

Knowledge Base Article: Article 255 How to use the new Click&DECiDE Queries Data Source? (Allowing queries on various data sources at the same time)

Last Update: February, 16th, 2017

Requirement:

You need to install Click&DECiDE Version 16 (64-bit) or greater.

Introduction:

Click&DECiDE version 16 provides a new data source named "Click and DECiDE Queries" allowing to use the queries from a Click&DECiDE project file (*.wfv) as a Data Source. If that project contains several queries made against various data sources, you can create a new query using each existing query as if it was a table. You can thus join data coming from different data sources. For example, one query has been made against an iSeries data source, another query has been made against an Oracle data source, another one against an SQL Server data source or against an Excel data source.

Targets:

This new data source has two targets:

- 1. To be able to make queries against various data sources at the same with the possibility to define some caches, scopes or index. The query result is stored in some hidden tables.
- 2. To be able to create a data warehouse using the Click&DECiDE automatic ETL feature whose target is to prepare for end-users some « synonyms », that means some views pointing to hidden tables defined in point 1.

1- Direct « Click and DECiDE Queries » data source:

Cette partie est plutôt réservée à des utilisateurs avertis ou aux administrateurs.

- The first step consists of using the new « Click and DECiDE Queries » data source to define some caches for the result (in various hidden tables) being updated according to the cache parameters, the « scopes » option (result set in cache by user or Globally) and with optional indexes.
- The second step consists of using this new « Click and DECiDE Queries » data source to join X queries from different data sources. By default, if « caches » or « scopes » or « index » are not defined, the result for each will be set in cache for 5 minutes by user.

2- Create and use synonyms with « Click and DECiDE Queries » data source:

- In the above target point 2 some advanced users or administrators can use this new « Click and DECiDE Queries » data source to create some synonyms, from single or multi data source queries.
- The synonyms are « views » pointing to the result saved with a cache for each table used in the query.
- The benefit to use synonyms is to give a unique name to this result, easy to understand by the end-users.
- The end-users will use a standard SQL Server data source to access to these synonyms.
- Note that in the synonyms, some of them can be created from within an iSeries, other from Excel or SQL Server etc., and that the end-user working with these synonyms can create queries against various data sources without knowing that, because all synonyms are made under SQL Server.

Use:

1- Create the Click and DECiDE Queries data source:

- 1. Open Click&DECiDE Administration Manager and select the Data Source Tab.
- 2. Make a right mouse click and select New Data Source.
- 3. Enter a name and select the Click and DECiDE Queries data source in the list box:

Click&DECiDE

New Data Source	×
Data Source Name: Query Builder	OK Cancel
Data Source:	
	~
ADO Multi Dimensional (MDX) Click and DECiDE Connect Click and DECiDE Model	
Click and DECIDE Queries Excel Google BigQuery Wa Infor M3 nee iSeries TCP/IP not ODBC dist OLEDB (MDX and other) Power BI SAP SQL Server	, Oracle, DB2-UDB etc.) ed first. If this module is ssing file may be
TeraData Web Service	

- 4. Validate with OK
- Specify the working directory where Builder can find the appropriate project files (*.wfv), for example a shared folder from the Web Portal or a directory in the server not visible through the Web Portal): Example: C:\CnD\My projects or Builder or

C:\Users\Public\Documents\Click and DECiDE Samples\Web Portal\Shared Folder)

m BAI Builder Source Configuration [Query Builder]				×
General Cache				
Projects Directory	Preven			
C:\Users\Public\Documents\Ulick and DEUDE Samples\Web Portal\Shared Folder	browse			
	OK	Cancel	Appl	v

6. In the Cache Tab, you can specify if you wish to enable some options:

🛷 BAI Builder Source Configuration [Query Builder]		×
General Cache		
Enable Indexing Support Enable Synonyms Support		

Enable Indexing Support: means that you want to be able to define some indexes for some fields.

Enable Synonyms Support: means that you want to be able to have access later to an end-user table corresponding to the result of the query you have created. That option will be described in the second part of this manual *2- Create and use synonyms with the « Click and DECiDE Queries »data source.*

www.clickndecide.com



- 7. Validate with OK. Make a right mouse click on this new "Query Builder" data source and click "Test connection".
- 8. If you are using Windows Authentication, just click OK.
- 9. If you are using Click&DECiDE Authentication, enter your UserID and password then click OK.
- 10. If this data source has to be used through the Web Portal, define it as Secured by Click&DECiDE:

Eile Edit View Tools Help	
: 🖸 X 🖻 🛍 🌱 🍽 🞯 : 🌺 🐼 🛃 🦚 🎎	
Data Sources	Data Source: ???
BigQuery ConnectSQL Excel32 Excel64	Name: Query Builder Properties Use a unique User ID/Password by default.
Iseries Local SQL Server Server Query Builder	Identification User ID: Password: Confirmation:
	Data Source Secured by Click and DECIDE Click and DECIDE Security Locked Access: Password Checked by the Data Source: Prevent Web User Defining ID: Max number of simultaneous connection:

- 11. Activate "Use a unique User ID/Password by default », without the need to enter a login if under Windows Authentication.
- 12. Add this secured data source the concerned User Groups.
- 13. You can add the full data source, that means automatically all projects being or that will be in the working directory, or only some of the Builder projects that will appear in that list:

Click and DECiDE Administration Manager [Mode: Windows]	_		\times
File Edit View Tools Help Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image: Comps Image:	All Builer pro located in th directory wil here.	ojects e specifi Il appea	ied r
User Groups 🔥 User's 🚎 Menus 👎 Authentications Ready	0	APS NUM S	CRL

- 14. Click File > Update Security from Click&DECiDE Administration Manager.
- 15. Note that the Click and DECiDE Queries data source can access several queries saved in a same project or saved in distinct project files.





Use from Click&DECiDE Builder

> Case of a query source having no parameters.

- 1. Start Builder (or Web Builder from the Web Portal)
- 2. Open or create a project (for example My Queries Multi Sources.wfv) file.
- 3. If the "Query Builder" data source has been enabled to the User Group you belong to, you should see the authorized project files available, or for all Groups you belong to.
- 4. Create a new query against this « Query Builder » data source:

Select the required project to display the existing queries, as table-sources, for example the project containing Excel queries:

ididier-portable/BAI Demonstration/Shared Folder/My Queries Mutli Sources.wfy				
All Data Sources	Content of [C)uery Builder]:"Excel quer		
ConnectSQLServer Excel32 ConnectSQLServer Excel64 Constraints Con	A Name Sales sales_deta salesman	Description Sales nils Detailed sales Salesmen		
🖶 🛅 Oracle queries	y <	>		
salesman				
Name Type Length Scal	le Description			
SAL Float 53	0 Salesman code			
SALINAIVIE Varchar 255	Salesman name			
AREA Varchar 255	Salesman area			
NameTypeLengthScalSALFloat53SALNAMEVarchar255MAILVarchar255AREAVarchar255	le Description 0 Salesman code Salesman name Salesman e-mail Salesman area			

5. Note that the Description only appears on the right side if you have entered e text in the "Description" box in the Properties General Tab for the concerned query in the Builder project:

🔟 didier-porta	able/BAI Demonstration/Shared Folder/E	xcel queries.wfv	▼ ×	Properties		ά×
	🔩 New 📁 Open 🛛 🗙 Delete			4 General	Properties OData	Þ
	Name 🛆	Data Source	Description	Name	salesman	<u> </u>
	📑 Sales	Excel64	Sales	Data Source	Excel64	
	sales details	Excel64	Detailed sales	Description	Salesmen	
	📴 salesman	Excel64	Salesmen	Lreated	01/1//2017 1:53:24 PM	
Queries				Modified	01/19/2017 4:08:57 PM	
				Author	Didier	

6. Select a table from this data source, for example "Salesman" from the "Excel queries" data source. Make a drag and drop this table. Then select another data source and another table from this other data source, for example "Sales" from the "SQL Server queries" data source:

www.clickndecide.com



📝 📑 Excel salesmen and SQL Server sales* 🔟 didier-portable/BAI Demonstration/Shared Folder/My Queries Mutli Sources.wfv* 💿 🔻 🗙									
All Data Sources		Co	ontent of [Qu	ery Build	er]:"SQL Serve	r queries"			
Iseries A Name Description Local SQL Server Clical SQL Server Ouery Builder Click and DECiDE Web Demonstration Click and DECiDE Web Demonstration Product-Family-Category Excel queries SQL Server Sales My Queries Mutti Sources Sales_details My Queries Mutti Sources Salesman My Queries Mutti Sources Salesman Project UNION State SQL Server queries State									
salesman				sales		Д.			
Name Type Length	Scale Description	on		Name	Туре	Length	Scale	Description	
SAL Float 53	0 Salesmar	n code	┏╴┐│	NO	Varchar	5		Invoice number	
SALNAME Varchar 255	Salesmar	n name		CUST	SmallInt	2	0	Customer code	
MAIL Varchar 255	Salesmar	n e-mail		SAL	SmallInt	2	0	Salesman code	
AREA Varchar 255	Salesmar	n area		DATE	TimeStamp	8		Order Date	
				TOTAL	Float	8	0	Amount	

7. Define the join condition(s) if needed, and switch to the Query Tab. The created query appears as follow: Click and DECiDE Builder - Excel salesmen and SQL Ser ver cales*

-97)1	Click and DECIDE I	Builder - Excel salesme	n and SQL Server sales"		— L	
÷ <u>в</u>	ile <u>E</u> dit <u>V</u> iew	<u>Query Layout Tools</u>	s <u>W</u> indow <u>H</u> elp			
1	s 🗔 📾 🖾 🖎		ት 🔹 🔿 🖃 🐼 🖬 🐘 😫			
ten	Excel sales	men and SQL Server sa	es*			7 X
A	🔍 🔍 💵 66'	Name	Туре	Len 🔺 🔻		^
st	😑 salesman.para	ameters				
		<caching></caching>	Varchar	32	0 Caching mode: ON or empty	(d
		<scope></scope>	Varchar	32	0 Caching scope: Empty: by us	se
		<index></index>	Varchar	4000	0 Indexes: list of IX(IndexName	4, °
	🗆 salesman					
		SAL	Float	53	0 Salesman code	-
		SALNAME	Varchar	255	0 Salesman name	
		MAIL	Varchar	255	0 Salesman e-mail	
		AREA	Varchar	255	0 Salesman area	
	😑 sales.paramet	ters				
		<caching></caching>	Varchar	32	0 Caching mode: ON or empty	(d
		<scope></scope>	Varchar	32	0 Caching scope: Empty: by us	se
		<index></index>	Varchar	4000	0 Indexes: list of IX(IndexName	4, °
	sales					
		NO	Varchar	5	0 Invoice number	-
		CUST	Smallint	2	0 Customer code	
		SAL	Smallint	2	0 Salesman code	
	▌┝╼ <u></u> ╺╞╼ <u></u> ╺╞╸	DATE	TimeStamp	8	3 Order Date	*
		TOTAL	Float	8	0 Amount	Ŧ
	<				2	>
	♦ Tables Quer	y SQL				⊳
Rea	dy				CAP NUM ON	/R

- The « salesman » table is coming from an Excel data source.
 The « sales » table is coming from a SQL Server data source.
- 10. Each table appears with a « Parameter » level and a « Table » level.



- 11. Note that in the above example the queries « sales » and « salesman » did not contain any parameter already defined. This means that al data will be saved in the corresponding hidden tables according to the cache values.
- 12. Run the above query. Automatically one or several temporary tables containing the result are created in the Server SQL Server database.
- 13. We will explain later on that manual the incidence of the already existing parameters in the queries sources feeding a multi-data source query.

Parameter Level

This level contains some options to improve the performances:

<Caching> Mode:

The idea is to update the hidden tables through a Scheduled Task according to the specified cache.

- > ON or empty (default): use or update cache, default value is 5 minutes. After that time the table is purged and deleted unless a scheduled task has recreated it.
- **OFF**: do not use cache.
- > **<Positive integer>:** update cache if needed and keep for **<integer>** minutes. After that delay the table is purged and deleted unless a scheduled task has recreated it.
- <Negative integer>: always update cache and keep for <integer> minutes. You need to specify the Scope (Global recommended in that case).
- \geq permanent: always update cache, keep 10 years (similar to the previous one but valid 10 years).

In the above example, we will define the cache as Permanent for both tables Salesmen and Sales:

📑 Criteria		×
Columns Operators Value Expression sales.parameters. = permanent Where Where]
Salesmen.parameters. <caching> = 'permanent' and sales.parameters.<caching> = 'permanent'</caching></caching>	Not Combine Uncombin	e 1e
OK Cancel <u>N</u> ew <u>R</u> emove <u>P</u> arameters		

<Scope>

The idea is to define who can use the result table. Global is recommended but empty means "by User" by default.

- Empty: by user by default.
- ➢ User: by user.
- ➢ Global: for all users.
- Memory: in memory for current session only.

In the above example, we will define the Scope as Global for both tables Salesmen and Sales:



📑 Criteria		×
Columns Operators Value Expression sales.parameters. <scope> = Global</scope>		
salesman.parameters. <caching> = 'permanent' and sales.parameters.<caching> = 'permanent' and salesman.parameters.<scope> = 'Global' and sales.parameters.<scope> = 'Global'</scope></scope></caching></caching>	Not Combi Uncomb	ne
OK Cancel New Remove Parameters		

<Index>

The idea is to be able to add one or several indexes on the table containing data in the cache, giving best performances to future queries.

- list of IX(IndexName, <fields>) = Index
 - IndexNAme = index name
 - <Field> = column name between 2 double-quotes if the name contains a space or special characters.
 - o If several indexes, separate them with a semi-colon
- VKC(IndexName, <fields>) = Unique Index
 - IndexNAme = Unique Index name
 - <Field> = column name between 2 double-quotes if the name contains a space or special characters.

Example: IX(Index1, CUST); IX(Index1, ""NO""); UKC(UniqueIndex1, SAL);

Case of query source already containing one or several parameters

Supposing that one of the queries from the previous example was already containing a parameter on the DATE field with a list of period, such as keywords.

Query Paramete	X zr
Parameters:	👛 🗙 🗲 🐰 🛍 🛍 🌠
P_DATE	
	Parameter Properties
Name	P_DATE
Туре	Timestamp
Status	Enable
Update Method	Input List
Title	Choose value for DATE
Mandatory	No
Editable	No
Default Value	IGNORE
Value List	in 2 days Day after tomorrow;in 1 days Tomorrow;0 days ago Today;1 days ago Y
Selection	Single
Hide List's First Column	Yes
	OK Cancel

Contact us: E-mail: <u>support@clickndecide.com</u> Tel: +33 (0)4 67 84 48 00 Headquarters: Click&DECiDE 130, rue du Baptistou - 34980 St Gély du Fesc, France. To contact your nearest Click&DECiDE partner, <u>click here</u>.

www.clickndecide.com

Business Application Intelligence | Promotions

Click&DECiDE

If we redo the multi data source query by joining the Excel (« salesman » table) with the SQL Server (« sales » table) and if this « Sales » query contains a parameter on the DATE filed, we will get this result:

all.	Click an	d DECiDE I	Builder - Query*					_		\times
Ei	le <u>E</u> dit	View	Query Layout To	ols <u>W</u> indow <u>H</u>	<u>H</u> elp					
	: 🗆 a		1 X B. (9.)	 [2]]]]]]]]	 • 🛹 👒 📰 🏊	a m				
						30° U-1				
ten		didier-por	table/BAI Demonstra	tion/Shared Folde	r/My Queries Mutli Sour	ces.wfv 📑	Query*			• X
A	Q	, 11 66	Name		Туре	Le 🔺	-	1		^
st	🗆 sale	sman.para	ameters							
			<caching></caching>	Varchar		3	2 (Caching mode	: ON or en	npty (
			<scope></scope>	Varchar		33	2 (Caching scop	e: Empty: t	by us
			<index></index>	Varchar		400	o () Indexes: list o	f IX(IndexN	lame,
	🗆 sale	sman								
			SAL	Float		5	3 0	Salesman coo	le	
			SALNAME	Varchar		25	5 0) Salesman nar	ne	
			MAIL	Varchar		25	5 C) Salesman e-n	ail	
			AREA	Varchar		25	5 0) Salesman are	а	
	🗆 sale	s.paramet	ters							
			<caching></caching>	Varchar		33	2 (Caching mode	: ON or en	npty (
			<scope></scope>	Varchar		33	2 () Caching scop	e: Empty: t	oy us
			<index></index>	Varchar		400	0 0) Indexes: list o	f IX(Index)	lame,
	<u> </u>		P_DATE	Varchar			0 0)		
	🗆 sale	S								
			NO	Varchar		:	5 0) Invoice number	er	
			CUST	SmallInt		:	2 (Customer cod	е	
			SAL	SmallInt		:	2 0) Salesman coo	le	~
			DATE	TimeStamp		1	8 3	Order Date		*
			TOTAL	Float			в () Amount		Ŧ
	<									>
	∢ \ Tab	les Quer	y SQL							⊳
Read	tv							0		OVR

In the « parameters » branch for the « Sales » table we can see a new row with the P_DATE parameter. Although this parameter was a Date or Date Time Type, it appears here as a String because the value you can enter will be translated by the current multi data source query. For example, if you enter as criteria 2017, it will be translated with an interval including all dates for the year 2017. If you enter « Last Month » or « 1 months ago », i twill be translated with an interval including all days of the last month etc.

If we add criteria on the left side of this parameter and if we run the query, only the records matching the criteria condition will be saved in the hidden temporary tables, according to the cache and scope values.

In case of Cache not defined, the default value will be 5 minutes and in case of Scope not defined, the default value will be "by user".

For example, we enter « Last Month » or « 1 months ago » as criteria for the P_DATE parameter:

📑 Criteria			_		×	
Columns sales.parameters.P_DATE v	Operators = V	Value Expression				
Where sales.parameters.P_DATE = 'Last Month'						
OK Cancel <u>N</u> ew	<u>R</u> emove	Parameters		Uncomb	ne bine	

Contact us: E-mail: <u>support@clickndecide.com</u> Tel: +33 (0)4 67 84 48 00 Headquarters: Click&DECiDE 130, rue du Baptistou - 34980 St Gély du Fesc, France. To contact your nearest Click&DECiDE partner, <u>click here</u>.

www.clickndecide.com



Now we run the query and we get the records for the concerned month:

SAL Float 53 0 Salesman code SALNAME Varchar 255 0 Salesman name MAIL Varchar 255 0 Salesman e-mail MAIL Varchar 255 0 Salesman e-mail Sales.parameters 255 0 Salesman area sales.parameters 32 0 Caching mode: ON or empty (default): use or update ci <ccching> Varchar 32 0 Caching scope: Empty: by user or for all users (chose <index> Varchar 32 0 Caching scope: Empty: by user or for all users (chose <index> Varchar 4000 0 Indexes: list of DX(IndexName, <fields>) = Index or UKC P_DATE Varchar 5 0 Invoice number 1 CUST Smailint 2 0 Castomer code V V NO Varchar 5 0 Invoice number V V CUST Smailint 2 0 Castomer code V V DATE TimeStamp 8 3 Or</fields></index></index></ccching>	sales	man									
SALNAME Varchar 255 0 Salesman name MAL Varchar 255 0 Salesman e-mail AREA Varchar 255 0 Salesman area sales.parameters Salesman area <coching> Varchar 32 0 Caching mode: ON or empty (default): use or update c: <coching> Varchar 32 0 Caching scope: Empty: by user or for all users (chose <index> Varchar 4000 0 Indexes: list of DX(IndexName, <fields>) = Index or UKC </fields></index></coching></coching>		SAL	Float				53	0 Salesman o	ode		
AREA Varchar 255 0 Salesman e-mail sales.parameters 0 Caching mode: ON or empty (default): use or update ci. 		SALNAME	Varchar				255	0 Salesman r	ame		
AREA Varchar 255 0 Salesman area sales.parameters Caching mode: ON or empty (default): use or update c <dscope> Varchar 32 0 Caching mode: ON or empty (default): use or update c <dscope> Varchar 32 0 Caching scope: Empty: by user or for all users (chose <dscope> Varchar 4000 0 Indexes: list of K/(indexName, <fields>) = Index or UKC P_DATE Varchar 0 0 sales CuST Smallint 2 0 Castesman code V NO Varchar 5 0 Invoice number SAL Smallint 2 0 Salesman code V NO Varchar 8 0 Amount OTATE TimeStamp 8 3 Order Date V V TOTAL Float 8 0 Amount V SAL SALINAME MAIL AREA NO CUST sales.SAL DATE TOTAL 1 Bill Raley bill raley@yopmail.com NORTH</fields></dscope></dscope></dscope>		MAIL	Varchar				255	0 Salesman e	-mail		
sales.parameters <caching> Varchar 32 0 Caching mode: ON or empty (default): use or update ci <scope> Varchar 32 0 Caching scope: Empty: by user or for all users (chose <index> Varchar 4000 0 Indexes: list of IX(IndexName, <fields>) = Index or UKC P_DATE Varchar 0 0 0 sales Varchar 0 0 CUST Smallint 2 0 Customer code SAL Smallint 2 0 Salesman code V TOTAL Float 8 0 Amount C SAL Smallint 2 0 Salesman code V TOTAL Float 8 0 Amount 4 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 12/22/2016 \$82,450.00 2 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16096 1003</fields></index></scope></caching>		AREA	Varchar				255	0 Salesman a	irea		
<caching> Varchar 32 0 Caching mode: ON or empty (default): use or update ci <scope> Varchar 32 0 Caching scope: Empty: by user or for all users (chose <index> Varchar 4000 0 Indexes: list of DX(IndexName, <fields>) = Index or UKC P_DATE Varchar 0 0 sales Varchar 0 0 CUST Smallint 2 0 Customer code SAL Smallint 2 0 Salesman code V TOTAL Float 8 3 Order Date V TOTAL Float 8 3 Amount 4 1 1 Bill Raley bill raley@yopmail.com NORTH-WEST 16095 1003 1 12/22/2016 \$82,450.00 2 1 Bill Raley bill raley@yopmail.com NORTH-WEST 16095 1003 1 12/22/2016 \$89,207.00 3 Jean Martin jean.martin@yopmail.com SOUTH 16091 102 3 12/22/2016 \$102,289.00 5 Jaean Martin <</fields></index></scope></caching>	🗆 sales	.parameters									
<scope> Varchar 32 0 Caching scope: Empty: by user or for all users (chose index> varchar <index> Varchar 4000 0 Indexes: list of DK(IndexName, <fields>) = Index or UKC P_DATE Varchar 0 0 0 0 sales 0 0 0 CUST Smallint 2 0 Customer code XAL Smallint 2 0 Salesman code DATE TimeStamp 8 0 Amount 4 1 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 12/23/2016 \$823,021.815 2 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 12/23/2016 \$83,305.00 3 2 Sandra.davis@yopmail.com NORTH-WEST 16095 1003 1 12/23/2016 \$83,205.00 3 2 Sandra.davis@yopmail.com NORTH-WEST 16095 1003 1 12/23/2016 \$83,205.00 3 2 Sandra.davi</fields></index></scope>		<caching></caching>	Varchar				32	0 Caching mo	de: ON or emp	ty (default): use or update	С
sindex> Varchar 4000 0 Indexes: list of DX((IndexName, <fields>) = Index or UKC P_DATE Varchar 0 0 0 sales 0 0 0 CUST Smallint 2 0 Customer code CUST Smallint 2 0 Salesman code AL Smallint 2 0 Salesman code DATE TimeStamp 8 3 Order Date V TOTAL Float 8 0 Amount SAL SALINAME MAIL AREA NO CUST sales.SAL DATE TOTAL 1 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 12/22/2016 \$82,450.00 2 Sandra Davis sandra.davis@yopmail.com NORTH-WEST 16095 1003 1 12/22/2016 \$93,305.00 3 2 Sandra Bavis sandra.davis@yopmail.com NORTH-WEST 16096 1003 1 12/23/2016 \$93,027.00 3 2 Sandra Bavis Sandra Advis</fields>		<scope></scope>	Varchar				32	0 Caching sc	ope: Empty: by	user or for all users (chos	se
P_DATE Varchar 0 0 ■ sales ■ ✓ N0 Varchar 5 0 Invoice number ✓ CUST Smallint 2 0 Customer code ✓ SAL Smallint 2 0 Salesman code ✓ DATE TimeStamp 8 3 Order Date ✓ TOTAL Float 8 0 Amount 4 ✓ SAL Salex NORTH-WEST 16095 1003 1 1/2/2/2/016 \$82,450.00 ✓ SAL Saltraley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 1/2/2/2/016 \$83,305.00 3 2 Sandra. Davis sandra. davis@yopmail.com NORTH-WEST 16096 1003 1 1/2/2/2/016 \$893,005.00 3 2 Sandra. Davis sandra. davis@yopmail.com NORTH-WEST 16096 1003 1 1/2/2/2/016 \$893,005.00 3 2 Sandra. Davis sandra. davis@yopmail.com NORTH-WEST 16096 1003 1 1/2/2/2/016 \$892,07.00 5 3 Jean Martin jean.martin@		<index></index>	Varchar				4000	0 Indexes: lis	t of IX(IndexNa	ime, <fields>) = Index or Uk</fields>	(C
Sales 5 0 Invoice number ✓ CUST Smallint 2 0 Customer code ✓ SAL Smallint 2 0 Salesman code ✓ DATE TimeStamp 8 3 Order Date ✓ TOTAL Float 8 0 Amount 4 X SALNAME MAIL AREA NO CUST sales.SAL DATE TOTAL I 1 Bil Raley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 12/22/2016 \$82,450.00 2 1 Bil Raley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 12/23/2016 \$83,305.00 3 2 Sandra Davis sandra.davis@yopmail.com NORTH-WEST 16096 1003 1 12/23/2016 \$89,207.00 4 3 Jean Martin jean.martin@yopmail.com SOUTH 16091 1021 3 12/22/2016 \$89,207.00 5 3 Jean Martin jean.martin@yopmail.com SOUTH 16091 1021 3<		P_DATE	Varchar				0	0			
Image: Solution of the second seco	🗆 sales	•									
Image: Cust of the cust		🖌 NO	Varchar				5	0 Invoice nun	nber		-
▲ ✓ SAL SmallInt 2 0 Salesman code ▲ ✓ DATE TimeStamp 8 3 Order Date ✓ ▲ ✓ TOTAL Float 8 0 Amount 4 ✓ TOTAL Float 8 0 Amount 4 ✓ TOTAL Float 8 0 Amount 4 ✓ SAL SALNAME MAIL AREA NO CUST sales.SAL DATE TOTAL 1 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 1/2/2/2/2016 \$82,450.00 2 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16096 1003 1 1/2/2/3/2016 \$93,305.00 3 2 Sandra.Davis sandra.davis@yopmail.com NORTH-WEST 16105 1010 2 1/2/31/2016 \$89,207.00 3 Jean Martin jean.martin@yopmail.com SOUTH 16091 1021 3 1/2/2/2/2016 \$102,289.00 6 4			Smallint				2	0 Customer c	ode		
SAL SALNAME MAIL AREA NO CUS sales.SAL DATE TOTAL Float X 1 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 1/2/22/2016 \$82,450.00 X 1 2/2/23/2016 \$83,050.00 X 1 1/2/23/2016 \$83,050.00 X 3 2 Sandra.davis@yopmail.com NORTH-WEST 16096 1003 1 1/2/23/2016 \$93,305.00 X <t< th=""><th></th><td>SAL</td><td>Smallint</td><td></td><td></td><td></td><td>2</td><td>0 Salesman o</td><td>ode</td><td></td><td></td></t<>		SAL	Smallint				2	0 Salesman o	ode		
SAL SALNAME MAIL AREA NO CUS sales.SAL DATE TOTAL TOTAL NO CUS sales.SAL DATE TOTAL Sales.SAL Sales.SAL DATE TOTAL Sales.SAL Sales.SAL		DATE	TimeStamp				8	3 Order Date			¥
SAL SALNAME MAIL AREA NO CUST sales.SAL DATE TOTAL 1 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 1003 1 1/2/2/2/016 \$82,450.00 2 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16096 1003 1 1/2/2/2/016 \$83,305.00 3 2 Sandra Davis sandra.davis@yopmail.com NORTH-WEST 16105 1010 2 1/2/31/2016 \$89,305.00 4 3 Jean Martin jean.martin@yopmail.com NORTH-WEST 16105 1010 2 1/2/31/2016 \$89,207.00 5 3 Jean Martin jean.martin@yopmail.com SOUTH 16094 1021 3 1/2/2/2/016 \$102,289.00 6 4 Diane Meyer diane.meyer@yopmail.com ATLANTIC 16101 1006 4 1/2/2/2/016 \$197,881.50 7 7 Georges Dunel georges.dunel@yopmail.com WEST 16106 1005		TOTAL	Float				8	0 Amount			*
SAL SALNAME MAIL AREA NO CUST sales.SAL DATE TOTAL 1 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 12/22/2016 \$82,450.00 2 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16096 1003 1 12/23/2016 \$93,305.00 3 2 Sandra Davis sandra.davis@yopmail.com NORTH-WEST 16105 1010 2 12/31/2016 \$230,218.15 4 3 Jean Martin jean.martin@yopmail.com SOUTH 16091 1021 3 12/22/2016 \$89,207.00 5 3 Jean Martin jean.martin@yopmail.com SOUTH 16094 1021 3 12/22/2016 \$102,289.00 6 4 Diane Meyer diane.meyer@yopmail.com ATLANTIC 16101 1006 4 12/24/2016 \$197,881.50 7 7 Georges Dunel georges.dunel@yopmail.com WEST 16106 1005 <td< th=""><th></th><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>Ŧ</td></td<>											Ŧ
SAL SALNAME MAIL AREA NO CUST sales.SAL DATE TOTAL 1 1 Bil Raley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 12/22/2016 \$82,450.00 2 1 Bil Raley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 12/23/2016 \$93,305.00 3 2 Sandra Davis sandra.davis@yopmail.com NORTH-WEST 1605 1010 2 12/31/2016 \$89,305.00 4 3 Jean Martin jean.martin@yopmail.com NORTH-WEST 16105 1010 2 12/31/2016 \$89,207.00 5 3 Jean Martin jean.martin@yopmail.com SOUTH 16091 1021 3 12/22/2016 \$102,289.00 6 4 Diane Meyer diane.meyer@yopmail.com ATLANTIC 16101 1006 4 12/24/2016 \$197,881.50 7 7 Georges Dunel georges.dunel@yopmail.com WEST 16106 100	<										>
1 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16095 1003 1 12/22/2016 \$82,450.00 2 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16096 1003 1 12/23/2016 \$93,305.00 3 2 Sandra.davis@yopmail.com NORTH-WEST 16105 1010 2 12/31/2016 \$83,207.00 4 3 Jean Martin jean.martin@yopmail.com SOUTH 16091 1021 3 12/21/2016 \$89,207.00 5 3 Jean Martin jean.martin@yopmail.com SOUTH 16094 1021 3 12/22/2016 \$102,289.00 6 4 Diane Meyer diane.meyer@yopmail.com ATLANTIC 16101 1006 4 12/24/2016 \$197,881.50 7 Georges Dunel georges.dunel@yopmail.com WEST 16106 1005 7 12/31/2016 \$246,334.29		SAL SALNAME	MAIL	AREA	NO	CUST	sales.SAL	DATE	TOTAL		^
2 1 Bill Raley bill.raley@yopmail.com NORTH-WEST 16096 1003 1 12/23/2016 \$93,305.00 3 2 Sandra Davis sandra.davis@yopmail.com NORTH-WEST 16105 1010 2 12/31/2016 \$230,218.15 4 3 Jean Martin jean.martin@yopmail.com SOUTH 16091 1021 3 12/21/2016 \$89,207.00 5 3 Jean Martin jean.martin@yopmail.com SOUTH 16094 1021 3 12/22/2016 \$102,289.00 6 4 Diane Meyer diane.meyer@yopmail.com ATLANTIC 16101 1006 4 12/24/2016 \$197,881.50 7 7 Georges Dunel georges.dunel@yopmail.com WEST 16106 1005 7 12/31/2016 \$246,334.29	1	1 Bill Raley	bill.raley@yopmail.com	NORTH-WEST	16095	1003	1	12/22/2016	\$82,450.00		
3 2 Sandra Davis sandra.davis@yopmail.com NORTH-WEST 16105 1010 2 12/31/2016 \$230,218.15 4 3 Jean Martin jean.martin@yopmail.com SOUTH 16091 1021 3 12/21/2016 \$89,207.00 5 3 Jean Martin jean.martin@yopmail.com SOUTH 16094 1021 3 12/22/2016 \$102,289.00 6 4 Diane Meyer diane.meyer@yopmail.com ATLANTIC 16101 1006 4 12/24/2016 \$197,881.50 7 Georges Dunel georges.dunel@yopmail.com WEST 16106 1005 7 12/31/2016 \$246,334.29	2	1 Bill Raley	bill.raley@yopmail.com	NORTH-WEST	16096	1003	1	12/23/2016	\$93,305.00		
4 3 Jean Martin jean.martin@yopmail.com SOUTH 16091 1021 3 12/21/2016 \$89,207.00 5 3 Jean Martin jean.martin@yopmail.com SOUTH 16094 1021 3 12/22/2016 \$102,289.00 6 4 Diane Meyer diane.meyer@yopmail.com ATLANTIC 16101 1006 4 12/24/2016 \$197,881.50 7 7 Georges Dunel georges.dunel@yopmail.com WEST 16106 1005 7 12/31/2016 \$246,334.29	3	2 Sandra Davis	sandra.davis@yopmail.com	NORTH-WEST	16105	1010	2	12/31/2016	\$230,218.15		
5 3 Jean Martin jean.martin@yopmail.com SOUTH 16094 1021 3 12/22/2016 \$102,289.00 6 4 Diane Meyer diane.meyer@yopmail.com ATLANTIC 16101 1006 4 12/24/2016 \$197,881.50 7 7 Georges Dunel georges.dunel@yopmail.com WEST 16106 1005 7 12/31/2016 \$246,334.29	- 4	3 Jean Martin	jean.martin@yopmail.com	SOUTH	16091	1021	3	12/21/2016	\$89,207.00		
6 4 Diane Meyer diane.meyer@yopmail.com ATLANTIC 16101 1006 4 12/24/2016 \$197,881.50 7 7 Georges Dunel georges.dunel@yopmail.com WEST 16106 1005 7 12/31/2016 \$246,334.29	5	3 Jean Martin	jean.martin@yopmail.com	SOUTH	16094	1021	3	12/22/2016	\$102,289.00		
7 Georges Dunel georges.dunel@yopmail.com WEST 16106 1005 7 12/31/2016 \$246,334.29	6	4 Diane Meyer	diane.meyer@yopmail.com	ATLANTIC	16101	1006	4	12/24/2016	\$197,881.50		
	7	7 Georges Dunel	georges.dunel@yopmail.com	WEST	16106	1005	7	12/31/2016	\$246,334.29		

But what is important is to know that only those records will be saved into the hidden and temporary tables corresponding to this SQL query. That means that if in the cache delay duration, here 5 minutes, the query is run again with the same criteria, the response is immediate because data are coming from temporary tables and not from the original real data source.

If now we modify these criteria by entering for example « Since 3 months » the result will be saved in a new temporary distinct table for this query with these criteria, during the cache duration.

If we use other criteria or parameters on this multi data source query, then theses filters will apply to the data from the temporary tables as far as the cache value is valid. For example we can add a parameter on the AREA field in the above query:

📑 Criteria					×
Columns salesman.AREA ✓	Operators = ~	Value Column Expression (Query		
Where sales.parameters.P_DATE = 'Last Mu and salesman.AREA = P_AREA	onth'			Not Combin Uncomb	ne
OK Cancel <u>N</u> ew	<u>R</u> emove	Parameters			

If we run, we get a question about the AREA:



Parameters	– 🗆 X
✓ Choose value for AREA	ATLANTIC CENTRAL NORTH-WEST Cancel
	WEST

And if we select any Area, the data are still coming from the temporary tables, only for that area, as far as the cache value is still valid for the DATE parameter:

╞╘╌═┝		AREA	Varchar				255	0	Salesman area		
sales	.paran	neters									
		<caching< th=""><th>> Varchar</th><th></th><th></th><th></th><th>32</th><th>0</th><th>Caching mode: ON c</th><th>or empty (default): use or update o</th><th>×</th></caching<>	> Varchar				32	0	Caching mode: ON c	or empty (default): use or update o	×
		<scope></scope>	Varchar				32	0	Caching scope: Emp	ty: by user or for all users (chose	ы
		<index></index>	Varchar				4000	0	Indexes: list of IX(Ind	dexName, <fields>) = Index or UK(</fields>	0
		P_DATE	Varchar				0	0			
🗆 sales											
		NO NO	Varchar				5	0	Invoice number		
		CUST	Smallint				2	0	Customer code		
╞╼═╾		SAL	Smallint				2	0	Salesman code		
		DATE	TimeStamp				8	3	Order Date		¥
┝┖╍ <u></u> ┣╸		TOTAL	Float				8	0	Amount		*
											Ŧ
<										>	
	SAL	SALNAME	MAIL	AREA	NO	CUST	sales.SAL	DAT	E TOTAL		
1	1	Bill Raley	bill.raley@yopmail.com	NORTH-WEST	16095	1003	1	12/22/20	16 \$82,450.00		
2	1	Bill Raley	bill.raley@yopmail.com	NORTH-WEST	16096	1003	1	12/23/20	16 \$93,305.00		
3	11	Joe Kramer	joe.kramer@yopmail.com	NORTH-WEST	16100	1010	11	12/24/20	16 \$186,527.50		
4	11	Joe Kramer	joe.kramer@yopmail.com	NORTH-WEST	16102	1010	11	12/26/20	16 \$200,114.50		
5	2	Sandra Davis	sandra.davis@yopmail.com	NORTH-WEST	16105	1010	2	12/31/20	16 \$230,218.15		

In summary, in the above query, we need to remember that:

The criteria defined on the sales.parameters branch level for the P_DATE parameter will modify the number of records saved in the temporary tables during all the valid cache period for this query.

The criteria defined on the AREA field level is used as a filter on the result extracted from these temporary tables.



2- Create and use synonyms with « Click and DECiDE Queries » data source:

- In this chapter, we will describe how to use this new « Click and DECiDE Queries » data source to create some "synonyms", from single or multi data sources queries.
- The synonyms are some « views » pointing to temporary tables set in cache, according several options such as cache, scope and indexes.
- The end-users will use a standard SQL Server data source to extract data from these synonyms.
- Note that in the synonyms, some of them can be created from an iSeries Data Source, other from an Excel
 or SQL Server data source etc., and that the end-user working with these synonyms can make queries mutli
 data source without knowing it.

Use :

1- Create a Click and DECiDE Queries data source:

In this step already describe in point 1, you just only need to activate the synonyms in the Click&DECiDE Queries data source configuration:

In the « Cache » Tab of this Click&DECiDE data source properties in Administration Manager, activate this
option:

BAI Builder Source Configuration [Query Builder]	— 🗆 X
General Cache	
 Enable Indexing Support Enable Synonyms Support 	A.

- Warning: you also need to check if the SQL Server data sources you will use on that Server where Click&DECiDE Enterprise Edition is installed have another option enabled to support the synonyms management (Local SQL Server and/or Secured SQL Server).
- Open the data source properties in Click&DECiDE Administration Manager :

Microsoft SQL Server Interface - S	Settings	×					
Server Name		OK Cancel					
Support for decimal and numeri							
Alphanumeric Data on Server is stored with the ANSI character set							
TCP/IP Connection							
SQL Azure							
Native Client Version							
Automatically selected (2008, 20	12, 2014, 2005) 🗸 🗸						
Synonyms Support	OFF V						
	Enabled Synonyms only						

Contact us: E-mail: <u>support@clickndecide.com</u> Tel: +33 (0)4 67 84 48 00 Headquarters: Click&DECiDE 130, rue du Baptistou - 34980 St Gély du Fesc, France. To contact your nearest Click&DECiDE partner, <u>click here</u>.



- « Enabled » option will allow the end-users to access the synonyms, but also access to tables and \checkmark views.
- ✓ The « Synonyms only » option will limit the visibility to the synonyms only. If no synonyms exist, Builder will display nothing when browsing the tables list.

Save this configuration and click the command File > Update Security.

Use from Click&DECiDE Builder

Query source containing or not containing parameters. \geq

- 1. Start Builder (or Web Builder from the Web Portal)
- 2. Open or create a project (for example the file My Queries Multi Sources.wfv).
- 3. If the "Query Builder" data source has been authorized for the User Group you belong to, you should see the list of the projects authorized for that User Group, or for these Groups if you belong to several User Groups.
- 4. Create a new query from this « Query Builder » data source: Select the required project to display the existing queries, as tables sources, for example the project « iSeries queries » containing iSeries queries (AS/400):

In this example we will use an iSeries query containing already 2 parameters:

A P DATE parameter on the DATE field.

A P AREA parameter on the AREA field.

Query Paramete	rs X
Parameters:	🗂 🗙 🗲 🐇 🗎 🛍 🛍 🔯
P_DATE	
P_AREA	
	Parameter Properties
Name	P_DATE
lype	Timestamp
Status	Enable
Update Method	Input List
Title	Choose value for DATE
Mandatory	No
Editable	No
Default Value	IGNORE
Value List	in 2 days Day after tomorrow;in 1 days Tomorrow;0 days ago Today;1 days ag
Selection	Single
Hide List's First Column	Yes
	OK Cancel

This query is saved under the name « iSeries Sales » in the iSeries Queries.wfv project file. Create this new query selecting that project file iSeries Queries.wfv and the query named « iSeries Sales »:

Click&DECIDE Business Application Intelligence | Promotions

My Queries Mutli Sources.wfv	📑 Query*					- 💌 ×
All Data Sources	Content of [Qu	ery Builder]:"iS	eries Que	ries"		
Excel32 Excel64 Excel64 Excel64 Excel64 Excel64 Excel Query Builder Excel queries Excel	Name iSeries Sales Julian and w query Test Julian a week istone	Des Sale reek nd Week	cription Table or	n AS/40	0	
	iSeries Sales				V	
	Name NO CUST SAL SALNAME AREA DATE TOTAL	Type Varchar SmallInt SmallInt Varchar Varchar TimeStamp Float	Length 5 2 2 15 10 26 31	Scale 0 0	Description Invoice Number Customer Code Salesman Code Salesman Name Sales Area Order Date Amount	

Go to the « Query » Tab and select the required columns:

🔟 My Queries	🔟 My Queries Mutli Sources.wfv* 📑 Query*									
🔍 🍳 👥 66	Name		Туре			^				
🗆 iSeries Sales.p	arameters									
	<caching></caching>	Varchar		32	0 Caching mode: ON or empty	у				
	<scope></scope>	Varchar		32	0 Caching scope: Empty: by a	u				
	<index></index>	Varchar		4000	0 Indexes: list of IX(IndexNan	n				
	<synonym></synonym>	Varchar		256	0 Synonym for the data table	::				
	P_DATE	Varchar		0	0					
	P_AREA	Varchar		0	0					
🗆 iSeries Sales										
V	NO	Varchar		5	0 Invoice Number					
V	CUST	SmallInt		2	0 Customer Code					
	SAL	SmallInt		2	0 Salesman Code					
V	SALNAME	Varchar		15	0 Salesman Name					
V	AREA	Varchar		10	0 Sales Area	4				
	DATE	TimeStamp		26	0 Order Date	*				
	TOTAL	Float		31	8 Amount	\mp				
<					>					
Invoice N	lumber Customer Code	e Salesman Name	Sales Area Order Date	Amount						

13



The options such as Cache, Scope and optional index have already been described in the first part of this manual. **Nevertheless, note that, when synonyms are used, the cache must be defined as « permanent » and the Scope as « Global »**. We can add (optional) an index on the DATE field and a unique index on the « NO » field (Invoice Numer):

IX	Index1.	""DATE"")	;	UKC (Unique	eIndex1.	. ""NO""));
T T 7 1	(TIIOCZT)		/	OTCO (onrya		, 10)	/ /

📑 Criteria —		\times				
Columns Operators Value Expression iseries Sales.parameters. = permanent Where Where						
Where iSeries Sales.parameters. <caching> = 'permanent' and iSeries Sales.parameters.<scope> = 'Global' and iSeries Sales.parameters.<index> = 'IX(Index1, ""DATE""); UKC(UniqueIndex1, ""NO"");'</index></scope></caching>						
OK Cancel <u>N</u> ew <u>R</u> emove <u>P</u> arameters						

<Synonyms>

The idea is to create a view pointing on a temporary result table that will be available for the end-users who will not be managing themselves these caches, scopes and indexes. The end-users will see and will use the synonyms as if it was some SQL Server data source tables.

We will thus define here a customized name for that synonym, that means a "view" name in SQL Server.

To do so, click the criteria square on the left of the « synonyms » parameter in the *iSeries Sales.parameters* and enter as name for example **DataSetReport.demo.Sales 2015 North-West**, or <**Database Name.Schema name.synomym name**>.



📑 Criteria			×
Columns Operators Value Expression iSeries Sales.parameters. <synonym> = > DataSetReport.demo.Sales 2015</synonym>	North	-West	
Where iSeries Sales.parameters. <caching> = 'permanent' and iSeries Sales.parameters.<scope> = 'Global' and iSeries Sales.parameters.<index> = 'IX(Index1, ""DATE""); UKC(UniqueIndex1, ""NO"");' and iSeries Sales.parameters.<synonym> = 'DataSetReport.demo.Sales 2015 North-West'</synonym></index></scope></caching>	^	Not Combin Uncombi	e
OK Cancel New Remove Parameters	Ŧ		

Validate with OK.

Also note that, in the *iSeries.Sales.parameters*, we can see that the parameters P_DATE and P_AREA from the query source are displayed. Of course if the query source contains no parameters, ignore this paragraph. If we add criteria on these parameters, it will limit the content of the temporary tables that will be updated when running this query, and according to the options given for the Cache, the Scope and the optional indexes, and depending of the criteria values.

To limit the records to the year 2015 for example and to the 'North-West' area, we can add the following criteria for the corresponding parameters:

- On the left side of P_DATE parameter enter as criteria 2015.
- On the left side of P_AREA parameter enter as criteria NORTH-WEST.

In fact, all criteria should appear as follow:

📑 Criteria –		×				
Columns Operators Value Expression iSeries Sales, parameters.P_AREA = NORTH-WEST						
Where iSeries Sales.parameters. <caching> = 'permanent' and iSeries Sales.parameters.<scope> = 'Global' and iSeries Sales.parameters.<index> = 'IX(Index1, ""DATE""); UKC(UniqueIndex1, ""NO"");' and iSeries Sales.parameters.<synonym> = 'DataSetReport.demo.Sales 2015 North-West' and iSeries Sales.parameters.P_DATE = '2015' and iSeries Sales.parameters.P_AREA = 'NORTH-WEST'</synonym></index></scope></caching>						
OK Cancel <u>N</u> ew <u>R</u> emove <u>P</u> arameters						

Contact us: E-mail: support@clickndecide.com Tel: +33 (0)4 67 84 48 00 Headquarters: Click&DECiDE 130, rue du Baptistou - 34980 St Gély du Fesc, France. To contact your nearest Click&DECiDE partner, click here.



My Queries Mutli Sources.wfy Query*										
9	. 11 66		Name		Туре 🔺 🔻					^
🗆 iSer	ies Sales.p	paramete	ers							
-=		<caching< th=""><th>g> '</th><th>Varchar</th><th></th><th></th><th>32</th><th>0</th><th>Cachi</th><th></th></caching<>	g> '	Varchar			32	0	Cachi	
=		<scope></scope>		Varchar			32	0	Cachi	1
		<index></index>	,	Varchar			4000	0	Index	
_=		<synony< th=""><th>/m></th><th>Varchar</th><th></th><th></th><th>256</th><th>0</th><th>Synoi</th><th></th></synony<>	/m>	Varchar			256	0	Synoi	
-=		P_DATE	,	Varchar			0	0		
		P_AREA	· · · ·	Varchar			0	0		
🗆 iSer	ies Sales									
		NO	,	Varchar			5	0	Invoic	
		CUST	:	Smallint			2	0	Custo	
		SAL	:	Smallint			2	0	Sales	
	- <i>- v</i>	SALNA	NE Y	Varchar			15	0	Sales	
	- <i>- v</i>	AREA	,	Varchar			10	0	Sales	4
		DATE		TimeStamp			26	0	Order	*
		TOTAL		Float			31	8	Amou	Ŧ
<									>	
	Invoice I	lumber	Customer Code	Salesman Name	Sales Area	Order Date	Amount			~
1	15001		1003	Bill Raley	NORTH-WEST	1/4/2015	\$94,029.60			
2	15012		1011	Robert Salta	NORTH-WEST	2/2/2015	\$107,565.00			
3	15016		1011	Robert Salta	NORTH-WEST	2/4/2015	\$122,289.50			
4	15019		1010	Joe Kramer	NORTH-WEST	3/6/2015	\$70,283.20			
5	15020		1010	Joe Kramer	NORTH-WEST	3/7/2015	\$80,889.60			
6	15021		1011	Robert Salta	NORTH-WEST	3/9/2015	\$44,940.00			
7	15032		1003	Bill Raley	NORTH-WEST	4/6/2015	\$79,845.00			
8	15033		1003	Bill Raley	NORTH-WEST	4/7/2015	\$90.069.50			

Validate with OK and run the query.

On the background, automatically a synonym named « Sales 2015 North-West » has been created in SQL Server and that synonym is pointing to a temporary table only containing the records for the year 2015 and for the North-West area.

How to make a query on that synonyms?

From Click&DECiDE Builder (or from Click&DECiDE Web Builder through the Web Portal), create a new query by selecting the « Local SQL Server » data source or « Secured SQL Server » data source if through the Web Portal, as in that case the data source must be defined as "Secured by Click and DECiDE".

Be sure that both data sources have the « Synonyms support » option enabled in Click&DECiDE Administration Manager.(See the paragraph « how to activate the synonyms).

For example, create a new query on the Secured SQL Server secured data source on the database and schema declared in the full synonym name: (DataSetReport.demo)



My Queries Mutli Sources.wfv*	📑 Query1*		▼ X			
All Data Sources	ontent of [Secured SQL Server]:DataSetReport.den					
BigQuery ConnectSQLServer Excel32 Excel64 ConnectSQLServer Excel64 ConnectSQLServer Connec	Name Category Customer Product Sales Sales 2015 North-West Sales_Details Salesman States	Description BASE TABLE BASE TABLE BASE TABLE BASE TABLE BASE TABLE BASE TABLE BASE TABLE	Tables Synonym Tables			

Among the tables list from this DataSetReport database we can see the new synonym that has been created when running the previous query. Select this synonym and select all fields then run the query: the data are corresponding to the year 2015 and the North-West area declared as criteria on the P_DATE and P_AREA parameters.

My Queries Mutli Sources.wfv* Query1*								▼ ×
9	. 11 66	Name	T	уре	• •			^
Sales 2015 North-West								
	🖌 NO	Va	rchar		:	5 O Ir	nvoice Number	
	— — ✓ CUST	Inte	eger		:	2 0 0	Customer Code	
	SAL	Inte	eger		:	2 0 5	Salesman Code	
	SALNAI	/E Va	rchar		1	5 0 5	Salesman Name	
	AREA	Va	rchar		10	0 0 5	Sales Area	~
	DATE	Tin	neStamp		1	8 30	Order Date	±
	TOTAL	Flo	at			B 0 A	Amount	Ŧ
	Invoice Number	Customer Code	Salesman Code	Salesman Name	Sales Area	Order Date	Amount	^
1	15001	1003	1	Bill Raley	NORTH-WEST	1/4/2015	\$94,029.60	
2	15012	1011	8	Robert Salta	NORTH-WEST	2/2/2015	\$107,565.00	
3	15016	1011	8	Robert Salta	NORTH-WEST	2/4/2015	\$122,289.50	
4	15019	1010	11	Joe Kramer	NORTH-WEST	3/6/2015	\$70,283.20	
5	15020	1010	11	Joe Kramer	NORTH-WEST	3/7/2015	\$80,889.60	
6	15021	1011	8	Robert Salta	NORTH-WEST	3/9/2015	\$44,940.00	

On that query result, you can now apply other criteria or parameters as in a standard Builder query.

Note that, to simplify, this manual has been done on a single data source query to explain the synonyms feature, but it is of course possible to do the same using a multiple data source query.

If for example, you have created several synonyms on various data sources, it is thus possible for the end-users to create a new standard SQL Server query by joining several synonyms together.

Example: the synonym created in this manual is named « Sales 2015 North-West » and is corresponding to data coming from an iSeries.



We can imagine another synonym named « North-West Salesmen » corresponding to data managed in another data source Oracle, then a query joining these 2 synonyms could be similar to a multiple data source query but without the need to manage technical options such as caches, scopes or indexes reserved to advanced users.

Conclusion:

The Click and DECIDE Queries data source is thus mainly reserved to Administrators, or advanced users, while the Synonyms access is mainly reserved to end-users through a data warehouse.

***** End of documentation *****