Viságe.BIT

An
OLAP / Data Warehouse
solution for
multi-valued databases

Abstract :

Viságe.BIT provides data warehouse/business intelligence/OLAP facilities to the multi-valued database environment. Boasting features normally associated with high end, *mainstream* solutions, and the best price/performance characteristics in the segment, as part of the broader Viságe Application Development Framework, **Viságe.BIT** offers a compelling solution.

Audience:

It is assumed that the reader will be familiar with terms associated with multi-valued databases, OLAP and related technologies in order to fully appreciate the information presented in this article.

Draft Release : Feb 2004

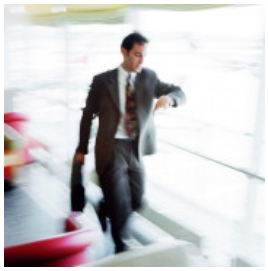
Prepared by : Ross Ferris

Viságe.BIT

Overview

Viságe.BIT is the Business Intelligence Toolkit (BIT) component of the Viságe Application Development Framework, and allows you to easily integrate 21st Century online analytical processing (OLAP) and data warehouse capabilities into your Viságe applications.

BIT is fully multi-value database aware, and provides a simple, intuitive means of converting the raw data currently stored in your database into a valuable, interactive reporting and data analysis resource that can be used for tactical advantage.



In today's fast-paced, information-driven economy, BIT empowers decision makers with a sophisticated analysis tool enabling the easy discovery of relationships, trends and basic facts that could otherwise remain dormant and unnoticed.

Although **Viságe.BIT** is an integral component of the Viságe ADF, it sets an impressive price/performance benchmark when compared to dedicated, single function tools.

BIT can invigorate existing applications, and free up IT resources.

You can define a BIT HiPerCube in minutes that can augment or replace tens, or even hundreds, of static reports and fixed enquiries.

Viságe.BIT will help make you more productive and identify opportunities **- what YOU will discover - Viságe.BIT** will help put you back in control.

Technology

MOLAP, not just OLAP!

At the core of **Viságe.BIT** is a powerful MOLAP engine written in C++ and crafted for maximum performance.

searchDatabase.com defines MOLAP as follows :

MOLAP (multidimensional online analytical processing) is online analytical processing (OLAP) that indexes directly into a multidimensional database. In general, an OLAP application treats data multidimensionally; the user is able to view different aspects or facets of data aggregates such as sales by time, geography, and product model. If the data is stored in a relational data base, it can be viewed multidimensionally, but only by successively accessing and processing a table for each dimension or aspect of a data aggregate. MOLAP processes data that is already stored in a multidimensonal array in which all possible combinations of data are reflected, each in a cell that can be accessed directly. For this reason, MOLAP is, for most uses, faster and more user-responsive than relational online analytical processing (ROLAP), the main alternative to MOLAP.

Please don't be tempted to assume that *multidimensional* is synonymous with *multi-valued*. In this context they are poles apart!

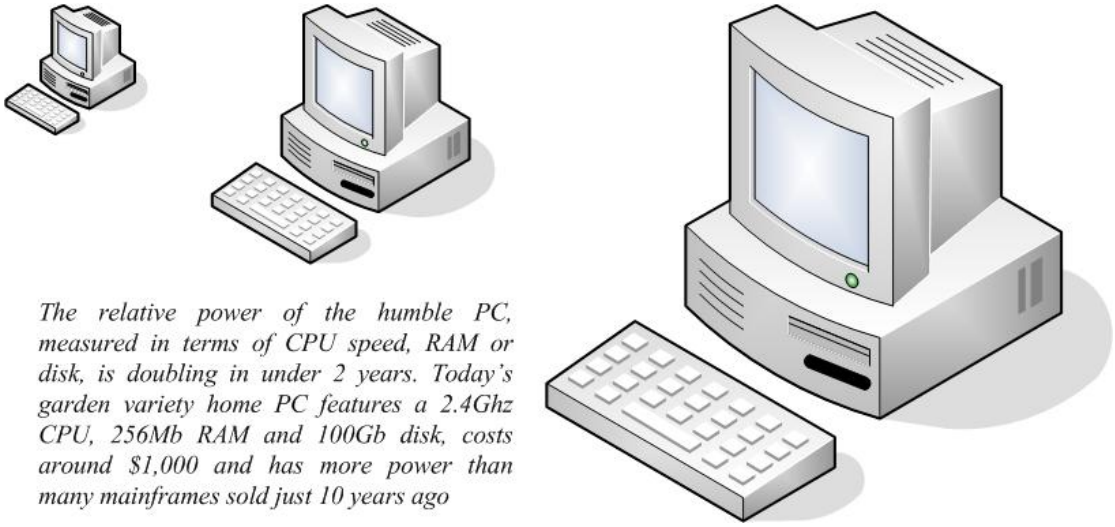
Our use of MOLAP technology means that with BIT you do **not** have to pre-define specific views of your HiPerCube – you can simply drag and drop dimensions in **any** order to view your cube in a different light.

BIT allows you to explore every possible combination of dimensions, and our use of technologies like advanced data compression means that contrary to popular belief, even though the MOLAP database (cube) keeps all dimensional combinations, it is a fraction of the size of the original database transactions.

Technology

21st Century Economics

Viságe.BIT has been designed to leverage the almost daily advances that are being made with PC based technology.



The relative power of the humble PC, measured in terms of CPU speed, RAM or disk, is doubling in under 2 years. Today's garden variety home PC features a 2.4Ghz CPU, 256Mb RAM and 100Gb disk, costs around $1,000 and has more power than many mainframes sold just 10 years ago

Whilst early iterations of BIT used the native multi-valued database for storage, and the CPU power of the main server to perform data transformations, we have now harnessed the power of the ubiquitous PC sitting on nearly every desktop.

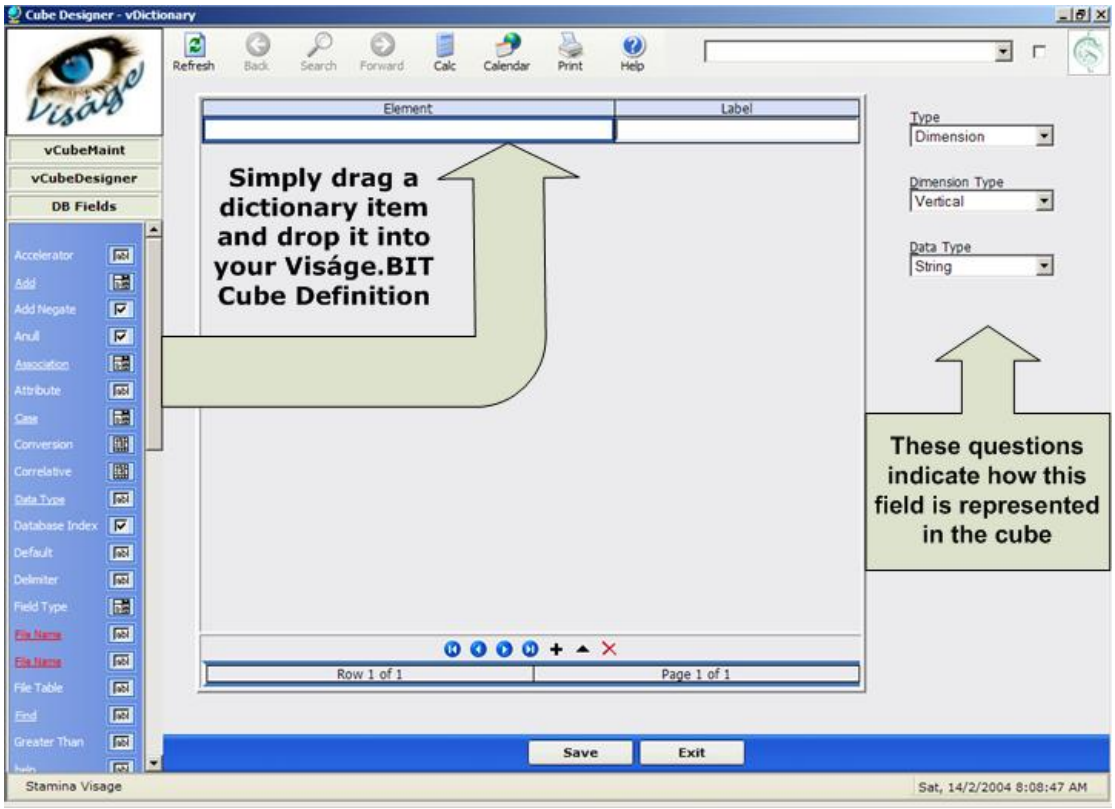
The results of this paradigm shift have been dramatic, on every front, and as each new generation of PC hardware is released, sporting faster CPU's and larger memory capacities, there is a direct performance jump with BIT.

However, you can achieve what I would classify as *astounding* results with quite modest hardware – especially if you make comparisons with solutions that try and achieve MOLAP by storing every possible dimensional combination in a relational database (like we used to do!).

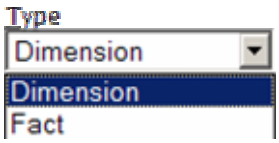
With the **Viságe.BITServer** component running on a PC with a 1Ghz PIII CPU, 512Mb RAM and Windows 2000 server, and data being extracted from a multi-valued database on a similarly configured machine, we have built cubes containing 14 dimensions & 4 facts from 1,000,000 records in under an hour !

We are not aware of any other product in the multi-valued marketplace that comes within **DAYS** of this figure ! **ASK THEM !!**

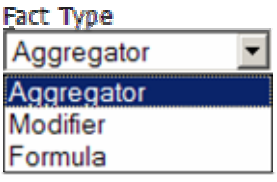
How do you define a Viságe.BIT Cube?



Broadly speaking, any element you drop onto the cube can be defined as either a **Fact** or a **Dimension.**

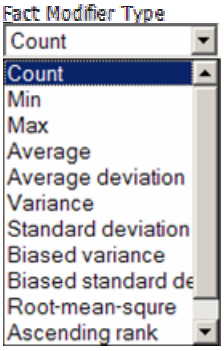


Facts are the things that you will ultimately be interested in, like *Quantity Sold, $ Profit, Productive Hours, Number of Calls* etc, and would typically have a numerical value.



BIT supports 3 types of facts, the most common being an **Aggregator,** which can be used to accumulate totals, or determine the minimum or maximum of the field, while a **Formula** lets you perform calculations on facts.

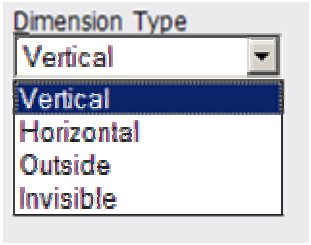
Modifier facts are derived from other facts too, and allow you to perform a variety of statistical calculations like counting, min/max, average, variance and biased variance, standard, average and biased standard deviation, root mean square and ranking operations.

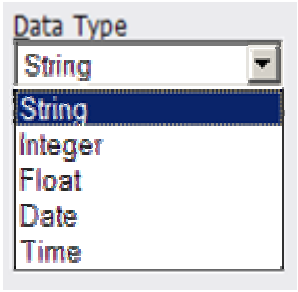


Dimensions have facts associated with them, and define **how** you want to be able to look at these facts. Examples of dimensions could include customer types, salesmen, regions, employee type, interests, product types and so on – the things you would *sort* and/or *break*-on with the native database query facilities (Access, Perform etc)

Adjusting and manipulating the dimensions of your BIT HiPerCube adjusts your ***view*** of the facts, the so called *slice and dice* functionality associated with OLAP/BI applications.

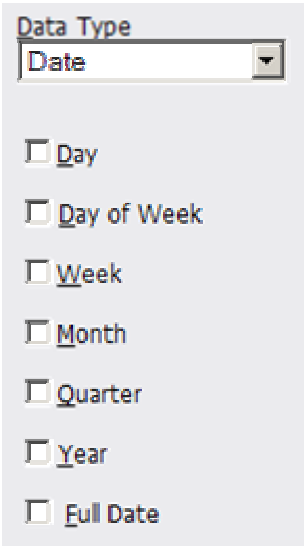
When you are defining your cube you get to nominate where dimensions will initially appear, either on the vertical or horizontal axis of your cube, outside of the main cube viewing area, or as invisible elements that preclude visual manipulation.





The data type for the dimension will be inherited from your dictionary definition, but can be changed if necessary. It is also possible to include complex dimensions that don't appear on your database which are populated with a Viságe function as data for your cube is being extracted.

You can also derive additional dimensions from some data types. For example, most BIT cubes are likely to include a dimension for a date.



From a basic *date* dimension you can also derive the year, quarter, month, week, day of the week etc, rather than having to have dictionary items for each of these derived dimensions.

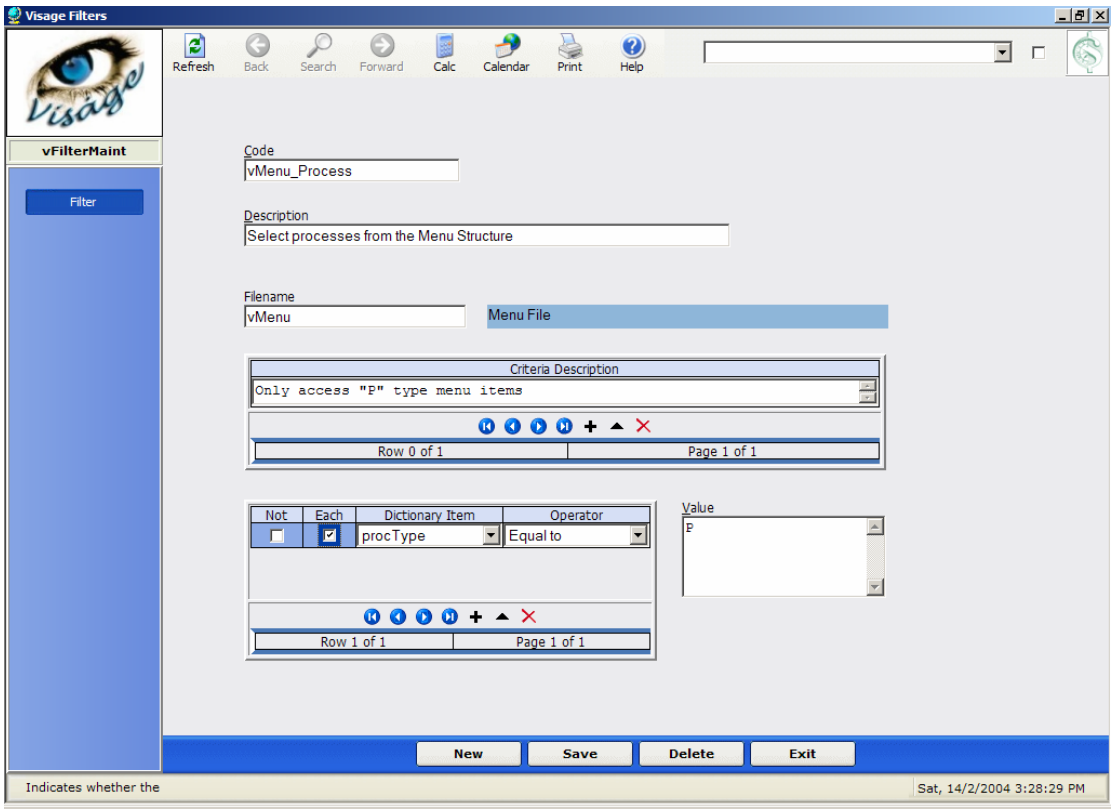
Likewise, you can extract hours, minutes and seconds from a *time* dimension.

Whilst a cube is notionally built from a single file, facts and dimensions can also be extracted from **any** field from a related file, using the information stored in the extended Viságe dictionary structure

If you can use a mouse, know how to *point, click, drag & drop* then you have the necessary skills to build a **Viságe.BIT** HiPerCube.

Populating the Viságe.BIT Cube

Once you have defined the structure of your cube, you still need to physically extract, transform and load (ETL) into the cube. Perhaps the easiest way to do this is to simply define a **filter** in Viságe., which is done using a graphical tool that lets you define selection criteria for a database **SELECT** statement



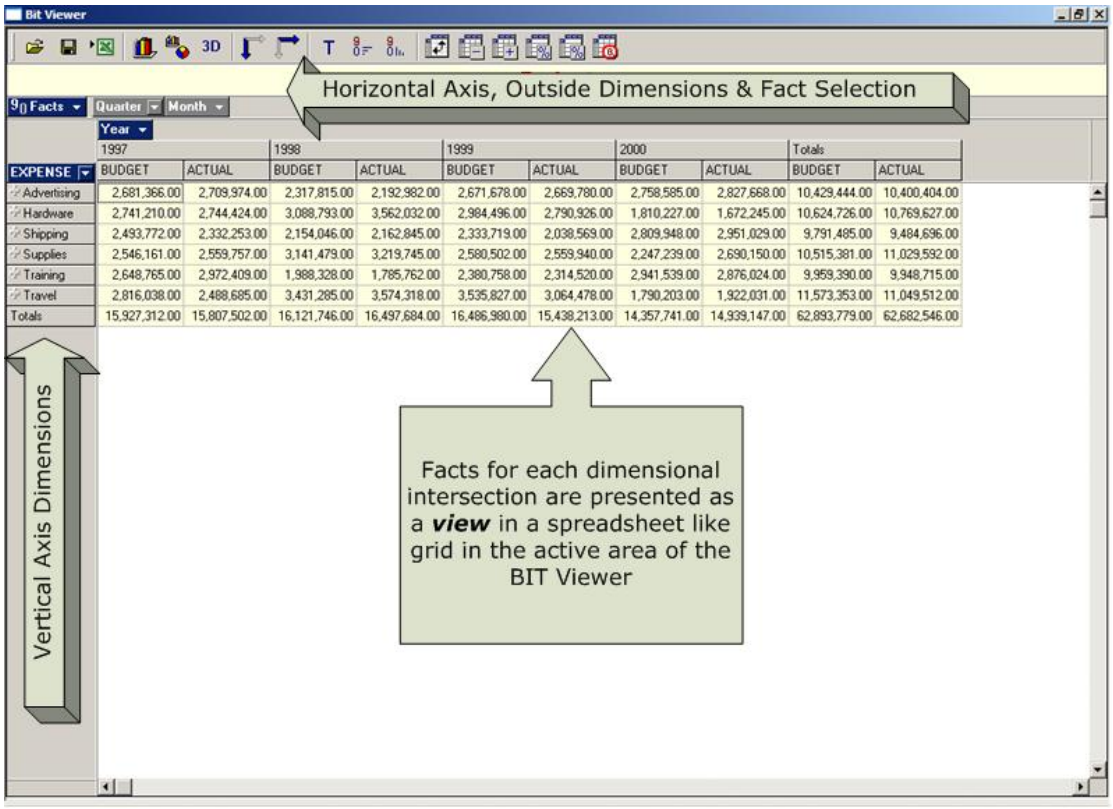
Alternatively you can simply run a Viságe function, which can execute your own custom host logic to return an *active select list* using indices, inversions files etc.

For maximum performance you can also use an API to perform your own customized data extractions. In this mode a host Basic subroutine can be written that identifies records that meet your selection criteria and passes a specially formatted multi-valued record to the API for extraction processing.

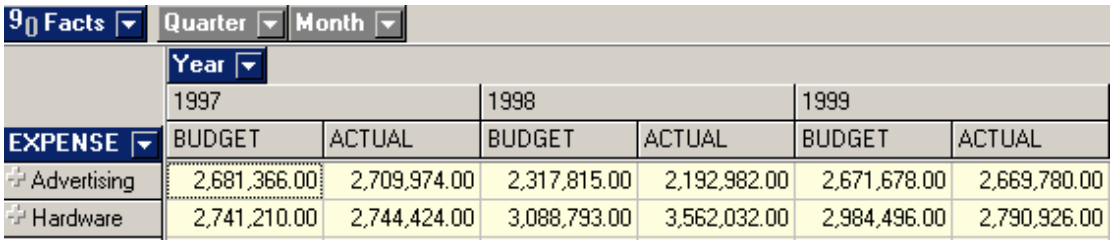
With your extraction method determined it is then a simple matter to set up a schedule for your cube extraction/build to be performed, and set up security to limit who has access to which cubes if necessary.

Using the Viságe.BIT Viewer

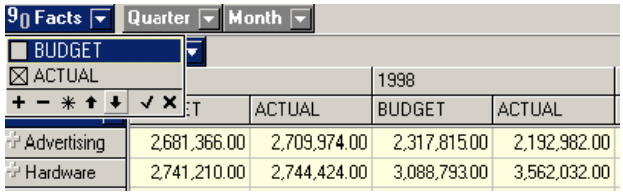
When you first access a **Viságe.BIT** cube you will be presented with a screen populated with **all** facts and dimensions in their initial starting points, as defined when you designed your cube.



One of the first things you may notice is that there is a drop down selection capability for all facts and dimensions, allowing you to dynamically filter what information is displayed



The image above is showing actual and budget figures. If we wanted to only show actuals, we would simply de-select ***BUDGET*** from our fact list.



Facts	Quarter	Month			
	Year				
EXPENSE	1997	1998	1999	2000	Totals
Advertising	2,709,974.00	2,192,982.00	2,669,780.00	2,827,668.00	10,400,404.00
Hardware	2,744,424.00	3,562,032.00	2,790,926.00	1,672,245.00	10,769,627.00

Notice that the ***Facts*** heading is now in italics, giving you visual feedback that there is filtering in place.

You may also note the **+** sign in the expense dimension, which indicates that there are other dimensions in the vertical axis which are currently hidden. To *drill down* into this additional detail, simply click on the **+** sign in the row you want to explode.

Facts	Quarter	Month				
		Year				
EXPENSE	DEPARTMENT	1997	1998	1999	2000	Totals
Advertising	Accounting	609,011.00	426,485.00	985,703.00	571,328.00	2,592,527.00
	Admin	551,674.00	697,202.00	947,596.00	240,703.00	2,437,175.00
	Marketing	591,434.00	589,601.00	225,479.00	915,291.00	2,321,805.00
	Sales	957,855.00	479,694.00	511,002.00	1,100,346.00	3,048,897.00
	Totals	2,709,974.00	2,192,982.00	2,669,780.00	2,827,668.00	10,400,404.00
Hardware		2,744,424.00	3,562,032.00	2,790,926.00	1,672,245.00	10,769,627.00
Shipping		2,332,253.00	2,162,845.00	2,038,569.00	2,951,029.00	9,484,696.00

You can also click on this button on the toolbar, which will explode **all** hidden dimensions.

This button will hide all dimensions except the outermost in the horizontal and vertical axes.

You can also perform drill downs, or change you view of the BIT Cube by dragging and dropping a dimension from one axis to the other, or by dragging a dimension from the outer area onto the horizontal or vertical axis.

Facts	Month					
		Year	Quarter			
		1997				
EXPENSE	DEPARTMENT	Quarter 1	Quarter 2	Quarter 3	Quarter 4	Totals
Advertising	Accounting	134,069.00	142,231.00	332,711.00		609,011.00
	Admin	68,834.00	151,696.00	84,325.00	246,819.00	551,674.00
	Marketing		157,726.00	118,674.00	315,034.00	591,434.00
	Sales	99,584.00	399,762.00	158,034.00	300,475.00	957,855.00
	Totals	302,487.00	851,415.00	693,744.00	862,328.00	2,709,974.00
Hardware	Accounting	367,168.00	213,737.00	170,867.00	21,456.00	773,228.00
	Admin	315,161.00	64,994.00	279,224.00	360,707.00	1,020,086.00
	Marketing		166,675.00	115,802.00	17,204.00	299,681.00
	Sales	300,189.00	135,826.00	118,577.00	96,837.00	651,429.00
	Totals	982,518.00	581,232.00	684,470.00	496,204.00	2,744,424.00

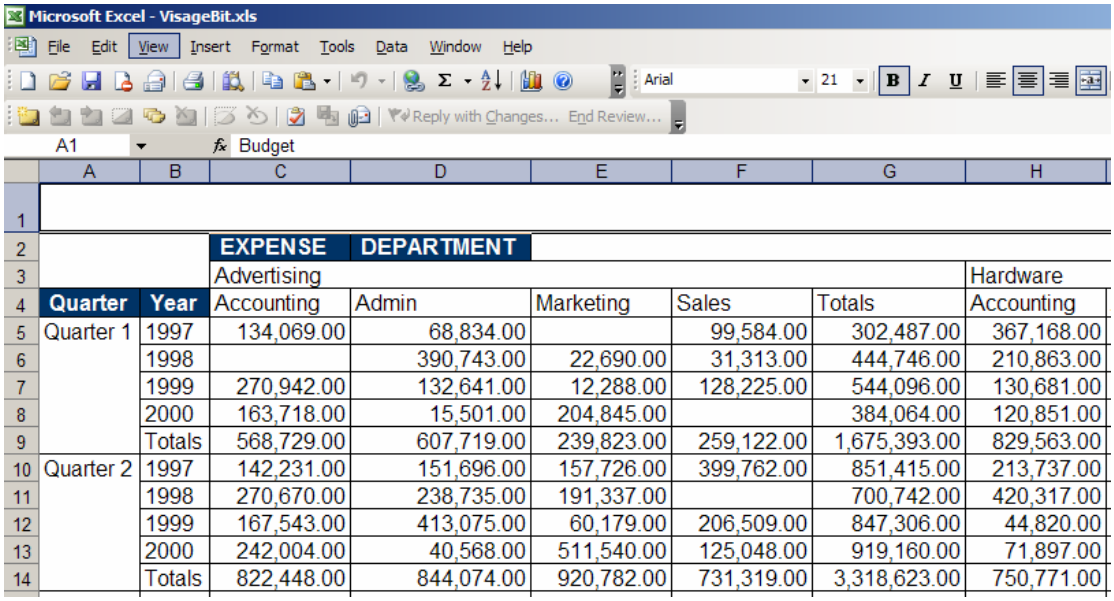
The order that dimensions appear on an axis impacts your view of the BIT Cube. For example, to compare quarters across multiple years, simply drag the **Quarter** dimension and drop it *before* the **Year** dimension on the horizontal axis.

90 Facts / Month						
		Quarter / Year				
		Quarter 1				
EXPENSE	DEPARTMENT	1997	1998	1999	2000	Totals
Advertising	Accounting	134,069.00		270,942.00	163,718.00	568,729.00
	Admin	68,834.00	390,743.00	132,641.00	15,501.00	607,719.00
	Marketing		22,690.00	12,288.00	204,845.00	239,823.00
	Sales	99,584.00	31,313.00	128,225.00		259,122.00
	Totals	302,487.00	444,746.00	544,096.00	384,064.00	1,675,393.00
Hardware	Accounting	367,168.00	210,863.00	130,681.00	120,851.00	829,563.00
	Admin	315,161.00	260,591.00	227,484.00	93,548.00	896,784.00
	Marketing		23,969.00		290,063.00	314,032.00
	Sales	300,189.00		444,077.00		744,266.00
	Totals	982,518.00	495,423.00	802,242.00	504,462.00	2,784,645.00

You can also pivot the axes by using the  button on the toolbar.

90 Facts / Month							
		EXPENSE / DEPARTMENT					
		Advertising					Hardware
Quarter	Year	Accounting	Admin	Marketing	Sales	Totals	Accounting
Quarter 1	1997	134,069.00	68,834.00		99,584.00	302,487.00	367,168.00
	1998		390,743.00	22,690.00	31,313.00	444,746.00	210,863.00
	1999	270,942.00	132,641.00	12,288.00	128,225.00	544,096.00	130,681.00
	2000	163,718.00	15,501.00	204,845.00		384,064.00	120,851.00
	Totals	568,729.00	607,719.00	239,823.00	259,122.00	1,675,393.00	829,563.00
Quarter 2	1997	142,231.00	151,696.00	157,726.00	399,762.00	851,415.00	213,737.00

If you discover something that you want to share with other parties who don't have access to **Viságe.BIT,** you can export the view to Excel with just one click of the  toolbar button.

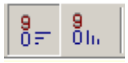


	A	B	C	D	E	F	G	H
1								
2			EXPENSE	DEPARTMENT				
3			Advertising					Hardware
4	Quarter	Year	Accounting	Admin	Marketing	Sales	Totals	Accounting
5	Quarter 1	1997	134,069.00	68,834.00		99,584.00	302,487.00	367,168.00
6		1998		390,743.00	22,690.00	31,313.00	444,746.00	210,863.00
7		1999	270,942.00	132,641.00	12,288.00	128,225.00	544,096.00	130,681.00
8		2000	163,718.00	15,501.00	204,845.00		384,064.00	120,851.00
9		Totals	568,729.00	607,719.00	239,823.00	259,122.00	1,675,393.00	829,563.00
10	Quarter 2	1997	142,231.00	151,696.00	157,726.00	399,762.00	851,415.00	213,737.00
11		1998	270,670.00	238,735.00	191,337.00		700,742.00	420,317.00
12		1999	167,543.00	413,075.00	60,179.00	206,509.00	847,306.00	44,820.00
13		2000	242,004.00	40,568.00	511,540.00	125,048.00	919,160.00	71,897.00
14		Totals	822,448.00	844,074.00	920,782.00	731,319.00	3,318,623.00	750,771.00

It has been said that a picture is worth a thousand words, and these buttons allow you to show a single fact in a variety of chart styles, optionally with 3D effects. If there are multiple dimensions on an axis, only the outer most is shown on the graph.

Another way you can gain insight to your columns of figures is to have them expressed as percentages, either across rows or down columns as appropriate, by using  the toolbar button.

Facts	Month	DEPARTMENT			
	Quarter	Year			
	Quarter 1				
EXPENSE	1997	1998	1999	2000	Totals
Advertising	7.34%	11.68%	13.35%	9.27%	10.38%
Hardware	23.84%	13.01%	19.69%	12.18%	17.25%
Shipping	13.77%	14.74%	17.38%	24.81%	17.74%
Supplies	20.30%	27.21%	12.94%	15.92%	18.95%
Training	12.09%	9.03%	8.48%	20.87%	12.71%
Travel	22.67%	24.32%	28.16%	16.94%	22.98%
Totals	100.00%	100.00%	100.00%	100.00%	100.00%

You can also have **Viságe.BIT** sort your cube values either across rows or down columns by first clicking in a cell in the desired row or column, and then clicking one of these  buttons.

Facts	Month	DEPARTMENT			
	Quarter	Year			
	Quarter 1				
EXPENSE	1997	1998	1999	2000	Totals
Advertising	302,487.00	444,746.00	544,096.00	384,064.00	1,675,393.00
Hardware	982,518.00	495,423.00	802,242.00	504,462.00	2,784,645.00
Supplies	836,575.00	1,035,750.00	527,265.00	659,577.00	3,059,167.00
Travel	934,458.00	925,947.00	1,147,571.00	701,894.00	3,709,870.00
Training	498,157.00	343,957.00	345,769.00	864,739.00	2,052,622.00
Shipping	567,672.00	561,263.00	708,178.00	1,027,973.00	2,865,086.00
Totals	4,121,867.00	3,807,086.00	4,075,121.00	4,142,709.00	16,146,783.00

Here we have sorted the 2000 Q1 column in $ terms, and % below

Facts	Month	DEPARTMENT			
	Quarter	Year			
	Quarter 1				
EXPENSE	1997	1998	1999	2000	Totals
Advertising	7.34%	11.68%	13.35%	9.27%	10.38%
Hardware	23.84%	13.01%	19.69%	12.18%	17.25%
Supplies	20.30%	27.21%	12.94%	15.92%	18.95%
Travel	22.67%	24.32%	28.16%	16.94%	22.98%
Training	12.09%	9.03%	8.48%	20.87%	12.71%
Shipping	13.77%	14.74%	17.38%	24.81%	17.74%
Totals	100.00%	100.00%	100.00%	100.00%	100.00%

As you can see, **Viságe.BIT** provides an impressive range of features to easily visualize and manipulate information.

Rather than having to wait weeks for a static report to be written to sort information in a specific order by your IT resource, you can do it yourself in seconds in an easy to use, drag & drop environment.

Because BIT uses MOLAP technology, you don't have to pre-define your cube views. When you discover an anomaly that warrants further investigation, you can simply adjust your dimensions to provide the **insider** information you need.

So, there is probably only 1 question that needs to be answered

How will **YOU** react when you install **Viságe.BIT** in your organization ?