

The next generation CONTROL®

version 10.5

Flex Views

Flex Views

Table of Contents

Why Flex views?.....	6
What can you do with flex views?.....	6
A standard variance report	6
A customized budget template	7
A mashup.....	8
Flex tables.....	9
The best of both worlds.....	10
How does it work?	11
Flexing	12
Using the Flex Pane.....	15
The structure of the Flex Pane.....	16
Functions	18
Flex function arguments	19
Reference type	20
Source type	23
Dealing with ranges	24
Authoring using the worksheet.....	27
Partial formulas	28
Errors.....	29
Creating a new flex view.....	31
Undoing flex formula changes.....	34
Using model view specifiers.....	34
Customizing the member lists.....	37
Flex Tables.....	37
What is a Flex Table?.....	38
Creating a Flex Table.....	38
Flex table examples.....	40



Flex Views

Computational view	40
Source data view	41
Flex table properties.....	42
Limitations and design considerations.....	47
Source data and object data in flex views	47
What is special about source and object data?	47
Row Numbers	48
Key pseudo-dimensions.....	48
Object Views.....	49
Updating considerations	50
Customizing your flex view	51
Adding Commands.....	51
Command Type	52
Standard Commands.....	52
Options	53
List Selection.....	54
List selection flex function	54
List selection features.....	56
Dynamically formatting flex views	60
Exceptions	60
Other dynamic formatting options.....	62
Numeric Formats	63
Options for flex views	65
Carve optimization.....	65
Synchronization.....	66
Protection state.....	67
Offline usage.....	68
Opening behavior.....	68



Flex Views

Keywords in filter expressions	70
Flex view performance	71
CONTROL® reference functions (CRFs)	71
New features and functionality	71
Converting CRF sheets to flex views.....	73
Converting individual CRFs to flex formulas	73
CRFs vs flex formulas.....	73
Keywords.....	74
KCI_NoRefreshPasteSize	74
KCI_FlexSizeWarning	75
KCI_FlexButtonWidth.....	75
Cut, Copy, and Paste.....	76
Best Practices	77
Paste Flex Special.....	77
Special cases.....	78
Conclusion	80
Appendix A.....	80
Flex Functions.....	80
ASQuery	80
BranchData.....	82
Command.....	84
Instructor led exercise	86
Comment.....	87
Data	87
DataReadOnly.....	88
HeaderData.....	89
Hyperlink.....	90
Keyword	90
List	91



Flex Views

ListSelection.....	94
Instructor led exercise	97
MemberData	98
MetaData	99
NestedBranchData	100
ViewData.....	101
SQLQuery.....	101



Flex Views

Why Flex views?

With decades of usage by thousands of users in a broad spectrum of industries, it is fascinating to observe the creative ways in which CONTROL® data is used for business models, analyses, input templates, reports, and presentations.

Flex views are the latest advance in empowering anyone who knows Excel to access and manipulate CONTROL® data for any purpose they can imagine – with the same security, data integrity, and scope of applications that are the hallmark of CONTROL®.

In previous releases, we supplemented the capabilities of views with the addition of *scratchpads* and CONTROL® Reference Functions (CRF). While these additions have been used successfully by many customers, they have functional and scalability limitations that restrict their use. They also require a deeper knowledge of CONTROL® than a typical Excel user would have.

Flex views are designed to overcome these limitations and to feel natural to existing Excel users. Our goal is to marry the richness and power of the CONTROL® architecture to the familiarity, flexibility, and simplicity of Excel. This is a crucial element in the value equation that CONTROL® brings to our customers.

What can you do with flex views?

A standard variance report

Flex views allow you to rearrange data from a CONTROL® view any way you want on the worksheet. For example, you can turn this view with variance information

	Actual Mar-2018	Budget Mar-2018	-CY Actual vs CY Budget Var	-CY Actual vs CY Budget %	Actual YTD-Mar-2018	Budget YTD-Mar-2018	-CY Actual vs CY Budget YTD Var	-CY Actual vs CY Budget YTD %	Budget YR-2018
11 FS - Account									
12 Full Time Equivalent	71	30	(41)	-138%	238	50	(188)	-375%	500
13 +Total Employee Compensation	572	227	(344)	-151%	1,681	665	(1,016)	-153%	2,437
14 +Travel & Entertainment	434	134	(300)	-223%	1,120	441	(679)	-154%	1,529
15 +Supplies & Services	793	296	(497)	-168%	2,223	979	(1,244)	-127%	3,788
16 +Communications & IT	133	1	(132)	-16750%	505	157	(348)	-222%	903
17 +Freight & Shipping	299	193	(107)	-55%	1,093	491	(602)	-123%	1,922
18 +Occupancy	111	17	(93)	-539%	252	158	(94)	-59%	638
19 +Maintenance	290	125	(165)	-133%	815	437	(378)	-86%	1,345
20 +Insurance	252	176	(77)	-44%	503	288	(215)	-74%	1,158
21 +Utilities	149	81	(68)	-85%	395	249	(146)	-58%	1,248
22 +Depreciation & Amortization	198	154	(43)	-28%	922	602	(320)	-53%	1,988
23 +Other Expenses	388	172	(216)	-125%	947	380	(566)	-149%	1,482
24 -Total Overhead Expenses	3,618	1,576	(2,043)	-130%	10,455	4,848	(5,606)	-116%	18,439

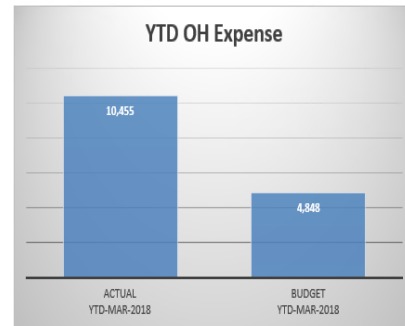
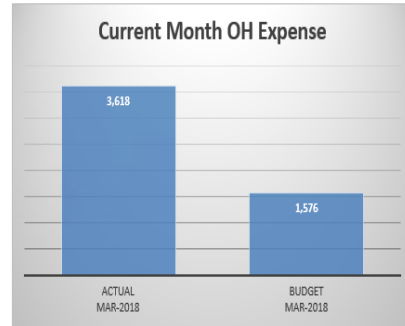
into a polished variance report just using simple Excel copy/paste and charting functions:



Flex Views

Overhead Variance Summary					
Total Company					
	Actual Mar-2018	Budget Mar-2018	CY Actual vs CY Budget Var	CY Actual vs CY Budget %	% Total
Full Time Equivalent	71	30	(41)	-138%	
Total Employee Compensation	572	227	(344)	-151%	16%
Travel & Entertainment	434	134	(300)	-223%	12%
Supplies & Services	793	296	(497)	-168%	22%
Communications & IT	133	1	(132)	-16750%	4%
Freight & Shipping	299	193	(107)	-55%	8%
Occupancy	111	17	(93)	-539%	3%
Maintenance	290	125	(165)	-133%	8%
Insurance	252	176	(77)	-44%	7%
Utilities	149	81	(68)	-85%	4%
Depreciation & Amortization	198	154	(43)	-28%	5%
Other Expenses	388	172	(216)	-125%	11%
Current Month OH Expense	3,618	1,576	(2,043)	-130%	100%

	Actual YTD-Mar-2018	Budget YTD-Mar-2018	CY Actual vs CY Budget YTD Var	CY Actual vs CY Budget YTD %	% Total
Full Time Equivalent	238	50	(188)	-375%	
Total Employee Compensation	1,681	665	(1,016)	-153%	16%
Travel & Entertainment	1,120	441	(679)	-154%	11%
Supplies & Services	2,223	979	(1,244)	-127%	21%
Communications & IT	505	157	(348)	-222%	5%
Freight & Shipping	1,093	491	(602)	-123%	10%
Occupancy	252	158	(94)	-59%	2%
Maintenance	815	437	(378)	-86%	8%
Insurance	503	288	(215)	-74%	5%
Utilities	395	249	(146)	-58%	4%
Depreciation & Amortization	922	602	(320)	-53%	9%
Other Expenses	947	380	(566)	-149%	9%
YTD OH Expense	10,455	4,848	(5,606)	-116%	100%



A customized budget template

With a flex view, you can insert rows and columns, add Excel formulas, and present data from other pages of the view. Perhaps you want to embed Excel formulas to calculate variable expenses based on total YTD company revenue. As you adjust your values, you can see the impact on the total company expense and see how your ratios compare to the total company.



Flex Views

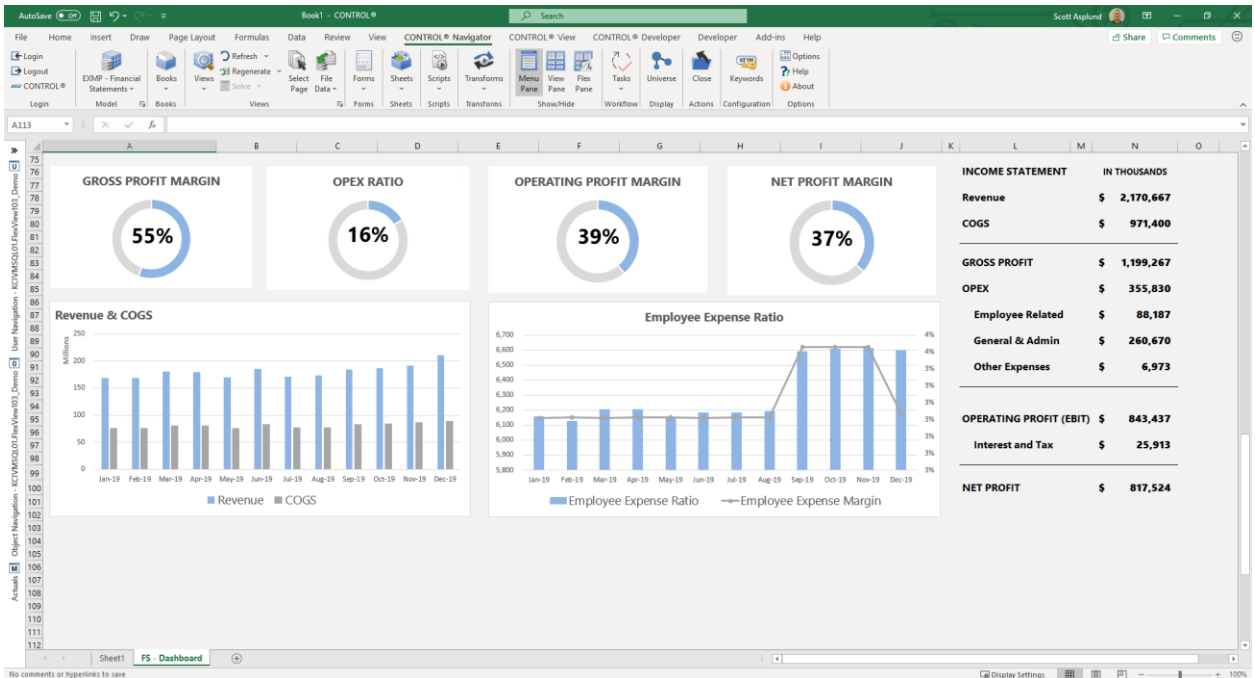
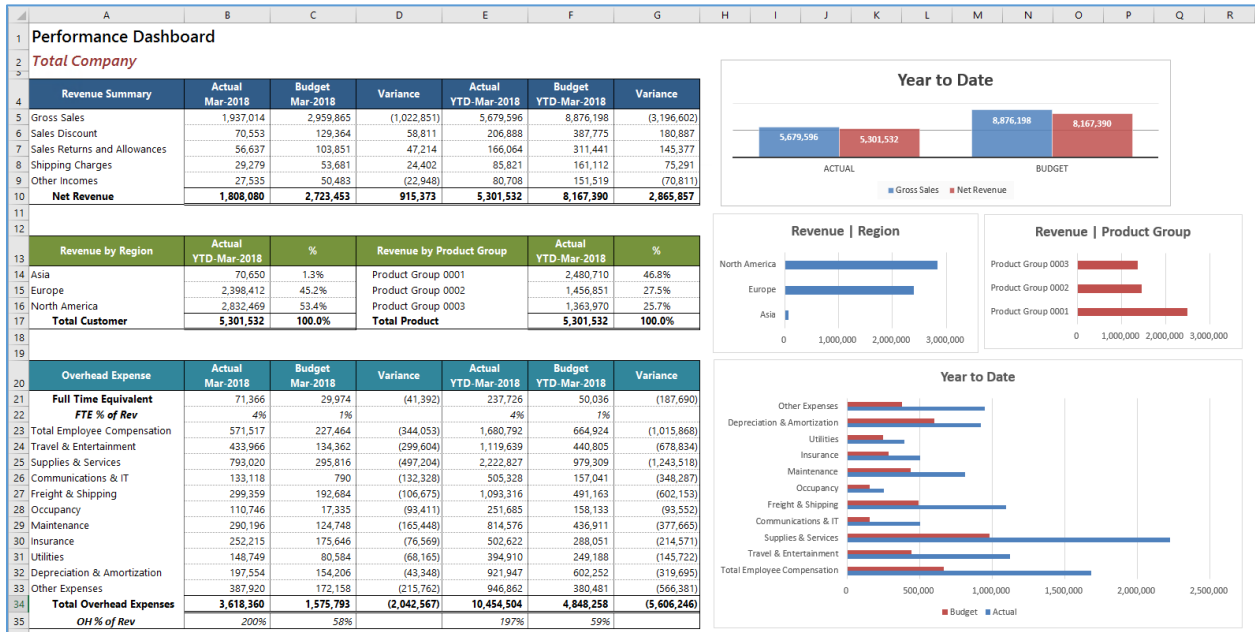
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S				
1	Input NY Budget as % Total Company of Revenue																						
2	Total Company																						
3																							
4		PY Act	CY Fcst	NY Bud																			
5	Revenue	80,000,000	82,000,000	88,000,000																			
6	Total OH Expense	34,964,785	32,584,785	28,965,741																			
7	% Revenue	44%	40%	33%																			
8																							
9		CY Fcst			NY Budget																		
10	Department 101	Year	% Revenue	% Expense	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Year	% Revenue	% Expense				
11	Total Employee Compensation	2,319,026	3%	13%	231,126	267,395	363,186	111,972	355,389	147,517	196,121	256,384	321,851	187,350	179,573	319,637	2,937,500	3%	14%				
12	Travel & Entertainment	1,847,836	2%	10%	50,347	85,595	169,372	131,674	263,104	153,983	164,956	49,241	179,901	149,622	67,935	126,412	1,592,143	2%	8%				
13	Supplies & Services	4,218,416	5%	23%	543,518	235,181	386,607	395,633	433,500	331,326	249,370	538,574	485,430	404,701	362,412	367,658	4,733,910	5%	23%				
14	Communications & IT	1,033,921	1%	6%	65,603	132,084	165,060	41,105	51,767	142,496	125,481	(1,420)	19,400	95,967	146,163	185,069	1,168,776	1%	6%				
15	Freight & Shipping	1,422,990	2%	8%	120,155	139,010	188,809	58,210	184,755	76,689	101,957	133,286	167,320	97,397	93,354	166,169	1,527,111	2%	7%				
16	Occupancy	536,953	1%	3%	54,251	(10,963)	41,380	95,741	(5,500)	46,460	94,036	74,871	11,940	27,710	21,498	110,940	562,365	1%	3%				
17	Maintenance	1,468,420	2%	8%	151,877	117,045	200,825	124,644	257,096	197,578	111,840	165,198	173,560	94,570	30,650	131,759	1,756,642	2%	8%				
18	Insurance	994,222	1%	6%	150,366	9,487	203,213	18,416	195,224	(5,486)	179,662	22,921	108,561	56,680	88,992	98,102	1,126,139	1%	5%				
19	Utilities	1,178,758	1%	7%	191,046	115,990	152,731	88,775	154,183	66,979	82,343	90,218	43,586	126,854	76,301	34,691	1,223,699	1%	6%				
20	Depreciation & Amortization	1,510,132	2%	8%	143,634	137,594	200,716	168,022	149,270	167,270	221,925	132,644	111,920	87,195	199,760	139,114	1,859,064	2%	9%				
21	Other Expenses	1,456,738	2%	8%	187,899	146,973	127,557	223,711	255,629	126,888	217,076	153,973	237,780	209,961	254,815	143,574	2,285,836	3%	11%				
22	Total OH Expense	17,987,412	22%	100%	1,889,824	1,375,391	2,199,458	1,457,904	2,284,417	1,451,700	1,744,767	1,615,889	1,861,248	1,538,008	1,521,453	1,823,125	20,773,183	24%	100%				
23	% Revenue	22%			2%	2%	2%	2%	3%	2%	2%	2%	2%	2%	2%	2%	24%						
24																							

A mashup

Have you ever wanted to analyze and compare data from different *models* or *datasources* in a single worksheet? For example, see key performance revenue information along with expense data. With flex views, you can add data from as many CONTROL® *models* or *datasources* as you like, and even page through their common *dimensions*:



Flex Views



Flex tables

You can now combine transactional source data that drives your budget or analysis on a single worksheet. In the example below, you can see the Actual detail, which you can use to enter transactional Budget detail that ultimately feeds into the Budget summary. As you change the



Flex Views

Department and Account selections, the detail tables will expand and compress based on the number of records for each selection.

Budget - Summary													
	Jan-20	Feb-20	Mar-20	Apr-20	May-20	Jun-20	Jul-20	Aug-20	Sep-20	Oct-20	Nov-20	Dec-20	Year 2020
6916 New Depreciation Expense	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	180,000.00

Budget - Detail															
Dept ID	Acct ID	Description	Jan-20	Feb-20	Mar-20	Apr-20	May-20	Jun-20	Jul-20	Aug-20	Sep-20	Oct-20	Nov-20	Dec-20	Year 2020
110110	6916	Executive Desk	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	180,000.00
110110	6916	Hon File Cabinet	-	-	-	-	-	-	-	-	-	-	-	-	-
110110	6916	Bookshelf	-	-	-	-	-	-	-	-	-	-	-	-	-
110110	6916	Thinkpad S20 Edge	-	-	-	-	-	-	-	-	-	-	-	-	-
110110	6916	Ergonomic Chair	-	-	-	-	-	-	-	-	-	-	-	-	-
110110	6916	Dell Inspiron 17 Series	-	-	-	-	-	-	-	-	-	-	-	-	-
Total			15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	180,000.00

Actual - Detail															
Dept ID	Acct ID	Description	Jan-19	Feb-19	Mar-19	Apr-19	May-19	Jun-19	Jul-19	Aug-19	Sep-19	Oct-19	Nov-19	Dec-19	Year 2019
110110	6915	Executive Desk	18,981.60	19,076.98	19,172.85	19,269.19	19,366.02	19,501.71	19,561.15	19,659.44	17,702.35	17,985.98	18,599.62	18,920.73	227,797.62
110110	6915	Hon File Cabinet	19,955.01	20,055.29	20,156.07	20,257.36	20,359.15	20,501.80	20,564.28	20,667.62	18,610.17	18,908.33	19,553.44	19,891.02	239,479.55
110110	6915	Bookshelf	968.17	973.04	977.93	982.84	987.78	994.70	997.73	1,002.75	1,052.43	1,071.77	1,116.37	1,138.37	12,263.88
110110	6915	Thinkpad S20 Edge	10,649.90	10,703.42	10,757.20	10,811.26	10,865.59	10,941.72	10,975.06	11,030.22	11,578.70	11,789.49	12,200.06	12,522.02	134,902.64
110110	6915	Ergonomic Chair	5,082.91	5,108.45	5,134.12	5,159.92	5,185.85	5,222.18	5,238.10	5,264.42	5,525.24	5,626.80	5,860.94	5,976.42	64,385.33
110110	6915	Dell Inspiron 17 Series	16,791.41	16,875.79	16,960.60	17,045.82	17,131.48	17,251.52	17,304.09	17,391.05	15,659.77	15,910.67	16,453.51	16,737.57	201,513.28
Total			72,429.01	72,792.97	73,158.77	73,526.40	73,895.88	74,413.63	74,640.41	75,015.49	70,126.67	71,293.04	73,863.93	75,186.12	880,342.31

The best of both worlds

While flex views offer all the flexibility of Excel worksheet design, they retain these very powerful CONTROL® view capabilities:

- Cell comments and Hyperlinks
- Dimension filtering
- Page selection
- Expand to Multi-Page
- Refresh and Regenerate
- Save view and Save view as
- For updateable views: Solve, Lock for solve, and File data and comments
- Drill to data (in the underlying view), subaccount, scratchpad, source, comments, and hyperlinks
- Scaling of data
- Currency, translation map, and currency code
- Exception highlighting
- Solve

Flex views do not support navigation operations that manipulate the content such as compress, expand, and rotation, because the designer determines their format, which is similar to an Excel worksheet. Some of these shortcomings can be addressed using standard Excel techniques (e.g., grouping and outlining, hiding of rows, etc.)



Flex Views

With CONTROL® 10.5, you can use Page Relative references and flex tables to take advantage of navigational customizations such as suppress zero rows and columns, group and outline, and sort within *views*.

How does it work?

The idea that powers flex *views* is very simple. Each cell of the worksheet containing data is assigned a special string, called a flex formula. The flex formula includes a flex function that retrieves either data or meta-data into the cell. Like Excel functions, the flex function includes arguments that identify a *model*, *view*, dimensional member(s), etc. in order to connect to the application.

The flex formula does **not** replace the Excel formula or the value in the cell. Both a flex formula and an Excel formula can co-exist in the same data cell. This is a critical difference to a CRF, or other software products that interface using Excel's user defined functions.

For example, in an updatable *view*, you can define the data cells for travel expense with a flex formula that references the travel account in CONTROL®, and in the same cells you can build an Excel formula that calculates travel expense based on other cells in the *view*, or from a different Excel worksheet. The Excel formula calculates the value, and the flex formula defines where the resulting value is in the application.

In addition, when you copy, cut, and paste cells, insert or delete rows and columns, the flex formulas are naturally reorganized without you having to do any extra work!

You can:

- Add Excel formulas to any part of the worksheet – even where there is CONTROL® content
- Add non- CONTROL® data
- Link to other Excel data
- Freely format the worksheet
- Use Excel grouping and outlining
- Add buttons, validations, charts, etc.
- Add CONTROL® data from the same *view* or any other *view*
- Save the finished product (its still a *view*), or save as a different *view*
- Save the workbook for offline use

A flex *view* works pretty much like any Excel worksheet.

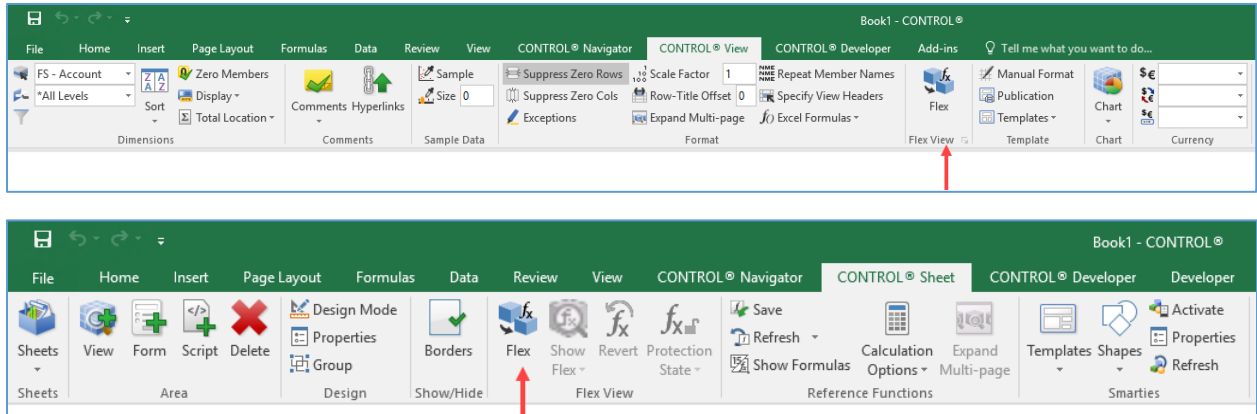


Flex Views

Flexing

You can turn any *view*, *scratchpad*, or CRF workbook *sheet* into a flex view with the click of a single button, here's how to do it:

1. Open a *view*, *scratchpad* or CRF workbook *sheet*.
2. On the CONTROL® View tab (CONTROL® Sheet tab for CRF workbook *sheets*), in the **Flex View** group, click **Flex**.



The flex process will remove the *view's* dimension-branch buttons and install the appropriate flex formulas in the cells where there is CONTROL® data. You are now ready to customize your flex *view*:

Flex Views

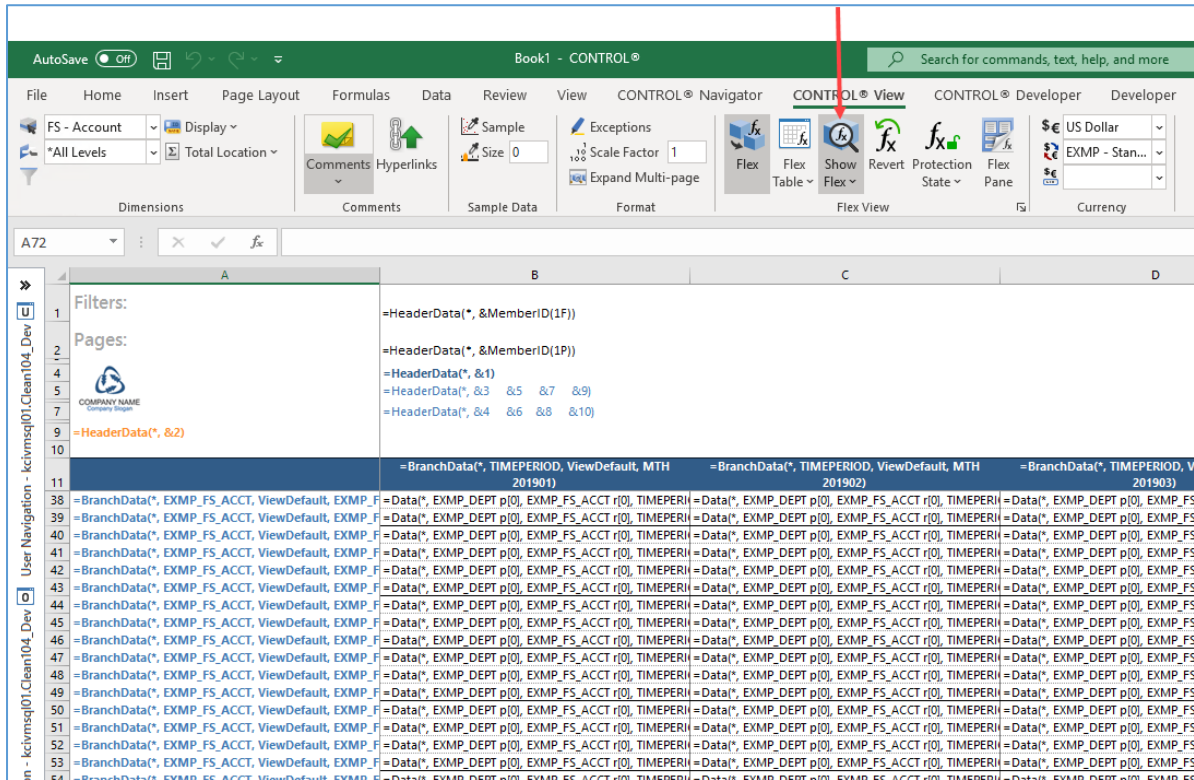
	Jan-19	Feb-19	Mar-19	Apr-19	May-19	Jun-19	Jul-19	Aug-19	Sep
38 Rent or Lease Expense	2,492,191	2,502,680	2,548,702	2,554,738	2,536,741	2,596,146	2,561,100	2,576,081	2
39 Occupancy	2,492,191	2,502,680	2,548,702	2,554,738	2,536,741	2,596,146	2,561,100	2,576,081	2
40 Maintenance Expense	713,072	716,210	727,129	729,301	726,217	740,417	733,271	737,441	
41 Repair and Maintenance Expense, Office	675,733	678,610	690,538	692,283	687,910	703,331	694,535	698,544	
42 Repair and Maintenance Expense, Vehicle	882,164	885,511	907,809	908,757	896,869	925,369	905,266	910,898	
43 Maintenance	2,270,969	2,280,331	2,325,476	2,330,341	2,310,996	2,369,117	2,333,072	2,346,883	2
44 Insurance Expense, Product Liability	986,759	989,994	1,023,308	1,022,712	1,001,725	1,044,017	1,010,801	1,017,560	1
45 Insurance Expense, Vehicle	839,449	842,366	867,987	868,018	852,662	885,258	860,486	866,091	
46 Insurance	1,826,208	1,832,360	1,891,294	1,890,730	1,854,387	1,929,275	1,871,287	1,883,651	1
47 Electric Expense	803,599	806,951	822,284	824,132	817,876	837,644	825,713	830,551	
48 Gas Expense	935,810	939,298	963,980	964,782	951,228	982,739	960,096	966,114	1
49 Utilities	1,739,409	1,746,249	1,786,264	1,788,915	1,769,104	1,820,383	1,785,809	1,796,665	1
50 Amortization Expense	1,612,343	1,615,901	1,698,705	1,692,119	1,631,769	1,736,162	1,645,532	1,658,084	1
51 Depreciation Expense	1,368,657	1,372,252	1,433,095	1,429,373	1,386,821	1,463,686	1,398,860	1,408,985	1
52 New Depreciation Expense	991,507	994,886	1,026,249	1,026,068	1,006,919	1,046,789	1,016,118	1,022,779	1
53 Depreciation & Amortization	3,972,507	3,983,038	4,158,048	4,147,560	4,025,509	4,246,637	4,060,510	4,089,848	4
54 Advertising Expense	214,171	215,247	216,329	217,416	218,508	220,039	220,710	221,838	
55 Dues and Subscriptions Expense	193,340	194,311	195,287	196,269	197,255	198,637	199,243	200,282	
56 Allocations	151,694	152,457	153,223	153,993	154,766	155,851	156,326	157,136	
57 Other Expenses	559,205	562,015	564,839	567,677	570,530	574,527	576,278	579,255	
58 Total Overhead Expenses	26,307,398	26,457,317	26,924,557	26,846,742	27,908,955	27,401,671	26,771,705	26,798,092	28

You can use your Excel skills to insert rows, columns, copy/paste, insert formulas, apply formatting, etc.

To see and edit the flex formulas, click **Show Flex** on the ribbon:



Flex Views



Show Flex is similar to Excel’s **Show Formulas** command. It displays the flex formula in each cell instead of the resulting value. A flex formula is a text string that looks like an Excel formula, with function names like **HeaderData**, **BranchData**, **Data**, etc. The arguments to these functions specify the details of where the data comes from or where to save the data in the CONTROL® application.

You can display flex formulas in different formats. Most of the time you will use the default **Understandable** format. Here is a list of the available formats that you can choose:

Show Flex	Used to
Understandable	Show the flex function and the Reference Type of each argument in a format that is most understandable to people.
Absolute	Show the flex function, with each argument converted to the source element. This format is extremely helpful, so you can clearly see the definition of the source data.



Flex Views

Persistent	Show the flex function as its CRF equivalent. You can typically ignore this format.
-------------------	---

Compact	Debug complex cases and is a technical format used by software developers. You will never use this format.
----------------	--

When you are satisfied with your new flex *view*, you can choose one of three save options.

- **Save** – Save the flex *view* only, and not the data.
- **Save and File** – Save the flex *view* and any unsaved data.
- **Save As** – Save the flex *view* and any unsaved data with a new name.

Note: These save options are slightly different from a standard *view*. Standard *views* only save the *view* definition, and not data.

To turn your flex *view* back to a standard *view*, click **Flex** and confirm that you want to discard any changes.

Using the Flex Pane

The Flex Pane helps to create and manage flex formulas in your flex *view*. To display the Flex Pane, go to the **CONTROL® Navigator** tab and click **Flex Pane**.



Flex Views

The screenshot shows the Microsoft Excel interface with the Flex Pane open on the right. The Flex Pane displays the content of the active cell (A12) from the worksheet. The worksheet shows a table with columns for months (Jan-19 to Aug-19) and rows for various expense categories. The Flex Pane shows the formula for the active cell: =BranchData!EXP_FS_ACCT, ViewDefault, EXP_FS_SUMMARY 100_FTE, [1].

	Jan-19	Feb-19	Mar-19	Apr-19	May-19	Jun-19	Jul-19	Aug-19
Full Time Equivalent	905	911	963	956	915	990	916	923
Total Employee Compensation	4,166,194	4,239,003	4,034,890	3,955,488	5,418,899	4,058,136	4,076,258	3,927,785
Travel & Entertainment	1,178,634	1,184,465	1,190,326	1,196,215	1,202,085	1,209,630	1,213,063	1,219,890
Supplies & Services	5,325,409	5,340,036	5,566,295	5,553,888	5,397,942	5,683,988	5,445,178	5,511,149
Communications & IT	1,347,770	1,351,668	1,405,704	1,403,202	1,366,699	1,435,078	1,378,775	1,388,430
Freight & Shipping	1,428,902	1,435,472	1,452,719	1,457,987	1,456,063	1,478,755	1,470,374	1,478,455
Occupancy	2,492,191	2,502,680	2,548,702	2,554,738	2,536,741	2,596,146	2,561,100	2,576,081
Maintenance	2,270,969	2,280,331	2,325,476	2,330,341	2,310,996	2,369,117	2,333,072	2,346,883
Insurance	1,826,208	1,832,360	1,891,294	1,890,730	1,854,387	1,929,275	1,871,287	1,883,651
Utilities	1,739,409	1,746,249	1,786,264	1,788,915	1,769,104	1,820,383	1,785,809	1,796,665
Depreciation & Amortization	3,972,507	3,983,038	4,158,048	4,147,560	4,025,509	4,246,637	4,060,510	4,089,848
Other Expenses	559,205	562,019	564,839	567,677	570,530	574,527	576,278	579,255
Total Overhead Expenses	26,307,398	26,457,317	26,924,557	26,846,742	27,908,955	27,401,671	26,771,705	26,798,092

The Flex Pane will open on the right side of the Excel window and the pane will display the content of the active cell on the worksheet.

The structure of the Flex Pane

The Flex Pane is similar to Excel's **Insert Function** dialog box:



Flex Views

Flex Function

View - SCA - OH Exp

Address of the active cell: **B12**

Flex formula: `Data(SCENARIO p[0], EXMP_DEPT p[0], EXMP_FS_ACCT r[0], TIMEPERIOD c[0])`

View Specifier	
SCENARIO	SCENARIO p[0]
EXMP_DEPT	EXMP_DEPT p[0]
EXMP_FS_ACCT	EXMP_FS_ACCT r[0]
TIMEPERIOD	TIMEPERIOD c[0]


Reference type: Absolute

Source type: In Use

Name	Argument Value
SCA - OH Exp	*

Buttons: Edit value, Resolve, Apply to cell B12, Is Flex, Is CRF, Show Flex, Update

Immediate update:

- The top of the pane shows the address of the active cell and the flex formula box with the complete formula as a text string. You can edit the flex formula directly in the box, or click  to display a list of available flex functions.
- The table immediately below the formula box has one row for each argument of the flex function. The first column contains the argument name and the second its current value.
- The lower portion of the pane contains a list of available options for the selected argument. Single click an option to update the argument value in the table, or double click to update the argument value in both the table and the formula box.
- The **Apply to cell/range** button at the bottom inserts the updated flex formula definition into the selected range of cells on the worksheet, and if immediate update is checked, refreshes the values in the worksheet.

Flex Views

Note: The selected range of cells on the worksheet may be broader than the active cell defined at the top of the Flex Pane.

Functions

Any data or meta-data you have access to in CONTROL® is available in flex *views*, via these functions:

View Specific Functions	Used to access
BranchData	Member IDs and Names from the various dimension-branches
Data	Numeric or textual data in the <i>view</i> that you can update if the <i>view</i> is updateable and you have write access to the data
DataReadOnly	Numeric or textual data in the <i>view</i> , but the data is not updateable
HeaderData	Contents of the <i>view's</i> header, with any <i>keywords</i> or header strings resolved in the <i>view</i> context
Comment	Comments from the <i>view's model</i> that you can update
Hyperlink	Hyperlinks from the <i>view's model</i> that you can update
NestedBranchData	Member IDs and names from a grouping of the <i>view's</i> dimension-branches, optionally subject to suppression of empty members
Non-View Specific Functions	Used to retrieve
MemberData	Meta-data (IDs, Names, reporting relationships) from a <i>level</i> or <i>hierarchy</i>
Command	Places a hyperlink or command button in the cell that runs a standard command, <i>script</i> , <i>mapping</i> , or <i>transform</i> , or opens another <i>view</i> , <i>form</i> , <i>sheet</i> or <i>book</i> .



Flex Views

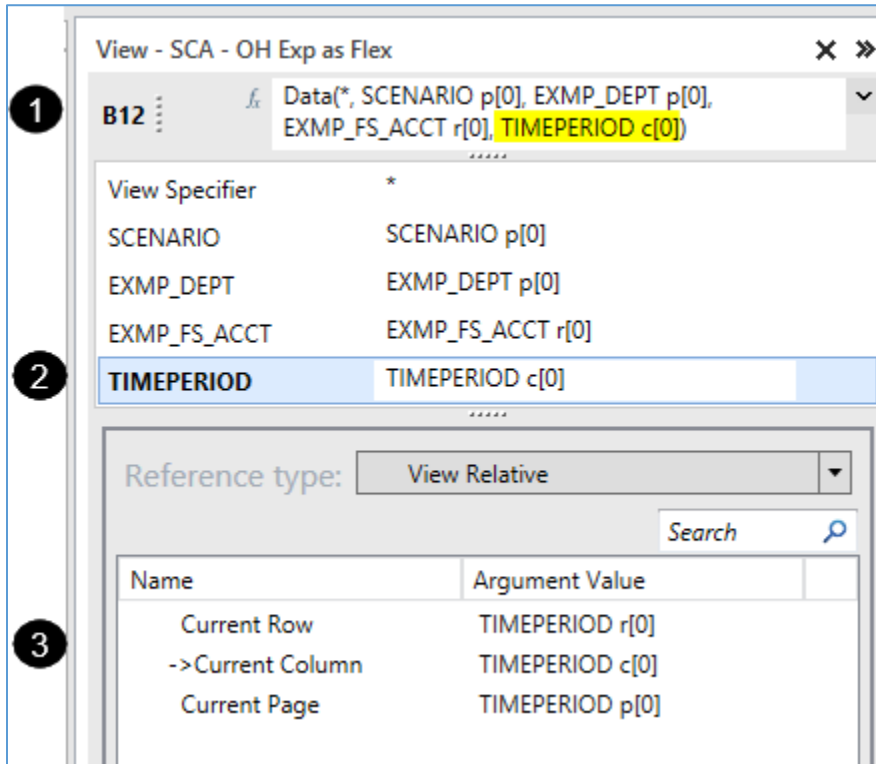
List	Meta-data from a <i>level</i> , <i>hierarchy</i> , custom <i>dimension</i> , <i>view</i> dimension-branch, or selection of objects in a convenient format for Excel cell validations
ListSelection	Places a hyperlink in a cell which launches a dialog to select an item from a <i>level</i> , <i>hierarchy</i> , custom <i>dimension</i> , <i>view</i> dimension-branch, list of objects, or unique values of a column in a data source
MetaData	Properties of objects
Keyword	Resolved value of a <i>keyword</i> for the current user
Transient Functions	Used to
ViewData	Assign a range of cells with the flex formulas of the current page of a <i>view</i> , translated to the appropriate function
External Data	Used to
ASQuery	Query data from a Microsoft Analysis Services model, the central repository for business intelligence and analytic applications
SQLQuery	Query data from relational tables and <i>views</i> , which are commonly used in operational systems, like ERP and CRM.

Flex function arguments

Each flex function has one or more arguments that specify which data or meta-data element to retrieve from the database to populate the active worksheet cell. Here is an example of the arguments used for a **Data** function:



Flex Views



You have three options for updating the value of an argument:

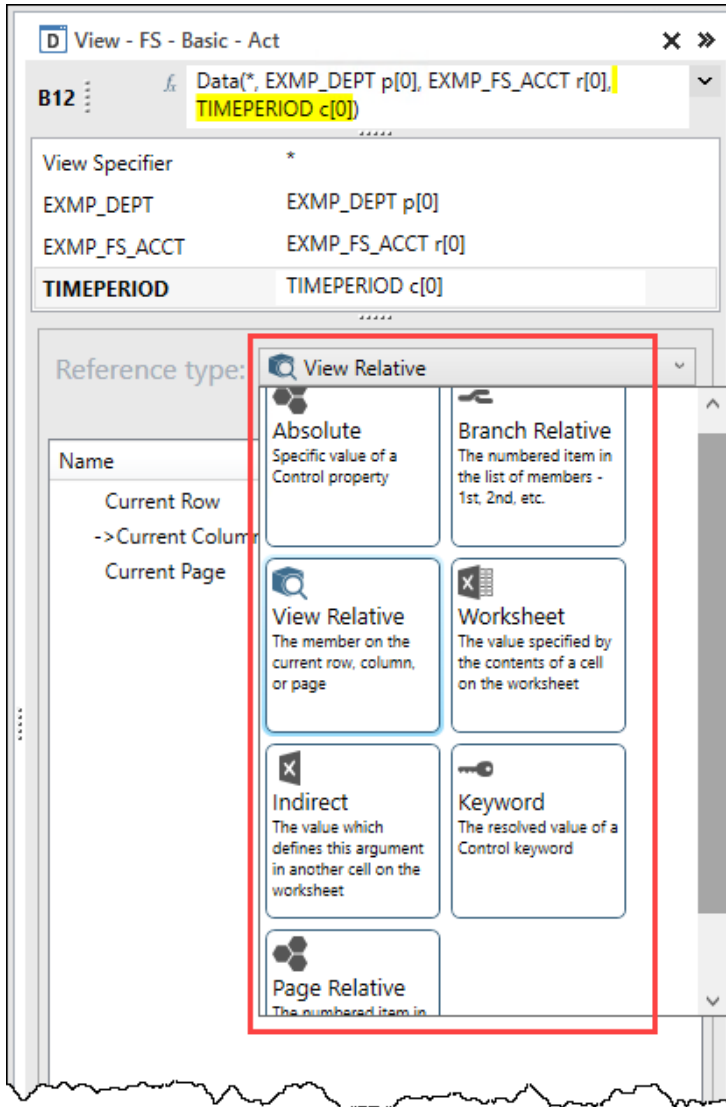
- 1) In the formula box, you can manually update it (the selected argument is highlighted)
- 2) In the table, you can type in the column to the right of the argument name (autocomplete is available)
- 3) In the section below the table, you can select an option from the list or tree. This is the easiest way to see the allowed values.

Reference type

Excel uses absolute and relative cell references as arguments in a formula. Flex views support multiple reference types because with CONTROL® there is multi-dimensional data that is not visible in the worksheet.

Flex formulas support seven different reference types, although some arguments only support a subset of the reference types.

Flex Views



Here are some guidelines for each reference type:

Reference type	Used to
Absolute	Select a specific member, for example, 201801. This is comparable to an absolute reference (\$A\$1) in Excel.
Branch Relative	Select the 1 st , 2 nd , 3 rd , etc. member of a dimension-branch, regardless of which members are included in the <i>filter</i> . This



Flex Views

option provides some flexibility to meta-data changes (new accounts, departments, time periods, etc.) or *filter* changes.

View Relative Select data based on the closest member specified flex function, for example, **BranchData**, defined on the left in the row, above on the column, or on the **Page** edge of the *view*. This is useful when you are displaying *view* data with row or column headers in a rectangular range, or if you want your *view* data to navigate or be expanded by pages. This option is only available for **Data**, **DataReadOnly**, **Comment**, and **Hyperlink** functions. (The **Current Edge** option uses the edge assignments of the underlying *view* to select row, column, or page edge. Use it in flex tables to support rotations.)

Worksheet Reference the absolute value of the argument from another cell on the worksheet. The reference may be Excel relative (A13), absolute (\$A\$13), or partially relative (\$A13 or A\$13). This reference type affords consistency with how many users build CRF worksheets.

Indirect Adopt like arguments from a flex formula in another cell. For example, if you want to specify the same account member that appears in cell A13, which may have a **BranchData**, **Data**, or **Comment** function, you can simply use an **Indirect** reference to A13. **Indirect** references have the same Excel absolute/relative options as **Worksheet** references.

Keyword Supply an argument value based on a CONTROL® *keyword*, evaluated for the current user and *model* scope of the flex *view*. The resolved *keyword* replacement value may be any other type of reference (except **Worksheet** and **Indirect**).

Page Relative Select the 1st, 2nd, 3rd, etc. member of the dimension-branch as it appears on the row, column or **Page** edge of a *view*, taking into consideration zero suppression, sorting, exceptions, compression/expansion/hiding states of the members.



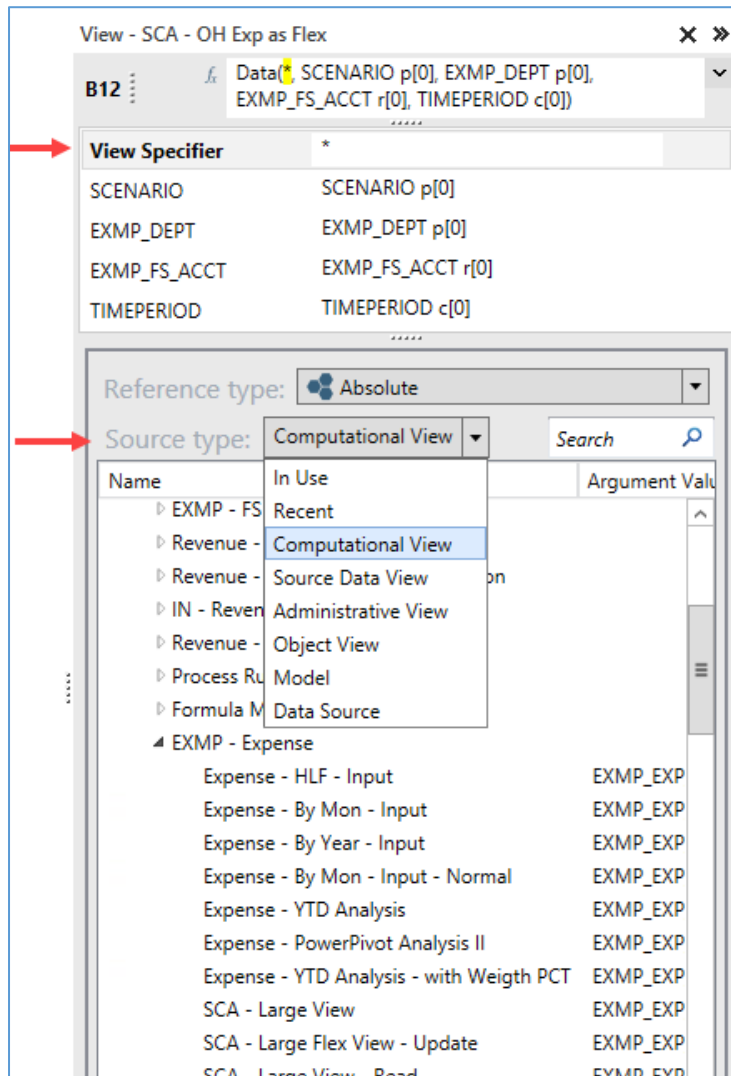
Flex Views

We expect the KCI consulting team will develop a more refined set of guidelines for these reference types as customers begin developing flex views.

Source type

Some arguments in flex functions may have a large number of possible values, depending on the complexity of your CONTROL® application. To simplify the selection of the desired value, the range of values are broken into logical groups.

The **Source Type** serves to filter and limit the available values displayed in the list or tree below it.



Flex Views

Since most of the flex functions include the **View Specifier** argument, which defines the source of the data or meta-data, we will define the different source types for this argument. The **Source Type** for the **View Specifier** argument filters the available objects that you can choose from as the source data.

Source type	Will list
In Use	<i>Active views</i>
Recent	<i>Most recently used views</i>
Computational View	<i>Computational views</i>
Source Data View	<i>Source Data views</i>
Administrative View	<i>Administrative views</i>
Object View	<i>Object views</i>
Model	<i>Models</i>
Datasource	<i>Datasources</i>

Note: There are other flex function arguments that include **Source Types** that reference other object types; for example, *levels*, *hierarchies*, etc.

Dealing with ranges

You can use the Flex Pane to define or modify the flex formula for either a single cell or a range of cells. To define a single cell, select that one cell on the worksheet. To define a range of cells, select the range of cells on the worksheet. The flex formula, function, and arguments in the Flex Pane only reflect the active worksheet cell identified in the upper left corner of the Flex Pane. However, when you select a range of cells, a message indicating the number of cells that will be affected displays in the Flex Pane, and the title of the **Apply** button will change to **Apply to range...**



Flex Views

The screenshot shows the Flex Views interface for a view named 'View - SCA - OH Exp as Flex'. The interface includes a data table on the left and a configuration pane on the right. A red arrow points to the 'Apply to range (26 cells)' button in the configuration pane. Another red arrow points to a warning message: 'Selections not matched: 1 values will be duplicated when applied to current Excel range (26 cells)'. The configuration pane shows 'Reference type: Absolute' and a table with 'FCST: Forecast' and 'SCENARIO FCST'.

Oct-18	Nov-18	Dec
88	256	
983	1,078	
469	395	
1,111	1,223	
446	522	
472	602	
68	205	
682	460	
354	538	
118	192	
510	576	
508	593	
5,722	6,384	

When you click **Apply**, all changes will propagate to the flex formulas in the selected range.

For example, if you only change the reference type of one or more arguments, say from **View Relative** to **Absolute**, all the selected cells will have those arguments changed to the absolute member ID appropriate to that cell.

If you change a specific *dimension* argument, say Project, to an absolute member ID of **PROJECTTOTAL**, all the selected cells will then refer to the total of all projects when you click **Apply to range**.

Some arguments and reference types allow you to select multiple values in the Flex Pane.



Flex Views

Depending on the number of rows and columns selected, CONTROL® will try to make a sensible assignment of the selected items to the selected cells. In the example below, the Flex Pane has nine values selected, but the worksheet only has four cells selected, so CONTROL® applies the first four selected members in the Flex Pane to the four rows selected on the worksheet (CAN, ESP, FRA, and ITA).

Only 4 rows selected on the worksheet

9 values selected in the Flex Pane

Name	Argument Value
CAN: Canada	EXMP_CNTRY CAN
ESP: Spain	EXMP_CNTRY ESP
FRA: France	EXMP_CNTRY FRA
ITA: Italy	EXMP_CNTRY ITA
JPN: Japan	EXMP_CNTRY JPN
THA: Thailand	EXMP_CNTRY THA
TWN: Taiwan, Province of C	EXMP_CNTRY TWN
USA: United States	EXMP_CNTRY USA
TOT_CO: Total Company	EXMP_CO TOT_CO

Selections not matched: first 4 (from 9) values will be applied to current Excel range.

If you select a single blank cell in the worksheet, and then select a function that defines a range, for example, **ViewData**, or select multiple values for one or more arguments, if the area below and to the right of the selected cell is empty, CONTROL® will fill out the range with the specified data.



Flex Views

Note: When using the **Worksheet** or **Indirect** reference types, the cell reference will update across the selected range of cells on the worksheet similar to Excel's functionality when you copy and paste a cell reference. For example, in the screenshot below, the flex function's **Time Period** argument for cell **B12** will be **B\$11** down all the rows in column B, but the value will be **C\$11** down all the rows in column C, etc.

	Jan-19	Feb-19	Mar-19	Apr-19	May-19
Full Time Equivalent	919	923	931	937	943
Total Employee Compensation	56,918,525	57,204,565	57,501,989	57,804,765	58,070,489
Travel & Entertainment	1,249,452	1,255,730	1,262,040	1,268,382	1,274,756
Supplies & Services	5,388,811	5,415,890	5,443,106	5,470,458	5,497,948
Communications & IT	1,362,792	1,369,640	1,376,523	1,383,440	1,390,392
Freight & Shipping	1,432,092	1,439,289	1,446,521	1,453,790	1,461,096
Occupancy	2,502,826	2,515,403	2,528,044	2,540,747	2,553,515
Maintenance	2,281,680	2,293,146	2,304,669	2,316,250	2,327,890
Insurance	1,842,020	1,851,277	1,860,580	1,869,929	1,879,326
Utilities	1,749,343	1,758,133	1,766,968	1,775,847	1,784,771
Depreciation & Amortization	4,021,791	4,042,001	4,062,312	4,082,726	4,103,242
Other Expenses	559,205	562,015	564,839	567,677	570,530
Total Overhead Expenses	79,308,537	79,707,089	80,117,591	80,534,014	80,913,955

For more details on all the flex functions and their arguments, see [Flex Functions](#) in Appendix A.

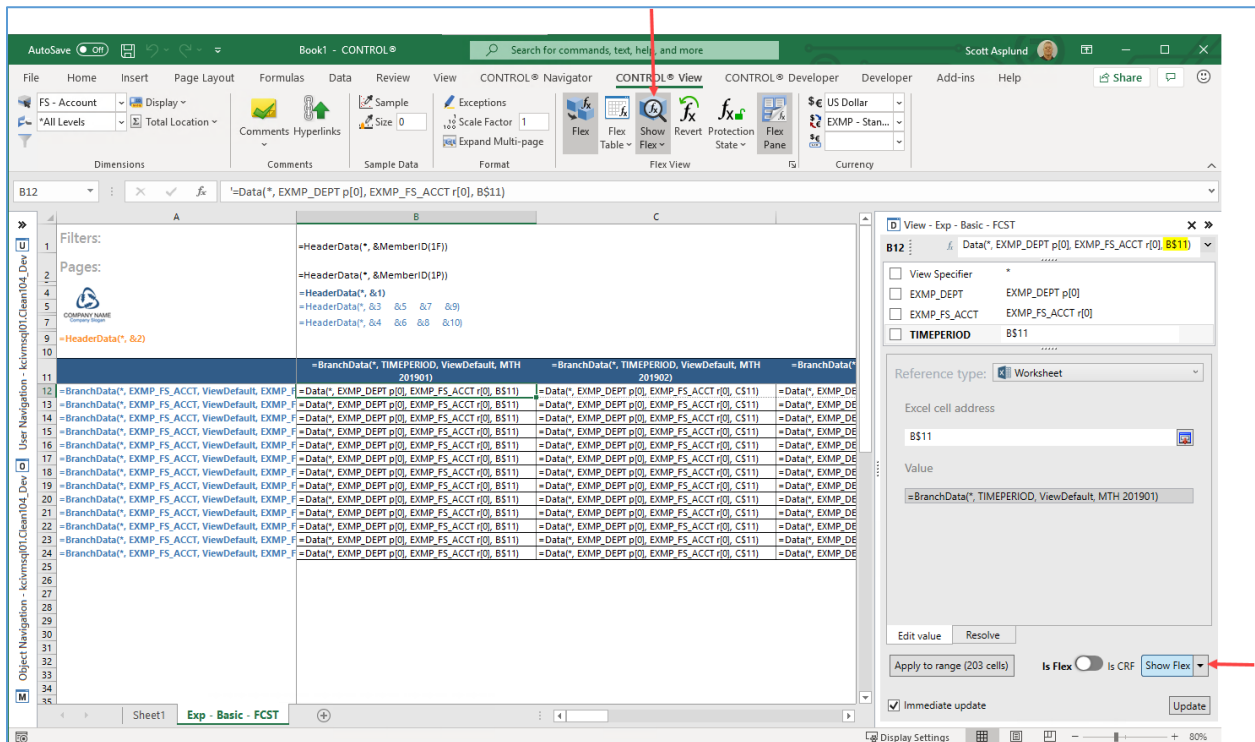
Authoring using the worksheet

While the Flex Pane offers detailed options for defining flex formulas, there are situations where it is easier or more convenient to simply edit the flex formula on the worksheet.

To see the flex formulas rather than the values, click **Show Flex** either on the CONTROL® View ribbon, or on the bottom right of the open Flex Pane.



Flex Views



The columns of the *view* automatically expand to improve readability of the flex formulas. An equal sign precedes each flex formula in every cell. Cells with no flex formula continue to display their value. Cells with Excel formulas also display their values because they would likely display **#VALUE!** errors if they referenced flex values that momentarily display as flex formulas.

You can copy, paste, edit, find, and replace the flex formulas on the worksheet as you wish. You can flip back and forth between flex formulas and values to verify that you are getting the results that you want. You can also copy and paste cells containing flex formulas even when you are not showing the flex formulas.

Even though you can edit the **Persistent** form of a flex formula, it is the best practice to only edit the **Understandable** form of a flex formula because it is more succinct, and is consistent with the syntax in the Flex Pane.

Partial formulas

As you can see, while flex formulas are very flexible and powerful, they can be lengthy and a bit intimidating.

If you want to focus on one or more specific arguments, select the check box next to the arguments in the Flex Pane. For example, to display the accounts only, select the check box next to **EXMP_FS_ACCT**:



Flex Views

The screenshot shows a Flex View interface with a data table and a right-hand pane. The data table has columns A, B, and C. Row 12 is highlighted with a red box, showing the formula `=Data(EXMP_FS_SUMMARY 100_FTE)` in both columns B and C. The right-hand pane, titled "View - SCA - BranchRelative", shows a list of arguments with checkboxes. The argument `EXMP_FS_ACCT` is checked, and its value is `EXMP_FS_SUMMARY 100_FTE`. A red arrow points from the right-hand pane to the data table.

Any cell containing a formula that uses the selected arguments will show only the function name and the selected arguments.

You can edit or perform replacements when showing partial arguments, and the arguments not displaying will remain unchanged. This feature is particularly useful when dealing with views that include many *dimensions*.

Errors

Similar to Excel formula, flex formulas display errors using a pound sign to prefix the error condition.

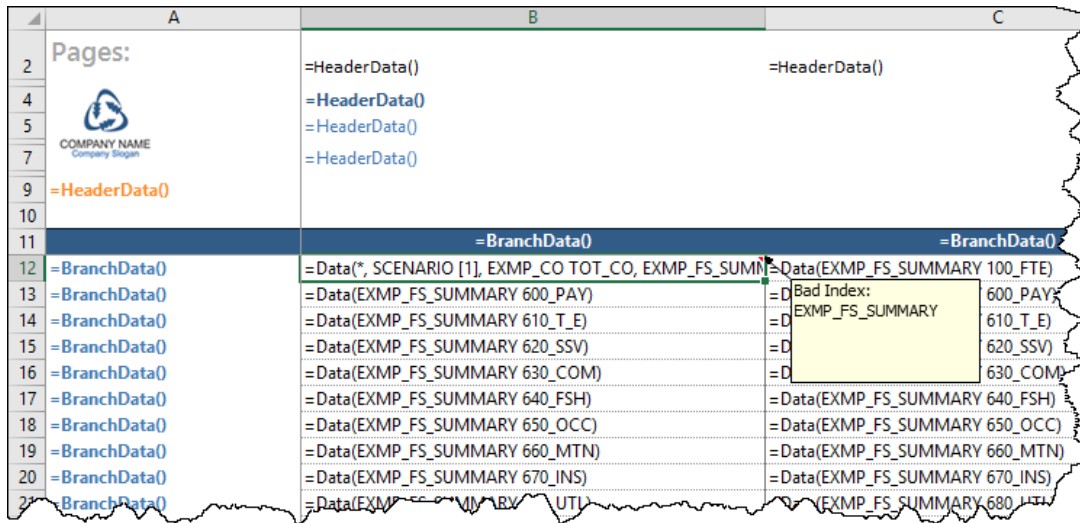
	A	B	C	D	E
2	Pages:	TOT_CO	FCST		
4		EXMP - Financial Statements SCA - BranchRelative			
7	COMPANY NAME Company Slogan	Department Total Company :: Scenario Forecast			
9	NOTTRANSLATION in Thousands				
11		Jan-18	Feb-18	Mar-18	Apr-18
12	Full Time Equivalent	#BadIndex: SCENARIO EXMP_FS_ACCT	246	181	
13	Total Employee Compensation	1,036	810	1,096	795
14	Travel & Entertainment	414	653	518	
15	Supplies & Services	1,254	1,309	1,213	
16	Communications & IT	347	367	312	3
17	Freight & Shipping	696	555	691	



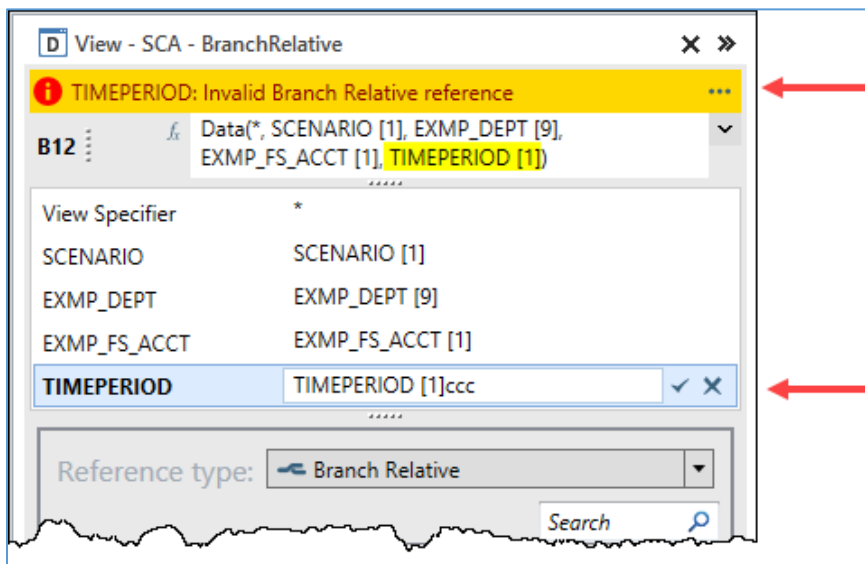
Flex Views

In most cases, flex formula errors will not prevent the flex view from opening, as cells without errors should populate correctly. However, with errors present, the content of the view will be suspect because formulas or references based on the cells with errors are also likely to be incorrect.

When you edit flex formulas in the worksheet, Excel cell comments will identify any cells that contain errors, after you refresh the worksheet:



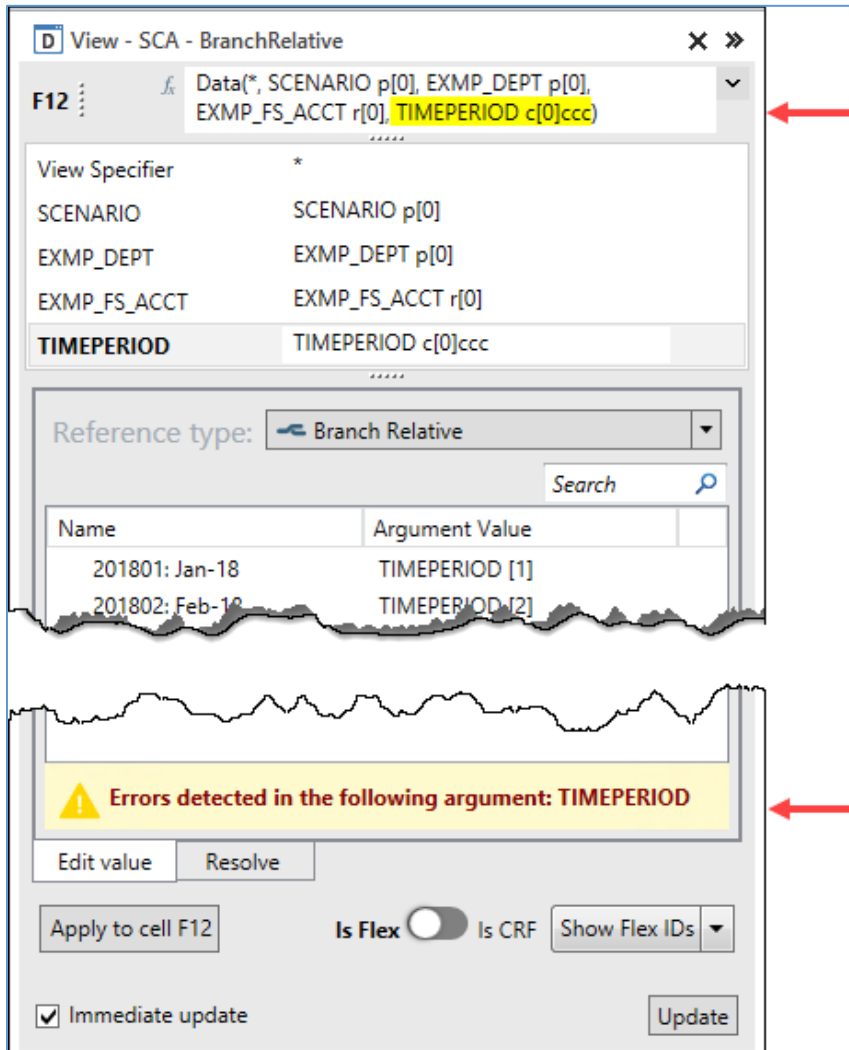
If you enter an invalid argument in the Flex Pane, the error displays above the **Flex formula** box.



If you enter an invalid flex formula in the box, then the error displays at the bottom of the Flex Pane.



Flex Views



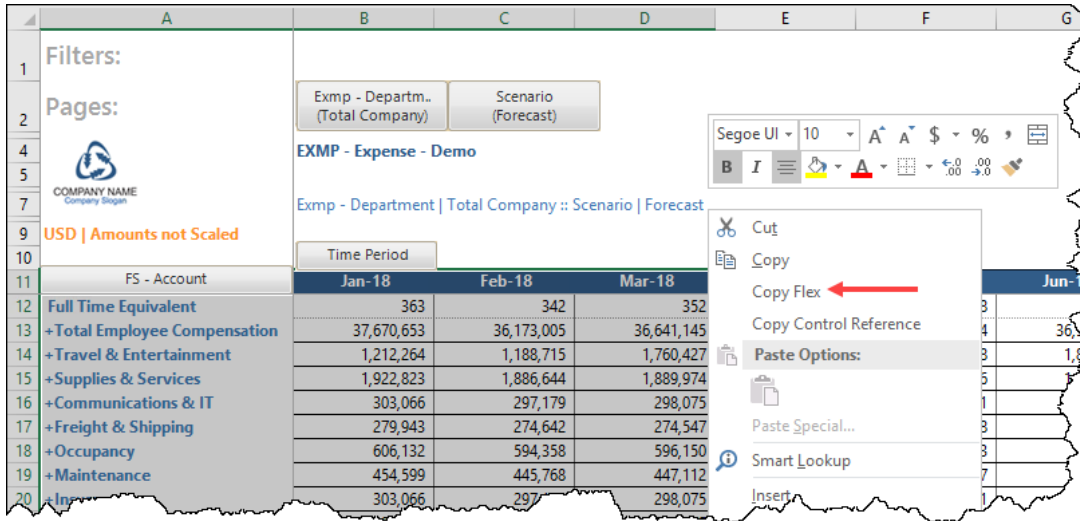
Creating a new flex view

You can create a new flex view from a blank worksheet.

1. From an open view, highlight the information you want to copy, right-click, and click **Copy Flex** from the shortcut menu.



Flex Views



The **Copy Flex** command converts the information from the standard view to flex formulas.

- Go to a blank worksheet and paste the information.

The screenshot shows a blank worksheet with the copied data from the previous image pasted into it. The data is now represented as flex formulas using relative references.

	Jan-18	Feb-18	Mar-18
Full Time Equivalent	=B3	=C3	=D3
+Total Employee Compensation	=B4	=C4	=D4
+Travel & Entertainment	=B5	=C5	=D5
+Supplies & Services	=B6	=C6	=D6
+Communications & IT	=B7	=C7	=D7
+Freight & Shipping	=B8	=C8	=D8
+Occupancy	=B9	=C9	=D9
+Maintenance	=B10	=C10	=D10
+Insurance	=B11	=C11	=D11
+Utilities	=B12	=C12	=D12
+Depreciation & Amortization	=B13	=C13	=D13
+Other Expenses	=B14	=C14	=D14
-Total Overhead Expenses	=B15	=C15	=D15

If you use **Copy Flex** and your copied range includes branch data on the rows and columns, then the resulting flex formulas will use the **View Relative** reference type.



Flex Views

The screenshot shows a spreadsheet with a table of expenses for Jan-18. The table is highlighted with a red box. The Flex View dialog box is open, showing the formula for cell S9. The formula is: `Data(*, SCENARIO p[0], EXMP_DEPT p[0], EXMP_FS_ACCT r[0], TIMEPERIOD c[0])`. The View Specifier is set to `*`. The arguments are: `SCENARIO SCENARIO p[0]`, `EXMP_DEPT EXMP_DEPT p[0]`, `EXMP_FS_ACCT EXMP_FS_ACCT r[0]`, and `TIMEPERIOD TIMEPERIOD c[0]`. The Reference type is set to `View Relative`.

	Jan-18
Full Time Equivalent	363
Total Employee Compensation	37,670,653
Travel & Entertainment	1,212,264
Supplies & Services	1,922,823
Communications & IT	303,066
Freight & Shipping	279,943
Occupancy	606,132
Maintenance	454,599
Insurance	303,066
Utilities	303,066
Depreciation & Amortization	1,363,797
Other Expenses	3,465,342
Total Overhead Expenses	47,884,750

However, if your copied range does not include branch data, then the resulting flex formulas will use the **Absolute** reference type.

The screenshot shows the same spreadsheet as above, but the cell containing the total value 47,884,750 is highlighted with a red box. The Flex View dialog box is open, showing the formula for cell S22. The formula is: `Data(EXMP_EXP;EXMP_EXPENSE_DEMO, SCENARIO SCENARIO FCST, EXMP_DEPT EXMP_CO TOT_CO, EXMP_FS_ACCT EXMP_FS_SUMMARY 699_OHX, TIMEPERIOD MTH 201801)`. The View Specifier is set to `EXMP_EXP;EXMP_EXPENSE_DEMO`. The arguments are: `SCENARIO SCENARIO FCST`, `EXMP_DEPT EXMP_CO TOT_CO`, `EXMP_FS_ACCT EXMP_FS_SUMMARY 699_OHX`, and `TIMEPERIOD TIMEPERIOD MTH 201801`. The Reference type is set to `Absolute`.

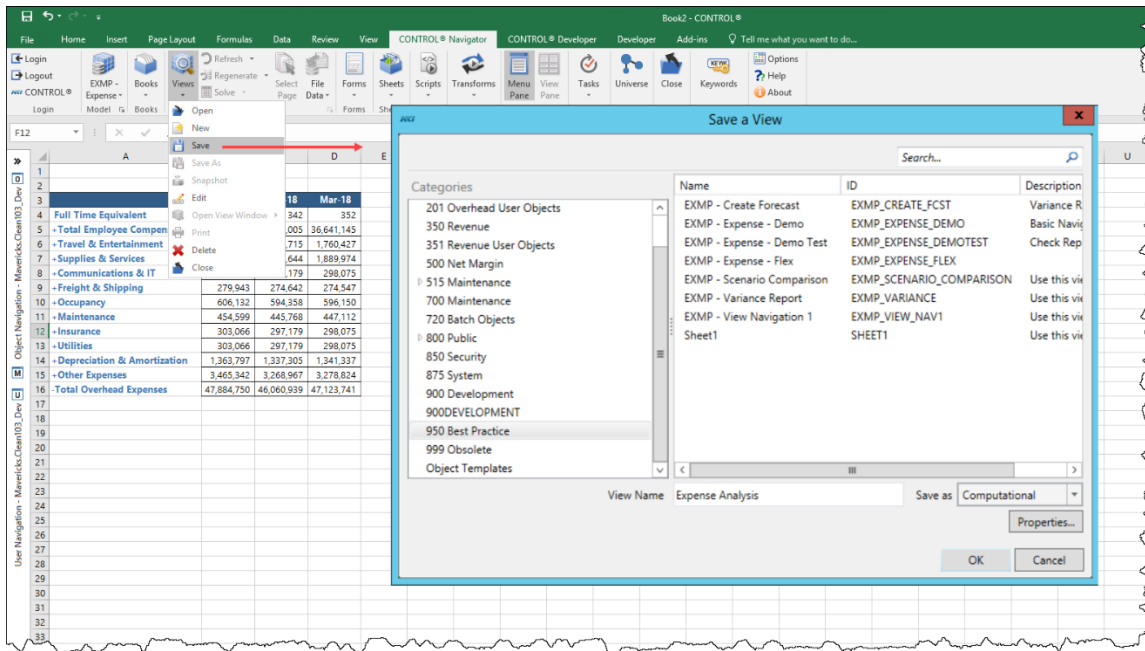
	Jan-18
Full Time Equivalent	363
Total Employee Compensation	37,670,653
Travel & Entertainment	1,212,264
Supplies & Services	1,922,823
Communications & IT	303,066
Freight & Shipping	279,943
Occupancy	606,132
Maintenance	454,599
Insurance	303,066
Utilities	303,066
Depreciation & Amortization	1,363,797
Other Expenses	3,465,342
Total Overhead Expenses	47,884,750

Note You can use **Ctrl + Shift + V** to paste the flex formulas with absolute references.

- From the CONTROL® Navigator ribbon, you can select **Views > Save** to save the worksheet as a new flex view.

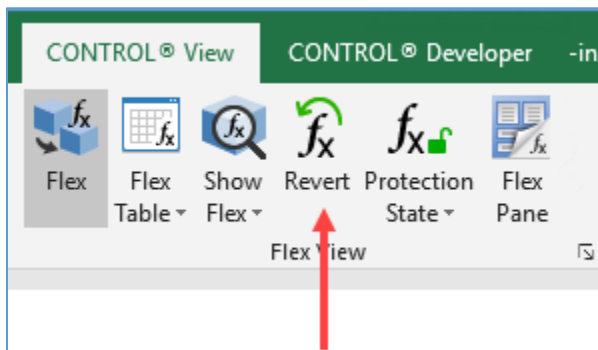


Flex Views



Undoing flex formula changes

If you make unwanted changes to your flex formulas, you can go to the CONTROL® View ribbon and click **Revert** to undo all flex formula changes since the last time you saved the view.



Using model view specifiers

In the past, you always had to reference a *view* in order to display data in a worksheet, for example, *views*, workbook *sheets*, CRFs, and *books*. The reason for that is because data in CONTROL® is dynamically computed and a *view's* context (*filters* and *branches*) determines the resulting values. For example, what data to retrieve from the database, which *hierarchy* to apply by scenario, and which *level* members to include.

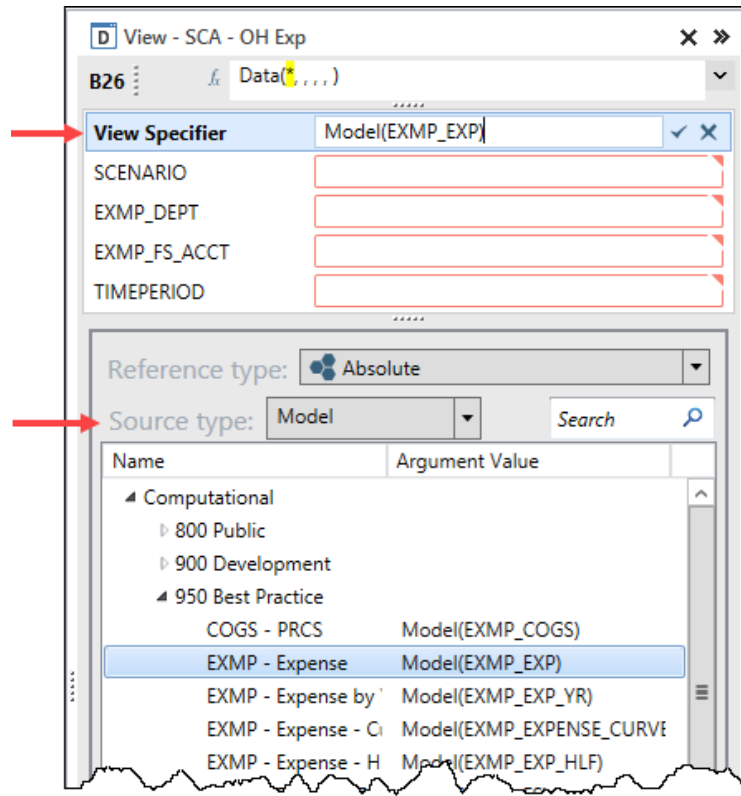


Flex Views

However, we noticed when users worked with CRFs they did not always know which *view* to use, they sometimes created a new *view* to include the information they wanted, or they could not create a *view* with the information they needed.

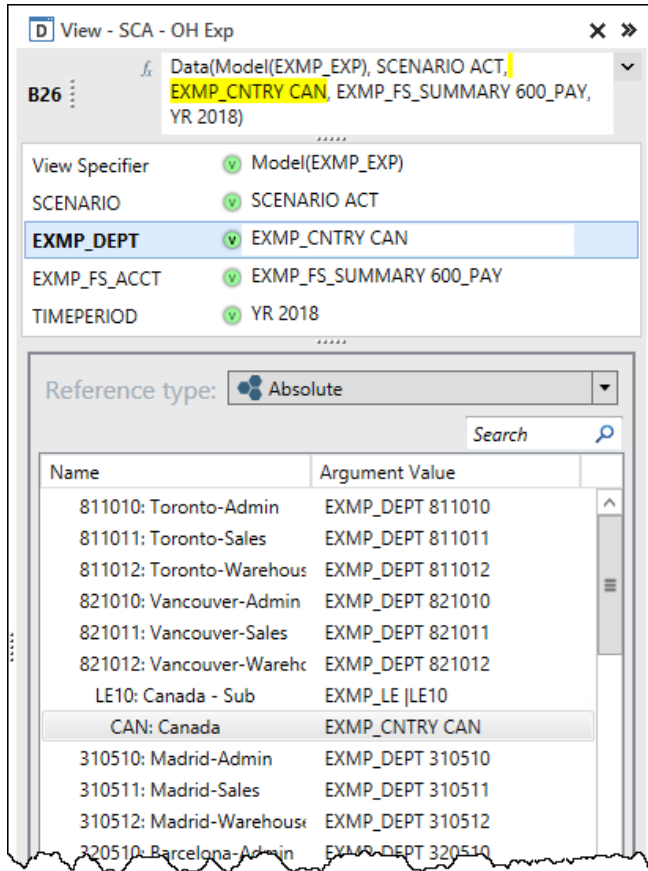
With flex *views*, you no longer have to reference a *view* to retrieve data from the database. flex formulas can directly reference a *model* without the *filter* or dimension-branch restrictions of a *view*.

If a flex function includes a **View Specifier** argument, you can select **Model** as the **Source Type**.



Then, you can select members from any of the *dimensions* in the *model* to retrieve data.

Flex Views



When you click **Apply**, the data appears on the worksheet in a “deferred” mode, so you will see **999** for any data cells.

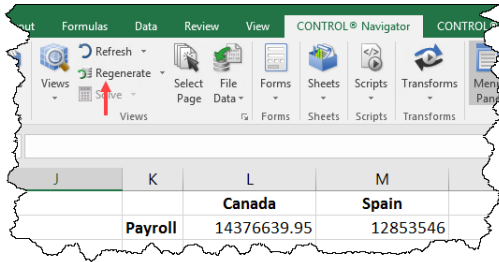
K	L	M
	Canada	Spain
Payroll	999	999

For large *models*, the initial carve could be a time consuming process. Deferring the carve avoids this potential delay.

Once your flex *view* is complete, CONTROL® is able to analyze all of the data and meta-data you have requested from the *model* and optimize the data retrieval and any calculations. This process is known as **Carve Optimization** and is discussed below. To retrieve the data, go to the CONTROL® Navigator ribbon and click **Regenerate**.



Flex Views



Note Using a Model view specifier does not bypass any object or data access security restrictions.

Customizing the member lists

Model view specifiers allow you to see all the data in a *model* rather than the limited and targeted data included in *views*.

The default branch for each *dimension* determines the display and order of members listed in the Flex Pane. In addition, both object and data access restrictions will limit the display of available members.

You can customize the list of members or the branch, for a user or all users, by using a special *view* with the ID of **FlexViewDesign**.

By default, this *view* has a dynamic definition and has an all *model*, all user scope.

You can make the following customizations:

- Add dimension-branches with specific *filters* or branches that will override the default behavior. Both *filters* and branches may contain *keywords* with replacement values scoped by *model* or user.
- Add *view* scopes with different *filter* and branch definitions for specific *models*, categories, users, or user groups.
- Define the **&FlexViewDesign** *keyword* whose replacement value is the ID of the *view* used to associate *model* and user scopes.

Flex Tables

With flex *views*, you can arrange CONTROL® information any way you like on your worksheet; however, they do not adjust automatically for new members, data changes with zero suppression, or other navigational parameters of the underlying *view*. To overcome this limitation, and to support update of source data and object meta-data, beginning with CONTROL® 10.4 you can add one or more flex tables on your worksheet that will dynamically adjust in size and content.



Flex Views

What is a Flex Table?

A flex table is an Excel table that contains CONTROL® flex formulas, in one or more columns. It can also have other columns without flex formulas. It can be adjustable – growing or shrinking in rows or columns, or fixed in size.

With a flex table, you can define what you want, and CONTROL® will automatically fill it in for you.

A flex table:

- Must have a single header row that contains flex formulas for meta-data or values that are referenced to determine the content of each column. The header can contain BranchData or NestedBranchData functions.
- Must have at least one data row below the header. If it has more than one row, the contents of all the rows in each column must be uniform if the column contains flex formulas. This means that all the rows have the same flex function, View Specifier, and argument values. An argument may differ between rows if the Reference Type is Branch or Page Relative.
- Must have at least one column with a flex formula.
- May have any number of columns without a flex formula, used for pure Excel calculations.
- May have the following flex functions in the data area:
 - Data or DataReadOnly
 - Comment or Hyperlink
 - List
 - BranchData or NestedBranchData
 - MemberData

Creating a Flex Table

To create a flex table:

1. Start with an open flex view.
2. Select a rectangular range that has uniform data in each column.
3. On the **CONTROL® View** ribbon, in the **Flex View** group, click **Flex Table**.



Flex Views

The screenshot shows the CONTROL® software interface. The ribbon is set to 'CONTROL® View'. A red arrow points to the 'Flex Table' button in the 'Flex View' group. The spreadsheet below shows a table with columns for months from Jan-19 to Sep-19 and rows for various expense categories. The table is partially visible, showing rows 12 through 24.

	Jan-19	Feb-19	Mar-19	Apr-19	May-19	Jun-19	Jul-19	Aug-19	Sep-19
Full Time Equivalent	905	911	963	956	915	990	916	923	981
Total Employee Compensation	4,166,194	4,239,003	4,034,890	3,955,488	5,418,899	4,058,136	4,076,258	3,927,785	4,166,212
Travel & Entertainment	1,178,634	1,184,465	1,190,326	1,196,215	1,202,085	1,209,630	1,213,063	1,219,890	1,161,531
Supplies & Services	5,325,409	5,340,036	5,566,295	5,553,888	5,397,942	5,683,988	5,445,178	5,511,149	5,845,226
Communications & IT	1,347,770	1,351,668	1,405,704	1,403,202	1,366,699	1,435,078	1,378,775	1,388,430	1,459,947
Freight & Shipping	1,428,902	1,435,472	1,452,719	1,457,987	1,456,063	1,478,755	1,470,374	1,478,455	1,553,347
Occupancy	2,492,191	2,502,680	2,548,702	2,554,738	2,536,741	2,596,146	2,561,100	2,576,081	2,678,678
Maintenance	2,270,969	2,280,331	2,325,476	2,330,341	2,310,996	2,369,117	2,333,072	2,346,883	2,459,243
Insurance	1,826,208	1,832,360	1,891,294	1,890,730	1,854,387	1,929,275	1,871,287	1,883,651	1,969,991
Utilities	1,739,409	1,746,249	1,786,264	1,788,915	1,769,104	1,820,383	1,785,809	1,796,665	1,877,583
Depreciation & Amortization	3,972,507	3,983,038	4,158,048	4,147,560	4,025,509	4,246,637	4,060,510	4,089,848	4,309,436
Other Expenses	559,205	562,015	564,839	567,677	570,530	574,527	576,278	579,255	604,626
Total Overhead Expenses	26,307,398	26,457,317	26,924,557	26,846,742	27,908,955	27,401,671	26,771,705	26,798,092	28,085,820

CONTROL® will determine the appropriate boundaries for the table, turn it into an Excel table, apply table formatting and data filters, adjust the flex IDs as necessary, and using default properties (based on the table's contents) define the flex table.

The screenshot shows the CONTROL® software interface. The ribbon is set to 'CONTROL® View'. A red arrow points to the 'Flex Table' button in the 'Flex View' group. The spreadsheet below shows a table with columns for months from Jan-19 to Jul-19 and rows for various expense categories. The table is partially visible, showing rows 12 through 24.

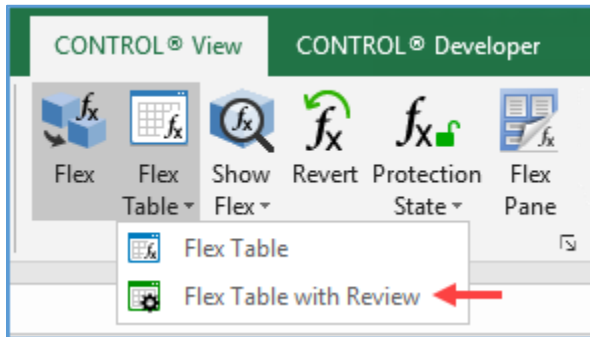
	Jan-19	Feb-19	Mar-19	Apr-19	May-19	Jun-19	Jul-19
Full Time Equivalent	905	911	963	956	915	990	916
Total Employee Compensation	144,634,175	150,241,462	149,667,824	149,591,313	170,525,125	149,591,286	150,555,416
Travel & Entertainment	25,209,458	27,762,988	27,574,970	26,254,958	27,394,819	26,635,171	27,269,586
Supplies & Services	52,536,429	57,563,979	57,394,029	54,754,413	56,864,845	55,618,127	56,629,556
Communications & IT	14,072,707	15,427,152	15,375,593	14,666,717	15,238,066	14,895,311	15,175,158
Freight & Shipping	25,720,225	28,301,262	28,124,218	26,791,675	27,931,786	27,181,783	27,810,054
Occupancy	38,587,946	42,418,805	42,186,389	40,219,480	41,872,651	40,800,992	41,704,866
Maintenance	33,059,854	36,389,207	36,480,920	34,762,308	35,869,210	35,160,572	35,912,663
Insurance	19,105,475	20,782,033	20,764,017	19,952,364	20,636,658	20,315,312	20,435,355
Utilities	24,782,045	27,188,363	26,959,598	25,738,099	26,873,061	26,154,547	26,688,690
Depreciation & Amortization	34,129,562	37,313,132	37,526,929	35,890,474	36,875,048	36,387,068	36,846,473
Other Expenses	8,113,258	8,719,231	8,767,913	8,521,878	8,743,046	8,717,577	8,563,733
Total Overhead Expenses	419,951,133	452,107,613	450,822,401	437,143,679	468,824,315	441,457,748	447,591,551

If the selected cell is in an existing Excel table, that table's boundaries, formatting and other properties are used.



Flex Views

If you want to review and adjust the **Flex table properties** before the conversion, then instead of clicking **Flex Table**, click the down arrow on **Flex Table**, and click **Flex Table with Review**.




For more details, see [Flex table properties](#) below.

Flex table examples

Computational view

Here is an example of a computational view converted to a flex table. **Suppress zero rows** is enabled and the data is sorted on **Actual YTD-Aug-2019**. As you select different products on the Pages edge, the row data will expand or compress and the **Total** row will recalculate.

Flex Views

	A	B	C	D	E	F	G
1		Revenue - Reporting Top Customers - Flex Table					
2	COMPANY NAME	Adjustment Total Adjustment					
4	Company Slogan	Product Total Product :: Revenue Account Net Revenue					
6	USD Amounts not Scaled						
7							
8	Column1	Actual Aug-2019	Budget Aug-2019	CY Actual vs CY Budget Var	Actual YTD-Aug-2019	Budget YTD-Aug-2019	CY Actual vs CY Budget YTD Var
9	Customer 0027	7,410,921	7,073,445	(337,476)	58,156,574	59,154,858	998,285
10	Customer 0026	7,333,216	9,662,512	2,329,296	57,663,595	79,624,125	21,960,531
11	Customer 0025	7,272,710	9,582,787	2,310,077	57,187,814	78,967,149	21,779,335
12	Customer 0024	7,212,204	9,503,062	2,290,858	56,712,033	78,310,173	21,598,139
13	Customer 0023	7,151,698	9,423,336	2,271,639	56,236,253	77,653,196	21,416,944
14	Customer 0022	7,091,191	9,343,611	2,252,420	55,760,472	76,996,220	21,235,748
15	Customer 0021	7,030,685	9,263,886	2,233,201	55,284,691	76,339,244	21,054,552
16	Customer 0020	6,970,179	6,731,170	(239,009)	54,808,911	53,427,188	(1,381,722)
17	Customer 0019	6,909,673	6,672,739	(236,934)	54,333,130	52,963,402	(1,369,728)
18	Customer 0018	6,849,167	6,614,307	(234,859)	53,857,349	52,499,616	(1,357,734)
19	Customer 0017	6,788,661	6,565,187	(223,474)	53,381,569	51,425,419	(1,956,150)
20	Customer 0016	6,728,155	6,506,672	(221,482)	52,905,788	50,967,073	(1,938,715)
21	Customer 0015	6,667,648	6,448,158	(219,490)	52,430,008	50,508,727	(1,921,280)
22	Customer 0014	6,607,142	8,705,811	2,098,668	51,954,227	71,740,409	19,786,182
23	Customer 0013	6,546,636	8,626,085	2,079,449	51,478,446	71,083,433	19,604,987
24	Customer 0012	6,486,130	8,546,360	2,060,230	51,002,666	70,426,457	19,423,791
25	Customer 0011	6,425,624	6,080,464	(345,160)	50,526,885	47,181,347	(3,345,538)
26	Customer 0010	6,365,118	6,023,208	(341,910)	50,051,104	46,737,070	(3,314,035)
27	Customer 0009	6,304,611	5,965,952	(338,660)	49,575,324	46,292,792	(3,282,532)
28	Customer 0008	6,244,105	5,908,696	(335,409)	49,099,543	45,848,514	(3,251,029)
29	Customer 0007	6,183,599	5,851,440	(332,159)	48,623,762	45,404,236	(3,219,526)
30	Customer 0006	6,123,093	5,794,184	(328,909)	48,147,982	44,959,958	(3,188,023)
31	Customer 0005	6,062,587	5,736,928	(325,659)	47,672,201	44,515,681	(3,156,521)
32	Customer 0004	6,002,081	5,679,672	(322,409)	47,196,421	44,071,403	(3,125,018)
33	Customer 0003	5,941,575	5,622,416	(319,159)	46,720,640	43,627,125	(3,093,515)
34	Customer 0002	5,881,068	5,565,160	(315,908)	46,244,859	43,182,847	(3,062,012)
35	Customer 0001	5,809,054	5,497,014	(312,040)	45,678,585	42,654,068	(3,024,517)
36	Total	143,870,898	143,870,898	143,870,898	143,870,898	143,870,898	143,870,898

Source data view

Here is an example where the *view* includes a flex *view* (**Budget – Summary**) and two flex tables (**Budget – Detail** and **Actual – Detail**). The user can reference the actual transactions to create a detail budget. The flex *view* uses an Excel formula to reference the **Budget – Detail Total**.



Flex Views

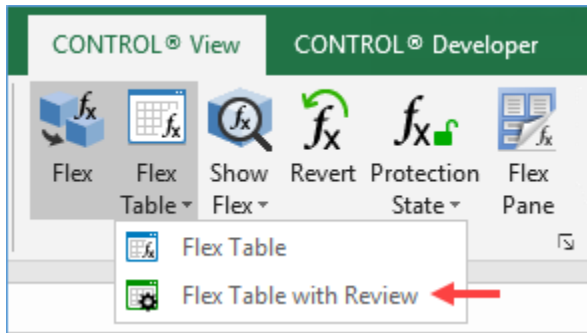
Budget - Summary													
	Jan-20	Feb-20	Mar-20	Apr-20	May-20	Jun-20	Jul-20	Aug-20	Sep-20	Oct-20	Nov-20	Dec-20	Year 2020
6916 New Depreciation Expense	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	180,000.00

Budget - Detail															
Dept ID	Acct ID	Description	Jan-20	Feb-20	Mar-20	Apr-20	May-20	Jun-20	Jul-20	Aug-20	Sep-20	Oct-20	Nov-20	Dec-20	Year 2020
110110	6916	Executive Desk	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	180,000.00
110110	6916	Hon File Cabinet	-	-	-	-	-	-	-	-	-	-	-	-	-
110110	6916	Bookshelf	-	-	-	-	-	-	-	-	-	-	-	-	-
110110	6916	Thinkpad S20 Edge	-	-	-	-	-	-	-	-	-	-	-	-	-
110110	6916	Ergonomic Chair	-	-	-	-	-	-	-	-	-	-	-	-	-
110110	6916	Dell Inspiron 17 Series	-	-	-	-	-	-	-	-	-	-	-	-	-
Total			15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	15,000.00	180,000.00

Actual - Detail															
Dept ID	Acct ID	Description	Jan-19	Feb-19	Mar-19	Apr-19	May-19	Jun-19	Jul-19	Aug-19	Sep-19	Oct-19	Nov-19	Dec-19	Year 2019
110110	6915	Executive Desk	18,981.60	19,016.98	18,172.85	19,269.19	19,366.02	19,501.71	19,561.15	19,659.44	17,702.35	17,985.98	18,599.62	18,920.73	227,797.62
110110	6915	Hon File Cabinet	19,955.01	20,052.29	20,156.07	20,237.36	20,339.15	20,501.80	20,564.28	20,567.62	18,610.17	18,908.33	19,553.44	19,891.02	239,479.55
110110	6915	Bookshelf	968.17	973.04	977.93	982.84	987.78	994.70	997.73	1,002.75	1,052.43	1,071.77	1,116.37	1,138.37	12,283.88
110110	6915	Thinkpad S20 Edge	10,649.90	10,703.42	10,757.20	10,811.26	10,865.59	10,941.72	10,975.06	11,030.22	11,576.70	11,789.49	12,280.06	12,522.02	134,902.64
110110	6915	Ergonomic Chair	5,082.91	5,108.45	5,134.12	5,159.92	5,185.85	5,222.18	5,238.10	5,264.42	5,529.24	5,626.80	5,860.94	5,976.42	64,385.35
110110	6915	Dell Inspiron 17 Series	16,791.41	16,873.79	16,960.60	17,045.82	17,131.48	17,251.52	17,304.09	17,391.05	15,659.77	15,910.67	16,453.51	16,737.57	207,513.28
Total			72,429.01	72,792.97	73,158.77	73,526.40	73,895.88	74,413.63	74,640.41	75,015.49	70,126.67	71,293.04	73,863.93	75,186.12	880,342.31

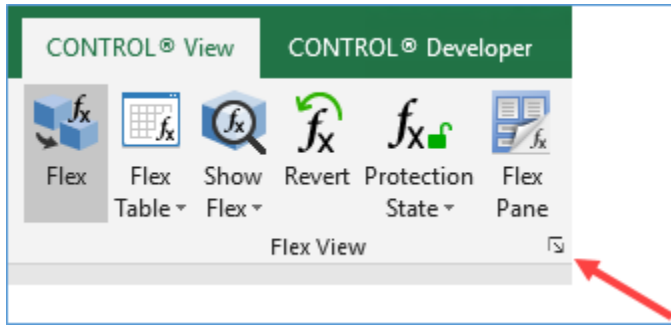
Flex table properties

There are properties that control the behavior of flex tables. You can define them before or after you create a flex table. If you want to review and adjust the flex table properties before the conversion, click on the **CONTROL® View** ribbon, in the **Flex View** group, click the down arrow on **Flex Table**, and click **Flex Table with Review**.

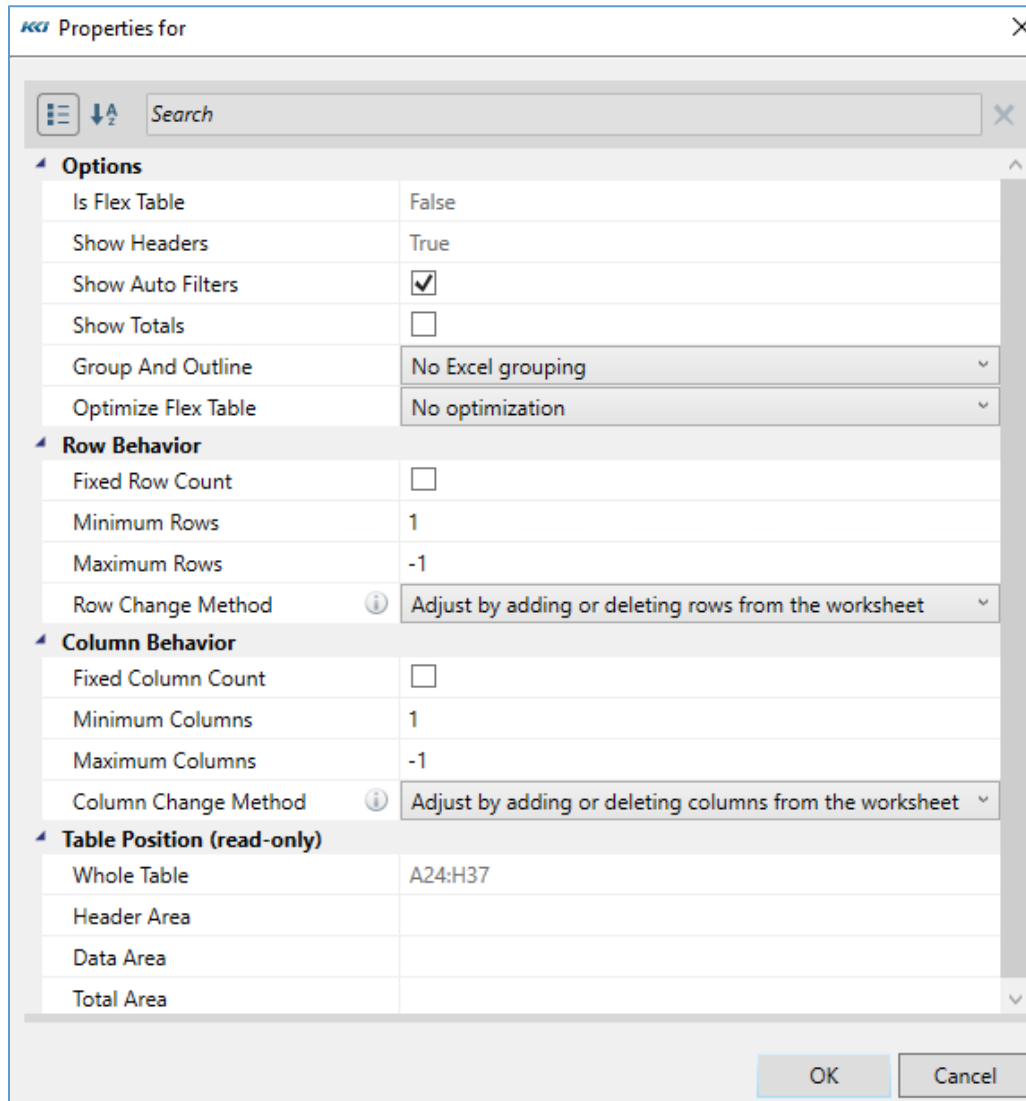


To define the properties after the flex table is created, click any cell in the flex table, click on the **CONTROL® View** ribbon, in the **Flex View** group, and click the **Flex table properties** dialog launcher.

Flex Views



The **Flex table property** dialog will open and display the options.



Note: Each flex table has its own set of properties.



Flex Views

The table below defines each of the flex table properties:

Property	Description
Is Flex Table	Indicates whether the selected table is a flex table or purely an Excel table (Not updateable).
Show Headers	Indicates whether the table has a header row – must be TRUE for flex tables.
Show Auto Filters	Determines whether Excel filters are added to each column of the table. TRUE by default when creating a flex table, but may be turned off if not desired.
Show Totals	Adds a total row below the table that automatically sums the values in the columns. FALSE by default when creating a table, but may be turned on if desired.
Group and Outline	<p>Enables the grouping structure for rows, columns, or both. The default is not to group and outline.</p> <p>This option is particularly useful for computational <i>views</i> where the row or column meta-data has a hierarchical relationship two or more levels deep.</p> <p>Data in the flex table or in the entire flex <i>view</i> can have grouping and outlining defined directly by the designer – this option governs the automatic definition for flex tables.</p>
Optimize Flex Table	This is automatically set based on whether the content of the table is updatable or not. The option for updatable data is No optimization , and Only track header and 1st data row for read only tables. If the flex table contains read-only data, CONTROL [®] can populate the table by tracking the flex formulas in just the table header and first data row.
Fixed Row Count	If selected, the rows will not grow or shrink in the flex table when the underlying information changes. You can control the number of rows to display with the Minimum Rows and Maximum Rows properties.



Flex Views

Any excess rows will display as blanks.

If the row count is less than the actual number of rows of data, you can edit the visible rows, but you cannot delete, or reorganize, source or object data.

Minimum Rows The minimum number of data rows to display.

Maximum Rows The maximum number of data rows to display. An entry of -1 specifies no maximum.

Row Change Method If your flex table can grow or shrink in number of rows, you have a choice as to whether a row will be inserted/deleted from the table or from the worksheet.

This option is relevant when you have other information on your worksheet outside of the table.

If there is other data (outside of the flex table) to the left or right of the table that you do not want impacted by the table size change, select **Adjust by adding or deleting rows from the table**.

If you do not have data to the left or right but you do have data below that you want shifted up or down, select **Adjust by adding or deleting rows from the worksheet**.

Fixed Column Count If selected, the columns will not grow or shrink in the flex table when the underlying information changes. You can control the number of columns to display with the **Minimum Columns** and **Maximum Columns** properties.

Any excess columns will display as blanks.

The data will truncate if there are fewer columns in the table than in the underlying *view* or list.

For source data, object *views*, and lists, you should generally select this check box.



Flex Views

Do not select the check box for computational *views* where the meta-data on the columns may vary.

Note: Unlike row addition, Excel does not always apply desired formatting to newly added columns, so you may need to format the table manually if the column count increases or decreases.

Minimum Columns The minimum number of columns to display. Must be greater or equal to one.

Maximum Columns The maximum number of columns to display. An entry of -1 specifies no maximum.

Column Count The minimum number of columns in the flex table, regardless of the number of columns in the underlying *view*. The table must always have one data column. For fixed column count, the number of columns.

Column Change Method If your flex table can grow or shrink in number of columns, you have a choice as to whether a row will be inserted/deleted from the table or from the worksheet.

This option is relevant when you have other information on your worksheet outside of the table.

If there is other data (outside of the flex table) above or below the table, which you do not want impacted by table size changes, select **Adjust by adding or deleting columns from the table**.

If you do not have data above or below but you do have data to the right that you want shifted right or left, select **Adjust by adding or deleting columns from the worksheet**.

Whole Table Defines the range of the entire table.

Header Area Defines the range of the table's header.

Data Area Defines the range of the table's data.



Flex Views

Total Area	Defines the range of the table's total row, if you select Show Totals .
-------------------	--

Limitations and design considerations

Flex tables offer a compromise between the automatic adjustment and navigability of standard CONTROL® *views* and the total flexibility of design that flex *views* offer.

The availability of Branch Relative and Page Relative reference types give you the ability to accommodate changes in meta-data; for example, new accounts, cost centers, or products, without having to define flex tables. However, flex tables let you go further to adjust the size of rectangular areas as meta-data changes, or the user selects different pages.

Here are some things to consider when deciding to use a flex table:

- Because flex tables are Excel tables, they have only a single header row, so multiple nested column branches must be displayed in a single cell.
- Flex tables can take advantage of a number of useful Excel table features:
 - Auto-Filtering
 - Sorting
 - Table formulas
 - Automatic formatting when rows and columns are added
 - Table styles
- You cannot insert extra rows for calculations within a flex table, although you can add extra columns.
- The flex formulas within a flex table can only refer to a single view specifier.
- You should not rely on absolute cell addresses within a flex table to reference the same underlying CONTROL® data.

Source data and object data in flex views

What is special about source and object data?

There is a fundamental difference between a computational *view* and a source data or object *view*:

- In a computational *view*, once you have specified the *view* and the values for all the dimensional arguments of a Data flex formula, you have **uniquely** identified the data element in CONTROL® for that cell.
- In a source data or object *view*, even if there are multiple *dimensions*, specifying all the arguments **may not uniquely** identify a single piece of data. For example, if you ask for



Flex Views

all of the accounts payable transactions for a cost center, account, and month, there may be only one record, or there may be several or none at all.

In a non-flex view, all the records for the selected *dimensions* are presented in tabular format on the worksheet.

For flex views, there are some special features that let you get more specific about the data you wish to see or change.

Row Numbers

In order to resolve which record's data to display or update on a source data or object view, the Data and DataReadOnly functions have one additional argument, after the View Specifier and *dimension* specifiers, called **Row Number**:

The screenshot shows the Microsoft Excel interface with the CONTROL® Navigator ribbon. The active view is 'View - Employee Info - Flex View'. The worksheet displays a table with the following data:

EmployeeID	LastName	FirstName	DeptID	State	Position	Phone
10001	Asplund	Scott	991192	UT	Director, Client Support	714-941-0603
10002	Song	Dave	991192	CA	Sr Application Consultant	310-921-6222
10003	Lu	Kai	991192	CA	Director, Consulting	310-921-6222

The View Specifier pane on the right shows the 'Row Number' field set to '<1>'. A red arrow points from this field to the first row of data in the table.

This argument is a sequential value from **1** to the number of available records (for the specified *dimensions*). If the specified row number exceeds the available number, the cell will be empty.

Key pseudo-dimensions

CONTROL®'s *datasource* columns have an **Is Key** property that indicates the column value is a useful way of identifying the contents of the record. A *datasource* may have zero or more key columns.

If a key column or set of key columns uniquely identifies a record, they are called the primary key for the table. For example, in the employee table above, the employee ID could be specified as the **Primary Key** column.



Flex Views

If the *datasource* has key columns, the **Data** and **DataReadOnly** flex functions have an additional argument for each of those columns:

	A	B	C	D	E	F	G	H	I	J
1	Filters:	sd	1							
2	Pages:									
3										
4										
5										
6										
7										
8		EmployeeID	LastName	FirstName	DeptID	State	Position	Phone		
9	*All*	10001	Asplund	Scott	991192	UT	Director, Client Support	714-941-0603		
10		10002	Song	Dave	991192	CA	Sr Application Consultant	310-921-6222		
11		10003	Lu	Kai	991192	CA	Director, Consulting	310-921-6222		
12										
13										
14										
15										
16										
17										

The identifier for the pseudo-dimension is **Key(IDColumnName)**.

For a *datasource* with Primary Key columns, you can access a record and field by setting the values of the keys. If the keys uniquely identify a record, the row number argument is irrelevant.

Object Views

Object views contain multi-record, tabular data like source data views, with the following differences:

- Object data cannot have *dimensions* other than the implicit property *dimension* on the columns, whereas source data *models* can have multiple *dimensions* like Computational *models*.
- Key and primary key columns are based on the underlying object and the properties selected in the *view*. The example below demonstrates a recommended practice for *hierarchy* data of including the ID property and all reporting relationships in the object *view*:

Flex Views

The screenshot displays a spreadsheet with an employee hierarchy table and a dynamic view configuration dialog. The table has columns for EMP_ID, EMP_Name, DEPT_ID, DEPT_Name, LE_ID, and LE_Name. The dialog box, titled 'View - Prototype object view (Dynamic)', shows the formula `Data(*, PropertyName c[0], Key(EXMP_EMP_ID) r[0])` in cell B7. The 'View Relative' reference type is selected, and a table maps 'Current Row' to 'Key(EXMP_EMP_ID) r[0]'. The dialog also includes options for 'Is Flex', 'Is CRF', 'Show Flex', and 'Immediate update'.

Notice the use of the **View Relative** reference type **Key(EXMP_EMP_ID) r[0]** to select the ID of the member on the current row.

Updating considerations

If you want to manage the contents of an object or a *datasource* by adding, deleting and reordering its records, you must use a flex table, and the underlying *view* must contain all the key columns. You may not add a record unless you supply the values for all the keys.

If you only need to update the fields of existing records or add new records without the need to delete or reorganize, you do not need to define a flex table and you can organize your *view* any way you like. You may not add a record unless you supply values for all the keys.

If any source data or object *view* is referenced by cells within and outside of a flex table, all cells outside of the flex table will be read only.

As with the underlying source data or object *views*, if there are any Primary Key columns, you will get an error message and you will not be able to file data for multiple records that have the same Primary Key values.

Note If you have a flex *view* that references a *datasource* that has no Primary Key or Selection Key columns, and you later change the *datasource* to include keys, the flex *view* will report **#InvalidKey** errors in cells that use either Data or DataReadOnly flex functions. You may run into similar problems if you remove key definitions from the *datasource* as well. Changing the *datasource* does not reorganize the flex formulas.



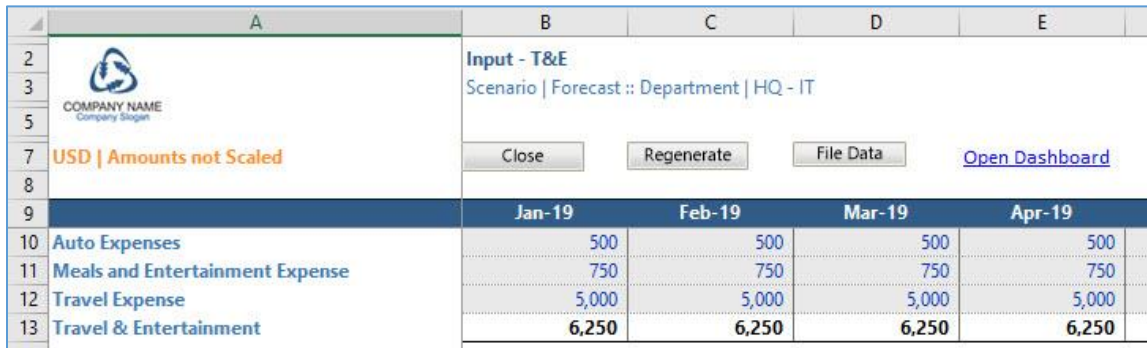
Flex Views

Customizing your flex view

Because of their ease and robust functionality, flex views will begin to replace the requirement to combine *views*, *forms*, and *sheets* for reporting or to run processes. The following new features help to accomplish this goal.

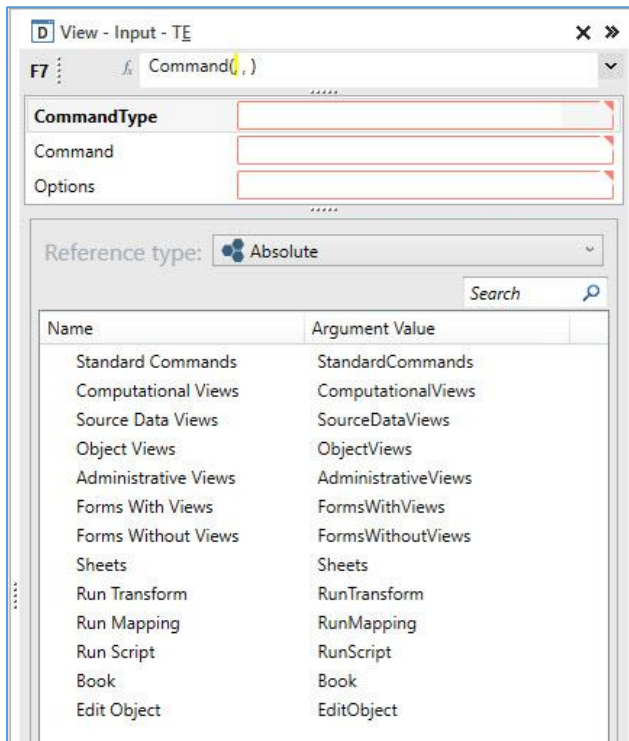
Adding Commands

The **Command** flex function allows you to place hyperlinks or buttons on your flex view that perform common operations – similar to Form or Sheet command buttons.



	Jan-19	Feb-19	Mar-19	Apr-19
Auto Expenses	500	500	500	500
Meals and Entertainment Expense	750	750	750	750
Travel Expense	5,000	5,000	5,000	5,000
Travel & Entertainment	6,250	6,250	6,250	6,250

The **Command** flex function takes three arguments: **Command Type**, **Command**, and **Options**.



Name	Argument Value
Standard Commands	StandardCommands
Computational Views	ComputationalViews
Source Data Views	SourceDataViews
Object Views	ObjectViews
Administrative Views	AdministrativeViews
Forms With Views	FormsWithViews
Forms Without Views	FormsWithoutViews
Sheets	Sheets
Run Transform	RunTransform
Run Mapping	RunMapping
Run Script	RunScript
Book	Book
Edit Object	EditObject



Flex Views

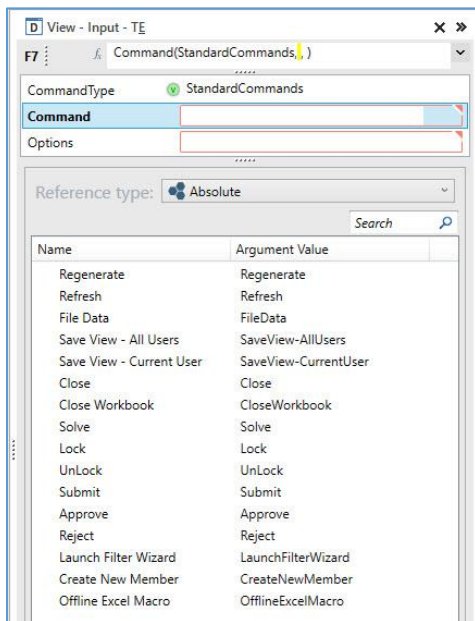
Command Type

The **Command Types** fall into three general categories:

Command Type	Description
Standard Commands	Common operations, typically on the open <i>view</i> . Most of these functions are available on the ribbon; however, for simplicity and clarity for casual users, you can install the commands on the flex <i>view</i> .
Opening another CONTROL® Object	Opens a specified <i>view, form, sheet, book, or edit book</i> . Use these options to make it easy for users to access related or supporting information
Run processes	Runs a predefined <i>script, transform, or mapping</i> . Use these options to give end-users the ability to import or export data, load or manipulate the data in their <i>view</i> , publish data, or perform other complex functions.

Standard Commands

The **Standard Commands** are self-explanatory. Some require additional information defined in the **Options** argument.



Flex Views

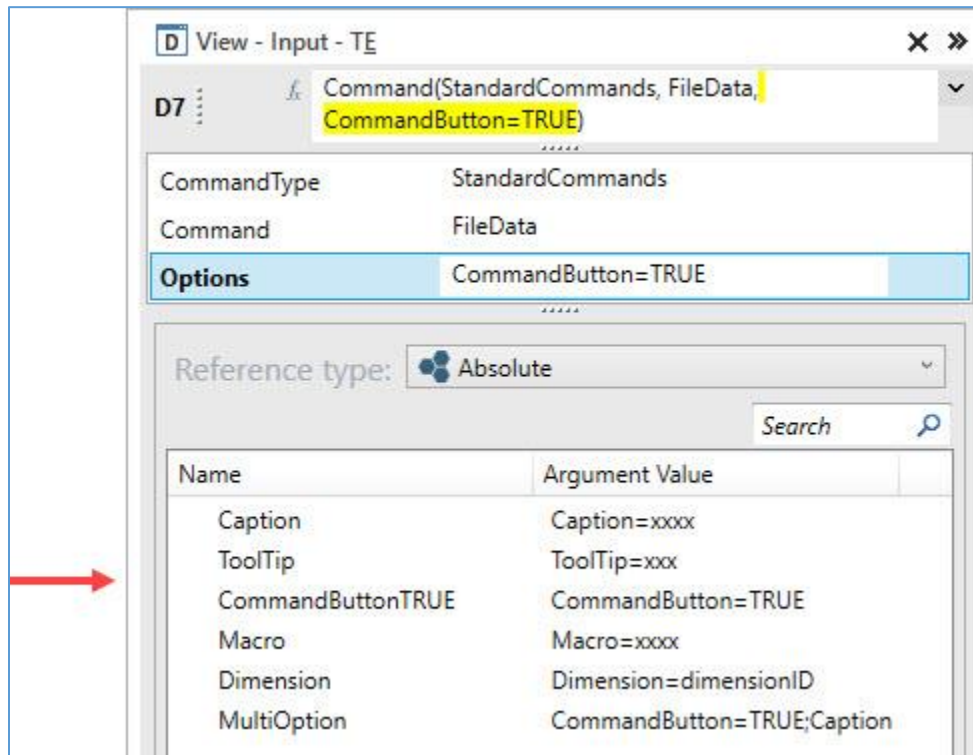
Options

The **Options** argument is a text string that contains one or more options, separated by a semi-colon. Each option is of the form:

OptionName=OptionValue

If you leave the argument blank, CONTROL® supplies an appropriate default value.

The available options display in the bottom pane:



Option

Description

CommandButton If TRUE, the cell contains a button, otherwise it will contain a hyperlink.

For most commands, the default is FALSE.

For commands that reference a selection in the worksheet (Launch Filter Wizard, Create New Member, Lock, Unlock and any run command, the default is TRUE.

Flex Views

Caption	<p>The text string that appears on the button or the value of the hyperlink.</p> <p>If not supplied, CONTROL® creates a readable caption such as “Open Sales Report”.</p>
ToolTip	<p>The text string that appears on the button or the value of the hyperlink.</p>
Macro	<p>For the Offline Excel Macro standard command, the name of the macro to execute (Required).</p>
Dimension	<p>For the Launch Filter Wizard and Create New Member standard commands, required to supply the ID of the <i>dimension</i>.</p> <p>You can also specify Dimension=Selection to have CONTROL® determine the <i>dimension</i> from the selected cell(s) on the worksheet.</p>

Note: The default width for the command buttons is 100 pixels. The button height defaults to the row height; however, you can manually adjust both the button width and height from the Excel **Developer** tab using either the **Design Mode** or **Properties** command. For information on how to override the default button width, refer to the **Keywords** section and the **KCI_FlexButtonWidth** keyword.

List Selection

Similar to a Form, use a List Selection to select the contents of a flex *view*, or to define a *keyword* used in a scripted process.

List selection flex function

With the **ListSelection** flex function, you can easily access all sorts of useful meta-data lists in CONTROL®. When a user clicks on a cell defined with a **ListSelection** flex function, a dialog will display all the members defined in the list.

Here is an example using the flex *view*'s **Department** dimension-branch, where the List Selection displays as a hyperlink.



Flex Views

The screenshot shows a financial statement grid with columns for months from Jan-19 to Aug-19. Cell B1 is highlighted with a red box and contains the text "Total Company". A dialog box titled "View - FS - Basic - Act - Flex" is open over cell B1. The dialog shows the formula for cell B1: `ListSelection(DimSpec(EXMP_DEPT), IDandName, TOT_CO, Name, SelectPage=TRUE)`. The "Source" is set to "DimSpec(EXMP_DEPT)". Below the formula, there are fields for "Filter", "Display Option", "Selection", "Property", "Keyword", and "Options". The "Reference type" is set to "Absolute" and the "Source type" is "ViewDimensionBranch". A table lists the arguments for the function:

Name	Argument Value
SCENARIO	DimSpec(SCENARIO)
EXMP_DEPT	DimSpec(EXMP_DEPT)
EXMP_ADJ	DimSpec(EXMP_ADJ)
EXMP_FS_ACCT	DimSpec(EXMP_FS_ACCT)
TIMPERIOD	DimSpec(TIMPERIOD)

At the bottom of the dialog, there are buttons for "Edit value", "Resolve", "Apply to cell B1", "Is Flex" (checked), "Is CRF", "Show Flex", and "Update".

When you click on cell **B1**, the following dialog opens, similar to the **Select Page** command:

The "List Selection" dialog box is shown with a search bar and a list of items. The list includes various departments and their sub-categories, with "TOT_CO -Total Company" selected at the bottom. The list items are:

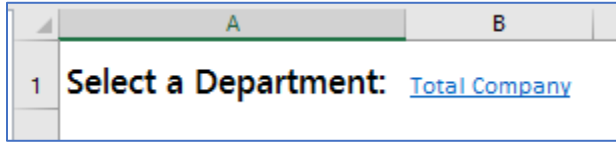
- 110112 LA - Warehouse
- 110121 LA - Manufacturing
- 110122 LA - Operations
- LE01 -LE01 US - LA - Sub
- 120210 SF - Admin
- 120211 SF - Sales
- 120212 SF - Warehouse
- LE02 -LE02 US - SF - Sub
- 130310 NY - Admin
- 130311 NY - Sales
- 130312 NY - Warehouse
- LE03 -LE03 US NY - Sub
- 991191 HQ - Executive Staff
- 991192 HQ - IT
- 991193 HQ - HR
- 991194 HQ - Admin
- 991195 HQ - R&D
- 991196 HQ - Accounting
- 991197 HQ - Finance
- 1LE11 -LE11 US HQ - Sub
- USA -USA United States
- TOT_CO -Total Company**

Buttons for "OK", "Apply", and "Cancel" are at the bottom.



Flex Views

When you make a selection and click **OK** or **Apply**, the property specified in the flex function (**Name** in this case) displays in cell **B1**.



If other flex functions or Excel formulas reference that value, the contents of the worksheet will refresh or regenerate. In the example below, the **Data** flex function references the value in cell **B1**.

	Jan-19	Feb-19	Mar-19	Apr-19	May-19	Jun-19	Jul-19
Full Time Equivalent	13	13	13	13	13	14	
Total Employee Compensation	526,203	528,847	531,504	534,175	536,859	540,621	542,000
Travel & Entertainment							
Supplies & Services	25,330	25,457	25,585	25,714	25,843	26,024	26,153
Communications & IT	10,856	10,910	10,965	11,020	11,075	11,153	11,208
Freight & Shipping	4,825	4,849	4,873	4,898	4,922	4,957	4,981
Occupancy	192,989	193,958	194,933	195,913	196,897	198,277	198,277
Maintenance	56,449	56,733	57,018	57,304	57,592	57,996	58,000
Insurance	69,476	69,825	70,176	70,529	70,883	71,380	71,380
Utilities	72,853	73,219	73,587	73,957	74,329	74,849	75,150
Depreciation & Amortization	61,756	62,067	62,379	62,692	63,007	63,449	63,449
Other Expenses							
Total Overhead Expenses	1,020,736	1,025,865	1,031,020	1,036,201	1,041,408	1,048,705	1,051,000

List selection features

Here are several important features of this function that can help in constructing rich flex views:

1. The **Source** argument option for **ViewDimensionBranch** includes access to the member lists of all the *views* referenced in the flex *view* – not just the primary *view*.
2. The **Filter** argument is not required, but you can specify:
 - A predefined *filter* for the *level* or *dimension*
 - A **Worksheet** reference to a cell containing a predefined *filter* ID or an ad hoc *filter*
 - An **Indirect** reference to a cell with a **List** or **ListSelection** flex function
 - An ad hoc *filter* enclosed in double quotes – e.g. "Region WEST"
 - Several additional *filter* options that limit the member list:
 - a. Children(memberID) – Immediate children of the member
 - b. Children(memberID, levelID) – Children of the member on a *level*
 - c. Children(memberID, All) – Children of the member on all *levels* in the branch
 - d. Level(levelID) – All members of the *level* in the branch



Flex Views

- e. For convenience:
 - i. memberID is equivalent to Children(memberID, All)
 - ii. levelID is equivalent to Level(levelID)

Note: The **Filter** argument for the **List Selection** does not filter the members in the **Select Page** dialog (launched from the CONTROL® Navigator ribbon).

3. The **Display Option** defines how the members will appear in the **List Selection** dialog.
4. The **Selection** argument specifies the initially selected member, which will change as the user selects different members from the list.
5. When the flex *view* opens, if the **Selection** argument is blank, then the value is set to the **Default Value** argument. If both arguments are blank, then the value is set to the first member of the list.
6. For the **All User** scope in a flex *view*, the **Selection** argument defaults to blank if there is a **Default Value** argument. This allows you to build flex *views* that will open with different selections for different users, if you specify a *keyword* in the **Default Value** argument.
7. The **Property** argument includes several options that defines how the members display.
8. You can define the **Keyword** argument with a *keyword*. The *keyword* will update with the values you select in the **List Selection** dialog after the flex *view* refreshes. Users can make a selection and have it update data in the underlying *view* if one of the *dimension filters* includes it, or to run a command that includes the *keyword*.
9. If you want to have the user choose from a set of unique values in a column of a *datasource*, the source argument has the form:

DataSource(dataSourceID, idColumn [, nameColumn])

The **nameColumn** is optional, and if specified will appear to the right of the **idColumn** values in the **List Selection** dialog.

10. The **Options** argument can specify zero or more special behaviors (separated by a semi-colon) for the List Selection:

Option Name	Description
-------------	-------------



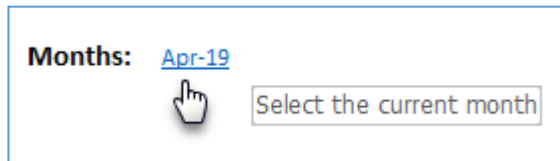
Flex Views

Tooltip

Specifies a tooltip that displays when the cursor is over the hyperlink of the cell containing the **ListSelection** function.

Use a tooltip to provide the user with a description of what the selection does.

Example: Tooltip=Select the current month

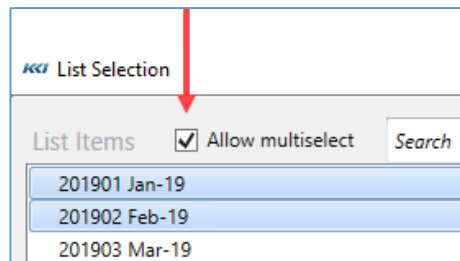


MultiSelectTRUE

Enables a check box in the **List Selection** dialog. If you select the check box, then you can select multiple members.

Use this option to set a *keyword* used in a *filter* with multiple selected members.

Example: MultiSelect=TRUE



SelectPageTRUE

Adds the List Selection to the **Select Page**, **Expand Multipage**, and **Print** dialogs launched from the CONTROL[®] Navigator and View ribbons.

Note: Only use this option if the List Selection does not reference a *dimension* on the *view's Pages* edge. If it does, then you will see the *dimension* twice in the dialogs, one for the *dimension* in the *view*, and another one for the List Selection.

Example: SelectPage=TRUE

Flex Views

SelectPageSynchronize

For List Selections that use one of the primary *view's* page dimension-branches as its source – causes page selection on the dimension-branch to set the List Selection member and vice-versa (except when the **Page Selection** dialog includes members that are not included in the **List Selection filter**).

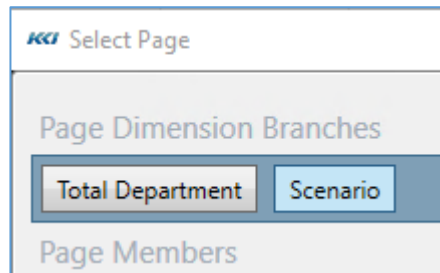
This option is identical to **SelectPageTRUE** in all other respects.

Example: SelectPage=Synchronize

Caption

This defines the caption of the List Selection button displayed on the **Select Page** dialog. This is used in conjunction with the **SelectPageTRUE** option.

Example: SelectPage=TRUE;Caption="Scenario" (double quotes are required)



MultiOption

This will automatically combine the **SelectPageTrue** and **Caption** options. You can use the Ctrl key and left mouse button to select other options as well. They need to be separated by a semi-colon and the entire string encapsulated by double quotes.

Example:

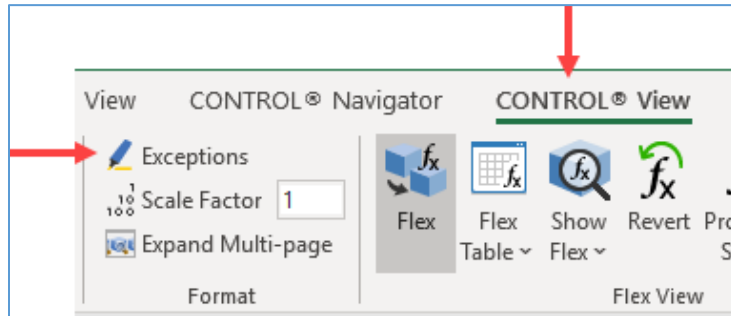
"MultiSelect=TRUE;SelectPage=TRUE;Caption="Scenario""

Flex Views

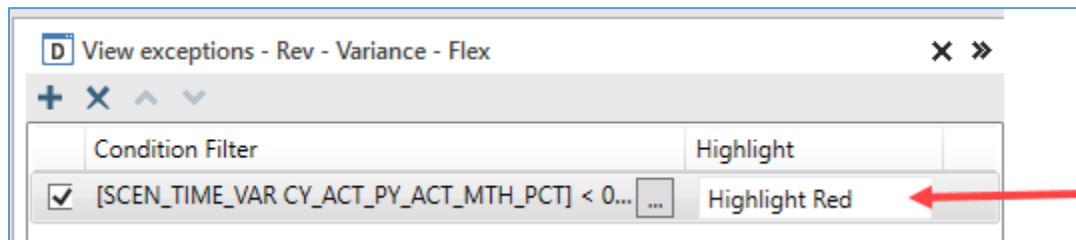
Dynamically formatting flex views

Exceptions

Like other *views*, you can create **Exceptions** for flex views as well. On the **CONTROL® View** ribbon, in the **Format** group, click **Exceptions** to open the **View Exceptions** task pane.



As part of defining a condition, you select a **Highlight** option, for example, **Highlight Red** or **Highlight Green**.

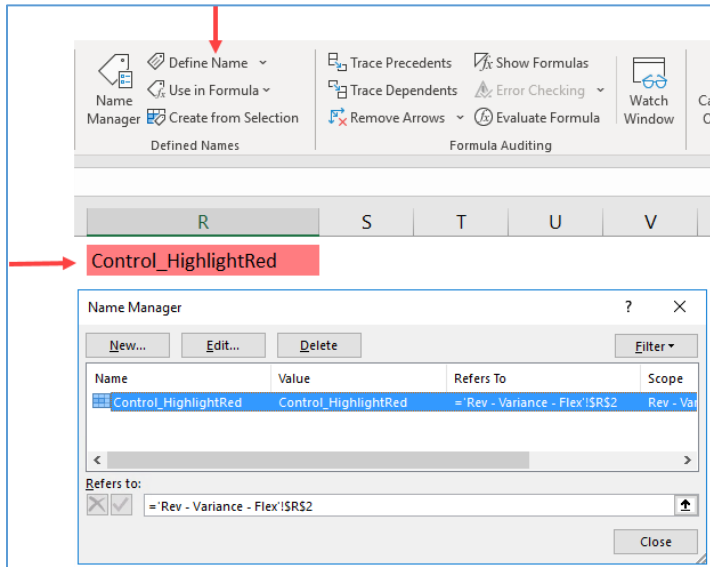


However, in flex *views*, you need to define each **Highlight** using an Excel defined name.

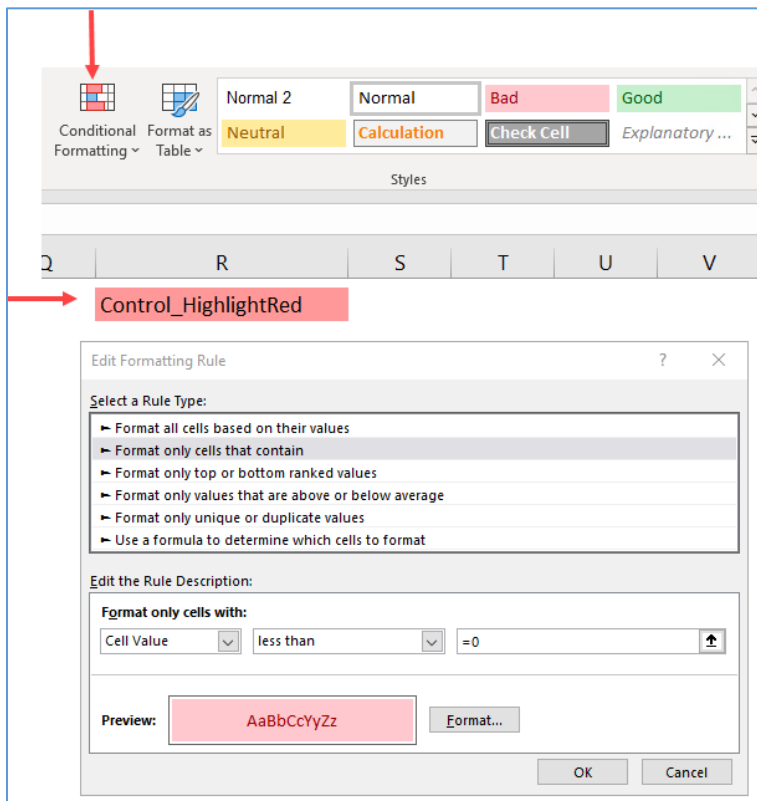
Select any single cell on the flex *view* worksheet and create a defined name called "**Control_**" plus the highlight option. Because Excel defined names cannot have blanks, omit blanks when creating the defined name. For example, define **Highlight Red** as **Control_HighlightRed** as the Excel defined name.

Note: You need to scope the Excel defined names to the worksheet.

Flex Views



After you have created the defined name, you can apply one or more Excel conditional formats to that cell, that will apply a background color, border, or font color and style to the cell.



Flex Views

When you open the flex view, the system evaluates all of the exceptions and applies the conditional formatting for each **Highlight** option.

	Actual Aug-2019	Actual Aug-2018	CY Actual vs PY Actual Var	CY Actual vs PY Actual %
Units	3,890,141	3,550,355	339,787	9.6%
Price	4.91	5.04	(0.13)	-2.6%
Gross Sales	19,106,601	17,901,255	1,205,346	6.7%
Sales Discount	691,475	647,854	(43,622)	-6.7%
Sales Returns and Allowances	559,469	524,174	(35,294)	-6.7%
Shipping Charges	295,163	276,542	(18,620)	-6.7%
Other Incomes	279,359	260,124	19,235	7.4%
Net Revenue	17,839,853	16,712,809	(1,127,044)	-6.7%

If an exception ceases to be applicable, then the conditional formats do not apply and the cells previous formatting returns.

Other dynamic formatting options


Here are some additional Excel defined names that will format data dynamically in flex views.

Excel Defined Name	Definition
Control_Updateable	Format the updateable cells in the flex view.
Control_Solved	Format cells that were altered by a solve operation.
Control_PermanentlyLocked	Format cells that have been permanently locked from being changed by a solve operation.
Control_Locked	Format cells that have been temporarily locked from being changed by a solve operation.
Control_Transformed	Format cells that changed by a <i>transform</i> operation.
Control_ListSelection	Format cells that contain a ListSelection flex function.



Flex Views

Just like **View Exceptions**, the Excel defined names can be placed anywhere on the flex view's worksheet. Use Excel's conditional formatting on the cell with the defined name to apply the format automatically to the data cells in the flex view that meet the definition, for example, updateable, locked, solved, etc.

	A	B	C	D	N
2	 COMPANY NAME Company Slogan	EXMP - Expense Exp - Input Budget - Flex			Control_Updateable
3		Scenario Budget			
5		Department Osaka - Sales			
7		NOTTRANSLATION Amounts not Scaled			
8					
9		Jan-19	Feb-19	Mar-19	
10	Full Time Equivalent	9	9	9	
11	Base Compensation	37,639,771	43,089,301	42,671,124	
12	Employee Other Compensation	9,409,943	10,772,325	10,667,781	
13	Payroll Tax Expense	3,387,579	3,878,037	3,840,401	
14	Employee Benefit, Health Insurance	1,693,790	1,939,019	1,920,201	
15	Employee Benefit, Pension Plans	846,895	969,509	960,100	
16	Total Employee Compensation	52,977,978	60,648,191	60,059,607	
17	Auto Expenses	1,952,458	2,235,137	2,213,446	
18	Meals and Entertainment Expense	1,288,622	1,475,191	1,460,874	
19	Travel Expense	3,904,916	4,470,275	4,426,891	
20	Travel & Entertainment	7,145,997	8,180,603	8,101,211	
21	Office Expense	780,983	894,055	885,378	

Numeric Formats

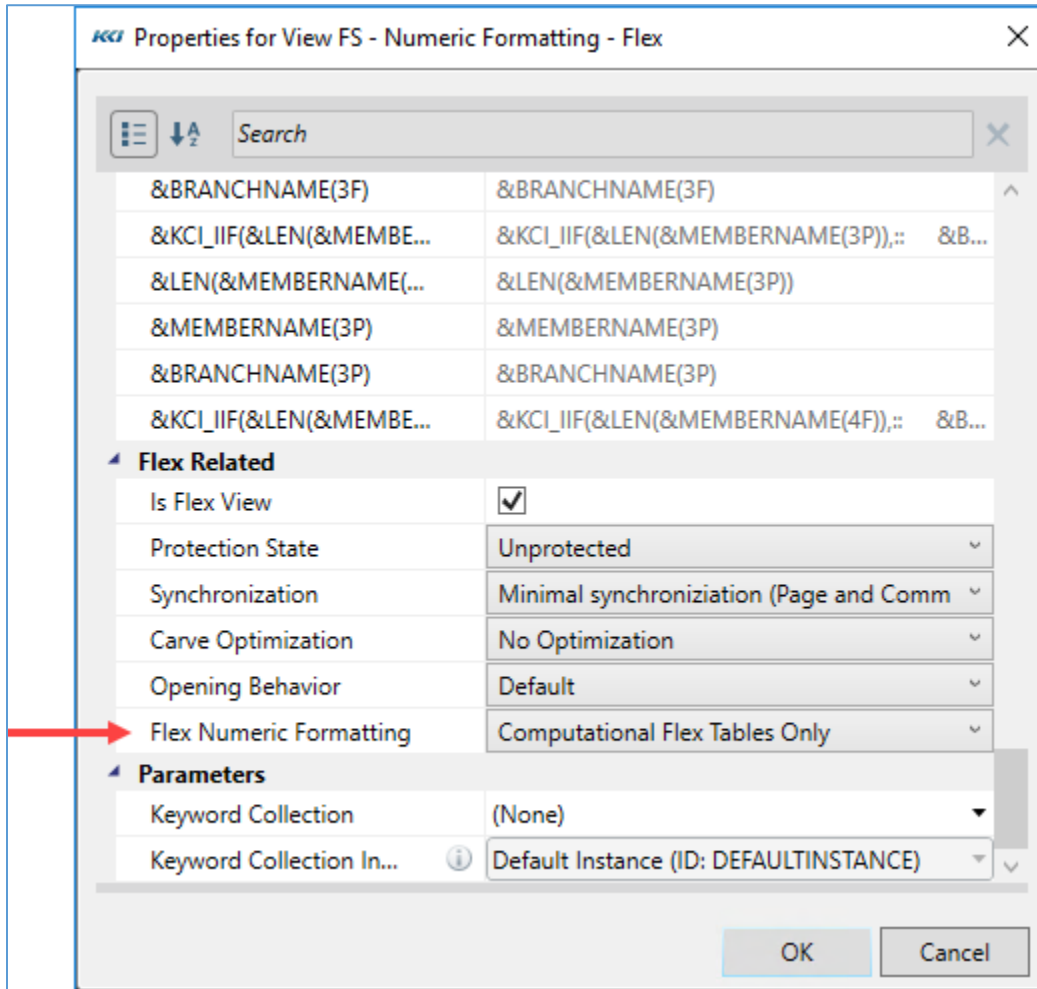
Since the goal with flex views is to provide an Excel-like user experience, the formatting of the cells on your worksheet is generally at your discretion.

However, in some cases, you may want to have the numeric formatting applied to the data cells automatically when the content of cells changes in a table or because underlying meta-data has changed. This can potentially save you a lot of manual reformatting and make your flex views more appealing.

In the **Flex Related** view property group, you can control this behavior with the **Flex Numeric Formatting** property.



Flex Views



This property determines which cells will automatically have numeric formats defined based on CONTROL® meta-data, that is, the **Numeric Format** or **Style** property of *variables*, *custom dimension* members, or *datasource* columns.

Note: This only applies to cells containing flex formulas using the **Data** or **DataReadOnly** functions.

Options	Description
Computational Flex Tables Only	Applies formatting only to flex tables containing Computational <i>view</i> data. (Default option)



Flex Views

All Computational Data	Applies formatting to all cells accessing data on Computational views.
All Data	Applies formatting to all cells accessing data for all <i>view</i> subclasses, for example, computational, source data, object, administrative, etc.
Manual Formatting	Applies no formatting.

Options for flex views

To support environments where flex *views* are created by an administrator or designer and then used by many end-users, there are three scoped *view* properties. These properties allow granular control over the *view's* behavior and performance characteristics. Casual users do not need to be concerned with these properties as the default settings work well.

Carve optimization

The underlying *views* of a flex *view* may have their *dimension filters* defined for a large range of data, for example, **YR 2018, 2019, 2020**. However, the referenced data in the flex *view* may only refer to year **2019**. The **Carve Optimization** property determines whether CONTROL® will limit the data retrieved in the underlying *views* when you open the flex *view*. The table below defines the available options.

Option	Used to
No Optimization	Retrieve all data specified in the underlying <i>views</i> . Note: This is the default.
Minimal Retrieval for Model Data Only	Retrieve all data in the underlying <i>views</i> , except for Model view specifiers that only retrieve the referenced data. Note: This becomes the default if you add a flex function that includes a Model <i>view</i> specifier.



Flex Views

Minimal Retrieval Retrieve only the referenced data for the underlying *views*. This will yield the most efficient generation time for the *views*.

Note: Do not use this option for updatable *views* because it will affect updatability and solve functionality. In addition, *views* using this option will not share data with other flex *views* or CRFs.

Synchronization

When your flex *view* references data from more than one *view* or *model*, you can synchronize the *view* options and common *dimension filters* and branches between the underlying primary and subsidiary *views*. The table below defines each option.

Option	Used to
Minimal synchronization (Page and Comments)	Synchronize the primary <i>view's</i> page selection with common <i>dimensions</i> on the subsidiary <i>view's</i> page edge, as well as, cell comments/hyperlinks of common <i>dimensions</i> . Note: This is the default.
Display synchronization (+Scaling and Currency)	Synchronize the primary <i>view's</i> scaling, currency, and member display (ID, Name, Both) <i>view</i> options with the subsidiary <i>views</i> , in addition to the Minimal synchronization options.
Basic Filters (+Filter Edge)	Synchronize the <i>filters</i> and branches of the <i>dimensions</i> on the primary <i>view's</i> filter edge with common <i>dimensions</i> in the subsidiary <i>views</i> , in addition to the Display synchronization options.
Advanced Filters (+Page Edge)	Synchronize the <i>filters</i> and branches of the <i>dimensions</i> on the primary <i>view's</i> filter and page edges with common <i>dimensions</i> in the subsidiary <i>views</i> , in addition to the Display synchronization options.
All Filters (+All Edges)	Synchronize the <i>filters</i> and branches of all <i>dimensions</i> in the primary <i>view</i> with common <i>dimensions</i> in the subsidiary <i>views</i> , in addition to the Display synchronization options.



Flex Views

This property is not relevant if:

- The flex *view* only contains data from its primary *view*.
- Users do not expect to change any display or *filter* options.
- Users work with the flex *view* offline.

Note: Display and *filter* synchronization can invalidate large sections of a flex *view* if you use absolute references.

Protection state

The **Protection State** property of a flex *view* limits the changes that an end user can make to the *view* while it is open (similar in principle to Excel's protection options). The table below defines each option.

Option	Used to
Unprotected	Grant full access to modify, edit, cut, copy and paste values, Excel formulas, and flex formulas.
No Filtering	Restrict changing underlying <i>view filters</i> .
Data Entry	Grant access to enter data and comments, but restrict changes to flex formulas.
Locked	Restrict all data and definition changes.

Notes:

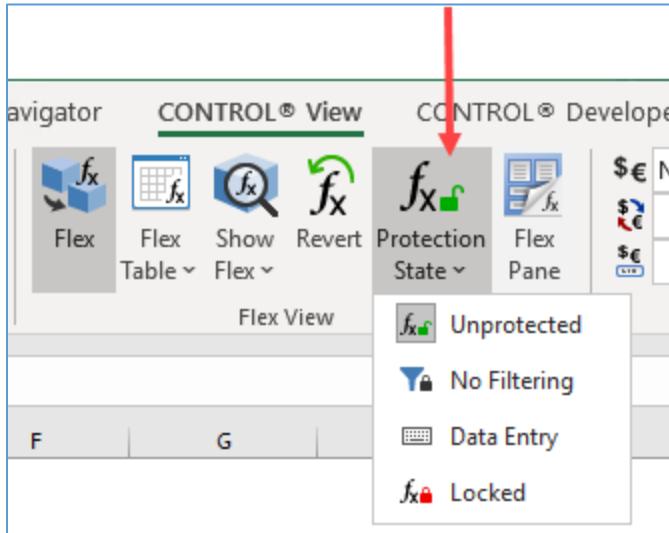
- The protection state options apply to the interaction with CONTROL® content. You can use Excel worksheet protection to govern cells with and without CONTROL® content.
- Pressing the delete key will delete the cell's formula, value, and flex formula, when the property is set to either **Unprotected** or **No Filtering**. Removing a cell's Excel formula will also delete the cell's flex formula.
- The protection state adds restrictions to existing data and object access privileges, in addition to the **Read Only** property of the underlying *view*. That is, if you enable the



Flex Views

view's read only property, then you cannot change any data regardless of the flex view's protection state.

If you have the required security privileges, you can update the **Protection State** property from the **CONTROL® View** ribbon.



Offline usage

Like scratchpads, you can save flex views as an Excel workbook and use them offline. This is useful when distributing reports for analysis or collecting budgets or forecasts from users who do not have a CONTROL® license.

When used offline, flex views are subject to the following limitations:

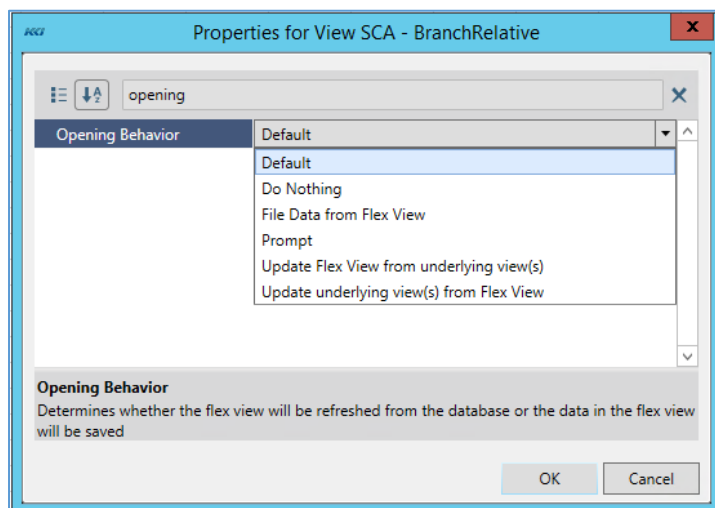
- You may not see or edit the flex formulas.
- The Flex Pane is not available.
- Page selection is not available.
- Excel sorting is not allowed.

Opening behavior

If you take a flex view offline and then reopen it when you are logged into CONTROL®, the **Opening Behavior** property determines how to handle data changes in the flex view. The table below defines the available options.

Flex Views

Option	Used to
Default	Update the data based on the Read Only property of the underlying <i>view</i> . If the primary <i>view</i> is defined as read only, then update the flex <i>view</i> with the data from the underlying <i>view(s)</i> . If the primary <i>view</i> is not defined as read only, then update the underlying <i>view(s)</i> with the data in the flex <i>view</i> .
Do Nothing	Open the flex <i>view</i> and do not update the data in the underlying <i>view</i> .
File Data from Flex View	Open the flex <i>view</i> and file the data from the flex <i>view</i> into the application automatically.
Prompt	Prompt the user what they want to do with the data in the flex <i>view</i> .
Update Flex View from underlying view(s)	Update the data in the flex <i>view</i> with the data in the underlying <i>view(s)</i> .
Update underlying view(s) from Flex View	Update the data in the underlying <i>view(s)</i> with the data in the flex <i>view</i> (but don't file it).



Flex Views

Notes:

- Unlike scratchpads, the opening behavior cannot be overridden by a *keyword*.
- Even if a *view* is read only, the comments are updatable.
- The protection state of the *view* is layered on top of these rules, so protecting the flex *view* from any change will stop all updates.

Keywords in filter expressions

You may have a flex *view* that has a *filter* expression defined by a *keyword*, and you update the replacement value with a *form*, *script*, or just manually update the *keyword*'s **Replacement** property. This could cause a problem if you save the flex *view* offline because the data reflects the *keyword*'s replacement value at that time. However, when you later reopen the offline file, the *keyword* in the application is not likely set to the original definition when you saved the file. If you were to click **File Data**, you could potentially save the data to the wrong member.

To accommodate this case, you will need to add a **Keyword** flex function in your flex *view* that will set the *keyword*'s replacement value to the same value when you saved the file offline.

The **Keyword** flex function includes a new **Resolution Behavior** argument that specifies the behavior of the *keyword* resolution in the *view*. There are three options:

Resolution Behavior	Used to
Dynamic	The <i>view</i> responds to dynamic changes in the <i>keyword</i> 's replacement value via a <i>form</i> , <i>script</i> , or direct edit of the <i>keyword</i> 's Replacement property, after you regenerate the <i>view</i> . This is the default behavior.
Fixed	The <i>view</i> does not respond to dynamic changes in the <i>keyword</i> 's replacement value, but uses the value displayed in the cell. Use this option if you do not want the <i>view</i> to conform to the current replacement value of the <i>keyword</i> .
Fixed On Save	The same behavior as Dynamic , unless you save the <i>view</i> offline in an Excel workbook, or save it online using View > Save As , then it will become fixed. Use this behavior if you are creating a <i>view</i> for multiple users who have different scoped values for a <i>keyword</i> and they might save it offline, or save a personal copy, save as a new <i>view</i> , or you will publish the <i>view</i> with different <i>keyword</i> definitions. This option reduces the chance of saving the <i>view</i> offline, and then later reopening it and saving the data to an incorrect member.



Flex Views

In this case, you would set the **Resolution Behavior** argument to **Fixed On Save**.

Flex view performance

Here are some performance guidelines to consider when developing flex *views*:

- **Number of flex formulas:** As with Excel, performance directly correlates with the number of flex formulas. Although, flex formulas perform much better than CONTROL® reference functions.
- **Worksheet reference type:** Flex *view*'s that use both **Worksheet** reference types in the flex functions in addition to Excel formulas are slower than flex *views* that do not have both of these characteristics. In this case, each refresh of the flex *view* results in a second refresh because the system does not know the dependencies (if any) between the Excel and flex formulas.
- **Offline workbooks:** If you save a flex *view* offline as an Excel workbook, enable Excel's worksheet protection on the worksheets that include flex *views*. This will prevent the software from having to create named ranges for the flex *view* content, which will significantly reduce the time to save and reopen the report.

CONTROL® reference functions (CRFs)

The introduction of flex *views* comes with new functionality for CRFs and some subtle considerations about the choice between using CRFs and flex formulas for new or existing applications.

New features and functionality

The **Flex Pane** is equally useful in authoring CRFs, as it is flex functions. There is nearly a one to one equivalence between a flex functions and a CRF, although the function names differ. The list of arguments for each function are the same, but the syntax and structure of the arguments differ slightly.

Here is the **Flex Pane** for a CRF versus a flex function:



Flex Views

CRF

vs.

Flex Function

Name	Argument Value
CAN: Canada	EXMP_CNTRY CAN
ESP: Spain	EXMP_CNTRY ESP
FRA: France	EXMP_CNTRY FRA
ITA: Italy	EXMP_CNTRY ITA
JPN: Japan	EXMP_CNTRY JPN
THA: Thailand	EXMP_CNTRY THA
TWN: Taiwan, Province of C	EXMP_CNTRY TWN
USA: United States	EXMP_CNTRY USA
TOT_CO: Total Company	EXMP_CO TOT_CO

Here are a few differences for CRFs:

- The **Is Flex/Is CRF** button at the bottom of the pane is set to **Is CRF**
- Textual arguments are enclosed in quotes, so they will be treated as strings by Excel
- **View Relative** and **Indirect** reference types are not supported
- Dimension specifiers for CRFs use the syntax
levelID memberID [On dimensionID [branchID]]
- Whereas flex functions use



Flex Views

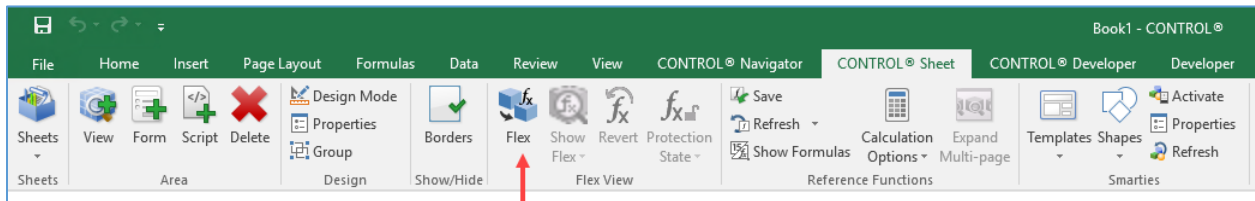
[dimensionID[(branchID)]] levelID memberID

- Not all flex functions are available as CRFs.

Converting CRF sheets to flex views

You can convert a workbook *sheet* object containing CRFs to a flex view from the **CONTROL® Sheet** ribbon:

1. Open the workbook *sheet* that includes the CRFs.
2. On the **CONTROL® Sheet** ribbon, in the **Flex View** group, click **Flex**.



The flex process will open a new worksheet and create a new flex view based on both the CONTROL® and Excel content. You can continue to customize the flex view on the new worksheet, and when you are finished, you can save the *view* using the **View > Save** or **View > Save As** commands on the **CONTROL® Navigator** ribbon.

Converting individual CRFs to flex formulas

The Flex Pane allows you to convert some or all CRFs on a worksheet to flex formulas.

You do this by following these steps:

1. Open the *sheet* or workbook containing the CRFs.
2. Open an existing flex *view*.
3. Copy and paste the CRFs from the first worksheet to the worksheet containing the flex *view*.
4. Select the range of cells on the flex *view* that contain the CRFs you want to convert.
5. Open the Flex Pane.
6. Change the button from **Is CRF** to **Is Flex**. Any cell in the selected range containing a formula beginning with **=Control...** will convert to a flex formula.
7. When you are done with your changes, save the updated flex *view*.

CRFs vs flex formulas

In the beginning of this document, we described the motivation behind flex *views*. We anticipate that flex *views* will eventually replace CRFs. However, only time and customer usage will tell.



Flex Views

There are several considerations you will want to evaluate when choosing between these two functions:

- Flex formulas can coexist with Excel formulas in the same cell whereas CRFs are the Excel formula for the cell. This is extremely useful in reporting when you need to display a different name other than the retrieved member name. In addition, when budgeting or forecasting, you can enter an Excel calculation in a cell with a flex formula in order to calculate data based on other user inputs.
- You can nest multiple CRFs as well as Excel functions in a single cell, which you cannot do with flex formulas.
- CRFs are treated as Excel formulas and are executed in a sequence determined by Excel's order of execution based on Excel formula dependencies, while flex formulas retrieve or post data for the entire worksheet pre or post the execution of Excel formulas. There may be cases where the two approaches yield differing results.
- CRFs will always share data among functions referencing the same *view* specifier whereas flex *views* will not share data if the **Carve Optimization** property is set to **Minimal Retrieval**.
- CRFs do not support **Carve Optimization** and should not reference **Model view** specifiers.
- We expect the performance profile of very large flex *views* will be significantly faster than the equivalent *sheet* with CRFs.

Keywords

There are some new system *keywords* related to flex *views*.

KCI_NoRefreshPasteSize

When you use the Excel **Paste** command in an open flex *view* the system will perform an automatic *view* refresh. If you are developing a very large flex *view* with thousands of flex formulas, the *view* refresh process could take a long time to complete. The replacement value of this *keyword* defines the maximum number of flex formulas you can have in your flex *view* before the automatic *view* refresh is disabled.

For example, if the *keyword* replacement value is set to 100, and you have 95 flex formulas in your *view*, then the *view* will refresh when you use the Paste command in the *view*. However, if you have 110 flex formulas in your *view* and you use the Paste command, the *view* will not automatically refresh, and a message will display in the Excel status bar indicating that you need to refresh the *view*.

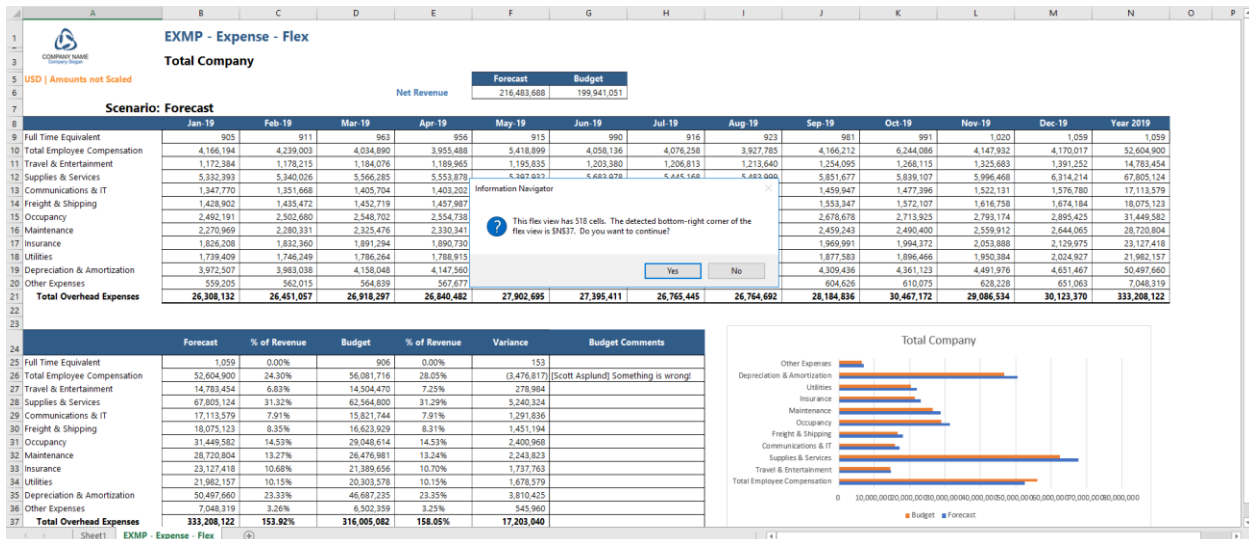


Flex Views

KCI_FlexSizeWarning

Because flex views include formulas that can reference cells anywhere in the worksheet, the system must evaluate the entire used portion of the worksheet for cells with information. This could affect the view's performance if you accidentally entered information in a cell outside of your flex view range. For example, if you were to convert an Excel worksheet to a flex view, and someone erroneously entered a value in CZ10000, then the system would have to evaluate all the cells from A1 thru CZ10000 even though the flex view only included actual information from A1 to H100.

The **KCI_FlexSizeWarning** keyword identifies the maximum number of cells allowed in the flex view before you get a warning message. In the example below, the keyword's replacement value is set to 250. When the view refreshes, a message box displays "This flex view has 518 cells. The detected bottom-right corner of the flex view is \$N\$37. Do you want to continue?"



The message is displayed since the view's range includes more cells (518) than the keyword's replacement value (250). You can either continue the view process or not. If the keyword is not present or its replacement value is non-numeric, then it will default to 100000.

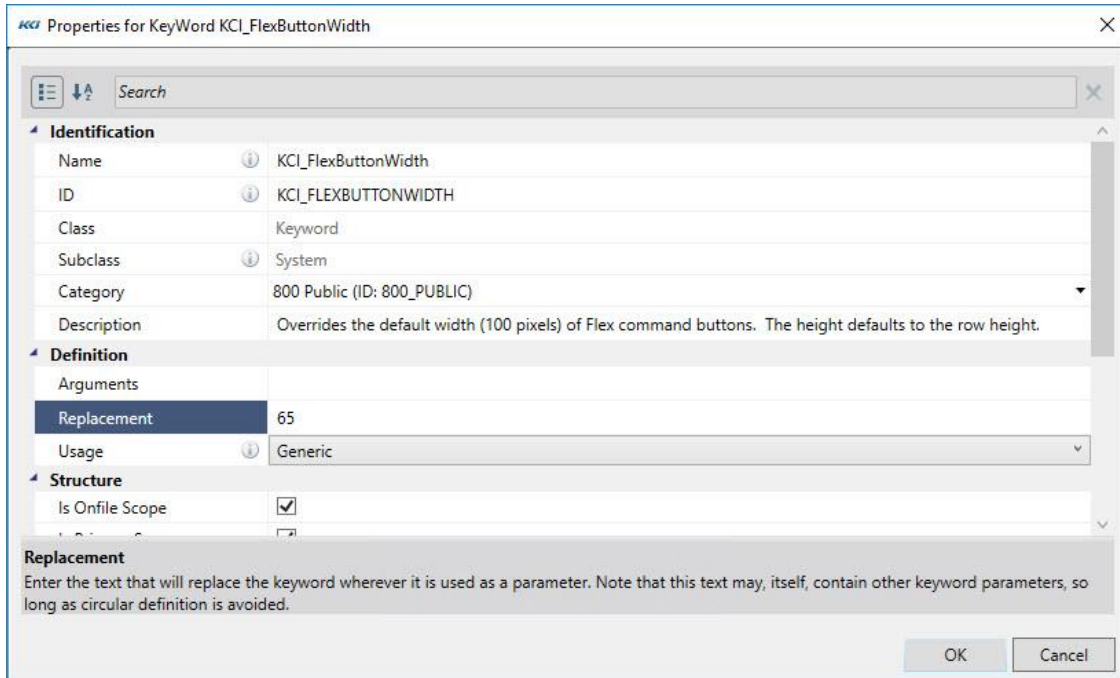
This keyword will help you discover if there are any erroneous Excel formulas or constants in your flex view that should be deleted to avoid negatively affecting the view's performance.

KCI_FlexButtonWidth

This keyword overrides the default width (100 pixels) of the command buttons that the **Command** flex function creates. Use the **Replacement** property to re-define the default number of pixels.



Flex Views



Cut, Copy, and Paste

Cut, Copy, and Paste are probably some of the most frequently used Excel functions. Performing these functions is easy and Excel provides various ways to carry out these functions:

- From the **Home** tab, in the **Clipboard** group
- From the right click shortcut menu
- Via short cut keys: Ctrl+X, Ctrl+C, Ctrl+V

Excel provides additional paste options using **Paste Special**.

Excel is a two dimensional interface, so these functions are relatively simple and straight forward whether you perform these functions on one cell, a range of cells, in a single worksheet, or across multiple worksheets.

However, with a multi-dimensional application like CONTROL®, these functions can become more challenging. For example:

- When you copy a flex formula that references a member on the Pages edge of a *view*, what do you want it to reference when you paste it to a different worksheet that does not include the same *dimension* on the Pages edge?
- When you copy a flex formula based on Relative References, and you paste it to a range on the worksheet that does not include row or column headers, what should be the behavior of the paste function?

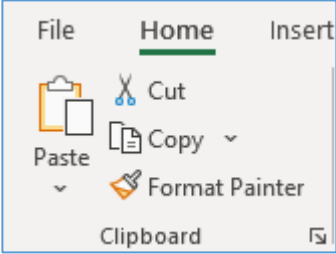
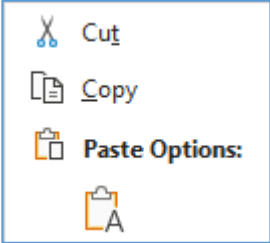


Flex Views

Because of this added complexity, CONTROL® includes additional functions and modified behaviors, in an effort to stay consistent with Excel yet keep these functions simple and predictable.

Best Practices

The following are the recommended best practices related to using Excel's **Cut**, **Copy**, and **Paste** functions in a flex view.

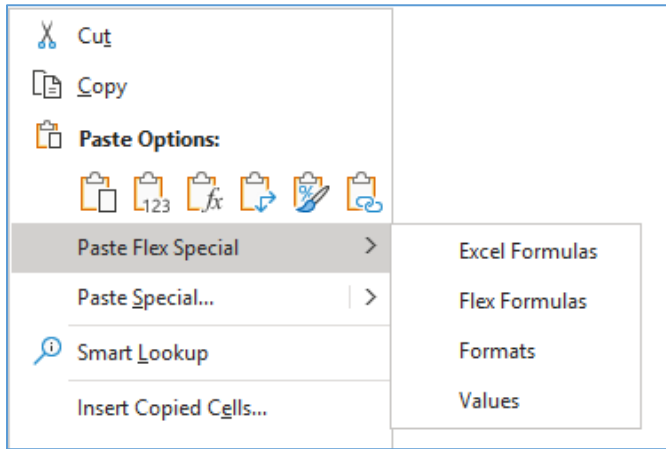
Excel options	Recommendation
Shortcut keys Ctrl+X, Ctrl+C, Ctrl+V	<i>Always</i> use Ctrl+X, Ctrl+C, Ctrl+V. These shortcuts will behave similarly to their Excel equivalents.
Home tab, Clipboard group 	Only use the default Cut , Copy , and Paste functions. These will behave similarly to their Excel equivalents.
Right-click shortcut menu 	Do not use the Cut , Copy , and Paste functions from this menu because it will not always provide the desired behavior.

Paste Flex Special

One of the special characteristics of flex views is that Excel and flex formulas can coexist within a single cell. This characteristic results in CONTROL® having its own paste special commands. You access these options from the right-click shortcut menu.



Flex Views



Paste Flex Special Function	Description
Excel Formulas	Paste only Excel formulas, do not overwrite flex formulas or Excel formatting.
Flex Formulas	Paste only flex formulas, do not overwrite Excel formulas or formatting.
Formats	Paste only Excel formatting, do not overwrite Excel or flex formulas.
Values	Paste only the cell's value, do not overwrite Excel formulas, formatting, or flex formulas.

Special cases

When you copy and paste cells containing flex formulas that include **View** or **Page Relative** reference types, and the target range does not have branch members in the respective rows, columns, or pages, then the target flex formulas will convert the missing references to an **Absolute** reference type.

In the example below, the flex formula was copied from cell **B12** to **C15**. The target cell does not include any row or column references for the account and time period arguments, so they convert to absolute references (based on their definitions in the source formula).

Flex Views

The screenshot shows a spreadsheet with a flex view for 'Full Time Equivalent' in cell C15. The value is 905. The flex pane on the right shows the formula and its arguments:

Name	Argument Value
EXMP_DEPT	EXMP_DEPT p[0]
EXMP_FS_ACCT	EXMP_FS_ACCT EXMP_FS_SUMMARY 100_FTE
TIMEPERIOD	TIMEPERIOD MTH 201901

In cases where you do not want this default behavior, but want to retain the arguments from the copied flex formula, use the shortcut key **Shift+Ctrl+V**.

The screenshot shows a spreadsheet with a flex view for 'Full Time Equivalent' in cell C15. The value is #BadIndex. The flex pane on the right shows the formula and its arguments:

Name	Argument Value
EXMP_DEPT	EXMP_DEPT p[0]
EXMP_FS_ACCT	EXMP_FS_ACCT r[0]
TIMEPERIOD	TIMEPERIOD c[0]

The target flex formula will display with a **#BadIndex** error, which will resolve after you copy and paste row and column branch references, or modify the reference type in the Flex Pane. You may want to use this option if you know that you will be copying the requisite branch references, and you do not want the arguments in the formula to be made absolute.



Flex Views

Conclusion

The goal of creating this new feature is to entice both expert and casual users to create interesting and innovative applications using the amazing flexibility of Excel in concert with the robust multi-dimensional data management capabilities of CONTROL®.

The challenge for you is to find and create those applications and take your contributions to a new level for your company.

Appendix A

Flex Functions

ASQuery

The **ASQuery** flex function is very similar to the **SQLQuery** flex function, but it queries either multi-dimensional or tabular models in Analysis Services, and the query is written in either DAX or MDX.

The screenshot shows an Excel spreadsheet with a DAX query in the formula bar and a PivotTable of results. The query is as follows:

```
MAXPTEST
// DAX Query
EVALUATE
TOPN(
    1001,
    SUMMARIZECOLUMNS(
        'AIC Family'[AIC Group Name],
        "Birds", Fact[Birds],
        "Cars", Fact[Cars],
        "Dogs", Fact[Dogs]
    ),
    [Birds],
    0,
    'AIC Family'[AIC Group Name],
    1
)
ORDER BY
[Birds] DESC, 'AIC Family'[AIC Group Name]
```

The PivotTable results are:

AIC_GROUP_NAME	BIRDS	CARS	DOGS
Adults	468115.06	5144.0106	40884.435
Children	73356.122	3943.2775	23294.5679
Common	489	489	489

Argument	Description
ASSource	The Analysis Services source. There are three Source types available: <ul style="list-style-type: none">• PowerPivot Models – any Power Pivot <i>model</i> with an Analysis Services Usage property of Exported to AS or Managed externally - uses the AS Server and AS Database associated with the PowerPivot <i>model</i>.



Flex Views

- **AS Query Datasources** - uses the **AS Server** and **AS Database** associated with the *datasource*.
- **All** – allows selection of any database on the default AS Server. To reference a database on a different AS Server, use *<ServerName.Database>*.

Query Expression

A DAX or MDX query expression.

- DAX queries begin with **EVALUATE**.
- MDX queries begin with **SELECT**

For an **AS Query Datasource**, if the query expression is blank, the unmodified *datasource* expression is used.

The **Query Expression** argument supports worksheet references, which is a convenient way to assemble, customize, and experiment with the query.

Row Number

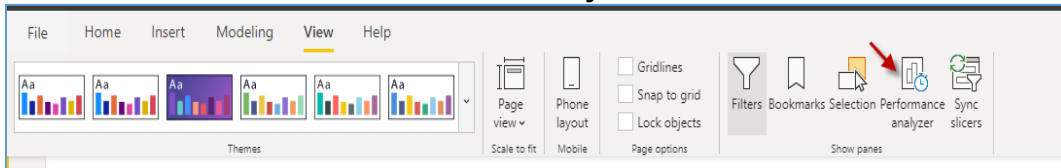
The row of the result set. **Row 1** contains the column names; **Row 2** is the first data row (Multi-select supported).

Column

The column of the result set (Multi-select is supported).

A few helpful hints:

- **Page Relative** is the default **Reference type** for the **Row Number** and **Column** arguments. This makes it simple to select a few rows and columns, apply the function, and then turn the result into a flex table, which will contain all the rows and columns of the result.
- A useful resource for DAX is *The Definitive Guide to DAX* by Russo and Ferrari. The authors have published other books on Analysis Services, which are also very good.
- In MS Power BI, to capture the DAX query used by Power BI:
 1. Open Power BI and select an AS Tabular Server and Model
 2. On the **View** tab, click **Performance analyzer**.



3. Click **Start Recording**, create a visualization (it's helpful to do a table), and **Copy Query**.

Flex Views

The screenshot displays the Performance analyzer tool. The table below shows the recorded events:

Name	Duration (ms)
Recording started (7/21/2020 9:26:11 AM)	-
Changed a filter	-
Table	197
DAX query	10
Visual display	42
Other	145

Two red arrows in the image point to the 'Start recording' button and the 'Copy query' link in the table.

You can paste the query into Notepad to examine it, or paste it directly into the Flex Pane or worksheet.

BranchData

Retrieves dimension-branch metadata from the application.



Flex Views

	Forecast Feb-2016	Budget Feb-2016	CY FCST vs CY Budget Var	CY FCST vs CY Budget %
Full Time Equivalent	410	410	-	
Total Employee Compensation	1,980,291	1,985,682	5,391	0%
Travel & Entertainment	937,833	950,444	12,611	1%
Supplies & Services	188,460	174,161	(14,299)	-8%
Communications & IT	79,705	79,838	133	0%
Freight & Shipping	404,098	412,929	8,831	2%
Occupancy	415,733	446,110	30,377	7%
Maintenance	110,746	161,613	50,867	31%
Insurance	186,988	204,666	17,678	9%
Utilities	173,313	185,978	12,665	7%
Depreciation & Amortization	104,757	112,418	7,661	7%
Other Expenses	74,683	80,934	6,251	8%
Total Overhead Expenses	4,656,606	4,794,773	138,167	3%

Arguments

Used to

View Specifier

Specify the *view* that includes the dimension-branch you want to define. An asterisk (*), refers to the primary *view*, or the flex *view* itself. Secondary *views* display both the *model* ID and *view* ID separated with a semi-colon (;).

Example: ModelID;ViewID

Dimension

Specify the *dimension* that includes the branch data you want to define.

Example: DimensionID

Property

Specify how you want to display the members.

- **Name** – display the member name only
- **ID** – display the member ID only
- **IDandName** – display both the member ID and name
- **LevelDefault** – display the members based on the *level's* **Member Display** property
- **ViewDefault** - display the members based on the definition selected on the CONTROL® View tab > **Dimensions** > **Display**.

Member

Specify the branch member you want to retrieve. The reference needs to include both the *level* ID and member ID separated with a space.

Example: LevelID MemberID



Flex Views

(Optional) Qualifier

If the branch includes a listing *level* or *attribute*, you can use this to identify the *level* or *attribute* to display. Typically, the **Reference type** will be **Branch Relative**.

Department	Division	Forecast Feb-2016
101 LA - Manufacturing	Gas Scooter	109,011
102 LA - Operations	Gas Scooter	37,107
103 LA - Quality Control	Gas Scooter	30,719
104 LA - Inventory Control	Gas Scooter	130,715
105 LA - Transportation	Gas Scooter	322,381
106 LA - Accounting	Gas Scooter	32,066
107 LA - IT	Gas Scooter	78,682
108 LA - HR	Gas Scooter	46,809
109 LA - Sales	Gas Scooter	36,520
110 LA - Parts & Services	Gas Scooter	28,109
111 LA - Admin	Gas Scooter	22,119
112 LA - Finance	Gas Scooter	75,637
113 LA - R&D	Gas Scooter	400,377
201 LA - Internet Sales	Sales & Distributi	281,847
202 LA - Warehousing	Sales & Distributi	52,385
203 LA - Export/Import Services	Sales & Distributi	198,836
204 Daytona - Sales & Service	Sales & Distributi	146,376
205 Detroit - Sales & Service	Sales & Distributi	91,041
206 Detroit - Warehousing	Sales & Distributi	54,614

Command

Places a hyperlink or command button in the cell that runs a standard command, *script*, *mapping*, or *transform*; or opens another view, *form*, *sheet* or *book*.

	A	B	C	D	E
2	 COMPANY NAME Company Slogan USD Amounts not Scaled	Input - T&E Scenario Forecast :: Department HQ - IT			
3		Close Regenerate File Data Open Dashboard			
7					
8					
9					
10	Auto Expenses	500	500	500	500
11	Meals and Entertainment Expense	750	750	750	750
12	Travel Expense	5,000	5,000	5,000	5,000
13	Travel & Entertainment	6,250	6,250	6,250	6,250

Arguments Description

Command Type

The available command types fall into three general categories:

- **Standard Commands** - Common operations, typically on the open view. Most of these functions are available on the ribbon; however, for simplicity and clarity for casual users, you can install the commands on the flex view.



Flex Views

- **Open CONTROL® objects** - Opens a specified *view, form, sheet, book, or Edit book*. Use these options to make it easy for users to access related or supporting information.
- **Run Processes** - Runs a predefined *script, transform, or mapping*. Use these options to give end-users the ability to import or export data, load or manipulate the data in their *view*, publish data, or perform other complex functions.

Command The available commands depend on the **Command Type** you select. The **Standard Commands** are self-explanatory. Some require additional information defined in the **Options** argument.

Options A text string that contains one or more options, separated by a semi-colon. Each option is of the form: **OptionName=OptionValue**
If you leave the argument blank, CONTROL® supplies an appropriate default value. Here is the description of each option:

- **CommandButton** - If TRUE, the cell contains a button, otherwise it will contain a hyperlink. For most commands, the default is FALSE. For commands that reference a selection in the worksheet (Launch Filter Wizard, Create New Member, Lock, Unlock and any run command, the default is TRUE.
- **Caption** - The text string that appears on the button or the value of the hyperlink. If not supplied, CONTROL® creates a readable caption such as "Open Sales Report".
- **Tooltip** - The text string that appears on the button or the value of the hyperlink.
- **Macro** - For the **Offline Excel Macro** standard command, the name of the macro to execute (Required).
- **Dimension** - For the **Launch Filter Wizard** and **Create New Member** standard commands, required to supply the ID of the *dimension*. You can also specify Dimension=Selection to have CONTROL® determine the *dimension* from the selected cell(s) on the worksheet.

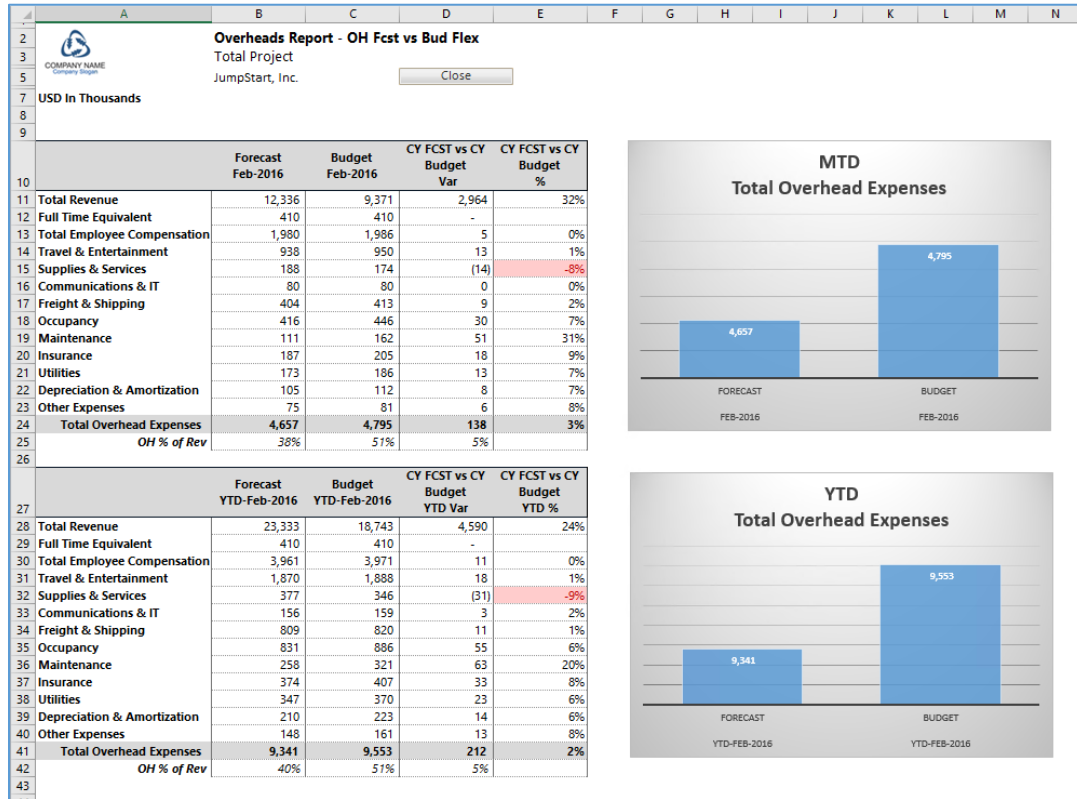
Note: The default width for the command buttons is 100 pixels. The button height defaults to the row height; however, you can manually adjust both the button width and height from the Excel **Developer** tab using either the **Design Mode** or **Properties** command. For information on how to override the default button width, refer to the **Keywords** section and the **KCI_FlexButtonWidth** keyword.



Flex Views

Instructor led exercise

In this exercise, we will add a command button to the **Report – OH Fcst vs Budget Flex** view that will close the view.



Steps:

1. Open the **Report – OH Fcst vs Bud Flex** view, and open the Flex Pane.
2. On the worksheet, select cell D5.
3. In the flex pane, double click the **Command** flex function.
4. Define the following arguments:

Argument	Definition
CommandType	<ul style="list-style-type: none"> • Reference Type: Absolute • Selection: StandardCommands
Command	<ul style="list-style-type: none"> • Reference Type: Absolute • Selection: Close
Options	<ul style="list-style-type: none"> • Reference Type: Absolute



Flex Views

- Selection:
 CommandButton=TRUE;Caption="Close";Tooltip=Close the view

5. Click **Apply to cell D5**.
6. Save the flex view.

Comment

Retrieves or saves *view* comments based on the defined arguments.

The screenshot shows a financial report titled "Overheads Report - Flex View Example" for "Total Project" at "Jumpstart, Inc.". The report displays a comparison between Forecast and Budget for February 2016, along with year-over-year percentage changes. A comment "[Administrative User] Find a new vendor" is visible in cell D14. To the right, the "View - Report - Flex View Example" configuration panel is open, showing the "View Specifier" with arguments: EXMP_DEPT p[0], EXMP_OH_ACCTS r[0], and SCEN_TIME_VAR c[0]. The "Reference type" is set to "Absolute" and the "Source type" is "In Use". The "Apply to cell" is set to "F14".

	Forecast Feb-2016	Budget Feb-2016	CY FCST vs CY Budget Var	CY FCST vs CY Budget %	Comments
Full Time Equivalent	410	410	-		
Total Employee Compensation	1,980,291	1,985,682	5,391	0%	
Travel & Entertainment	937,833	950,444	12,611	1%	
Supplies & Services	188,460	174,161	(14,299)	-8%	[Administrative User] Find a new vendor
Communications & IT	79,705	79,838	133	0%	
Freight & Shipping	404,098	412,929	8,831	2%	
Occupancy	415,733	446,110	30,377	7%	
Maintenance	110,746	161,613	50,867	31%	
Insurance	186,988	204,666	17,678	9%	
Utilities	173,313	185,978	12,665	7%	
Depreciation & Amortization	104,757	112,418	7,661	7%	
Other Expenses	74,683	80,934	6,251	8%	
Total Overhead Expenses	4,656,606	4,794,773	138,167	3%	

Arguments

Used to

View Specifier

Specify the view that includes the *dimensions* you want. An asterisk (*), refers to the primary *view*, or the flex view itself. Secondary *views* display both the *model* ID and *view* ID separated with a semi-colon (;).

Example: ModelID;ViewID

Dimension ID

There is a separate argument for each *dimension* defined in the *view*. Specify a branch member for each *dimension*. The reference needs to resolve both the *level* ID and member ID separated with a space.

Example: LevelID MemberID

Data

Retrieves and/or saves data based on the defined arguments.



Flex Views

	Forecast Feb-2016	Budget Feb-2016	CY FCST vs CY Budget V%	CY FCST vs CY Budget %
11 Full Time Equivalent	410	410	-	
12 Total Employee Compensation	1,980,291	1,985,682	5,391	0%
13 Travel & Entertainment	937,833	950,444	12,611	1%
14 Supplies & Services	188,460	174,161	(14,299)	-8%
15 Communications & IT	79,705	79,838	133	0%
16 Freight & Shipping	404,098	412,929	8,831	2%
17 Occupancy	415,733	446,110	30,377	7%
18 Maintenance	110,746	161,613	50,867	31%
19 Insurance	186,988	204,666	17,678	9%
20 Utilities	173,313	185,978	12,665	7%
21 Depreciation & Amortization	104,757	112,418	7,661	7%
22 Other Expenses	74,683	80,934	6,251	8%
23 Total Overhead Expenses	4,656,606	4,794,773	138,167	3%

Arguments

Used to

View Specifier

Specify the *view* that includes the dimension-branch you want to define. An asterisk (*), refers to the primary *view*, or the flex *view* itself. Secondary *views* display both the *model* ID and *view* ID separated with a semi-colon (;).

Example: ModelID;ViewID

Dimension ID

There is a separate argument for each *dimension* defined in the *view*. Specify a branch member for each *dimension*. The reference needs to include both the *level* ID and member ID separated with a space.

Example: LevelID MemberID

DataReadOnly

Retrieves read-only data based on the defined arguments.

Arguments

Used to

View Specifier

Specify the *view* that includes the dimension-branch you want to define. An asterisk (*), refers to the primary *view*, or the flex *view* itself. Secondary *views* display both the *model* ID and *view* ID separated with a semi-colon (;).

Example: ModelID;ViewID



Flex Views

Dimension ID

There is a separate argument for each *dimension* defined in the *view*. Specify a branch member for each *dimension*. The reference needs to include both the *level* ID and member ID separated with a space.

Example: LevelID MemberID

HeaderData

Specify text and system-defined *keyword* related to *views*.

The screenshot displays a Flex View report titled 'Overheads Report - Flex View Example'. The report includes a table with the following data:

	Forecast Feb-2016	Budget Feb-2016	CY FCST vs CY Budget Var	CY FCS
Full Time Equivalent	410	410	-	
Total Employee Compensation	1,980,291	1,985,682	5,391	
Travel & Entertainment	937,833	950,444	12,611	
Supplies & Services	188,460	174,161	(14,299)	
Communications & IT	79,705	79,838	133	
Freight & Shipping	404,098	412,929	8,831	
Occupancy	415,733	446,110	30,377	
Maintenance	110,746	161,613	50,867	
Insurance	186,988	204,666	17,678	
Utilities	173,313	185,978	12,665	
Depreciation & Amortization	104,757	112,418	7,661	
Other Expenses	74,683	80,934	6,251	
Total Overhead Expenses	4,656,606	4,794,773	138,167	

The configuration panel on the right shows the 'HeaderData' property set to 'Prepared by: &UserName'. The 'View Specifier' is set to '*'. The 'Reference type' is set to 'Absolute'. The 'Apply to cell A9' button is visible.

Arguments

Used to

View Specifier

Specify the *view* that includes the information you want to include. An asterisk (*), refers to the primary *view*, or the flex *view* itself. Secondary *views* display both the *model* ID and *view* ID separated with a semi-colon (;). Example: ModelID;ViewID

String

Specify one or more system-defined *keywords* related to *views*. You can enter characters and spaces between *keywords*.

Example: &MemberID(1P) | &MemberName(1P)

Note: You can also use the *keywords* &1, &2, etc. to reference each row of the *view* header.



Flex Views

Hyperlink

Retrieves or saves *view* hyperlinks based on the defined arguments.

Arguments	Used to
View Specifier	Specify the <i>view</i> that includes the dimension-branch you want to define. An asterisk (*), refers to the primary <i>view</i> , or the flex <i>view</i> itself. Secondary <i>views</i> display both the <i>model</i> ID and <i>view</i> ID separated with a semi-colon (;). Example: ModelID;ViewID
Dimension ID	There is a separate argument for each <i>dimension</i> defined in the <i>view</i> . Specify a branch member for each <i>dimension</i> . The reference needs to include both the <i>level</i> ID and member ID separated with a space. Example: LevelID MemberID

Keyword

Retrieves resolved *keyword* values.

The screenshot displays a Flex View report titled "Overheads Report - Flex View Example". The report includes a table with columns for Forecast, Budget, and Budget Var. A red box highlights the value "201602" in the "Current Month" field. A red arrow points from this box to the "Keyword" field in the configuration panel, which is set to "&CURRENT_MONTH, Dynamic". The configuration panel also shows a list of resolved keyword values.

Name	Argument Value
ObjectVerbDialogSettings	&OBJECTVERBDIALOGSETTING
PRCS_EXMP_SUBACCT_VAR	&PRCS_EXMP_SUBACCT_VAR_
SLCT_USER	&SLCT_USER
700 Maintenance	
800 Public	
850 Security	
875 System	
910 Overheads	
950 Best Practice	

Arguments	Used to
Keyword	Specify a <i>keyword</i> to resolve.
(Optional) Scope	This function does not use this argument.



Flex Views

Resolution Behavior

Specify the behavior of the *keyword* resolution in the *view*. There are three options:

- **Dynamic** – The *view* responds to dynamic changes in the *keyword's* replacement value via a *form*, *script*, or direct edit of the *keyword's* **Replacement** property, after you regenerate the *view*. This is the default behavior.
 - **Fixed** – The *view* does not respond to dynamic changes in the *keyword's* replacement value, but uses the value displayed in the cell. Use this option if you do not want the *view* to conform to the current replacement value of the *keyword*.
 - **Fixed On Save** – The same behavior as **Dynamic**, unless you save the *view* offline in an Excel workbook, or save it online using **View > Save As**, then it will become fixed. Use this behavior if you are creating a *view* for multiple users, who have different scoped values for a *keyword* and they might save it offline, or save a personal copy, as a new *view*, or you will publish the *view* with different *keyword* definitions. This option reduces the chance of saving the *view* offline, and then later reopening it and saving the data to an incorrect member.
-

List

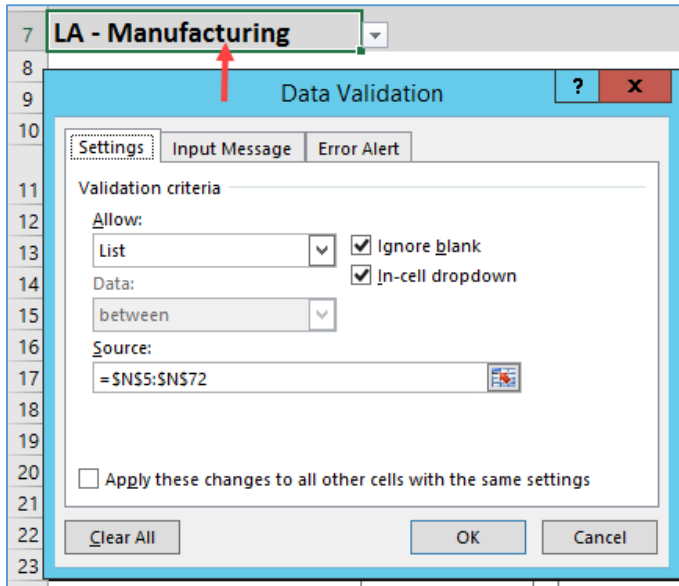
Creates a list based on an object. The original intent of this function was to provide a list of members based on one of the *view's* dimension-branches, *hierarchies*, or *levels*.

List Flex Function	
LA - Manufacturing	101
LA - Operations	102
LA - Quality Control	103
LA - Inventory Control	104
LA - Transportation	105
LA - Production	100_FUNC

You could then reference the list with an Excel data validation function.



Flex Views



Based on the member you selected, you could use an Excel **VLOOKUP** function to find the ID from the member list.

	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Model/View:	EXMP_OH;EXMP_RPT_OH_ACT												
2	Department:	=VLOOKUP(B7,SNS5:SO\$72,2,FALSE)												
3	Time Period:	2018	201802											
4	Projects:	100_PROD	200_MKTG	300_COST	400_GENL	10_TOT_PROJ								
5		Report - OH Expense Review												
6		YTD - Actuals												
7	LA - Manufacturing													
8														
9	Total YTD Sales		5,532,549											
10														

LA - Manufacturing	101
LA - Operations	102
LA - Quality Control	103
LA - Inventory Control	104
LA - Transportation	105
LA - Production	100_FUNC

Your flex functions or CONTROL® Reference Functions could then reference the ID to update the information in your report for the selected member.



Flex Views

	B	C	D	E	F
1	Model/View:	EXMP_OH;EXMP	RPT_OH_ACT		
2	Department:	10_CO			
3	Time Period:	2018		201802	
4	Projects:	100_PROD		200_MKTG	300
5				Report - OH Expense	
6				YTD - Actuals	
7	JumpStart, Inc.				
9	Total YTD Sales	5,532,549			
11		Product Enhancement	Market Growth	Co	
12	Full Time Equivalent	-	-		
13	Total Employee Compensation	-	-		
14	Travel & Entertainment	=ControlGet(\$C\$1,\$C\$2,\$C\$4,\$A14,\$E\$3)/100			
15	Supplies & Services	21		820	

However, with the addition of the **List Selection** flex function, you most likely will not use this function that often.

Arguments	Used to
Source	Specify the object used as the basis for the list. Typically, the Source Type will be ViewDimensionBranch .
Filter	Specify a pre-defined <i>filter</i> to limit the list of members defined in the Row argument. (Note: You cannot use ad hoc <i>filters</i> .)
Column Option	Specify how you want the member Names and IDs displayed on the worksheet. There are three options: <ul style="list-style-type: none"> • ID and Name – Separate Columns: The first column displays the ID and the second column displays the Name. Note: You have to select a range that includes two contiguous columns for this option. • Name and ID – Separate Columns for VLOOKUP: The first column displays the ID and the second column displays the Name. Note: You have to select a range that includes two contiguous columns for this option

Flex Views

- **ID Name in one column:** Both the ID and Name display together in one column.

Row	Specify the available members from the source object.
Column	Display the member ID, name, or both. The options differ based on the Column Option you select. If you selected Column Option : <ul style="list-style-type: none"> • ID and Name – Separate Columns: Then select Member Identifier or Member Name, to display one column with either the member ID or name; or select both to display two columns with the member ID in the first column and the member name in the second column. • Name and ID – Separate Columns for VLOOKUP: Then select Member Name or Member Identifier, to display one column with either the member name or member ID; or select both to display two columns with the member name in the first column and the member ID in the second column. • ID Name in one column: Then select Combined Member ID and Name, to display one column with both the member ID and name.

ListSelection

Creates a list based on an object. The intent of this function is to replace the use of a *form* to select members from a *view's* dimension-branch. The members you select can update the *filter* in the underlying *view(s)*, or define a *keyword* used in a scripted process.

The screenshot shows a financial report titled "Overheads Report - Flex View Example" with a table of expenses. A "List Selection" dialog box is open, showing a list of months from Jan-16 to Oct-16. A red arrow points from the "Feb-16" entry in the list to the "Current Month: Feb-16" field in the report. To the right, a configuration panel for the "ListSelection" function is visible, showing the source as "Level(MTH)", the filter as "MTH 201601 thru 201612", and the display option as "Name". The panel also includes a search bar and a "Resolve" button.

Forecast Feb-2016		CY FCST vs CY Budget %
Full Time Equivalent	410	
Total Employee Compensation	1,980,29	0%
Travel & Entertainment	937,833	1%
Supplies & Services	188,460	-8%
Communications & IT	79,705	0%
Freight & Shipping	404,098	2%
Occupancy	415,733	7%
Maintenance	110,744	31%
Insurance	186,988	9%
Utilities	173,313	7%
Depreciation & Amortization	104,751	7%
Other Expenses	74,683	8%
Total Overhead Expenses	4,656,600	3%



Flex Views

Arguments	Used to
Source	Specify the object used as the basis for the list. Typically, the Source Type will be ViewDimensionBranch .
Filter	<p>(Optional) Specify a <i>filter</i> to limit the list of members defined in the Row argument. You can specify:</p> <ul style="list-style-type: none"> • A predefined <i>filter</i> for the <i>level</i> or <i>dimension</i> • A Worksheet reference to a cell containing a predefined <i>filter</i> ID or an ad hoc <i>filter</i> • An Indirect reference to a cell with a List or ListSelection flex function • An ad hoc <i>filter</i> enclosed in double quotes – e.g. "EXMP_DIV 100_DIV" • If the Source argument's Source type is ViewDimensionBranch, then you can use these additional <i>filter</i> options that limit the member list: <ul style="list-style-type: none"> a. Children(memberID) – Immediate children of the member b. Children(memberID, levelID) – Children of the member on a <i>level</i> c. Children(memberID, All) – Children of the member on all <i>levels</i> in the branch d. Level(levelID) – All members of the <i>level</i> in the branch e. For convenience: <ul style="list-style-type: none"> i. memberID is equivalent to Children(memberID, All) ii. levelID is equivalent to Level(levelID) <p>Note: The Filter argument for the List Selection does not filter the members in the Select Page dialog.</p>
Display Option	Specify how to display the members in the List Selection dialog. There are three options: ID and Name , Name Only , ID Only .
Selection	(Optional) If the argument is blank, then the value is set to the Default Value argument. For the All User <i>view</i> scope, the argument defaults to blank if there is a Default Value argument. By using a <i>keyword</i> in the Default Value argument, the <i>view</i> will open based on the <i>keyword's</i> scoped replacement value for each user. If both arguments are blank, then the <i>view</i> will use the first member in the list.
Property	Specify how to display the members on the worksheet. There are five options: Name , ID , IDandName , LevelDefault , and ViewDefault .

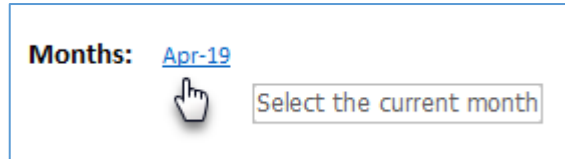
Flex Views

Keyword (Optional) Enter a *keyword* that will update temporarily with the value selected in the **List Selection** dialog. You can reference the *keyword* in a scripted process, or in the *filter* expression of the *view's dimension*.

Options You can select zero or more behaviors (separated by a semi-colon).

- **Tooltip** - Specifies a tooltip that displays when the cursor is over the hyperlink of the cell containing the **ListSelection** function.

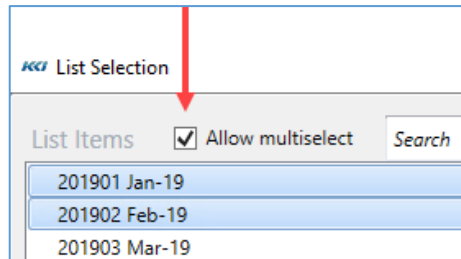
Example: Tooltip=Select the current month



- **MultiSelectTRUE** - Enables a check box in the **List Selection** dialog. If you select the check box, then you can select multiple members.

Use this option to set a *keyword* used in a *filter* with multiple selected members.

Example: MultiSelect=TRUE



- **SelectPageTRUE** - Adds the list selection to the **Select Page**, **Expand Multipage**, and **Print** dialogs launched from the CONTROL® Navigator and View ribbons.

Note: Only use this option if the list selection does not reference a *dimension* on the *view's Pages* edge. If it does, then you will see the *dimension* twice in the dialogs, one for the *dimension* in the *view*, and another one for the list selection.

Example: SelectPage=TRUE

- **SelectPageSynchronize** - For list selections that use one of the primary *view's* page dimension-branches as its source – causes page selection on the dimension-branch to set the list selection member
-

Flex Views

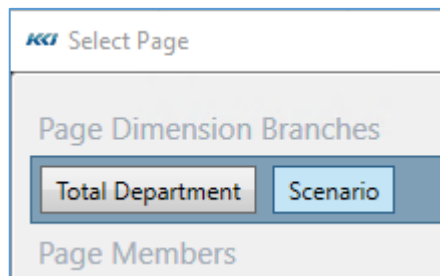
and vice-versa (except when the **Select Page** dialog includes members that are not included in the **List Selection filter**).

This option is identical to **SelectPageTRUE** in all other respects.

Example: `SelectPage=Synchronize`

- **Caption** - This defines the caption of the list selection button displayed on the **Select Page** dialog. Use this in conjunction with the **SelectPageTRUE** option.

Example: `SelectPage=TRUE;Caption="Scenario"` (double quotes are required)



- **MultiOption** - This will automatically combine the **SelectPageTrue** and **Caption** options. You can use the Ctrl key and left mouse button to select other options as well. They need to be separated by a semi-colon and the entire string encapsulated by double quotes.

Example: `"MultiSelect=TRUE;SelectPage=TRUE;Caption="Scenario""`

Default Value

Specify the default value of the selected member when the *view* opens. There are three **Reference types** to choose from **Absolute**, **Keyword**, and **Branch Relative**.

Instructor led exercise

In this exercise, we will replace the **HeaderData** flex function with the **ListSelection** flex function in the **Report – OH Fcst vs Bud Flex** *view*.

Flex Views

The screenshot displays the CONTROL software interface. The main window shows a worksheet titled "Report - OH Fcst vs Bud Flex" with columns for Forecast, Budget, and CY FCST vs CY Budget. Two bar charts are visible: "MTD Total Overhead Expenses" and "YTD Total Overhead Expenses". A red arrow points from cell B5 in the worksheet to the Flex Pane configuration window on the right. The Flex Pane window shows the configuration for the flex function in cell B5, including the source, display options, and arguments.

	Forecast Feb-2016	Budget Feb-2016	CY FCST vs CY Budget Var	CY FCST vs CY Budget %
Total Revenue	12,336	9,371	2,964	32%
Full Time Equivalent	410	410	-	-
Total Employee Compensation	1,980	1,986	5	0%
Travel & Entertainment	938	950	13	1%
Supplies & Services	180	174	(14)	-8%
Communications & IT	80	80	0	0%
Freight & Shipping	404	413	9	2%
Occupancy	416	446	30	7%
Maintenance	111	162	51	31%
Insurance	187	205	18	9%
Utilities	173	186	13	7%
Depreciation & Amortization	105	112	8	7%
Other Expenses	75	81	6	8%
Total Overhead Expenses	4,657	4,795	138	3%
OH % of Rev	38%	51%	5%	

	Forecast YTD-Feb-2016	Budget YTD-Feb-2016	CY FCST vs CY Budget YTD Var	CY FCST vs CY Budget YTD %
Total Revenue	23,333	18,743	4,590	24%
Full Time Equivalent	410	410	-	-
Total Employee Compensation	3,961	3,971	11	0%
Travel & Entertainment	1,870	1,888	18	1%
Supplies & Services	377	346	(31)	-9%
Communications & IT	156	159	3	2%
Freight & Shipping	809	820	11	1%
Occupancy	831	886	55	6%
Maintenance	258	321	63	20%
Insurance	374	407	33	8%
Utilities	347	370	23	6%
Depreciation & Amortization	210	223	14	6%
Other Expenses	148	161	13	8%
Total Overhead Expenses	9,341	9,553	212	2%
OH % of Rev	40%	51%	5%	

1. Open the **Report – OH Fcst vs Bud Flex** view, and open the Flex Pane.
2. On the worksheet, select cell **B5** that has the **HeaderData** flex function.
3. In the Flex Pane, click the **f_x** icon next to the flex formula box, select **ListSelection**.
4. Define the following arguments:
 - **Source:** DimSpec(EXMP_DEPT)
 - **Display Option:** IDandName
 - **Property:** Name
 - **Options:** SelectPage=Synchronize
 - **Default Value:** 10_CO
5. Click **Apply to cell B5**.
6. Save the flex view.

MemberData

Retrieves member data from the application for a *level* or *hierarchy*. It is independent of the **View Specifier**.

Arguments

Definition



Flex Views

Source	Define a <i>level</i> or <i>hierarchy</i> .
Member	Select the member(s) to display.
Property	Specify how you want to display the members. <ul style="list-style-type: none"> • ID – display the member’s ID • Name – display the member’s name • Description – display the member’s description • LevelID – display the member’s <i>level</i> ID • LevelName – display the member’s <i>level</i> name • OriginalID – Not applicable • MemberKey – display the member’s SQL member key

MetaData

Retrieves metadata for selected object properties from the application. It is independent of the **View Specifier**.

The screenshot displays a financial report titled "Overheads Report - Flex View Example" for "Jumpstart, Inc." with the current month set to "Feb-16". The report includes columns for "Forecast Feb-2016", "Budget Feb-2016", "CY FCST vs CY Budget Var", and "CY FCST vs CY Budget %". A red box highlights the "Variance" column header. To the right, a metadata configuration window is open, showing the formula `MetaData(Dimension(SCEN_TIME_VAR), Name,)` in cell C9. The "Object" dropdown is set to "Dimension(SCEN_TIME_VAR)". The "Source type" is "Dimension". A list of objects is shown, with "Variance" selected. The "Reference type" is "Absolute". The "Apply to cell" is "C9".

Arguments	Definition
Object	Select the object. If you choose the Absolute reference type, then you can select the object type from the Source type drop down, which will filter the available objects listed in the tree below.
Property	Select the one of the member’s properties to display.



Flex Views

(Optional) Scope Select a scope of the object.

NestedBranchData

Displays nested dimension-branch data in either separate columns, or a single column.

Arguments	Definition
View Specifier	Specify the <i>view</i> that includes the dimension-branch you want to define. An asterisk (*), refers to the primary <i>view</i> , or the flex <i>view</i> itself. Secondary <i>views</i> display both the <i>model</i> ID and <i>view</i> ID separated with a semi-colon (;). Example: ModelID;ViewID
Edge	Specify the <i>dimension(s)</i> to include. There are five Source types : <ul style="list-style-type: none">• ViewEdge - Reference the <i>dimensions</i> based on their edge assignment.• 1 Dimension – Reference a single <i>dimension</i>• 2 Dimensions – Reference a combination of two <i>dimensions</i>• 3 Dimensions - Reference a combination of three <i>dimensions</i>• 4 Dimensions – Reference a combination of four <i>dimensions</i>
Property	Specify how to display the members on the worksheet. There are five options: Name , ID , IDandName , LevelDefault , and ViewDefault .
Suppress Option	Specify how to handle the display of members with no data. There are four options available: <ul style="list-style-type: none">• No zero suppression – All Combinations• Suppress zero dimensional members• Suppress combinations with no data• Use all presentation options in the <i>view</i>
Merge Option	Specify how you want the members displayed on the worksheet. There are two options: <ul style="list-style-type: none">• Don't Merge – Separate Columns: the members for each <i>dimension</i> will be displayed in separate columns (Note: you will first need to select a range of columns based on the number of <i>dimensions</i> before creating this function.)• All identifiers in one column: the members for each <i>dimension</i> are concatenated into a single column.



Flex Views

Row	Select or reference the member(s) to display.
Column	Specify the <i>dimension(s)</i> you want to display. If the Edge source type is ViewEdge , then use the Shift key to make your selection; else select Merged Identifiers .

ViewData

Assigns flex IDs to a range of cells on the worksheet based on a defined *view* and selected page.

Arguments	Definition
View Specifier	Specify the <i>view</i> that includes the dimension-branch you want to define. An asterisk (*), refers to the primary <i>view</i> , or the flex <i>view</i> itself. Secondary <i>views</i> display both the <i>model</i> ID and <i>view</i> ID separated with a semi-colon (;). Example: ModelID;ViewID
Preferred ID Type	Specify the Reference Type for member IDs.
Preferred Data Type	Specify the Reference Type for data functions.
Row Edge Branch(es)	Select the <i>dimension(s)</i> to be on the Rows edge. It is not recommended to assign a <i>dimension</i> on the Pages edge of the underlying <i>view</i> to the Rows edge in the flex <i>view</i> .
Column Edge Branch(es)	Select the <i>dimension(s)</i> to be on the Columns edge. It is not recommended to assign a <i>dimension</i> on the Pages edge of the underlying <i>view</i> to the Columns edge in the flex <i>view</i> .
Force Read Only	Define if the flex <i>view</i> is updatable or not.

SQLQuery

SQLQuery accesses data from relational tables and *views*. The data is always read-only.

Note: Use the **Data** flex function on a source data *view* to create an updatable *view*.

Argument	Description
SQLSource	The ID of the datasource to query.



Flex Views

For security reasons, there are two **Source types**:

- **Specific Data Source** – a *datasource* where the user has read access. The query is restricted to accessing the underlying table or view. (For non-relational sources, the query will be applied to the relational image of the file or Excel range)
- **General Query** – a *datasource* in the **Dynamic Datasources** category, to which your administrator must give you access. You can write any query you want, but you must have **SELECT** privilege on the tables and views in your query. (For more details about the security, see *Release 10.5 Announcement, Enhancements for the Administrator*.)

Query Expression

A SQL expression or fragment.

When the **Source type** is **Specific Data Source**:

- The expression can be a query, a **WHERE** clause, an **ORDER BY** clause or both, but it can only reference the underlying table or view of the *datasource*.
- If the **SELECT** portion of the query is not specified, **SELECT * FROM table** is presumed.
- If the *datasource* has a non-empty SQL Query property, it is added to the **WHERE** clause with an **AND**.
- If the supplied query has no **ORDER BY** clause, the ordering in the *datasource* is used.

When the **Source type** is **General Query**, the query expression executes without modification.

The **Query Expression** argument supports worksheet references, which is a convenient way to assemble, customize, and experiment with the query.

Row Number

The row number in the *datasource* to retrieve.

Column

The column in the *datasource* to retrieve.

A few helpful hints:

- **Page Relative** is the default **Reference type** for the **Row Number** and **Column** arguments. This makes it simple to select a few rows and columns, apply the function, and then turn the result into a flex table, which will contain all the rows and columns of the result.
- **General Query** – Carefully consider the security ramifications before adding sources, and grant access to appropriate users only.

