Cubes an alternative to spreadsheets to mitigate risk and improve performance?

www.xcubes.net
Marc BRAUN
marcbraun@gmail.com

ABSTRACT

Multidimensional or even simple business modelling with traditional spreadsheets often leads to complex models which are difficult to understand and to maintain, is prone to errors and generally time consuming.

A cube-based system offers a structured and scalable approach to modelling, forecasting, reporting and data analysis with little manual intervention once the model is set up.

The practical demonstration will be made with XCubes a Multidimensional Spreadsheet System software package.

1. INTRODUCTION

For the purpose of this paper we started from the Paul Mireault, Honorary Professor at HEC Montréal, Multi-Dimensional challenge Kit on Excel ("Acme TechnoWidgets") to design a XCubes solution.

(Mireault, Structured Spreadsheet Modelling and Implementation: A Methodology for Creating Effective Spreadsheets, 2017) has developed a Structured Spreadsheet Modelling and Implementation methodology (SSMI) which is a strongly recommended starting point whatever software solution is being chosen.

We will first describe the key concepts of Cubes, present the result of the XCubes solution and summarize the benefits of using cubes instead of spreadsheets.

2. CUBES VERSUS SPREADSHEETS

The purpose here is not to stress out limitations of traditional spreadsheets for modelling purposes whether flat or multidimensional. There have been numerous articles on this subject.
(Panko, 2015) has written several articles about Spreadsheet errors.

(Read & Batson, 1999) in Spreadsheet Modelling Best Practice have concluded that spreadsheets are week at handling multi-dimensional data.

We will rather focus on the benefits of a full multidimensional approach.

(Litt, 2017) wrote an article about a Spreadsheet risk mitigation in Complex Multi-Dimensional Models in Excel where the solution appears to be a "PivotModel" inside Excel, similar to a MOLAP solution.

(Braun, XCubes Download for PC, 2019) has developed XCubes a software solution which does not rely on Excel and uses natively cubes to perform complex multi-dimensional modelling.

(Murphy) has listed in his comparison of Spreadsheets with other development tools several areas where other tools perform better in terms of Data separation, Scalability, Type Safe, Links which are addressed in XCubes.

The main differentiators between Cubes and Spreadsheets are as follows.

2.1 Data separation (n-tier architecture)

The multidimensional nature of the cubes cannot accommodate the Rows and Columns flat reference system used by traditional spreadsheets.

Therefore, the formulas are removed entirely from the spreadsheet surface which is only meant to receive data.

The removal of formulas at this level eliminates the risk of damaging the model during routine operations like model maintenance, data entry or zeroing a model to bring it to a clean state.

The n-tier architecture effectively separates the data from the business logic and also from the presentation level.

2.2 Structure with Dimensions

A dimension is list of two kinds of items:

- **Detail items** which receive data.
- **Calculated items** (in bold by convention) which store user defined formulas: calculation logic or business rules for the model.

The items are strongly typed for safety: number, text, date, … to prevent data entry mismatches. In spreadsheets date types are inferred during data entry.

The formulas are written in plain text using self-explanatory labels defined by the user in the 'Item Code' field. The 'Item Description' is used for displaying the meaningful label.

Item Code	Item Description	Formula
SA	Sales	
COGS	Cost of goods sold	
Margin	**Gross margin**	**SA - COGS**

Figure 1- Sample dimension items

Storing formulas in dimensions reduces drastically the number of formulas required in a model. The system automatically propagates the calculations across rows and columns as required.

Dimensions can be combined freely to build any kind of cubes desired by the user. There is no need for preexisting data to create dimensions or cubes.

Dimension are strictly independent from each other (self-contained) so that can be used in different cubes.
As a result, changes made to a dimension will automatically be reflected in other cubes using it.

2.3 Cubes as efficient data containers

Cubes are built with at least two dimensions representing Rows and Columns.

Dimensions can be further added, removed or exchanged as needed. This will automatically trigger the reorganization of cubes. Existing data is preserved during the process.

The same is true when items are added or removed from individual dimensions, giving its dynamic nature to the system.

Cubes can be sliced and diced using Page dimensions. This will break a large area of data into smaller parts. It allows easier data analyzes and facilitates data entry by showing a specific portion of a cube like a Region (see below).

Data can be quickly filtered on empty rows/columns, on detail/formula items or by turning on or off items in the dimensions.

The views are created by dragging and dropping dimensions in various areas (Row, Column, Page).

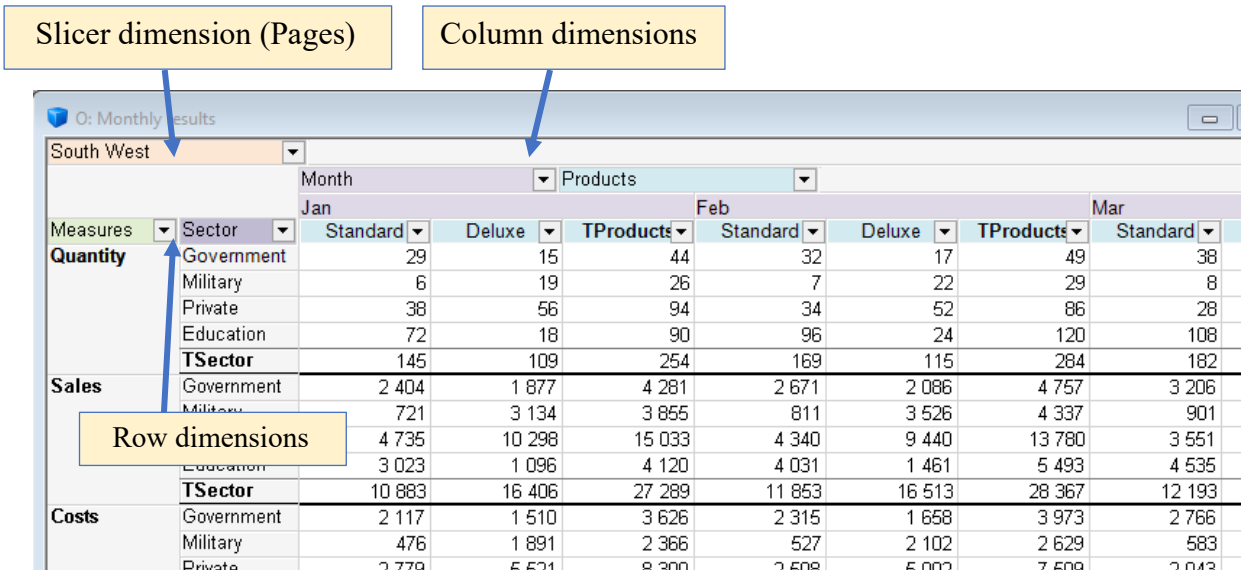


Figure 2 - Monthly results sample view

Cubes are compact as they often replace multiple tabs in a spreadsheet used to perform intermediary data display or calculation.

2.4 Centralized view of all Links, References and variables

The interrelations between cubes using data Links, References, Variables are maintained in a single place as part of user projects.

Figure 3 - List of references used in the project

Objects which cannot be treated as cubes because they do not have the two required dimensions can be defined as global variables for a project.

2.5 Data entry

As cubes are efficiently structured it easy to create robust and standardized data loading mechanisms. The most obvious is to load structured data directly from SQL Databases tables.
This is achieved via data links using industry standard data sources: ODBC, OleDB or DataLink.

Data loading prevents common data entry mistakes like omitted or incorrect entries due to typo errors, entries made in wrong cells leading to possibly overwriting formulas in spreadsheets.

2.6 Project oriented approach

Storing and managing dimensions, cubes, tables, links, references, variables, data sources and other objects in a single place (Project) managed entirely by the application eliminates the risk of having broken External Links which can render spreadsheets potentially unusable.
This will guaranty the integrity of the project and all of its components.

2.7 Pivot Tables in comparison

This feature has been built into Excel to allow data visualization and can be somewhat compared to a cube.

However, a Pivot Table cannot be created without and existing dataset. Calculation capabilities are limited although some level of customization is possible.
PivotTable generally work from a single data source and it is not convenient to load data from multiple sources.

On the other hand, cubes can be freely built based on ad hoc created dimensions and populated either by manual data entry or by multiple data links from various.

2.8 When to use cubes

Most of the time business models have several dimensions and thinking multidimensional right from the start will bring immediate benefits liked structured and scalable models.

Dimensions can be easily construed from various lists: time periods, products, employees, geographies, versions, financial statement items, and so on.

3. THE ACME TECHNOWIDGETS CASE STUDY

3.1 Model layout

The challenge from (Mireault, Multi-Dimensional Spreadsheet Challenge, 2018) is summarized below. The list of formulas used in this model can be found in Appendix 1.

Based on an annual production capacity and using input parameters from:

Sector data	Month split by sector
Monthly fixed costs	Regions-sector split
Sector-product split	Product data
Delivery cost	

The result is a monthly summary showing the following calculated values:

Unit sales by product and region	Unit sales by product
Unit sales by month	Sales amount by product and month
Profit by month	

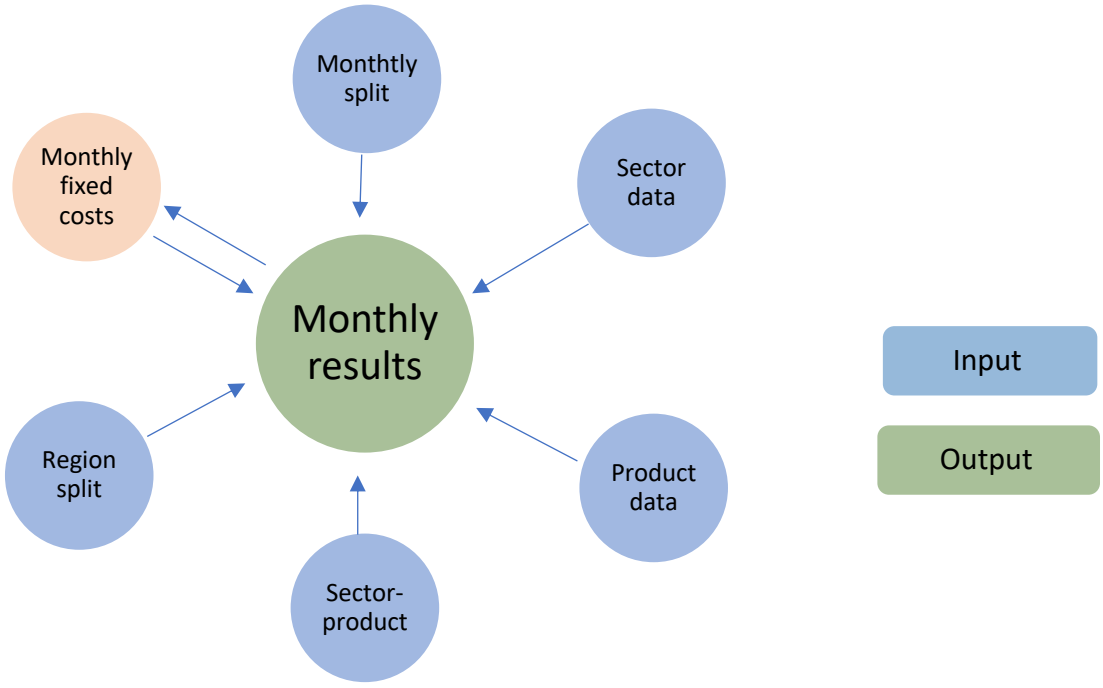
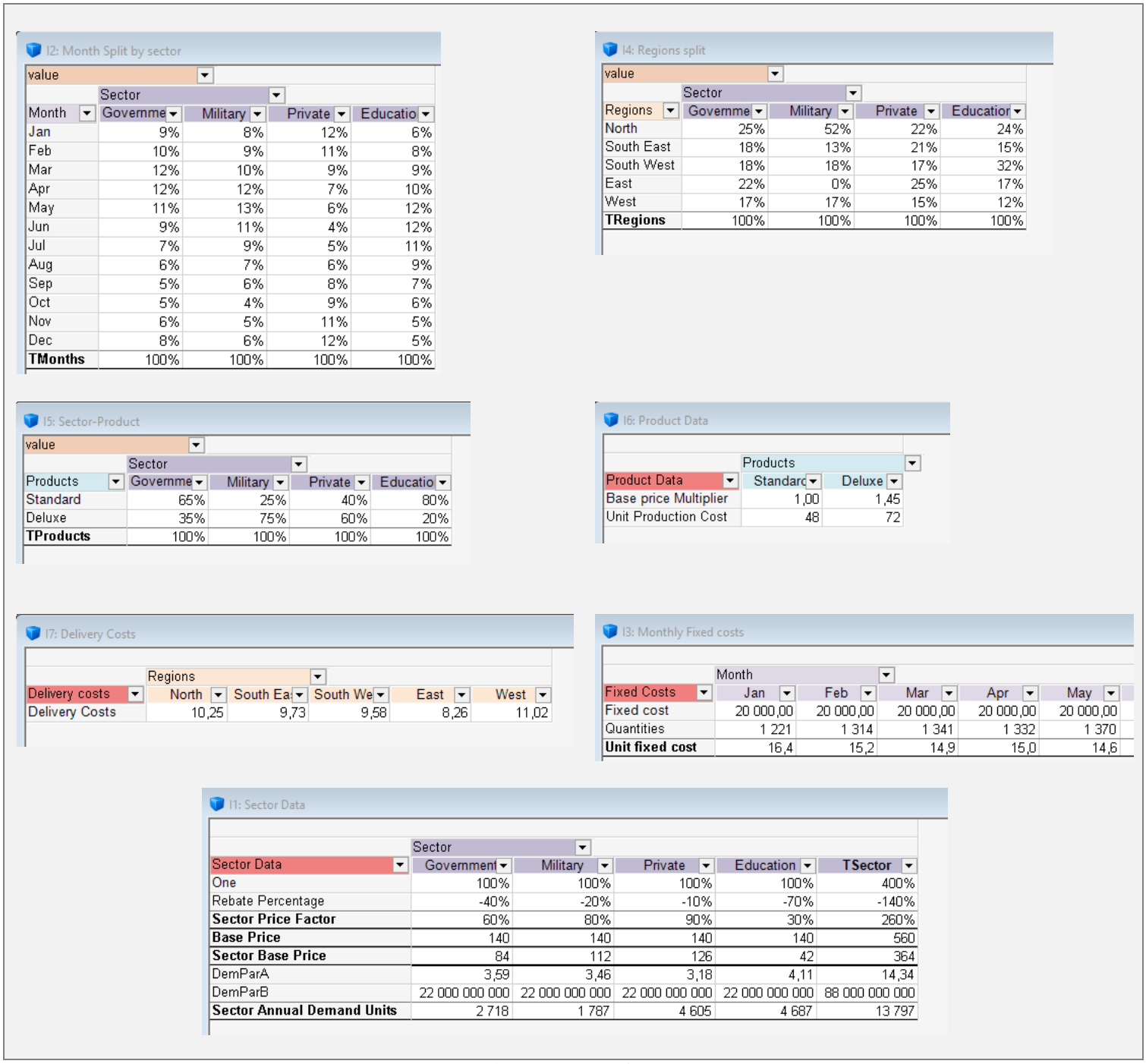


Figure 4 - ACME TechnoWidgets set of cubes.



I2: Month Split by sector

value	Sector			
Month	Governme	Military	Private	Educatio
Jan	9%	8%	12%	6%
Feb	10%	9%	11%	8%
Mar	12%	10%	9%	9%
Apr	12%	12%	7%	10%
May	11%	13%	6%	12%
Jun	9%	11%	4%	12%
Jul	7%	9%	5%	11%
Aug	6%	7%	6%	9%
Sep	5%	6%	8%	7%
Oct	5%	4%	9%	6%
Nov	6%	5%	11%	5%
Dec	8%	6%	12%	5%
TMonths	100%	100%	100%	100%

I4: Regions split

value	Sector			
Regions	Governme	Military	Private	Educatior
North	25%	52%	22%	24%
South East	18%	13%	21%	15%
South West	18%	18%	17%	32%
East	22%	0%	25%	17%
West	17%	17%	15%	12%
TRegions	100%	100%	100%	100%

I5: Sector-Product

value	Sector			
Products	Governme	Military	Private	Educatio
Standard	65%	25%	40%	80%
Deluxe	35%	75%	60%	20%
TProducts	100%	100%	100%	100%

I6: Product Data

	Products	
Product Data	Standarc	Deluxe
Base price Multiplier	1,00	1,45
Unit Production Cost	48	72

I7: Delivery Costs

	Regions				
Delivery costs	North	South Ea	South We	East	West
Delivery Costs	10,25	9,73	9,58	8,26	11,02

I3: Monthly Fixed costs

	Month				
Fixed Costs	Jan	Feb	Mar	Apr	May
Fixed cost	20 000,00	20 000,00	20 000,00	20 000,00	20 000,00
Quantities	1 221	1 314	1 341	1 332	1 370
Unit fixed cost	16,4	15,2	14,9	15,0	14,6

I1: Sector Data

	Sector				
Sector Data	Government	Military	Private	Education	TSector
One	100%	100%	100%	100%	400%
Rebate Percentage	-40%	-20%	-10%	-70%	-140%
Sector Price Factor	60%	80%	90%	30%	260%
Base Price	140	140	140	140	560
Sector Base Price	84	112	126	42	364
DemParA	3,59	3,46	3,18	4,11	14,34
DemParB	22 000 000 000	22 000 000 000	22 000 000 000	22 000 000 000	88 000 000 000
Sector Annual Demand Units	2 718	1 787	4 605	4 687	13 797

Figure 5 - Cubes used as input for ACME Techno

The Unit Fixed Costs in "Monthly Fixed costs" have to be calculated separately so that they can be properly allocated to the products in the "Months result" cube.

The resulting cube ('Months result") is built with 5 dimensions (4.680 cells):

- Months: Jan to Dec, Total
- Regions: North, South East, South West, East, West, Total
- Products: Standard, Deluxe, Total
- Sector: Government, Military, Private, Education, Total
- Measures: Quantities, Sales, Costs, Profit

O: Monthly results

South West		Month			Products		
		Jan			Feb		
Measures	Sector	Standard	Deluxe	TProducts	Standard	Deluxe	TProducts
Quantity	Government	29	15	44	32	17	49
	Military	6	19	26	7	22	29
	Private	38	56	94	34	52	86
	Education	72	18	90	96	24	120
	TSector	145	109	254	169	115	284
Sales	Government	2 404	1 877	4 281	2 671	2 086	4 757
	Military	721	3 134	3 855	811	3 526	4 337
	Private	4 735	10 298	15 033	4 340	9 440	13 780
	Education	3 023	1 096	4 120	4 031	1 461	5 493
	TSector	10 883	16 406	27 289	11 853	16 513	28 367
Costs	Government	2 117	1 510	3 626	2 315	1 658	3 973
	Military	476	1 891	2 366	527	2 102	2 629
	Private	2 779	5 521	8 300	2 508	5 002	7 509
	Education	5 324	1 763	7 087	6 988	2 323	9 310
	TSector	10 695	10 684	21 380	12 337	11 084	23 421
Profit	Government	288	367	655	356	428	784
	Military	245	1 244	1 489	284	1 424	1 708
	Private	1 956	4 777	6 733	1 833	4 438	6 271
	Education	-2 300	-667	-2 967	-2 956	-861	-3 818

Figure 6 - ACME result cube

The formula for the Quantity measure would be:

```
  REFERENCE("Sector Annual Demand Units")
* REFERENCE("Month and Sector Split")
* REFERENCE("Region and Sector Split")
* REFERENCE("Sector and Product Split")
```

REFERENCE () can be compared to the LOOKUP () function in Excel.
It automatically walks thru the items of any common dimensions of the source and destination cubes.

I2: Month Split by sector

value	Sector			
Month	Governme	Military	Private	Educatio
Jan	9%	8%	12%	6%
Feb	10%	9%	11%	8%
Mar	12%	10%	9%	9%
Apr	12%	12%	7%	10%
May	11%	13%	6%	12%
Jun	9%	11%	4%	12%
Jul	7%	9%	5%	11%
Aug	6%	7%	6%	9%

Figure 7 - Month Split by Sector as source for REFERENCE()

For example, REFERENCE("Month and Sector Split") will walk thru the 'Month Split by Sector' (Figure 7) which is the source cube and pick the value (split percentage) at the intersection of Month and Sector during each iteration since the Month and Sector dimensions are shared in both cubes.

As the dimensions are strictly independent from each other the REFERENCE ("Label") function must be defined on a cube by cube basis.
The Label is just a description to identify the reference to define and has no further meaning.
The reference manager is used to connect each reference to a specific input cube (cf. Figure 3)
The references are resolved i.e. the values are fetched during cube recalculation.

3.2 Size of cubes

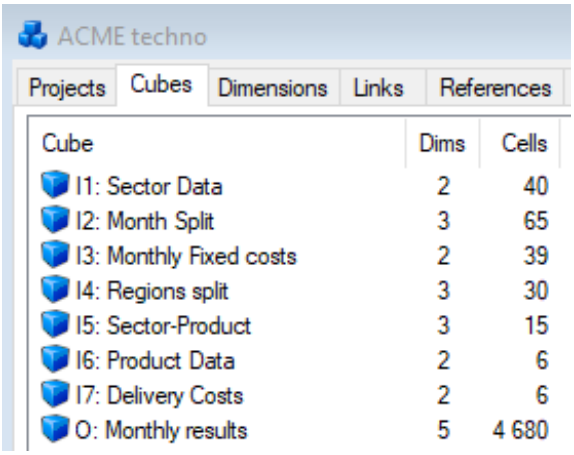


Figure 8 - Cube sizes

The size of cubes is the product of all item count in each dimension for each dimension.

For Monthly the result is:

Months (13) X Regions (6) X Products (3) X Sectors (5) X Measures (4) = 4.680 cells

Although there are actually 12 months, 5 regions, 2 products and so on, the Total item is accounted for as an additional item.

3.3 Number of formulas

The challenge model contains 20 variables and is summarized below:

	Items	12	4	2	5		
Calculated & Output variables	**Count**	**Month**	**Sector**	**Product**	**Region**	**Dimension Size**	**Number of Formulas**
Month	5	1				12	**60**
Sector	3		1			4	**12**
Product				1		2	**0**
Region					1	5	**0**
Product-Region	1			1	1	10	**10**
Sector-Product	3		1	1		8	**24**
Sector-Region			1		1	20	**0**
Month-Sector		1	1			48	**0**
Month-Sector-Product	2	1	1	1		96	**192**
Month-Sector-Poduct-Region	2	1	1	1	1	480	**960**
Month-Product-Region	1	1		1	1	120	**120**
Month-Product	2	1		1		24	**48**
Total Profit	1					1	**1**
Total	**20**						**1 427**

Table 1 - Formulas required in Excel

Dimensions	Number of Formulas	Kind of calculation
Measures	4	Quantity
		Sales
		Costs
		Profit
Month	1	Dimension total
Products	1	Dimension total
Regions	1	Dimension total
Sector	1	Dimension total
Sector data	4	Sector price Factor
		Base price (reference to a global variable)
		Sector base price
		Sector Annual Demand Units
Fixed Costs	1	Unit fixed cost
Total	**13**	

Table 2- Formulas required in XCubes

In the challenge the model contains **1.427** formulas whereas XCubes requires only **13** formulas to achieve the same results.

3.4 Scalability

The scalability of a model determines what efforts are required to grow or shrink it.
Some models might be of fixed or static others need to be adjustable.
One of the metrics that can be used to assess the complexity and the risks of making changes is the number of formulas needed to be added/altered in order to accommodate the change of structure.

Based on the current Acme model the impacts are as follows:

Adding one item in any dimension	In Excel	In XCubes
Sector	297	0
Product	677	0
Region	218	0

Table 3 - Adding an item in one dimension

The number of additional formulas required in Excel is computed by adding one item in the dimensions in the Table 1 - Formulas required in Excel, and calculating the difference with the initial 1.427 formulas.

Adding a new dimension like Years	In Excel	In XCubes
With N items	(N-1)[1] X 1.427	1 (Total item – if wanted)

Table 4 - Adding a dimension in a model

[1] N-1 indicates that there is already an implicit year in the model

Due to its extensible nature, XCubes requires no additional formulas when adding any number of items in any of the considered dimensions.

Furthermore, adding a new dimension with any number of items will generally add only a formula which is usually the Total for that dimension.

4. CONCLUSION

Cubes are best fitted when the model can be broken down in blocks to store well-structured data. They make analysis, calculation and visualization even on large data sets much easier.

They are most beneficial when scalability is an issue with models expected to grow over time.

Overall, they contribute to limit the risks of complex spreadsheet models by drastically reducing the number of required formulas and make even complex models easier to understand and maintain.

Finally, the user will be able to spend more time on analysis rather than worrying about reliability and performance.

The XCubes ACME TechnoWidgets solution is available for download (Braun, Windows Tutorials, 2019).

Appendix 1: List of the 20 formulas used in ACME TechnoWidgets

Variable	T	Dimension Set	Value / Formula	Comments
Sector Price Factor	C	Sector	1-Rebate Percentage	
Sector Base Price	C	Sector	Base Price * Sector Price Factor	Base price of the Standard widget sold in the sector
Sector Annual Demand Units	C	Sector	DemParB/Sector Base Price^DemParA	Estimated demand for the sector, in units
PR Unit Cost	C	ProductRegion	Unit Production Cost + Unit Delivery Cost	Cost of producing and delivering one unit
Annual Sector-Product Unit Sales	C	Sector-Product	Sector Annual Demand Units * Product Distribution per Sector	Annual sales
Price	C	Sector-Product	Sector Base Price * Base Price Multiplier	Price of each product in each sector
Annual Sector-Product Sales Amount	C	Sector-Product	Annual Sector-Product Unit Sales * Price	Annual sales amount of each product in each sector
MSP Unit Sales	C	Month-Sector-Product	Annual Sector-Product Unit Sales * Monthly Sales Distribution per Sector	Unit Sales per month, sector and product
MSP Sales Amount	C	Month-Sector-Product	Annual Sector-Product Sales Amount * Monthly Sales Distribution per Sector	Sales Amount per month, sector and product
MSPR Unit Sales	C	Month-Sector-Product-Region	MSP Unit Sales * Region Sales Distribution per Sector	Unit Sales per month, sector, product and region. (This is the finest granularity of the Unit Sales)
MSPR Variable Cost	C	Month-Sector-Product-Region	MSPR Unit Sales * PR Unit Cost	The variable cost of producing and selling the widgets per month, sector, product and region.
Monthly Variable Cost	C	Month	SUM(MSPR Variable Cost)	The monthly variable cost of producing and selling all the products in all the regions and for all the sectors.
Monthly Unit Sales	O	Month	SUM(MSPR Unit Sales)	The monthly number of units of all the products in all the regions and for all the sectors.
Monthly Sales Amount	C	Month	SUM(MSP Sales Amount)	The monthly sales amounts of all the products in all the regions and for all the sectors.
Monthly Costs	C	Month	Monthly Fixed Cost + Monthly Variable Cost	The monthly costs of all the products in all the regions for all the sectors.

Monthly Profit	C	Month	Monthly Sales Amount - Monthly Costs	The monthly profit of all the products in all the regions for all the sectors.
MPR Unit Sales	O	Month-Product-Region	SUM(MSPR Unit Sales)	The number of units sold for all the sectors per month, per product and per region.
MP Unit Sales	O	Month-Product	SUM(MSP Unit Sales)	The number of units sold for all the sectors and in all the regions per month and per product.
MP Sales Amount	O	Month-Product	SUM(MSP Sales Amount)	The sales amount for all the sectors and in all the regions per month and per product.
Total Profit	O		SUM(Monthly Profit)	The total profit.

T (Type): C = Calculated, O = Output

SUM in a formula, needs to be implemented with the Excel function

Braun, M. (2019, 03). *Windows Tutorials*. Retrieved from XCubes: https://www.dropbox.com/s/mb2fhtts0l30krg/ACME%20techno.zip?dl=1

Braun, M. (2019, 03 31). *XCubes Download for PC.* Retrieved from XCubes: https://www.dropbox.com/s/hudpts0il5zgf59/XCubes.zip?dl=1

Litt, S. (2017). Migating Spreadsheet Risk in Complex Multi- Dimensional Models in Excel. *EuSpRIG European Spreadsheet Risks Interest.* London: ISBN : 978-1-905404-54-4.

Mireault, P. (2017). *Structured Spreadsheet Modelling and Implementation: A Methodology for Creating Effective Spreadsheets* (2 ed.). Montréal: SSMI International.

Mireault, P. (2018). *Multi-Dimensional Spreadsheet Challenge.* Retrieved from SSMI International: https://www.ssmi.international/multidimensional-spreadsheet-challenge

Murphy, S. (n.d.). *Comparison of Spreadsheets with other development tools (limitations, solutions, workarounds and alternatives).* Retrieved from https://arxiv.org/ftp/arxiv/papers/0801/0801.3853.pdf

Panko, R. R. (2015). What We Don't Know About Spreadsheet Errors Today. *EuSpRIG.* London.

Read, N., & Batson, J. (1999, April). *Spreadsheet Modelling Best Practice.* Retrieved from EuSpRIG: http://www.eusprig.org/smbp.pdf