟
ect Explorer 👻 👎 🗙		- dbo.Factdata ⊰¤				
mect = 🛱 🎽 🔲 🍸 🖒 🧀	dim_Actua_VS	-	dim_Regions	dim_Months	Amount	
😑 📄 USING_OLATION 🔺	Budget	Sales	Brazil	March	1844	
🗉 📕 Database Diagrams	Budget	Cost of Sales	Brazil	March	737	
Image: Imag	Budget	Sales	Brazil	April	1948	
 System Tables FileTables 	Budget	Cost of Sales	Brazil	April	779	
External Tables	Budget	Sales	Brazil	May	2059	
🗉 📕 Graph Tables	Budget	Cost of Sales	Brazil	May	824	
dbo.dim_Accounts	Budget	Sales	Brazil	June	2177	
dbo.dim_Actual_vs_Budget	Budget	Cost of Sales	Brazil	June	871	
⊞ dbo.dim_Months ⊞ dbo.dim_Regions	Budget	Sales	Brazil	July	2301	
 III dbo.dim_Regions III dbo.Factdata 	Budget	Sales	Brazil	August	2432	
B B dbo.OLATION_USING_OLATION	Budget	Sales	Brazil	September	2571	
🗉 🎹 dbo.Product	Budget	Cost of Sales	Brazil	July	920	
dbo.Product_Factdata	Budget	Cost of Sales	Brazil	August	973	
 I doo.Product_hierarchy Views 	Budget	Cost of Sales	Brazil	September	1028	
External Resources	Budget	Sales	Brazil	October	1185	
🗉 🛑 Synonyms	Budget	Cost of Sales	Brazil	October	474	
🗉 📕 Programmability	Budget	Sales	Brazil	November	688	
🗉 🛑 Service Broker	Budget	Cost of Sales	Brazil	November	275	
🕢 📠 Storage	-	Sales	Brazil	December	1659	
🗉 📁 Security 🗑 🚍 USING OLATION testdoc	Budget					
USING_OLATION_testdoc Security	Budget	Cost of Sales	Brazil	December	664	
🗑 📕 Server Objects	Budget	Sales	Mexico	January	10000	
🗉 📻 Replication	Budget	Cost of Sales	Mexico	January	8888	
🗉 🛑 PolyBase	Budget	Sales	Mexico	February	20000	
🗉 🧮 Always On High Availability	Budget	Cost of Sales	Mexico	February	9999	
Integration Services Catalogs	NULL	NULL	NULL	NULL	NULL	
🗃 🛑 Integration Services Catalogs a 🛃 SQL Server Agent						
XE Profiler	[4 €] 1 of	66 🕨 🔰 🌬 🛛				

Once you refresh or update the Olation model, the PowerExcel report (or other front-end client) will show the values reflected there—as in the following image:

AS	9			• : ×	√ f _x	=OLAPivotTa	able(\$B	3\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:	\$E\$4,\$B\$5:\$E\$5,\$B\$6:\$E\$6,	۲
	А	В	С	D	E	F				
1	Database:	USING_OLATIO	N					PowerExcel	- × ×	ς
2	Cube:	SALES						Database		
3	Dimensions:	Filter	SALES Measure	Members	Amount			USING OLATION	~ 2	4
4		Filter	Version	Members	Budget			Cube		••
5		Filter	Region	Members	Mexico			SALES	~ 2	5
6		Column	Account	Range	\$B\$10:\$E\$10			T Filters		-
7		Row	Month	Range	\$A\$11:\$A\$26			CALES Measure: Amount		
8								∠ Version: Budget		
9	OLAPivotTable							忆 Region: Mexico		
10		Sales	Cost of Sales	Margin	Margin Pcnt					
11	January	10000	8888	1112		0				
12	February	20000	9999	10001		0		-		_
13	March	0	0	0		0		Columns		_
14	1st Quarter	30000	18887	11113		0		☑ Account: Sales,Cost of S	ales, Margin, Margin Pcnt	
15	April	0	0	0		0				
16	May	0	0	0		0				
17	June	0	0	0		0		🚰 Rows		
18	2nd Quarter	0	0	0		0				_
19	July	0	0	0		0		🖾 Month: January,February	y, March, 1st Quarter, April, May, June,	.21
20	August	0	0	0		0				
21	September	0	0	0		0				
22	3rd Quarter	0	0	0		0		<	1	>
23	October	0	0	0		0		Options	PowerExcel Slice	
24	November	0	0	0		0		Constrain Empty Rows	PivotTable	
25	December	0	0	0		0		Delete Removed Rows	Read/Write Formulas	
26	4th Quarter	0	0	0		0		Expandable Members	O Power Query	

6.2. Write-back (for Planning) to the Source Relational Database

For this exercise we will enter data in a dynamic front end—PowerExcel, in this example—and see how it is written back to the source USING_OLATION SQL relational database.

Assume that you are a Salesperson in Brazil and you need to enter Budget figures based on Prior Year Actuals data; the objective is to increase our Sales by 10%.

Additionally, we want to assess our *Prior Year Actual performance* and get the Average Cost Ratio, and then increase it by 2% to cover the expected increase in raw materials. We will then multiply our Actuals data by the new Cost Ratio to get our new Cost of Sales Budget figures.

First create a formula in Excel to calculate the Cost Ratio by month. Go to Column G and define a formula where:



IMPORTANT: For this exercise, we will make use of the PowerExcel Slice report we have created to show customer calculations that check the results from the Olation model. It is worth noting that you can always remove these calculations that are added in Excel.

1. Go to cell **G11** and enter the formula **=C11/B11** whereby:

 $\ensuremath{\textbf{C11}}\xspace$ – is the Cost of Sales for the month of January; and

 $\ensuremath{\textbf{B11}}\xspace$ – is the Sales for the month of January

- 2. Copy this formula across all individual months *February* to *December*. Delete formula for aggregate Members: Quarters and Total Quarters.
- 3. Press the **F9** key to refresh values.

The Excel report will appear as follows:

,	AutoSave 💽 Off	B 9- 9-	÷	I	Book2 - Excel 🤇	5				Ŧ	-		×
Fi	ile Home I	insert Page La	yout Formulas	Data Revie	w View De	veloper	Help Pov	verExcel	𝒫 Searce	:h	andh	Ŕ	₽
N		e Refresh Show Sidebar	Find Options OLA Fx	Connections Op PowerExcel C	Cache	Edit Dimensi Dimensi		Change el Password telp	4				^
G1	1			• : ×	$\checkmark f_x$	=C11/B	11						~
	А	В	С	D	E	F	G	н	T	J	К	L	
1	Database:	USING_OLATIO	N										
2	Cube:	SALES											
3	Dimensions:	Filter	SALES Measure	Members	Amount								
4		Filter	Version	Members	Actual								
5		Filter	Region	Members	Brazil								
6		Column	Account	Range	\$B\$10:\$E\$10								
7		Row	Month	Range	\$A\$11:\$A\$27								
8							Cost Ratio						
9	OLAPivotTable						=COS/Sales						_
10		Sales	Cost of Sales	Margin	Margin Pcnt								
11	January	602	500	102		0	83%						
12	February	1586	500	1086		0	32%						
13	March	1676	500	1176		0	30%						
14	1st Quarter	3864	1500	2364		0							
15	April	1771	500	1271		0	28%						
16	May	1872	500	1372		0	27%						
17	June	1979	500	1479		0	25%						
18	2nd Quarter	5622	1500	4122		0							
19	July	2092	500	1592		0	24%						
20	August	2211	500	1711		0	23%						
	September	2337		1837		0	21%						_
	3rd Quarter	6640	1500	5140		0						_	_
	October	1077		577		0	46%						_
	November	625		125		0	80%						
	December	1508		1008		0	33%						_
	4th Quarter	3210		1710		0							_
	Total Quarter	19336	6000	13336		0							
28													
29													
		heet1 +					: •						Þ
Rea	idy Calculate						💁 Display Setti	ngs 🏢		── - ──		+ 10	30%

4. Add the Cost Ratio for all months and divide it by 12. This will give you an Average Cost Ratio of about 38%.

TIP: There is an average value found that can be found at the bottom of the Excel window (circled in the following image) that averages the highlighted values.

	A	В	C	D	E	F	G	н	1 I	J	K	L	
1	Database:	USING_OLATI	DN										٦.
2	Cube:	SALES											
3	Dimensions:	Filter	SALES Measure	Members	Amount								
4		Filter	Version	Members	Actual								
5		Filter	Region	Members	Brazil								
б		Column	Account	Range	\$B\$10:\$E\$10								
7		Row	Month	Range	\$A\$11:\$A\$27								
8							Cost Ratio						Т
9	OLAPivotTable						=COS/Sales						
10		Sales	Cost of Sales	Margin	Margin Pcnt								
11	January	60	2 500	102		0	83%						
12	February	158	6 500	1086		0	32%						
13	March	167	6 500	1176		0	30%						Т
14	1st Quarter	386	4 1500	2364		0							
15	April	177	1 500	1271		0	28%						
16	May	187	2 500	1372		0	27%						
17	June	197	9 500	1479		0	25%						
18	2nd Quarter	562	2 1500	4122		0							
19	July	209	2 500	1592		0	24%						
20	August	221	1 500	1711		0	23%						
21	September	233	7 500	1837		0	21%						
22	3rd Quarter	664	0 1500	5140		0							
23	October	107	7 500	577		0	46%						
24	November	62	5 500	125		0	80%						
25	December	150	8 500	1008		0	33%						
26	4th Quarter	321	0 1500	1710		0							
27	Total Quarter	1933	6 6000	13336		0							
28													
29													
	< > SI	neet1 (+)				: 4						F
Re:	idy Calculate			Average: 38%	iount: 12 Sum:	452%	S Display Setti	ngs 🗄		四		+ 100	7%
AC.	my calculate			Average, 50%	Sund 12 Sund	43270	28 Disbiak setti	iya H				100	"

- 5. Now that we have identified the Average Cost of 38%, we will increase it by 2%, i.e., making budgeted Cost of Sales is 40%. Therefore, **type 40%** in **cell J7**. We can now delete data or hide column G.
- 6. Next, as we intend to calculate new targeted Sales figures (110% of Last Year's Actual Sales): copy the *Sales values* for individual months and **Paste Special as Values** to Column H. Again, remove values for aggregate months.
- In Column I create a formula that will calculate the new target Sales Amount per month, which is 110% of each month Sales values.
 To do this:

To do this:

- Enter the formula: =H11*1.1 in cell I11.
- Copy and paste the formula to cells **I12 to I25**.
- Press F9 to refresh values.
- In **Column J**, create a formula that will compute for the budgeted Cost of Sales by multiplying **New Target Sales** values (in column I) to 40% (cell J7).
- Enter the formula: **=I11*\$J\$7** in **J11**. (Note: Make J7 an absolute reference, as indicated.) To summarize:

I11 - is the New Target Sales for the month of January; and

\$J\$7 – is the cell containing the New Cost Ratio.

- Copy and paste the formula to cells **J12 to J25**.
- Press F9 to refresh values—see results in the following image:

Fi (N	ew Open Save		ayout Formula Find Options OLA Fx	Connections Op	Cache	Develo L Edit Dimensi	About PowerExcel		rch 🖻	न
	P	owerExcel Slice		PowerExcel C		Dimensi		p		
J1:	1			• E ×	$\checkmark f_x$	= 11*\$J	\$7			
	А	В	С	D	E	F	н	I.	J	к
1	Database:	USING_OLATIO	N							
2	Cube:	SALES								
3	Dimensions:	Filter	SALES Measure	Members	Amount					
4			Version	Members	Actual					
5		Filter	Region	Members	Brazil					
6			Account	Range	\$B\$10:\$E\$10				New Cost Ratio	
7		Row	Month	Range	\$A\$11:\$A\$27				40 %	
8							Actual Sales	New Target Sales	New Cost of Sales	
	OLAPivotTable							=Actual Sales * 1.1	=New Target Sales * 0.4	
10		Sales	Cost of Sales	Margin	Margin Pcnt					
_	January	602	500	102		0	602	662	265	
12	February	1586	500	1086		0	1586	1745	698	
13	March	1676	500	1176		0	1676	1844	737	
	1st Quarter	3864	1500	2364		0				
15	April	1771	500	1271		0	1771	1948	779	
16	May	1872	500	1372		0	1872	2059	824	
17	June	1979	500	1479		0	1979	2177	871	
18	2nd Quarter	5622	1500	4122		0				
	July	2092	500	1592		0	2092	2301	920	
	August	2211	500	1711		0	2211	2432	973	
	September	2337	500	1837		0	2337	2571	1028	
22	3rd Quarter	6640	1500	5140		0				
23	October	1077	500	577		0	1077	1185	474	
	November	625	500	125		0	625	688	275	
25	December	1508	500	1008		0	1508	1659	664	
26	4th Quarter	3210	1500	1710		0				
	Total Quarter	19336	6000	13336		0				
28						_				
29										
	 Sł 	neet1 +					: 4			Þ
Cali	:ulate 🔚						😹 Display	Settings 🏢 🗉	P	+ 100%

- 8. You can apply Excel formatting to this worksheet so you can reuse it in the future. For this example, save this as **USING_OLATION Budget Entry Template**.
- To make the spreadsheet true to its name, we want it to show Budget values: to do this, doubleclick on cell E4 and change the *Version* to Budget. Press the F9 key to refresh values. We now have the Budget Sales and Budget Cost of Sales figures for Brazil—all *zero* values at this point.
- 10. Next, to enter those values into the USING_OLATION database: you can manually enter the numbers, but an easier method is to copy the computed *New Target Sales* and *Budgeted Cost of Sales* figures.
 - Highlight the range of cells that contains the new values (in the example, **I11 to J25**), copy the values and then put your cursor on cell **B11** then select **Paste as Values**.

11. Press **F9** to refresh values.

Upon refreshing the values, notice how, dynamically, the figures appear and also aggregate for *Quarters, Total Quarter* and *Margin.*

	AutoSave 💽 Off	89.6.		USING_OL	ATION Bugdet Ent	ry Tem	plate.xlsx - Excel		m –	o x
Fi	ile Home	Insert Page L	ayout Formul	as Data Re	view View	Develo	oper Help	PowerExcel 🔎 Sea	rch	r P
	aste 🗳	bri • 1 I ∐ • ∰ • Font	1 • A A • <u>A</u> •			E I	Conditional Form: Format as Table * Cell Styles * Styles	-	deas	
В1	.1			• : ×	√ <i>f</i> x 6	62.2				~
	А	В	С	D	E	F	Н	L	J	К
1	Database:	USING_OLATIO	N							
2	Cube:	SALES								
3	Dimensions:	Filter	SALES Measure	Members	Amount					
4		Filter	Version	Members	Budget					
5		Filter	Region	Members	Brazil					
6		Column	Account	Range	\$B\$10:\$E\$10				New Cost Ra	tio
7		Row	Month	Range	\$A\$11:\$A\$27				4	0%
8							Actual Sales	New Target Sales	New Cost of Sales	
9	OLAPivotTable							=Actual Sales * 1.1	=New Target Sales * (0.4
10		Sales	Cost of Sales	Margin	Margin Pcnt					
11	January	662	265	397	0		602	662		265
12	February	1745	698	1047	0		1586	1745	6	598
13	March	1844	737	1106	0		1676	1844	-	737
14	1st Quarter	4250	1700	2550	0					
15	April	1948	779	1169	0		1771	1948	-	779
16	May	2059	824	1236	0		1872	2059	6	324
17	June	2177	871	1306	0		1979	2177	6	371
18	2nd Quarter	6184	2474	3711	0					
19	July	2301	920	1381	0		2092	2301	9	920
20	August	2432	973	1459	0		2211	2432	9	973
21	September	2571	1028	1542	0		2337	2571	10	028
22	3rd Quarter	7304	2922	4382	0					
23	October	1185	474	711	0		1077	1185		474
24	November	688	275	413	0		625	688		275
25	December	1659	664	995	0		1508	1659	6	564
26	4th Quarter	3531	1412	2119	0					
27	Total Quarter	21270	8508	12762	0					
28										
••	 Sł 	neet1 +					: 4			
Cal	culate 🔝			Average: 18	320 Count: 30	Sum: 54	4611 🛛 🌄 Display	Settings 🏢 🗉	· · · · · · · · · · · · · · · · · · ·	+ 100%

12. Next, we can see that the real-time result of the Budget entries have propagated dynamically through the application—next, utilize SQL Server Management Studio to access the newly added data (which should be reflected in the *dbo.Factdata* SQL table) of the USING_OLATION SQL database. See below, showing the new *Budget Sales, Cost of Sales,* and *Margin* values for *Brazil* are added into the fact data table.

	7 × E0	-4		dbo.Factdata +¤			•
nect• 🛱 🏋 🗏 🝸 🖒 -ሎ			dim_Actua_VS	dim_Accounts	dim_Regions	dim_Months	Amount
🕀 📄 SmallDemo	^		Actual	Sales	Brazil	June	1979
H H TE			Actual	Sales	Brazil	July	2092
🕀 🗑 USABudget 🖃 🗑 USING_OLATION			Actual	Sales	Brazil	August	2211
🗃 📕 Database Diagrams		_	Actual	Sales	Brazil	September	2337
🖃 💻 Tables		1	Budget	Cost of Sales	Brazil	January	265
😠 🛑 System Tables			Budget	Sales	Brazil	February	1745
🕀 💼 FileTables			Budget	Cost of Sales	Brazil	February	698
🕢 📻 External Tables			Budget	Sales	Brazil	March	1844
🕢 📻 Graph Tables 🕢 🖬 🖬 dbo.dim_Accounts			Budget	Cost of Sales	Brazil	March	737
i iii aborann_Accounts i iii aborann_Accual_vs_Budget		Т	Budget	Sales	Brazil	April	1948
		Т	Budget	Cost of Sales	Brazil	April	779
🕀 🎹 dbo.dim_Regions		Т	Budget	Sales	Brazil	May	2059
🗄 🌐 dbo.Factdata		Т	Budget	Cost of Sales	Brazil	May	824
Image:		Т	Budget	Sales	Brazil	June	2177
🗑 📕 External Resources		Т	Budget	Cost of Sales	Brazil	June	871
🖅 🚃 Synonyms		Т	Budget	Sales	Brazil	July	2301
😠 🛑 Programmability	11	Т	Budget	Sales	Brazil	Auqust	2432
🕀 📕 Service Broker		Т	Budget	Sales	Brazil	September	2571
		Т	Budget	Cost of Sales	Brazil	July	920
		Т	Budget	Cost of Sales	Brazil	Auqust	973
j 📕 Security		T	Budget	Cost of Sales	Brazil	September	1028
3 💼 Server Objects		T	Budget	Sales	Brazil	October	1185
) 🛑 Replication		T	Budget	Cost of Sales	Brazil	October	474
3 📁 PolyBase 3 📁 Always On High Availability		T	Budget	Sales	Brazil	November	688
) 📕 Management		T	Budget	Cost of Sales	Brazil	November	275
🛛 🛑 Integration Services Catalogs			Budget	Sales	Brazil	December	1659
3 🛃 SQL Server Agent		t	Budget	Cost of Sales	Brazil	December	664
📧 🗶 Profiler		1	NULL	NULL	NULL	NULL	NULL
		1					

7. Define Formula in OLATION®

We will next cover how to define formulas within an Olation Cube. For this exercise, we will create a simple Internal Cube Formula that computes the value of **Margin Pcnt** (abbreviation for *Margin Percent*).

When writing a Cube formula in Olation, be aware of the correct formula syntax and terminologies:

OLATION

Left Hand Side Expression (LHS) = Right Hand Side Expression (RHS);

Parameters explained below:

Parameter	Description
Left Hand Side or LHS	The defined expression or range where the calculation "goes"
Right Hand Side or RHS	Contains a constant value, a mathematical expression, and may include Cube Functions and a Cube Reference
Semi-colon (;)	Indicates the end of a particular Formula

It is also important to understand the SYNTAX for each side (LHS and RHS).

Left Hand Side (LHS) or RANGE Reference

Qualifier{"Dimension Name.Member Name"}

Qualifier	Determines in what areas or intersections within the Cube the value be returned.
	 The different types of Qualifiers are: All And {} Aggregates And {} Details And {}
{"Dimension Name.Member Name"}	This indicates the specific Dimension and Member that will be populated by the resulting values.

The QUALIFIERS

Below is a brief explanation of what the qualifiers are:

• All And {} - This qualifier indicates that LHS Range will apply to all intersections and cells within the cube, both for Aggregate and Detail level cells.

- Aggregates And {} This qualifier indicates that LHS Range will apply only to intersections or cells that are on an Aggregate level.
- Details And {} This qualifier indicates that LHS Range will apply only to intersections or cells which are on a Detail level, or the lowest level Members.

Right Hand Side (RHS) or CUBE Reference (where applicable)

	"Cube Name".["Dimension Name.Member Name"]				
"Cube Name".		This specifies the source Cube where the data or range from the RHS are coming from.			
		For Internal Cube Formulas, the Cube name is the same as the name of the active Cube. For cross-cube formulas, the Cube name is that of a different Cube.			
{"Dimension N	lame.Member Name"}	This specifies the specific Dimension and Member or range from within the source Cube.			

Note: You can write a COMMENT within the Formula Editor, and these 'Comments' will be skipped or not read by Olation when parsing through the formula syntax. To write a comment, use the symbols:

- *I** marks the start of the comment
- */ marks the end of the comment
- *II* indicates that the whole line will be read as a comment

7.1. Define the Cube Formula Statement

Note: As earlier mentioned, we will define an internal Cube formula. We will add this formula to the SALES Cube of the USING_OLATION database. The formula to compute for Margin Pcnt is:

MARGIN PCNT = Margin / Sales

To define the formula:

- 1. Go back to the OLATION[®] Studio and, to edit the **SALES** cube, double-click on it. **Note:** Alternatively, you can also **right-click on the appropriate Cube** and select **Edit Cube**.
- 2. Go to the Formulas Tab (shown in the next image, with the Comment //Compute MARGIN PCNT)



3. Write the LHS Expression.

You can make use of the command buttons along the Cube Tab of the Olation ribbon in writing your formula: click the **Formula button** and select **All And {}** as the qualifier; double-click on **Account** as the Dimension, and; double-click on **Margin Pcnt** as Member.



4. Move your cursor after the **equals symbol** (=). Here you will define the RHS expression of your formula. Recall that the formula for computing *Margin Pcnt* is: **Margin / Sales**.

- 5. Write the **RHS Expression**. Again, you can make use of the command buttons along the Cube Tab, in this case the Cube button:
 - (On the **Cube button**) double-click on **SALES** as the Cube; double-click to select **Account** as the Dimension and double-click to select **Margin** as Member. This will be the numerator.
 - Outside the rightmost square bracket, enter the **forward slash symbol** (*I*) to denote division as the operation.
 - On the **Cube** button select **SALES** as the Cube; double-click to select **Account** as the Dimension and double-click to select **Sales** as Member. This will be our denominator.
- 6. Outside the rightmost square bracket, type a **semi-colon** (;) to indicate the end of the formula. The formula will appear as follows:



- 7. Click the **Check** or **Check Syntax button** (located along the Cube Tab of the Olation ribbon) to verify that the formula is free from syntax errors. You should get a prompt that says, 'The formula syntax is correct'.
- 8. Save the **Cube**.

7.2. Checking the Cube Formula Results

Having defined the Cube formula for the Account *Margin Pcnt*, we can check the results in any front end that provides a view of that account.

[Note: Make sure to refresh or recalculate your Slice or Excel view before seeing results.]

In the POWEREXCEL SLICE (see following image)

Fi	ile Home	Insert Page Layo	out Formulas	Data Review	View Developer	Help PowerExcel 🔎 Search 🔄	P
N) 🗁 🖳 lew Open Save	Refresh Show F Sidebar Ol	ind Options Con	nections Options C	lear Edit Dimension	About Change PowerExcel Password	
	P	owerExcel Slice		PowerExcel Connectio	n Dimension	Help	^
AS)		Ψ.	1 × 🗸	fx =OLAPivotT	able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E\$4,\$B\$5:\$E\$5,\$B\$6:\$E\$6,\$B\$7:	~
	А	в	с	D	E	F	
1		USING OLATION	_	_		PowerExcel	×
2	Cube:	SALES				Database	
3	Dimensions:	Filter	SALES Measure	Members	Amount		¥.
4		Filter	Version	Members	Actual	Cube	* &:
5		Filter	Region	Members	United States	SALES	X
6		Column	Account	Range	\$B\$10:\$E\$10	T Filters	
7		Row	Month	Range	\$A\$11:\$A\$27	Filters L' SALES Measure: Amount	
8						2. Version: Actual	
9	OLAPivotTable				\frown	🖾 Region: United States	
10		Sales	Cost of Sales	Margin	Margin Pcnt		
11	January	5000	840	4160	83%		
12	February	5285	960	4325	82%	Columns	
13	March	5586	580	5006	90%	L'Account: Sales,Cost of Sales,Margin,Margin Pont	_
14	1st Quarter	15871	2380	13491	85%	A Account Sates, cost of Sates, margin, margin rate	
15	April	5905	700	5205	88%		
16	May	6241	660	5581	89%		
17	June	6597	700	5891	89%	🔂 Rows	
18	2nd Quarter	18743	2060	16683	89%	レージョン Month: January,February,March,1st Quarter,April,May,Jur	ne.2i
19	July	6973	1020	5953	85%		•
	August	7370	800	6570	89%		
	September	7791	860	6931	89%	<	>
	3rd Quarter	22134	2680	19454	88%	Options PowerExcel Slice	
	October	3591	0	3591	100%	Constrain Empty Rows PivotTable	
	November	4453	0		100%	Delete Removed Rows Read/Write Formulas	
	December	5027		5021	100%	Expandable Members O Power Query	
	4th Quarter	13071	0		100%		
	Total Quarter	69819	7120	62699	90%	Insert Into	
28						New Workbook O New Workbook O Current Worksheet	
29						O New Worksheet Location: \$4\$1	<u>-</u>
30 31					T		
	 Sł 	eet1 +	· · · · · ·	÷ •		Update	
Cal	culate 🔝					🙀 Display Settings 🔠 🗐 🖳 – — — + 10	00%

[Note: You can keep this Slice open or save it as an Excel file to your desktop, or wherever you wish. Further on in this manual we will want to use it to make a comparison to a Slice that we create in a new Cube.]

		PowerOLAP - [SAL	ES : Comparative Margin	per Quarter]		- 🗆	×
Home Model Slice	Data Tools					🗟+ Style 👻 🕢	- 7
+ C Silce Si	Manage DRDC	Paste Cut Copy	Clear Fill Fill Fill Right	Graph Constrain Slice T Options	hrough 🗾		inge ons
Filter:		Sales	Cost of Sales	+ Margin	Margin Pont		_
SALES Measure: Amount	January	5000.00	840.00	4160.00	0.83		
Version: Actual	February	5285.00	960.00	4325.00	0.82		
C Region: United States	March	5586.00	580.00	5006.00	0.90		
	+ 1st Quarter	15871.00	2380.00	13491.00	0.85		
	April	5905.00	700.00	5205.00	0.88		
	May	6241.00	660.00	5581.00	0.89		
	June	6597.00	700.00	5897.00	0.89		
	+ 2nd Quarter	18743.00	2060.00	16683.00	0.89		
Column Labels:	July	6973.00	1020.00	5953.00	0.85		
Account: Subset	August	7370.00	800.00	6570.00	0.89		
	September	7791.00	860.00	6931.00	0.89		
	+ 3rd Quarter	22134.00	2680.00	19454.00	0.88		
	October	3591.00	0.00	3591.00	1.00		
Devel also have	November	4453.00	0.00	4453.00	1.00		
Row Labels: V. Month: Subset	December	5027.00	0.00	5027.00	1.00		
	+ 4th Quarter	13071.00	0.00	13071.00	1.00		
	+ Total Quarter	69819.00	7120.00	62699.00	0.90		
or Help, press F1			loc	alhost	OLATION_Report		

In the POWEROLAP SLICE (see following image)

A screenshot of the Comparative Margin per Quarter Slice (the PowerOLAP Slice we created in earlier topics)

In the POWEROLAP SLICE TO EXCEL

Fi	le Home	Insert Page Layout	Formulas Data	Review View	Developer Help	PowerExce	Powe	rolap D	Search	ß	₽
	unch Save odeler Database PowerOLAP	Cut Copy OLAP Cut Cipboard	Cube Dimension	Select New Members Slice	Formula Outline	Print Report Set Print	Execute	PowerOLAP Portal files PowerOLAP P.	Import T	Settings *	
A	}			X V fx	=OLAPTable(\$B	\$1.\$B\$2.\$C	3.\$C\$4.\$0	C\$5.B9:E9.A1):A26.\$B\$1	.0)	~
	A	В	с	D	E	F	G	н			
1	Database:	USING OLATION	L.	U U	E	F	6	п	1	J	
	Cube:	SALES									
	Page:	SALES Measure	Amount								
3 4	rage.	Version	Actual								_
4 5		Region	United States								-
6		Account	Along Columns								-
7		Month	Along Rows								
8	OLAPTable	Month	Along Now 2								
9	ob il lubic	Sales	Cost of Sales	Margin	Margin Pcnt						
	January	5000		4160							
	February	5285		4325							
	March	5586	5 580	5006							
13	1st Quarter	1587	L 2380	13491	0.85						
	April	5905		5205							
	May	6243	L 660	5581	0.89						
	June	659	7 700	5897	0.89						
17	2nd Quarter	18743	3 2060	16683	0.89						
18	July	6973	3 1020	5953	0.85						
19	August	7370	008 0	6570	0.89						
20	September	7793	L 860	6931	0.89						
21	3rd Quarter	22134	4 2680	19454	0.88						
22	October	3593	L 0	3591	1.00	1					
23	November	4453	3 0	4453	1.00						
24	December	502	7 0	5027	1.00						
25	4th Quarter	1307:	L 0	13071	1.00						
26	Total Quarter	69819	9 7120	62699	0.90						
27											
28											

8. Create the PRODUCT_SALES Cube

In this section, the objective will be to create a 5-dimensional *PRODUCT_SALES* Cube composed of the four relational-table – based Dimensions created previously (*Account, Version, Month, Region*) and a new *Product* dimension. The new Dimension will *not* be based on a pre-existing relational table, pointing up Olation's ability to add new/independent "meta data" logic to a model; we will reference this "non-relational – based" dimension as being "multidimensional" in nature. [It is worth noting that, in the end—as will be shown—the creation of the "multidimensional" data will result Olation creating relational tables; so, in effect, these dimensions do have a consequent basis in relational table logic.]

We will also demonstrate the use of a Cross-Cube Formula in the second Cube (PRODUCT_SALES) that will pull in fact data from the Cube created previously (*SALES* Cube).

Important: Please note that specific figures may be different from the data set you are working on. This exercise is intended to server as a guide to help you understand how to use OLATION so that you can work with your own data set.

8.1. **Pre-Work: Prepare the Product Factdata SQL Table**

Before proceeding to create the new *PRODUCT_SALES* Cube, we will first need to prepare a necessary SQL table, i.e., a *dbo.Product_Factdata* SQL table. We need this table to be able to define the relationship between all our Dimension and a Measures table.

Note: If the Product_Factdata table is not yet created into the USING_OLATION SQL Database, you need to go to the SQL Server Management Studio and create that table from there. Then make sure to modify this factdata table to include an additional *Products column*.

8.1.1. Create the Product_Factdata SQL Table in SQL Server Studio

To create the Product_Factdata SQL table (a copy of the Factdata SQL table):

- 1. Launch SQL Server Management Studio.
- 2. In SQL Server's Object Explorer pane, expand **Databases** and locate the source Olation database, i.e., **USING_OLATION**.
- 3. Expand the USING_OLATION database, expand Tables and locate the dbo.Factdata table.
- 4. Right-click on the **dbo.Factdata** table, select **Script Table as**, **CREATE To** and choose **New Query Editor Window** (see next image).

😡 Solution1 - Microsoft SQL Ser	ver Management Studio			Quick Lau	unch			
File Edit View Project Debug Tools Window Help								
🛛 G + O 🛛 🏠 + 🎦 - 🚔 💾	📲 🔎 New Query 🔎 😥 🕅 🕅	☆ 太 口 台 り・ペ・	81	-				
Object Explorer		▼ ₽ ×						
Connect 🕶 🌹 🍟 🔳 🝸 🖒 🚸	L							
🖽 📄 USABudget		A						
🕢 📻 Database Diagran 🖃 🛑 Tables	ns							
🖃 🛑 Tables 😠 💼 System Table	e							
🕀 📕 FileTables	,							
🐨 💻 External Table	25							
🔢 🛑 Graph Tables								
🕢 🎛 🕀 🕀								
🕀 🎹 dbo.dim_Actu								
🕢 🎹 dbo.dim_Mor								
😥 🌐 dbo.dim_Reg 😥 🎹 dbo.Factdata								
🗄 🎞 dbo.OLATI	New Table							
🕁 💻 Views	Design							
🕀 📁 External Resou	Select Top 1000 Rows							
🕀 🛑 Synonyms	Edit Top 200 Rows							
🕢 📻 Programmabil 🕀 🛑 Service Broker	Script Table as	CREATE To	• 🗊	New Query Editor Window				
🕢 😥 📰 Service Broker 🛄 🕀 💼 Storage	View Dependencies	ALTER To						
😥 💼 Security	Memory Optimization Advisor	DROP To	1	File				
🕀 📕 Security		DROP And CREATE To	Ē	Clipboard				
🕀 🛑 Server Objects	Encrypt Columns		<u> </u>	Agent Job				
🕀 💼 Replication	Full-Text index	SELECT To						
⊕ 📁 PolyBase ⊛ 📁 Always On High Avail	Storage 🕨	INSERT To	•					
		UPDATE To	•					
	Stretch	DELETE To	•					
🗉 🛃 SQL Server Agent	Policies •	EXECUTE To						
🕀 街 XE Profiler	Facets	.						
	Start PowerShell							
Ready								
	Reports •							
	Rename							
	Delete							
	Refresh							
	Properties							
	rioperdes							

 In the New Query Editor Window that appears, edit the script by changing the name of the *dbo.Factdata* table to dbo.Product_Factdata as shown in the following image. The highlighted text in blue is typed in to rename the table name (see where arrow points).

OLATION

```
SQLQuery2.sql - EC..._OLATION (sa (60)) 🗠 🗶
     1
         USE [USING_OLATION]
     2
         GO
     3
         /****** Object: Table [dbo].[Factdata] Script Date: 4/28/2020 5:28:55 AM
     4
           *****/
         SET ANSI_NULLS ON
     5
     6
         60
     7
     8
         SET QUOTED IDENTIFIER ON
     9
         GO
    10
    11 CREATE TABLE [dbo].[Product_Factdata](
    12
             [dim_Actua_VS_Budget] [nvarchar](50) NOT NULL,
              [dim_Accounts] [nvarchar](50) NULL,
    13
    14
             [dim_Regions] [nvarchar](50) NULL,
    15
             [dim Months] [nvarchar](50) NULL,
    16
             [Amount] [decimal](18, 0) NULL
    17
        ) ON [PRIMARY]
    18
         60
    19
    20
    21
```

- Click Execute button or press the F5 key to execute the query. You should see a Message that says 'Commands completed successfully'.
- Refresh the USING_OLATION SQL database. You will see the *dbo.Product_Factdata* table in the Object Explorer pane of the SQL database, as shown in the following image.



8.1.2. Modify the Product_Factdata SQL Table to Include Products Column

There are several ways to add a new column into an SQL database table, commonly performed from SQL Server Management Studio.

However, in this exercise, we will demonstrate how to do it from the Olation Studio.

To do this:

- 1. Launch OLATION[®] Studio and open the correct database (USING_OLATION).
- In Database Explorer, expand the USING_OLATION database, expand Tables, then locate and right-click on the Product_Factdata table.
 Note: If this database was already open when you created the dbo.Product_Factdata table, simply refresh the Tables in Olation to ensure that all tables are shown.
- 3. Right-click on **Product_Factdata** and select **Edit Table**. A Table view of the *Product_Factdata* table will appear on the right.
- 4. Go to the last row and in the Column Name field, type **Products**, and for the Data Type click on the drop-down and select **int** (abbreviation for integer)—see next image.

→ −			Olation Studio			- 🗆 X
File Home Relational Multi-Dimensional	Table					Style 👻 🥅 MAX 🕻
8 2 2 0						
Save Refresh New Delete						
Table Dimension						
Database Explorer 🛛 📮 🗙	/ <u>0</u> P	roduct_Factdata				→ ×
Servers	Design					
 Iocalhost USING_OLATION 		Column Name	Data Type	Allow Nulls	2↓ □	
□ USING_OLATION □ 12 Dimensions		dim_Actua_VS_B	nvarchar		✓ Database Column	
		dim_Accounts	nvarchar		Catalog Name	USING_OLATION
Version		dim_Regions	nvarchar		Schema Name	dbo
12 Month		dim_Months	nvarchar		Table Name Column Name	Product_Factdata Products
🖾 Region		_			Allow Nulls	True
SALES Measure	<u> </u>	Amount	decimal		Identitifier Column	False
G Cubes		Products	int ~			
🗘 SALES	•				Catalog Name	
Tables					Catalog Name	
🗆 🎹 dbo						
iii dim_Accounts						
<pre>dim_Actual_vs_Budget</pre>						
iii dim_Months						
dim_Regions						
III Factdata						
Product_Factdata						
 Gueries External Data 						
Local Databases						

 Click the Save icon in the Table Tab of the Olation ribbon. The *Product_Factdata* table has now been modified to include the additional *Products column* that we need.

8.2. Create a Multidimensional Dimension - Product Dimension

As previously mentioned, we will use the same Dimensions we created in Exercise 1 for our new Cube, i.e., the Dimensions *Account*, *Version*, *Month* and *Region*. We will only need to create an additional Dimension, a multidimensional Dimension type, called **Product**.

To create the *Product* Dimension:

1. In OLATION, expand the USING_OLATION database; right-click on Dimensions and select New Dimension.

0	Ŧ					Olation Studio				
File	Home	Relational	Multi-Dimensional	Table	•					
	S	≌ ⊘								
Save	Refresh	New Delete								
٦	able	Dimension								
	ise Explorer		ά×	/0	Product_Factdata					
	Servers	aat		Desig	n Data					
		ING_OLATION			Column Name	Data Type	Allow Nulls	•	2↓ □	
		Dimensions	New Dimension			nvarchar		~	D'diabase column	
		Account Version	New Time Dimen			nvarchar			Catalog Name Schema Name	USII dbo
		Month	New Dimension F	rom R	Creates a new dime	nsion along with a nvarchar	new relational table t	o tra	ck your custom dimension Column Name	Proc Proc
		Kegion	Refresh			decimal			Allow Nulls	True
		SALES Mea	asure	•	Products	int 🗸		H	Identitifier Column	Fals
		Cubes								
		SALES		-				C	atalog Name	

2. In the **Create New Dimension** dialog, type **Product**, and click **Create**. The Dimension Editing window for the Product dimension appears on the right.

○	Olation Studio	– 🗆 X
File Home Relational Multi-Dimensional	Table	Style 👻 🥅 MAX 🔿
Image: Save Refresh Save Refresh Table New Delete Database Explorer Image: Save Refresh Servers Image: Servers Image: Save Refresh Servers Image: Servers Image: Servers	Product_Factdata Product Settings Aliases Properties Subsets Members Imension Name Display Name Dimension Type All Hierarchy Data Source Type Database Source Total Aggregate Hierarchies	Attributes Product Standard True Relational Table db.Product False Product_hierarchy
 Product_Factdata Views Queries External Data Bookmarks Security Local Databases 	Dimension Name Display name for this dimension	

8.2.1. Add Dimension Members

We will begin by adding the Members to the Product dimension. To add Members to the Product dimension:

1. Go to the Members Tab. Here, we will be adding the following Members:

MEMBERS:	
Current Pr	oduct
New Produ	uct 1
New Produ	uct 2
New Produ	uct 3
New Produ	uct 4

In the Member List pane (left-hand pane), click Add Member icon (# icon). In the textbox that appears, type the <member name>, i.e., Current Product, then press the Enter key. Note: You can keep adding Members by clicking on the Add Member icon and typing in the Member name; alternatively, you can press CTRL+Enter keys after adding a Member. When completed, the Members Tab will look as follows:

File Home Relational Hierarchy Severas Insert Database Explorer Image: Comparison of the stress of the str
Save Refresh New New Property New Subset Hierarchy Insert Database Explorer ↓ × I Servers I localhost I localhost > # 12 · · · F 4 ; I Collmensions ↓ Current Product 1 I Version ↓ New Product 2 I New Product 3 ↓ New Product 3 I SALES I mables I mables I mables
Alias Subset Hierarchy Insert Database Explorer Image: Comparison of the structure
Database Explorer
 Servers Settings Aliases Properties Subsets Members Attributes Settings Aliases Properties Quests Members Attributes Settings Aliases Properties Quests Members Attributes Settings Aliases Properties Subsets Attributes Settings Aliases Properties Subsets Attributes Settings Aliases Properties Subsets Attributes Settings Aliases Attrib
 Servers Settings Aliases Properties Subsets Members Attributes USING_OLATION <li< td=""></li<>
 I localhost USING_OLATION I Dimensions I Account I Account
Image: Solution of the second state
 Account Mew Product 1 New Product 2 Month New Product 3 Region New Product 4 SALES Measure SALES Measure SALES SALES Tables Month Here Here
Image: Second secon
Version Monto Region SALES Measure SALES SALES SALES Alter
Implementation
Image: A kegion Image: A kegion Image: B SALES Measure Image: P SALES Image: P SAL
 □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
Image: Second
□ III Tables □ III dbo
🗆 🎹 dbo
iii dim Accounts
III dim_Actual_vs_Budget
III dim_Months
III dim_Regions
III Factdata
III Product_Factdata
⊕ 🔯 Views
Z External Data
Becurity
Local Databases

Next, you will define additional Dimension attributes, namely:

ALIAS

You will create an Alias Group called Codes, to assign Product Codes for each Member.

PROPERTIES

You will classify each product item according to whether it is sold in a specific season (Season) or whether it is sold regularly, regardless of season (All-year-round).

• SUBSET

You will create a Subset group that includes only New Products.

8.2.2. Define Aliases

- 1. In the Dimension Editing Window for *Product*, go to the **Aliases Tab**. **Note:** Before you can define Aliases, you must first create the Alias Group.
- 2. Right-click on the Aliases window and select New Alias.
- 3. In the Create New Alias dialog, type the <alias group name>-in this example, CODE.
- 4. Click Create. The Alias group CODE appears in this window, as in the following image.



Next, define Aliases and assign them to the respective Members in the Product Dimension.

Note: It is important to note that Aliases, like Members, must be unique within the Dimension. So, once you assign an Alias to a Member, you cannot assign it to any other Member within that Dimension.

- 5. Go to the Attributes Tab.
- 6. In the column *CODE*, create the Aliases for each Member, as shown in the following image:

○	Olation Studio	– 🗆 X
File Home Relational Multi-Dimensional	Hierarchy	Style 👻 🥅 MAX 🔿
Save Refresh New New Property New Alias Subset		
Hierarchy Insert		
Database Explorer 🛛 📮 🗙	Roduct_Factdata	- ×
Servers Contemporation Servers Servers Servers Servers Servers Servers Servers Servers Servers Servers Servers Servers Servers Servers Servers Servers Servers Servers Servers Servers Serv	Settings Aliases Properties Subsets Members Attributes Member CODE Current Product CP_001 New Product 1 NP_001 New Product 2 NP_002 New Product 3 NP_003 New Product 4 NP_004	
8.2.3. **Define Properties**

Next, define Properties for product items that identify the type of season each product sells in.

- 1. In the Dimension Editing Window for the *Product* dimension, go to the **Properties Tab**. **Note:** Before you can define Properties, you must first create the Property Group.
- 2. Right-click on the Properties window and select New Property.
- 3. In the Create New Property dialog that appears, type the **<property group name>--**for this example, type **CLASSIFICATION**.
- 4. Click Create. The Property group called CLASSIFICATION appears listed in this window.

○ <i>≂</i>	Olation Studio	– 🗆 X
File Home Relational Multi-Dimensional	Hierarchy	Style - 🥅 MAX 🔿
Save Refresh Hierarchy		
Database Explorer 🛛 🗘 🗙	Product_Factdata	• ×
Servers	Settings Aliases Properties Subsets Members Attributes	
Iccalhost USING_OLATION	21 21 I	
Dimensions	✓ Misc	
🖾 Account	> CLASSIFICATION CLASSIFICATION	
Version	New Perdua 2	
🖾 Month		
🛂 Region		
SALES Measure		

Next, define the Property and assign them to respective Members within the *Product* dimension.

Note: Unlike Aliases, a single Property can be assigned to multiple Dimension members since it is only used for annotation purposes.

- 5. Go to the **Attributes** Tab.
- 6. In the Column CLASSIFICATION, create the Property for each Member, as shown in the following image:

○ <i>∓</i>	Olation Studio	– 🗆 X
File Home Relational Multi-Dimensional	Hierarchy	Style 👻 🥅 MAX 🔿
🖰 💋 🗍 С		
Save Refresh New New Property New Alias Subset		
Hierarchy Insert		
Database Explorer 🛛 🗘 🗙	Product_Factdata	- ×
Servers	Settings Aliases Properties Subsets Members Attributes	
Iocalhost	Settings Aliases Fropenies Subsets Members Attributes	
	Member CODE CLASSIFICATION	
L Dimensions	7	
☑ Account	Current Product CP_001 All-year-round	
☑ Version	New Product 1 NP_001 All-year-round	
₩ Month	New Produkt 2 NP_002 Seasonal	
☑ Region I SALES Measure	New Produkt 3 NP_003 All-year-round	
	New Product 4 NP_004 Seasonal	
🗘 SALES		

8.2.4. Define a Subset

Subsets are particularly helpful when you often use a specific set of Dimension Members for reporting, analytical and planning activities. Instead of repetitively picking these Members each time you create a new view of the data, you can select a Subset—the custom set of Members— and obtain results for its Members, all in a group, immediately.

For this exercise, we will create a subset group called New Products.

- In the Dimension Editing Window for the *Product* dimension, go to the **Subsets Tab**.
 Note: You need to create a Subset Group first, then, in the Attributes Tab, select the Members that will comprise the Subset.
- 2. Right-click on the Subsets window and select New Subset.
- 3. In the Create New Subset dialog that appears, type the **<subset group name>--** for this exercise, type **New Products**.
- 4. Click Create. The Subset group New Products now appears listed in this window:

0 :	Ŧ		Olation Studio	– 🗆 X
File	Home Relational Multi-Din	mensional	Hierarchy	Style 👻 🥅 MAX 🔿
	Refresh New New Property N Alias Sul	lew bset		
Database		Ψ×	Roduct_Factdata	▼ X
🗉 🗐 S			Settings Aliases Properties Subsets Members Attributes	
	localhost			
	Dimensions		✓ Misc	
	🖾 Account		> New Products New Products	
	Version			
	🖾 Month			
	🖾 Region			
	SALES Measure			

Next, select the Members for this Subset group:

 Go to the Attributes Tab. Tick the corresponding checkboxes of Members to be included in the Subset group. In this case, check the boxes corresponding to New Product 1 to New Product 4.

The Attributes Tab will look as follows:

○ <i>∓</i>	Olation Studio	– 🗆 X
File Home Relational Multi-Dimensional	Hierarchy	Style 👻 🥅 MAX 🔿
Save Refresh Hierarchy Insert		
Database Explorer 4 ×	Product_Factdata Product Settings Aliases Properties Subsets Members Attributes	▼ X
 Iocalhost USING_OLATION 	Member CODE CLASSIFICATION New Products	
□ 12 Dimensions 12 Account 12 Version	Current Product CP_001 All-year-round	
Month Region	New Produkt 1 NP_001 All-year-round Image: Comparison of the second	
SALES Measure	New Produxt 3 NP_003 All-year-round New Produxt 4 NP_004 Seasonal	
🗘 SALES		

8.2.5. Configure Other Dimension Settings - Settings Tab

For this exercise, we will enable the All Hierarchy option and disable and Total Aggregate option.

- 1. In the Dimension Editing Window for *Product* Dimension, go to the **Settings Tab**.
- 2. Locate the All Hierarchy option and set to True.

Locate the Total Aggregate option and set to False.

When completed, the Settings Tab will look as follows:

<mark>○</mark>	Olation Studio							
File Home Relational Multi-Dimensional	Hierarchy	Style - 🥅 MAX 🚫						
🖰 🖸 🗍 С								
Save Refresh New New Property New Alias Subset								
Hierarchy Insert								
Database Explorer 🛛 📮 🗙	Product_Factdata	• X						
Servers	Settings Aliases Properties Subsets Members	Attributes						
Iocalhost		Attributes						
	✓ Common							
Dimensions	Dimension Name	Product						
🖾 Account	Display Name	Floduct						
☑ Version	Display Ivalie Dimension Type	Standard						
🖾 Month								
	All Hierarchy	True						
Region	Data Source Type	Relational Table						
SALES Measure	> Database Source	dbo.Product						
Cubes	Total Aggregate	False						
SALES	> Hierarchies	Product_hierarchy						

8.2.6. Save the Dimension

After completing the creation of the Dimensions and defining its attributes:

- 1. Go to the **Hierarchy Tab** on the Olation ribbon.
- 2. Click the **Save** command.

	0	Ŧ						Olation Studio
	File	Home	Rela	ational	Multi-Dimensio	inal	Hierarchy	
(Save	Cefresh	New	New Prop	erty New		,	
	\smile	archy	Alias	Insert	Subset			

In the Database Explorer pane, note the newly created Dimension, i.e., **Product** (see circled in next image).

Refresh Tables: note also the two additional tables: **Product** and **Product_hierarchy** (see arrows in the next image. Indeed, Olation creates these tables as a result of adding them as (multidimensional) Dimensions in Olation Studio.

○	Olation Studio	- 🗆 ×
File Home Relational Multi-Dimensional	Hierarchy	Style 👻 🥅 MAX 📿
B B Image: Constraint of the sector of the		
Database Explorer 🛛 📮 🗙	Product_Factdata	. ×
Servers Iocalhost OUSING_OLATION		Attributes
L' Dimensions	✓ Common	
🖾 Account	Dimension Name Display Name	Product
🖾 Version	Display Name Dimension Type	Standard
🖾 Month	All Hierarchy	True
Region	Data Source Type	Relational Table
Product	> Database Source	dbo.Product
SALES Measure	Total Aggregate	False
🗉 ᠹ Cubes	> Hierarchies	Product_hierarchy
🗘 SALES		
Tables		
🖃 🎹 dbo		
iii dim_Accounts		
iii dim_Actual_vs_Budget		
iii dim_Months		
III dim_Regions		
III Factdata		
III Product		
Product_Factdata		
Product_hierarchy		
🗉 🔯 Views		
Queries		
🗉 🗹 External Data	Dimension Name	
Bookmarks	Display name for this dimension	
Local Databases		

Notice (following image) finally that the icon indicator for the *Product* dimension is blue, which is different from the other Dimensions (they are orange). This is indicative of the nature of the

Dimension. Whereas the first four Dimensions were created from a relational source (relational table), the *Product* dimension is of a multidimensional data type.



8.3. Create the PRODUCT_SALES Cube

With five Dimensions now existing in Olation—and having created the *Product_Factdata* table in SQL Server, we are now ready to create the PRODUCT_SALES Cube. (Note that if you view the Product_Factdata table in SQL Studio, you will see that it does not have any records, i.e., no transactional values.)

To create the PRODUCT_SALES Cube:

- 1. Under **Tables** in Database Explorer: right-click on **Product_Factdata** and select **New Cube**. The **Create New Cube** dialog box appears.
- In the dialog box, type a <name for your cube>.
 For this exercise, type PRODUCT_SALES as the Cube name.

Create New Cube							
New Cube							
Name: PRODUCT_SALES							
Create	Cancel						

3. Click Create. The Add Dimensions to Cube dialog box appears.

Add Dimensions to Cube
28 Discoving
Dimensions
Account
Version
Month
Region
Product
OK Cancel

- 4. Select the Dimensions to be included into the Cube by checking the corresponding checkboxes. (In this example select all five dimensions: **Account**, **Version**, **Month**, **Region** and **Product**, as shown in the image above.)
- 5. Click OK. This opens the Cube Definition Window on the right.

–		Olation Studio —	· □ >
File Home Relational Multi-Di	imensional	Cube Style	MAX
a 🖴 🗸 🔼 🔐	fx	fx _# 📦 🖄 fx fx 🕩 🌍 🎲 🖄	
	-		
ave Print Check Dimension Measure	e Formula	Persistent Cube (F1) Member (F2) Function Generate PC Dependency Region Cube (F3) Dimension (F4) Calculation	
- · · · ·	•		
Cube		Formulas/Persistent Calculations Dependencies General	
tabase Explorer	ų ×	Product_Factdata	-
Servers	- F	Settings Measures Dimensions Relationships Formulas Dependencies Persistent Calculations	1
Iocalhost			1
USING_OLATION			
□ 12 Dimensions		✓ Configuration	
Account		Name PRODUCT_SALES	
		Display Name	
☑ Version		Type Relational Source	
🖾 Month		Numeric Values Only False	
🖾 Region		Administrative Cube False	
🖾 Product		Relational Synchronization Rebuild	
SALES Measure		Synchronization Frequency On Demand	
🗉 🌍 Cubes		Read-Only False	
SALES		Transaction Log False Batch Relational Updates False	
		Faise Formulas	
		V rollinas Unlimited Calculation Depth False	
		Maximum Calculation Depth 30	
dim_Accounts		Use Dependencies False	
dim_Actual_vs_	Budget	Protection False	
iii dim_Months		Expire False	
iii dim_Regions		✓ Measure Dimension	
III Factdata		Dimension Name PRODUCT_SALES Measure	
III Product		Database Source Product_Factdata	
Product_Factda	ta	Data Source Type Relational Table	
Product_hierarcl	hv	Query Condition	
	· ·	Microsoft® SSAS Integration	
		Synchronize Olation Cube False	
I Gueries I Create Strain S		SSAS Server Name/Address	
R Bookmarks		The name of the cube	
B Security Security			
Local Databases			

8.3.1. Assign the Measure Member

- 1. In the **PRODUCT_SALES Cube Definition Window**, go to the **Measures** tab.
- 2. Select from the list which item(s) you want to designate as Measures. For this exercise, check the **Amount** checkbox.

	/ 🔯 Produc	ct_Factdata	🔽 Product	🌍 PRODUC	T_SALES					→ ×
	Settings	Measures	Dimensions	Relationsh	ips Formulas	Dependenci	es Persis	tent Calculat	ions	
	Column Name	!	Data Type	Unique Id	Display Name	Format	Increme	Version	Drill	Virtual
	🗌 dim_Actua	a_VS_Budget	nvarchar			*0.0				
	dim_Acco	unts	nvarchar			*0.0				
	🔲 dim_Regio	ons	nvarchar			*0.0				
	dim_Mont	hs	nvarchar			*0.0				
-D	🗹 Amount		decimal			*0.0				
	Products		int			*0.0				

OLATI**O**N

8.3.2. Arrange the Dimension Order

To define the order of the Dimensions:

- 1. In the **Sales Cube Definition Window**, go to the **Dimensions** Tab.
- 2. Drag and drop the Dimensions in the following order as shown in the image below:

∕ 🔯 Product_Factdata	Product SPRODUCT_SALES	→ ×
Settings Measures	Dimensions Relationships Formulas Dependencies Persistent Calculations	
Dimension Name	Туре	
PRODUCT_SALES Measure	🖾 Measure	
Version	1∠, Standard	
Region	🗠 Standard	
Product	12, Standard	
Account	🗠 Standard	
Month	12, Standard	

8.3.3. **Define Relationships**

These steps will be similar to how relationships were defined in the SALES Cube. The only difference is the need to link an additional Dimension Table (i.e., *Product* Dimension) to the Measure Table—thus, go to the **Relationships Tab** and link the five (5) Dimension Tables (*Account, Version, Month, Region* and *Product*) to the Measure Table (*Product_Factdata*):

- 1. Click on **dim_Actual_Vs_Budget** in the *Measure Dimension* table on the left; drag and drop to **Actual vs Budget** of the *Version Dimension* table on the right.
- 2. Click on **dim_Accounts** in the *Measure Dimension* table; drag and drop to **Account** in the *Account Dimension* table on the right.
- 3. Click on **dim_Regions** in the *Measure Dimension* table; drag and drop to **Region** column in the *Region Dimension* table on the right.
- 4. Click on **dim_Months** in the *Measure Dimension* table on the left; drag and drop to **Month** in the *Month Dimension* table on the right.
- 5. Click on **Products** in the *Measure Dimension* table; drag and drop to the **id** column in the *Product Dimension* table on the right.

Product_Factdata	PRODUCT_SAL	ES				• ×
Settings Measures Dimensions	Relationships	Formulas	Dependencies	Persistent Calcul	ations	
	Relationships	Formulas	Version Dimension (Actual vs Budget Parent Weight Account Dimension Account Parent Weighting Ordering Region Dimension (Region Continent World Month Dimension (d Year Quarter Month Ordering Subset_Months	(dim_Actual_vs_E (dim_Accounts) (dim_Accounts) (nvarcha (nvarcha (nvarchar (nvarchar (nvarchar (nvarchar (nvarchar (nvarchar (nvarchar (nvarchar (nvarchar (nvarchar (nvarchar) (nvarchar (nvarchar) (nvarcha	Budget) nvarchar nvarchar int	• X
				nvarchar nvarchar		

8.3.4. Define Cube Setting: Disable READ-ONLY Setting

For this new Cube—as with the *SALES* cube—we want to configure it to have write-back capability, so we will disable the Read-Only setting. To do this:

- 1. In the **PRODUCT_Sales Cube Definition Window**, go to the **Settings** tab.
- 2. In the Configuration section, locate the **Read-Only setting**, and set this to **False**.

8.3.5. Save the Cube

- 1. Go to the **Cube Tab** of the Olation ribbon.
- 2. Click the Save Cube icon.

Note: The Cube Tab becomes visible when you bring up the $\ensuremath{\textbf{Cube Definition Window}}.$

The **PROUDCT_SALES** Cube now appears listed under Cubes in Database Explorer.

Ŧ	Olation Studio	>
File Home Relational Multi-Dimensio	al Cube	Style - MAX
🍋 🚔 🖌 🗠 🚽 🗠	x ^f x _# 📬 🖄 fx fx 🚦)= 🔇 🎲 🖄
		endency Region Cube (F3) Dimension (F4)
- Cube	Calculation	Dependencies General
Servers		▼
Servers	Settings Measures Dimensions Relationships Form	nulas Dependencies Persistent Calculations
	✓ Configuration	
□ 12 Dimensions		PRODUCT_SALES
[™] Account	Display Name	
Version	Type	Relational Source
🖾 Month	Numeric Values Only	False
🖾 Region		False
Product		Rebuild
SALES Measure		On Demand
PRODUCT SALES Meas		False False
□ 🗘 Cubes		raise False
SALES		Faise
PRODUCT SALES		False
		30
		False
🗆 🎹 dbo		False
dim_Accounts		False
iii dim_Actual_vs_Budg		
iii dim_Months		PRODUCT_SALES Measure
iii dim_Regions		Product_Factdata
III Factdata		Relational Table
III Product	Query Condition Microsoft® SSAS Integration	
III Product Factdata		False
III Product_hierarchy	SSAS Server Name/Address	
∃ Views		True
Queries		True
I Queries I I I External Data	Olation Web API Server URL	http://localhost
	SSAS Log File	
Bookmarks	✓ OLEDB for OLAP	
B Becurity		False
📄 Local Databases	Persistent Calculations	T
		True 3
		5 False
	✓ Power0LAP● Integration	
		False
		localhost
	PowerOLAP Database Name	
	Olation Server Name	localhost
	Name	
	The name of the cube	

8.4. View the PRODUCT_SALES Cube

We will next explore the *PRODUCT_SALES* cube to view the metadata structure that we just built. Remember that there is NO FACT DATA yet for this Cube since the relational table from which it was created from (**dbo.Product_Factdata**) has no transactional values/records. Once again, for this exercise, we will view this (second) Cube using PowerExcel as the front-end client.

To view the PRODUCT_SALES Cube:

- 1. Go to Excel and click the **PowerExcel Tab** from the Excel ribbon.
- 2. Create a new PowerExcel Slice:
 - In the PowerExcel Tab, click the New icon.
 This will display the PowerExcel sidebar on the right-hand area of the Excel worksheet.
 - In the PowerExcel sidebar, click on the Database drop-down and select the correct PowerExcel Database connection, i.e., USING_OLATION.
 Note: Since we have previously configured this database connection in an earlier exercise, we will already see it listed in the drop-down menu.
 - Click on the **Cube drop-down** and select the correct Cube, i.e., **PRODUCT_SALES**. The five Dimensions of the selected Cube appear in the Filters, Columns and Row boxes.

Aut	oSave 💽							Во	ok3 - Excel) 🖬 –		
File	Home	Insert	Pag	e Layout	Formulas	5 Data	Review	View	Developer	Help	Pov	werExcel 🔎 Search	,O Search	ß	P
O New		ve Refres	OLA Fx		Connection PowerEx	s Options	Clear Cache Dir	Edit nension	About PowerExcel Help						^
Al					Ŧ	: ×	$\checkmark f_x$								~
	А	В	С	D	E	F	G	н	1	J	ŀ≜	PowerExcel		-	×
1															
3												Database USING_OLATION		~	X
4												Cube			
5 6											_	PRODUCT_SALES		\sim	X.
7												Filters			
8													re: Amount Variance		
9											_	12 Region:	World		
10 11												14 Product:	All		
12															
13												Columns			_
14											_	Account: All			
15 16															
17															
18												Rows			_
19											_	A Month: ALL			
20 21															
22															
23												Options	PowerExcel Slice		
24												Constrain Empty Rows	PivotTable		
25 26											_	Delete Removed Rows	Read/Write Form	mulas	
20												Expandable Members	O Power Query		
28												Insert Into			
29) Current Worksheet		
30												New Worksheet	Location:	X	
31											-	🔟 Insert			
	Þ	Sheet1	+				: •				Þ				
Ready	Calculate	.												+	100%

• Check the Dimension Members and attributes of the two newly created Dimensions, **PRODUCT_SALES** Measure, and **Product** by double-clicking on them.

Select Members			- 0	×		
Members Hierarchy			Q	8 🗸		
t2 • •	▼ <u>4</u> Q	# Amount				
	Select Members				_	×
	Members Hierarchy		i 🗆 🛇 ⊃ 🄁 🛪		Q	8 🗸
	Find: Q	ES XK	# Amount			

Dimension: PRODUCT_SALES Measure

Select Members						_		×		
Members Hierarchy				⊃ 🚬 🗶		Q		8 🗸		
# All # Current Product # New Product 1 # New Product 2 # New Product 3 # New Product 4	• ¥ 4	Q	# All							
	Select Members								_	×
	Members Hierarchy				⊡ ⊘	⊃ ⊇	X		Q	8 🗸
		rent Product w Product 1 w Product 2 w Product 3 w Product 4	X X		# All					

Dimension: Product Measure

- 3. Next, to create a PowerExcel Slice:
 - Leave the default Slice arrangement.
 - Pick **PivotTable** as the PowerExcel Slice type.
 - Select to insert into the **Current Worksheet** starting at **cell A1**. Click the **Insert** button. The PowerExcel Slice will appear as follows (it is worth mentioning that this is a kind of

"default" Slice, as we made no selections along Filters, Columns or Rows): **Note:** As you can see, the PowerExcel Slice currently has NO FACT DATA. What has been built so far is just the metadata structure of the Cube.

Ą	AutoSave 💽 升	<u>ا الا</u>	(1 + +			Book	3 - Excel 🛛				×
Fi	e Home	Insert F	Page Layout	Formulas	Data Revie	w View	Developer	Help	Pov	werExcel 🔎 Search	r S
< N		Refresh Fir		Connections Opt	Cache		About owerExcel				
	Powe	rExcel Slice		PowerExcel Co	nnection	Dimension	Help				
A1	0				$\times \checkmark$	fx =OLAP	ivotTable(\$B	\$1,\$B\$2,\$E	3\$3:\$E	\$3,\$B\$4:\$E\$4,\$B\$5:\$E\$5,\$B\$	\$6:\$E\$6,\$B\$7:\$E\$7,\$B\$8:\$E\$8,
	A	В	с	D	E	F	G	н			
1	Database:	USING_OLA	TION							PowerExcel	▼ X
2	Cube:	PRODUCT_S	ALES							Database	
3	Dimensions:	Filter	PRODUCT_9	6A Members	Amount					USING_OLATION	~ X
4		Filter	Version	Members	Variance				_	Cube	
5		Filter	Region	Members	World				_	PRODUCT_SALES	~ ¥.
6		Filter	Product	Members	All				_	T Filters	
7		Column	Account	Range	\$B\$11:\$F\$11				_	A PRODUCT_SALES Measure	e: Amount
8		Row	Month	Range	\$A\$12:\$A\$29					12 Version:	Variance
9										12 Region:	World
-	OLAPivotTable									14 Product:	All
11		All	Sales	Margin	Cost of Sales	-					
	All Total Quarter			0 0	-	0				🔁 Columns	
	January			0 0		0				レ Account: All, Sales, Margi	n,Cost of Sales,Margin Pcnt
	1st Quarter		-	0 0	-	0					
	February			0 0	-	0			-11		
_	March		0	0 0		0					
	April		-	0 0	-	0				🚰 Rows	
_	2nd Quarter		-	0 0	-	0				🖾 Month: All,Total Quarter,	January, 1st Quarter, February, March,
	May			0 0	-	0					
	June		-	0 0	-	0			-		
	July		-	0 0	-	0				<	>
	3rd Quarter		0	0 0) 0	0				Options	PowerExcel Slice
24	August		0	0 0) 0	0				Constrain Empty Rows	PivotTable
25	September		0	0 0	0	0				Delete Removed Rows	O Read/Write Formulas
26	October		0	0 0) 0	0				Expandable Members	O Power Query
27	4th Quarter		0	0 0) 0	0					
28	November		0	0 0) 0	0				Insert Into	
	December		0	0 0	0	0				0 0	CurrentWorksheet
30										New Worksheet	Location: 👫
31											
22										Update	
		neet1 (+		: •				Þ		
Calc	ulate 🔚										+ 100%

8.5. Create a Cross-Cube Formula for the PRODUCT_SALES Cube

We next want to populate the newly created Cube with fact data: we will pull the fact data from the first Cube (created earlier, the *SALES* Cube) into specific intersections of the *PRODUCT_SALES* cube by using a Cross-Cube Formula.

Because there is a difference in the dimensionality of the two Cubes—the second Cube has an additional Dimension (*Product*), we will need to define the specific intersections where the data will go.

First, let's make an assumption for the sake of this exercise: in the past, our example company carried one product (*Current Product*) but now wants to launch 4 additional products (*New Product 1, New Product 2, New Product 3* and *New Product 4*). The Finance Director wants to use *Current Product* data as the basis to create analyses and plans for the new products. Our objective, therefore, will be to pull in data from the *SALES* CUBE, targeting results to the **Current Product** Member of the *Product* Dimension in the *PRODUCT_SALES* cube.

To create the Cross-Cube formula for the PRODUCT_SALES Cube:

- 1. In Olation Studio double-click on the **PRODUCT_SALES** Cube.
- To define the formula, go to the Formulas Tab
 In the formula pane, you can type in a descriptive comment to identify what it is for. Make use of
 the symbols // or; /* and */.
- 3. Write the **LHS Expression (**aka **Range Reference)**. You can use the command buttons along the Cube Tab of the Olation ribbon:
 - Click on the Formula button and select All And {} as the qualifier; double-click to select PRODUCT_SALES Measure as the Dimension and double-click for Amount as Member (as in the following image).

🖉 Prod	uct_Factdata 🛛 🔽 Product 🔗 PRODUCT_SALES	• ×
Settings	Measures Dimensions Relationships Formulas Dependencies Persistent Calculations	
1	//Pull Fact Data from SALES Cube to this Cube	
2	//Cross-Cube formula\	
3	All and {"PRODUCT_SALES Measure.} =	
4	Amount	
5		
6		
7		

Place your cursor on the line below.
 Once again, click on the Formula button and select All And {} as the qualifier; select
 Product as the Dimension and Current Product as Member.



• Delete All at the beginning of the second expression.

• Delete the first **equals symbol** ("="), to the right on the top line, so the formula looks as follows:

🖉 Prod	uct_Factdata 🛛 😰 Product 🔗 PRODUCT_SALES	→ ×
Settings	Measures Dimensions Relationships Formulas Dependencies Persistent Calculations	
1	//Pull Fact Data from SALES Cube to this Cube	
2	//Cross-Cube formula\	
3	All and ("PRODUCT_SALES Measure.Amount")	
4	and ("Product.Current Product")	
5	=	
6		
7		
8		

- 4. Move your cursor after the equals symbol (=).
- To write the RHS Expression (aka Cube Reference).
 Click on the Cube button and select the SALES Cube; double-click to select SALES Measure as the Dimension and double-click for Amount as Member.

Cube								St
fx _#		12	fx	fx	_	\$		Ľ,
a Persistent Calculation	Cube (F1)	Member (F2)	Function	Generate PC	Dependency	Region	Cube (F3)	Dimension (F4
	SALES	;			Depender	ncies	(General
🛛 🔀 Prod	PROD	UCT_SALES	1	PRODUCT_SA	LES			
Settings	Measures	Dimensi	ions I	Relationships	Formulas	Depende	ncies Po	ersistent Calculat
1	//Pull	Fact Dat	a from	SALES Cub	e to this	Cube		
2	//Cross	-Cube fo	rmula\					
3	All and	{ "PRODU	CT_SAL	ES Measure	.Amount"}			
4	and ("P	roduct.C	urrent	Product"}				
5	=							
6								
7	•							
8								

6. Type a **semi-colon** (;) to indicate the end of the formula. The formula will appear as follows (you may hit **Enter** between lines—it will not affect the formula):

🛛 💆 Prod	uct_Factdata	🔽 Product 🧃	PRODUCT_SA	LES			▼ ×
Settings	Measures	Dimensions	Relationships	Formulas	Dependencies	Persistent Calculations	
1 2 3 4 5 6 7 8	//Cross- All and and {"Pr =	act Data fro Cube formula ("PRODUCT_S! oduct.Currer SALES Measur	a) LES Measure at Product")	≘.Amount") }			^

- 7. Click the **Check icon** (the green checkmark), aka **Check Syntax button** to verify that the formula is free from syntax errors. You will get a prompt that says, 'The formula syntax is correct'.
- 8. Save the **Cube**.

8.5.1. Viewing the Results of the Cross-Cube Formula

- 1. Go back to the PowerExcel Slice we have created for the PRODUCT_SALES Cube.
- To update the PowerExcel Slice data, click on the Refresh icon found on the PowerExcel Tab. You can also just simply press F9.
 Note: Make sure that your PowerExcel Database connection is active.

AutoSave 💽 所 E PowerExcel File Home Insert Page Layout Formulas Data Review View Developer Help Z * 0 Ê New Open Save Refresh Show Find Options Connections Options Clear Edit Abouit Change PowerExcel Password idebar OLA Fx Cache Dimension Slice PowerExcel Connection Dimension Help Pow

You can now see data come into the Slice.

A	AutoSave 💽	日 り· 🤆	- -			Book3 - Exc	el 🔍				m –		×
Fi	e Horne	Insert Pa	ge Layout - F	ormulas D	ata Review	View De	veloper	Help	Power	rExcel 🔎 Search	,0 march	Ŕ	₽
Clip	> 💞 –		11 ▼ A^ A` ▼ 24 ▼ <u>A</u> ▼	E = = = :	ient 🕞	General ▼ \$ ~ % 9 50 - 300 Number 5	😿 Forn	Styles	-	E Delete • Format • Cells	AZY O Sort & Find & Filter * Select * Editing	J Ideas	~
A1	0			· ·	$\times \sqrt{f_x}$	=OLAPivot	Fable(\$B\$	1,\$B\$2,\$B\$3	3:\$E\$3	;,\$B\$4:\$E\$4,\$B\$5:\$E\$5,\$B	\$6:\$E\$6,\$B\$7:\$E\$	\$7,\$B\$8:	Ý
1	A	В	с	D	E	F	G	Н		PowerExcel		-	×
	Database:	USING_OLATI								POwerExcer			
_	Cube:	PRODUCT_SA								Database			
_	Dimensions:	Filter	PRODUCT_SAI		Amount					USING_OLATION		~	×
4 5		Filter Filter	Version	Members	Variance World					Cube			
5 6		Filter	Region Product	Members Members	All					PRODUCT_SALES		~	X.
о 7		Column	Account	Range	AU \$B\$11:\$F\$11					T Filters			
/ 8		Row	Month	Range	\$A\$12:\$A\$29					2 PRODUCT_SALES Measur			
0 9		NUW	WORKI	Nalige	ŞAŞI2.ŞAŞ23					ピ. Version: ピ. Region:	Variance World		
-	OLAPivotTable									i∠, Region: 1∠, Product:	All		
11	OBAINOCIUDIC	All	Sales	Margin	Cost of Sales	Margin Pont				- House			
12	ΔΠ	50685			-4991	4.11							
	Total Quarter	50685		60663	-4991	4.11				🔁 Columns			
_	January	85431	91995	98558	-6563	-0.96				🖾 Account: All,Sales,Margi	in,Cost of Sales,Mar	gin Pcnt	
_	1st Quarter	57846	71924	86001	-14077	-1.00							
16	February	-29822	-21285	-12748	-8537	-0.50							
17	March	2237	1214	191	1023	0.46							
18	April	2622	1700	779	921	0.83				Rows			
19	2nd Quarter	8251	5273	2297	2976	2.21				🖾 Month: All,Total Quarter	r, January, 1st Quarte	r,February,M	larch,
20	May	2835	1788	742	1046	0.66							
21	June	2795	1785	776	1009	0.72				<			
22	July	3007	1666	326	1340	0.66							>
23	3rd Quarter	8498	3974	-549	4523	1.25				Options	PowerExcel Slice		
24	August	3452	1391	-670	2061	-0.12				Constrain Empty Rows	PivotTable		
25	September	2040	917	-205	1122	0.71				Delete Removed Rows	🔘 Read/Write F	ormulas	
26	October	-5696	-6223	-6749	526	0.66				Expandable Members	O Power Quen		
27	4th Quarter	-23910	-25499	-27086	1587	1.65				1			
28	November	-8907	-9632	-10357	725	0.12				Insert Into			
_	December	-9307	-9644	-9980	336	0.87) Current Workshee		Mil
30										New Worksheet	Location: \$4\$1		¥.
31									Ŧ				
	⊂ → SI	heet1 (+)		: •			•		Update			
Rea	dy Calculate '	F						🚮 Di	isplay S	settings 🌐 🗉 🗄	·	+ 1	00%

3. Given that we created a Cross-Cube Formula to get data from the SALES cube and push it into the specific Product Member—*Current Product*—of the *PRODUCT_SALES* cube, we can make a comparison of two Slices.



First, change the latest Slice like so:

- In the Filters box for the Slice (from the PRODUCT_SALES cube), select the following: PRODUCT_SALES Measure: Amount Version: Actual Region: United States Product: Current Product
- Change the display Members along the columns for the *Account* Dimension to only show: **Sales**, **Cost of Sales**, **Margin** and **Margin Pcnt**.
- Change the display Members along the rows to show the following:

#January
井 February
井 March
∑ 1st Quarter
井 April
井 May
井 June
∑ 2nd Quarter
井 July
🗰 August
井 September
∑ 3rd Quarter
October
🗰 November
🗰 December
∑ 4th Quarter
Total Quarter

The PowerExcel Slice will update to show the following:

	А	В	с	D	E	F	G	н			
Datab	ase:	USING_OLATI	ON							PowerExcel	*
Cube:		PRODUCT_SA	LES							Database	
Dimer	nsions:	Filter	PRODUCT_SA	Members	Amount					USING OLATION	~ ¥
		Filter	Version	Members	Actual					 Cube	
		Filter	Region	Members	United States					PRODUCT_SALES	~ X
		Filter	Product	Members	Current Produc	t				Filters	
		Column	Account	Range	\$B\$11:\$E\$11					24 PRODUCT SALES Measure	Amount
		Row	Month	Range	\$A\$12:\$A\$28					4 Version:	Actual
										12 Region:	United States
OLAPi	votTable									14 Product:	Current Product
		Sales	Cost of Sales	Margin	Margin Pcnt						
Januar	ry	5000	840	4160	0.83					Columns	
Febru	ary	5285	960	4325	0.82					Columns	
March		5586	580	5006	0.90					Account: Sales, Cost of Si	ales, Margin, Margin Pont
i 1st Qu	larter	15871	2380	13491	0.85						
i April		5905	700	5205	0.88						
' May		6241	660	5581	0.89					Rows	
3 June		6597	700	5897	0.89						y, March, 1st Quarter, April, May, June, 3
2nd Q	uarter	18743	2060	16683	0.89					🔄 Month: January, rebruary	y, March, 1st Quarter, April, May, June, .
) July		6973	1020	5953	0.85						
Augus		7370	800	6570	0.89					<	>
Septe	mber	7791	860	6931	0.89						
3rd Qu	uarter	22134	2680	19454	0.88					Options	PowerExcel Slice
Octob		3591	0		1.00					Constrain Empty Rows	PivotTable
Nover	nber	4453	0		1.00					Delete Removed Rows	Read/Write Formulas
5 Decen	nber	5027	0		1.00				_	Expandable Members	O Power Query
4th Qu		13071	0		1.00				_	Insert Into	
) Total (Quarter	69819	7120	62699	0.90						CurrentWorksheet
1										0 0	
)										○ New Worksheet	Location: \$A\$1 🎽
									-		
$\prec \longrightarrow$	Sh	ieet1 (+	•		÷ •				F	🛅 Update	

 Next, show (or open) the Slice created earlier from the SALES cube to do a side by side comparison with this new PowerExcel Slice from the PRODUCT_SALES cube.
 Notice that, indeed, the data matches across the two Cube, as shown in the images below:

Fi	le Home	Insert	Page Layo	ut Formul	las Da	ta Review	View Deve	eloper	Help	р	PowerExce	I P s	Gearch				P				
	🔵 🗁 📙 ew Open Save	Refresh	Show Fir	nd Options	Conne		Clear Ed		Abo	-	* 🗗 Change										
D.			Sidebar OLA				ache Dimer	ision			I Password							~			
AS		Dimerizader .	Silice		-	× ✓			able(\$		\$B\$2,\$B\$3::	\$E\$3,\$B\$4	:\$E\$4,\$	B\$5:\$E\$	\$5,\$B\$6:	\$E\$6,\$B		~			
	A		В	с		D	E		F		_						~				
	Database: Cube:	USING_C SALES	DLATION						-		Powerl Database	Excel				*	×				
3 4		Filter Filter		SALES Measu Version		embers embers	Amount Actual				USING_OL	ATION					~ ¥				
5		Filter	1	Region	M	embers	United State				Cube SALES						~ ¥				
6 7		Column Row		Account Month		nge nge	\$B\$10:\$E\$10 \$A\$11:\$A\$2				Filters	Measure: A	mount								
8 9	OLAPivotTable								_		ピ. Version ピ. Region	n: A	kctual Inited St	ates							
10	lanuanu	Sales	5000	Cost of Sales	5 M	argin	Margin Pont	83%													
	January February		5285		960	416 432	5	82%			💀 Column	15						-			
	March 1st Quarter		5586 15871		580 2380	500 1349		90% 85%	_			nt: Sales,Co	ost of Sal	es, Marg	in, Margir	Pont					
15	April May		5905 6241		700 660	520 558	5	88% 89%													
17	June		6597		700	589	7	89%			🚰 Rows							-			
	2nd Quarter July		18743 6973		2060 1020	1668 595		89% 85%			14 Month	:: January,F	ebruary,	March,1:	st Quarte	,April,May	/June,	21			
	August September		7370 7791		800 860	657 693		89% 89%			<						>				
22	3rd Quarter		22134		2680	1945	4	88%			Options			Powerf	xcel Slice						
	October November		3591 4453		0 0	359 445		100% 100%			_	ain Empty Ro Removed Ro		Pive	otTable ad/Write I	ornular					
	December 4th Quarter		5027 13071		0	502 1307		100% 100%	_		_	lable Memb			ver Queŋ						
	Total Quarter		69819		7120	6269		90%			Insert Into	D									
29			10				· ·	\times	~	f_{x}	=OLAPi	votTable(\$B\$1,\$I	B\$2,\$B\$	3:\$E\$3,	\$B\$4:\$E\$	\$4,\$B\$	5:\$E\$5	,\$B\$6:\$E\$6,\$B\$7:\$E	\$7,\$B\$8:	~
30 91			A		В	с	D		E		F	G	Н		Pou	verEx	col			-	×
Cal	ulate 🛅	2		PRO	IG_OLAT DUCT_SA	LES									Datal		CEI				^
		3		ons: Filte Filte		PRODUCT_SA Version	Members Members	Amou Actua							USIN	IG_OLATIO	N			~	X.
		5		Filte		-	Members Members	Unite	d State nt Proi						PRO	DUCT_SA	LES			~	¥.
		7		Colu	ımn	Account	Range	\$B\$11	:\$E\$11	L						RODUCT_	SALES	Measure	e: Amount		
		8 9		Row		Month	Range	\$A\$12	2:\$A\$21	8					14 6	/ersion: Region:			Actual United States		
		10	OLAPivot	tTable Sale	s	Cost of Sales	Margin	Margi	n Pcnt						4	roduct:			Current Product		
			2 January 3 February	r	5000 5285		4160		0.8							olumns					
		14	March 1 St Quart		5586	580	5006		0.9	0					14,	(ccount:	Sales,C	ost of Si	ales, Margin, Margin Po	nt	
		16	5 April	lef	15871 5905	700	13491 5205		0.8 0.8	8											
			7 May 3 June		6241 6597		5581 5897		0.8						🚰 R						
			2nd Quar July	rter	18743 6973		16683 5953		0.8 0.8							4onth: Ja	inuary, I	Februan	y, March, 1st Quarter, Ap	oril, May, June	2,2t
		21	I August		7370	800	6570		0.8	9					<						>
		23	Septemb 3 3rd Quar		7791 22134	2680	6931 19454		0.8 0.8	8					Opt	ions			PowerExcel Slice		
			October Novemb	er	3591 4453		3591 4453		1.0							onstrain E elete Rem			 PivotTable Read/Write Form 	mulas	
		26	5 Decembe 7 4th Quar	er	5027 13071	0	5027 13071		1.0 1.0	0					E	xpandable	e Memb	pers	O Power Query		
		28	3 Total Qu		69819		62699		0.9							rt Into New Work	book	۲	CurrentWorksheet		
		29 30)													New Work		9	Location: \$A\$1	Ya	
		31	I → →	Sheet1	(Ð			4					•	٦U	pdate					~
		Ca	alculate 📲												Display Se	ttings	=	Ξ		+	100%

Slices from the SALES cube and the PRODUCT_SALES cube showing United States, Actuals.



Version Region Account Month 200 Cost of Sal 500 750 2793 4043 2052 3121 328 9371 3487 3685 3695 11067 1795 1294 3695 3	500 500 500 500 500 500 500 500	ES.	\$27 cnt 0% 33% 82% 63% 82% 63% 84% 85% 84% 86% 86% 86% 86% 87% 86% 72% 52% 80% 72% 80%	Rows Constrain Empty Rows Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into	al ida of Sales, Margin, Margin Pont uary, March, 1st Quarter, April, May, June, 2i PowerExcel Silce PowerExcel Silce PivotTable	
SALES Mea Version Region Account Month Cost of Sal	Members Members Range Range Range S00 S00 S00 S00 S00 S00 S00 S00 S00 S0	Actual Canada S\$510.555 S41 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2623 2798 2830 2849 2013 2849 2013 2849 2013 2849 2013 2849 201 201 201 201 201 201 201 201 201 201	\$27 ont 0% 33% 82% 63% 83% 84% 85% 86% 86% 86% 86% 86% 86% 86% 86	Database USING_OLATION Cube SALES Filters SALES Columns Cara Columns Constrain EmptyRows Delete Removed Rows Expandable Members Insert Into	unt al da da of Sales, Margin, Margin Pont uary, March, 1st Quarter, April, May, June, 2r warg, March, 1st Quarter, April, May, June, 2r PowerExcel Silce PowerExcel Silce PowerExcel Silce PowerExcel Silce PowerExcel Silce Power Query able (\$B\$1, \$B\$2, \$B\$3: \$E\$3, \$B\$4: \$E	
Version Region Account Month 2002 2793 2793 2793 2793 2793 2793 2793 279	Members Members Range Range Range S00 S00 S00 S00 S00 S00 S00 S00 S00 S0	Actual Canada S\$510.555 S41 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2623 2798 2830 2849 2013 2849 2013 2849 2013 2849 2013 2849 201 201 201 201 201 201 201 201 201 201	\$27 ont 0% 33% 82% 63% 83% 84% 85% 86% 86% 86% 86% 86% 86% 86% 86	USING_OLATION Cube SALES Filters L' SALES Measure: Amo L' Version: Actu L' Region: Cana Columns L' Account: Sales,Cost of Rows L' Month: January,Febr Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into Sales L' Account: Sales,Cost of Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into	unt al da da of Sales, Margin, Margin Pont uary, March, 1st Quarter, April, May, June, 2r warg, March, 1st Quarter, April, May, June, 2r PowerExcel Silce PowerExcel Silce PowerExcel Silce PowerExcel Silce PowerExcel Silce Power Query able (\$B\$1, \$B\$2, \$B\$3: \$E\$3, \$B\$4: \$E	
Version Region Account Month 2002 2793 2793 2793 2793 2793 2793 2793 279	Members Members Range Range Range S00 S00 S00 S00 S00 S00 S00 S00 S00 S0	Actual Canada S\$510.555 S41 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2621 2798 2623 2798 2830 2849 2013 2849 2013 2849 2013 2849 2013 2849 201 201 201 201 201 201 201 201 201 201	\$27 ont 0% 33% 82% 63% 83% 84% 85% 86% 86% 86% 86% 86% 86% 86% 86	Cube SALES Filters L' SALES Measure: Amo L' Version: Actu L' Region: Cana Columns L' Account: Sales,Cost of Rows L' Month: January,Febr Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into	unt al da da of Sales, Margin, Margin Pont uary, March, 1st Quarter, April, May, June, 2r warg, March, 1st Quarter, April, May, June, 2r PowerExcel Silce PowerExcel Silce PowerExcel Silce PowerExcel Silce PowerExcel Silce Power Query able (\$B\$1, \$B\$2, \$B\$3: \$E\$3, \$B\$4: \$E	
Region Account Month Cost of Sal 750 2793 2793 2952 3121 3298 9371 3487 3685 1061 2513 35349 29830 10641 2053 3549 2054 2055	Members Range Range Range S00 S00 S00 S00 S00 S00 S00 S00 S00 S0	Canada \$B\$10:\$E\$ \$A\$11:\$A Margin Po 2293 2254 2452 2543 2452 2621 2798 2621 2987 3895 9557 1295 541 2013 3849 2013 3849 2013 3849 2013 2014 2015 541 2015	\$27 ont 0% 33% 82% 63% 83% 84% 85% 86% 86% 86% 86% 86% 86% 86% 86	SALES Filters SALES Filters SALES Measure: Amo SALES Version: Actu Region: Cana Columns Actu Region: Cana Columns Actu Region: Cana Columns Constrain Sales, Cost of Options Constrain Empty Rows Expandable Members Insert Into SALES COLAPIVOT	unt al Ida If Sales, Margin, Margin Pcnt Uary, March, 1st Quarter, April, May, June, 2i Uary, March, 1st Quarter, April, May, June, 2i PowerExcel Silce PowerExcel Silce PowerExcel Silce PowerExcel Silce PowerExcel Silce Power Query able(\$B\$1, \$B\$2, \$B\$3: \$E\$3, \$B\$4: \$E	
Account Month Cost of Sal Cost of Sal Cost of Sal 2793 2795	Range Range Range S00 500 <td>\$B\$10:\$E\$ \$A\$11:\$A Margin Pc 2293 2543 2452 2452 2452 2452 2452 2452 245</td> <td>\$27 ont 0% 33% 82% 63% 83% 84% 85% 86% 86% 86% 86% 86% 86% 86% 86</td> <td>Filters SALES Measure: Amo L' SALES Measure: Amo L' Version: Actu L' Region: Cana Columns Acture Columns A</td> <td>unt al Ida If Sales, Margin, Margin Pcnt Uary, March, 1st Quarter, April, May, June, 2i Uary, March, 1st Quarter, April, May, June, 2i PowerExcel Silce PowerExcel Silce PowerExcel Silce PowerExcel Silce PowerExcel Silce Power Query able(\$B\$1, \$B\$2, \$B\$3: \$E\$3, \$B\$4: \$E</td> <td></td>	\$B\$10:\$E\$ \$A\$11:\$A Margin Pc 2293 2543 2452 2452 2452 2452 2452 2452 245	\$27 ont 0% 33% 82% 63% 83% 84% 85% 86% 86% 86% 86% 86% 86% 86% 86	Filters SALES Measure: Amo L' SALES Measure: Amo L' Version: Actu L' Region: Cana Columns Acture Columns A	unt al Ida If Sales, Margin, Margin Pcnt Uary, March, 1st Quarter, April, May, June, 2i Uary, March, 1st Quarter, April, May, June, 2i PowerExcel Silce PowerExcel Silce PowerExcel Silce PowerExcel Silce PowerExcel Silce Power Query able(\$B\$1, \$B\$2, \$B\$3: \$E\$3, \$B\$4: \$E	
Month Cost of Sal	Range	\$A\$11:\$A Margin Pc 0 2293 2543 2452 2621 2798 7871 2987 7871 2987 7871 2987 7871 2987 7871 2987 7871 2013 3849 2013 3849 2013 3849 2033 3849 2033 2035 2015 2015 2015 2015 2015 2015 2015 201	\$27 ont 0% 33% 82% 63% 83% 84% 85% 86% 86% 86% 86% 86% 86% 86% 86	L SALES Measure: Amo L Version: Actu L Region: Cana Columns L L Account: Sales, Cost of Rows L Month: January, Febr Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into Sales Columns	al Ida of Sales, Margin, Margin Pont uany, March, 1st Quarter, April, May, June, 21 PowerExcel Slice PowerExcel Slice Power Query able(\$B\$1, \$B\$2, \$B\$3:\$E\$3, \$B\$4:\$E A	
Cost of Sal Cost of Sal 2793 2793 2793 2952 3121 3298 9371 3487 3487 3487 1041 2513 1041 2513 1041 2513 1041 2513 1041 2513 1041 2513 1041 2513 1041 2513 1041 2513 1041 2513 1041 2513 1041	les Margin 500 500 500 500 500 500 500 50	Margin Po 0 250 2293 2543 2452 2621 2798 7871 2987 3185 3395 9567 1295 541 2013 3849 2033 3849 2033 3849 2033 2035	cnt 0% 33% 63% 83% 83% 83% 85% 86% 86% 86% 86% 86% 86% 86% 86% 86% 86	L: Version: Actu L: Region: Cana Columns L: Account: L: Account: Sales,Cost of Rows L: Month: Jone Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into Sales	al Ida of Sales, Margin, Margin Pont uany, March, 1st Quarter, April, May, June, 21 PowerExcel Slice PowerExcel Slice Power Query able(\$B\$1, \$B\$2, \$B\$3:\$E\$3, \$B\$4:\$E A	
500 750 2793 4043 2952 3121 3298 9371 3487 3685 3895 1001 2513 5349 2513 5349 2513 5349 2533 5349 29830 4 2014 2014 2014 2014 2014 2014 2014 2	500 500 500 500 500 500 500 500	0 250 2293 2543 2452 2621 2798 7871 2987 3185 3395 9567 1295 541 2013 3849 2013 3849 20330 C C DN ES	0% 33% 82% 63% 83% 84% 88% 84% 86% 86% 86% 86% 86% 86% 80% 72% 80% 72% 80%	L: Region: Cana Columns Columns Account: Sales,Cost of Rows Month: Month: January,Febr Constrain EmptyRows Delete Removed Rows Delete Removed Rows Expandable Members Insert Into Sales	ida of Sales, Margin, Margin Pont uary, March, 1st Quarter, April, May, June, 2r PowerExcel Silce PowerExcel Silce Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E A	
500 750 2793 4043 2952 3121 3298 9371 3487 3685 3895 1001 2513 5349 2513 5349 2513 5349 2533 5349 29830 4 2014 2014 2014 2014 2014 2014 2014 2	500 500 500 500 500 500 500 500	0 250 2293 2543 2452 2621 2798 7871 2987 3185 3395 9567 1295 541 2013 3849 2013 3849 20330 C C DN ES	0% 33% 82% 63% 83% 84% 88% 84% 86% 86% 86% 86% 86% 86% 80% 72% 80% 72% 80%	Columns	rf Sales, Margin, Margin Pent uary, March, 1st Quarter, April, May, June, 2i PowerExcel Silce PowerExcel Silce PowerExcel Silce Power Query able(\$B\$1, \$B\$2, \$B\$3: \$E\$3, \$B\$4: \$E	
500 750 2793 4043 2952 3121 3298 9371 3487 3685 3895 1001 2513 5349 2513 5349 2513 5349 2533 5349 29830 4 2014 2014 2014 2014 2014 2014 2014 2	500 500 500 500 500 500 500 500	0 250 2293 2543 2452 2621 2798 7871 2987 3185 3395 9567 1295 541 2013 3849 2013 3849 20330 C C DN ES	0% 33% 82% 63% 83% 84% 88% 84% 86% 86% 86% 86% 86% 86% 80% 72% 80% 72% 80%	Account: Sales, Cost of Rows Annth: January, Febr Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into fr =OLAPivotT	uary, March, 1st Quarter, April, May, June, 2i PowerExcel Slice PivotTable Read/Write Formulas Power Query able(\$B\$1, \$B\$2, \$B\$3: \$E\$3, \$B\$4: \$E	
750 2793 4043 2952 2952 3121 3298 9371 3487 3685 3895 3895 3895 3895 1067 1795 1041 2513 5349 29330 404 2513 5349 29330 404 2513 5349 29330 404 2052 2	500 500 500 500 500 500 500 500	250 2293 2543 2452 2621 2798 7871 2987 3185 3395 9567 1295 541 2013 3849 2033 2033 2033 2033 2035 2045 20	33% 82% 63% 83% 84% 85% 86% 86% 86% 86% 86% 86% 72% 52% 80% 72% 80%	Account: Sales, Cost of Rows Annth: January, Febr Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into fr =OLAPivotT	uary, March, 1st Quarter, April, May, June, 2i PowerExcel Slice PivotTable Read/Write Formulas Power Query able(\$B\$1, \$B\$2, \$B\$3: \$E\$3, \$B\$4: \$E	
2793 4043 2952 3121 3298 9371 3487 3685 3995 1067 1795 1041 2513 5349 29830 40 Cube:	500 1500 500 500 500 500 500 500	2293 2543 2452 2621 2798 7871 2987 3185 3395 541 2013 3849 2013 3849 23830	82% 63% 83% 84% 85% 84% 86% 86% 86% 86% 86% 86% 87% 86% 80% 72% 80%	Account: Sales, Cost of Rows Annth: January, Febr Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into fr =OLAPivotT	uary, March, 1st Quarter, April, May, June, 2i PowerExcel Slice PivotTable Read/Write Formulas Power Query able(\$B\$1, \$B\$2, \$B\$3: \$E\$3, \$B\$4: \$E	
4043 2952 3121 3298 9371 3487 3665 3895 1067 1795 1041 2513 5349 25349 1041 2513 1041 2513 1041 2513 1041 2513 1041 2513 1041 2513 1041 2513 1041 2513 1041 1041 1057 1041 1057 1041 1057 1041 1057 1041 1057 1041 1057 1041 1057	1500 500 500 500 500 500 500 500	2543 2452 2621 2798 7871 2987 3185 3395 9567 1295 541 2013 3849 2013 3849 2013 2014 2015 541 2015 541 2013 2014 2015 2015 2015 2015 2015 2015 2015 2015	63% 83% 84% 85% 84% 86% 86% 86% 86% 87% 86% 80% 72% 80% 72% 80%	Rows A Month: January, Febr C Month: January, Febr C Options Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into	uary, March, 1st Quarter, April, May, June, 2i PowerExcel Slice PivotTable Read/Write Formulas Power Query able(\$B\$1, \$B\$2, \$B\$3: \$E\$3, \$B\$4: \$E	
2952 3121 3298 9371 3487 3685 3695 3695 1067 1795 1041 2513 5349 22830 4 A Database: Cube:	500 500 500 500 500 500 500 500	2452 2621 2798 7871 2987 3185 3395 9567 1295 541 2013 3849 2013 3849 2033 0 C DN ES	83% 84% 85% 86% 86% 86% 87% 86% 77% 86% 72% 80%	A Month: January, Febrer Options Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into fr =OLAPivotT	PowerExcel Slice PivotTable Read/Write Formulas Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E	
3121 3298 9371 3487 3685 3895 1067 1795 1041 2513 5349 29830 	500 500 500 500 500 500 500 500	2621 2798 7871 2987 3185 3395 9567 1295 541 2013 3849 2013 3849 23830 C	84% 85% 84% 86% 86% 86% 86% 72% 52% 80% 72% 80%	A Month: January, Febrer Options Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into fr =OLAPivotT	PowerExcel Slice PivotTable Read/Write Formulas Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E	
3298 9371 3487 3685 3895 1067 1795 1041 2513 5349 29830 4 29830 4 200 200 200 200 200 200 200 200 200	500 1500 500 500 500 500 500 500	2798 2787 2887 3185 3395 9567 1295 541 2013 3849 23830 C C DN ES	85% 84% 86% 86% 87% 86% 52% 80% 72% 80%	A Month: January, Febrer Options Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into fr =OLAPivotT	PowerExcel Slice PivotTable Read/Write Formulas Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E	
9371 3487 3685 3895 1067 1775 1041 2513 5349 29830 	1500 500 500 500 500 500 500 500	7871 2987 3185 3395 9567 1295 541 2013 3849 20330 20830 C C	84% 86% 86% 87% 86% 52% 80% 72% 80%	A Month: January, Febrer Options Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into fr =OLAPivotT	PowerExcel Slice PivotTable Read/Write Formulas Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E	
3487 3685 3895 11067 1795 1041 2513 5349 99830 10 10 10 2513 5349 29830	500 500 500 500 500 500 500 500	2987 3185 3395 9567 1295 541 2013 3849 3849 3830 C DN ES	86% 86% 87% 86% 72% 52% 80% 72% 80%	 Options Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into fr =OLAPivotT 	PowerExcel Slice PivotTable Read/Write Formulas Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E	
3685 3895 11067 1795 1041 2513 5349 99830 10 10 A Database: Cube:	500 500 500 500 500 500 1500 6000 20 8 USING_OLATIO PRODUCT_SAL	3185 3395 9567 1295 541 2013 3849 3830 C DN ES	86% 87% 86% 72% 52% 80% 72% 80%	 Options Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into fr =OLAPivotT 	PowerExcel Slice PivotTable Read/Write Formulas Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E	
3895 1067 1795 1041 2513 5349 29830 10 A Database: Cube:	500 1500 500 500 1500 6000 20 B USING_OLATIO PRODUCT_SAL	3395 9567 1295 541 2013 3849 23830 C DN ES	87% 86% 72% 52% 80% 72% 80%	Options Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into f_{π} =OLAPivotT	PowerExcel Slice PrivotTable Read/Write Formulas Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E	E\$4,\$B\$5:\$E\$5,\$B\$6:\$E\$6,
11067 1795 1041 2513 5349 29830 10 10 A Database: Cube:	1500 500 500 500 1500 6000 2 B USING_OLATIC PRODUCT_SAL	9567 1295 541 2013 3849 23830 C DN ES	86% 72% 52% 80% 72% 80%	Options Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into f_{π} =OLAPivotT	PowerExcel Slice PrivotTable Read/Write Formulas Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E] = \$4,\$B\$5:\$E\$5,\$B\$6:\$E\$6,
1795 1041 2513 5349 29830 A10 A Database: Cube:	500 500 500 1500 6000 2 B USING_OLATIO PRODUCT_SAL	1295 541 2013 3849 23830 C DN ES	72% 52% 80% 72% 80%	Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into	PivotTable Read/Write Formulas Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E	E\$4,\$B\$5:\$E\$5,\$B\$6:\$E\$6,
1041 2513 5349 29830 10 A Database: Cube:	500 500 1500 6000 2 B USING_OLATIO PRODUCT_SAL	541 2013 3849 23830 C DN ES	52% 80% 72% 80%	Constrain Empty Rows Delete Removed Rows Expandable Members Insert Into	PivotTable Read/Write Formulas Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E	⊑\$4,\$B\$5:\$E\$5,\$B\$6:\$E\$6,
2513 5349 29830 A10 Database: Cube:	500 1500 6000 2 B USING_OLATIO PRODUCT_SAL	2013 3849 23830 C DN ES	80% 72% 80%	Delete Removed Rows Expandable Members Insert Into f_r =OLAPivotT	Read/Write Formulas Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E	¢
5349 29830 A10 Database: Cube:	B USING_OLATIO PRODUCT_SAL	3849 23830 C DN ES	72% 80%	Expandable Members Insert Into X fr =OLAPivotT	Power Query able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E	E\$4,\$B\$5:\$E\$5,\$B\$6:\$E\$6,
29830 A10 Database: Cube:	6000 2 B USING_OLATIC PRODUCT_SAL	23830 C DN ES	80%	Insert Into	able(\$B\$1,\$B\$2,\$B\$3:\$E\$3,\$B\$4:\$E	E\$4,\$B\$5:\$E\$5,\$B\$6:\$E\$6,
A10 A Database: Cube:	B USING_OLATIO PRODUCT_SAL	C DN ES	• : :	× ✓ f _x =OLAPivotT		E\$4,\$B\$5:\$E\$5,\$B\$6:\$E\$6,
A Database: Cube:	USING_OLATIO	DN .ES		× ✓ f _x =OLAPivotT		E\$4,\$B\$5:\$E\$5,\$B\$6:\$E\$6,
A Database: Cube:	USING_OLATIO	DN .ES				E\$4,\$B\$5:\$E\$5,\$B\$6:\$E\$6,
	FILLER	PRODUCT_SAL	. Members	Amount	Database USING_OLATION	~
	Filter	_	Members	Actual	Cube	~
	Filter	Region	Members	Canada	PRODUCT_SALES	~
	Filter		Members	Current Product		
	Column	Account	Range	\$B\$11:\$E\$11	Filters	
	Row	Month	Range	\$A\$12:\$A\$28	2. PRODUCT_SALES Measur	Actual
					L'A Region:	Canada
) OLAPivotTable	<u>.</u>]				12 Product:	Current Product
	Sales	Cost of Sales	Margin	Margin Pcnt		
January	500	500	0	0.00		
February	750	500	250	0.33		
March	2793	500	2293	0.82	Account: Sales,Cost of S	Sales, Margin, Margin Pcnt
1st Quarter	4043	1500	2543	0.63		
i April	2952	500	2452	2 0.83		
May	3121	500	2621	L 0.84		
June	3298	500	2798	0.85		
2nd Quarter	9371	1500	7871	0.84	li⊂ Month: January,Februar	ry, March, 1st Quarter, April, May, Ju
) July	3487	500	2987	7 0.86		
August	3685	500	3185	5 0.86		
September	3895	500	3395	5 0.87		
3rd Quarter	11067	1500	9567	7 0.86	Options	PowerExcel Slice
October	1795	500	1295	5 0.72	Constrain Empty Rows	PivotTable
November	1041	500	541	0.52	Delete Removed Rows	Read/Write Formulas
December	2513	500	2013	8 0.80	Expandable Members	O Power Query
4th Quarter	5349	1500	3849	0.72		
Total Quarter	29830	6000	23830	0.80		
					🔘 New Workbook 🛛 🖲) Current Worksheet
)					New Worksheet	Location: \$4\$1
< > S			÷ .		- IIIUpdate	
	January February March 1st Quarter April May 2nd Quarter July August September 3rd Quarter October November December 4th Quarter Total Quarter	Sales January 500 February 750 March 2793 Ist Quarter 4043 April 2952 May 3121 June 3298 2nd Quarter 9371 July 3487 September 3895 3rd Quarter 11067 October 1795 November 1041 December 2513 4th Quarter 5349 Total Quarter 29830	Sales Cost of Sales January 500 500 February 750 500 March 2793 500 Lst Quarter 4043 1500 April 2952 500 May 3121 500 June 3298 500 2nd Quarter 9371 1500 July 3487 500 September 3895 500 September 1067 1500 October 1795 500 November 1041 500 December 2513 500 Total Quarter 29830 6000	Sales Cost of Sales Margin January 500 500 0 February 750 500 255 March 2793 500 2293 1st Quarter 4043 1500 2263 April 2352 500 2455 May 3121 500 2621 June 3298 500 2798 August 3685 500 3185 September 3895 500 3395 3rd Quarter 11067 1500 9563 October 1795 500 1295 November 1041 500 544 December 2983 3600 2013 Ath Quarter 1295 500 3395 November 1067 1500 544 December 2513 500 23830 Ath Quarter 5349 1500 3843 Total Quarter 29830 6000	Sales Cost of Sales Margin Margin Pcnt January 500 500 0 0.00 February 750 500 229 0.82 March 2793 500 2543 0.63 April 2952 500 2452 0.83 May 3121 500 2452 0.83 June 3298 500 2452 0.83 June 3298 500 2798 0.85 July 3487 500 2987 0.85 July 3487 500 3395 0.86 August 3685 500 3395 0.86 September 3895 500 3395 0.87 Std Quarter 11067 1500 9567 0.86 October 1795 500 1295 0.72 November 1041 500 541 0.52 December 2513 500 3849	Sales Cost of Sales Margin Margin Pcnt Image: Columns January 500 500 0 0.00 Image: Columns <

Slices from the SALES cube and the PRODUCT_SALES cube showing Canada, Actuals.

5. If, in the Filters box, we change the *Product* Member to show any other Members (e.g., *New Product 1*, as in the example image below), notice that these intersections show empty values:

File Home	Insert I	Page Layout	Formulas	Data Revie	w View	Develope	r Help Po	werExcel $ ho$ Search	ß	\square
O Den Sa		ebar OLA Fx		ions Options Cl Ca	che Dimensio	n Powe	out Change rExcel Password			
	PowerExcel Slice	2	Powe	rExcel Connectior	n Dimension	r l	Help			^
A10				×	fx =OLAPiv	otTable(\$	\$B\$1,\$B\$2,\$B\$3:\$	E\$3,\$B\$4:\$E\$4,\$B\$5:\$E\$5,\$B	\$6:\$E\$6,\$B\$7:\$E\$7,\$B\$8:	~
А	В	с	D	E	F	G				
1 Database:	USING_OLA	-	D	E	F	6	H	PowerExcel	-	×
2 Cube:	PRODUCT S									
3 Dimensions:	Filter	PRODUCT_SA	Members	Amount				Database		
4	Filter	Version	Members	Actual				USING_OLATION	~	Y
5	Filter	Region	Members	United States				Cube		
6	Filter	Product	Members	New Product				PRODUCT_SALES	~	¥.
7	Column	Account	Range	\$B\$11:\$E\$11	-			T Filters		_
8	Row	Month	Range	\$A\$12:\$A\$28				ビ PRODUCT_SALES Measure ビ Version:	e: Amount Actual	
9								L. Region:	United States	
10 OLAPivotTabl	e							12. Product:	New Product 1	
11	Sales	Cost of Sales	Margin	Margin Pcnt						
12 January		0 0	-	0 0						
13 February		0 0		0 0				🌄 Columns		
14 March		0 0	(0 0				ピ Account: Sales,Cost of S	ales, Margin, Margin Pcnt	
15 1st Quarter		0 0	(0 0						
16 April		0 0	(0 0						
17 May		0 0	1	0 0						
18 June		0 0	(0 0				Rows		_
19 2nd Quarter		0 0		0 0				🗠 Month: January,Februar	/,March,1st Quarter,April,May,Ju	ine,2i
20 July		0 0		0 0						
21 August		0 0		0 0				<		>
22 September		0 0		0 0						-
23 3rd Quarter		0 0		0 0				Options	PowerExcel Slice	
24 October		0 0						Constrain Empty Rows	PivotTable	
25 November		0 0		0 0				Delete Removed Rows	Read/Write Formulas	
26 December		0 0		0 0				Expandable Members	O Power Query	
27 4th Quarter		0 0		0 0				Insert Into		
28 Total Quarter		0 0	(0 0					CurrentWorksheet	
29										¥.
30								New Worksheet	Location: \$A\$1	* <u>.</u>
31								Update		
32								Copuace		
	Sheet1	+		: •						
		<u> </u>						av Settings 🗰 🗉 🖽	□+ 1	00%
alculate Recover	eu 🔠						TO DISPI	ay Settings 🔠 🗉 🗄	- + -	GQ 20

The Finance Director can use these "templates"—for different products (the "new" products in the PRODUCT_SALES cube)—providing access to a shared Olation planning model for users across all regions.

In sum, we have demonstrated the creation of different data models/Cubes, all working together dynamically, for the various purposes of planning, analytics, and reporting.